Word Sense Disambiguation Techniques for Indian and other Asian Languages: A Survey

Mulkalapalli Srinivas
Research Scholar JNTUH
Associate Professor
Department of CSE, GCET

B. Padmaja Rani, PhD
Professor
Department of CSE
JNTUH, Hyderabad

ABSTRACT

Natural Languages used by people for establishing proper communication consist of many words having multiple meanings known as polysemous but implies a single sense depending on the context. Word sense disambiguation is a method of determining the appropriate sense of a polysemous word in the context. WSD is almost finished for English. It is a challenging task for Indian languages since these are morphologically rich in nature and development of various resources like machine readable dictionaries, WordNet etc. are in progress. We have discussed the unsupervised Graph based WSD for English. Then, we have discussed the various efforts accomplished by several researchers to develop WSD systems for Indian languages like Hindi, Kannada, Malayalam, and Assamese. Finally, we have discussed about WSD for other Asian languages like Nepali, Arabic and Myanmar.

General Terms

Natural Language Processing

Keywords

Knowledge based, Supervised and Unsupervised techniques, word sense disambiguation, Indian languages

1. INTRODUCTION

The creatures are using the language as their communication media. Through language information can be exchanged among the races. Verbal communication involves alphabets, words, sentences etc. In almost all natural languages, there are words having different meanings depending on the context. Those words are known as polysemous words making verbal communication ambiguous. Fortunately, human beings resolve the ambiguity instantly depending on the context with lot of ease. But, machines find it as a very difficult problem. This involves processing unstructured textual information to build the appropriate data structures. We determine the most appropriate meaning through analyzing the data structures thoroughly. This is known as Word Sense Disambiguation, a common problem in Natural Language Processing (NLP).

Word sense Disambiguation [1] is the process of identifying the correct sense of a word that has several meanings in the context in a computational paradigm. Machine translation is one of the most former on growing research topic in computational linguistics. The problem WSD is as complex as most of the difficult problems in Artificial Intelligence and hence it is deemed as an AI complete problem. In 1940's WSD was developed as discrete field in computational linguistics to help the research in machine translation. In 1950's Weaver identified that context is crucial and hence statistical semantic studies have been undertaken as a necessary primary step. The automatic disambiguation of word senses has been given an utmost priority from the earliest days of computer treatment of languages in the 1950's. WSD depends heavily on knowledge sources like dictionaries, thesauri, ontology's, collocations, WordNet etc. WSD can be described as a method of providing the most appropriate sense to all or some words in the text where T is a sequence of words (w_1, w_2...w_n). It is called as All-words WSD when it attempts to disambiguate all words in a text such as nouns verbs, adjectives, adverbs¸ etc. Otherwise Targeted WSD as it disambiguates some restricted words only. It consists of mainly discovering the mapping M from words to senses such that $M (k) \subseteq Senses_D(w_k)$ where M(k) is the subset of senses of w_k which are appropriate in the text T and $Senses_D(w_k)$ is the set of senses in dictionary D for word w_k. The mapping M can assign multiple senses to w_k belonging to T but finally the most appropriate sense is selected. Hence, WSD can be seen as a classification task where word senses form the classes and a method classifies each occurrence of the word to multiple classes by exploiting information available from the context and external knowledge sources such as dictionary, thesauri, ontology's, collocations, WordNet, unlabelled or annotated sense corpora.

The input text is preprocessed to build a structured format suitable for our WSD system. It consists of the following steps in sequence:

a) Tokenization - Dividing the text into basic units (tokens) called as words.
b) Part-of-Speech Tagging - Determining the appropriate grammatical category for each word.
c) Lemmatization - Performs morphological analysis to provide the root words.
d) Chunking - Partitioning the text in syntactically correlated parts.
e) Parsing- Provides the parse tree of sentence structure.

Following the above preprocessing, each word is represented as a feature vector making the assignment of the appropriate sense easy by the WSD system.

Word sense disambiguation is used in NLP applications like Machine Translation, Information retrieval, Document summarization, Question Answering Systems, and so on.

This paper is organized as follows: In section II description of Graph based WSD for English, presenting WSD current state of the art for various Indian languages and other Asian languages in sections III & IV respectively. Finally Conclusion in section V followed by the references.

2. UNSUPERVISED GRAPH BASED WSD FOR ENGLISH

The various acceptable solutions currently available for WSD are supervised and unsupervised methods. The former has good accuracy but requires extensive training through manually annotated data making them suffer from the knowledge acquisition bottleneck problem, while the latter take less time but accuracy is low.

But, recently unsupervised graph-based WSD techniques [2, 3, 4, 5, 6] succeeded in minimizing the accuracy gap from the supervised methods thereby gaining attention from researchers. This involves construction of a semantic graph for words to be disambiguated by taking into account nodes and semantic edges from word thesauri like WordNet. Then node ranking or node activation algorithms are used to determine the best candidate sense for each word.

In [3], authors have proposed an unsupervised graph-based method for WSD. The graph representation is used to model dependencies among word senses in text known as semantic graph. Six different measures of word semantic similarity known as the Leacock & Chodorow, the Lesk, the Wu-Palmer, the Resnik, the Lin and Jiang & Conrath are used to determine the dependency between word senses represented as nodes in the graph. Next, four graph-based centrality algorithms the indegree, closeness, betweenness and PageRank are used to assign scores to vertices of the graph. Finally, the node that has the highest value is assigned as the sense for the word. They achieved a precision of 61.22, 45.18 and 54.79 and recall of 60.45, 40.53 and 54.14 for nouns, verbs and adjectives respectively.

In [7], authors have explored several measures for analyzing the connectivity of semantic graph structures in local as well as global level. In local measures of centrality they selected degree, closeness and betweenness and their variants. In global measures of centrality they have considered compactness, graph entropy, and edge density. The use of global connectivity measures may make WSD as combinatorial problem which require lot of time. This may be avoided by using heuristic search methods like local search, simulated annealing, and genetic algorithms. Finally, they concluded that use of local measures results in better performance of WSD than using global measures.

In [8], authors experimentally investigated the performance of unsupervised graph-based methods involving construction of a semantic graph. They have selected four graph processing methods namely SAN, PageRank, HITS and P-Rank for evaluation. To obtain comparative evaluation, the same semantic representation is used for all methods. The performance is evaluated based on two criteria on Senseval. They are the accuracy, and the inter-agreement rate in the sense selection level.

3. INDIAN LANGUAGES

3.1 Kannada

In [9], authors have proposed an integrated Kannada word sense disambiguation system consisting of the following modules: corpus builder, sentence extractor, dictionary builder, Kannada shallow parser, word classifier, Kannada Target Word Sense Disambiguator (KTWSD), Kannada Verb Sense Disambiguator (KVSD), and Kannada Rule Based Word Sense Disambiguation (KRBWSD). The morphological information of each word and syntactic information of the sentence are obtained through Kannada shallow parser. Word classifier provides a list of polysemous words. KTWSD works based on Naive Bayes classifier using the compound words clues and syntactic features in a local context to disambiguate a word that appears in the target word list. They included KVSD based on their observation that more verbs are ambiguous. It uses the argument structure of verb for disambiguation. The correct sense is identified based on matching relevant cluster of arguments with the argument structure frame of the verb. KRBWSD resolves ambiguity by formulating set of syntactic and semantic rules. It can be considered as an initial attempt to WSD for Kannada language.

In [10], the decision list based all-word WSD is provided for Kannada language based on hypothesis that word implies one sense per collocation. The decision list is created using training corpora for each ambiguous word and it is ordered based on the log-likelihood correspondence between each context vector and each sense. When decision list fails in assigning the sense, the most frequent sense determined based on training data is the default sense. The results are encouraging and the authors opinioned that addressing the discourse level and compound words issues will definitely improve the performance of the system.

3.2 Malayalam

Malayalam is a Dravidian language spoken by around 36 million people in Kerala state in Southern India. In [11], Rosna P. Haroon and others have given the first attempt for an automatic WSD for Malayalam, which is a knowledge based approach. It consists of two approaches. First one is based on hand devised knowledge source and using Lesk and Walker algorithm. For each word 'w' to be disambiguated, they collected context words surrounding w that is denoted as 'c'. For each sense of w, the bag of words 'B' is collected. Measure the overlap between 'c' and 'B' and corresponding score is incremented by '1'. Finally, the sense associated with the maximum score is chosen as the winner or appropriate and returned. The second method is based on conceptual density measured through semantic relatedness using WordNet. The algorithm proceeds as follow. Obtain list of words for each sentence, ignore stop words, stemming is performed to have base words for remaining words. If the sentence has any ambiguous words, extract the nouns and save them. The correct sense of ambiguous word is one that is associated with the minimum depth (highest conceptual density) with nouns. When the sentence has multiple nouns, depth is taken as the sum of the depth of sense with each noun.

In [12], authors have developed supervised WSD system for Malayalam based on Naïve Bayes classifier. It consists of preprocessing module which takes the sentence as input and produces the list of root words as output. It consists of tokenizer, stop word remover and stemmer as components. Next, a list of ambiguous words is created through ambiguity checker using the corpus of ambiguous words. Nouns in the sentence are taken as feature vectors. Finally, conditional probability of different senses of an ambiguous word with respect to feature vector is calculated using the sense corpus applying Naïve Bayes classifier. Output the sense associated with the highest probability. They concluded that better corpus will greatly improve the efficiency of the system.

3.3 Assamese

Assamese language spoken by people of Assam belongs to Indo-Aryan language family.

In [13], authors have proposed a supervised WSD system based on decision tree. J48 a Java implementation of C4.5 decision tree algorithm using information gain ration determining the splitting attribute is used for implementation. The system consists of the modules: preprocessing raw data, sense inventory preparation, feature/attribute selection, preparing the decision tree. Preprocessing includes data cleaning, stop word removal, stemming and finally correction of inconsistent data. Sense inventory preparation involves identifying 160 ambiguous words in corpus as well as 100 ambiguous words in WordNet, and then 50k sentences are tagged with the appropriate sense manually. Local lexical features are extracted. Finally, Decision tree is created based on the features extracted in the previous step. Two types of evaluation procedures are performed. First, Hold out evaluation splits the sense-annotated data ensuring that each class is represented in both training set and test set. Second, k-fold cross validation to improve the performance. It results in average F-measure of 0.611 when 10-fold cross validation evaluation was performed on 10 Assamese ambiguous words.

In [14], authors have proposed WSD system based on Walker algorithm that uses the subject category or domain in determining the implied sense of nouns or adjectives only. They have prepared a modified WordNet text file for a sample of words that includes FEATURE defining the subject category or domain for each word. Its equivalent XML file is created which supports easy extraction of required component. Stop words are removed and root words are obtained for remaining words. In a given sentence if it has the ambiguous word. Now, the subject category of the ambiguous word is extracted from the XML file. Then, CONTEXTBAG that includes categories of all the context words is prepared. The category with the maximum matches in the CONTEXTBAG is determined. Then, sense corresponding to that category is output as answer by the system. For random sentences from the Internet, its precision and recall are 86.66 and 61.09 respectively. Authors have felt large context window will improve the performance of the system.

3.4 Hindi

In [15], authors have proposed the first WSD system for HINDI using the WordNet for nouns only. A polysemous word is assigned a sense using an algorithm consisting of following steps. First step consists of preparing the context bag from the set of words surrounding the polysemous word. Next, the semantic information of each sense is collected from the WordNet that includes Glosses, example sentences of Synonyms, Hypernyms, Hoponyms and Meronyms. Then measure the overlap between the context bag and semantic information. Finally, output the sense corresponding to maximum overlap. Accuracy of the system ranges from 40% to 70% for documents from various domains like Agriculture, Science& Sociology etc.

In [16], authors have proposed a Graph based algorithm for Hindi WSD. It uses a graph encoding the similarities identified among word senses. Applying centrality algorithms on the graph, it attempts to annotate to all words in a text. It proceeds as follow. A graph G= (V, E) is constructed for each target sentence, where node represents word senses and edge represents semantic relation. Final graph is obtained using DFS and Hindi WordNet. Graph connectivity measures such as distance function, local measure and global measure were used. For clustering, Hierarchical Agglomerative clustering algorithm was chosen. They tried to minimize the computing time using some useful assumptions. Finally, it is assigning a synset for each noun in the sample test. Accuracy of the WSD system is 65.17%.

In [17], authors have proposed a WSD system based on the Leacock- Chodorow semantic relatedness measure. It uses Hindi WordNet hierarchy to compute distance between the two concepts. The algorithm proceeds as follow. Consider a test instance that consists of an ambiguous word. Remove stop words from the test instance. Then it is represented as vector of words present in a window that includes two nouns on either side of the target word. The sense definitions are also represented as a vector of words present in the sense definitions. For each sense of the target word, semantic relatedness is computed for all senses of words other than target word in the vector of test instance. The overall score for each sense is obtained by summing the above values. The sense received the maximum score is considered as the winner sense. They evaluated their algorithm on a data set consisting of 20 Hindi polysemous nouns obtaining the average precision and recall as 60.65% and 57.11% respectively with an improvement of 32.53% over direct overlap measure.

In [18], authors have proposed a WSD for Hindi nouns based on mining association rules. The algorithm works on sentences. Sentences to be considered as input must have at least 5 words of which more than one word is ambiguous. Then produce the frequent item sets from the context data base. Generate the association rules X->Y from maximum frequent item sets satisfying the minimum threshold of the confidence degree. To determine the sense of an ambiguous word, first select the association rules according to its context words. Finally, the sense is determined by the rules committee voting. The average precision obtained is 72%.

WSD system based on word clusters obtained using Probabilistic Latent Semantic Analysis (PLSA) is proposed in [19]. This system has two phases: the training phase and the testing phase. The training phase creates the word clusters representing one sense through the following steps. Stop words are removed from the training data. Statistical stemmer is used for reduction of inflectional and derivational variants to their root words. PLSA incorporating Expectation Maximization (EM) is used to cluster similar words. They considered possibility of expanding the clusters using the semantic information like synonyms, homonyms, hypernyms and hyponyms in an attempt to improve the accuracy of the system. During the testing phase, the most appropriate sense of an ambiguous word, resent in the test corpus, is determined by computing the similarity score based on cosine coefficient of test corpus containing the target ambiguous word with each cluster generated during the training phase. The sense of the ambiguous word denoted by the cluster associated with highest similarity score is returned. This may be considered as relatively generic WSD since it is independent of languages. They have conducted experiments on English and Hindi achieving an accuracy of 83% and 74% respectively.

In [20], authors have proposed a WSD algorithms using Genetic Algorithm (GA) to disambiguate nouns in the Hindi text. Genetic Algorithms are best known for solving efficiently many NP hard optimization problems. The algorithm prepares a list of nouns that involves POS tagging and may also require morphological analysis to obtain the base form of nouns in the input sentence when they are not present in the Hindi WordNet (HWN). Then, available senses of each noun in the list are obtained from HWN which is input to the GA. They have tuned the GA to obtain better results through setting up the parameters like chromosome length as the number of nouns to be disambiguated together, population size as 30, cross-over probability as 0.8 to have more diversity that reduces local maxima and mutation probability as 0.03. The algorithm generated initial population and calculated fitness values of each chromosome. It repeatedly performs selection, crossover, mutation followed by evaluation until the termination condition is met. Elitism is incorporated to assure the best individuals in the next generation. Finally, the GA outputs the sense number to be used. They have experimented on a list of 12 nouns and achieved a recall of 91.6%.

In [21], authors have proposed an approach to disambiguate words in Hindi that consists of two phases- Training phase and Testing phase. The training phase involves processing of training documents as per the Hyperspace Analogue to Language (HAL) model to generate N X N HAL matrix where N is the total number of unique words in the documents. Reduced HAL matrix is obtained by removing the rows and columns corresponding to stop words that are not significant with respect to the disambiguation process. For each significant word HAL vector is obtained by normalizing the reduced HAL matrix. Fuzzy C-means clustering algorithm is used to create a set of clusters. Each cluster denotes the context in which an ambiguous word may occur. In the testing phase HAL vectors of significant words in the test data are obtained in the manner similar to the training phase. HAL vector of the ambiguous word is mapped to the HAL vector of the corresponding word obtained in the training phase to capture the similarity between them. Finally, target ambiguous word is disambiguated based on the Euclidian distance calculated between target word's HAL vector and the centers of the clusters generated during the training phase. The cluster with the minimum distance corresponds to the most related sense and is returned. This system achieved an accuracy of nearly 79.16% thereby outperforming all the previously developed approaches for Hindi WSD.

4. ASIAN LANGUAGES

4.1 Nepali

In [22], authors have proposed overlap based and conceptual distance combined with semantic graph based approaches to Nepali language. Overlap based approach consists of preprocessing phase that includes tokenizer, context selection after discarding stop words. Then for each sense of the target word prepared a collection of words from synsets, glosses of synsets, example sentences, hypernyms, glosses of hypernyms etc. Then the winner sense is determined based on maximum overlap between context of target word and collection of words gathered from WordNet for each sense of target word. Finally, the sense corresponding to maximum overlap is the winner sense. Conceptual distance based approach depends on the formula for calculating conceptual distance that is inversely proportional to the length of the path between two synsets in the WordNet graph and directly proportional to the depth of the two synsets in the WordNet hierarchy. Semantic Graph distance is the shortest path between two synsets in the WordNet graph where the edges can be any semantic relation. Here they used MODIFIES-NOUN). They evaluated the WSD system on a data set with 912 nouns and 751 adjectives. Overlap based approach results in accuracy of 54% for Nouns and 42% for Adjectives, where as conceptual distance combined with semantic graph distance resulted in 62% accuracy for Nouns and 58% accuracy for adjectives.

4.2 Myanmar

In [23], authors have proposed a WSD system for Myanmar language that uses Naïve Bayesian classifier to disambiguate ambiguous words with part-of-speech 'noun' and 'verb'. It includes Myanmar-English parallel corpus as training data. The algorithm proceeds as follow: the system accepts the Myanmar sentence having ambiguous words and the ambiguous word to be disambiguated as input. It is followed by preprocessing that consists of identifying words and removing stop words. Next, the system collects the possible English sense definitions of an ambiguous word from the corpus. System calculates the priori probability and the likelihood based on Bayes theorem. Finally, it consists of the disambiguation process that computes the score of each sense of ambiguous word in the test sentence using Bayes Decision rule. They evaluated the system for 60 ambiguous nouns and 100 ambiguous verbs which results in 89% Precision, 92% Recall and 90% F-Score.

4.3 Arabic

In [24], authors have evaluated the variants of the LESK algorithm to disambiguate Arabic words. In the first experiment, they have used original version of the LESK algorithm involving dictionary to obtain the possible definitions of words. The implied sense of ambiguous word is determined by calculating the overlap between each possible definition of ambiguous word and other context words in the sentence. In the second experiment, they modified the Lesk algorithm by taking into consideration five measures of similarity obtained from Arabic WordNet and corpus. They are as follow:

1. The semantic similarity measure of Wu and Palmer that is based on the distance between two nodes in the hierarchy and their position relative to the root. It is

$$\mathrm{consim}(C_1, C_2) = \frac{2*\mathrm{depth}\ (C)}{(\mathrm{depth}\ (C_1)+\mathrm{depth}\ (C_2)}$$

Where depth(C): Number of arcs that separate the root from C (the common subsequence between C_1 and C_2)

2. Information content introduced by Resnik and defined as follows

$$\mathrm{IC}_{\mathrm{res}}\ (C) = -\log P(C)$$

Where P(C) is the probability to find an instance of the concept C in the text.

The similarity measure of Resnik is

$$\mathrm{Sim}_{\mathrm{res}}\ (c_1\,, c_2) = \mathrm{IC}(\mathrm{lcs}(c_1\,, c_2))$$

where lcs represents the most specific concept that subsumes two concepts in the ontology.

3. The similarity measure defined by Jiang and Conrath is

$$\mathrm{Rel}_{\mathrm{jcn}}\ (c_1\,, c_2) = \frac{1}{\mathrm{IC}\,(c_1\,)+\mathrm{IC}\,(c_2\,)-2*\mathrm{IC}\,(\mathrm{lcs}\,(c_1\,,c_2))}$$

4. The similarity measure defined by Lin is

$$\mathrm{Rel}_{\mathrm{LIN}}\ (c_1\,, c_2) = \frac{2*\mathrm{IC}\,(\mathrm{lcs}\,(c_1\,,c_2))}{\mathrm{IC}\,(c_1\,)+\mathrm{IC}\,(c_2\,)}$$

5. The similarity measure defined by Chodorow and Leacock is

$$\mathrm{Sim}_{\mathrm{lch}}\ (c_1\,, c_2) = \max -\log \frac{\text{shortest Len}(c_1\,, c_2)}{2 * \text{depth of the taxonoy}})$$

The Figure 1 gives detailed description of the algorithm used by these authors.

It is shown that original Lesk algorithm achieves 59% precision, whereas modified Lesk algorithm achieves 67% on using the similarity measure proposed by Leacock and Chodorow.

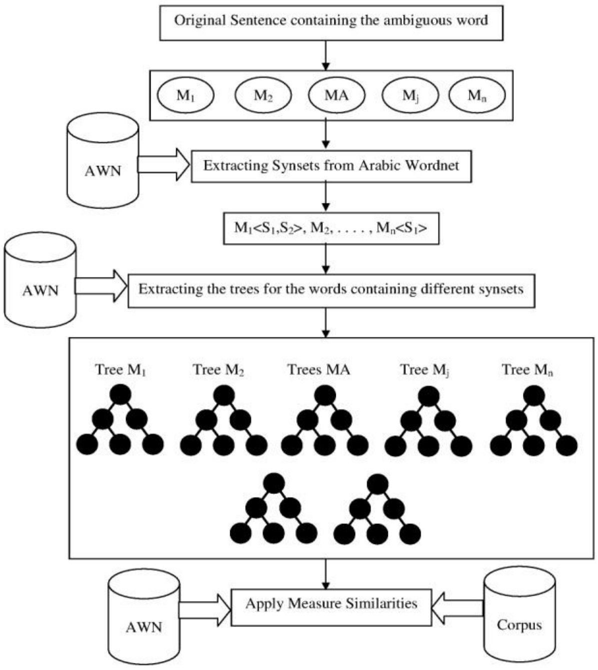
Original Sentence containing the ambiguous word
M1
M2
MA
Mj
Mn
AWN
Extracting Synsets from Arabic Wordnet
M1<S1,S2>, M2, ... , Mn<S1>
AWN
Extracting the trees for the words containing different synsets
Tree M1
Tree M2
Trees MA
Tree Mj
Tree Mn
AWN
Apply Measure Similarities
Corpus

Figure 1: Architecture of Modified Lesk algorithm using AWN

In [25], authors proposed a WSD system using the support vector machine (SVM) classifier following the Levenshetin Distance algorithm to determine the matching distance between words. They compared the performance of this technique to supervised and unsupervised Machine Learning algorithms like NBC and LSA with k-means clustering.

In [26], authors have proposed a hybrid system of Arabic words disambiguation. They used Latent Semantic Analysis, Harman, Croft, Okapi, methods in the domain of information retrievals, followed by Lesk algorithm.

They extracted the ambiguous words from the corpus collected from the web. They performed several pre-processing steps to the words denoting the different contexts of use of eh ambiguous word to enhance the performance of the system. Signatures describing a unique sense for the different senses of ambiguous word are created from the collection of possible contexts of use of each ambiguous word and using the tf x idf measure. Also, the contribution of syntactic knowledge on the outcome of disambiguation is considered. Then they implemented and tested several methods used in the domain of information retrieval namely LSA, Harman, Croft and Okap to calculate the similarity between the current context of occurrence of the ambiguous word and the different possible contexts of use of the word to disambiguate. Figure 2 provides a detailed method used by them to disambiguate the ambiguous Arabic words. Finally, they have adapted the Lesk algorithm to measure the proximity between the words that appear in the different definitions given by the methods of information retrieval. The output is the most relevant sense. They conducted experiment for a small sample of 10 ambiguous words achieving 73% of disambiguation rate.

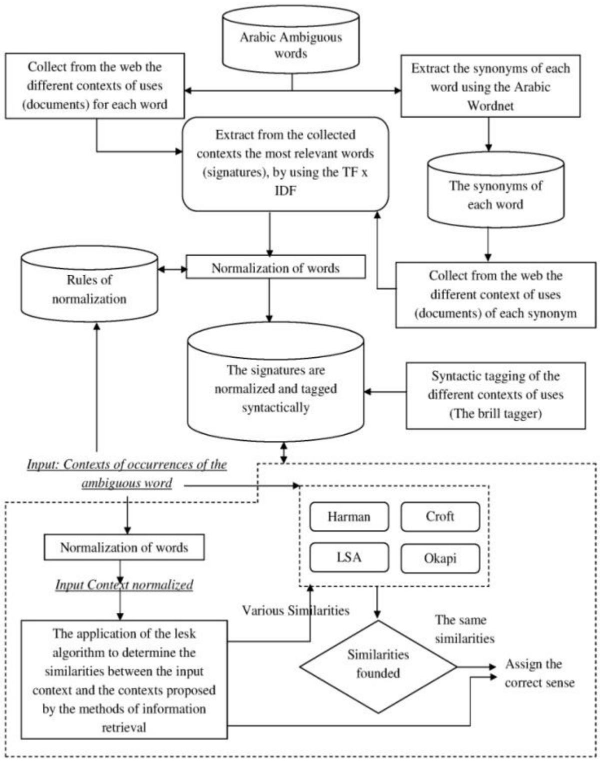
Arabic Ambiguous words
Collect from the web the different contexts of uses (documents) for each word
Extract the synonyms of each word using the Arabic Wordnet
Extract from the collected contexts the most relevant words (signatures), by using the TF x IDF
The synonyms of each word
Rules of normalization
Normalization of words
Collect from the web the different context of uses (documents) of each synonym
The signatures are normalized and tagged syntactically
Syntactic tagging of the different contexts of uses (The brill tagger)
Input: Contexts of occurrences of the ambiguous word
Harman
Croft
LSA
Okapi
Normalization of words
Input Context normalized
Various Similarities
The same similarities
The application of the lesk algorithm to determine the similarities between the input context and the contexts proposed by the methods of information retrieval
Similarities founded
Assign the correct sense

Figure 2: Method to disambiguate the ambiguous Arabicword senses

In [27], authors have proposed a semi-supervised method for Arabic word sense disambiguation. Using the Khoja stemmer and the approximate string matching following the pre-treatment clustered the sentences containing ambiguous words extracted from the corpus with the help of Arabic WordNet. The sentences in each cluster are transformed to binary trees. Then semantic trees are created through merging the binary trees corresponding to sentences in clusters using breadth first traversal. A weighted directed graph is constructed by matching the tree of the original sentence with semantic trees of each sense candidate. The weights are calculated based on one of the three collocation measures namely T-test, mutual information and the chi-square $\chi 2$ defined by Manning and Schutze. The closest semantic tree to the tree of the original sentence is determined by defining score measure based on weights in the weighted directed graph. The correct sense is determined using a novel supervised voting procedure. This system results in high recall and precision of 83%.

In [28], authors have proposed an Arabic WSD system that depends equally on information extracted from the local context of the word to be disambiguated and the global context extracted from the full text. They proposed a system that consists of the following steps. Sense Inventory of Non ambiguous words and Ambiguous words is created using the Arabic WordNet (AWN). Word senses are represented as vectors. The unique sense S_i of each Non ambiguous word w_i is considered. Then, the vector space spanned by the standard basis B= {e_i} i=1...n is built. Word sense is represented by the vector $V=\sum a_i e_i$ where a_i is the wu-p semantic distance between word sense and the sense S_i. The global context $contx_{Global}$= {v_1, v_2 …vn} ids defined by the sense vectors set of non ambiguous words present in the full text. Local context is also defined similarly but considers only non ambiguous present locally. Finally, an ambiguous word a_w that has m senses will be represented by the set of its sense vectors such as $a_w = \{w_1, w_2 \ldots w_n\}$. The disambiguation involves three distance measures namely dot product, Cosine, and Jaccard to calculate the similarity measurement between any two vectors. Local semantic proximity and Global semantic proximity for each sense S_i is calculated through identifying the percentage of similar vectors. Output the sense associated with the maximum of the average of local and global semantic proximity. This system achieves a precision of 74%.

In [29], authors have investigated the possibility of applying Genetic Algorithms (GA) in designing a state of the art WSD system for Modern Standard Arabic language (MSA).Supervised WSD methods outperforms knowledge based methods but are limited to small contexts. Knowledge based methods take exponential computational time as the number of words increases. Authors have proposed using GA to reduce the time by approximating the solutions without compromising on accuracy. They have obtained a bag of words $\{w_1, w_{2\ldots}w_k\}$ from a given text T through preprocessing phase that involves stop-word removal, tokenization and rooting. They have reduced the words to their roots using Khoja's stemmer that achieves higher accuracy than their counterparts. The most appropriate mapping from words w_i to senses $\text{senses}_{AWN}(w_i)$ in the context T is determined through GA. Here, the GA returns the best individual S_{best} that is decoded into the phenotype space to get the appropriate sense of words. They defined the necessary elements to formulate the WSD problem in terms of GA. An individual Ind_p representing a possible sequence of sense indexes assigned to the words in the context T is represented by a fixed-length integer string. The initial population is generated through either Random generation or Constructive generation. Fitness of an individual is determined through Lesk measure and extended Lesk measure. Single-point crossover and single-point mutation were considered. The roulette wheel and tournament selection methods to select parents for the mating pool were considered. The elitist survivor selection method is considered by them to improve the performance of GA's. They have performed number of generations and number of fitness evaluations before terminating the GA. They presented results of experiments with GAWSD on a set of 5218 words extracted from Arabic data corpus. They considered words within a text window of size 2 to limit the context size. Using this data, they have evaluated the performance of GAWSD under different settings of the parameters namely population size population_{size}, crossover rate $P_{crossover}$, mutation rate $P_{mutation}$ and termination condition $T_{condition}$. The best performance is achieved for $\text{population}_{size} = 50$, $P_{crossover} = 0.70$, $P_{mutation} = 0.15$, $T_{condition} \geq 4000$. Finally $GAWSD_{TS}$ is compared against a WSD using Naïve Bayes Classifier. The best mean Precision 0.79 is given by $GAWSD_{TS}$, however best mean Recall 0.68 is given by Naïve Bayes Classifier but mean Recall of $GAWSD_{TS}$ is 0.63 not significantly different. From this they concluded that $GAWSD_{TS}$ is not only able to find more relevant word senses than the Naïve Bayes Classifier but also can return more relevant senses.

In [30], authors have proposed an approach for Arabic WSD involving Wikipedia as the lexical resource. The text containing ambiguous words is preprocessed and ambiguous words are identified using Arabic WordNet (AWN). For each sense of ambiguous word either first sentence or first paragraph containing the word is retrieved from Wikipedia. The word's context and the retrieved sense context are represented mathematically using a vector space model. Then the cosine of the angle between vectors of word's text and each retrieved sense are determined using an equation

$$\cos(x, y) = \frac{x.\,y}{||x||\,||y||}$$

where x.y is the inner product of the vectors and $||x|| = \text{sqrt}(x[0]^2 + x[1]^2 + \ldots)$.

Two vectors x and y are similar if the result is equal to 1, and they are not similar at all if it is zero. They have selected the most appropriate sense based on the cosine similarity calculated as above. They have conducted three experiments to evaluate the approach. First experiment considered only on sentence retrieved from Wikipedia and the raw frequency VSM is used. Second experiment is similar to the first except that Tf-Idf vector space mode is used. In the third experiment first paragraph retrieved from Wikipedia is considered. They observed using the first paragraph results in better solutions than using one sentence and using Tf-Idf outperforms use of raw frequency.

In [31], authors have proposed a novel approach for Arabic WSD which involves the use of two external resources Arabic WordNet (AWN) and English WordNet. After preprocessing the given text, words are mapped into concepts if they are in AWN. Otherwise, term-to-term Machine Translation System from Arabic to English is used to have the equivalent word in English. Then WordNet is used to map the word into concept. Their idea in selecting the most appropriate concept is that it establishes more semantic relationship with different concepts in the local context. Then the concept selected is translated back to Arabic using Machine Translation System from English to Arabic if required. They have used Wu and Palmer's similarity measures, Chi-Square statistics for feature selection and local and global weighting concepts. Finally their WSD system has achieved an accuracy of 73.2%.

5. CONCLUSION

This paper initially focused on unsupervised graph-based word sense disambiguation methods for English, which have been attracting a wide focus due to their ability to truncate the accuracy gap from the supervised methods. Then, we have focused on currently available WSD techniques for various Indian languages and some Asian languages. From this, we observed that research work in WSD has been preceded up to different extents according to the availability of different resources like corpus, WordNet, thesauri, tagged data set etc. In Asian languages, development of corpus, WordNet and other resources is progressing slowly due to the more morphological inflections. The accuracy of the WSD and performance of the system depends on size of the corpus; accuracy can be improved by large training corpus. For languages like TELUGU, spoken by the people of states Telangana and Andhra Pradesh in India, the availability of required resources for developing WSD is less to our knowledge. In the future, we would like to develop a comprehensive WSD system for TELUGU.

6. ACKNOWLEDGMENTS

We thank various authors who have developed WSD systems for various languages which have helped us in preparing this survey paper. And also thanking reviewers of our paper whose remarks have motivated us in completing the paper covering major issues.

7. REFERENCES

[1] R.Navigli, Word Sense Disambiguation: a Survey, ACM Computing Surveys, Vol. 41, No.2, ACM Press, pp. 1-69 2009.

[2] E. Agirre and A.Soroa, Personalizing pagerank for word sense disambiguation. EACL, pages 33-41, 2009.

[3] R. Mihalcea, P. Tarau, and E. Figa. Pagerank on semantic networks with application to word sense disambiguation. In *Proc. of COLING*, 2004.

[4] R.Navigli, Online word sense disambiguation with structural semantic interconnections. In *Proc. of* EACL 2006

[5] R.Sinha and R.Mihalcea, Unsupervised graph-based word sense disambiguation using measures of semantic similarity. In *Proc. of* ICSC, 2007

[6] G.Tsatsaronis et al, Word Sense Disambiguation with Spreading activation networks generated from thesauri, In Proc. of IJCAI, pages 1725-1730.

[7] R.Navigli, and M.Lapata, An experimental study of graph connectivity for unsupervised word sense disambiguation IEEE Transactions on Pattern Analysis and Machine Intelligence, 2010

[8] G.Tsatsaronis, IraklisVarlamis, and Kjetil Norvag, An experimental study on unsupervised graph-based word sense disambiguation, In *Proc. of* CICLING, 2010

[9] Parameswarappa S and Narayana V.N, Kannada word sense disambiguation for machine translation, IJCA vol.34, Nov 2011.

[10] Parameswarappa S and Narayana V.N, Kannada word sense disambiguation using decision lists, IJETTCS, 2013

[11] R P Haroon, Malayalam Word Sense Disambiguation Computational Intelligence and Computing Research (ICCIC), IEEE 2010.

[12] Sreelakshmi Gopal, and Rosna P Haroon, Malayalam word sense disambiguation using Naïve Bayes Classifier, IEEE Conference HMI-2016

[13] Jumi Sarmah, and Shikhar Kr. Sarma, Decisin tree based word sense disambiguation for Assamese, IJCA, Vol-141 May 2016.

[14] PurabiKalita, and Anup Kumar Barman, Implementation of Walker algorithm in word sense disambiguation for Assamese language, IEEE Conference 2015.

[15] Manish Sinha, Pushpak Bhattacharya et al, Hindi word sense disambiguation.

[16] Sundeep Vishwarkarma, and Chanchal Vishwarkarma, A graph-based approach to Word sense disambiguation for Hindi language, IJSRET 2012.

[17] Satyendar singh et al, Hindi Word sense disambiguation using semantic relatedness measure, Springer 2013.

[18] Preeti Yadav, and Sundeep Vishwarkarma, Mining Association rules based approach to Word sense disambiguation for Hindi language. IJETAE 2013.

[19] Gaurav et al, Probabilistic Latent Semantic Analysis for Unsupervised Word Sense Disambiguation, IJCSI, Vol. 10, Issue 5, No 2, September 2013.

[20] Sabnam Kumari, and Paramjit Singh, Optimized Word Sense Disambiguation in Hindi using Genetic Algorithm, IJRCCT, Vol. 2, Issue 7, July 2013.

[21] Devendra K.Tayal et al, Word Sense Disambiguation in Hindi Language Using Hyperspace Analogue to Language and Fuzzy-C Means Clustering, International Conference on Natural Language Processing , 2015.

[22] Arindam Roy, Sunita Sarkar, and Bipul Syam Purkayastha, Knowledge based approaches to Nepali Word sense disambiguation, IJNLC June 2014.

[23] Nyein Thwet Aung, Khin Mar Soe, Ni Lar Thein A word sense disambiguation system using Naïve Bayes algorithm for Myanmar language, IJSER September 2011.

[24] A. Zouaghi1, L. Merhbene2, and M. Zrigui2 Word sense disambiguation for Arabic language using variants of the Lesk algorithm , ICAI 2011

[25] Mohamed M. El-Gamml, M. Waleed Fakhr, A comparative study for Arabic word sense disambiguation using document preprocessing and machine learning techniques, ALTIC -2011, Alexandria, Egypt.

[26] Laroussi Merhben, Anis Zouaghi, and Mounir Zrigui Ambiguous Arabic Words Disambiguation: The results, RANLP 2009.

[27] Nadia Bouhriz, Faouzia Benabbou, and El Habib Ben Lahmar, Word sense disambiguation approach for Arabic text, IJACSA, 2016.

[28] Laroussi Merhbene, Anis Zouaghi, and Mounir Zrigui A Semi-Supervised Method for Arabic Word Sense Disambiguation Using a Weighted Directed Graph, International Joint Conference on Natural Language Processing, 2013.

[29] Mohamed El Bachir Menai, Word Sense Disambiguation Using an Evolutionary Approach, Informatica, Vol.38, 2014.

[30] Marwah Alian et al, Arabic Word Sense Disambiguation Using Wikipedia, IJCSI, Vol. 12, September 2016.

[31] Meryeme Hdni et al, Word Sense Disambiguation for Arabic Text Categorization, IAJIT, Vol.13, 2016.