Word Sense Disambiguation: Supervised Program Interpretation Methodology for Punjabi Language

Himdweep Walia[1], Ajay Rana[2], Vineet Kansal[3]

[1,2]Amity University, Noida, Uttar Pradesh, India
[3]IET, Lucknow, Uttar Pradesh, India
[1]himdweep@yahoo.com, [2]ajay_rana@amity.edu, [3]vineetkansal@yahoo.com

Abstract: **Word Sense Disambiguation (WSD) is the capability of finding the right interpretation of the given word in the given context through computation. Punjabi is among one of the 10 most widely spoken languages which is also morphologically rich but surprisingly, not much work has been done for computerization and development of lexical resources of this language. It is therefore motivating to develop a corpus of Punjabi language that will convey the correct sense of an ambiguous word. The availability of sense tagged corpora largely contributes in WSD and some of the most accurate WSD systems use supervised learning algorithms (like Naïve Bayes, k-NN and Decision Trees classifiers) to learn contextual rules or classification models automatically from sense-annotated examples. These algorithms have shown high accuracy in WSD and we are discussing these three supervised techniques, their algorithm, implementation and result when applied on Punjabi Corpora.**

*Keywords:***Word Sense Disambiguation, Supervised Approach, Naïve Bayes, k-Nearest Neighbor, Decision Tree, Sense-tagged corpora**

I. INTRODUCTION

The idea of communicating with machines has led to the study of Natural Language Processing (NLP). NLP works to make the machines understand the language just as humans do. And as is evident, a language contains many ambiguous words and the study of this is essential to make a machine understand any language better. This comes under Word Sense Disambiguation (WSD) which has in itself evolved as a significant topic under NLP. The primary reason being that WSD becomes the basis for the various applications like, machine translation, question answer, natural language understanding, natural language generation, speech recognition, to name a few under the blanket of Artificial Intelligence. By "ambiguous" we mean a word having more than one interpretation. We have spectrum of ambiguous words in all the Natural Languages which are easily distinguishable by a human from the context in which that word is being referred to, but this may not be the case when it comes to machines. To understand this concept better, let us consider the following two sentences:

A. The yogi is praying near the bank.

B. The cashier is standing near the bank.

In the above given examples, our ambiguous word is "bank" and with its usage in context with the surrounding words it is evident that it is referring to two entirely different senses. In the first sentence it is being used in reference with river bank and in the second sentence it implies a financial institution from where people and businesses can invest or borrow money.

Mallery [1988] has described WSD as an AI-complete problem, analogous to NP-Complete problem as discussed in complexity theory, thereby articulating the difficulty equivalence. This acknowledged difficulty has arisen due to number of factors like for example, finding the meaning of the word through the context in which it is being used. For this we may rely on the words surrounding the target word and it may be possible that the surrounding words are not explanatory enough to deduce the meaning.

There are two variants of WSD task – lexical sample task and all-words task. In lexical sample task, we pre-select a small set of target words and prepare an inventory of senses for each word. In case of all-words task, every word is an entire text and the sense of every word is known.

The WSD can be worked upon with respect to three techniques, namely, knowledge-based, machine learning and hybrid. In case of knowledge-based technique, we make use of dictionary or thesauri to locate the correct sense of the ambiguous word. The machine-based technique is sub-categorized into supervised, semi-supervised and unsupervised. For supervised techniques, we require a sense-tagged corpora which is not a pre-requisite in case of unsupervised and semi-supervised, which falls in betweenthe two. The hybrid technique is the amalgamation of all the above discussed techniques.

The rest of the paper has been divided in the following manner: Section II explains the different supervised techniques. Section III discusses the related work in WSD which has been done in various languages using these three supervised techniques. Section IV discusses the methodology that has been used for finding the correct sense of the ambiguous word with regard to these techniques. The next section discusses the observations and results and the final section concludes the paper.

978-1-5386-4692-2/18/$31.00 ©2018 IEEE

II. SUPERVISED TECHNIQUES

The machine learning approach works with the concept of training the system with the known results and then using the same knowledge on test cases to bring out the inferences.In this approach, the machine learns about the features of the word and using that assigns sense to the unseen words. We start with the target word which is to be disambiguated and also pick out the sentence which explains its context. This process is referred to as part-of-speech (POS) tagging which helps in finding the relation between adjacent words. The ambiguous words that we are using gives us the features. This feature value is the occurrence of the words around the target word. The machine learning approach has been further classified into three, namely, supervised, semi-supervised and unsupervised.

All the sub-classification approaches under machine learning require a labelled corpus for training the data. The system once trained is then applied on the test cases to device a mathematical model. The supervised methods assume that through context alone we can gather enough information to deduce the correct meaning of the ambiguous word and therefore reasoning is not required. We have to train a classifier for each word and this trained classifier then can tag words in the new text. Therefore for implementing any of the supervised techniques, we require:

1) tag set i.e. sense inventory

2) training corpus

3) set of features extracted from the training corpus classifier

4) In this paper, we will be discussing three of the supervised methods:

A. Naïve Bayesian Classifier Method

The Naïve Bayesian Classifier is a popular supervised approach as it converges the data quickly thereby yielding better results in WSD problems. This classifier has been designed based on Baye's Theorem. It uses statistical methods and is used to determine probabilistic parameters for detecting disambiguation. Baye's Theorem calculates joint probability of each sense, say S_i, for the given word W over the features defined (S_1, S_2, …., S_n) in the given context. The sense having the highest joint probability is chosen as the correct sense of the word. The classifier is implemented on a trained annotated corpora.

k-NN Method

k-NN is another popular supervised approachand is also considered to be one of the best options as it does not forget exceptions. In this method, we store all the examples in memory at the time of training and every new test case is stored with respect to its k value from its nearest neighbor. When we have to find the set of nearest neighbors, then we compare the stored examples E = (E_1, …., E_n) with the new sets E_i = (E_{i1}, …., E_{in}), and the distance between them is calculated using any basic metric like Euclidian Distance.

C. Decision Tree Method

Decision Tree Method is again one of the promising methods for WSD. In this case we use selective rules related with every word sense. The system selects one or more rules which predicts the closest sense of the given ambiguous word based on the selected feature. The decision tree classifier is a word specific classifier and therefore we need to train a separate classifier for each word. Here we use the concept of 'yes' and 'no' rules where the exception conditions form the root node of the tree, having high weight, and the general conditions appear at the bottom, having low weight. By default the tree accepts all the remaining conditions. Then using a scoring function we calculate the weight which forms an association between the condition and the particular sense of the given ambiguous word which is estimated from the labelled corpora.

III. LITERATURE REVIEW

An extensive amount of work for WSD [2, 3]has been done in various languages, predominantly in English and European languages like German and French. In recent years, noteworthy research has been done in Asian languages like Japanese and Chinese as well. Coming to Indian regional languages, like Hindi, Bengali, Punjabi, the main emphasis has been on machine translation and preparation of machine readable dictionarieswhich can be credited to the fact that sense-tagged corpora is not available for these languages [4, 5]. For this paper, our language of focus is Punjabi [6, 7], spoken in the state of Punjab and some parts of its neighbouring states.

A. Naïve Bayes Classifier Method

This classifier is one of the most popular method used by researchers to find the correct context of the given ambiguous word. In the paper [18] by N. T. T. Aung, et. al., they used F-score, calculated using the precision P and recall R to get an accuracy of 90%. Work has been done in Hindi language [19] and in their paper, S. Singh and T. Siddiqui used Bayesian Classifier on Hindi Language. Features like nouns, pronouns, prepositions, totaling to 11 different categories were identified and then sense-coded with 60 polysemous Hindi nouns in the Hindi corpora.Through their experimentation, they calculated a precision of 77.52% on unordered list and 86.11% on ordered list.

The approach followed in the paper of H. Walia, A. Rana, V. Kansal [20] has been used in this comparative study. In this paper, the findings of the work done on Punjabi language have been discussed. The authors sense-tagged the corpora with 100 Punjabi nouns. Three words were taken for experimentation and were subjected to two different window sizes to judge how it would impact in reaching the correct conclusion. It was observed that by increasing the size of the context window, the accuracy with which the correct context is found, increases.

B. k-NN Method

k-NN Method has been extensively used in WSD for various languages like English and in IndianRegional Languages, like Hindi and Bengali.

In the paper by A. R. Rezapour, et. al., [22]two sets of features have been extracted, frequently occurred words with the ambiguous word and surrounding words with the ambiguous word. In their paper, R. Pandit, et. al., [21], have calculated the distance between the two vectors by using the overlap metric, which helped them to get an accuracy of 71%.

The algorithm applied in the paper of H. Walia, A. Rana, V. Kansal [23] has been used in this comparative study. In the paper, the algorithm was applied on 120 testset sentences with 8 ambiguous words. A pair of testset (60 each) were being used, one consisting of frequent words and the other consisting of surrounding words. The 5 fold cross-validation was used to estimate the performance of the algorithm. It was observed that slightly better results were seen in case of surrounding words than frequently used words alongside the ambiguous word.

C. Decision Tree Method

This method has the primary advantage of being able to screen the feature extraction. This methodology has been implemented on two Indian Regional languages, namely Assamese [24] and Manipuri [25]. In their paper, J. Sarmah, et. al., [24], the authors discussed C4.5 decision tree and implemented it in Java using information gain ratio to determine the splitting attribute. The experiment calculates the F-measure equal to 0.611 after performing a 10-fold cross validation evaluationon 10 Assamese ambiguous words. The second paper by R. Singh, et. al., [25] discusses the architecture applied on Manipuri Language. In their architecture they have used the window size of 5 which contains the ambiguous word along with the surrounding words to detect the closest meaning of the focus word from this context information.

The approach followed in this paper for the comparative analysis has been taken from the paper by H. Walia, A. Rana, V. Kansal [26]. This paperuses the supervised approach – decision tree with cross validation evaluation on Punjabi language.

IV. METHODOLOGY

The Punjabi Corpora [9] has been obtained from Evaluations and Language Resources Distribution Agency, Paris, France. The Punjabi WordNet [8] has 23255 nouns, 2836 verbs, 5830 adjectives and 443 adverbs. Out of this 100 ambiguous words were taken from the WordNet. The Punjabi Corpora obtained contains more than 5000 ambiguous words. This corpora was then sense-tagged with 100 ambiguous words for our study.

The three techniques that we are discussing in this paper, undergoes a common step and thereafter every technique's algorithm is applied on the corpus to get the results.

In pre-processing stage, we remove the noisy data (i.e. spelling mistakes, blanks, punctuation marks and unidentified symbols) from the corpus, followed by stop words (i.e. prepositions, pronouns, conjunctions, etc.). In a study conducted by J. Kaur, et.al. [11], a list of 184 stop words for Punjabi language has been released for public usage. Finally we perform stemming where we reduce the target word to its base form which will consequently make it easier for us to disambiguate.

A. Naïve Bayes Classifier Method

For this comparative study, we are using the approach discussed in paper [20]. We take a sentence from the corpora [9] and perform preprocessing on it i.e. removing all stop words and consequently converting it into "bag of words". For our comparative study, we have chosen the size of context window as 3, 5 and 7. In the referenced paper, two sizes were used, 5 and 7, to deduce the correct context of the ambiguous word. It was observed that larger window size yielded better results. But as we are doing a comparative study with different supervised techniques, so different windows sizes are considered to calculate the precision of the results obtained so. Keeping this observation in mind, the above decision was taken.

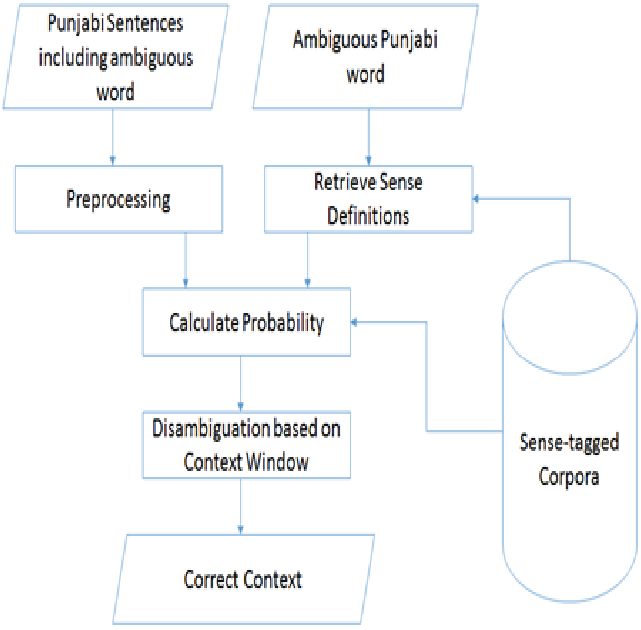

Punjabi Sentences including ambiguous word
Ambiguous Punjabi word
Preprocessing
Retrieve Sense Definitions
Calculate Probability
Sense-tagged Corpora
Disambiguation based on Context Window
Correct Context

Fig. 1. Naïve Bayes approach flowchart

Using the Bayes Theorem, we calculate the prior probability and then finally we apply the Bayes decision rule in order to perform the disambiguation process. This decision rule works on probability where we calculate the closeness of the given ambiguous word with respect to the sense it is closest to. If the calculated value is on the higher side then we conclude that we are more close to the correct sense of the ambiguous word being looked into. The approach followed is illustrated through the flowchart in Fig 1.

B. k-NN Method

For this comparative study, we are using the approach discussed in paper [23].

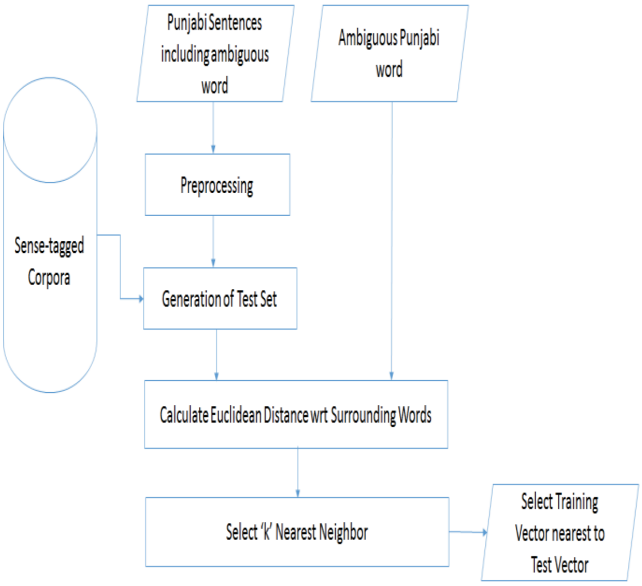

Punjabi Sentences including ambiguous word
Ambiguous Punjabi word
Sense-tagged Corpora
Preprocessing
Generation of Test Set
Calculate Euclidean Distance wrt Surrounding Words
Select 'k' Nearest Neighbor
Select Training Vector nearest to Test Vector

Fig. 2. k-NN Method approach flowchart

The proposed methodology includes two steps. In the first step, we perform the feature extraction process and converts the given paragraph of the corpus [9] into a vector. Here we are only using the list of surrounding words which appear with the ambiguous word. The primary reason being that in the referred paper, the surrounding words have shown better results as compared to frequently used words alongside the ambiguous word. In the second step we pass these vectors through the k-NN classifier. The approach followed is illustrated through the flowchart in Fig 2.

C. Decision Tree Method

For this comparative study, we are using the approach discussed in paper [26]. The first step is procurement of data which is then preprocessed i.e. stop words like prepositions are removed and the given sentence is thus reduced into "bag of words". The corpora was then manually sense tagged with 100 ambiguous words. Then the attributes were selected with the range {-2, -1, 0, +1, +2}. The features were then fed to the classifier and the algorithm identifies pattern and infer predictions from them. With the help of Gain Ratio the splitting attribute is determined. The values of the splitting attribute are the outcome of the test splitting attribute. Subtracting the splitting attribute we get the remaining attributes and are portioned accordingly. And then again the splitting attribute is derived from the each partition table and values of the test attribute helps to reach the sense class label. The values were discrete-values in our case. The process of creation of the tree is terminated when all the tuples belong to the same class label for the attribute values. The approach followed is illustrated through the flowchart in Fig 3.

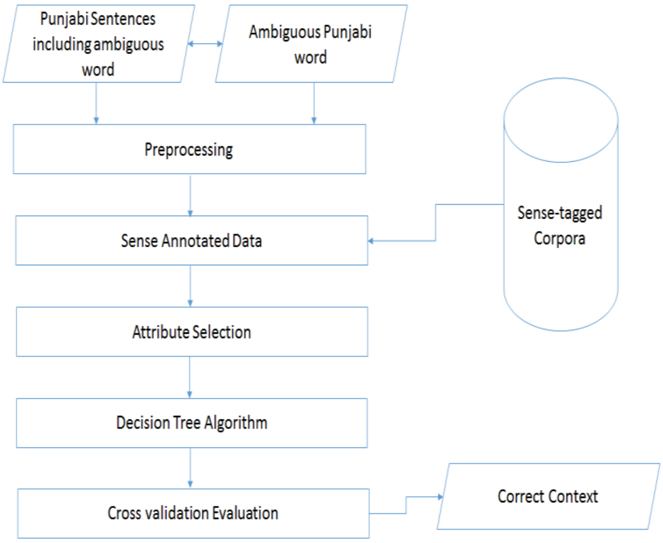


Fig. 3. Decision Tree approach flowchart

V. OBSERVATION AND INTERPRETATION

TABLE 1: Dictionary meaning and the extracted meaning of the ambiguous word

Ambiguous Word	English Transcription	Number of senses in dictionary	Number of senses in the extracted samples
ਉੱਤਰ	uttar	5	3
ਕੱਚਾ	kacha	4	2
ਹਾਰ	haar	4	3
ਉਲਟ	ulta	4	3
ਚਾਰ	chaar	4	2

In order to give a comparative view of the three supervised techniques, we selected 20 ambiguous words (5 of them have been listed in Table 1 for reference) and used the corpora[9] to show the effectiveness.

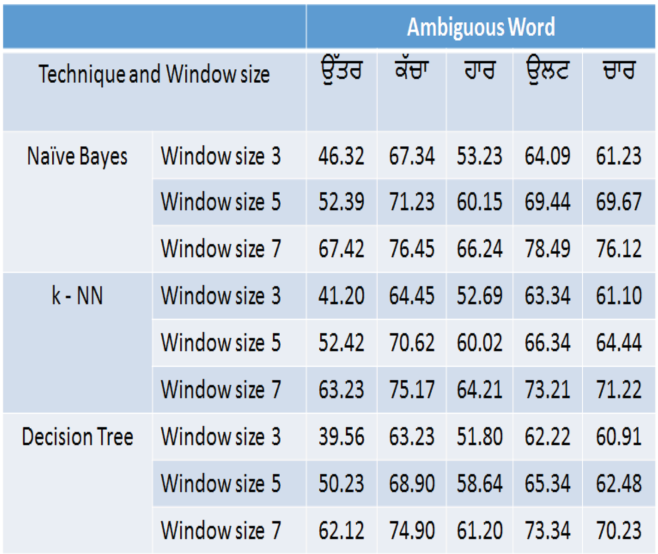
We have worked with three different window context sizes [17], i.e. 3, 5 and 7 for measuring the performance of each of the techniques. The precision is measured as:

P (precision) = number of correct results obtained / number of total results.

We then used the 5-fold cross validation evaluation procedure where the data was divided into training and testing sets such that every training set is the test set atleast once.

The Table 2 illustrates the results obtained by varying the window size i.e. 3, 5 and 7 and then calculating the precision for the given ambiguous word with respect to the three supervised techniques – Naïve Bayes, k-NN, and Decision tree – being used.

TABLE 2: Precision of ambiguous word with different window size for different techniques



Technique and Window size		Ambiguous Word				
		ਉੱਤਰ	ਕੱਚਾ	ਹਾਰ	ਉਲਟ	ਚਾਰ
Naïve Bayes	Window size 3	46.32	67.34	53.23	64.09	61.23
	Window size 5	52.39	71.23	60.15	69.44	69.67
	Window size 7	67.42	76.45	66.24	78.49	76.12
k - NN	Window size 3	41.20	64.45	52.69	63.34	61.10
	Window size 5	52.42	70.62	60.02	66.34	64.44
	Window size 7	63.23	75.17	64.21	73.21	71.22
Decision Tree	Window size 3	39.56	63.23	51.80	62.22	60.91
	Window size 5	50.23	68.90	58.64	65.34	62.48
	Window size 7	62.12	74.90	61.20	73.34	70.23

The results in Table 2 clearly show that we obtained better precision for a larger window size. It also shows that of the three supervised techniques, Naïve Bayes demonstrates a better performance than the other two techniques. The primary reason being that Naïve Bayes can show even good results when database is comparatively small.

One suggestion is that we can increase the database size by using deep learning technology and therefore the results will reflect a better comparison.

VI. CONCLUSION

In this paper, we have illustrated the results of three supervised approaches, namely, Naïve Bayes, k-NN and Decision Tree for Punjabi language. The Naïve Bayes classifier is super simple and converges the data quickly so we can work even if we have less training data. The strength of k-NN lies on the fact that it is robust to noisy training data and is effective if the training data is large. The Decision Tree subliminally performs variable screening or feature selection. Using these advantages of these classifiers, we have prepared this comparative study for the Punjabi Word Sense Disambiguation.

A classifier algorithm learns from a training sample made up of database associated with their sense labels. Various challenges like sense-inventory along with their senses were discovered. Also the sense-annotated data as a training sample was manually prepared. K-fold cross validation evaluation was performed on the dataset.

Punjabi is a less computationally aware language and WSD task using a supervised approach – Naive Bayes, k-NN and Decision Tree - with cross validation evaluation is the first initiative towards Punjabi Language. This will provide a helpful contribution to Natural Language Processing.

ACKNOWLEDGEMENTS

The author[1] acknowledges Amity University, Uttar Pradesh, where the author is registered as scholar and also for providing with the library facility and also an extensive online resources for research.

REFERENCES

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

[2] J. Kaur, "Word Sense Disambiguation (WSD)", International Journal for Technological Research In Engineering, Volume 1, Issue 5, January2014.

[3] R. Kumar, R. Khanna, and V. Goyal, "A Review of Literature on Word Sense Disambiguation", International Journal of Engineering Sciences, Vol. 6, July2012.

[4] 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.

[5] H. Walia, A. Rana, and V. Kansal, "A study on different Word Sense Disambiguation Approaches and their application on Indian Regional Languages", Proceedings of International Conference on Technology and Trust (ICTT'17), 28-29 December, 2017.

[6] R. Kumar, and R. Khanna, "Natural Language Engineering: The Study of Word Sense Disambiguation in Punjabi", IJES, July2011.

[7] 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.

[8] Punjabi WordNet. [Online]. Available: http://tdil-dc.in/indowordnet/index.jsp

[9] Punjabi Corpora. Obtained from Evaluations and Language Resources Distribution Agency, Paris, France.

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

[11] J. Kaur, J. R. Saini, "Punjabi Stop Words: A Gurmukhi, Shahmukhi and Roman Scripted Chronical", Proceedings of the ACM Symposium on Women in research, March 2016, Page 32-37.

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

[13] 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.

[14] 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.

[15] 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.

[16] 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.

[17] 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.

[18] N. T. T. Aung, K. M. Soe, and N. l. Thein, “A Word Sense Disambiguation System using Naïve Bayesian Algorithm for Myanmar Language”, International Journal of Scientific & Engineering Research, volume 2, Issue 9, September 2011.

[19] S. Singh, T. J. Siddiqui, S. K. Sharma, “Naïve Bayes classifier for Hindi Word Sense Disambiguation”, Proceedings of 7th ACM India Compute Conference (Compute’14), Nagpur, India, 9 – 11 October, 2014, Article No. 1, ACM Digital Library.

[20] H. Walia, A. Rana, and V. Kansal, “A Naïve Bayes Approach for working on Gurmukhi Word Sense Disambiguation”, 6th international Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO), 2017, IEEE Explorer.

[21] 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.

[22] 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.

[23] H. Walia, A. Rana, and V. Kansal, “A Supervised Approach on Gurmukhi word Sense Disambiguation using k-NN Method”, 6th international Conference on Cloud System and Big Data Engineering, 2018

[24] J. Sarmah and S.S.Sarma, “Decision Tree based Supervised Word Sense Disambiguation for Assamese”, International Journal of Computer Applications, Volume 141, No 1, May, 2016.

[25] R. L, Singh, K. Ghosh, K. Nongmeikapam and S. Bandyopadhyay, “A Decision Tree Based Word Sense Disambiguation System in Manipuri Language”, Advanced Computing: An International Journal, Vol 5, No 4, July 2014.

[26] H. Walia, A. Rana, and V. Kansal, “A Decision Tree based Supervised Program Interpretation Technique for Gurmukhi Language”.