Received: March 15, 2017

INASS

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Shallow Parsing and Word Sense Disambiguation Used for Machine Translation from Hindi to English Languages

Shachi Mall[1]* **Umesh Chandra Jaiswal[1]**

[1]*Madan Mohan Malaviya University of Technology, Gorakhpur, India*
* Corresponding author's Email: shachimall@gmail.com

Abstract: This paper developed innovative algorithms such as shallow parsing and modified Lesk's algorithm to resolve the issues in Word Sense Disambiguation and performed correct translation from Hindi language to English language. Shallow parsing method is based on Hidden Markov model. We also perform an evaluation for 1657 Hindi tokens with 990 phrases for Parts of speech tagging and Chunking for given Hindi sentence as input and able to achieve Precision, Recall, F-score, Accuracy for Parts of speech tagger: Accuracy: 92.09%; precision: 84.76%; recall: 89.29%; F-score: 86.97, system accuracy for Chunk: Accuracy: 93.96%; precision: 89.33%; recall: 91.31%; F-score: 90.315%. The evaluation is performed by developing confusion matrix in which the system result of Parts of speech tagger and Chunk is compared with Gold standard date provided by IIIT Hyderabad in the summer school 2015. In this paper we discuss the second problem Word Sense Disambiguation in which we enhance the Modified Lesk algorithms by using overlap based method which will find information between three pieces of words in a given context. The system generated result resolves the issues of Word Sense Disambiguation and shows the comparison result with the website Google Translator in which we input polysemy word in a Hindi sentence and same sentence input in our generated system and shows the comparison result of both. The output result shows that our system resolves Word Sense Disambiguation and produces correct translation and Google Translator is fails to resolves the correct Translation.

Keywords: Parsing, Word sense disambiguation, Parts of speech and chunk.

1. Introduction

This paper is based on Machine Translation [1]. Machine translation has been well understood in the area of artificial intelligence from the field natural language processing. In the field of machine translation, researchers are still focusing on parsing technique [2] and Word Sense Disambiguation [3] (WSD) these two methods are very important concept that is to be evaluated for performing machine translation. This tool is needed to perform disambiguation so that computers would be able to interpret a word in its proper sense according to its context. There are various Machine Translations system has been developed for Indian languages such as Google Translator, BabelFish Translator etc. but they are fails to provide a good quality of translation. In India various languages people speak among that Hindi is a national language. In our work we decide to develop a system that translates Hindi language to English language. There are various challenges faced in Machine Translation such as Morphological analyzers [4, 5], parsing [6] Word sense disambiguation and Translation. Parsing method for Hindi language is important task to resolve the issues in the Hindi sentences because Hindi language is morphologically rich and free order in nature the tokens are expressed in different form in the same Hindi sentences. Another issue is Word Sense Disambiguation (WSD). This paper proposed Knowledge-based in which we use WordNet tools [7, 8], supervised, minimally supervised, unsupervised approach and domain specific method to resolve disambiguation problem. Hindi words are polysemy which has ambiguity in an individual word in phrase [10] that can be used in different contexts to express two or more different

meanings. To distinguish correct sense in words is a challenge in Natural Language Processing systems. We enhance the Modified Lesk algorithms in which we use Hindi and English WordNet tools which is used in lexical knowledge. Our contribution in this paper to improve Parts of speech tagging for a word which is multiple tags this can be handled by Maximum likelihood estimation and improvement help us to resolve the problem in word sense disambiguation for Hindi language.

The rest of the paper is organized as follows. Section 2 Related works on Machine Translation. Section 3 elaborately describes our approach Proposed model for Parsing and Word sense disambiguation task. Experimental results of the development and the test sets are reported in Section 4. Finally, Section 5 concludes the paper.

2. Related work

Research has been going on for several years on Machine Translation. They are failing to resolve polysemy word. Translation quality is also not good as compare to human translator.

2.1 Indian languages machine Translation

Different Indian Researchers are working to improve machine translation system.

- Authors [12] used pattern directed rule based and Example Based system the accuracy result-90% for simple and compound sentence.
- Authors [13] used morphological analyser. The accuracy of the system reaches 69%.
- Authors [14] proposed a system Anusaaraka (English-Hindi) based on Paninian grammar formalism and shallow parser approach. Drawback-word sense disambiguation is not resolved
- Authors [15] developed a system for Hindi to English machine translation using Context free Grammar parsing technique. Drawback-Case (karaka) and gender is not resolved.
- Authors [16]. Approach- Dependency parsing. Result-76.5%. Drawback-person, number gender is not resolved.
- Authors [17] used Statistical phrase-based approach for word alignments. They present a model that decouples the steps of lexical selection and lexical reordering with the aim of minimizing the role of word-alignment in machine translation. Drawback-The bag-of-words model per- formed very well in predicting lexical items but was not as good as Moses at ordering them
- Authors [18] used Hybrid approach for word alignment for English-Hindi. Result-AER obtained using 270 training sentences 57.06%. Drawback-Adjectives may have several declensions in Hindi but not in English. Nouns and pronouns can also have different declensions in Hindi.
- Authors [19] used finite rules like Moses and Stanford Phrasal. BLEU (Bilingual Evaluation Understudy) is an algorithm. Result-Moses 37.4% and Phrasal 29.1%. Drawback-Data was set before training, the English -Hindi corpus (of Indian names) using Phrase based statistical machine translation.
- Authors [20] developed Word sense disambiguate algorithm in which they combine supervised and unsupervised method. The accuracy of the work is evaluated for 30 words and produces 80% result.

2.2 Problem statement

The issues identified from the literature review are as follows:

- Parts of speech tagging and chunking to multiple sentence is a challenging task
- Supervised method fail in mapping and labelling each word with corresponding Parts of speech tag in linear function.
- Hindi words are polysemy words. To handle Words Sense Disambiguation for Hindi language.

3. Proposed model

The proposed system is divided into following modules:

3.1 Split the sentence from the Hindi text

User input the Hindi text H. Tokenizer has two tasks. First it takes the raw text and mark the sentence boundary. Second it takes the boundary marked text and produce the token in the form of SSF (Shakti Standard Format) format.

- A token may be any of the following: word, abbreviation, punctuation mark, real number, special symbol etc.
- No token has white space in it.
- Purnaviram "|" (DevanagriDanda), full stop "." and new line ("\n") are treated as end of sentence marker. Store the list of sentence in separate
- Input-Output Specifications

Input Hindi text - भाजपा के राष्ट्रीय अध्यक्ष राजनाथ सिंह शाम छह बजे लखनऊ एयरपोर्ट पहुंचेंगे।

Output- Output is stream of tokens with sentence boundaries marked in Shakti Standard Format.

<Sentence id="1">

0. भाजपा unknown
1. के unknown
2. राष्ट्रीय unknown
3. अध्यक्ष unknown
4. राजनाथ unknown
5. सिंह unknown
6. शाम unknown
7. छह unknown
8. बजे unknown
9. लखनऊ unknown
10. एयरपोर्ट unknown
11. पहुंचेंगे unknown
12. | unknown

</Sentence>

3.2 The sentence is sequentially tags with their related Parts of speech

- User input Hindi sentence. The sentence is converts the input file into Shakti Standard Format (SSF) to Trigram (TnT) and display of output file is again converted form Trigram to Shakti Standard Format (SSF).
- Build Transition Count Matrix and Build Emission count matrix. Build a hash of the tag sequence and its frequency calculated by Eq. (1)
- N-grams smoothing technique is used as discussed in Eqs. (7) – (10). The tag sequence of a given word sequence.
- Convert the output generated by part of speech tagger which is in TnT format to SSF format.

To remove ambiguity in multiple tags for a single word we use Hidden Markov [21] for Parsing to identify the dependency between each predicate in a given input sentence. We use Viterbi approximation in Eq. (11) to choose the most probable tag sequence for given input Hindi sentence. To estimate we read off count from the training corpus and then computer the maximum likelihood. Firstly we calculate Transition matrix we have a set of words in a given sentence

W_1-----W_T represents the sequence of the word, P is a probability and T is the probable tag sequence.

$$T = t_1, t_2 \dots t_n$$
$$\hat{T} = arg_{T\epsilon t}\, P(\frac{T}{W}) \quad (1)$$

Equation (1) is used to choose the sequence of tags that maximizes.

$$\{P(T)P\ (W/T)\} / P\ (W)$$
$$\hat{T} = \frac{arg_{T\epsilon t}\ \max\{P(T)\times\frac{W}{T}\}}{\left(\frac{P}{W}\right)} \quad (2)$$

Equation (2) is used to calculate the sequential of total words tags with their corresponding parts of speech.

Equation (3) is used to calculate the probability of the tag given the past depends on the last two tags. Where P is a probability of word which depends on its tags, W is sequence of words and t is a sequence of tags corresponding to words:

$$P(T)P\left(\frac{W}{T}\right) = \prod_{i=1}^{n} P\left(\frac{W_i}{W_1 t_1} \dots . W_{i-1} t_i\right) P(\frac{T_i}{W_1 t_1 \dots . W_i t_{i-1}}) \quad (3)$$

Equation (4) is the derivation from chain rule, where argmax is taken overall sequences of paired words $W_{1\dots n}$ with the corresponding tag $t_{1\dots n}$ used to choose the tag sequence that maximizes:

$$P(T_1)P\left(\frac{t_2}{t_1}\right) = \prod_{i=1}^{n} P\left(\frac{T_i}{t_{i-2}t_{i-1}} \dots . W_{i-1} t_i\right) \prod_{i=1}^{n} \left(\frac{W_i}{t_i} \dots . W_{i-1} t_i\right)$$
$$argmax\left[\prod_{i=1}^{n} P\left((t_i(t_{i-2}t_{i-1})\ P(W_i t_i)\right)\right] \quad (4)$$

To resolve the multiple tags for same word, we use Eqs. (5) and (6) Maximum likelihood estimation from relative frequency to estimate these probabilities. Where C is a chunk for each sentence we seek to find the best possible chunk ordering denote all the chunk combination for the sentences:

$$P\left(\frac{t_i}{t_{i-2}t_{i-1}}\right) = \frac{C(t_{i-2}t_{i-1}t_i)}{CC(t_{i-2}t_{i-1})} \quad (5)$$

$$P\left(\frac{W_i}{t_i}\right) = \frac{C(W_i t_i)}{C(t_1)} \quad (6)$$

Equations (7) - (10) are used to calculate the probabilities of N-gram smoothing technique. This technique is used to resolve the issues of multiple tags and assign correct tag to resolve the problem for polysemy word by assigng correct tag this

produce correct transaltion.If we have some tagged text available we can compute the number of times (W,t) the number of times f(t_1,t_2,t_3) in this text we can esstimate the probability by using the context window size two in which it take left and right side of the context word to find the high frequency probability of words. Equation (7) Unigram techniqueto calculae only left word next to context word. Equation (8) Bigram techniqueto calculae only right word next to context word. Equation (9) Trigram techniqueto calculae both left and right words next to context word. Equation (10) is used to calculate probabilty of three words.

$$u\,nigram = P(t_3) = \frac{f(t_3)}{N} \quad (7)$$

$$Bigram = P\left(\frac{t_3}{t_2}\right) = f\left(\frac{t_3,t_2}{t_2}\right) \quad (8)$$

$$T\,rigrams = P\left(\frac{t_3}{t_1,t_2}\right) = f\left(\frac{t_1,t_2,t_3}{t_1,t_2}\right) \quad (9)$$

$$Lexical = \hat{P}\left(\frac{W_3}{t_3}\right) = f\left(\frac{W_3,t_3}{t_3}\right) \quad (10)$$

Let us consider an example Input Hindi sentence एशिया की सबसे बड़ी मस्जिदों में से एक है। The process starts from one word to another word. Each move is called a step. If the chain is currently in state X i then it moves to state X i at the next step with a probability denoted by P i,j, and this probability does not depend upon which states the chain was in before the current state this method is known as Transition matrix. To calculate Transition matrix Q i = [W i = i]. Suppose we have N-state Hidden Markov Model parameterized by (E, Q, R) where emission probability represents E, Q is an initial probability and transition probability matrix represent R. Let rows of R identical and given by vector r, the joint probability of the hidden states and observations over a sequence of length O can be O = O1, O2, O3…On where U is a sequence of word and V is corresponding tag for sequence of words with their related Parts of speech and Z is normalization function drawn from set of tags t. are calculated by Eq. (11):

$$Z\frac{(U,V)}{E,Q,R} = \frac{Z\,U_i}{\begin{bmatrix} Q\pi_{0-2}^{0}\,Z(U_0 r \\ )Z\frac{U_0}{U_0,E} \end{bmatrix}} \quad (11)$$

Output POS Tag:

यह<JJ>एशिया<NP>की<PSP>सबसे<QF>बड़ी <JJ>मस्जिदों<NNP>में<PRP>से<PSP<QC>है <VM>।<SYM>

Table 1 Chunk with their abbreviation

Chunk symbol	Chunk phrase
B-CCP	Beginning Conjuncts Chunk
B-JJP	Beginning Adjectival Chunk
B-NP	Beginning Noun Chunk
B-RBP	Beginning Adverb Chunk
B-VG	Beginning Verb Chunks
I-CCP	Inside Conjuncts Chunk
I-JJP	Inside Adjectival Chunk
I-NP	Inside Noun Chunk
I-RBP	Inside Adverb Chunk

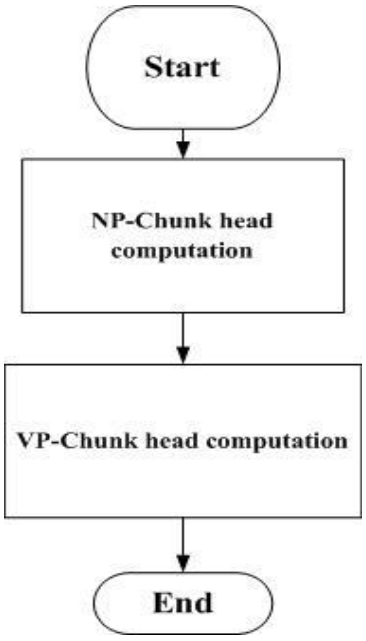

Figure.1 Chunk computation

Through the above calculation we find tag for other words in a given sentence and input for the process of Chunk. Figure 4 shows the snapshot of Parsing with Parts of speech tagging for given Hindi sentence. Chunking [3] is an important process to identifying and segmenting the text into syntactically correlated chunk tag such as is NP chunk label the word in the sentence start with different Phrases, we label the word with boundary marker B represents -Beginning phrase and I represent as Inside phrase for example we input Hindi sentence: दफ्तर के सभी लोग अपने अपने घरों को जाने की जल्दी में थे।

दफ्तर NN B-NP के PSP I-NP सभी QF B-NP लोग NN I-NP अपनेPRP B-NP SYM I-NP अपने RDP I-NP घरोंNN B-NP

The sentence is individually tokenize by the delimiter "?|" in sentence start with<Sentence id="">chunk start with assigning chunk number "((chunk phrase<fsaf='Hindi word, the features of the word and chunk Table 1 shows the abbreviation

of chunk symbols. Chunk is an arbitrating step towards parsing. In Fig. 1, Head computation is used for functional specification to compute the phrase with heads of different phrases of groups such as noun, verb groups etc. Chunk head provides the sufficient information for further processing of the sentence. Figure 4 shows the output result of Hindi token label with related parts of speech tagging and chunk.

3.3 Parsing

Parsing uncover the hidden structure of Hindi text input it can provides structural description that can identifies the break intonation and analyse a given sentence to determine its syntactical structure according to the part of speech tag and chunk. In natural language processing the syntactic analysis of Hindi language can vary from low level such as Part of speech tagging.

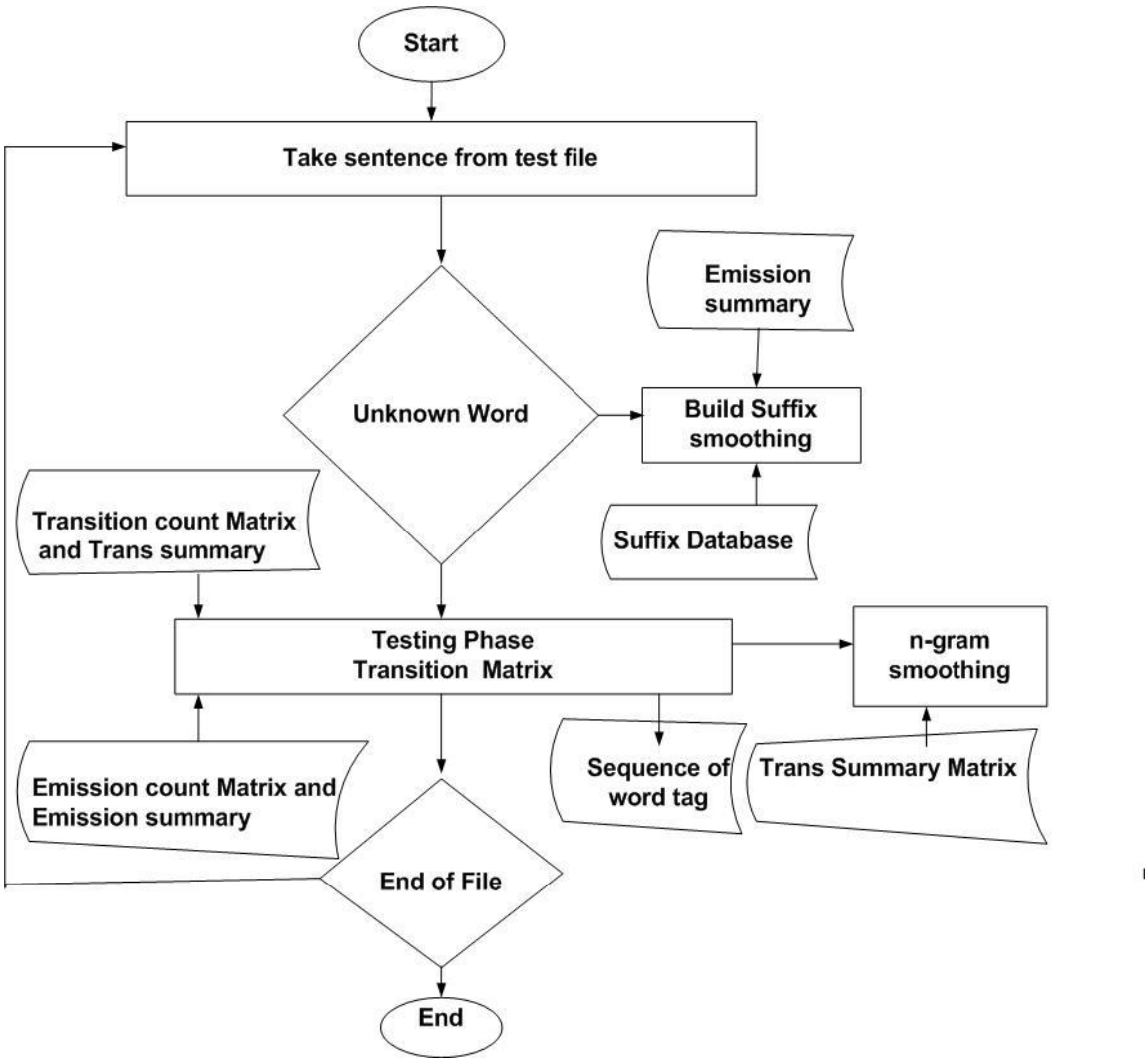

Figure.2Folw chart of Parts of speech tagging

Algorithm 1 Parsing

1. for i ← 0 to length words
2. do
3. for each word is Chunk with Noun phrase then
4. Select parent head word "B"
5. Select part of speech H
6. Select voice of H
7. Select position of H (left, right)
8. Else if word is a verb then
9. Select nearest word N to the left word such that word is the parent head word of "I"
10. Select nearest word r to the right of word such that word is the parent head word of r
11. Select part of speech of l
12. Select part of speech r
13. Select the part of speech word
14. Select voice of word

15. Else if word is adjective then
16. Select parent head word head
17. Select part of speech of head
18. end

Parsing is used to estimate the number of useful probability concerning and its syntactical structure of the sentence. In the parsing algorithm we develop some identification rule are as follows:

- In case of NN most of the time ambiguity is in case marking (direct, oblique). We can decide the case on the basis of following PSP.

1) Rule 1: If NN is just followed by PSP, then we will take only the feature structures having oblique case. Else we will take the direct case.

- In case of JJ the case (d/o), should agree with the noun it is modifying.

2) Rule 2: If JJ has multiple morph analysis then we will look for noun it is modifying and we will take the morph analysis of JJ having case marked same as that of modified noun and eliminate the rest.

- In case of PSP the pruning module is giving multiple morph analysis for 'ke' and 'kI'.

3) Rule 3: We will look for the noun to which our PSP is related and will keep the morph analysis having gender and case agreeing with the gender and case of the noun to which our PSP is related and eliminate the rest.

- Most of the time the noun is related with PSP is found in the next chunk to chunk containing PSP. Then most probably head of the chunk is NP.

WSD is identifying which sense of a word and meaning is used in a sentence, when the word has multiple meanings. We use Modified Lesk algorithms in which overlap is finding between three pieces of words.

This can be done by using WordNet [7] tools and Domain specific sense Table. The tables are created in three categories C_1, C_2, C_3. Where C_1 is a Fields or Domain name, C_2 is a general words present in the sentence and C_3 is a Meanings of each words. We assign unique ID to each words in general words category. Word Meanings are given from WordNet tools in which ID, words and respective domain ID assigned to each word. These three categories help us to find the correct translation.

When user input the sentence then each word are tokenize and label with grammatical tag. If the tag is multiple then it depends on tokens of the input sentence. Words which is polysemy is labeled with multiple tag their related parts of speech here we use N gram technique. We divide the context in three windows window 1, window 2 and window 3. Let window of context is, 2t+ 1 with the grammar R. Were list of the word in WordNet is define Ti, $1 \leq i \leq R$. compare the list of in the WordNet is less than 2t + 1, if all the list of words in WordNet belong to the context. Were Ti is list of words contain more than two meaning in the gloss, list of words are assign with unique synset having a unique sense tag. Were Ti is the lists of sense tag are represented by |Ti|. We evaluate sense tag for each pair of words in the context of the window. Were R i=1|Ti| represents combinations of words this is referred as candidate combination.

In a given context to find the correct word sense by counting word overlaps between glosses or word meaning of the words in the context. All the glosses of the key word are compared with the glosses of other words. The sense for which the maximum number of overlaps occur, represents the desired sense of the of the polysemy word.

Algorithm 2 Modified Lesk Algorithm

1. Input the word which is multiple tag
2. Now Distributed Domain approach is used
3. The domain is distributed to the context words from the WordNet Domain
4. This Domain maintains the table as shown in Fig. 1
5. Read a single line of Hindi text.
6. Decompose this paration of sentence into three categories as C1, C2 and C3 for finding results.
7. It is required to detect correct sense of word with the help of most suitable domain for a word using various algorithms and finally the meaning of a sentence.
8. Find specific tag from prepares h i.e. our machine readable dictionary (med?)
9. Sense the tag and scan for unknown tags or tag which have more than one frequency
10. Assume overlap based (tagging ambiguity) approach and find out actual tag in the Hindi word and context
11. The target word is selected by comparing WordNet, available domain and the domain of target word is displayed
12. When definite tag is found it is again stored prepares
13. Identification of Domain
14. The accurate Domain of the target word is identified by supervised and semi supervised approach

Table 2. Result of Parts of speech tagging

Abréviation of Parts of Speech	Prcesion %	Recall%	F-Score	Accuracy
CC	95.75	98.235294	97.909091	94.33333333
DEM	63.157895	92.307692	75	60
INJ	80	36.363636	50	33.33333333
JJ	69.230769	66.176471	67.669173	51.13636364
NEG	100	100	100	100
NN	76.352705	96.455696	85.234899	74.26900585
NNP	100	27.272727	42.857143	27.27272727
NST	100	100	100	100
PRP	91.891892	80	85.534591	74.72527473
PSP	97.154472	95.6	96.370968	92.99610895
QC	85.714286	100	92.307692	85.71428571
QF	75	81.818182	78.26087	64.28571429
RB	100	42.857143	60	42.85714286
RP	84.090909	88.095238	86.046512	75.51020408
SYM	100	100	100	100
VAUX	91.715976	85.635359	88.571429	79.48717949
VM	80.269058	85.238095	82.678984	70.47244094
WQ	89.473684	89.473684	89.473684	80.95238095
OVERALL SYSTEM	89.655647	92.862734	91.717502	94.97284249

Table 3.Result of Chunk

Chunk symbol	Precision	Recall	F-Score	Accuracy
CCP	100	100	100	
JJ--P	65.52	66.56	65.58	
NP	76.49	95.95	81.25	
RBP	90.63	83.33	93.68	
VG	83.33	97.47	75.76	
NP	76.49	69.44	64.52	
NEGP	100	62.05	94.12	
BLK	95.97	93.64	97.28	
FRAGP	0	0	0	
				95.17

15. Many texts have been manually tag as training example in pre parse edge and it uses a semi supervised approach. The semi supervised approach starts from journal Hindi text conversion it into specific tag manually and provide the machine readable dictionary at training example some sure free decision rule are applied in the condition to enhance the word tagging and generation facility.
16. After this semi supervised approach the supervised approach is used to enhance the context of the word which are also based on frequency wise context search and finding out the appropriate meaning of supervised and semi supervised approach can be denoted at advanced Lesk algorithm overlap based approach for word sense disambiguation.
17. Obtain sense of word
18. The sense of target word belonging to the domain is obtained which is added to the domain distribution table i.e. the table is updated using supervised and semi supervised approach

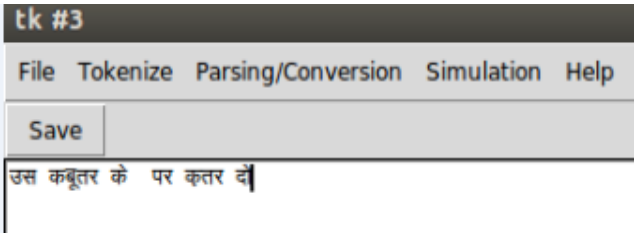

Figure.3 Snapshot of Hindi input sentence

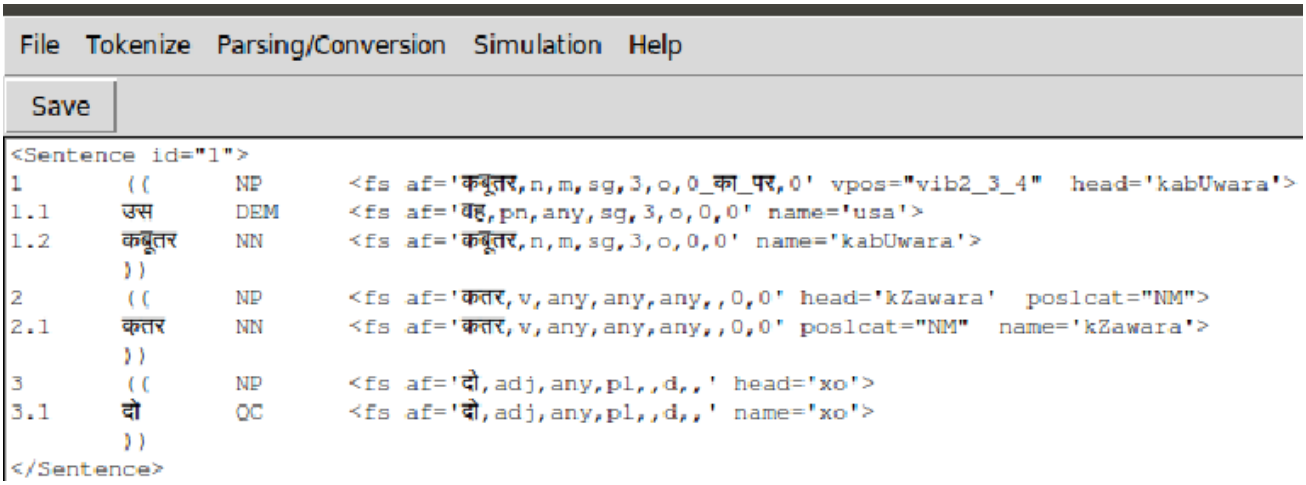

Figure.4 Snapshot of Hindi input sentence is parsed

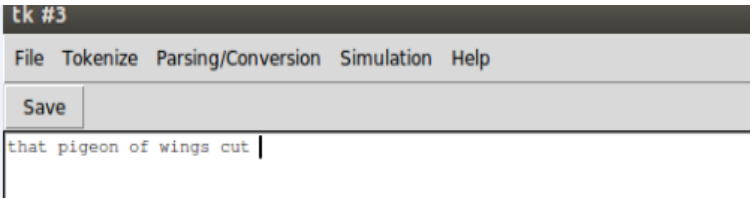

Figure.5 Snapshot of Hindi to English Translation with Word Sense Disambiguation पर=wing

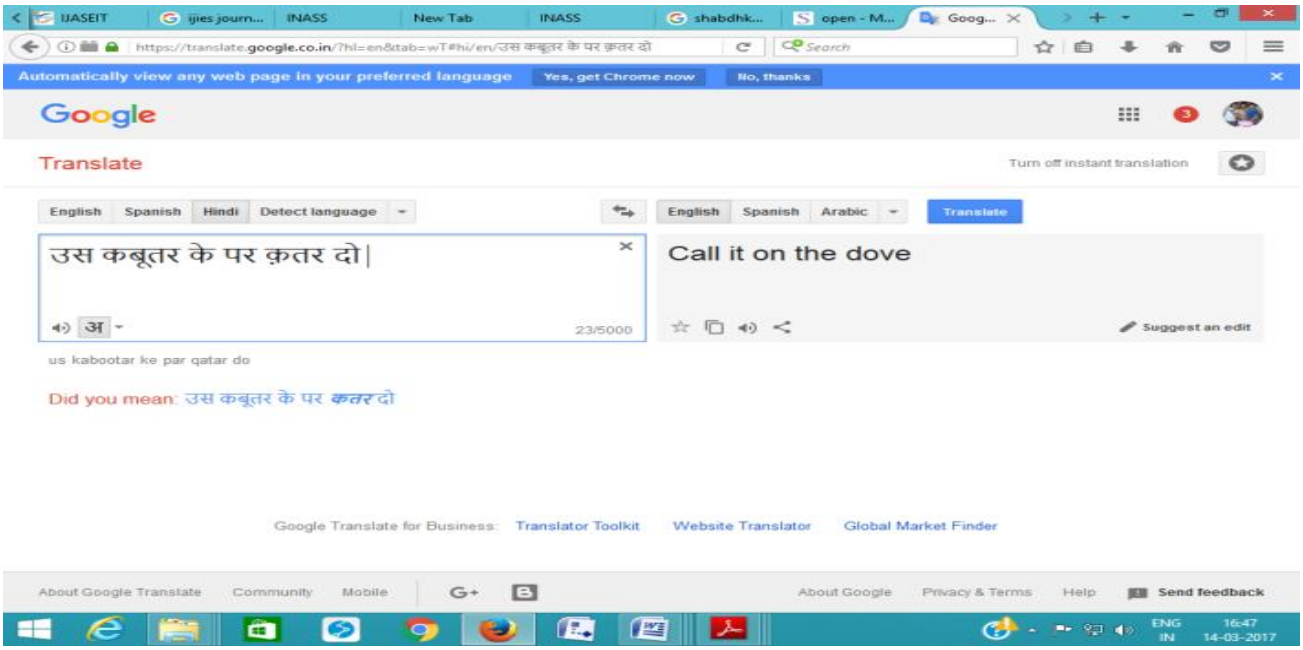

Figure.6 Snapshot of Hindi to English Translation Google Translator fails to resolve polysemy word

Let us consider an example Input Hindi sentence उस कबूतर के पर क़तर दो. As shown in Fig. 3 we perform following step:

- Tokenize the sentence with the delimiter "|"
- Tokenize the words from the sentence with the help of delimiter white space between two words.
- Stop the general words with the help of Parts of speech tagging roots words are identified.
- Parse the sentence as shown in Fig. 4
- Here word पर is polysemy word we use domain specific table in which we which we take two words from left and two words from right.
- Total numer of words are counted now we use N-gram smoothing technique Eqs.(7)-(9) and (10).
- Roots word match with Hindi WordNet tools which contains Field Id. Sense

Frequency is used to assign frequency to each sense of a word. This task is very much essential for division of word meaning into senses relevant task and of the Domain specific sense Table

4. Results

The simulations have been carried out using Python language to obtain the accuracy results of parts of speech tagging and Chunk. Chunking is a method used for parsing the Hindi sentences. The evaluation result of Precision, Recall and F-score for parts of speech tagging is discussed in Table 2 and Chunk evaluation results is discussed in Table 3, The output result of system generated parts of speech tag and Chunk are compared with Gold Standard parts of speech tag and Chunk [2]. Gold standard contains correct output of the parts of speech tag and Chunk for the given words. The total Hindi token was 1657 tokens with 990 phrases. Label each token with related parts of speech. THE Modified Lesk algorithm improves word sense disambiguation and the system generated result as shown in Fig. 5 is compared with Google translator website as shown in Fig. 6. This work shows that Google Translator cannot handled word sense disambiguation but our system can resolve word sense disambiguation. We compare our system generated output with available translating website such as Google Translator.

5. Conclusion

This paper developed innovative algorithms such as shallow parsing and modified Lesk's algorithm to resolve the issues in Word Sense Disambiguation and performed correct translation from Hindi language to English language. Shallow parsing method is based on Hidden Markov model .We also perform an evaluation for 1657 Hindi tokens with 990 phrases for Parts of speech tagging and Chunking for given Hindi sentence as input and able to achieve Precision, Recall, F-score, Accuracy for Parts of speech tagger: Accuracy: 92.09%; precision: 84.76%; recall: 89.29%; F-score: 86.97, system accuracy for Chunk: Accuracy: 93.96%; precision: 89.33%; recall: 91.31%; F-score: 90.315%. The evaluation is performed by developing confusion matrix in which the system result of Parts of speech tagger and Chunk is compared with Gold standard date provided by IIIT Hyderabad in the summer school 2015. The Gold data contains correct Parts of speech tagging. In this paper we discuss the second problem Word Sense Disambiguation in which we enhance the Modified Lesk algorithms by using overlap based method which will find information between three pieces of words in a given context. To find the correct word sense by counting word overlaps between glosses of the words in a given context. All the glosses of the key word are compared with the glosses of other words. The sense for which the maximum number of overlaps occur, represents the desired sense of the of the polysemy word and automatically decide the correct meaning of an ambiguous word based on the surrounding context in which it appears. The system generated result with resolved issue of Word Sense Disambiguation is compared with the website Google Translator website in which we input polysemy word पर is translated"on" in a given input Hindi sentence as shown in Fig. 7 and same sentence input in our generated system and shows in Fig. 6. Our system can resolve word sense disambiguation and generate translation for each Hindi polysemy word पर =Wing our system not performs Word alignment. This work can be further extracted by resolving the issues of language in which subject object and verb appear. Hindi language is subject verb object and English language is subject object verb.

References

[1] T. Xiao, D.F. Wong and J. Zhu, "A Loss-Augmented Approach to Training Syntactic Machine Translation Systems", *International Journal of IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol.24, No.11, pp 2069-2083, 2016.

[2] T. Tran, and D.T. Nguyen, "Method of Mapping Vietnamese Chunked Sentences to Definite Shallow Structures", In: *Proc. of International Conf. on Advanced Computing and Applications (ACOMP)*, pp. 74-80, 2016.

[3] F.R. Lopez, L.L. Arevalo, D. Pinto, and V.J.S. Sosa, "Context Expansion for Domain-Specific Word Sense Disambiguation", *International Journal of IEEE Latin America Transactions*, Vol.13, No.3, 784-789, 2015.

[4] B. Hettige and A.S. Karunananda, "A Morphological analyzer to enable English to Sinhala Machine Translation", In: *Proc. of IEEE International Conf. on Information and Automation*, pp. 21-26, 2006.

[5] J.R. Bellegarda, "Part-of-Speech tagging by latent analogy", *International Journal of IEEE Signal Processing*, Vol. 4, No. 6, pp 985-993, 2010.

[6] Z. Li., M. Zhang, W. Che, T. Liu, T and W. Chen, "Joint Optimization for Chinese POS Tagging and Dependency Parsing", *International Journal of IEEE/ACM Transactions on Audio, Speech, and Language Processing*, Vol.22, No.1, pp 274-286, 2014.

[7] H. Redkar, S. Bhingardive, D. Kanojia, P. Bhattacharyya, "World WordNet Database Structure: An Efficient Schema for Storing Information of WordNets of the World", In: *Proc. of* AAAI, pp. 4290-4291. 2015.

[8] M. Hwang, C. Choi and P.K. Kim, "Automatic enrichment of semantic relation network and its application to word sense disambiguation", *International Journal of IEEE Transactions on Knowledge and Data Engineering*, Vol.23, No.6, 845-858, 2011.

[9] W. Xiong and Y. Jin, "A new Chinese-English machine translation method based on rule for claims sentence of Chinese patent", In: *Proc. of International Conf. On Natural Language Processing and Knowledge Engineering (NLP-KE)*, pp. 378-381, 2011.

[10] S. Mall and U.C. Jaiswal, "Innovative algorithms for Parts of Speech Tagging in Hindi-English machine translation language", In: *Proc. of IEEE International Conf. On Green Computing and Internet of Things (ICGCIoT)*, pp. 709-714, 2015.

[11] S.C. Pammi and K. Prahallad, "POS tagging and chunking using decision forests", In *IJCAI Workshop on Shallow Parsing for South Asian Languages*, pp. 33-36, 2007.

[12] R. M .K. Sinha and A. Thakur, "Machine translation of bi-lingual Hindi-English (Hinglish) text", In: *Proc. of International Conf. On 10th Machine Translation summit (MT Summit X), Phuket, Thailand*, 149-156, 2005.

[13] B. Hettige and A.S. Karunananda, "A Parser for Sinhala Language-First Step Towards English to Sinhala Machine Translation", In: *Proc. of IEEE First International Conf. on Industrial and Information Systems*, pp. 583-587, 2006.

[14] S. Chaudhury, A Rao, and D.M. Sharma, "Anusaaraka: An expert system based machine translation system", In: *Proc. of IEEE International Conf. on* Natural Language Processing and Knowledge Engineering (NLP-KE), pp. 1-6, 2010.

[15] R.S. Sugandhi, R. Shekhar, T., Agarwal, R.K. Bedi, and V.M. Wadhai, "Issues in Parsing for Machine Aided Translation from English to Hindi", In: *Proc. of IEEE International Conf. On* Information *and Communication Technologies (WICT)*, pp. 754-759, 2011.

[16] B.R. Ambati, S. Husain, S. Jain, D.M. Sharma, and R. Sangal, "Two methods to incorporate local morph syntactic features in Hindi dependency parsing", In: *Proc. of IEEE International Conf. On NAACL HLT 2010 First Workshop on Statistical Parsing of Morphologically- Rich Languages, Association for Computational Linguistics*, pp. 22-30, 2010.

[17] S. Venkatapathy and S. Bangalore, "Discriminative machine translation using global lexical selection" *International Journal of Transactions on Asian Language Information Processing (TALIP)*, Vol.8, No.2, 2009.

[18] J. Srivastava, and S. Sanyal, "A hybrid approaches for word alignment in English-Hindi parallel corpora with scarce resources", In*: Proc. of IEEE International Conf. On Asian Language Processing (IALP)*, pp. 185-188, 2012.

[19] M. Halder, M. and A.D. Tyagi, "English-Hindi Transliteration by Applying Finite Rules to Data before Training Using Statistical Machine Translation", In*: Proc. of IEEE International Conf. On IT Convergence and Security (ICITCS)*, pp. 1-4, 2013.

[20] P. Saktel, and U. Shrawankar, "Context based Meaning Extraction for HCI using WSD algorithm: A review", In: *Proc. of IEEE International Conf. On Advances in Engineering, Science and Management (ICAESM)*, pp. 208-212, 2012.

[21] M. Qiao, W. Bian, R. Y. D. Xu, and D. Tao, "Diversified hidden Markov models for sequential labeling", *International Journal of IEEE Transactions on Knowledge and Data Engineering*, Vol.27, No.11, pp 2947-2960, 2015.