# Design of Machine Translator Based on QNN

*A Thesis*

*Submitted in fulfilment of the requirements for the*

*award of the degree of*

## DOCTOR OF PHILOSOPHY

*Submitted by*

**Ravi Narayan**

(Registration No:- 950803010)

Under the Supervision of:-

**Dr. V.P. Singh**
Asstt. Professor
Computer Science & Engineering Department,
Thapar University, Patiala.

**Dr. S. Chakraverty**
Professor and Head
Department of Mathematics
National Institute of Technology, Raurkela

**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING,**

**THAPAR UNIVERSITY,**

**PATIALA -147004, PUNJAB, INDIA**

**October, 2014**

# CERTIFICATE

This is to certify that the Thesis entitled **"DESIGN OF MACHINE TRANSLATOR BASED ON QNN",** which is being submitted by Mr. Ravi Narayan (Registration No. 950803010) to the Department of Computer Science and Engineering, Thapar University, Patiala, Punjab, INDIA, in fulfilment of the requirements for the award of the degree of DOCTOR OF PHILOSOPHY, is a record of bonafide research work carried out by him under our guidance and supervision. The matter presented in this thesis has not been submitted either in part or full to any other University or Institute for award of any degree.

Supervisors

**(Dr. V.P. Singh)**
Thapar University,
Patiala, Punjab, INDIA

**(Dr. S. Chakraverty)**
National Institute of Technology,
Raurkela, Odisha, INDIA

# ACKNOWLEDGEMENTS

I would like to express my gratitude to my supervisors for their valuable guidance during the research work. As my supervisor, I would also express my great appreciation with regard to Dr. V. P. Singh for his criticism and suggestions helped me to establish the overall direction for the research and contribution immensely for the success of this research work. Foremost, I express my gratitude to my Ph.D supervisor Prof. S. Chakraverty, Professor and Head, Department of Mathematics, National Institute of Technology, Raurkela, Odisha, India. During this research his continuous guidance nurtured me a lot. The valuable knowledge, which I gained from him, is valuable for me. As my supervisor his insight, observation and quick guidance helped me to establish the overall direction for the research and contribution immensely. His broad knowledge, interest, enthusiasm and constant encouragement has been essential in my development as a researcher. I am thankful for his patient guidance and encouragement during this doctoral degree.

I am grateful to the Thapar University, Patiala, Punjab, INDIA, especially to the Department of Computer Science and Engineering for providing me the opportunity to do this research work.

I am very thankful to the Director, Thapar University for providing me university resources and facilities necessary to carry out this research work.

I am also grateful to Dean (Research and Sponsored Projects) for providing me university resources and facilities necessary to carry out this research work.

I am also grateful to Dr. Deepak Garg, Head, Department Computer Science and Engineering, for his valuable guidance, help and assistance, given to me during the research work.

With the great pleasure and deep sense of gratitude, I acknowledge Dr. Maninder Singh former Head, Department Computer Science and Engineering for his support, encouragement and guidance.

I express my gratitude to the Doctoral Committee comprising Dr. Deepak Garg Head, CSED, Prof. Seema Bawa, Prof. R. K. Sharma and Dean (Research and Sponsored Projects) for monitoring the progress and providing valuable suggestions for improvement of my research work. I am also thankful to the entire faculty and staff of CSED for healthy discussions, suggestions and necessary cooperation to complete this work.

With deepest gratitude I dedicate my PhD research work to my Father, Er. Sushil Kumar Sharma, my Mother Mrs. Asha Sharma and My Aunt, Mrs. Sarla Sharma for the unconditional continuous support and guidance given. Without them this would have been impossible to successfully complete this work.

I would like to give thanks to my Brother in-law Mr. Anshul Goswami for his time and help in using statistical tools to process the huge amount of data generated during this research work.

I must give big thanks to my beloved younger sister Ms. Renu Goswami and brother Rohit Narain for their continuous help in editing this thesis for successfully completing this work.

I also want to give thanks to my friends and loved ones for their understandability during the days I ignored them due to busy in this research work.

Above all, I pay my reverence to the almighty God as well as to my beloved grandparents (Late.) Mr. Ram Narain Sharma and (Late.) Mrs. Rajeshwari Devi, whose blessings bring this attainment.

Ravi Narayan

(Ravi Narayan)

## CONTENTS

## LIS T OF FIGURES

# LIST OF TABLES

# NOTATIONS AND ABBREVIATIONS

| | |
|---|---|
| MT | Machine Translation |
| ANN | Artificial Neural Network |
| QNN | Quantum Neural Network |
| RBMT | Rule based Machine Translation |
| EBMT | Example based Machine Translation |
| $\theta^{r}$ | Quantum Interval |
| r | Quantum Level |
| $\delta_j$ | Error Rate |
| $\eta$ | Learning Rate |
| $\delta_k$ | Error Rate of Output layer |
| $\delta_j$ | Error rate of Hidden layer |
| $O_j$ | Output of hidden layer |
| $O_k$ | Output of Output layer |
| $W_{ij}$ | Weights between Input and Hidden layers |
| $W_{kj}$ | Weights between Hidden and Output layers |
| $n_s$ | Number of Excitation levels |
| BP | Brevity Penalty |
| $p_n$ | Precision Score |
| BLEU | Bilingual evaluation understudy |
| NIST | National Institute of Standard and Technology |
| ROUGE-L | Recall-Oriented Understudy for Gisting Evaluation- Longest Common Subsequence |
| METEOR | Metric for Evaluation of Translation with Explicit Ordering |

| | |
|---|---|
| $C$ | Total number of fragmentation (of one or more words) between reference translation and candidate translation, |
| $u_m$ | Total number of common words between reference translation and candidate translation. |
| $X$ | Reference translation |
| $Y$ | *C*andidate translation |
| $M$ | *T*otal number of words in reference translation |
| $n$ | Total number of words in candidate translation |
| $w_c$ | Total number of words in candidate translation (t) |
| $w_r$ | Total number of words in reference translation(r). |
| $P_{lcs}$ | Precision |
| $R_{lcs}$ | Recall |
| $LCS(X,Y)$ | Length of a longest common subsequence of $X$ and $Y$. |
| $Len_{Can}$ | Total number of words in candidate sentence |
| $Len_{Ref}$ | Average number of words in all reference translation. The |
| $\beta$ | Brevity Penalty Factor. |
| $N$ | Number of words in longest candidate translation. |
| $n$-gram | sentence having n number of words |

# Abstract

Machine translation (MT) along with natural language processing (NLP) always remained an area of interest for researchers since the computers were invented. Many researchers have tried to build the system which can understand multiple languages to translate from one source language to another target language. They also searched the way how computer understand and generate the human languages with semantics and syntactic. However, they realized that still many languages have translation difficulties, grammatically and semantically. Machine translation is a field of natural language processing. It involves the complete linguistic analysis of sentence used for automatic translation from one language to another. The main challenging issues need to be addressed are word ambiguity, word order, word sense, idioms, pronoun resolution, syntactic ambiguity and structural ambiguity.

Recently some work has been done with Hindi to English and vice versa by several researchers using different methods of machine translation, like example based system, rule based, statistical machine translation, and parallel machine translation system. Some researchers have described the use of corpus pattern for alignment and reordering of words for English to Hindi machine translation using the neural network, but still there are a lot of possibilities to develop a MT System for Hindi to increase the accuracy of MT.

This work presents the machine learning based translation system for Hindi to English and vice versa, which learns the semantically correct corpus. The quantum neural based pattern recognizer is used to recognize and learn the pattern of corpus using the information of parts of speech of individual word in the corpus like a human. The system performs the machine translation using its knowledge gained during the learning by inputting the pair of sentences of Devnagri (Hindi) and English. To analyze the effectiveness of the proposed approach, 2600 sentences have been evaluated during simulation and evaluation.

The accuracy of proposed quantum neural network based machine translation system for Devanagari (Hindi) to English has been compared on different scores viz. BLEU, NIST, ROUGE-L, METEOR and human based evaluation, the accuracy are respectively 0.7502

on scale of 1, 6.5773 on scale of 10, 0.9233 on scale of 1, 0.5460 on scale of 1 respectively and 98.154 %. In case of English to Hindi MT system the accuracy achieved on BLEU, NIST, ROUGE-L, METEOR and human based evaluation respectively are 0.9809 on scale of 1, 7.3066 on scale of 10, 0.9887 on scale of 1, 0.9655 on scale of 1 and 98.261%. The accuracy of proposed system for both Hindi to English and English to Devanagari (Hindi) are found to be significantly higher in comparison with the existing English to Devanagari (Hindi) and Devanagari (Hindi) to English MT system like Google and Bing, Artificial Neural Network (ANN) based MT system and Anuvadaksh. The proposed system also learns and recognizes the Parts of Speech (POS) Tagging Pattern of English and Hindi corpora using the Quantum Neural Network based pattern recognition. To analyze the effectiveness of the proposed approach, 2600 sentences of news items having 11500 words from various newspapers have been evaluated. During simulations and evaluation, the accuracy of 98.40% for English POS Tagger and 99.13% for Hindi POS tagger has been achieved, which is significantly better in comparison with other existing approaches for Hindi and English parts of speech tagging.

# CHAPTER 1
# INTRODUCTION

Chapter 1

# Introduction

Languages play an important role in enrichment of human civilization. By the aid of language, humans can easily express their feelings, views and communicate with each other as well as share their knowledge to others. According to the United Nations Educational, Scientific and Cultural Organization (UNESCO), there exist more than 6000 languages today. In India, 22 Indian languages are recognized as official by Government of India.Despite of language differences, people still want to communicate with persons who use different languages. Sometimes different languages become a barrier for cross-cultural communications.

Although English has been a language used by most part of the world, but still most of the world population does not have sound knowledge of English.

Hindi language is spoken by more than 500 million Indians and is recognised as National language of India. Hindi is morphologically rich language and relatively free word order language. Therefore many permutations of the same sentence convey similar meaning (Garje *et al.* 2013).

Recently, few works have been done with Hindi by researchers using different methods of Machine Translation (MT), like example based, rule based, statistical machine translation, parallel machine translation, neural network based English to Hindi MT *etc*., but still there are a lot of possibilities to develop MT System for English to Hindi and vice versa with better accuracy (Raman *et al*., 1997, Dave *et al*., 2001a, 2001b, Sinha *et al*., 2004, Sinha *et al*., 2005, Kumar, 2005, Dutta *et al*.,2010, Dwivedi *et al*.,2010, Shahnawaz *et al*,2012).

Machine translation can fill the gap of barriers in cross-language communications. The machine translation along with Natural Language Processing (NLP) always remained an area of interest for researchers since the time when computers were invented. It involves the complete linguistic analysis of sentence used for automatic translation from one language to another (Indurkhya *et al.* 2010). The main challenging issues need to be addressed are word ambiguity, word order, word sense, idioms, pronoun resolution, syntactic ambiguity and structural ambiguity. Many researchers have tried to build the system which can understand multiple languages to translate from one source language to another target language (Brown *et al.*,1996; Puscasu *et al.*,2004). They also searched the way how computer understand and generate the human languages with semantics and syntactics. However, we may realize that still many languages have translation difficulties, grammatically and semantically.

MT is the process to translate source language to target language using the aid of computer program. Main steps to translate one language to another are shown in Figure 1.1.

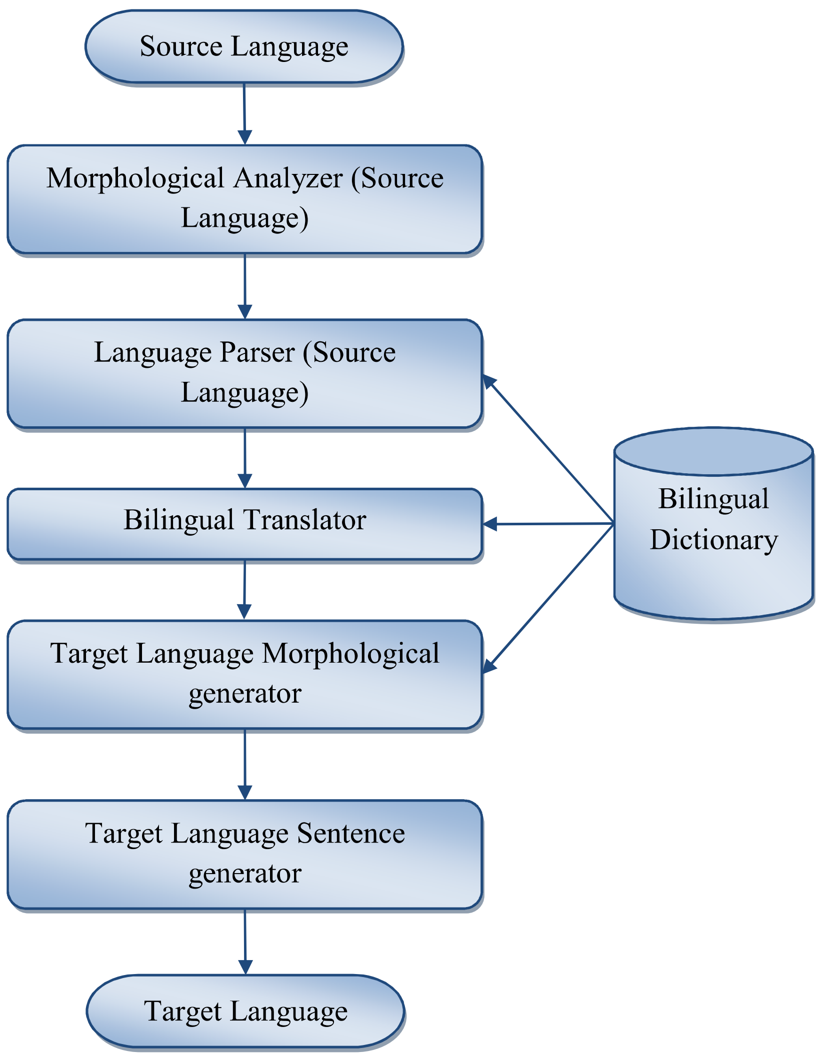


**Figure 1.1: Steps of Machine Translation**

To translate the sentence of source language to target language, the source sentence passes to MT system.

The Morphological Analyzer (MA) processes the input source sentence based on morphological analysis and identification of the structure of a given language. For every word present in the input sentence, the morphological analyzer gives the output for each token in the form of token based information like morphemes, gender, number and possessive suffix, affixes, Parts Of Speech (POS), intonations and stresses.

The Language Parser parses the morphological analyzed source sentence into words using the tokens and other morphological information. After this step the system has all the language specific knowledge, like Parts of Speech with token and other morphological information, needed for Bilingual Translator.

The Bilingual Translator translates every word of source language to target language.

The morphological analyzer identifies and assigns the correct morphological information to every word based on the structure of target language structure. Bilingual Translator uses the language semantic and semantic knowledge to generate the target language sentence as shown in Figure 1.1.

Machine translation along with natural language processing (NLP) is always remained an area of interest for researchers since the time when computers were invented. Many researchers have tried to build MT Systems for translating from one language to another language. They also searched the way how computer understand and generate the human languages with semantics and syntactic. However, they realized that still many languages have translation difficulties, grammatically and semantically.

MT involves the complete linguistic analysis of sentence used for automatic translation from one language to another. The statistical machine translation approach was inspired by noisy channel model and this was introduced by research community as a statistical tool (Zens *et al*, 2002; Koehn *et al*, 2003; Och *et al*, 2001).

Most of the machine translation researchers have done their work in the field of high exactitude in alignment of words in sentences (chunk). The translation model based on the structure, and the relationship of the two languages can be produced after aligning the parallel text upto the level of words and phrases. More refined models generally define their probability distributions over flanking phrases. In a sentence the appropriate meaning of the word may be chosen only by considering the whole sentence, because all the words in a sentence are related to adjoining word. Bilingual dictionaries can be used as a good source for directly pulling out the word correspondences. The attractive approach applied is to align higher level of syntactic or semantic structures, in addition to aligning individual words and contiguous strings or 'phrases'. Each language has its own pattern to align the parts of speech in any sentence. Chandola *et al* also explained the method to utilize the corpus pattern for alignment and reordering of words for English to Hindi machine translation (Chandola *et al*, 1994).

## 1.1. Research Gap

Different approaches do exist to solve the problem of Machine Translation. All the available approaches have some drawbacks too. Sometimes the existing approaches do not give accurate results and also are, time consuming. Problems associated with the available Machine Translation are:

- Given words may have different meaning. In every language many words have multiple meanings and each meaning is meaningful to its own scenario i.e. the meaning is ambiguous. Some examples of ambiguous words are given below, in which the word "shoot" is common in sentence 1, 2 and 3 but having different meanings. Similarly the word "plant" is also an ambiguous word, which is common in sentence 3 and 4 but having different meanings.

Sentence 1: Shoot with Gun.

Sentence 2: Shoot with Camera.

Sentence 3: Shoot of a Plant.

Sentence 4: I have to Plant a Tree.

- In Syntactic ambiguity single sentences can be parsed in different ways. The examples of Syntactic ambiguity are given below:

Bob saw the man on the mountain with a telescope.

This sentence may be parsed by two ways. On the bases of parsing the sentence may be interpreted as:

a. Bob saw the man on the mountain and the man was having the telescope.
b. Using the telescope Bob saw the man on the mountain.

No such system exists that can handle Morphological Processing, Syntactic Analysis, Semantic Analysis, Context processing *etc.* at a time with good accuracy. Recently Shahnawaz *et al.*(2012) have given their important contribution to MT system. They have implemented the ANN based MT system for English to Hindi and achieved the accuracy on blue score is 0.604, METEOR is 0.830.

## 1.2. Aim of Research

All the available MT approaches having some drawbacks which affect the accuracy of MT System. During study of different approaches we found that Hybrid Machine Translation System may perform better than other existing MT approaches. HMT systems combine the advantages of the individual approaches for achieving the quality of Machine translation system. On the other hand for Hindi, no such available MT system exists, that can handle Morphological Processing, Syntactic Analysis and Semantic Analysis etc. at the same time with good accuracy. To overcome the above discussed issues/ drawbacks with the Hindi- English MT System, the quantum neural network based hybrid machine translator for Hindi to English and vice versa has been proposed here.

In this research work, the multi-hierarchical approach has been used to process the complex long sentence. In first step by using the conjunctive words the long complex sentence is divided into several small parts. Each part then translated separately to get the final translation of the complex long sentence and recombined the divided sentences.

Quantum Neural Network (QNN) based machine translation system is potential solutions to this problem, as this has ability to learn from examples by recognizing their pattern. In this research work, the main focus is to enhance the accuracy. The proposed approach uses quantum neural network for reordering of words for POS tagging and their alignment during the MT. As shown in the Figure 1.2.the conceptual architecture of proposed QNN based MT System, the raw sentence first passes through the Syntactic Translation module. The syntactic translation is used when the analysis of language structure is necessary to translate source language to target language, which has different structure. Under the syntactic translation, the raw sentence first passes through Tokenizer. The Tokenizer splits the sentence into words and indexes it as token and then the resulting words with token, pass through the rule based POS Tagger. The rule based POS tagger tag the POS by simply using the Lexicon. The outcome of the Rule based POS Tagger is not perfect, for correction and accuracy it finally passes through the QNN based POS tagger, which refines the identified rule based POS using the pattern recognition of corpus. Here the QNN is used for pattern recognition of corpus to identify and correct the POS tagging. For learning purpose, some manually tagged sentences are input in the QNN based POS tagger, on the bases of input tagged sentences, the QNN based POS tagger learns all the patterns of POS tagging, then the tagged sentence passes through Rule Based Grammar Analyzer, to identify the appropriate tense rule of Grammar. On the bases of identified grammar rule, the sentence partially translated without semantic translation and word alignment. The Outcome of this step further passes through Sentence Mapper (QNN and Rule Based Sentence Mapping module) and then to the Semantic Transition module to achieve the translated (target language) sentence.

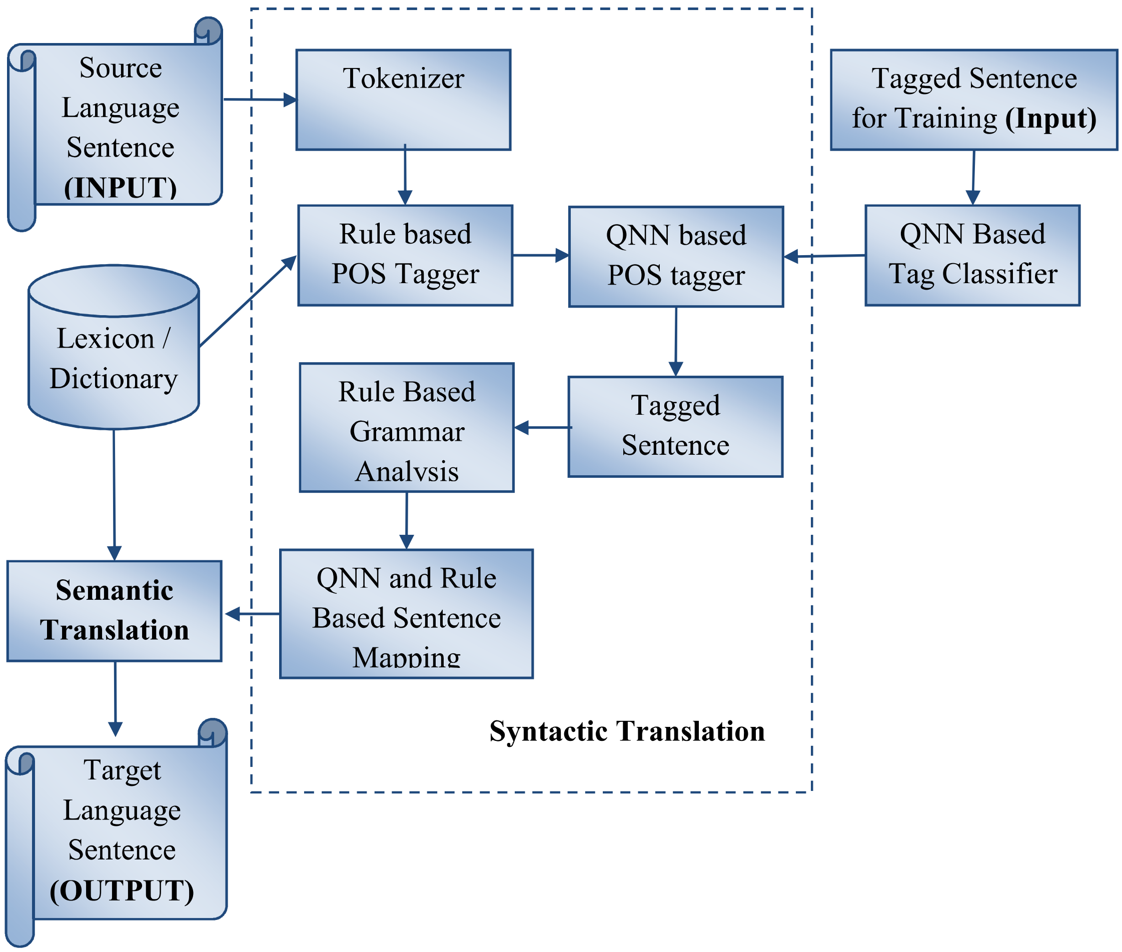


**Figure 1.2: Conceptual architecture of Proposed QNN based MT System**

In proposed machine translation (MT) system, the main objective of semantics translation is to transfer the literal meaning from the source language to the target language, word by word. Without the correct meaning of every word in the corpus, it is impossible to translate one language to another. Thus the semantic translation plays a very important role in machine translation. For semantic translation each word of source language has been directly mapped using the lexicon/ dictionary for its associate meaning of target language to achieve the final translation.

### 1.3. Objectives of the Research

In view of the above, following are the main objectives of this investigation:

**Objective 1.** To explore various models of machine translation from English to Devanagari and vice- versa.

**Objective 2.** To design Quantum Neural Network model for machine translation.

**Objective 3.** To identify the optimal learning rule /algorithm.

**Objective 4.** To test and validate the proposed model.

All the objectives are accomplished successfully and those are explained in the thesis. Table 1.1 shows the objective wise distribution of Chapters.

**Table 1.1 Objective wise distribution of Chapters**

| Objective | Chapter |
|---|---|
| **1)** To explore various models of machine translation from English to Devanagari and vice-versa. | **1)** Introduction<br>**2)** Overview of various models/approaches of Machine Translation<br>**3)** Background knowledge of English and Hindi Language Structure. |
| **2)** To design Quantum Neural Network model for Machine translation. | **4)** Quantum Theory and Quantum Neural Network<br>**5)** Proposed Quantum Neural Network based model for Machine translation. |
| **3)** To identify the optimal learning rule /algorithm. | **6)** Optimal learning rule for Proposed QNN based MT. |
| **4)** To test and validate the proposed model. | **7)** Validation of the proposed QNN based MT model<br>**8)** Conclusions and Future Scope |

### 1.3.1. Organization of the Thesis

**Chapter 1: Introduction**

As mentioned above, the goal is to design a machine translator based on QNN which translates Devanagari (Hindi) to English and vice versa. In First chapter, introduction about the subject along with the basic knowledge of machine translator is given. It is known that the field of machine translation is still having many problems to address, like word sense ambiguity, structural ambiguity, inaccurate parts of speech tagging *etc*. This chapter explains the challenges and problems of MT fundamental aspects of the MT.

**Chapter 2**: **Overview of various models/approaches of machine translation**

In this chapter, the literature survey was carried out, and extensive overview of various recently developed models/approaches of machine translation has been included. Detailed review of existing Hindi to English and English to Hindi machine translator systems has been given and discussed. In this chapter the technological advancements of different MT approaches are also mentioned.

**Chapter 3: Background knowledge of English and Hindi Language Structure**

As it is known that the English and Hindi both are grammatically rich languages and for translating Hindi to English and English to Hindi, one needs to understand both the languages at structural level. As such this chapter explains the structure of English and Hindi Languages at grammatical level. It also gives the structural differences between the mentioned languages and the challenges in translation due to their structural differences at grammatical level.

**Chapter 4**: **Quantum Theory and Quantum Neural Network (QNN)**

Before explaining QNN model for machine translation, this chapter addresses first the basic concepts of quantum theory with traditional artificial neural network. Quantum theory is the body of scientific principles which expresses the behaviour of matter and its relations with energy on the level of atoms and subatomic particles, and how these phenomena may possibly be connected to daily life. The fundamental principles of quantum theory are superposition principle, measurement principle and unitary evolution.

QNN has been developed by various researchers following the basics of quantum theory. As such the QNN training method, architecture *etc*, are discussed in this chapter.

**Chapter 5**: **Proposed Quantum Neural Network based model for Machine translation.**

This chapter includes details of QNN model for machine translation. The complete architecture of the model is developed. Moreover, the method used to implement the machine translation by using the QNN is also included. The coding mechanism is used to decode and encode the sentences into numeric code to pass these to QNN. This chapter also covers the QNN based pats of speech tagging in detail.

**Chapter 6**: **Optimal learning rule for Proposed QNN based MT**.

In this chapter, the optimal learning rule /algorithm for proposed QNN based MT is achieved and explained with methodology in this chapter. Simulation has been performed with QNN. Results are also compared between traditional and quantum neural networks. The optimum rules are different for English to Hindi translation compared to Hindi to English translation. The rule algorithm for parts of speech tagging (intermediate step of MT) for Hindi and English are also identified. Optimal learning rules have been achieved after 500 independent tests performed with the system for each value of quantum interval (θ) with random data sets.

**Chapter 7**: **Validation of the proposed QNN based MT model**

This chapter addresses the tests which are performed on the proposed system for validation with a highlight on its underlying ideas and intrinsic advantages. The performance of proposed system is evaluated together with other existing systems like Bing translation, Google translation, neural network based MT and Anuvadksh. The proposed system is compared with existing system for validation on the scale of NIST (National Institute of Standard and Technology), BLEU (Bilingual Evaluation Understudy), ROUGE-L (Recall-Oriented Understudy for Gisting Evaluation- Longest Common Subsequence) and METEOR (Metric for Evaluation of Translation with Explicit Ordering) scales.

The accuracy of proposed QNN based MT system for Devanagari (Hindi) to English has been compared on different scores viz. BLEU, NIST, ROUGE-L, METEOR and the accuracy is respectively 0.7502 on scale of 1, 6.5773 on scale of 10, 0.9233 on scale of 1 and 0.5460 on scale of 1. In case of English to Hindi MT system the accuracy achieved on BLEU, NIST, ROUGE-L, METEOR and human based evaluation respectively is 0.9809 on scale of 1, 7.3066 on scale of 10, 0.9887 on scale of 1, 0.9655 on scale of 1 and 98.261%. The accuracy of proposed system for both Hindi to English and English to Devanagari (Hindi) are found to be significantly higher in comparison with the existing English to Devanagari (Hindi) and Devanagari (Hindi) to English MT system like Google and Bing, ANN based MT system and Anuvadaksh.

**Chapter 8**: **Conclusions and Future Scope**

In this chapter the conclusion and future scope are given in detail.

# CHAPTER 2

# LITERATURE SURVEY

# Chapter 2

# Literature survey

Literature survey was carried out for the study related to Machine Translation (MT), parts of speech tagging, quantum neural network and other adoptive algorithms. In this chapter, various models of machine translation from English to Devanagari (Hindi) and vice-versa are explored. The technological advancement of different MT approaches are identified. MT is a complex procedure, because language translation is non deterministic in nature.

## 2.1. Machine Translation Approaches

Many researchers have tried to build the MT systems, which may translate from one source language to a different target language. Different approaches have been used by researchers for MT, but all the approaches having strengths and weakness. Some of the significant models / approaches are discussed here.

Komeili *et al.* (2011) investigated the translation problem occurred in the English to Persian Machine Translation (MT). They introduced the frameworks which have the capability of eliminating the issues of English translation. The Machine translation software like Padideh, Pars and Google and bilingual dictionary of Hezareh were explored by him. There several issues are involved in the available MT softwares such as lexical problems, word conjugation and ambiguity problem, syntactic problem and problems on the level of production and transmission.

### 2.1.1. Rule-based Machine Translation

Rule Based Machine Translation (RBMT) Approach works by using the linguistic rules, the three types of rules are used at different steps: analysis, transfer and generation. The Rule based machine translation requires: syntax analysis, semantic analysis, syntax generation and semantic generation. The rules and the bilingual dictionaries are used for translating one language to other language. The Parts of Speech (POS) tag and dependency information are used in this approach, which are obtained from the parser.

Due to the rich use of rule-base, these systems are called as Rule-based systems (Okpor, 2014).

Dorr (1993) described the interlingual MT system UNITRAN. This system works on two Massachusetts developed approaches of theoretical linguistics, viz. Chomskyan principles with parameters Government Binding (GB) hypothesis for the syntactic part and Jackendovian Lexical Conceptual Structure (LCS) for the lexical – semantic part, which moreover work as Interlingua. A simple systematic translation mapping with no language specific rules has been done on the basis of lexical-semantic divergences between source and target languages. The main problems with their system are, the UNITRAN system which was only accurate upto 80% and needed 30 to 50 Seconds to translate average length sentence. They found that, by increasing the number of constraints into the precompilation phase, the required time to translate sentence also increased.

Fung *et al.* (1996) introduced an algorithm for translating the technical words and termsfrom noisy parallel corpora across language groups. When a word related to the technical word is given to the algorithm from the source language then the algorithm finds the right candidate match for it in target language. Potential translation for the term is also compiled from the matched words and is also ranked. The algorithm is independent to language and character-set and the advantage is that, it is more robust to noise in the corpus. It is able for the translation of technical words without sentence alignment and sentence boundary identification. For the compilation of technical terms bilingual lexicon was used to help the translator. It achieved 55.35% precision in the technical term for AWK corpus (English to Japanese) and 89.93% precision for the HKUST corpus (English to Chinese). The main problem of their system is that, their system skips some words during translation. An iterative process discards unlikely candidate words. Means, the output sentence of their system as target language sentence might not be a complete translated sentence.

Puscasu (2004) introduced multilingual method for clause splitting. The first step in building the hierarchical structure of sentences was the combination of language independent machine learning techniques with specific rules of language. The rule-based module deals with the clause boundaries that were not included in the learning process. The method was computed for the Romanian and English languages. The clause splitting problem was mixed with the methods that could be easily ported to other languages. The method mixed the language independent machine learning techniques with the language specific rules. The system worked by taking the part-of-speech text as an input and then processes each sentence. The coordinating conjunctions and the punctuations marks were considered as ambiguous delimiters. The method was computed on Romanian and English and F-measure for the clause start detection for Romanian is 95% and for English is 92%.Their results are lower for English as compared to Romanian due the reasons that, in many cases clauses annotated as finite subordinate clauses lacked a finite verb. On the otherhand in some cases a coordination of finite verbs was not separated by a clause boundary, as required on the basis of basic definition of the system. Some errors were also increased due to misclassification in the learning process.

Zhang *et al*. (2006) introduced the combination of machine translation with speech recognition. The main issues involved in the speech recognition were to reduce the negative effect of errors produced in speech recognition on MT. For improving the speech translation quality the new statistical MT decoding algorithm was used. The speech recognition word lattice was translated by the algorithm. The approach was experimentally tested on the speech translation task of Japanese-to-English and the translated results were measured in terms of the number of automatic evaluation metrics. They achieved the accuracy on BLEU scale was .5600, on NIST 6.1800, on GTM 0.6600, on mWAR 0.4800 and on mPER 0.3800. The only problem with their system was that, their system needed huge amount of data for learning the system, to achieve the high accuracy.

Rajan *et al.* (2009) introduced RBMT approach for translating English to Malayalam. In this work the rules and the bilingual dictionaries are used for translating source language into target language. The rules used in this method are based on the POS tag and other relevant information collected from parser. Two different rules are used in this technique, first is transfer link rule and second is morphological rule. The transfer link rules generate the target structure and morphological assigns morphological features. The main limitation of their RBMT system is that, the system can only translate the sentence having upto six words. The main problem with their system is that, the system is unable to handle the ambiguous words. According them the problem of ambiguous words could be addressed by using the rich Word Dictionary.

Mridha *et al.* (2010) introduced the approach for Morphological rules for Bangla root, verbal suffix and primary suffix for the Universal Networking Language (UNL). The framework of the UNL was used within the development of Bangla enconversion. To improve the quality and maximize the development efficiency, the lingware method was proposed as a technique. In two states of India and Bangladesh, Bangla language is spoken by about 245 million people. However, the study was limited only to few number of words of Bangla. The main problem with their system is that, they did not use rich Bangla Word Dictionary and complete set of rules.

**Major problems with the RBMT:**

There are some problems with the RBMT. The RBMT system requires individual language analyzer and language generator for each language, i.e. source and target language. The analysis of target language requires complex semantic analysis based on predefined complex rules, which requires extensive coding of predefined complex rules. To make any change in rules is very difficult and time consuming and requires lots of programming efforts. The analysis of source language requires rich set of word knowledge. Practically the true meaning of the sentence may not always be extracted. Another disadvantage of RBMT systems is, its ineffectiveness in word alignment, when the source and target language have totally different language structures.

### 2.1.2. Example-based Machine Translation

The Example Based Machine Translation (EBMT) was proposed by Nagao, 1984. They first introduced the concept of using the input pair of source and translated sentence as example to generate translation for a given input sentence. As discussed in Rule based machine translation, in the traditional MT system, the machine translation of source language into the target language is done with the help of rules. That's why these systems are known as Rule Based Machine Translation (RBMT). In RBMT it is very difficult to improve translation quality because the machine translator is using a large scale rule based, and which is also very time consuming. To reduce the problems of RBMT, the EBMT was developed. The EBMT has many advantages over the RBMT such as EBMT has no rules and the use of examples is comparatively localized. Secondly EBMT has a high translation speed than RBMT. In EBMT appropriate translations for given domain have been obtained using domain specific examples. This EBMT system translates the

Japanese noun phrase into English noun phrase. Their system has achieved the accuracy of 78% in the experiment (Sumita *et al.* 1991).

Brown *et al.* (1996) introduced the Pangloss Example Based Machine Translation. In pangloss EBMT, several translation engines run in parallel to process various portions of the input, so it is known as a multi engine translation system. This translation system does not require structural knowledge. In this the input texts are divided into series of words that are occurred in the corpus for which the translations are obtained. The EBMT requires minimal knowledge which helps in quick retargeting which is an advantage. As the size and quality of the bilingual dictionary and synonyms lists are decreased, the quality and number of translation are degraded in the EBMT. The main drawback of Pangloss EBMT is its bad performance, when the source language and target language words are not one-to-one.

Veale *et al.* (1997) introduced a template driven bootstrapping approach for Example Based Machine Translation (EBMT). EBMT approach provides robustness, scalability and graceful degradation. They introduced the architecture of EBMT system which is known as Gaijin. Gaijin is used for the translation of German and English. The Gaijin system contains following stages namely, bilingual Corpora alignment, automatic lexica construction, transfer template generation, example retrieval, example adaptation and new example acquisition. Gaijin executes the alignment of bilingual sentence based on the information collected from the source and target word corpora. The Gaijin achieved is 63% accuracy, with the dataset of 791 tests sentences.

Cicekli and Guvenir (1996, 1998, and 2001) introduced the method for learning the translation templates using examples. The learning has been performed at lexical level between two languages by using the set of paired translated sentence. The similarities and differences in the different parts of sentences have been learnt in the way of translation templates. The knowledge acquisition process in the conventional Knowledge Based Machine Translation (KBMT) is time consuming as well as costly. The purpose of the method was to automate the retrieval of the needed knowledge from the MT. The method was based on the simple pattern matcher. The main goal of this method is to make it compatible learning from examples *i.e.* the way by which human learns the algorithm.

Brown *et al*. (1999) had refined Pangloss EBMT system under the Pangloss and DIPLOMAT projects. The DIPLOMAT EBMT system finds the entire occurrence within the example base of any contiguous phrase from the input and also processes the overlapping partial exact matches. The basic matching algorithms are used to search the contiguous occurrences of successive words in the input. The strengths of this approach are very quick at run-time. In this approach for building the parser, no linguists are required and examples are generated by the translator only if they are not already available.

Carl *et al.* (1999, 2003) had explored and compared different EBMT experimentally. They had done experiments to show the comparison between the outcomes of three corpus based MT systems. The automatic evaluation technique is used to judge the results of String Based Translation Memory (STM), Lexeme Based Translation Memory (LTM) and EBMT. According to their experiments, when near matches were found in database then STM shows the higher translation precision and EBMT performed better than the other two MT systems, if the reference corpus doesn't have any similar translation. Their experiments also showed that, their EBMT system was capable to decompose and simplify the translation sentences, translate parts or single words for recombining into the target language sentence.

Somers *et al.* (1998, 1999) reviewed the Example Based Machine Translation (EBMT). The main knowledge- base stems are acquired from the examples. Generally examples are used as a shortcut to gather the knowledge in the EBMT. They found out the limitation of EBMT that, sometimes examples act as a bottle neck for the Rule Based Machine Translation. The original idea of EBMT was based on the RBMT paradigm. Examples were stored in the form of tree structures and therefore rules are needed to analyze these examples. According to this paradigm the transfers are done on the basis of examples. The three main components are used in EBMT. Matching fragments are used for matching against the database of real examples, after that it identifies the equivalent translation parts and again combines the divided parts to achieve the target language. The Case Based Reasoning (CBR) systems have achieved the accuracy of 90% of their optimum from the 1000 housing domain case base size examples. However the accuracy was improved by 4% when further 1000 cases were added. They explained the problems with the CBR systems which are the problem of addition of cases in the system automatically as well as manually and the problem of storing multiple similar or even identical examples that affected the case base learning.

Brown *et al.* (2001) introduced the transfer-rule for Example-Based Translation (EBMT). In EBMT three approaches for generalization are used namely manually generated correspondence classes, automatically-extracted correspondence classes along with transfer rule generation. In the manually generated classes, the information from the machine readable dictionary with the information of parts-of-speech is used to change the words with the tokens that are indicating the class of words. In the automatically-extracted equivalence classes word level clustering is used. In the transfer rule induction approach, some assumptions are needed. According to these assumptions when two sentence pairs in corpus contain some part are common but different in the other part then the similar and dissimilar parts correspond to some coherent part.

Somers *et al.* (2001) used EBMT for Case Based Reasoning (CBR) paradigm. This work described the possible applications of CBR in Machine Translation. Comparison between EBMT and CBR was performed to see the similarities to push forward the EBMT. By using CBR terminology, the EBMT was explained as some features of the standard process-cycle details for CBR are not very well described in EBMT literature. The main difference between both is that the complexity of examples representation in CBR as compared to EBMT. The CBR is more like to RBMT model. It rarely copies one minimum case to set for search and capable of using good retrieval, reuse and revision techniques. In place of doing more research on CBR, researchers are using new technologies for the burning issues of linguistic knowledge representation for EBMT, viz. linking of discontinuous Source Language (SL)-Translated Language (TL) sentence elements for indexing and recombination of TL fragments in the face of boundary friction. The CBR techniques interact with retrieval in the area of adaptation is well developed and is also shown by Dublin group (Collins and colleagues) whereas the less developed issues of CBR for learning adaptation and integration of domain knowledge as well as repair and retain strategies might be accomplished with the use of EBMT. The Case Based Reasoning (CBR) systems have achieved the accuracy of 90%.

Kit *et al.* (2002) introduced the Example Based Machine Translation (EBMT). In EBMT there are four stages which are known as example acquisition, example base management, example application along with target sentence synthesis. The example acquisition stage acquires example from parallel bilingual corpus, the example base management is used to store and maintain the examples. The example application uses the examples to facilitate the translation involving the breakdown of an input sentence into examples as well as the conversion of source texts into the target texts. The target sentence synthesis is used to combine the target sentences. This was done by using the transformed examples into a smoothly readable order. The main limitations with their EBMT System are user interface, example filtering in addition to pattern inference, learning and usage modelling etc. The EBMT systems have achieved the accuracy of 96%, and for unknown terms the accuracy was 92%. The problems with the EBMT are acquiring inadequate knowledge for reliable disambiguation.

**Major problems with the EBMT:**

EBMT approach also has some problems like, the performance of an EBMT system depends on the quality of collected examples and the structural similarity between examples and input sentences. The other most important problem with EBMT is Quantity, Suitability and structure of example base. Means, the way in which the examples stored and used, may also affect the number of examples needed by the EBMT system. (Mima et al. 1998)

### 2.1.3. Statistical Machine Translation

In Statistical Machine Translation (SMT) approach, the MT is done by using statistical models. The parameters of statistical models are derived after analysis of bilingual text corpora. In Statistical machine translation, based on the input pair of sentences(Language A and Language B), the Translation Model and, Language Model are derived from the statistical analysis of bilingual test corpora, which further pass through the Decoding algorithm for generating the Statistical model for Machine translation (Koehn, 2003, 2009). The SMT is very effective in word alignment of a translation corpus, multilingual document retrieval, automatic dictionary construction and data preparation for word sense disambiguation (Brown et al., 1990);

Brown *et al.* (1990) presented the statistical approach to French to English. The history suggests that statistical methods were used in various fields like automatic speech recognition, lexicography and natural language processing and machine translation. The translator, in machine translation used to translate the meaning of source language into the target language. Their MT System has achieved the 84% accuracy. However they found that the errors were increased rapidly while increasing sentence length, which affected the accuracy of the system.

Brown *et al.* (1991) introduced a statistical technique for aligning sentences in parallel corpora. The statistical technique is used in the Machine Translation (MT) for one language to other language. The extraction of parallel corpora sentences from parallel English and French languages *i.e.* translation of one another is considered as a problem. The problem of extracting parallel corpora sentences was not minor, because the translation pattern of different languages is different as one sentence can be translated in two or more sentences in different language. For calculating the alignments in the sentences, only numbers of tokens are used as information. Because no lexical details are used, so computation of alignment is fast. In a random selected set of 1000 sentence pairs, it achieved an accuracy of 99%. The expected error achieved in the approach is 0.9%. The correlation between lengths of aligned sentences was strong enough to achieve the accuracy between 96% and 97%.

Brown *et al.* (1992) introduced the Machine translation based on Markovian model, based on phrases instead of words, the words also attached with a phrase-to-phrase translation table. In this method, translating a text-amount to its appropriate translation, is depending on the obtainable parameters. Inferring the parameters of this model from bilingual corpora is a matter of statistics. By model inference, the task of extracting is performed for all tables, parameters and functions from the corpus. Brown *et al.* (1993) introduced the mathematics of statistical Machine Translation and gave five translation models that are used to translate the source language to target language. The main focus is in the word by word alignment between the sentence pairs. These models assigns a probability for every possible word by word alignments in the given sentence pairs. Their system works only for two languages *i.e.* English and French. They have compared the accuracy of old and new systems. The accuracy of old system was 39% and new system has the accuracy of 60%.

Gale *et al.* (1991, 1992a, 1992b) introduced a method for Aligning sentences in Bilingual corpora. This method was working on the concept that longer sentences in one language are likely to be translated into same length sentences in target language, on other hand the shorter sentences likely to be translated into shorter sentences. Based on the scaled difference of lengths of the two sentences a probabilistic score was assigned to each proposed correspondence of sentences. The accuracy of the system was 60% and in 95% of cases the words were found correct. In their later research they modified their method of alignment of words, resulting they have achieved the best scoring of 80% accuracy in alignment.

McEnery *et al.* (1994) introduced the use of estimated String matching techniques in the arrangement of sentences in parallel corpora. Parallel corpora were provided an ideal test-bed for many tasks like training of translation and production of the probabilistic dictionaries. For using, firstly those were aligned because it was known, which segment in one corpus correspond with which segment in the other. Parallel corpora were provided an ideal test-bed which was used for many tasks like training of translation and production of the probabilistic dictionaries. The algorithm was re-implemented and achieved the success rates of 98% for 100 sentences of the English and French section, for news items in English and German were 75% and for short passages in English and Polish were varying from 69.5% to 100%. They have achieved the accuracy of 98% for 100 sentences of the English/French section of ITU corpus, the accuracy for news items in English and German was 75% and for short passages in English and Polish the accuracy was 69.5% and 100%.

Wu *et al.* (1995) introduced the extraction of an English-Chinese Translation Lexicon on large amount. Using limited amounts of linguistic knowledge, the statistical analysis of a large parallel corpus was used in the automatic extraction approach. The first experimental results were for Indo-European and non-Indo-European language designed for any significant vocabulary and corpus size. The learned vocabulary size was approximately 6500 English words. It achieved a translation accuracy varying in the range of 86-96%, with alignment at paragraph, sentence along with word levels. In monolingual lexicon, the average of learned translation lexicon was 2.33 Chinese translations per English sentence. The precision achieved in the manually filtered precision and automatically filtered weighed precision were 95.1% and 86.0% respectively. The main limitation of their system is that, they had used hand-derived monolingual lexicon, which needs extensive human efforts to build.

Nieben *et al.* (1998, 2000, 2001 and 2004) presented a method for evaluation the translation quality. They have discussed the need of such a method for MT research. The key feature of this method is to carry out the quick semi-automatic, suitable and dependable way with the help of graphical user interface. The evaluation was also defined for more sufficient than pure edit distance with the measurement along these quality criteria. Regular experiments were carried out to control the development of Translation system by the MT research group at the University of Technology, Aachen. The outcome was evaluated on three different test sets viz., the Verbmobil corpus with suddenly spoken dialogs in the domain of appointment scheduling and the other were EuTrans 2 Zeres corpus texts from touristic domain. The different translation methods in terms of Subjective Sentence Error Rate (SSER) showed 17% – 26% for Verbmobil-147, 57% – 76% for EuTrans-closed and 42%– 59% accuracy for EuTrans-open. The higher complexity of the EuTrans corpus results in higher SSER.

Och *et al.* (2001, 2002, 2004 and 2006) introduced the discriminative training and maximum entropy models for Statistical Machine Translation (SMT) and presented a framework for SMT for natural languages. This framework was based on the direct maximum entropy models. Those models contain the source channel approach as a special case. The problem associated with that framework was handling the complex features in searching. The knowledge sources associated with that framework were considered as feature functions. These functions were dependent upon the source language sentence, target language sentence and possible hidden variables. It was also referred as the source channel approach for the SMT. Och et al. (2002), the comparison was done between the 6 models i.e. IBM-1, IBM-2, Hidden Markov Model (HMM), IBM-3, IBM-4 and IBM-5. The HMM model has shown the better results than IBM-2; however the performance was improved by involving the IBM-3. The first order dependence and a fertility model could increase improved results than simple models IBM-1 or IBM-2 by using the sophisticated alignment models.

Dorr *et al.* (2002) introduced a technique known as DUSTer designed for cross language Divergences for Statistical word level arrangement. The proposed technique was used to identify the common divergences and convert an English sentence structure into another language to carry a closer resemblance. The objective of DUSTer is to allow more accurate alignment and estimation of dependency trees in other language with no training data on dependency tree data. When the different gist of a sentence is distributed in different words in different language then divergence occurred. The DUSTer translates the English sentences into the pseudo English sentences which closely matches with the physical form of the foreign language. The approach is applied to all divergence types. By the use of templates, divergence has been handled by converting the syntactic structures of the English sentences to bear the resemblance. The future work includes the automation of process for detection of the divergence. The accuracy of the system was 95% with the Spanish sentences and with Arabic divergences the accuracy was 39%.

Germann *et al.* (2001, 2004) introduced the algorithm for quick and best possible decoding for Machine Translation. For success of any SMT a good decoding algorithm is important. The decoder was used to search the translation that is similar to a set of earlier learnt parameters. The SMT system in the MT framework contained three components. The first component is a Language Model (LM), the work of LM is to assign a probability P(e) to any given string e of English. The second component is the Translation Model (TM), the work of TM is to assign a probability P(f | e) to the given pair of strings e and f of English and French language. The third component is the decoder. The task of decoder is to perform the search by using the decoding algorithm. There are some cases for choosing the correct algorithm. In first case, if the source and target languages contain the same word order then linear Viterbi algorithm was used and if word ordering is limited to nodes in a binary tree, then high-polynomial algorithm was used. For simplicity the experiment was setup so that all decoders worked on similar search space. They found the usefulness of multiple decoders at the same time for machine translation decoding and the multiple decoder can only be used through the Integer Programming(IP) output.

Fai *et al.* (2006) introduced MT based on the Translation Corresponding Tree (TCT) structure. The TCT structure is applied to the Portuguese to Chinese MT system is an annotation schema. The TCT shows the flexible framework which was used to describe the relationship among the inner levels of structure against its substrings and is also used for the translation equivalences. In the MT partially automated strategy was used to build the bilingual knowledge base. Each TCT structure is used to elaborate the syntactic structure of the sentences as well as the translation correspondences. It factorized the input sentences into the structure of stored examples. The approach used for storing the translation examples is nearly related to the match searches. All these approaches add examples using a pair of analyzed structure found that in EBMT equivalent relationships between source and target sentences has been recognized at the structural level for constructing the translation alignments based on strict constraints, which may indirectly limits the MT systems for free-word-order languages.

Roark and Saraclar (2004a, 2004b, 2006 and 2007) introduced the discriminative n-gram language modelling. It elaborated the language modelling for the large vocabulary speech recognition task. The evaluation is considered as a very helpful function, using N-gram occurrences statistics, the evaluation requires an evaluation corpus of source material along with high quality reference translation. The MT quality of IBM algorithm is measured in terms of a weighed sum of counts of matching N-grams. The scoring has been done by computing the fraction of N- grams in the test translation that occur in the reference translation. The scoring is performed segment-by-segment, the minimum units of translation coherence (usually one or few sentences) are known as segments. The capability to forecast human judgment of quality is the sine quanon of any automatic MT score. The connection between human judgment and N-grams scores were above 90% for all the comparisons, with the exception of the fluency score for information is used to integrate Hierarchical Reordering Model (HRM) for source phrases. This model has shown significant improvement in accuracy in comparison to other systems like PBSMT and HRM systems using lexical hierarchical reordering model for English-Vietnamese pair. Based on their previous research they have improved the accuracy on the BLEU score from 35.39 to 36.76.

Andres-Ferrer *et al.* (2008) introduced the different loss functions in statistical pattern recognition applied to MT. Statistical machine translation needs alignment mapping of words among the source and target sentence. On one hand alignments are used for training the statistical models and during the decoding of words in the source sentence to the words of target sentence According to them the powerful way in pattern recognition is to deal with the classification problems. These word arrangement problems are based on the minimization of classification risk. To measure the penalty for wrong decisions in terms of loss functions is known as risk function. For each different function in the loss function there is an optimal Baye’s rule. There are different optimal classification rules, depending on the loss function of system. It expresses direct translation rules and inverse translation rules. The theoretical approach in decision theory elaborates the similarities and differences in the direct translation rules and inverse translation rules.

Rodriguez *et al.* (2008) introduced the method of evolutionary algorithms for alignment problem in SMT. In SMT, the mappings among the words in source and target sentences are known as alignment. The purpose of using the alignments is for training the statistical models and for decoding. The statistical method was used to translate the sentence from source to target language by using the parallel corpus. The parallel corpus was described in terms of set of pairs. The each pair has the sentence of source and target language. They found that, due to the limitations of the statistical alignment model, their system had not performed alignment accurately even though using the improved search algorithm.

Deng *et al.* (2008) introduced a procedure for Hidden Markov model Word and Phrase alignment for SMT. The assessment along with alignment for the words arrangement of parallel text for word and phrase alignment HMM. Alignment is the main issue in the SMT. The alignment process is used to identify translation equality between documents. Through various alignment experiments, these components have been analyzed. They found that exact computation of posterior statistics under IBM-4 models may be used for phrase-pairs extraction from the word aligned parallel text which may improve translation performance.

Ortiz-Martinez *et al.* (2008a, 2008b) introduced the scaling problem in the pattern recognition approach to MT. SMT was characterized by two significant advances, Firstly the phrase based statistical translation models, which allowed incorporating contextual information to the MT system. Other perspective is the accessibility of huge parallel corpora, made of huge amount of pairs of the sentences, huge amount of running words. For dealing with the scaling problems in SMT without introducing considerable time, a general framework was used by grouping the multiple scaling techniques. The model was totally or partially transformed the Random Access Memory requirements of this scaling techniques into hard disk requirements. The comparison of conventional estimation system with fragment by fragment estimation system has shown that the conventional system needed 2GB memory requirement when the maximum phrase size was equal to 8. This estimation was impossible on the 32bit machines whereas fragment by fragment system only needed 0.12 GB memory requirement. The translation system needed only 0.2 seconds to translate the sentence.

Tillmann *et al.* (2008) introduced an online Relevant set algorithm for SMT. The SMT decoder was a new procedure which was used to directly optimize global scoring function. The SMT decoder was considered as a black box by the training procedure and it was able to optimize the tens of millions parameters automatically which made it better from other approaches. They found that reach ability of reference translation for a given block set can improve present algorithm that is the oracle block sequence may not be good translation, just for the reason an excellent translation may be impossible through the present block set.

Costa-jussa *et al.* (2009) introduced an N-gram Based Reordering (NBR) model. The powerful technique *i.e*. Statistical Machine Translation (SMT) was used in the NBR model which was used for generating the weighed reordering graph. The SMT technique which was used in NBR, converts the source language corpora into the intermediate language representation, in which the order of the words of source language were similar to that of the words of the target language. The NBR model was made of bilingual units, the bilingual units contain the reordering information. Those units were known as tuples. During the training phase NBR model reduces the vocabulary sparseness of N-gram based SMT system. For training and test cases, the NBR approach was used as a pre-processor. The NBR approach has the capability of generating the rearrangement hypotheses of sequences of words and also it provides for each reordering hypotheses a smooth context based weight. It was done through the well developed language model method. The computational cost of the system was increased by introducing reordering abilities. They found that NBR technique is well enough to produce rearrangement method of sequences of words which had not been seen during training. The main problem they identified that, by establishing rearrangement capabilities, there was an increase in system computational cost. The main limitation with their system was its outperformance during the reordering with the NBR technique.

Khaliov *et al.* (2009a, 2009b, 2011) introduced the syntax based rearrangement for SMT. To handle the problem of word ordering in the SMT, they introduced the method which is known as syntax based reordering. The various challenges in the SMT included the placement of words in proper sequence in the target language. The problem in monotone SMT was that it was capable to produce accurate word-by-word translation but positions of one semantic and syntactic unit in source language were appeared at different positions in the target language. The Syntax Based Reordering (SBR) method combined the SBR in phrase-based and N-gram based SMT. In the pre translation step, the SBR model had several steps. In the first step the system automatically learned the difference between the word order of a source and target language. In the second stage, rules had applied to change the structure of the source language sentence which closely matched the target language word order. In the final step the quality of the translation was improved by the SBR algorithms with word alignment- based reordering framework. The accuracy achieved for all extracted rules was 59.3% and 57.8% which was only one of its kind for the BTEC Arabic- to-English along with for Chinese-to-English translation tasks was 67.8% and 66.09% accuracy using the NIST.

Dutta *et al.* (2010) introduced the probabilistic method for identification of demonstrative pronouns for indirect anaphora in Hindi. The approach was used to classification scheme of demonstrative pronouns using Probabilistic Neural Network (PNN) for Hindi corpus. Indirect anaphora was considered as a main issue, for resolving this problem needed special-purpose techniques. The need of semantic and word knowledge became the process of implementing the automated system complex. Due to the referring expression and antecedent, it didn't contain explicit relationship, indirect anaphora was difficult to resolve. By examining the method of creation, having the non referential demonstrative pronoun allowed building a set of detailed syntactic linguistic and semantic patterns associated with the non referential occurrences of demonstrative pronouns. The model has achieved the 94.54% accuracy with the single σ value equal to 0.8388 in training and the performance falls to 50.20% in cross validation. The method was effectively classified 61.75% instances. The model achieved the 63.85% success rate by assigning σ to each variable, which didn't show much improvement in cross validation. An improved performance of 81.97% was shown in PNN model by the optimized number of neurons. The performance of the model was improved by increasing the training dataset and enhanced characteristic assigned in training set.

Reddy *et al.* (2010) introduced the method for integrating the statistical models for dictation of document translation in a Machine Aided Human Translation (MAHT) task. The integration method combined the source with target language information for generating the source translation and target language document. The information was retrieved from the source language document, which consisted of the translation probabilities derived from the SMT. These entity tags obtained from the Named Entity Recognition (NER) were included with the acoustic phonetic information derived from an automatic speech recognition system. On a French to English document translation this combined system performance showed the decrease in word error rate 29%. For well formed and disfluent utterances the decrease in WER was 31% and 26%.

Komeili *et al.* (2011) introduced some methods for eliminating the issues involved in English and Persian languages. The process of automatically transferring the messages from one language to other is known as Machine Translation (MT). There were various issues involved in MT some of which are related with the employed strategy such that in transmission strategy a transmission stage is necessary. The other issue was the problem of syntax. It was defined as a combination of words together in order. Some problems were related to the lexical. Lexicon may be defined as the group of words and each word in the group was related to the syntactic category such as noun, adjective, verb *etc* and some issues included the problem at the level of production and transmission. The software which was based on the machine translation approach saved time, money as well as assisted the human beings. The results showed that the translator translated the fifty percent sentences of English language into Persian language. They had performed the experiment on different Machine Translator (MT) systems like Google, Pars, Padideh. These systems were able to translate 43% of English word into the Persian and 38% in syntax and conveying meaning. The systems have many problems like lexical problems, word conjugation and ambiguity, syntactic problems and problems at the level of production and transmission.

Raj *et al.* (2011) introduced the pattern based approach for Natural Language Processing (NLP). The learning in NLP could be performed on the basis of individual words, phrases and concepts. For developing the self learning text conversational entity or chatterbot called RONE was required to develop the suitable knowledge representation schemes and appropriate sentence dissemination methods. When RONE performed NLP, it was found that it was more effectively aid to its own vocabulary instead of making everything all at once.

Xiong et al. (2011) introduced the Maximum Entropy Segmentation (MES) model for SMT. Segmentation played an important role in the SMT. It was the process of splitting a source sentence into sequence of translatable sentences. The proposed maximum-entropy segmentation model captured the desirable phrasal and hierarchical segmentations for SMT. For building the segmentation model, this method used automatically learned the cohesive segments by using the word-aligned bilingual data from beginning to end. It had been done with no additional linguistic resources. The experimental results showed the improved quality of translation of MES model in terms of BLEU. Their experimental results have shown to facilitate the MES model was enhanced from syntactic constraints in capturing desirable segmentations. The segmentation model was statistically significantly performs on the newswire domain and broadcast news domain and achieved the 1 and 1.06 BLEU point respectively.

Zyglarski *et al.* (2011) introduced the neural networks aided automatic keyword selection. The two approaches were used, *i.e.* statistical based and neural network based. These approaches contained the complete description and differences of two approaches. The neural network based approach was divided into two types *i.e.* simple neural network and neural network with reinforcement. The simple statistical methods have given the poor results, in best case the accuracy was 65% and in worst case the accuracy was 5%. The kohonen networks gave the better result than statistical method without reinforcement, the accuracy in best case was 80% and 10% - 40% accuracy was in worst case. The best result was generated with Reinforced Kohonen Network, the accuracy was 90% and 10%-40 % with worst case.

Silvestre-Cerda *et al.* (2012) introduced the explicit length modelling for SMT. In length modelling the length of the sequence was indirectly captured. The translation systems that were grounded on the phrase based models implicitly model the length information of sentence by some features like word and phrase penalty which were used to control the number of words and phrases in the resulting translation. The goal of SMT was to perform the automatic translation between languages, based on SMT examples. Generally the phrase based systems are based on the large bilingual phrase dictionary which is also known as phrase tables. The phrase based tables did not cast the conditional phrase length correlation.

Zhi-ying *et al.* (2012) introduced machine translation for Chinese to English. They did not get the right translation result by which the understanding of full text was greatly affected. It traversed the chance of changing the sentences to passive voice from the perspective. These were based on the Hierarchical Network Concept (HNC) theory. The properties of passive voice in both Chinese and English documents were examined and summarized the three states of the sentences when translating Chinese into English in processing the voice. The old system gave the accuracy with the active voice was 79.03%, the accuracy of passive sentences with grammar mark in Chinese was 91.25% and accuracy of passive sentences without grammar mark in Chinese was 11.43%. Whereas the new system gave the accuracy with active voice was 85.48%, the accuracy of passive sentences with Grammar mark in Chinese was 92.17% and accuracy of passive sentences without grammar mark in Chinese was 82.46%. These transformation algorithms were very useful in producing the better translation quality.

Ananthakrishnan *et al.* (2013) introduced the batch mode semi supervised active learning for SMT. The Batch-mode active learning maximizes in domain coverage, this was done by selecting sentences that shows the balance between the translation difficulty, batch diversity and domain match. This approach automatically learns the method to choose from the large monolingual pool. The selection technique improves the performance of SMT at a faster rate than the existing selection methods. The difference between the active learning and others is that in active learning approach firstly we utilize a sample of candidate pool instead of additional in-domain development set. Secondly it constructs the batches. It could be done by using incremental greedy selection strategy with parallel ranking rather than the traditional batch rank and select method. They found that, in general the iterative active learning's cost is equal to the re-training of the SMT system for every batch. By providing the Small batches system it may achieve the smooth performance trajectories along with improved error recovery. Their approach has achieved an improvement of 45.9% in BLEU over the seed baseline, while the closest competitor gained only 24.8% with the equal number of sentences.

Balahur *et al.* (2014) performed comparative experiments using supervised learning and machine translation for multilingual sentiment analysis. They proposed an extensive model for semantic analysis. Working with the translated data, it increased the number of features, sparseness and noise in the data points in the classification tasks. Three classification techniques were used to limit these problems. It had found that, the union of same training data translated with different systems helped the classifiers, which helped in learning of various linguistic aspects by using same data, which was noticed in the case of good translation quality. The dependency of the proposed approach was based on the availability of translation engines for the required languages. However commercial engines had the capability to translate from and into a large no. of languages, but the problem was translating the large amount of data. The future work included the syntax information which helped in reducing the effects of translation errors. Their results have shown the little variation in the performance of the sentiment analysis system using English and translated system. The maximum drop in the worst case was 11.8% using SMO, 11.5% with AdaBoost and 8% with Bagging. Between systems trained on English along with translated data the gap in classification performance was minimal i.e. maximum 12% in favour of source language data.

**Major problems with the SMT:**

The main problem with the Statistical machine translation is the excessive requirement of Corpus, which may be difficult to create and also costly to users with limited resources. On other hand the results are unexpected. The most important problem of SMT is that the Statistical machine translation does not work effectively with the languages that are having significantly different word orders.

### 2.1.4. Corpus-based Machine Translation

Gale *et al.* (1991b) introduced an approach for Aligning sentences in Bilingual corpora. It describes a technique for the alignment of sentences that was based on the simple statistical models of character lengths. The evaluation has done using the trilingual corpus of economic reports. Although it was possible to retrieve a large sub corpus, that had a much smaller rate. By selecting the best-scoring 80% of the alignments, the error rate was reduced from 4% to 0.7%. The approach was very simple and accurate and a very less error rate *i.e.* 4.2% out of 1316 alignments. It was possible to achieve 80 % of the alignments from the subsets with the error rate of 0.7%. The performance of the approach was better when it was used with the characters instead of using the words. The future work includes the approach used is extended to build the use of lexical constrains.

Grishman *et al.* (1992) introduced two approaches for the machine translation (MT) *i.e.* Rationalist and empiricist. It also included the intermediate approach that included the both approaches. To produce parses and syntactically regularized tree structures the parallel corpora of both source and target languages were analyzed. The individual trees of source and target languages were aligned and yielding a correspondence between both languages. It also involves specific words. The rationalist system consisted of seven layers for sentence analysis as well as structural regularization. It was followed by the transfer and generation of an Interlingua. On the other hand, Empiricist methods had tried to gain the correspondences from large paired billingua corpora. The advantages of empiricist were that because the method was not limited by deficiencies of the manually developed deeper analysis methods.

Watanabe *et al.* (1992 and 2000) described the approach for finding structural correspondences by using the paired dependency structures of source and target language. At initial step, the system finds the correct word correspondence and then works on phrasal correspondences based on word correspondences. The user would check whether the received word correspondence was correct or not with the help of Graphical User Interface (GUI) system and got the correct one. Besides this, a GUI system was also developed for manual. They have achieved 98.03%. The experimental result showed that they have achieved the high accuracy. The method showed highly accurate precision of word correspondences and some categories of phrasal correspondence.

### 2.1.5. Knowledge based Machine Translation

Bennett (1990) recognized the necessity of semantics for Machine Translation systems. Since the beginning of research in the MT, the semantics became the issue. The need of dealing with semantics was recognized by the Weaver (1949). The purpose of author was to examine deeply the necessity of semantic analysis in MT systems. The desirability of a semantics in Machine Translation system was non controversial. The basic presumption for having a viable MT system was that one must have some sort of full-blown semantics. For handling all the peculiarity of a source language the deep semantic component was needed for the MT systems. In the list of issues of MT systems, ambiguity was the top issue. The main issue with their MT system was verb semantics which was the difficult one, differentiating verb meaning and eliminating erroneous verbal complements and anaphora resolution from a semantic standpoint.

Raj *et al.* (2009) introduced the learning for the available information and storage for chat based conversational systems'. RONE is a tele-text based conversational system. The

Structured Query Language (SQL) and accessed using the main Java application was used to make the RONE's knowledge base. The performance of RONE was compared with the Ultra Hal, Jabberwacky and ElBot which is 100%, 44.4% and 0% respectively. RONE had various features such as the process of fetching the answer for item, the RONE could fetch the answers in, two ways *i.e.* in the first way, to form the answer, it required to obtain a piece of information from its knowledge base and second way was to compare the information regarding the question and responds to Yes or No. Therefore RONE had the capability to think in different ways. RONE was based on the formula based computation. For compilation process RONE uses the separate grammar module.

### 1.1.6. Hybrid Machine Translation

As discussed above RBMT, EBMT and SMT have some problems. It is not possible to produce highly accurate system by using single MT approach. The Hybrid Machine Translation (HMT) system was proposed to combine the advantages of the individual approaches for achieving the quality of MT system. The main motivation behind the HMT systems is to overcome the failure of any single technique and to achieve the satisfactory level of accuracy in Machine Translation. Hybrid machine translation is a method of machine translation in which multiple MT approaches are used in a single MT system. The approach is very useful to remove the translation ambiguity problem of Rule-Based MT. On other hand the Statistical Machine Translation (SMT) may be the ultimate method to identify the most suitable option when word/phrase has more than one meaning. As shown in Figure 1.5.in first step the Translations are performed using rule based system, then for refining the outcome of RBMT, it passes through SMT. This is also known as statistical smoothing and automatic post editing. It may be possible that, instead of using RBMT with SMT, the RBMT may be used with EBMT or with some other approach. Generally the RBMT is used with SMT in HMT. (Siddiqui and Tiwary. 2008).

Shirai *et al.* (1997) introduced a new example-based method of machine translation system in which the examples needed no direct translations. The system automatically ruled out unknown examples during translation and uses the currently available sentence aligned corpora as data. At suitable positions the rule based modules were used. Generally machine translations were divided into the Rule Based Machine Translation

(RBMT) and Example Based Machine Translation (EBMT). The RBMT consisted of a method to analyze the input sentences along with the method of generating the sentences which were based on the internal structure. These steps of RBMT were controlled by the dictionary and rules. The RBMT has the problems of dictionary and rules, these problems were reduced by the EBMT. The EBMT translated the sentences by acquiring the aligned translated example sentences and imitating the translation of sentences and resembled it. The hybrid translation model consisted of the advantages of both RBMT and EBMT. A prototype Japanese-to-English system had been implemented that allows multiple users to share corpora. The major advantage of using examples was the translation of idioms, literal or domain dependent, in other words the system was reversible and it translated in either direction. Earlier the possibility of large word or aligned corpora is supposed to be in many example based systems. The prototype was implemented which used the currently available level of the corpora to translate from Japanese to English. The prototype was tested with a corpus of 5000 sentences translating from Japanese to English.

Habash *et al*. (2002) introduced the approach for handling translation divergences by combining statistical and symbolic techniques in the generation of heavy machine translation. Large amount of explicit symmetric knowledge for both source and target language was the major requirement of these approaches. The handling approach did not require the transfer rule or interlingual representation. It depended on the lexical semantics, categorical variations and sub categorization from the target language. It described a way to handle the translation divergences by using the hybrid machine translation approach. Combination of complex lexical and structural mapping was required for the translation divergence problem which was reserved for Transfer and Interlingual machine translation. Extensive evaluation of the whole system was to be done with the comparative analysis of other models of Spanish - English MT and retargeting the system to Chinese input.

Sofianopoulos *et al.* (2010) introduced the multi-objective optimization method for hybrid MT system using Genetic Algorithms (GA). The pattern recognition approach collectively worked with monolingual corpora in target language from which statistical information were extracted. The MT automatic evaluation technique was evolved for computing the translation quality. These methods for automatic evaluation of MT were involved BLEU and METEOR. The most widely-used method was SMT and hybrid MT. The single objective elitist real valued GA was used as an optimization method which was used to adjust the parameters of the system. These parameters were guided by the different MT evaluation metrics fitness functions. The results showed the improvements from 20.35% to 41.2% by sing the BLEU score.

**Major problems with the HMT:**

The major disadvantage of HMT is that, It needs huge amounts of dataset of example pair, i.e. the set of source and translated sentence as translation example, which may be difficult to collect in huge number. The second main disadvantage is complexity, as HMT combines two or more different approaches, so it requires more computation power for machine translation (Labaka *et al.2014).*

## 1.2. Hindi-English Machine Translation Approaches

Recent work on Hindi to English and vice versa by several researchers using different methods of machine translation are given below in this section.

### 1.2.1. Neural Network based Machine Translation

Recently Shahnawaz *et al* (2012) had given their important contribution for English to Hindi MT system. They had implemented the ANN based MT system for English to Hindi and had achieved the accuracy on blue score was 0.604, METEOR was 0.830. They had implemented feed forward back propagation artificial neural network. The network was used as the knowledge base and for mapping from dictionary and linguistic rules. They found that, the efficiency of the system was increased by improving the case marking.

### 1.2.2. Hybrid Machine Translation

Dave *et al.* (2001a, 2001b) introduced the method for extracting the knowledge from Hindi text. All over the world approximately 4000 languages are spoken and Hindi is placed in the 5th position in the world. Accordingly they had focused on the need of processing of knowledge extraction of Hindi. The goal of knowledge extraction was to allow language access rather than machine translation. The human aided machine translation system known as MATRA which was used for mainly translation of English to Hindi. The processing of it was done in several ways, firstly it breaks the English sentences into parts which were known as chunks, then it analyses the structure and in the last step it displayed the intuitive browser like representation. The knowledge extraction system analyses the knowledge from lexicon and then translate the analysis rules. In knowledge extraction, the predicate preserving parser had 5200 rules. Out of these rules, 1200 rules are used for semantic analysis and 800 rules for morphological analysis. The main problem with their system is that, the system needs word-sense disambiguation modules. Due to this the main limitation of their system is that, their system cannot efficiently sense disambiguation for postposition markers.

Sinha *et al.* (2004) introduced ANGLABHARTI-II in 2004 which was a system based on the Generalized Example Base (GEB) with Raw Example Base (REB). At the time of development, the author established that the alteration in the rule-base was hard and the outcome was possibly random. This system consisted of error-analysis component and statistical language component for post-editing.

Sinha *et al.* (1984. 2003, 2005) developed Hinglish MT system in 2004 for standard Hindi to Standard English. This system was developed to incorporate the added enhancement to available AnglaBharti-II MT System for English to Hindi and to AnuBharti-II systems for Hindi to English translation. The accuracy of this system was satisfactory more than 90%. As the verbs had multiple meanings, it was not able to determine the sense, due to non-deep grammatical analysis. The performance of the AnglaBharti-2 system has improved from an average of 40% to an average of 80% correctness. The overall accuracy of the system was higher because each of the modules of AnglaBharti-2 architecture has configured to enrich itself with additional knowledge.

Kumar (2005) introduced the EB-ANUBAD translator. The translation scheme was based on the hybrid translation which was used for the translation of English to Bangla. The system took the input sentences in the form of paragraph of English sentences and translates the equivalent Bangla sentences. The EB-ANUBAD system was made of a preprocessor, morphological parser, semantic parser using English word ontology in support of context disambiguation, an electronic lexicon associated with grammatical information and a discourse processor, lexical disambiguation analyzer. The use of lexical processing was to establish meanings of every word. Syntactic analysis was used to deal with syntactic structure. It had achieved an accuracy of 98% that had been tested with various texts. The error found in the system was 1% - 2%. The system was designed for rural people to understand the English text. The system was easily upgradable with new grammar rules and lexicons. The cost of the system was low.

Dwivedi *et al.* (2010) introduced Machine Translation (MT) system in Indian perspective. Machine Translation (MT) systems in India are developed for translation of English to Indian languages. The MANTRA MT system (1999) was based on the synchronous Tree adjoining Grammar and used tree transfer for translation. The MATRA MT system (2004) had a text categorization component which determines the type of news story. On the basis of type of news, it used a suitable dictionary. The Hindi to Punjabi MT system (2010) was built on the basis of direct word-to-word translation approach. It contained parts like preprocessing, word-to-word translation approach, morphological analysis, word sense disambiguation and post processing. They found the need of important features that must have every MT system just as, the MT system should do correct manipulation of the terms and concepts of the domain.

### 1.2.3. Rule-based Machine Translation

Ananthakrishnan *et al.* (2006) introduced the MaTra which was based on transfer approach using a frame-like structured representation. In this the rule-based and heuristics approach is used to resolve ambiguities. The text classification module is used for deciding the category of news item before working in entered sentence. The system selects the appropriate dictionary based on domain of news. It requires human support for analyzing the input. After examining the structure, this system also divides the complex English sentence to easy sentences to produce Hindi sentences. This system is developed to work in the domain of news, annual reports, and technical phrases.

### 1.2.4. Statistical Machine Translation

Udupa *et. al.*, (2004, 2006) introduced the IBM-English-Hindi MT System developed by IBM India Research Lab in 2006. At the beginning of this project they started to develop an example based MT system but later on shifted to the statistical machine translation system from English to Indian languages. They proposed an English-Hindi SMT system based on IBM Models 1, 2, and 3. The system was evaluated using English-Hindi parallel corpus of 150,000 sentence pairs. They have proposed two novel algorithms for transferring the fertility parameters through Model 2 to Model 3. The worst case time complexity of their algorithms was O (m 3). They found that, when the maximum fertility of a word is small then their algorithms have O(m 2) complexity.

Rao *et al.* (2000) presented a method for Syntactic Transfer of compound- complex sentences for English-Hindi Machine Translation. The framework was described on the basis of template like representation. The mapping of finite and nonfinite verb groups was the major component of the framework, used to cover the simple and compound complex sentences. The mapping was non-trivial because of the differences in the style and structure of Hindi and English. The approach was easy to implement and maintain and it did not need very elaborate linguistic knowledge.

Och *et al.* (2007) developed the Google Translate. This model used the statistical MT approach to translate English to other languages and vice versa. Among the 57 languages, Hindi and Urdu are the only Indian languages present with Google Translate. Accuracy of the system is good enough to understand the sentence after translation (Brants *et al.* 2007)

### 2.3. Parts Of Speech Taggers

The main function of the tagger is to categorize words in a text properly into a finite set of syntactic categories. This process is indefinite as the mapping between words to the tag-space is often one-to-many. POS tagging is a difficult task with challenges like ambiguous Parts of Speech (Manning *et al.* 2002). Various approaches are used for POS tagging systems such as rule-based model, statistical model, and neural networks. The major disadvantages of rule-based and stochastic approaches are their inherent inability to deal with unknown words, *i.e.*, words that are not the parts of the training set.

#### 2.3.1. Probabilistic Tagger

Tufis *et al.* (1998) had introduced probabilistic taggers for Romanian corpus. In probabilistic taggers, there is a window of three words called trigram, where the probability of each possible tag for a current word is combined with the possibility of the tag is preceded by the two previously assigned tags. Here by using the training Romanian corpus of approximately 250000 words, the initial probabilities of tags are calculated, among which the tag, with highest score is selected and achieved the accuracy 95.63%.

### 2.3.2. Rule-based Tagger

Brill *et al.* (1992, 1994, and 1995) described that the in natural language processing, statistical techniques are better than rule based methods for automatic parts of speech tagging. They presented a simple rule based part of speech tagger. Their tagger is also called as Brill tagger. It acquired the rules automatically and tag with high accuracy in comparison to stochastic taggers. There are many advantages of rule based taggers over other taggers viz. it requires less information stored, clear small sets of meaningful rules, easily finds taggers and apply improvements to the tagger and easily change one tagset, corpus type or language to another. In Brill tagger the set of rules for determining the word tags as the initial set of naive tags are assigned to the corpus of words, after which transition rules are learned by correcting the falsely identified word tags. While tagging, these rules are applied for correct word tag. In their work, the major contribution is showing that only the stochastic method is not the only feasible method for parts of speech tagging but other methods are also there to do parts of speech tagging. Learning automatically is the key feature of Simple rule based tagger for better performance. It is a motivation for researchers to do explicit research on rule based tagging, simple parts of speech tagger performs well in comparison to existing stochastic taggers and has considerable advantages over these taggers. In a stochastic tagger, to capture contextual information, numbers of lines of statistical information are required in the form of table of trigram statistics. This feature makes the tagger more logical and helping in better understanding and simplifying further development of the tagger. The compact representation of contextual information is equally effective as of the information provided in the large tables of contextual information. The Brill tagger was trained on 600000 words and achieved the accuracy of 97.2% on tested with separate 150000 words from the same corpus. The results showed the accuracy of 99.0% with the average of 2.28 tags per word.

### 2.3.3. Hidden Markov model based Tagger

Cutting *et al.* (1992) introduced the practical Parts Of Speech (POS) Tagger. The implementation is based on the hidden Markov model. The POS tagger in the language processing system should have several properties. The first is, it should be robust *i.e.* it is necessary that a tagger should deal well in complex situations such as in some situations text corpora may contain ungrammatical, non linguistic data. The second is, it should be accurate and reusable. The approach allows robust and accurate tagging and the accuracy achieved was 96%. The POS Tagger used two types of training methods. The first is tagged training corpus, a relatively small amount of text is manually tagged and used to train a partially accurate model and in the second method training was not required to tagging training-corpus. The tagger was tunable which helped in removing the errors and anomalies and systematic tagging. The tagger was used in three applications i.e. phrase recognition, word sense disambiguation and grammatical function assignment.

According to Ortiz-Martínezb *et al.* (2008, 2012) the SMT used the pattern recognition for automatic machine translation systems for available parallel corpora. Statistical machine translation needs alignment mapping of words between the source and target sentence. The alignments are used for training the statistical models as well as, during the decoding process to correlate the words in the source sentence to the words of target sentence.They introduced the scaling problem in the pattern recognition approach to MT. SMT has been characterized by two significant advances. Firstly the phrase based statistical translation models, which allowed incorporating local contextual information to the translation models. Second perspective is the availability of huge parallel corpora. For dealing with the scaling problems in SMT without introducing significant time a general framework was used by means of the combination of the different scaling techniques. The model was totally or partially transformed the Random Access Memory (RAM) requirements of given scaling techniques into hard disk requirements. The comparison of conventional estimation system with fragment by fragment estimation system has shown that the conventional system needed 2GB memory requirement when the maximum phrase size was equal to 8 this estimation which was not possible on the 32 bit machines whereas fragment by fragment system only needed 0.12 GB memory requirement. The translation system needed only 0.2 seconds to translate the sentence.

Okhovvat *et al.* (2011) introduced Hidden Markov model based parts of speech tagger for Persian corpus. In this system, major features of Persian morphology were introduced and developed. Accuracy of 98.1% was achieved in experiments done on both homogeneous and heterogeneous Persian corpus.

### 2.3.4. Neural Network based Tagger

Sejnowski *et al.* (1987) introduced by Multilayer perceptrons based sliding window approach for parts of speech. They assumed that the letters have been pre-classified and recognized and these letter sequences comprising words are then inputted to the neural network during training and during performance testing. They found that more words the network learns, the better it is at generalizing and correctly pronouncing new words.

Schmid (1994a, 1994b) performed most prominent work and did his experiment on a large corpus called Penn Treebank Corpus. He trained the Net-Tagger using a context window. The context window consists of three preceding words and two succeeding words. He reported improvement of 2% over the statistical approach based on Hidden Markov Model (HMM). The accuracy of the net tagger was achieved 97.79% with the 4.6% of ambiguous words.

Qing *et al.* (1997) presented a POS Neuro Tagger having 3-layer perceptron with elastic input. In an experiment, with the training of 22,311 ambiguous words of Thai corpus, Neuro tagger showed an accuracy of 94.4% for tagging ambiguous words. In his comparative experiments, it was showed that Neuro tagger is certainly better than the statistical models including the frequency model, local n-gram model and HMM.

Ahmed *et al.* (2002) introduced the Multilayer Perceptron Network(MLPN) for Parts-of-Speech tagging. The MLPN with three layers was used. The error back-propagation algorithms were used to train the MLP tagger. The SUSSANE English tagged-corpus was used to train the tagger. The corpus was consisting of 156,622 words. The MLP tagger achieved an accuracy of 90.04% on the test data which was based on the tag mappings learned. The tagger was used to classify words in a text correctly into a finite set of syntactic categories. The Multilayer Perceptron(MLP) model had many benefits such as faster tagging rates, capability of handling unknown words and high modularity. They found that despite of using traditional tagged approaches the MLP tagger with combined representation schemes gave the better performance.

### 2.3.5. Conditional Random Fields based POS Tagger

Agarwal *et al* (2006). developed POS tagger based on conditional random fields for Hindi. In this approach the Hindi morph analyzer was used for training purpose and to get the root-word and possible POS tag for each word in the corpus. The training and testing is performed on the corpus size of 1, 50,000 words. The performance of the system was 82.67%.

### 2.3.6. Maximum Entropy based POS Tagger

Dalal *et al.* (2006), introduced the Maximum Entropy (ME) based POS tagger for Hindi, their approach requires the feature functions extracted from a training corpus. The average performance of the system is 88.4%. There is an increase in performance till it reaches 75% of the training corpus after which there is a reduction in accuracy due to over fitting of the trained model to training corpus. The least and best POS tagging accuracy of the system was found to be 87.04% and 89.34% and the average accuracy over 10 runs was 88.4%. The main limitation of their system is that, the system did't had the language specific features, particularly for chunking.

### 2.3.7. Morphological rules based POS Tagger

Singh *et al.* (2006) introduced the Morphological rules based POS tagger for Hindi and it was not designed for learning. Locally annotated modestly-sized corpora of 15,562 words were used in this system. The high-coverage lexicon and a decision tree based algorithm were used for morphological analysis. The POS categories identified by Lexicon lookup in this system. The performance of the system was evaluated by a 4-fold cross validation over the corpora of 15,562 words and found 93.45% accuracy. The major strength of their system was hand-coded learning rules to handle the disambiguation.

### 2.3.8. Memory based Tagger

Daelemans *et al.* (1996) introduced the memory based approach Parts-of-Speech (POS) Tagger generator. The memory based learning is based on the similarity-based reasoning. The learning approach is workable in terms of accuracy and computational efficiency.

They have performed the experiment on 200,000 test words. These test words were divided into two types i.e. known, unknown words. The accuracy achieved for the known words was 96.7%, for unknown words the accuracy achieved was 90.6%. The total accuracy of the tagger was 96.4%. The system has advantages like accurate generalization, incremental learning and automatic selection of optimal text etc.

### 2.4. Softcomputing Approaches

Morel (2000) introduced the biologically plausible learning rules for neural networks and quantum computing. It is supposed that Hebb's rule is tightly coupled with the biological learning. The Hebb's rule is implemented in a quantum algorithm is much faster. The origin of the approach is based on the Quantum entanglement. The biological learning rules are difficult. The reason behind that, may be the quantum entanglement having a neuronal equivalent. Artificial neural network (ANN) is extremely simplified as the architecture of brain. The advantages of ANN include the parallel processing and are powerful processors. Many algorithms and approaches were developed for ANN. The good example of ANN is back propagation, it is a very efficient algorithm. For dealing with the quantum states, quantum computers are needed rather than the classical computers which manipulate bits of information. The laws and principles of quantum physics show the efficient way for processing the information.

Narayanan *et al.* (2000) introduced Quantum Artificial Neural Networks architectures and components (QUANN). In terms of efficiency QUANN are more efficient and powerful in some situations than the classical artificial neural network (CLANN). QUANN are more powerful than CLANN which is measured in terms of what the network learns. QUANN have the capability to reduce and eliminate the catastrophic forgetting and also avoid problems like problem of interference of training patterns because one set of weights is trained for each pattern.

Gorecki *et al.* (2006) presented the parallel version of Quantum Dynamics (QD) for the cluster architecture. They represented an algorithm of a parallel version of QD in a wave function representation. Using the parallel version of a 3D FFT library it was optimized for the Linux cluster and also for the Cray T3E. With the optimization of parallel QD algorithms the efficiency in computing the numerical problems has increased. Although these problems are time consuming for the scalar computing and there is a possibility that these algorithms needed significant simplifications. The application area of the parallel QD algorithm is not limited but it is also used in the various fields such as proton transfer simulations, correlated proton transfer in quantum chemistry *etc*.

Theodorou *et al.* (2007) introduced the correspondence analysis with fuzzy data. They proposed the algebraic foundations for this fuzzy extension of the usual correspondence analysis. The two step method is described by him, to convert the fuzzy eigen value problem to an ordinary one. The data analysis methods were divided into two types, the top-down model and bottom-up model. In the top-down model, it presumed the probability model. The probability model produced the data. In the bottom-up model free methods have been used which belong to descriptive statistics.

Panigrahi *et al.* (2009). presented the use of genetic algorithm for the cantilever steel beams strength testing. In their work they have optimized the genetic search function for optimization procedure along with the residual force method to identify the macroscopic structural damage in a uniform strength beam.

Luitel *et al.* (2010) introduced the quantum inspired Particle Swarm Optimization (PSO) for the optimization simultaneous recurrent neural networks. The difficult problem was the training of a single Simultaneous Recurrent Neural Network (SRN) to learn outputs of a Multiple Input Multiple Output (MIMO). The effectiveness of the learning was improved by introducing the two step learning approach in the training. In achieving the goal of learning, the first step was to find the optimal set of weights in the SRN bearing all output errors. In the second step, the aim was to maximize the learning of each output dynamics by tuning the respective SRN output weights. The results showed that MIMO SRN could be successfully trained with better accuracy.

Panda *et al.* (2012) designed an interval Type-2 Fuzzy logic controller for automatic voltage regulator system. The interval value fuzzy sets were used for the representation of the membership of the system variables. The Fuzzy logics have the ability of modelling the uncertainty and imprecision in a better way. The performance of the controller was better than PSO optimized PID controller as the simulations suggested. The implementation was feasible because it did not need heavy computation.

Su *et al.*(2011) introduced Chaos quantum behaved particle swarm optimization based neural networks for short term load forecasting. For dealing with the impulsive topic of optimization of Quantum behaved Particle Swarm Optimization (QPSO). The Chaos Quantum behaved Particle Swarm Optimization (CQPSO) algorithm consists of chaos optimization strategy and QPSO algorithm. The learning speed of QPSO is fast due to method in which first of all the algorithm applied QPSO algorithm to apply evolution operation till QPSO algorithm was in early state. To implement the Short Term Load Forecasting (STLF), the CQPSO was applied to optimal the weight values of BP neural network. The results have been achieved high forecasting accuracy, and are an perfect optimal algorithm.

Liu *et al.* (2013) introduced the single hidden layer feed forward quantum neural network based on the Grover learning. The model was based on the concepts and principles of quantum theory. The quantum hidden neurons and the connected quantum weights were defined by joining the quantum mechanism with the feed forward neural network. These were used as a fundamental processing unit in a model. The Grover searching algorithms used the optimal parameter setting iteratively and thus made very efficient neural network learning possible. The quantum neuron and weights along with the Grover searching algorithm showed the result in an efficient neural network characteristic of reduced network.

Li *et al.* (2013) introduced the hybrid quantum inspired neural networks with sequence inputs. The performance of classical neural network was increased with the use of Quantum Inspired Neural networks (QINN) model. The QINN model was based on the controlled Hadamard gates. The inputs in the model were the discrete sequences which were described by the matrix where number of input nodes represent the number of rows and sequence length represents the number of columns. The model has three layers, the hidden layer contained the quantum neurons and the output layer contained the classical neuron. The CHQNN effectively achieved the sample characteristics by way of breadth and depth.

Takahashi *et al.* (2014) introduced the multi layer quantum neural network controller trained by real coded genetic algorithm. A multi – layer quantum neural network (QNN) was considered in which qubit neurons were used as a information processing unit. A real coded genetic algorithm was used to improve the learning process rather than of back propagation algorithm for supervised training of the multi layer QNN. For evaluating the capabilities of the direct quantum neural network controller, the computational experiments were conducted for controlling a discrete time nonlinear system and a non holonomic system (a two wheeled robot). Qubits were used to store the states circuits during quantum computation.

Li *et al*. (2014) introduced the application of a Hybrid Quantized Elman Neural Network (HQENN) in short term load forecasting. The least no of quantized inputs, hourly historical loads, and hourly predicted target temperature and time index were used with the HQENN. The objective was to view the features of HQENN for learning of the complex dynamics of hourly power load time series. It had the capability of predicting the future loads with high precision. The HQENN model was made of the qubit neurons and the classic neurons. The genetic algorithm was used to improve the forecasting accuracy as well as for obtaining optimal or suboptimal structure of the HQENN model. The HQENN model had the high accuracy in forecasting, as showed in the result. The adaptive learning rate was used to improve the convergence speed. The HQENN approach has achieved the higher accuracy of the short term power load forecasting as compare to the Elman Neural Network (ENN) based model and Multilayered Feed Forward Neural Network (MFNN) model based on the Back Propagation (BP) learning.

# CHAPTER 3

# BACKGROUND KNOWLEDGE OF ENGLISH AND HINDI LANGUAGE STRUCTURE

# Chapter 3

# Background knowledge of English and Hindi Language Structure

## 3.1. Introduction

Before going through the technical aspects of our research work, one needs to understand English and Hindi languages at structural level for translating Hindi to English and English to Hindi. It is known that both the languages are grammatically rich languages. This chapter explains the structure of English and Hindi Languages at grammatical level. It also gives the structural differences between the mentioned languages and challenges in translation due to their structural differences at grammatical level.

The study of any language can be broadly divided into:

- Morphology
- Syntax
- Lexicon

**Morphology:** Morphology is a Greek word, which consists of Morph plus -ology, Morph represents “different forms" while -ology represents the "the study of something". The word Morphology actually belongs to Biology, but now it has been used for linguistic as a linguistic Morphologic. Morphology is a sub discipline of linguistic, which studies the different form of words (Aronoff *et al.* 2011, Hutchins. 2003).

Morphological analysis gives us knowledge about how the uses of words morphemes affect the parts of speech of that word and classifies its parts of speech. The morphemes are mainly of five types: affixes, prefix, suffix, infix and circumfix. Although, infixes and circumfixes are not very common in English and Hindi Languages. The parts of speech are nouns, pronouns, preposition, conjunction, verbs, adverbs, adjectives, particles, interjections and connectives.

**Syntax:** The word Syntax consists of ancient Greek words: Syn and Taxis, Syn means "together" and Taxis "an ordering”. In linguistics Syntax explains how the sentence must be constructed. It defines the rules and principles that govern the sentence structure for formation of grammatical sentences from words in a language (Siddiqui and Tiwary. 2008).

**Lexicon:** The word Lexicon is derived from ancient Greek word lexikos means "of or for words". It is the collection of words with their meanings. Meanings may be one or more, the selection of meaning is always based on subject and domain specific language knowledge.

## 3.2. English Language Structure

English is a morphologically and lexically rich language. This language generally follows subject – verb – object (SVO) sentence structure. This structure indicates that first the subject comes then verb after that object comes last. Present section gives the overview of Morphology, Syntax and Lexicon of English.

### 3.2.1. English Language Morphology

At morphological level English has different Parts of Speech classes like nouns, pronouns, preposition, conjunction, verbs, adverbs, adjectives, particles, interjections and connectives (Singh 2003, Wren 1989).

#### 3.2.1.1. Noun

A Noun defines the name of an object, human or a place (Singh 2003, Wren 1989).

***a)*** ***Mahatma Gandhi*** was an exceptional ***leader***.

***b)*** ***Agra*** city is situated on the bank of ***Yamuna***.

***c)*** The ***bulb*** shines ***bright***.

##### 3.2.1.1.1. Proper Noun

A proper noun is used to refer a unique entity, such as Calgary, Sun, Tom, or Google (Singh 2003, Wren 1989).

**a)** ***Renu*** is my sister.

**b)** ***BMW*** is a good car.

**c)** ***India*** is a big country.

**d)** ***Mumbai*** is a rich city.

**e)** ***Delhi*** *is a big city.*

**f)** ***Newyork*** *is a rich city.*

#### 3.2.1.1.2. Common Noun

A common noun denotes a person or thing of the same group or class or kind instead of an individual, such as people, places, things, or ideas(Singh 2003, Wren 1989).

**a)** ***Aeroplanes*** fly in the sky.

***b)*** He has made many ***chairs.***

***c)*** Goats eat ***grass.***

**d)** ***Vitamins*** are useful for health.

#### 3.2.1.1.3. Collective Noun

Collective Noun is a noun which is used to refer group of people or things(Singh 2003, Wren 1989).

**a)** Twelve students in the ***class*** are intelligent.

**b)** Indian cricket ***team*** played well.

#### 3.2.1.1.4. Material Noun

Material Noun describes the name of a material or a substance by which other materials may be formed. Such as *Copper, Gold* (Singh 2003, Wren 1989).

**a)** ***Copper*** is the good conductor of electricity.

***b)*** My bracelet is made of ***Gold.***

***c)*** Ice is made from ***water.***

#### 3.2.1.1.5. Abstract Noun

An abstract noun is a type of noun that refers to something intangible (Singh 2003, Wren 1989).

**a)** Your ***move*** is good.

**b)** ***Honesty*** is the necessity of life.

#### 3.2.1.2. Pronouns

A word that is used instead of a noun (Singh 2003, Wren 1989).

**a)** Rohan is not present, because ***he*** is not feeling well.

**b)** The books are placed where, she left ***them***.

##### 3.2.1.2.1. Personal Pronouns

Personal pronouns are pronouns that indicate particular grammatical person. *I, my, mine, me* denotes the first person. *You, your, yours* denote the second person. *He*, *she*, *it, its, his, him, her, hers, they, theirs, them* denote the third person (Singh 2003, Wren 1989).

**a)** ***He*** is going.

**b)** ***She*** is reading.

**c)** ***You*** are singing.

**d)** ***We*** are sleeping.

##### 3.2.1.2.2. Interrogative Pronouns

Interrogative Pronouns are nouns, which are used for asking questions (Singh 2003, Wren 1989).

**a)** ***What*** is your name?

**b)** ***Where*** do you live?

**c)** ***Which*** is your car?

**d)** ***Who*** wrote this letter?

**e)** ***Whose*** is this house?

##### 3.2.1.2.3. Relative Pronouns

Relative pronouns are used to introduce a relative clause (Singh 2003, Wren 1989).

**a)** The car ***which*** you drive is mine.

**b)** The girl ***who*** sang the song is my cousin.

#### 3.2.1.2.4. Demonstrative Pronouns

Demonstrative Pronouns point to specific objects (Singh 2003, Wren 1989)

**a)** ***This*** is my house.

**b)** ***These*** are birds.

**c)** ***That*** is a car.

**d)** ***Those*** are boys.

#### 3.2.1.2.5. Distributive Pronouns

Distributive Pronouns are pronouns which are used to refer persons or things one at a time (Singh 2003, Wren 1989).

**a)** Read ***either*** of these articles.

**b)** ***Each*** of you may go now.

**c)** ***Nobody*** can do it.

**d)** ***Any*** of these five boys may drive the car.

### 3.2.1.3. Adjectives

An adjective qualifies a noun (Singh 2003, Wren 1989); as,

**a)** Ram is not a ***good person***.

**b)** There are ***ten*** chairs in ***this*** room.

### 3.2.1.4. Verbs

A verb is used to describe something about some person, place or thing (Singh 2003, Wren 1989).

A verb tells us (i) What a person or thing does as (ii) What is being done to person or thing as the child was beaten. (iii.) What a person or thing is as

**a)** The boy ***wrote*** a letter.

**b)** The chain is broken.

**c)** The man is blind, The glass is broken.

#### 3.2.1.5. Adverbs

An adverb is used to qualify or modify the verb, an adjective, or adverb (Singh 2003, Wren 1989), as:

**a)** This car is ***very*** fast.

#### 3.2.1.6. Interjections

An interjection expresses sudden feelings (Singh 2003, Wren 1989); as,

**a)** ***Hurrah!*** I have won the race.

**b)** ***Alas!*** He is dead.

### 3.2.2. Syntax of English Language

Syntax explains how the sentence must be constructed. It defines the rules and principles that govern the sentence structure for formation of grammatical sentences from words in a language. English language has three Tenses *i.e.* Present, Past and Future and all the categories and subcategories are explained with the help of tree given below.

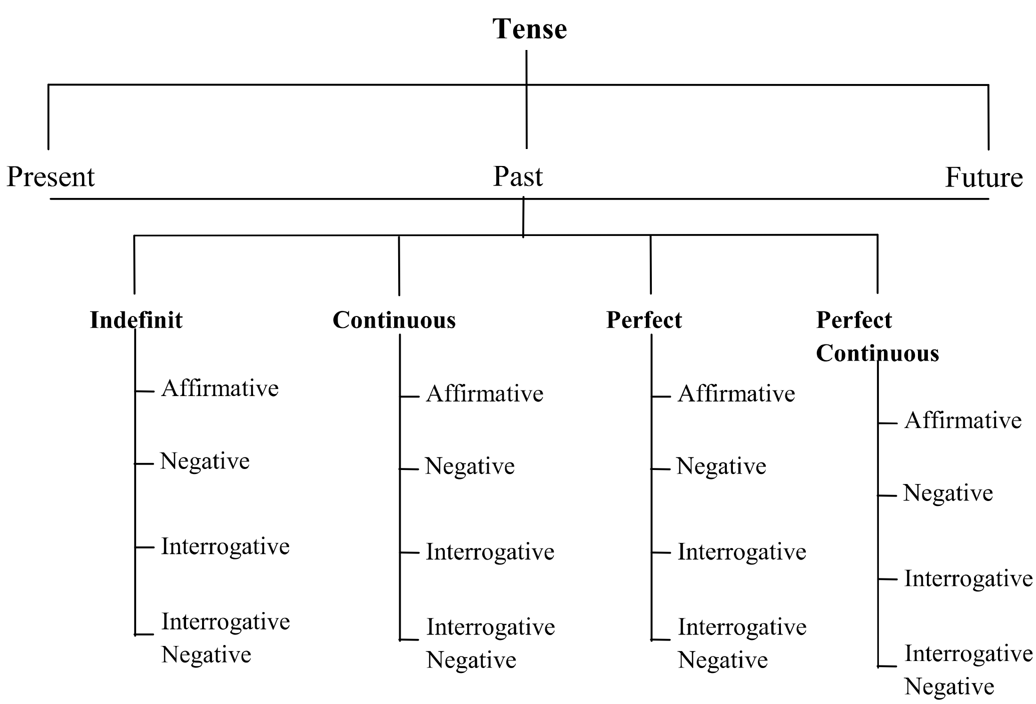


**Figure 3.1: English Tenses with its all the subgroup and forms**

## English Grammer Rules

**3.2.2.1. Present Indefinite** (Wren 1989).

**a) Affirmative**

Singular Subject (Pre-noun) + Verb ($1^{st}$ Form) s, es + Predicate

Plural Subject (Pre-noun) + Verb ($1^{st}$ Form) + Predicate

Example: Ram goes to the market

**b) Negative**

Singular Subject (Pre-noun) + does + not + Verb ($1^{st}$ Form) + Predicate

Plural Subject (Pre-noun) + do + not + Verb ($1^{st}$ Form) + Predicate

Example: Ram does not go to the market

**c) Interrogative**

Does + Singular Subject (Pre-noun) + Verb (1st Form) + Predicate

Do + Plural Subject (Pre-noun) + Verb ($1^{st}$ Form) + Predicate

Example: Does Ram go to the market?

**d) Interrogative Negative**

Does + Singular Subject (Pre-noun) + not + Verb ($1^{st}$ Form) + Predicate

Do + Plural Subject (Pre-noun) + not + Verb ($1^{st}$ Form) + Predicate

Example: Does Ram not go to the market?

**3.2.2.2. Present Continuous**

**a) Affirmative**

Subject (Pre-noun) +is, are, am + Verb ($1^{st}$ Form) + ing + Predicate

Example: Ram is going to the market

**b) Negative**

Subject (Pre-noun) +is/ are/ am + not + Verb ($1^{st}$ Form) + ing + Predicate

Example: Ram is not going to the market

**c) Interrogative**

Is/ are/ am + Subject (Pre-noun) +Verb ($1^{st}$ Form) + ing + Predicate

Example: Is Ram going to the market?

**d) Interrogative Negative**

Is/ are/ am + Subject (Pre-noun) + not + Verb ($1^{st}$ Form) + ing + Predicate

Example: Is Ram not going to the market?

### 3.2.2.3. Present Perfect (Wren 1989).

**a) Affirmative**

Subject (Pre-noun) + has/ have + Verb ($3^{rd}$ form) + Predicate

Example: Ram has gone to the market

**b) Negative**

Subject (Pre-noun) + has/ have + not + Verb ($3^{rd}$ form) + Predicate

Example: Ram has not gone to the market

**c) Interrogative**

Has/ have + Subject (Pre-noun) + Verb ($3^{rd}$ form) + Predicate

Example: Has Ram gone to the market?

**d) Interrogative Negative**

Has/ have + Subject (Pre-noun) + not + Verb ($3^{rd}$ form) + Predicate

Example: Has Ram not gone to the market

### 3.2.2.4. Present Perfect Continuous (Wren 1989).

**a) Affirmative**

Subject (Pre-noun) + has / have + been + Verb ($1^{st}$ Form) + ing + Predicate + since / for + Time Period

Example: Ram has been going to the market for two days.

**b) Negative**

Subject (Pre-noun) + has / have + not + been + Verb ($1^{st}$ Form) + ing + Predicate + since / for + Time Period

Example: Ram has not been going to the market for two days.

**c) Interrogative**

has / have + Subject (Pre-noun) + been + Verb ($1^{st}$ Form) + ing + Predicate + since / for + Time Period

Example: Has Ram been going to the market for two days?

**d) Interrogative Negative**

has / have + Subject (Pre-noun) + not + been + Verb ($1^{st}$ Form) + ing + Predicate + since / for + Time Period

Example: Has Ram not been going to the market for two days?

**3.2.2.5. Past Indefinite (Wren 1989).**

**a) Affirmative**

Subject (Pre-noun) + Verb ($2^{nd}$ Form/ d, ed) + Predicate

Example: Ram went to the market

**b) Negative**

Subject (Pre-noun) + did not + Verb ($1^{st}$ Form) + Predicate

Example: Ram did not go the market

**c) Interrogative**

Did + Subject (Pre-noun) + Verb ($1^{st}$ Form) + Predicate

Example: Did Ram go to the market?

**d) Interrogative Negative**

Did + Subject (Pre-noun) + not + Verb ($1^{st}$ Form) + Predicate

Example: Did Ram not go to the market?

**3.2.2.6. Past Perfect (Wren 1989).**

**a) Affirmative**

Subject (Pre-noun) + had + Verb (3rd Form) + Predicate

Example: Ram had gone to the market

**b) Negative**

Subject (Pre-noun) + had not + Verb (3rd Form) + Predicate

Example: Ram had not gone to the market

**c) Interrogative**

had + Subject (Pre-noun) + Verb (3rd Form) + Predicate

Example: Had Ram gone to the market?

**d) Interrogative Negative**

had + Subject (Pre-noun) +not + Verb (3rd Form) + Predicate

Example: Had Ram not gone to the market?

**3.2.2.7. Past Perfect (Before) (Wren 1989).**

Past Perfect + Before + Past Indefinite

Example: He had gone to his house before you came.

**3.2.2.8. Past Perfect (After) (Wren 1989).**

Past Indefinite + After + Past Perfect

Example: She slept after she had written a letter.

**3.2.2.9. Past Without After (Wren 1989).**

Past Indefinite + Past Perfect

Example: (Jab Vah college pahucha tab master sahib aa gaye the)

When he reached the college, the teacher had come

**3.2.2.10. Past Continuous (Wren 1989)**

**a) Negative**

Subject (Pre-noun) + Was / were + not + Verb ($1^{st}$ Form) + ing + Predicate

Example: Ram was not going to the market

**b) Interrogative**

Was / were + Subject (Pre-noun) + Verb ($1^{st}$ Form) + ing + Predicate

Example: Was Ram going to the market?

**c) Interrogative Negative**

Was / were + Subject (Pre-noun) + not + Verb ($1^{st}$ Form) + ing + Predicate

Example: Was Ram not going to the market?

**3.2.2.11. Past Perfect Continuous (Wren 1989).**

**a) Affirmative**

Subject (Pre-noun) + had been + Verb ($1^{st}$ Form) + ing + Predicate + since / for + Time period

Example: Ram had been going to the market for two days.

**b) Negative**

Subject (Pre-noun) + had not been + Verb ($1^{st}$ Form) + ing + Predicate + since / for + Time period

Example: Ram had not been going to the market for two days.

**c) Interrogative**

Had + Subject (Pre-noun) + been + Verb ($1^{st}$ Form) + ing + Predicate + since / for + Time period

Example: Had Ram been going to the market for two days?

**d) Interrogative Negative**

Had + Subject (Pre-noun) + not + been + Verb ($1^{st}$ Form) + ing + Predicate + since / for + Time period

Example: Had Ram not been going to the market for two days?

**3.2.2.12. Future Indefinite (Wren 1989).**

**a) Affirmative**

Subject (Pre-noun) + will / shall + Verb ($1^{st}$ Form) + Predicate

Example: Ram will go to the market

**b) Negative**

Subject (Pre-noun) + will / shall + not + Verb ($1^{st}$ Form) + Predicate

Example: Ram will not go to the market

**c) Interrogative**

Will / shall + Subject (Pre-noun) + Verb ($1^{st}$ Form) + Predicate

Example: Will Ram go to the market?

**d) Interrogative Negative**

Will / shall + Subject (Pre-noun) + not + Verb ($1^{st}$ Form) + Predicate

Example: Will Ram not go to the market?

**3.2.2.13. Future Continuous (Wren 1989).**

**a) Affirmative**

Subject (Pre-noun) + will / shall + be + Verb ($1^{st}$ Form) + ing + Predicate

Example: Ram will be going to the market.

**b) Negative**

Subject (Pre-noun) + will / shall + not + be + Verb ($1^{st}$ Form) + ing + Predicate

Example: Ram will not be going to the market.

**c) Interrogative**

Will / shall + Subject (Pre-noun) + be + Verb ($1^{st}$ Form) + ing + Predicate

Example: Will Ram be going to the market?

**d) Interrogative Negative**

Will / shall + Subject (Pre-noun) + not + be + Verb (1st Form) + ing + Predicate

Example: Will Ram not be going to the market?

**3.2.2.14. Future Perfect (Wren 1989).**

**a) Affirmative**

Subject (Pre-noun) + will / shall + have + Verb (3rd Form) + Predicate + by + time Period

Example: Ram will have gone to the market by 7 o' clock

**b) Negative**

Subject (Pre-noun) + will / shall + not have + Verb (3rd Form) + Predicate + by + time Period

Example: Ram will not have gone to the market by 7 o' clock

**c) Interrogative**

Will/ shall + Subject (Pre-noun) + have + Verb (3rd Form) + Predicate + by + time Period

Example: Will Ram have gone to the market by 7 o' clock?

**d) Interrogative Negative**

Will / shall + Subject (Pre-noun) + not have + Verb (3rd Form) + Predicate + by + time Period Example: Will Ram not have gone to the market by 7 o' clock?

**3.2.2.15. Future Perfect (Before) (Wren 1989).**

Future Perfect + Before + Present Indefinite

Example: (Tum mere aane se poorva apna kaam samapt kar chukoge)

You will have finished your work before I came.

**3.2.2.16. Future Perfect Continuous (Wren 1989).**

**a) Affirmative**

Subject (Pre-noun) + will have been /shall have been + Verb (1st Form) + ing + Predicate + since / for + Period Example: Ram will have been going to the market for two days.

**b) Negative**

Subject (Pre-noun) + will /shall + not have been + Verb ($1^{st}$ Form) + ing + Predicate + since / for + Period

Example: Ram will not have been going to the market for two days.

**c) Interrogative**

Will /shall + Subject (Pre-noun) + have been + Verb ($1^{st}$ Form) + ing + Predicate + since / for + Period

Example: Will Ram have been going to the market for two days?

**d) Interrogative Negative**

Will /shall + Subject (Pre-noun) + not + have been + Verb ($1^{st}$ Form) + ing + Predicate + since / for + time Period

Example: Will Ram not have been going to the market for two days?

## 3.3. Hindi Language Structure

Hindi is a morphologically and lexically rich language. It is a relatively free word order language. Therefore many permutations of the same sentence convey similar meaning. Hindi generally follows subject– object– verb (SOV) in a Hindi sentence first the subject comes, then object and verb comes in the last. This section gives overview of Morphology, Syntax and Lexicon of English.

### 3.3.1. Hindi Language Morphology

At morphological level Hindi has different Parts of Speech classes like nouns, pronouns, adjectives, verbs, adverbs, particles, connectives, and interjections *etc*.(Bender 1961, Kachru 1980, Singh 2003, Koul, 2008)

#### 3.3.1.1. Noun

In Hindi Nouns may reflect gender, number and case. The sub categories based on their inflection are of three types. First are masculine nouns ending with /a/, second are masculine nouns other than first sub category, third are feminine nouns.

### 3.3.1.1.1. Gender

Hindi contains two types of genders masculine and feminine. Based on their suffixation, phonological changes and suppletion, we may easily identify the type of gender *i.e.* masculine or feminine, the examples are given below.

**a) Mostly the masculine nouns which end with आ /a:/, the feminine forms of those nouns end with ई /i:/.**

**Table 3.1 Examples of Gender - masculine nouns ending with आ /a:/**

| **Masculine** | | | **Feminine** | | |
|---|---|---|---|---|---|
| **Hindi** | **Roman** | **English** | **Hindi** | **Roman** | **English** |
| मामा | Mama | Uncle | मामी | Mami | Aunt |
| शेर | Sher | Lion | शेरनी | Sherni | Lioness |
| गधा | Gadha | He Donkey | गधी | Gadhi | She Donkey |
| राजा | Raja | King | रानी | Rani | Queen |
| भतीजा | Bhatija | Nephew | भतीजी | Bhatiji | Nicce |
| पुत्र | Putra | Son | पुत्री | Putri | Daughter |
| मुर्गा | Murga | Cock | मुर्गी | Murgi | Hen |

**b) Mostly the animate masculine nouns end with ई /-i:/ they end with their feminine in -अन /-an/.**

**Table 3.2 Examples of Gender - masculine nouns end with ई /-i:/**

| **Masculine** | | | **Feminine** | | |
|---|---|---|---|---|---|
| **Hindi** | **Roman** | **English** | **Hindi** | **Roman** | **English** |
| सुनार | Sunar | Goldsmith(He) | सुनारन | Sunaran | Goldsmith(she) |
| दर्जी | Darji | Tailor(He) | दर्जन | Darjan | Tailor(she) |
| लुहार | Luhar | Blacksmith(He) | लुहारन | Luharan | Blacksmith(she) |
| कुम्हार | Kumhar | Potter(He) | कुम्हारन | Kumharan | Potter(she) |

| | | | | | |
|---|---|---|---|---|---|
| जमादार | Jmadaar | Sweeper(He) | जमादारन | Jmadaran | Sweeper(she) |

**c) The nouns which end with - आ /-a:/ their feminine form end with - इया /-iya:/.**

**Table 3.3 Examples of Gender- nouns which end with - आ /-a:/ their feminine form end with - इया /-iya:/.**

| Masculine | | | Feminine | | |
|---|---|---|---|---|---|
| **Hindi** | **Roman** | **English** | **Hindi** | **Roman** | **English** |
| लोटा | Lota | A round pot | लुटिया | Lutia | A small round pot |

**d) Mostly the inanimate nouns which end with -आ */-a:/* are masculine and the inanimate nouns which end with ई */-i:/* are feminine.**

**Table 3.4 Examples of Gender - nouns which end with -आ */-a:/* are masculine and the inanimate nouns which end with ई */-i:/* are feminine.**

| Masculine | | | Feminine | | |
|---|---|---|---|---|---|
| **Hindi** | **Roman** | **English** | **Hindi** | **Roman** | **English** |
| कंघा | Kangha | Comb(He) | कंघी | Kanghi | Comb(she) |
| प्याला | Pyala | Cup(He) | प्याली | Pyali | Cup(she) |
| पतीला | Patila | Pan(He) | पतीली | Patili | Pan(she) |
| डंडा | Danda | Stick(He) | डंडी | Dandi | Stick(she) |

**e) The feminine nouns can be changed to masculine nouns, by adding the suffix नी /-ni:/ to masculine nouns.**

**Table 3.5 Examples of Gender- suffix नी /-ni:/ to masculine nouns.**

| Masculine | | | Feminine | | |
|---|---|---|---|---|---|
| **Hindi** | **Roman** | **English** | **Hindi** | **Roman** | **English** |
| कलाकार | Kalakar | Actor | कलाकारनी | Kalakarni | Actress |
| खरगोश | Kharghosh | Heir | खरगोशनी | Kharghoshni | Doe |

| | | | | | |
|---|---|---|---|---|---|
| हिरन | Hiran | Deer(He) | हिरनी | Hirani | Deer(She) |
| चोर | Chor | Theif(He) | चोरनी | Chorni | Theif(she) |

### 3.3.1.1.2. Number

The number can be devided into two parts: - Singular number and Plural number.

**a) The masculine nouns, pronouns and adjectives ending with -आ /-a:/, mostly convert as -ए /-e/ making plural form.**

**Table 3.6 Examples of Number- ending with -आ /-a:/, change into -ए /-e/ making plural form.**

| Singular | | | Plural | | |
|---|---|---|---|---|---|
| **Hindi** | **Roman** | **English** | **Hindi** | **Roman** | **English** |
| केला | Kela | Banana | केले | Kele | Bananas |
| कुत्ता | Kutta | Dog | कुत्ते | Kutte | Dogs |
| नीला | Neela | Blue | नीले | Neele | Blue |

**b) The following -आ /-a:/ ending masculine nouns do not change in their plural form.**

**Table 3.7 Examples of Number--आ /-a:/ ending masculine nouns**

| Singular | | Plural |
|---|---|---|
| माता | Mata | Mother/Mothers |
| अभिनेता | Abhineta | Actor/Actors |

**c) The consonant- ending singular forms, by adding the suffix -एं (e) can be changed to plural forms .**

**Table 3.8 Examples of Number- the suffix -एं (e)**

| Singular | | | Plural | | |
|---|---|---|---|---|---|
| **Hindi** | **Roman** | **English** | **Hindi** | **Roman** | **English** |
| पुस्तक | Pustak | Book | पुस्तके | Pustake | Books |
| दीवार | Deewar | Wall | दीवारे | Deeware | Walls |
| बोतल | Botal | Bottle | बोतले | Botale | Bottles |

| | | | | | |
|---|---|---|---|---|---|
| **चादर** | Chadar | Sheet | **चादरे** | Chadare | Sheets |

### 3.3.1.2. The Postposition

In Hindi postpositions are some special key words which are used after the noun and pronoun and verb to pointing out them. In English instead of postposiotion, the prepositions are used before the noun (Koul, 2008).

Some of the main postpositions are: ने (ne), को (ko), से (se) ,में ह्यम्ẽ) ,पर (par), का (ka)

#### 3.3.1.2.1. The Postposition ने (ne)

In Hindi postposition ने (ne) generally comes before transitive verbs.

**a)** मैने पुस्तक पढी।

Maine pustak padhi.

I read a book.

**b)** उसने मैच खेला।

Usne match khela.

He played a match.

#### 3.3.1.2.2. The Postposition को (ko)

In Hindi sentences the postposition को ko is placed after nouns.

**a)** पुस्तक को पढो।

Pustak (ko) padho.

Read the book.

**b)** घर को जाओ।

Ghar (ko) jao.

Go home.

**c)** किताब को साफ रखो।

Kitab (ko) saaf rakho.

Keep the book clean.

#### 3.3.1.2.3. The Postposition से (se)

To show association, से se is used.

**a)** मै अपने मित्र के साथ  रहता हू।

Me apne mitra ke sath rahta hu.

I live with my friend.

**b)** राम ठंड से काप रहा है।

Ram is shivering with cold.

### 3.3.1.2.4. The Postposition में (me)

To show presence,  में mẽ is used.

**a)** मुझमे विश्वास रखो।

Mujhme Vishwas rakho.

Have confidence in me.

**b)** मुझे उस पर गर्व है।

Mujhe uspar garv hai.

I have pride in him.

**c)** वह कक्षा मे फेल नही होगा।

Vah kaksha mein fail nahi hoga.

He will not fail in the class.

### 3.3.1.2.5. The Postposition पर (par)

To show place, point, event, postposition पर par is used.

**a)** मै तीसरी मंजिल पर रहता हूँ।

Mai Tisri manjil par rahta hun.

I live on the third floor.

**b)** मेज पर किताब रखी है।

Mage par kitab rakhi hai.

The book is on the table.

**c)** राम स्कूल सही समय पर पहुचा।

Ram school sahi samay par pahuncha.

Ram reached school at right time.

### 3.3.1.2.6. The Postposition का (ka)

To show mutual relation with noun or pronoun, the postposition का ka is used.

**a)** मेरे मित्र का बेटा कनाडा मे रहता है।

Mere mitra ka beta Canada mein rahta hai.

My friend's son lives in Canada.

**b)** मार्बल के फर्श स्थायी होते है।

Marble ke farsh esthai hotey hi.

The marble floors are stable.

**c)** राम अयोध्या के निवासी थे।

Ram Ayodhaya ke nivasi the.

Ram was the resident of Ayodhaya.

**d)** यह मन्दिर सफेद पत्थर का बना है।

Yah mandir safed pathar ka bana hai.

This temple is made of white stone.

**e)** यह मेज लकडी की बनी है।

Yah mej lakri ki bani hai.

This table is made of wood.

### 3.3.1.2.7. Compound Postpositions

In Hindi Compound postpositions consist of के ke, की ki:, and से se to join other words(Koul, 2008).

**a)** ठंड से राम को बुखार हो गया।

Thand (se) Ram ko bukhar ho gya.

Ram caught fever because of the cold.

**b)** मेरे मित्र ने अपनी यात्रा के बारे मे बताया।

Mere mitra ne apni yatra ke bare mein bataya.

My friend told me about his visit.

**c)** इन सब बातो के बावजूद मै तुम्हे बहुत पंसद करता हू।

In sab baton ke babjud me tumhe bahut pasand karta hun.

Despite of all these things I like you very much.

**d)** वह नदी की तरफ गया।

Vah nadi ki taraf gaya.

He went towards the river.

**e)** उसने अपनी लापरवाही से अपना थैला खो दिया।

Usne apni laparwahi se apna thela kho diya.

She lost her bag through her carelessness.

**f)** कल तुम्हारा भाई मेरे साथ पार्टी मे बैठा था।

Kal tumhara bhai mere sath party mein baitha tha.

Yesterday your brother sat beside me in the party.

**g)** मन्दिर मे कोई नही रहता है।

Mandir mein koi nahi rahata hai.

No one lives inside the temple.

**h)** राम अपने मित्र के पास सोया।

Ram apne mitra ke pass soya.

Ram slept near his friend.

**i)** दो मित्रो के बीच झगडा हो गया।

Do mitron ke beech jhagra ho gaya.

Quarrel between two friends took place.

### 3.3.1.3. Noun Derivation

By using prefixes and suffixes, a large number of nouns are derived from nouns, adjectives and verbs.

#### 3.3.1.3.1. Nouns from Nouns

**Table 3.9 Examples of Nouns from Nouns**

| बे be | | | | | |
|---|---|---|---|---|---|
| **Hindi** | **Roman** | **English** | **Hindi** | **Roman** | **English** |

| | | | | | |
|---|---|---|---|---|---|
| जान | Jaan | Life | बेजान | Bejaan | Lifeless |
| कसूर | Kasoor | Fault | बेकसूर | Bekasoor | Faultless |
| रहम | Raham | Mercy | बेरहम | Beraham | Mercyless |
| छादर | Chadar | Sheet | चादरे | Chadare | Sheets |
| धाना | Thana | Policestation | थानेदार | Thanedar | Policeincharge |
| चौकी | Chowki | Policestation | चौकीदार | Chowkidar | Guard |
| मंत्री | Mantri | Minister | मंत्रालय | Mantralya | Ministeroffice |
| देव | Dev | God | देवालय | Devalaya | Temple |
| विद्या | Vidya | Knowledge | विद्यालय | Vidyalay | School |
| णाम | Naam | Name | बदनाम | Badnaam | |
| इन्तजाम | Intjaam | Manage | बदइन्तजाम | Badintjaam | Mismanage |
| रवि | Ravi | Sun | रविवार | Ravibar | Sunday |
| शोम | Som | Moon | सोमवार | Somvar | Monday |
| झायज | Jayaj | Legal | नाजायज | Najayaj | Illegal |
| ळायक | Layak | Able | नालयक | Nalayak | Unable |
| खुश | Khush | Happy | नाखुश | Nakhush | Unhappy |
| यश | Yash | Fame | अपयश | Apyash | Defame |
| शगुन | Shagun | Omen | अपशगुन | Apshagun | Ominous |
| भावना | Bhavana | Will | दुंभावना | Durbhavana | Illwill |
| झन | Jan | Devil | दुर्जन | Durjan | Evil |
| माता | Mata | Good Mother | कुमाता | Kumata | Bad mother |
| पुत्र | Putra | Good Son | कुपुत्र | Kuputra | Bad Son |
| मार्ग | Marg | way | कुमार्ग | Kumarg | Wrong way |
| आधार | Aadhar | Base | निराधार | Niradhar | Baseless |
| झन | Jan | People | निर्जन | Nirjan | Lonely |
| टोप | Tope | Gun | तोपची | Topchi | Gunner |
| निशाना | Nishana | Target | निशानची | Nishanchi | Gunner |
| पान | Paan | Betel | पानदान | Paandan | Betelbox |
| फूल | Phool | Flower | फूलदान | Phooldan | Flowerpot |
| डवा | Dava | Medicine | दवाखाना | Davakhana | Medical Store |

#### 3.3.1.3.2. Nouns from Adjectives

**Table 3.10 Examples of Nouns from Adjective**

| बे be | | | | | |
|---|---|---|---|---|---|
| **Hindi** | **Roman** | **English** | **Hindi** | **Roman** | **English** |
| मजबूत | Majboot | Strong | मजबूती | Majbooti | Toughness |
| खट्टा | Khatta | Sour | खटाई | Khatai | Sourness |
| छोटा | Chota | Small | छोटी | Chotti | Small |
| भहादुर | Bahadur | Brave | बहादुरी | Bahaduri | Braveness |
| घुलाम | Ghulam | Slave | गुलामी | Ghulami | Slavery |
| ईमान | Imaan | Honest | ईमानदारी | Imaandari | Honesty |
| आजाद | Azad | Free | आजादी | Azadi | Freedom |
| प्रसन्न | Prasann | Glad | प्रसन्नता | Prasannta | Gladness |
| नीच | Neech | Mean | नीचता | Neechta | Meanness |
| व्ययस्त | Vyast | Busy | व्यस्तता | Vyastata | Busyness |
| अच्छा | Accha | Good | अच्छापन | Acchapan | Goodness |
| मठिए | Meetha | Sweet | मीठापन | Meethapan | Sweetness |
| खुला | Khula | Open | खुलापन | Khulapan | Openness |
| बुन | Buun | Weaver | बुनाई | Bunai | Weavering |
| उस | Has | Laugh | हसाई | Hasai | Laughing |
| इंसान | Insaan | Human | इंसानियत | Insaniyat | Humanity |

#### 3.3.1.3.3. Nouns from Verbs

**Table 3.11 Examples of Nouns from verbs**

| बे be | | | | | |
|---|---|---|---|---|---|
| **Hindi** | **Roman** | **English** | **Hindi** | **Roman** | **English** |
| जा | Ja | Go | जाना | Jana | Going |
| देख | Dekh | See | देखना | Dekhna | Seeing |
| खेल | Khel | Play | खेलना | Khelna | Playing |
| शोना | Sona | Sleep | शोना | Sona | Sleeping |
| लिखा | Likha | Write | लिखावट | Likhavat | Writing |

#### 3.3.1.4. Pronouns

The words which are used in place of noun are called pronoun, In Hindi pronouns are of seven types i.e. personal, demonstrative, relative, possessive, reflexive, interrogative, and indefinite.

##### 3.3.1.4.1. Personal Pronouns

Personal pronouns are pronouns that are associated with specific person. In Hindi some of the Personal pronouns are मैं (mai), हम (Hum), तू (Tu), तुम (Tum), आप (Aap), यह (Yah), ये (Yeh), वह (Wah), वे (We)

**Table 3.12 Examples of Personal Pronouns**

| Hindi | Roman | English |
|---|---|---|
| हमे | Hame | us |
| हमने | Hamne | |
| हमपर | Hampar | |
| हमसे | Hamse | |
| हमारा | Hamara | |
| आपका | Aapka | You |
| आपको | Aapko | |
| आपने | Aapne | |
| आपपर | Aappar | |
| आपसे | Aapse | |
| तुमने | Tumne | Your |
| तुमपर | Tumpar | |
| तुमसे | Tumse | |
| तुम्हारा | Tumhara | |
| इन्हे | Inhe | Them |
| इन्होने | Inhone | |
| उन्हे | Unhe | |
| उन्होने | Unhone | |
| इनपर | Inpar | |
| इनसे | Inse | |
| उनपर | Unpar | |

| | | |
|---|---|---|
| उनका | Unka | |
| उसका | Uska | His |
| उनसे | Unse | |

### 3.3.1.4.2. Demonstrative Pronouns

Demonstrative pronouns represent particular entities. In Hindi demonstrative pronouns also represent personal pronouns of third person. In Hindi Some of them are the demonstrative pronouns of Hindi are

यह yeh, ये ye, वह vah, वे ve, इस is, इन in, उस us, उन un

### 3.3.1.4.3. Relative Pronouns

Relative pronouns are pronouns which are used to correlate two clauses sharing common word. Some of the Relative pronouns of Hindi are जिस jis, जिसने jisne, जिन jin.

### 3.3.1.4.4. Reflexive Pronouns

Compound personal pronouns and Reflexive pronouns are the same. When the action done by the subject reflects upon the subject. By adding –self to my, her and –selves to your, our *etc*. are named as compound personal pronouns and also Reflexive pronouns.

a) मै खुद को घ्रणा करता हूँ।

Mai khud ko ghrina karta hu.

I hate ***myself.***

b) वह उससे घ्रणा करती है।

Vah us se ghrina karti hai.

She hates ***himself.***

### 3.3.1.4.5. Interrogative Pronouns

Interrogative Pronouns are those which are used to ask questions. e.g

a) मै कौन हूँ।

Mai kaun hu?

***Who*** am I?

b) तुम्हारा क्या नाम है।

Tumhara kya naam hai?

***What*** is your name?

c) तुम कहा रहते हो।

Tum kaha rahte ho?

***Where*** do you live?

d) तुम्हारी कार कौन सी है।

Tumhari car kaun si hai?

***Which*** is your car?

### 3.3.1.4.6. Indefinite Pronouns

All these Pronouns in italics refer to person thing in a general, but do not refer to any person or thing in particular. So they are known as Indefinite Pronouns.

a) इसे कोई नही कर सकता है।

Ise koi nahi kar sakta hai.

Nobody can do this.

### 3.3.1.4.7. Compound Pronouns

To explain the many types of meanings, two or more than two pronouns may be combined or repeated just as -

a) हर कोई।

Har koi.

All, everybody.

b) बहुत कुछ।

Bahut kuch.

A great deal.

### 3.3.1.5. Adjectives

Adjectives are describing words which give some extra information about noun or pronoun. Hindi language has two types of adjectives (i) inflected (ii) uninflected.

### 3.3.1.5.1. Types of Adjectives

**Table 3.13 Examples of types of adjectives**

| **Hindi** | **Roman** | **English** |
|---|---|---|
| कठोर पेन्सिल | Kathor Pencil | Hard Pencil |
| लाल गुलाब | Lal Gulab | Red Rose |
| मोटा आदमी | Mota Aadmi | Thick Man |
| अच्छी किताब | Acchi kitab | Good book |
| तेज चाकू | Tej chaku | Sharp knife |
| कुछ कागज | Kuch kagaj | Few papers |
| बहुत धन | Bahut dhan | Much money |
| पर्याप्त भोजन | Paryapt bhojan | Enough food |
| यह शर्ट | Yah shirt | This shirt |
| ये शर्टे | Ye shirte | These shirts |
| वह शर्ट | Vah shirt | That Shirt |
| वे शर्टे | Ve Shirte | Those Shirts |
| किस का कलम | Kis ka kalam | Whose pen |
| क्या समय | Kya samay | What time |
| कौन सा घर | Kaun sa ghar | Which house |
| मेरे पिता | Mere pita | My father |
| उसका भतीजा | Uska bhatija | His nephew |
| कुछ पक्षी | Kuch pakshi | Some birds |

**a)** मैने पांच कलम बेचे।

Maine Panch kalam beche.

I five pens sold.

I sold five pens.

**b)** पांच मीटर लम्बी मछली।

Panch meter lambi machli.

Five meter long fish.

### 3.3.1.5.2. Degree of Adjectives

**Table 3.14 Examples of degree of Adjectives**

| Hindi | Roman | English |
|---|---|---|
| लता सबसे बुद्धिमान लडकी है। | Lata sabse buddiman larki hi | Lata is wisest of all the girls. |
| लता रेनू से लम्बी है। | Lata Renu se lambi hi. | Lata is taller than Renu. |
| लता एक अच्छी लडकी है। | Lata ek acchi larki hi. | Lata is a wise girl. |
| अपने मित्र की अपेक्षा छोटी। | Apne mitra ki apeksha choti | Smaller than her friend. |

### 3.3.1.5.3. Derivation of Adjectives

**Table 3.15 Example of Derivation of Adjectives**

| Noun | | | Adjective | | |
|---|---|---|---|---|---|
| **Hindi** | **Roman** | **English** | **Hindi** | **Roman** | **English** |
| उपयोग | Upyog | Use | उपयोगी | Upyogi | Useful |
| | | Hand | | | Handful |
| वफादार | Vafadar | Faith | वफादारी | Vafadari | Faithful |
| | | Play | | | Playful |
| दुख | Dukh | Sorrow | दुखी | Dukhi | Sad |
| लम्बा | Lamba | Long | लम्बी | Lambi | Lengthy |
| गरीब | Garib | Poor | गरीबी | Garibi | Poverty |

### 3.3.1.6. Verbs

In Hindi there are mainly seven types of verbs, *i.e.* intransitive, transitive, ditransitive, causative, dative, conjunct, and compound.

#### 3.3.1.6.1. Intransitive Verbs

Intransitive verbs like Aa *a:* ‘come,’ jaa *ja:* ‘go’ उठ *uth* ‘get up,’ and बैठ *baith* ’sit.’

**a)** राम सोता है।

Ram sota hai.

Ram sleep is.

Ram sleeps.

**b)** वह दिल्ली जायेगा।

Vah dilli jyega.

He Delhi go.

He will go Delhi.

#### 3.3.1.6.2. Transitive Verbs

Transitive verbs, such as पढ, *parh* 'read,' लिख *likh* 'write,' ला *la:* 'bring,' दे *de* 'give,' ले *le* 'take,' and कर *kar* 'do,'

**a)** उसने मुझे एक कहानी सुनाई।

Usne mujhe ek kahani sunai.

He me a story told.

He told me a story.

**b)** राम ने मुझे एक पत्र भेजा।

Ram ne mujhe ek patra bheja.

Ram me a letter send.

Ram send me a letter.

**c)** गाडी रुकी ।

Gadi ruki.

The train stopped.

**d)** शीशा टूट गया।

Sheesha tut gaya.

The glass broke.

#### 3.3.1.6.3. Ditransitive Verbs

Some verbs like देना *dena* 'to give,' सुना *suna* 'to tell,' बेचना *becna* 'to sell' are known as ditransitives.

**a)** राम ने मित्र को उपहार दिया।

Ram ne mitra ko uphar diya.

Ram gave a gift to the friend.

**b)** सीता ने अपने मित्र को पत्र भेजा।

Sita ne apne mitra ko patra bheja.

Sita sent a letter to her friend.

### 3.3.1.6.4. Causative Verbs

Causative verbs are derived, when causative suffixes are added to transitive verbs.

**a)** सीता ने सहेली को मिठाई **दी**।

Sita ne saheli ko mithai di.

Sita made her friend to give sweets.

**b)** डाक्टर ने मरीज को मॉ से दवा **पिलवायी**।

Doctor ne marij ko maa se dava pilvai.

Doctor caused the patient to drink medicine from the mother.

### 3.3.1.6.5. Dative Verbs

Most dative verbs may be found by placing the intransitive events होना *hona:* 'to be, and' आना *ana.* .

**a)** मुझे यह कलम पंसद है।

Mujhe yah kalam pasand hai.

I like this pen.

**a)** मुझे यह कलम पंसद आयी।

Mujhe yah kalam pasand aayi.

I liked this pen.

**a)** मैने यह कलम पंसद की।

Maine yah kalam pasand ki.

I liked this pen.

**b)** मुझे सारी कहानी याद आयी।

Mujhe sari khane yaad ayi.

I remembered the whole story.

**b)** मैने सारी कहानी याद की।

Maine sari khane yaad ki.

I remembered the whole story.

#### 3.3.1.6.6. Conjunct Verbs

A conjunct verb is that which has a verb, a noun or adjective. Both transitive and intransitive verbs may be there as conjunct verbs. करना *karna:* 'to do' and होना *hona:* 'to be.'.'

**a)** मैने मेरा पाठ याद कर लिया।

Maine mera paath yaad ker liya hai.

I learnt my lesson.

**b)** मेरा पाठ समाप्त हुआ।

Mera paath samapt hua.

My lesson is finished.

**c)** काम समाप्त करो।

Kaam samapt karo.

Finish this work.

**d)** काम समाप्त हुआ।

Kaam samapt hua.

The work was finished.

**e)** सीता को हंसी आयी।

Sita ko hansi aayi.

Sita laughed.

**f)** सीता को दया आयी।

Sita ko daya aayi.

Sita felt pity.

### 3.3.1.6.7. Compound Verbs

In Hindi Compound verbs consist of Verb 1 + Verb 2 (+ inflections). In which Verb 1 is the main verb and verb 2 has all the inflexions and is known as explicator/operator. The original meaning of the explicator is lost.

**a)** पार्टी समय पर शुरु हुई।

Party samay par shuru hui.

Party started on time.

**b)** मैने पुस्तक पढ ली।

Maine pustak padh li.

I read the book.

**c)** मैने तस्वीर खींच ली।

Maine tasveer kheench li.

I drew the picture.

**d)** वह सारे दिन चलता रहा।

Vah sare din chalta raha.

He went on going whole day.

**e)** वह ध्यान से पढता रहा।

Vah dhyan se padta raha.

He kept on reading attentively.

### 3.3.1.7. Adverbs

An Adverb qualifies a verb, adverb and adjective. e.g

**a)** मोहन तेजी से चलता है।

Mohan teji se chalta hain.

Mohan *walks* fastly.

**b)** गन्ना बहूत मीठा है।

Ganna bahut meetha hain.

The sugarcane is *very* sweet.

**c)** राधा बिल्कुल साफ बोलती है।

Radha bilkul saaf bolti hai.

Radha speaks *quite* clearly.

In sentence 1 *fastly* qualifies the verb walks.

In sentence 2 *very* qualifies the adjective sweet.

In sentence 3 *quite* qualifies the adverb clealy.

### 3.3.1.8. Connectives

The words which join two elements are known as connectives.

#### 3.3.1.8.1. Mono-morphemic

Mono-morphemic consists of one morpheme.

**a)** सीता स्कूल गयी और मीना देहली गयी।

Sita school gayi aur meena delhi gayi.

Sita went to school and Meena went to Delhi.

#### 3.3.1.8.2. Poly-morphemic

Two or more morphemes are converted in Poly-morphemic.

**a)** राम मैच जीत गया क्योकि उसने कठोर परिश्रम किया।

Ram match jeet gaya kyunki usne kathor parishram kiya.

Ram won the match because he worked hard.

### 3.3.1.9. Phrasal

Phrasals consist of two elements interrupted by intervening words, such as अगर *agar* … तो *to* 'if … then.'

**a)** अगर तुम मुझे पुस्तक दो तो मै पढ़ूंगा।

Agar tum mujhe pustak do to mai padoonga.

If you give me the book so then I will read.

### 3.3.1.10. Interjections

Interjections represents cry, greet, surprise, anger, pleasure *etc*. They may be preferred to nouns.

**Table 3.16 Examples of Interjections**

| | | |
|---|---|---|
| हे राम ! | He ram ! | Oh God! |
| ओ माँ ! | Oh maa! | Oh mother! |

**a) Joy**

Example: हा ! हा ! माँ आ गई।

Ha! Ha! Maa aa gai.

Ha! Ha! Mother has come.

**b) Sorrow, grief**

Example: हाय मै बर्बाद हो गया।

Hay mai barbaad ho gaya.

Alas! I am finished.

Ah! I am finished.

**c) Surprise**

Example: क्या! तुम पास हो गये।

Kya ! tum pass ho gaye.

What! Have you passed.

**d) Disgust**

Example: छी/ छी/ थू / तुमने किताब चोरी की।

Chi/chi/ thu/ tumne kitab chori ki.

Shame! You theft the book.

### 3.3.2. Syntax of Hindi Language

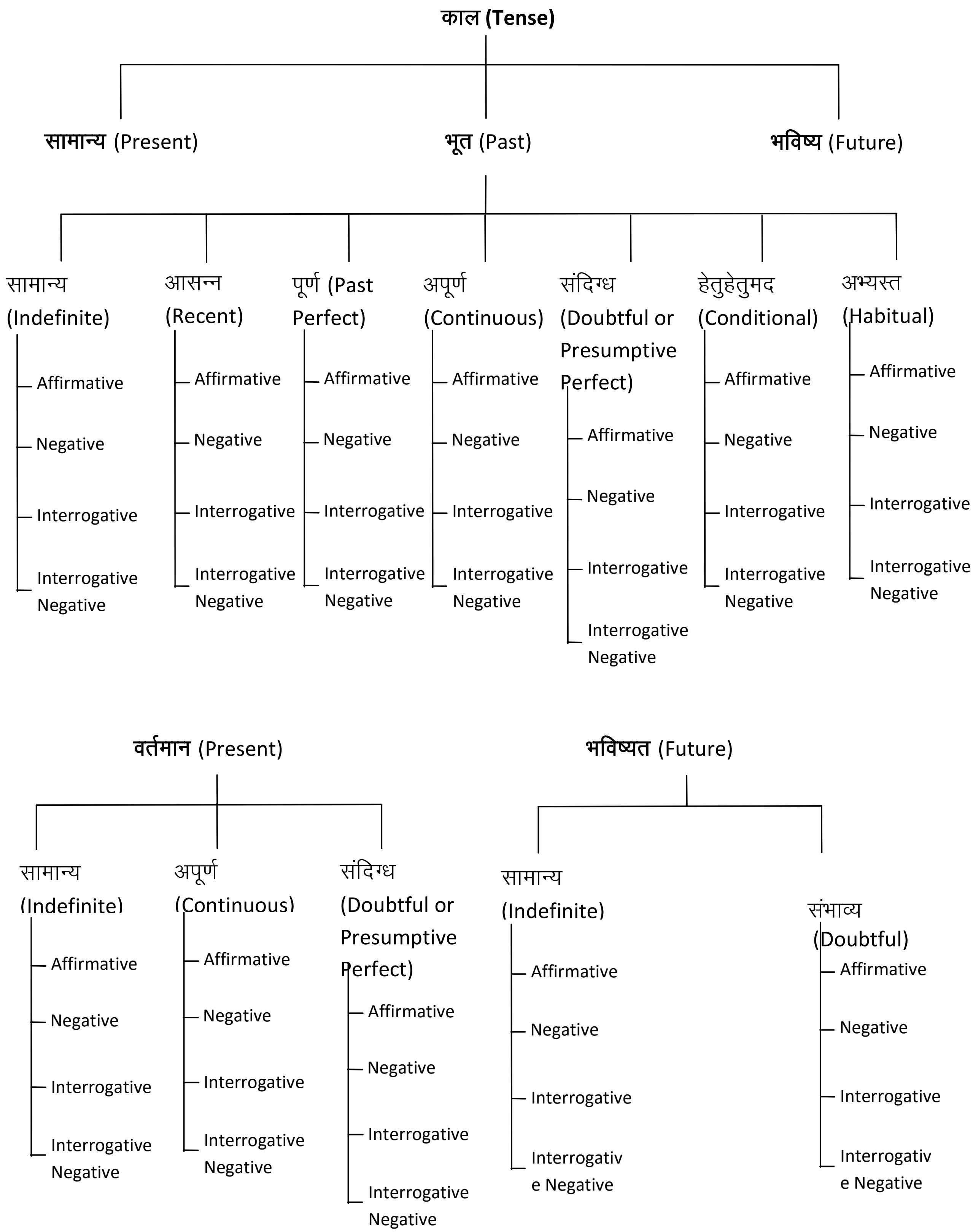


**Figure 3.2 Hindi Tenses with its all the subgroup and forms**

### 3.3.2.1. भूतकाल (Past Tense)

In Hindi, there are seven types past tenses:

**a) सामान्य भूत (Past Indefinite Tense)**

Past Indefinite Tense shows the work done or action taken in the past.

Example: मैने देहली गया था। (I went to Delhi.)

**b) आसन्न भूत (Recent Past Tense)**

The action which was completed over a time or in the past is known as Recent Past Tense.

Example: मैने चावल खाये है। (I have eaten the rice.)

**c) पूर्ण भूत (Past Perfect Tense)**

This Tense shows an action occurred before another action.
Example: मै चावल खाये थे। (I had eaten rice.)

**d) अपूर्ण भूत (Past Continuous Tense)**

The Past Continuous Tense shows an ongoing action in the past.

Example: मै चावल खा रहा था। (I had been eating rice.)

**e) संदिग्ध भूत (Doubtful Past Tense or Presumptive Perfect)**

The tense shows the doubtful action which took place in the past.

Example: मैने चावल खाये होगे। (I might have eaten rice.)

**f) हेतुहेतुमद भूत (Conditional Past Tense)**

This tense shows condition in the action, done in the past (Singh, 2003).

Example: अगर तुम भूखे होते तो तुमने चावल खाये होते। (If you had hunger you would have eaten rice.)

**g) अभ्यस्त भूत (Habitual Past Tense)**

This tense shows regular happening of a particular action in the past.

Example: मै चावल खाया करता था। or मै चावल खाता था। (I used to eat rice.)

### 3.3.2.2. वर्तमानकाल (Present Tense)

There are two types of tenses in Hindi :

**a)** सामान्य वर्तमान (Present Indefinite Tense)

The action taken in present, is the Present Indefinite Tense.

Example: मै गाडी चलाता हूँ। (I drive car.)

**b)** अपूर्ण वर्तमान (Present Continuous Tense)

The ongoing action in the present represents the Present Continuous Tense.

Example: मै गाडी चला रहा हूँ। (I am driving the car.)

**c)** संदिग्ध वर्तमान (Doubtful Present Tense)

The doubtful present tense shows doubt in action which may occur or not.

Example: मै गाडी चला रहा होंगा। (I would be driving the car.)

### 3.3.2.3. भविष्यत काल (Future Tense)

There are two types of Future Tense :

**a)** सामान्य भविष्यत (Future Indefinite Tense)

The action taken in the future, describes the future indefinite tense.

Example: मै गाडी चलाऊंगा। (I will drive the car.)

**b)** संभाव्य भविष्यत (Doubtful Future Tense)

This tense shows the doubtful occurrence of an action or may occur or not in future (Singh, 2003).

Example: शायद मै गाडी चलाऊंगा। (Maybe I will drive the car.)

### 3.4. Translation difficulties due to structural difference between English and Hindi

As discussed above in detail, the English and Hindi languages are morphologically and lexically rich languages and differ with each other at structural level, due to their structural differences, translation of Hindi to English and English to Hindi is difficult. English generally follows subject–verb–object (SVO) sentence structure, in English the subject comes first, the verb second and the object third.

On other hand Hindi is relatively free word order language. Therefore many permutations of the same sentence convey similar meaning. Hindi generally follows subject– object– verb (SOV) sentence structure where the subject comes first, the object second and the verb third.

In English, tenses are of three types i.e. Present, Past and Future and every tense categorize in four subcategories: Indefinite, Continuous, Perfect and Perfect Continuous. Same as English, Hindi also has three types of Tenses *i.e.* Present, Past and Future. But each are having different subcategories *i.e.* in Present tense there are only three subcategories, in Past tense there are seven subcategories and in Future tense there are seven subcategories. However, due to this insufficiency in combining them, it is difficult to convey the sense of sentence after translation.

In English there are ten types of pronouns. On other hand Hindi has only seven. In Hindi there is no distinguish between male and female in third person pronoun, for example हमे (hume) in English “We”, second example is उसे (Ose) in English “He or She”

In Hindi there is no word equivalent to “do”. Therefore to convey a question, intonation is used. In Hindi, in conditional sentences the future tense is used in the independent clause. In English, requests are generally used in the form of question. Hindi does not have definite articles, while English language has. The number “one” is used in place of the indefinite article (Vikram, 2013).

# CHAPTER 4

# QUANTUM THEORY AND QUANTUM NEURAL NETWORK

# Chapter 4

# Quantum Theory and Quantum Neural Network

## 4.1. Introduction

The motivation behind the study of QNN is its similarity with the living brain. As humans perform well in every situation whether normal or difficult, *i.e.* in unseen and unusual circumstances also. Human can successfully work in these unrealistic situations where classes of information are not bounded tightly or overlapped on each other. Human thinking employs rules-of-thumb, experience, perception and other heuristics. Similarly QNN is able to independently distinguish the occurrence of undetermined sample data and the learning for determination is done automatically due to its adaptive feature. QNN detects and determines the sample data without any assumption about the class and its levels. If the resulting dataset comes in two different classes, the QNN assigns this fragmentary overlapped data to all overlapped classes. In realistic situation, if the classification is not determined then as per the training set, the class is assigned by the QNN (Purushothaman and Karayiannis, 1994, 1997, 1998). The traditional Artificial Neural Network (ANN) advocates the study of intracellular structures but it can only cover the realistic space in given data classes, having the nature of certainty (Purushothaman and Karayiannis, 1994, 1997, 1998; Kretzschmar *et al*, 2000).

Quantum neural network (QNN) is effective to classify indeterminate data, because QNN has inherently fuzzy properties which encode the sample information into discrete levels of certainty or uncertainty. The transfer function of quantum neural network expressed as linear superposition of sigmoid function (Zhou *et al,* 1999). In this way, a hidden layer neural cell expresses more states. Each sigmoid function has different quantum interval. Thus, QNN decreases indeterminacy and increases veracity of pattern recognition (Behrman *et al,* 2000; Mitsunaga *et al,* 2006).

The major difference between conventional neural network and quantum neural network is that in QNN instead of the ordinary sigmoid functions, a multilevel activation function is used. Each multilevel function consists of the sum of sigmoid functions shifted by the quantum intervals. (Daqi *et al*., 2007). The quantum computing may be used as a general framework for producing quantum analogs of well-known classical artificial neural networks.

In QNN, a neuron is regarded as a basic information unit of a quantum computer (qubit) and a synaptic connection corresponds to a qubit interaction (Mitsunaga *et al.*, 2006). According to Li *et al.* (2002), three layer architecture of QNN consists of inputs, one layer of multilevel hidden units, and output units.

Zhu *et al.* (2007) developed a method for handwritten recognition using QNN. The system which was based on the QNN shows the best performance. The experiments on the digital recognition system were performed on the MNIST database. The 96.5% performance was achieved in the recognition rate and 98.8% was achieved in the recognition reliability. The issues involved in the system were to describe the character feature.

### 4.2. Quantum theory

Quantum theory is the collection of theoretical principles that describes the nature and behaviour of matter at sub-atomic level. This theory also explains the day to day life phenomenon very well.

The fundamental principles of Quantum theory are:

- **The superposition principle:** this principle explains about the possible quantum states of a particle in a given system.
- **The measurement principle:** this principle provides information about the accessible state of any particle.
- **Unitary evolution: -** The change in state with time is predicted with this principle of quantum theory.

#### 4.2.1. Quantum Superposition Principle

One particle is in two quantum states at the same time. The quantum superposition principle explains the occurrence of the observable effects at sub- atomic level. The principle of superposition explains the fact that the state of any object or particle is not known, as actually it is simultaneously possible in all states, and *i.e.* any particle can occupy all of its quantum states simultaneously.

### 4.3 Quantum Neural Network

The major difference between the QNN and NN is the superposition state in hidden layer of QNN. In QNN, instead of the ordinary activation function a multilevel activation function is used. The multilevel function consists of the addition of sigmoid function which is moved by the quantum intervals. More states and levels can be shown by such type of hidden layer neurons. The detailed information gathered and updated at micro intervals of quantum can manage and pass the uncertain data to various patterns and the accuracy of pattern recognition can be improved.

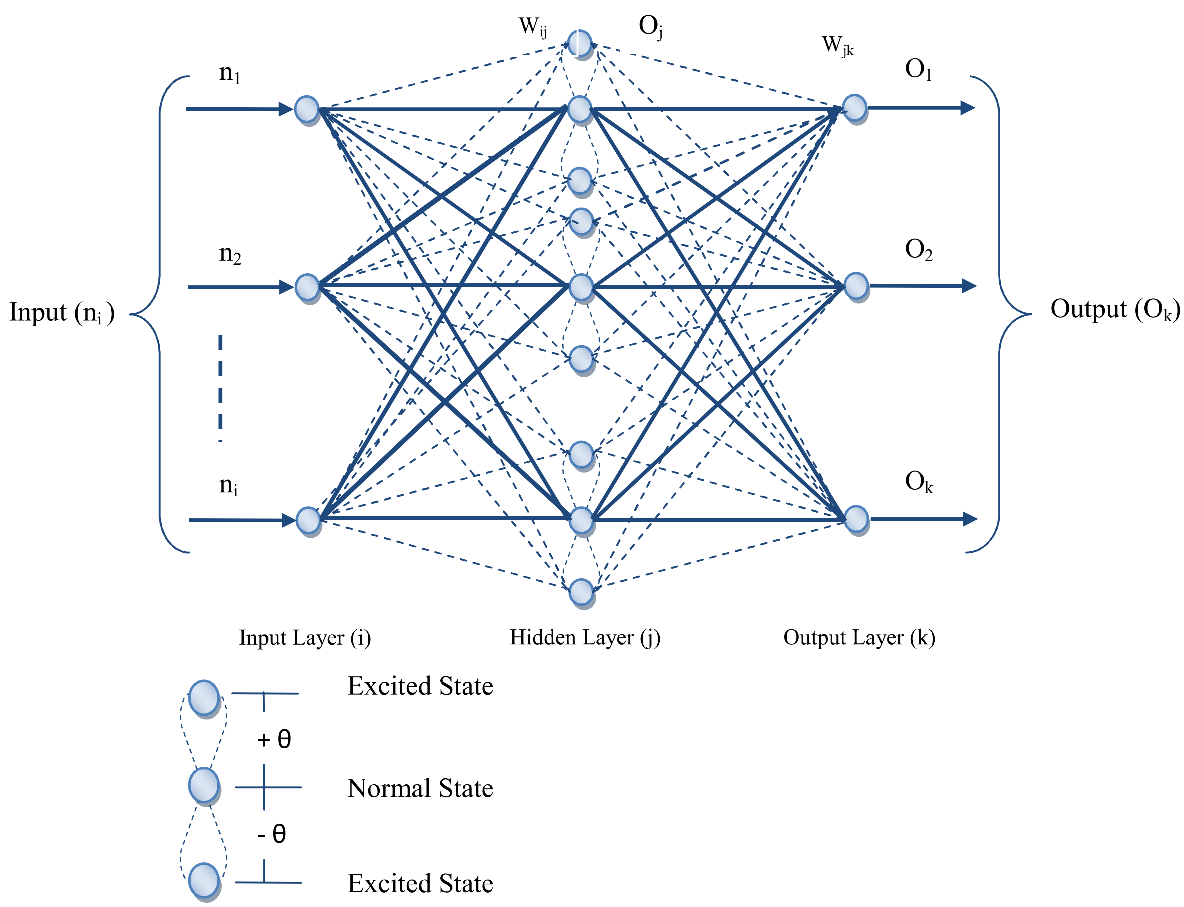


**Figure 4.1 Architecture of Quantum Neural Network**

As shown in the Figure 4.1, the three layer architecture of QNN is composed of input layer, hidden node with several levels, and output node has been proposed for this research work. Here, each hidden layer neuron is having two excitation states which virtually represent two more nodes but any one of the node can be used for learning. The random selection of node is done using QNN algorithm based on the inclination towards the decreasing error rate.

The quantum sigmoid function with various graded levels has been used as the activation function of hidden layer and is expressed as follows:

$$sgm(x) = \frac{1}{n_s}\sum_{r=1}^{n_s}\left(1/(1+\exp(-x \pm \theta^r))\right) \qquad \text{... (4.1)}$$

The simple sigmoid function has been used as the activation function of output layer and is expressed as:

$$sgm(x) = \left(1/(1+\exp(-x))\right) \qquad \text{... (4.2)}$$

Here the total number of quantum levels are three *i.e.* One normal with two excited states. Here, using the concept of quantum physics, every hidden layer node represents three substates in itself with the difference of quantum interval $\theta^r$. Here r is quantum level. Here the Levenberg-Marquardt method has been used as training method.

**4.4. Proposed Quantum Neural Network algorithm**

The proposed QNN model is shown in Figure 4.1, and the step by step flow of this algorithm is shown below in the flow chart as Figure 4.2. The QNN consist of input, output and hidden units. Only one hidden layer has been used. Every node of hidden layer represents three sub states in itself with the difference of quantum interval $\theta^r$ with quantum level r.

In this algorithm $n_s$ denotes the number of excitation levels, $\mu$ is combination coefficient which is a small random value, $\delta_k$ is error rate of output layer and $\delta_j$ error rate of hidden layer. Where $n_i$ denotes the input to QNN. $O_j$ and $O_k$ denotes the output of hidden and output layer, respectively. The weights between input and hidden layers are denoted by

$W_{ij}$ and the weights between hidden and output layers are denoted by $W_{kj}$. The initial weights are small random numbers and t denotes Target value. I denotes the Identity matrix, J denotes the Jacobian matrix and $J^T$ transpose matrix of J. (Purushothaman and Karayiannis, 1994, 1997, 1998, Chakraverty *et. al.,* 2010, Singh *et al.* 2009).

Here in this algorithm the Levenberg-Marquardt method has been used as training method.

Given R training pairs $\{n_1, t_1; n_2, t_2; n_R, t_R\}$

$n_i$ *(*$j_n$ *X* $i_n$*) is input and* $t_i$ *(*$k_n$ *X* $i_n$*)* is target values for given input.

**Step 1:-** Weights $W_{jk}$ and $W_{ij}$ are initialized at small random values and $\mu > 0$ is combination coefficient, the value of $\mu$ should always be positive and selected randomly.

**Step 2:-** Calculate the linear activation $net_j$, by calculating the sum of signals on each of the neuron at hidden layer.

$$net_j = \sum_i W_{ij} n_i \qquad \dots (4.3)$$

**Step 3:-** Calculate the output of hidden layer by applying quantum sigmoid function on each neuron of hidden layer.

$$O_j = \frac{1}{n_s} \sum_{r=1}^{n_s} \left(1/(1+\exp(-net_j \pm \theta^r))\right) \qquad \dots (4.4)$$

Here the initial value of $\theta^r$ is initialized by zero.

**Step 4:-** Calculate the linear activation $net_k$ , by calculating the sum of signals on each of the neuron at output layer.

$$net_k = \sum_j W_{jk} O_j \qquad \dots (4.5)$$

**Step 5:-** Calculate the output of output layer by applying the logarithmic sigmoid function as activation function for each neuron of output layer.

$$O_k = \left(1/(1+\exp(-\, net_k))\right) \qquad \ldots (4.6)$$

**Step 6:-** Calculate the amount of error signal term $\delta_k$ of the output layer.

$$\delta_k = \frac{1}{n}\sum_{k=1}^{n}(t_k - O_k)^2 \qquad \{k=1,\ 2,\ 3....,\ k_n\} \qquad \ldots (4.7)$$

**Step 7:-** Update the weights of output layers from the following expression:

$$W_{kj}^{new} = W_{kj}^{old} - \left(J_{kj}^{T} J_{kj} + \mu I\right)^{-1} J\delta_k \qquad \{j=1,\ 2,\ 3....,\ j_n\}\ and\ \{k=1,\ 2,\ 3....,\ k_n\} \ \ldots (4.8)$$

**Step 8:-** Calculate the amount of error signal term $\delta_j$ of the hidden layer by using the new weight $W_{kj}^{new}$:

$$\delta_j = O_j(1-O_j)\sum_k W_{kj}^{new}\delta_k \qquad \{j=1,\ 2,\ 3....,\ j_n\}\ and\ \{k=1,\ 2,\ 3....,\ k_n\} \qquad \ldots (4.9)$$

**Step 9:-** Update the weights of hidden layers from the following expression:

$$W_{ji}^{new} = W_{ji}^{old} - \left(J_{ji}^{T} J_{ji} + \mu I\right)^{-1} J\delta_j \qquad \{i=1,\ 2,\ 3....,\ i_n\}\ and\ \{j=1,\ 2,\ 3....,\ j_n\} \ \ldots (4.10)$$

Now apply Step 1 to Step 9 on every input and iterate several times upto $\delta_k$ reaches the lowest possible error and then apply Step10.

**Step 10:-** Update the Quantum Interval:

After *i* iterations when minimum possible error is obtained, then increase in

Quantum interval by very small quantum interval $\Delta\theta$, for proposed system the value of $\Delta\theta = 0.25$ has been used.

$$\theta^r = \theta^r + \Delta\,\theta^r \qquad \ldots (4.11)$$

Repeat Step 10 until error $\delta_k$ is decreased to an acceptable accuracy.

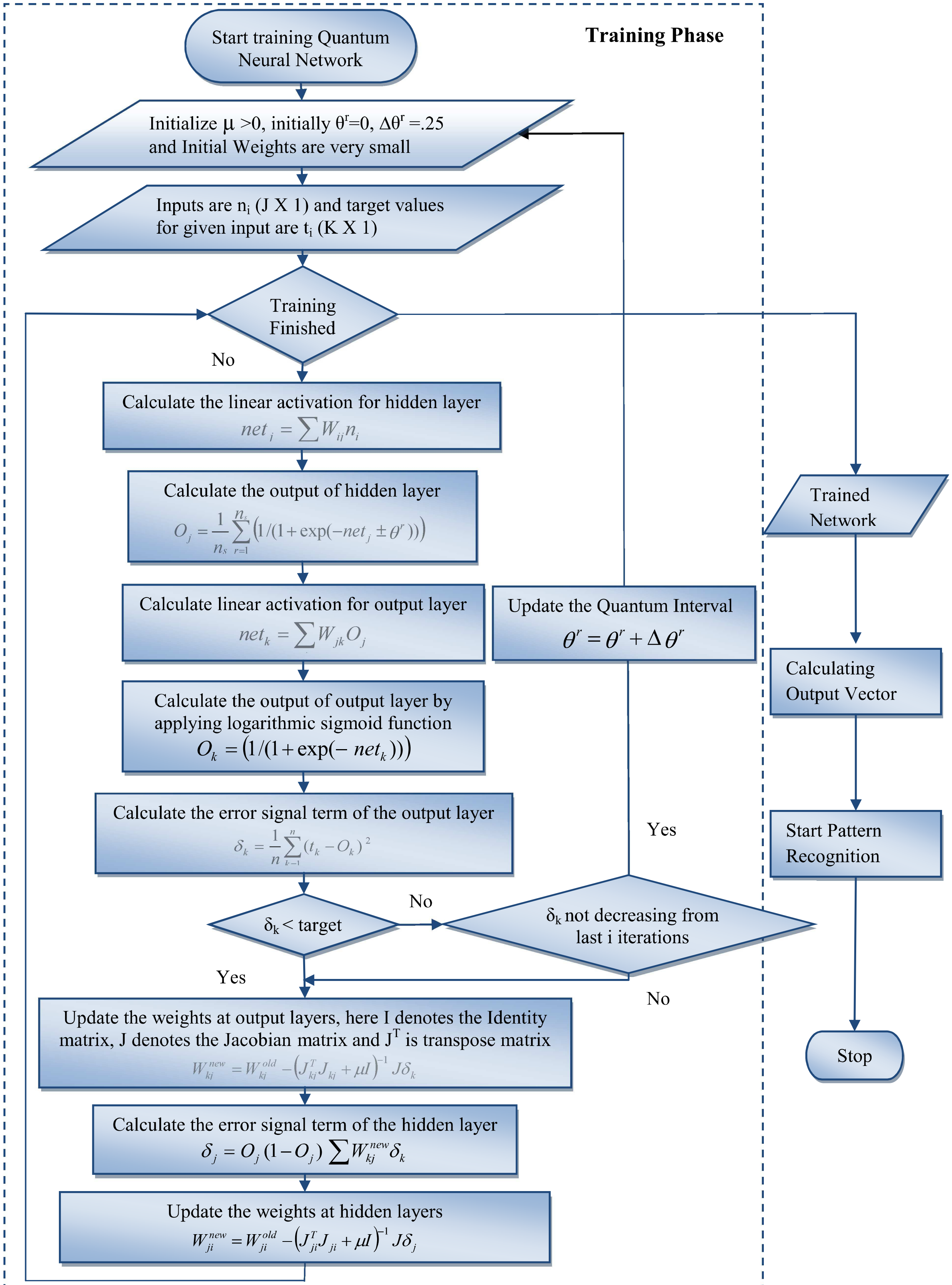


**Figure 4.2 Flow chart of proposed Quantum Neural Network**

### 4.5. Advantages of Quantum Neural Network

The main advantages of Quantum Neural Network are:

a) QNN needs less computation time to train the network, as compared to traditional Artificial Neural Network (ANN). *i.e.* QNN is faster than ANN.

b) QNN easily and accurately handles uncertain overlapped data, due to its ability to independently detect the occurrence of uncertainty in the sample data. It adaptively learns to quantify the existing uncertainty using its multilevel function.

c) QNN is more accurate in pattern recognition, as compared to traditional ANN.

# CHAPTER 5

# PROPOSED QUANTUM NEURAL NETWORK BASED MODEL FOR MACHINE TRANSLATION

# Chapter 5

# Proposed Quantum Neural Network based model for Machine translation.

## 5.1. Introduction

Machine Translation is one of the major field of NLP in which research interest is growing since the time of computers were invented. Many machine translation systems are available with their pros and cons for variety of languages. Researchers have also presented different approaches to understand and generate the languages with semantics and syntactic. But still many languages have translation difficulties due to ambiguity in their words and the grammatical complexity. The machine translator should have the key characteristic to increase the performance of machine translation up to the level of human performance in translation. Most of the machine translators are working on the alignment of words in chunk (sentence).

This research work presents the quantum neural based machine translation for Hindi to English and vice versa. The Quantum Neural Network (QNN) based approach increases the accuracy during the knowledge adoptability. In this research work the main focus is to show the significant increase in the accuracy of machine translation during this research with the pair of Hindi and English sentences. The machine translation is done using the new approach based on QNN which learns the patterns of language using the pair of sentences of Hindi and English and vice versa.

## 5.2. Methodology used to implement proposed Machine Translation system

As discussed above about the existing approaches of machine translation, semantic translation, syntactic translation, all the concepts are used in machine translation, inspired from human translator. The intermediate steps used by human translator during translation of sentence from language **a** to language **b** are:

**Step 1:-** To identify and tag the parts of speech, using the word meaning knowledge and by corpus matching in which words are used in the sentence.

**Step 2:-** Sentence type analysis by human translator, that is, which grammatical tense it belongs to on the basis of parts of speech identification and the available tense rules/ grammar rules.

**Step 3:-** The human translator refers the appropriate grammatical tense rule for translation to language **b.** Then the rule is implemented for translation.

**Step 4:-** Semantically rearrange the words in sentence *i.e.* semantic translation, and then put the equivalent meaning of all the words *i.e.* syntactic translation. The semantic rearrangement of words is done by human translator by using the language knowledge of target language. Without knowing the language knowledge of target language and by only using the grammar rules/ tense rules and dictionary, it is very difficult to do accurate translation by human translator as well as for machine translator. Humans generally gain the language knowledge by reading text and by using his/her experience about the target language.

Hence, similar to human translator, machine translator also needs to learn the language knowledge for accurate translation. By implementing the translation rules with QNN, the machine translator can learn the language knowledge as patterns, and this process is implemented in the proposed system. As discussed about the features of quantum neural network in chapter 4, it has all the qualities to implement the translation rules.

## 5.3. Architecture of Proposed QNN based MT System Model

The proposed Machine Translation (MT) system consists of two approaches, one is Rule based MT system, and another is QNN Based MT system. The source language goes into the RBMT system and passes through the QNN based MT system to refine the MT done by RBMT Module, which basically recognize and classify the sentence category. 2600 sentences are used with English and their corresponding Devanagari-Hindi sentences. Each sentence may contain a Question Word, Noun, Helping Verb, Negative Word, Verb, Preposition, Article, Adjective, Post Noun and Adverb *etc*. The data used to train, is produced by an algorithm, which is based on simple deterministic grammar. The entire architecture of the proposed MT system model is given in Figure 5.1.

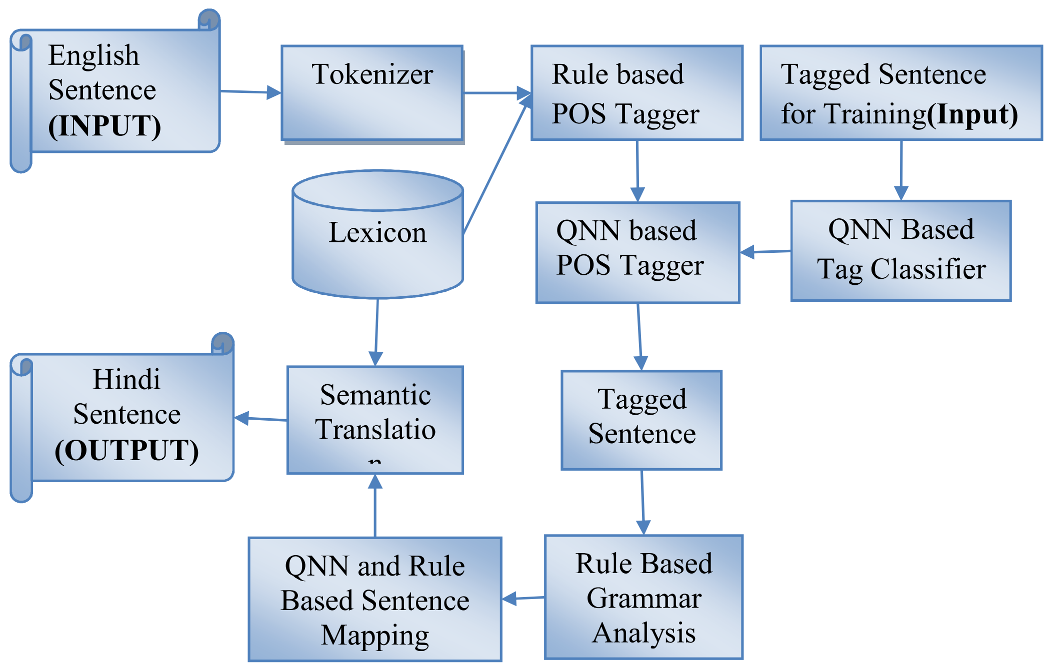


**Figure 5.1 Architecture of MT System Model**

### 5.3.1. Proposed QNN based Tagger

The proposed Parts of speech (POS) tagging system is inspired with the human translator. What are the steps Human generally follows for identifying the POS tagging, they first refer the dictionary/ Lexicon and then pick the parts of speech information directly from the dictionary/ Lexicon and then match with the sentence pattern on the basis of grammar rules, if it suits the pattern then it is ok, else human correct their decision for parts of speech on the basis of sentence pattern. Similarly the proposed system uses the same method. In this system, the raw sentence first passes through the Tokenizer, the Tokenizer splits the sentence into words and indexes it as token and then the resulting words with token, pass through the rule based POS Tagger. The rule based POS tagger tag the POS by simply using the Lexicon. The outcome of the Rule based POS Tagger is not perfect, for correction and accuracy it finally passes through the QNN based POS tagger, which refines the identified rule based POS using the pattern recognition of corpus. Here the QNN is used for pattern recognition of corpus to identify and correct the POS tagging. For learning purpose, some manually tagged sentences are input in the QNN based POS tagger, on the bases of input tagged sentences the QNN based POS tagger learns all the patterns of POS tagging. The whole process is shown in Flow Diagram of QNN based parts of speech tagger in Figure 5.2.

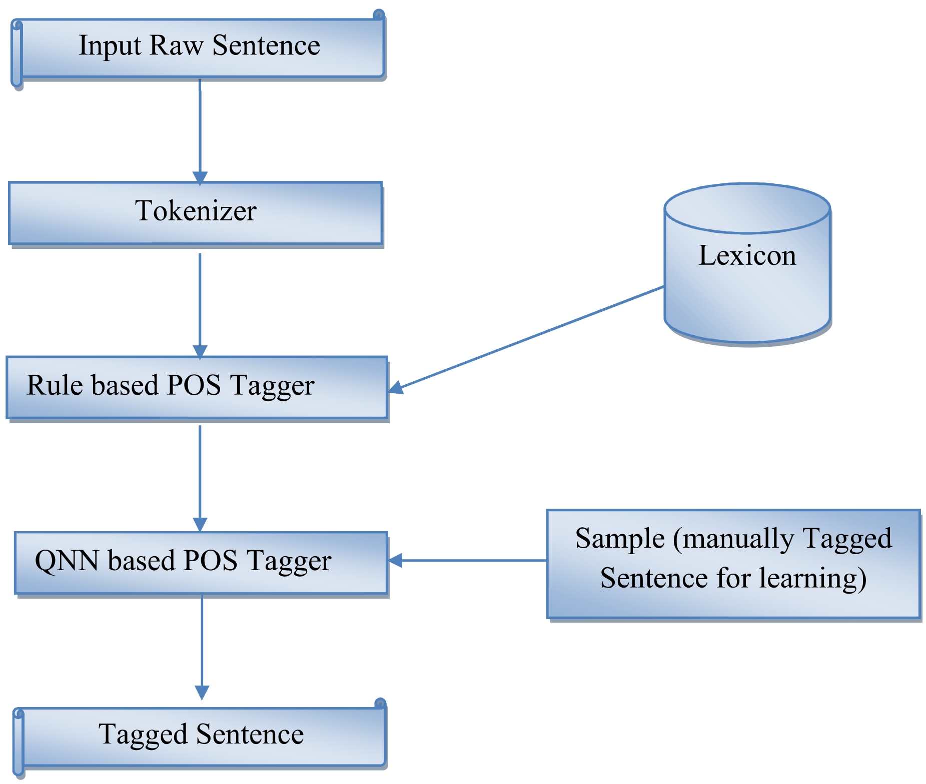


**Figure 5.2 Flow Diagram of QNN based Parts of Speech Tagger**

#### 5.3.1.1. Representation of Input and Output patterns

There are 2600 Hindi sentences of news items from various newspapers which are used for training purpose. The corpus used for the training and testing purposes contains 11500 words. The training set is generated from a simple deterministic grammar by a program. The POS tag of words in a sentence must be represented in numeric form. Table 5.1 shows the input POS tags which use 3 bits encoding scheme representation and their corresponding numeric code for the target word Parts of Speech tags.

#### 5.3.1.2. Tokenizer

Tokenizer split a sentence into meaningful elements, which are often referred as words. Literally a tokenizer breaks up sentences into pieces called tokens. A token is an instance of a sequence of characters or numbers for a sentence to group collectively as a useful semantic unit for processing. Here, in the proposed model the tokenizer splits the sentence into words and indexes it as token.

**For Example:**

Sentence: He is a good boy.

After passing the sentence to Tokenize, the outcome of tokenizer is:

[He] [is] [a] [good] [boy]

#### 5.3.1.3. Tagset and Coding Mechanism

Tagset is the set of parts of speech tags from which the tagger uses the POS of a relevant word. Some of the generally parts of speech are N (Noun), V (Verb), ADJ (Adjective), ADV (Adverb), PREP (Preposition), CONJ (Conjunction) which depend on the morphological structure of any language. Here, in the proposed Hindi and English parts of speech tagger the Tagset is listed below with its coding mechanism in Table 5.1.

**Table 5.1: TagSet with its numeric codes**

| **Parts of Speech (Sub Class)** | **Occurrence Number** | **Numeric Code based on Class(Parts of Speech ) - POS base id** | **Resulting code - POS id** |
|---|---|---|---|
| Pre Noun (PN) | Noun(0) | .10 | .100 |
| Noun-infinitive (Ni) | Noun(1) | | .101 |
| Pronoun (PRO) | Noun(2) | | .102 |
| Gerund (GER) | Noun(3) | | .103 |
| Relative Pronoun (RPRO) | Noun(4) | | .104 |
| Post Noun (POSTN) | Noun(5) | | .105 |
| Verb (V) | Verb(0) | .11 | .110 |
| Helping verb (HV) | Verb(1) | | .111 |
| Adverb (ADV) | Verb(2) | | .112 |
| Auxiliary verb (AUX) | Verb(3) | | .113 |

| | | | |
|---|---|---|---|
| Interrogative (Question Word) (INT) | Determiner(0) | .12 | .120 |
| Demonstrative words (DEM) | Determiner(1) | | .121 |
| Quantifier (QUAN) | Determiner(2) | | .122 |
| Article (A) | Determiner(3) | | .123 |
| Adjective (ADJ) | Adjective(0) | .13 | .130 |
| Adjective-particle (ADJP) | Adjective(1) | | .131 |
| Number (N) | Adjective(3) | | .132 |
| Preposition (PRE) | Preposition(0) | .14 | .140 |
| Postposition (POST) | Preposition(1) | | .141 |
| Punctuation (PUNC) | | .15 | .150 |
| Conjunction (CONJ) | | .16 | .160 |
| Interjection (INTER) | | .17 | .170 |
| Negative Word (NE) | | .18 | .180 |
| Determiner (D) | | .19 | .190 |
| Idiom (I) | | .20 | .200 |
| Phrases (P) | | .21 | .210 |
| Unknown Words (UW) | | .22 | .220 |

In the above parts of speech tagset, resulting codes are generated on the basis of their base class of POS and the occurred number. Here, the occurrence of the number starts with 0, means at very first time if noun occurs in sentence then the resulting code is .100 and if second time the noun occurs in sentence then the resulting code is .101 and so on. Numerically, the coding mechanism expressed as:

$$\text{Re}\,sulting\;\;code(POS\,id) = POSbase\,id + \frac{Occurrence\,Number}{1000} \qquad \text{... (5.1)}$$

For Example:

If we calculate the POS id of “Article” then, by using the table 5.1. The POSBase id of “Article” is .12 and occurrence number is 3.

. by using the Eq 5.1

Resulting POS id for Article = .12 + 3/10000

= .123

#### 5.3.1.4. Rule based POS Tagger

In the Rule based POS tagger POS tags are labeled by most likely POS by using the lexicon / dictionary, and defined rules. As in dictionary every word has word meaning along with the POS information, but it is possible that in a dictionary a single word contains multiple POS tagging information. The POS of a word always depends on the relative sentence in which the word is used. That is why the POS tagging is very ambiguous.

**For Example:**

**Sentence:** He is a good boy.

After passing the sentence to Tokenize, the outcome of tokenizer is:

[He] [is] [a] [good] [boy]    As already shown in Sub Section 5.3.1.2.

Now the outcome of Tokenizer passes through Rule base POS Tagger, the steps are as follows:

**Step 1:** Identify the parts of speech of each word by using the lexicon / dictionary.

| | | | | |
|---|---|---|---|---|
| [He] | [is] | [a] | [good] | [boy] |
| [ProNoun] | [Helping Verb] | [Preposition] | [Adjective] | [Noun] |

**Step 2: Now based on Table 5.1 and Eq 5.1 tagg each word based on their POS**

| | | | | |
|---|---|---|---|---|
| [He] | [is] | [a] | [good] | [boy] |
| [ProNoun] | [Helping Verb] | [Article] | [Adjective] | [Noun] |
| **[.102]** | **[.111]** | **[.123]** | **[.130]** | **[.105]** |

#### 5.3.1.5. Quantum Neuro tagger algorithm

For given sentence, perform the following steps:

- **Learning Phase:**

  **INPUT:** Manually tagged training corpus

  **OUTPUT:** The Patterns of POS Tagging rules learned.

- **Tagging Phase using RBMT:**

**INPUT:** Untagged Corpus

**Step 1:** Tokenizer splits the sentence into words and indexes it as token

**Step 2:** Rule based POS Tagger Labels most likely tag by Using / picking up the POS information directly from dictionary. And go to the Step 4

If (word is having more than one POS available in dictionary)

Then

Go to Step 3.

**Step 3:** Check the word in Article and Adjective list, if found then tag the word on the bases of POS, else call the procedure "Ambiguous word handler" for selecting the correct POS of that word,

**Step 4:** Based on identified POS of word and tagged the word by using Equation 5.1 and Table 5.1.

**Step 5:** Passes to the QNN based POS Tagger

**OUTPUT:** Most accurate POS Tagged Corpus

- **Ambiguous word handler:**

**INPUT:** Ambiguous word (having multiple meanings or POS in dictionary)

**Step 1:** Select the Noun word from the sentence and check the class of that.

**Step 2:** from multiple POS and meaning of ambiguous word, select the POS and meaning which belongs to the same class as Noun word.

**OUTPUT:** Most appropriate POS information with meaning of Ambiguous word.

**For Example:**

**Sentence:** He is a good boy.

**Step 1:** After passing the sentence to Tokenize, the outcome of tokenizer is:

[He] [is] [a] [good] [boy] As already shown in Sub Section 5.3.1.2.

Now the outcome of Tokenizer passes through Rule base POS Tagger, the steps are as follows:

**Step 2:** Identify the parts of speech of each word by using the lexicon / dictionary.

[He] [is] [a] [good] [boy]

| | | | | |
|---|---|---|---|---|
| [ProNoun] | [Helping Verb] | [Preposition] | [Adjective] | [Noun] |

**Step 3:** Now based on Table 5.1 and Eq 5.1 tagg each word based on their POS

| | | | | |
|---|---|---|---|---|
| [He] | [is] | [a] | [good] | [boy] |
| [ProNoun] | [Helping Verb] | [Article] | [Adjective] | [Noun] |

**Or**

| | | | | |
|---|---|---|---|---|
| [ProNoun] | [Helping Verb] | [Preposition] | [Adjective] | [Noun] |
| [.102] | [.111] | [.140] | [.130] | [.105] |

Here, the word "a" may have two different POS Tagg, the correct preposition in respect of whole sentence of word "a" should be [Article], but the output of Rule Base POS Tagger(RBMT) might be second one [Preposition] in this situation, the outcome of RBMT must be refined by Quantum Neuro tagger algorithm.

**Step 4:** Based on identified POS of word and tagged the word by using Equation 5.1 and Table 5.1.

| | | | | |
|---|---|---|---|---|
| [ProNoun] | [Helping Verb] | [Preposition] | [Adjective] | [Noun] |
| [.102] | [.111] | [.140] | [.130] | [.105] |

**Step 5:** Passes to the QNN based POS Tagger

| | | | | |
|---|---|---|---|---|
| [He] | [is] | [a] | [good] | [boy] |
| [ProNoun] | [Helping Verb] | [Article] | [Adjective] | [Noun] |

Here, in the outcome of Step 5 the word "a" is now correct POS i.e. preposition selected in respect of whole sentence word "a" is Article

#### 5.3.1.6. Implementation of Quantum Neural Based POS Tagger

As discussed in the above sections 5.1 and 5.2, the implementation concept of QNN based tagger is purely inspired from the human interpreter. Thus the steps are similar with the steps used by human interpreter, to implement the POS tagging rules with QNN. This system first picks the parts of speech of any word using the well defined rules and lexicon, the word has different POS in different sentences. The part of speech of any word in respect of any sentence depends on how the word acts in sentence. To overcome this ambiguous situation, this system uses a smart approach, which will be discussed in section 5.3.5. by rules based parts of speech tagger. Rules based parts of speech tagger uses well defined rules and dictionary/Lexicon. the set of parts of speech then passes through the QNN based POS tagging system which is here used as pattern recognizer, which learns and corrects the POS tag information on the basis of corpus/sentence patterns learned in past during training.

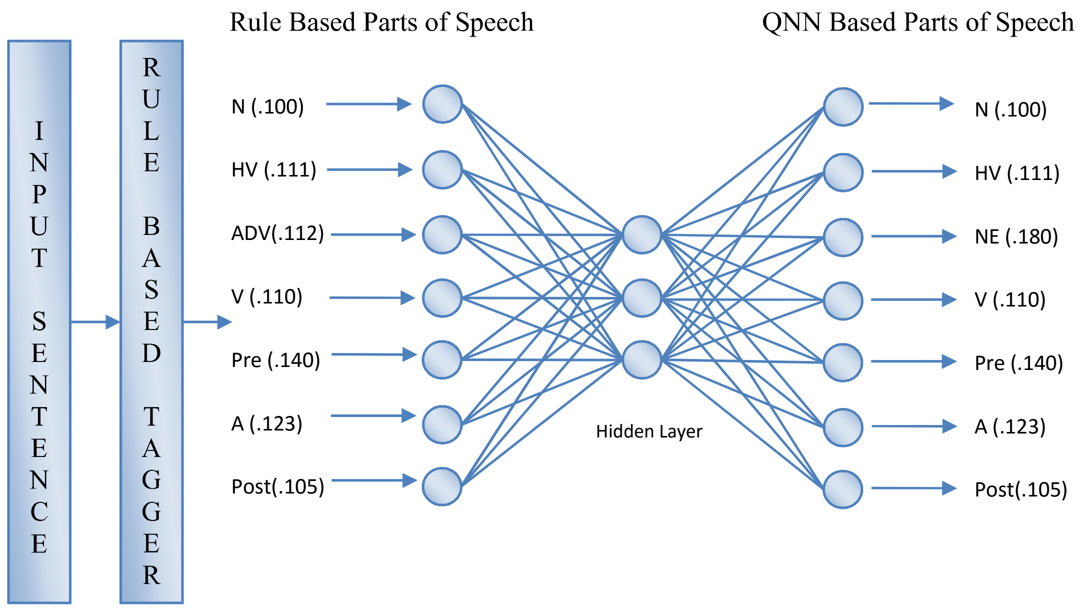


**Figure 5.3 Architecture of Quantum Neural Network for Parts of Speech Tagging**

The above Figure 5.3 explains the QNN architecture used for POS Tagging. In this Figure 5.3 the input to QNN and the output is shown by three digit numeric codes.

In this, input to QNN is generated by Rule based tagger, which is not accurate in context of sentence. The outcome of rule based tag further passes though the QNN. As shown in Figure 5.3 the outcome of rule based tagger *i.e.* N (.100) HV (.111) ADV (.112) V (.110) Pre (.140) A (.123) Post (.105) and the outcome of QNN based tagger is as N (.100) HV (.111) NE (.180) V (.110) Pre (.140) A (.123) Post (.105) with its accurate POS tagging in context of which the sentence is used for. Unlike the example shown above, the outputs of the network are not perfectly integer. Thus the outputs must be round off to the nearest integer and some basic error corrections are necessary to obtain the symbolic codes.

### 5.3.2. Proposed algorithm for Complex Sentences

QNNMTS (SENTENCE, TOKEN, N, LOC)

Here SENTENCE is an array with N elements containing words. Parameter TOKEN contains the Token of each word, and LOC keeps track of position.

ICOUNT contains the max number of Interrogative words encountered in the sentence, NCOUNT contains the max number of negative words encountered in the sentence, and CCOUNT contains the max number of conjunction words encountered in the sentence, ILOC, CLOC and NLOC contains location of interrogative word, conjunction word and negative word respectively.

**Step 1:** Check whether the sentence is Complex sentence or Simple sentence

```
            [Initialize.] Set ICOUNT=1, NCOUNT=1, CCOUNT=1, ILOC, NLOC,
            CLOC.
            Repeat for LOC = 1 to N:
                  if SENTENCE[LOC] = "Interrogative word" , then:
                        Set ILOC[ICOUNT] :=LOC,
                        ICOUNT=ICOUNT+1.
                              [End of if structure]
                  if SENTENCE[LOC] = "Negative word" , then:
                        Set NLOC[NCOUNT] :=LOC,
                        NCOUNT=NCOUNT+1.
                              [End of if structure]
                  if SENTENCE[LOC] = "Conjunction", then:
                        Set CLOC[CCOUNT] :=LOC.
                        CCOUNT=CCOUNT+1
                              [End of if structure]
            [End of for]
            if ILOC= NULL, or NLOC=NULL, or CLOC=NULL, then:
                  Go to Step 5.
```

**Step 2:** Remove the interrogative word from the complex sentence to make it

```
            Affirmative.
                  Repeat for X= 1 to ICOUNT
                        Set ITemp[X] := SENTENCE[ILOC[X]].
```

Set SENTENCE[ILOC[X]] :=Null.

[End of for]

**Step 3:** Then Remove the negative to make it simple sentence

Repeat for Y= 1 to NCOUNT

Set NTemp[Y] := SENTENCE[NLOC[Y]].

Set SENTENCE[NLOC[Y]] :=Null.

[End of for]

**Step 4:** Split the sentence into two or more simple sentences on the basis of conjunction.

Repeat for Z= 1 to CCOUNT

Set CTemp[Z] := SENTENCE[CLOC[Z]].

Set SENTENCE[CLOC[Z]] := Hindi Full-stop (“ | ”).

[End of for]

**Step 5:** Pass each sub sentence with TOKEN to QNN based MT for reposition.

**Step 6:** Refine the Translated sentences by applying the grammar rules.

**Step 7:** Add the interrogative word if removed in Step 2.

Repeat for X= 1 to ICOUNT

if ITEMP[X] = NOT NULL

Set SENTENCE[ILOC[X]] := ITemp[X].

[End of if structure]

[End of for]

**Step 8:** Add the negative word if removed in Step 3.

Repeat for Y= 1 to NCOUNT

if NTEMP[Y] = NOT NULL

Set SENTENCE[NLOC[Y]] := NTemp[Y].

[End of if structure]

[End of for]

**Step 9:** Rejoin the entire sub sentences, if split in Step 4.

**Step 10:** Semantic Translation

**Step 11:** Exit.

**For Example:**

**Sentence:** The plane had not takenoff before John reached the airport.

After passing the sentence to algorithm for Complex Sentences

**Step 1:** Check whether the inputted sentence is complex sentence or simple sentence by checking the Interrogative word, Negative word, Conjunction.

Interrogative = False and Interrogative word= Null
Negative =True and Negative word= "not" and Negative word location =4
Conjunction = True and Conjunction= "before" and Conjunction word location =6

That sentence found as Complex Sentence.

**Step 2:** Remove the interrogative word from the complex sentence to make it affirmative.

Not applicable for this example sentence.

**Step 3:** Then remove the negative to make it simple sentence

The plane had takenoff before John reached the airport--------"not" removed

**Step 4:** Split the sentence into two or more simple sentences on the basis of conjunction.

[The plane had takenoff] [John reached the airport]

**Step 5:** Pass each sub sentence with token to QNN based machine translator for reposition.

[ The plane had taken off] [ John the airport reached]
----- according to Hindi language structure

**Step 6:** Refine the translated sentences by applying the grammar rules.

[ The plane had taken off] [ John the airport reached]

**Step 7: -** Add the interrogative word if removed in step 2.

Not applicable for this example sentence.

**Step 8:** Add the negative word if removed in step 3.

[ The plane had not taken off] [ John the airport reached]

**Step 9:** Rejoin the entire sub sentences, if split in step 4.
[John the airport reached before the plane had not taken off]

**Step 10:** Semantic translation
[जॉन के हवाई अड्डे पहुचने से पहले हवाई जहाज नही उडा था। ]

**Step 11:** Exit.

### 5.3.3. Quantum Neural Implementation of translation Rules

The strategy is to first identify and tag of parts of speech by using the Table 5.1 and then translate the Source Language sentences literally into Target Language with no rearrangement of words. After syntactic translation the rearrangement of the words has done to build accurate translation with appropriate sense. The rules are based on parts of speech, not based on meaning.

For a special case when input sentence and the resulting sentence are having unequal number of words, then the dummy numeric code .000 is used for giving a similar word alignment.

***Case 1:*** *When input sentence and the resulting sentence are having unequal number of words.*

The coded version of sentence is thus

The Trinamool supremo said a definitive no. (In English Language)

.123 .100 .102 .110 .123 .130 .180

तृणमूल मुखियाने निश्चित ना कहा। (In Hindi Language)

.100 .102 .130 . 180 . 110 .000 .000

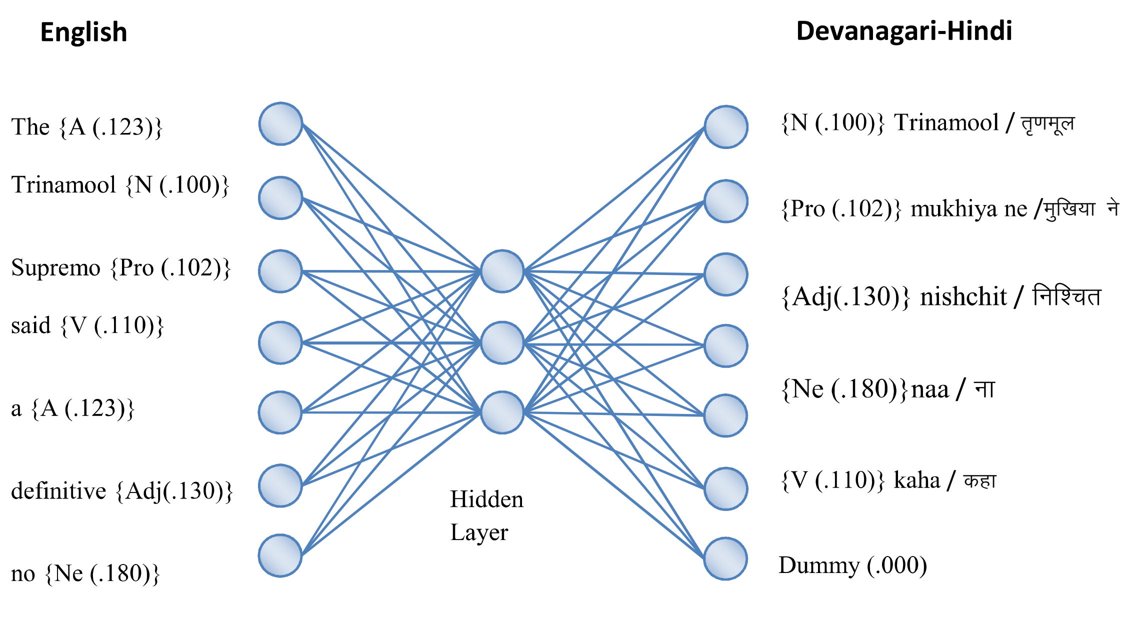


**Figure 5.4 Example of input/ out to the QNN used for MT**

Thus, the input code sequence is [.123 .100 .102 .110 .123 .130 .180] and the corresponding output is [.100 .102 .130. 180 .110 .000 .000]. Here, dummy numeric code

.000 is used as shown in Figure 5.4. The outcome of neural network might not be the perfect integer, it should be round off and few basic error adjustments might be needed to find the output numeral codes. Even the network is likely to arrange the location of 3 digit codes. By this, it learns the target language knowledge which is needed for semantic rearrangement, and also helps in POS tagging, by pattern matching. It is also helpful to adopt and learn the grammar rules up to a level. For handling the complex sentences the algorithm is used. The algorithm first removes the interrogative and negative words, on the basis of conjunction the system breaks up and converts the complex sentence into two or more small simple sentences. After the translation of each of the simple sentence, the system again rejoins the entire sub-sentences and also adds the removed interrogative and negative words in the sentence. The whole process is explained in the above section 5.3.2.

### 5.3.4. Semantic Translation

Semantic translation is a branch of linguistics which studies about meaning. The main objective of semantics translation is to transfer the literal meaning from the source language to the target language, word by word. Without the correct meaning of every word in the corpus, it is impossible to translate one language to another. Thus the semantic translation plays a very important role in machine translation. For semantic translation each word of source language directly mapped using the lexicon/ dictionary for its associate meaning of target language. The main problem with the semantic translation is ambiguous words.

### 5.3.5. Approach for handling ambiguous words.

In every language many words have multiple meanings and each meaning is meaningful to its own scenario *i.e.* the meaning is ambiguous. Some examples of ambiguous words are given below, in which the word "shoot" is common in sentence 1, 2 and 3 but having different meanings. Similarly the word "plant" is also an ambiguous word, which is common in sentence 3 and 4 but having different meanings.

**Sentence 1:** **Shoot** with Gun.

**Sentence 2:** **Shoot** with Camera.

**Sentence 3:** **Shoot** of a **Plant**.

**Sentence 4:** I have to **Plant** a Tree.

In the proposed system we have created a lexicon/ dictionary of approximately 8000 words and many words are having multiple meaning *i.e.* the meanings are ambiguous.

The proposed system also takes care of the ambiguous words. As per our study, if the word meanings of ambiguous words are mapped in respect of their parts of speech and their neighbouring noun/pronoun encountered in the sentence, then we can (or the accuracy may be increased) increase the accuracy in mapping of correct word meaning of ambiguous words from the lexicon/dictionary.

**Table 5.2: Lexicon Structure**

| **Word** | **Meaning** | **Parts of speech** | **Gender Information (Male or Female)** | **Quantitative Information (Singular/ Plural)** | **Correlation of word meaning with specific Noun(Only in case ambiguous words)** |
|---|---|---|---|---|---|
| Shoot | गोली मारना | Verb | N/A | N/A | Gun |
| Shoot | फोटो खीचना | Verb | N/A | N/A | Camera |
| Shoot | अन्कुर | Verb | N/A | N/A | Class Veg |
| Plant | पैघा | Noun | N/A | N/A | Class Veg |
| Plant | लगाना | Verb | N/A | N/A | To |

For implementing this process in the lexicon/dictionary, the words are having ambiguous meanings which are stored with its every meaning as separate entity. Some additional information with words meanings are also stored, like parts of speech, gender information, quantitative information *i.e.* singular or plural, and information about the correlation of word meaning with specific noun or class of nouns. The structure of lexicon/dictionary is shown below in the Table 5.2. By taking the example of sentence 1, 2, 3 and 4 and applying the above said process we may easily find out the correct meaning of ambiguous words. The algorithm of this process is already given as procedure "Ambiguous word handler" in Quantum Neuro tagger algorithm of section 5.3.1.5.

## 5.4. Conclusion

In this chapter we have discussed the details of QNN model for machine translation for Hindi to English and vice versa. The complete architecture of the proposed model is discussed in detail. The proposed algorithm for handling the complex sentences is also discussed. Moreover, the method used to implement the machine translation by using the QNN is also included. The coding mechanism is used to decode and encode the sentences into numeric code to pass these to QNN. This chapter also covers the QNN based pats of speech tagging in detail.

# CHAPTER 6

# OPTIMAL LEARNING RULE FOR PROPOSED QNN BASED MT

# Chapter 6

# Optimal learning rule for Proposed QNN based MT.

In this chapter, the optimal value of Quantum Interval **($\theta^r$)** has been obtained to identify the Optimal learning rule for QNN model. The optimum rules are different for English to Hindi translation compared to Hindi to English translation. The rule algorithm for parts of speech tagging (intermediate step of MT) for Hindi and English have been identified.

The experiments have been carried out to achieve lowest possible Mean Square Error (MSE) for training, validation and testing of the model. The random data set has been used for training, validation and test with POS tags of 2600 sentences. The dataset is divided in 4:3:3 ratios respectively for training, validation and testing from 2600 English sentences and their Devanagari-Hindi translations. Separate optimal learning rules for QNN have been developed for Hindi to English MT system, English to Hindi MT system, Hindi POS tagger and English POS tagger.

In every experiment, the results between traditional ANN and QNN have been compaired. The equvilent architectures of the ANN and QNN have been used to compair the results. Both of them have three layers Architecture, which have inputs layer of 10 nodes, one hidden layer of 10 nodes, and output layer of 10 nodes. Only the diffierence between QNN and ANN is that, in QNN each hidden layer neuron having two excitation states which virtually represent two more nodes but any one of the node can be used for learning. In ANN and QNN, the simple sigmoid function has been used as the activation function of output layer. At Hidden layer for ANN, the simple sigmoid function has been used, on the other hand for QNN, the quantum sigmoid function with various graded levels has been used. The Levenberg-Marquardt method has been used as training method in both ANN and QNN.

### 6.1. Experiment 1: Optimisation of QNN model for English POS Tagger

Experiment-1 has been performed for identifying the optimal learning rule, used in QNN based English POS tagger. The optimal learning rule has been achieved by optimizing the value of Quantum Interval **($\theta^r$)** for Equation 4.6. With the different values of Quantum Interval **($\theta^r$),** the same experiment is done several times to identify the optimal value of Quantum Interval **($\theta^r$)**. This experiment is carried out with the random data sets for training, validation and testing from POS tags of 2600 English sentences.

The results shown in Table 6.1 are the average of 500 computed results. In Table 6.1, the best performance is shown for value of Quantum Interval **($\theta^r$)** 2.75 with respect to all the parameters *i.e*. epoch or iterations needed to train the network, the training, validation and test performance. The first row of table *i.e.* at $\theta = 0$ represents the training value of ANN model and rest of the rows show the training performances of QNN model with different value of $\boldsymbol{\theta^r}$.

**Table 6.1: Results of performance parameter of QNN and ANN for English POS tagger**

| S.No | Quantum Interval (θ) | Epoch (average Iterations) | Training performance (MSE) | Validation performance (MSE) | Test performance (MSE) |
|---|---|---|---|---|---|
| 1 | θ= 0=> ANN | 21.608782 | 0.002041 | 0.002036 | 0.002026 |
| 2 | 0.25 | 16.261477 | 0.000102 | 0.000104 | 0.000103 |
| 3 | 0.5 | 16.367265 | 0.000240 | 0.000241 | 0.000240 |
| 4 | 0.75 | 16.209581 | 0.000116 | 0.000114 | 0.000116 |
| 5 | 1 | 16.221557 | 0.000362 | 0.000362 | 0.000367 |
| 6 | 1.25 | 15.906188 | 0.000298 | 0.000297 | 0.000298 |
| 7 | 1.5 | 16.309381 | 0.000142 | 0.000148 | 0.000140 |
| 8 | 1.75 | 16.606786 | 0.000281 | 0.000284 | 0.000284 |
| 9 | 2 | 16.822355 | 0.000239 | 0.000246 | 0.000238 |
| 10 | 2.25 | 17.013972 | 0.000169 | 0.000166 | 0.000167 |
| 11 | 2.5 | 16.828343 | 0.000135 | 0.000135 | 0.000137 |
| **12** | **2.75** | **17.389222** | **0.000038** | **0.000038** | **0.000038** |
| 13 | 3 | 18.153693 | 0.000040 | 0.000042 | 0.000040 |
| 14 | 3.25 | 18.201597 | 0.000070 | 0.000070 | 0.000070 |
| 15 | 3.5 | 19.011976 | 0.000102 | 0.000104 | 0.000103 |

| | | | | | |
|---|---|---|---|---|---|
| 16 | 3.75 | 19.369261 | 0.000134 | 0.000136 | 0.000135 |
| 17 | 4 | 20.241517 | 0.000037 | 0.000037 | 0.000038 |

### 6.2. Experiment 2: Optimisation of QNN model for Hindi POS Tagger

To identify the optimal learning rule for the QNN based Hindi POS tagger the experiment-2 has been performed. The value of Quantum Interval **($\theta^r$)** for Equation 4.6 has been optimised for achieving the optimal learning rule. The same experiment is carried out many times for the different values of Quantum Interval **($\theta^r$)**, to identify the optimal value of Quantum Interval **($\theta^r$)**. This experiment is performed with the random data sets for training, validation and testing from POS tags of 2600 Hindi sentences. The results shown in Table 6.2 are the average of 500 times calculated results. In Table 6.2, the best performance is shown for value of Quantum Interval **($\theta^r$)** 3.5 with respect of all the parameters *i.e.* Epoch or iterations needed to train the network, the training, validation and test performance. The first row of table *i.e.* at $\theta = 0$ represents the training value of ANN model and rest of the rows show the training performances of QNN model with different value of $\boldsymbol{\theta^r}$.

**Table 6.2: Training, Testing and Validation of QNN model for Hindi POS Tagger**

| S.No | Quantum Interval(θ) | Epoch (average Iterations) | Training performance(MSE) | Test performance(MSE) |
|---|---|---|---|---|
| **1** | **ANN** | **20.81437** | **0.001502** | **0.001495** |
| 2 | 0.25 | 15.68663 | 0.000216 | 0.000223 |
| =3 | 0.5 | 15.56088 | 0.000360 | 0.000370 |
| 4 | 0.75 | 16.07784 | 0.000074 | 0.000079 |
| 5 | 1 | 15.82635 | 0.000265 | 0.000273 |
| 6 | 1.25 | 15.89421 | 0.000264 | 0.000273 |
| 7 | 1.5 | 16.09381 | 0.0004 | 0.000409 |
| 8 | 1.75 | 15.49301 | 0.000362 | 0.000369 |
| 9 | 2 | 15.61677 | 0.000272 | 0.000277 |
| 10 | 2.25 | 16.3513 | 0.000329 | 0.000338 |
| 11 | 2.5 | 16.42715 | 0.000311 | 0.000327 |

| | | | | |
|---|---|---|---|---|
| 12 | 2.75 | 16.68663 | 0.000137 | 0.000149 |
| 13 | 3 | 18.05988 | 0.000201 | 0.000211 |
| 14 | 3.25 | 17.76846 | 0.000202 | 0.000207 |
| **15** | **3.5** | **19.12176** | **0.000041** | **0.000049** |
| 16 | 3.75 | 19.21357 | 0.000073 | 0.000081 |
| 17 | 4 | 20.15968 | 0.000072 | 0.000082 |

## 6.3. Experiment 3: Optimisation of QNN model for English to Hindi MT System

Experiment-3 has been carried out for identifying the optimal learning rule for QNN based Hindi to English machine translation system. The optimal learning rule has been achieved by optimizing the value of Quantum Interval **($\theta^r$)** for Equation 4.6. With the different values of Quantum Interval **($\theta^r$),** the same experiment is done a number of times to identify the optimal value of Quantum Interval **($\theta^r$)**. This experiment is performed with the random data sets for training, validation and testing from POS tags of 2600 English with its Hindi translated sentences. The results shown in Table 6.3 are the average of 500 times calculated results. In Table 6.3, the best performance is shown for value of Quantum Interval **($\theta^r$)** 2.25 with respect of all the parameters *i.e.* Epoch or iterations needed to train the network, the training, validation and test performances. The first row of table *i.e.* at $\theta = 0$ represents the training value of ANN model and rest of the rows show the training performance of QNN model with different value of **$\theta^r$**.

**Table 6.3: Results of performance of QNN and ANN for English to Hindi MT system**

| S.No | Quantum Inte=rval(θ) | Epoch (average Iterations) | Training performance (average MSE) | Validation performance (average MSE) | Test performance (average MSE) |
|---|---|---|---|---|---|
| **1** | **θ=0=> ANN** | **20.58** | **0.00162840** | **0.00163146** | **0.00167585** |
| 2 | 0.25 | 15.81 | 0.00006845 | 0.00007465 | 0.00007460 |
| 3 | 0.5 | 16.64 | 0.00003666 | 0.00004423 | 0.00004053 |
| 4 | 0.75 | 16.40 | 0.00010053 | 0.00010867 | 0.00010874 |
| 5 | 1 | 16.02 | 0.00006894 | 0.00007508 | 0.00007592 |
| 6 | 1.25 | 16.24 | 0.00006850 | 0.00007437 | 0.00007431 |

| | | | | | |
|---|---|---|---|---|---|
| 7 | 1.5 | 15.89 | 0.00003667 | 0.00004250 | 0.00004195 |
| 8 | 1.75 | 16.15 | 0.00003565 | 0.00004415 | 0.00004411 |
| 9 | 2 | 16.10 | 0.00006816 | 0.00007560 | 0.00007471 |
| **10** | **2.25** | **16.12** | **0.00003628** | **0.00004337** | **0.00004331** |
| 11 | 2.5 | 16.63 | 0.00006861 | 0.00007665 | 0.00007528 |
| 12 | 2.75 | 16.72 | 0.00003673 | 0.00004357 | 0.00004267 |
| 13 | 3 | 16.52 | 0.00003611 | 0.00004363 | 0.00004285 |
| 14 | 3.25 | 17.06 | 0.00003544 | 0.00004510 | 0.00004417 |
| 15 | 3.5 | 18.02 | 0.00006807 | 0.00007557 | 0.00007731 |
| 16 | 3.75 | 18.01 | 0.00007494 | 0.00008566 | 0.00008155 |
| 17 | 4 | 18.48 | 0.00003583 | 0.00004504 | 0.00004302 |

### 6.4. Experiment 4: Optimisation of QNN model for Hindi to English MT System

To identify the optimal learning rule for the QNN based Hindi to English machine translation, the experiment-4 has been carried out. The value of Quantum Interval **($\theta^r$)** for Equation 4.6 has been optimised for achieving the optimal learning rule. The same experiment is performed out many times for the different values of Quantum Interval **($\theta^r$)**, to identify the optimal value of Quantum Interval **($\theta^r$)**. This experiment is done with the random data sets for training, validation and testing from POS tags of 2600 Hindi with its English translated sentences. The results shown in Table 6.4 are the average of 500 times calculated results. In Table 6.4, the best performance is shown for value of Quantum Interval **($\theta^r$)** 1 with respect of all the parameters *i.e*. Epoch or iterations needed to train the network, the training, validation and test performances. The first row of table *i.e.* at θ = 0 represents the training value of ANN model and rest of the rows show the training performance of QNN model with different value of **$\theta^r$**.

**Table 6.4: Results of performance parameter of QNN and ANN for Hindi to English MT**

| S.No | Quantum Interval(θ) | Epoch (average Iterations) | Training performance (MSE) | Validation performance (MSE) | Test performance (MSE) |
|---|---|---|---|---|---|
| **1** | **θ= 0=> ANN** | **20.04391** | **0.002027** | **0.002066** | **0.002075** |

| | | | | | |
|---|---|---|---|---|---|
| 2 | 0.25 | 15.36727 | 0.000185 | 0.000205 | 0.000203 |
| 3 | 0.5 | 15.47305 | 0.000248 | 0.000273 | 0.000273 |
| 4 | 0.75 | 15.28343 | 0.000231 | 0.000248 | 0.000252 |
| **5** | **1** | **15.48303** | **0.00025** | **0.00027** | **0.000272** |
| 6 | 1.25 | 15.67066 | 0.000284 | 0.0003 | 0.0003 |
| 7 | 1.5 | 15.64271 | 0.000296 | 0.000318 | 0.000316 |
| 8 | 1.75 | 15.78643 | 0.000349 | 0.000374 | 0.000375 |
| 9 | 2 | 15.68663 | 0.000184 | 0.000204 | 0.000209 |
| 10 | 2.25 | 15.96607 | 0.000249 | 0.000273 | 0.000273 |
| 11 | 2.5 | 16.27345 | 0.000256 | 0.000281 | 0.000276 |
| 12 | 2.75 | 16.54092 | 0.00022 | 0.000242 | 0.000242 |
| 13 | 3 | 16.49301 | 0.000348 | 0.000361 | 0.00037 |
| 14 | 3.25 | 16.93214 | 0.000192 | 0.000214 | 0.000216 |
| 15 | 3.5 | 17.76248 | 0.000294 | 0.000315 | 0.000323 |
| 16 | 3.75 | 17.85429 | 0.000185 | 0.000202 | 0.000207 |
| 17 | 4 | 18.72056 | 0.000217 | 0.000237 | 0.00024 |

## 6.5. Conclusion

In this chapter, we have identified the optimal learning rule for proposed QNN based machine translation model. Different experiments have been performed with QNN and the results are also compared between traditional ANN and quantum neural networks. The optimum rules are different for English to Hindi translation compared to Hindi to English translation. The learning rule for parts of speech tagging (intermediate step of MT) for Hindi and English are also identified. Optimal learning rules have been achieved after 500 independent tests performed with the system for each value of quantum interval (θ) with random data sets. The identified optimal learning rules are used in QNN based MT System for Hindi to English and English to Hindi and in QNN based POS tagger for Hindi and English.

# CHAPTER 7

# TESTING AND VALIDATE THE PROPOSED QNN BASED MT MODEL

Chapter 7

# Testing and validation of the proposed QNN based MT model.

In this chapter, the identifying ability of the Quantum Neural Network (QNN) based MT model has been analyzed. To test and validate the proposed model with 2600 sentences of news items in English and their corresponding human translated Devanagari-Hindi sentences are used as input/output as given in Table 7.1. Here, the 2600 English sentences contain 11500 words and their equivalent translated Hindi sentences contain 12500 words, this corpus of words is used for testing and validation of QNN based POS tagger. The news papers used as sources are Times of India (www.timesofindia.indiatimes.com), Reuters (http://in.reuters.com/), and Hindustan Times (www.hindustantimes.com). Only grammatically correct English sentences are collected and their corresponding Devanagari-Hindi translations are also performed manually, which is used for validation of the model. Each English sentence may consist of words with Question Word, Noun, Helping Verb, Negative Word, Verb, Preposition, Article, Adjective, Post Noun and Adverb *etc*. Each Devanagari-Hindi sentence may contain a Question Word, Noun, Helping Verb, Negative Word, Verb, Preposition, Article, Adjective, Post Noun and Adverb *etc*.

**Table 7.1: Input/Output sentences used for testing and validation**

| # | English - input sentences | Devanagari-Hindi – output sentences |
|---|---|---|
| 1. | The court has also issued a notice to the state government. | Nyayalaya ne Pradesh sarkar ko bhi soochna jaari ki hai<br>न्यायालय ने प्रदेश सरकार को भी सूचना जारी की है। |
| 2. | All great players retire one day. | Samast mahaan khiladi ek din nivrata hote hai.<br>समस्त महान खिलाडी एक दिन निवृत्त होते हैं। |

| | | |
|---|---|---|
| 3. | You need someone to be dedicated, be committed and disciplined. | Tumhe uski avashyakta hai jo samarpit, vachanbadh avam anushasit ho.<br><br>तुम्हे उसकी आवश्यकता है जो समर्पित, वचनबद्ध एवं अनुशासित हो। |
| 4. | Sachin said Dravid will be surely missed in the dressing room. | Sachin ne kaha ki Dravid avashya srangar kaksha me yaad kiye jayege.<br><br>सचिन ने कहा कि द्रविड अवश्य श्रृंगार कक्ष मे याद किये जायेगे। |
| 5. | The Trinamool supremo said a definitive no. | Trinamool mukhiya ne nishchit naa kaha.<br><br>तृणमूल मुखिया ने निश्चित ना कहा। |
| 6. | The Indian government has also desired to attract investments from Venezuela. | Bahrtiya sarkaar ne Venezuela se nivesh ko aakarshit karne ki bhi iccha vyakt ki.<br><br>भारतीय सरकार ने वेनेजुएला से निवेश के आकर्षित करने की भी इच्छा व्यक्त की। |
| 7. | The insurgents have vowed to carry their fight across Yemen. | Yudhrat vidrohi Yemen ke virudh apni ladai jaari rakhne ki shapath kha chuke hain.<br><br>युद्धरत्त विद्रोही यमन के विरुद्ध अपनी लडाई जारी रखने की शपथ खा चुके हैं। |
| 8. | They are planning big strikes in Kashmir | Ve kashmir mai hamle ki saajish kar rahe hai.<br><br>वे कश्मीर में बडे हमले की साजिश कर रहे हैं। |

### 7.1. Methods used for Evaluation and Validation of proposed QNN based MT model

Evaluation and Validation has been performed by translating the same set of input sentences by using the proposed system, Google translation and Bing translation. The accuracy of proposed quantum neural network based MT system for Devanagari (Hindi) to English and vice versa has been compared on different scores viz. BLEU, NIST, ROUGE-L, METEOR and human based. The human based translation has been performed because the automatic MT evaluation methods cannot sense the meaning of translation, as compaired to the human based.

### 7.1.1. Evaluation by BLEU (Bilingual evaluation understudy)

BLEU score has been used to compute the score of MT System outcome. BLEU is developed by IBM. In BLEU, modified n-gram precision is used to evaluate candidate's translations by compairing with reference translations. (Papineni *et al*., 2002).

On BLEU scale the accuracy of outcome is always shown on the scale 0 to 1. Here, 1 represents maximum similarity between machine (candidate) translated sentence and human (reference) translated sentence.

We first compute the *n*-gram (sentence having n number of words) by maching candidate sentence with reference sentence, word by word. Next, we add the clipped *n*-gram counts for every candidate sentences and divide by the number of candidate *n*-grams in the test corpus to compute a modified precision score, $p_n$, for the entire test corpus.

$$P_n = \frac{Total\ number\ of\ common\ words\ between\ candidte\ sentence\ and\ reference\ translation}{Toal\ number\ of\ word\ in\ candidate\ translation} \quad \ldots(7.1)$$

The Brevity Penalty (BP) is followed by using the Equation 7.2:

$$BP = \begin{cases} 1 & if\ c > r \\ e^{(1-r/c)} & if\ c \le r \end{cases} \quad \ldots(7.2)$$

Here $c$ is number of words in candidate translation and $r$ is the number of words in reference translation.

The BLEU score is calculated by using Equation 7.4

$$BLEU = BP.\exp\left(\sum_{n=1}^{N} w_n \log\ p_n\right) \quad \ldots(7.3)$$

The above Equation 7.3 can be expressed as:

$$\log\ BLEU = \min\left(1 - \frac{r}{c}, 0\right) + \left(\sum_{n=1}^{N} w_n \log\ p_n\right) \quad \ldots(7.4)$$

Here the $w_n$ is a positive weight and $w_n = 1/N$, $N$ is number of words in longest candidate translation.

### 7.1.2. Evaluation by NIST (National Institute of Standard and Technology).

This is proposed by NIST (national institute of standard and technology). It reduces the effect of longer n-grams by using arithmetic mean over n-grams counts instead of geometric mean of co-occurrences over N (Doddington, 2002, George, 2002). On NIST scale the accuracy of outcome is always shown on the scale 0 to 10. This value indicates,how informative a particular n-gram text is, with values closer to 10 representing more informative the text is:

$$BP_{NIST} = exp\left\{\beta * log^{2}\left[min\left(\frac{Len_{Can}}{Len_{Ref}}, 1\right)\right]\right\} \qquad \ldots(7.5)$$

Where $Len_{Can}$ is total word count in candidate sentence and $Len_{Ref}$ is average word count in all reference translation. The brevity penalty factor $\beta$ must be taken as very small number. $N$ is word count in longest candidate translation.

The precision is calculated by using:

$$PRECISION_{NIST} = \sum_{n=1}^{N}\left\{\frac{Info_{(w_1 \ldots w_n)}}{Total\ number\ of\ words\ in\ all\ candidate\ translations}\right\} \qquad \ldots(7.6)$$

Where

$$Info_{(w_1 \ldots w_n)} = -log_2\left(\frac{Total\ number\ of\ words\ in\ all\ reference\ translations}{Total\ number\ of\ common\ words\ between\ candidate\ and\ all\ reference\ translations}\right) \qquad \ldots(7.7)$$

NIST score is calculated by using:

$$NIST_{score} = BP_{NIST} * PRECISION_{NIST} \qquad \ldots(7.8)$$

### 7.1.3. Evaluation by ROUGE-L (Recall-Oriented Understudy for Gisting Evaluation- Longest Common Subsequence)

ROUGE-L (Recall-Oriented Understudy for Gisting Evaluation- Longest Common Subsequence) that measures sentence-to-sentence similarity based on the longest common subsequence statistics between a candidate translation and a set of reference translations.

On ROUGE-L scale the accuracy of outcome is always shown on the scale, 0 to 1. Here, 1 represents maximum similarity based on the longest common subsequence statistics between machine (candidate) translated sentence and human (reference) translated sentence.

To apply Longest Common Subsequence (LCS) in machine translation evaluation, every sentence and its translation is the sequence of words. The perception is that, longer is the LCS of two translations, more similar the two translations are. (Lin *et al.*, 2004).

Here $X$ represents reference translation and $Y$ represents candidate translation. $m$ is word count in reference translation and $n$ is word count in candidate translation.

$$R_{lcs} = \frac{LCS(X,Y)}{m} \qquad \ldots(7.9)$$

$$P_{lcs} = \frac{LCS(X,Y)}{n} \qquad \ldots(7.10)$$

$$F_{lcs} = \frac{(1+\beta^2)R_{lcs}P_{lcs}}{R_{lcs} + \beta^2 P_{lcs}} \qquad \ldots(7.11)$$

*Where* $P_{lcs}$ is Precision and $R_{lcs}$ is Recall and $LCS(X,Y)$ is the length of a longest common subsequence of $X$ and $Y$, and $\beta = P_{lcs}/R_{lcs}$

$$Rouge_L = Harmonic\ Mean(P_{lcs}, R_{lcs}) = (2 * P_{lcs} * R_{lcs}) / (P_{lcs} + R_{lcs}) \qquad \ldots(7.12)$$

### 7.1.4. Evaluation by METEOR (Metric for Evaluation of Translation with Explicit Ordering)

METEOR (Metric for Evaluation of Translation with Explicit Ordering) is an MT evaluation metric, which has been developed at Carnegie Mellon University. The Meteor metric is based on the weighted harmonic mean of

Unigram precision $\left(P = \frac{m}{w_c}\right)$ and $\qquad \ldots(7.13)$

Unigram recall $\left(R = \frac{m}{w_r}\right)$ $\qquad \ldots(7.14)$

Here m is total number of common words between reference and candidates translations, $w_c$ is the total number of words in candidate translation (t) and $w_r$ is the total number of words in reference translation(r).

$F_{mean}$ is calculated by combining the recall and precision via a harmonic-mean that places equal weight on precision. Again α must be taken as very small number.

$$\left( F_{mean} = \frac{PR}{\alpha P + (1-\alpha)R} \right) \quad \ldots(7.15)$$

This measure is for congruity with respect to single words but for considering longer n-gram matches, a penalty p is calculated for the alignment as:

$$\left( p = 0.5 \left( \frac{c}{u_m} \right)^3 \right) \quad \ldots(7.16)$$

Here $c$ is total number of fragmentation (of one or more words) between reference translation and candidate translation, and $u_m$ is total number of common words between reference translation and candidate translation. (Banerjee 2005, Lavie *et al.* 2009).

Final Meteor-score (M-score) can be calculated as:

$$Meteor_{score} = F_{mean}(1-p) \quad \ldots(7.17)$$

On Meteor scale the accuracy of outcome is always shown on the scale 0 to 1. Here, 1 represents maximum similarity based on the longest common subsequence statistics between machine (candidate) translated sentence and human (reference) translated sentence. During evaluation if multiple reference translation are available, then the given translation is scored against each reference independently, and the best score is reported.

Meteor word matches between translation and references includes semantic equivalents.

## 7.2. Experiment and Results

In this section experiments are carried out to validate the QNN based model for Hindi to English, English to Hindi machine translator, Hindi and English POS tagger. For validation purpose same sets of English and Hindi sentences have been tested with proposed system, Bing Translation, Google Translator and Anuvadaksh. Their comparison charts are also shown in this section.

### 7.2.1. Validation of QNN based English POS Tagger

The validation of QNN based English POS tagger has been carried by comparing the results with rule based POS tagger and proposed QNN based POS tagger. In the experiment all words in English sentence are assigned with a unique numeric code as POS tag. The three digits numeric codes have been used to encode all the words in English sentences. As shown in Figure 5.4, the encoding scheme has produced numeric codes in the input layer and output layer of the Quantum Neural Network. All the errors in Parts of Speech (POS) tagging of words in English sentences have been evaluated and recorded. The parts of speech distribution for English sentences according to their number and percentage are shown in Table 7.2.

**Table 7.2: POS Distribution with 2600 English Sentences**

| **Parts of Speech** | **Number wise POS Distribution with English** | **Percentage wise POS Distribution for English (%)** |
|---|---|---|
| Question Word | 500 | 4.00 |
| Noun | 2500 | 20.00 |
| Helping Verb | 2300 | 18.40 |
| Negative Word | 900 | 7.20 |
| Verb | 1900 | 15.20 |
| Preposition | 700 | 5.60 |
| Article | 900 | 7.20 |
| Adjective | 700 | 5.60 |
| Post Noun | 1500 | 12.00 |
| Adverb | 600 | 4.80 |
| **Total** | **12500** | |

The QNN memorizes the pattern of POS, based on the Input pair of POS set. Here for testing purpose the rule based POS of English sentences has been used as input to QNN based POS tagger. During experiments it is identified that the QNN easily learns and identifies the POS of preposition due to the grammar structure of English language. On other hand, for QNN it is hardest to learn to tag the correct POS tagging between the adjective and the second noun. Furthermore, it is also slightly harder to learn to tag the correct parts of speech of adverb in English grammar, due to the random change in positions of the verb and adverb. Figure 7.1 below, clearly shows that the proposed English parts of speech tagger correctly disambiguates and correctly identifies the POS with higher accuracy.

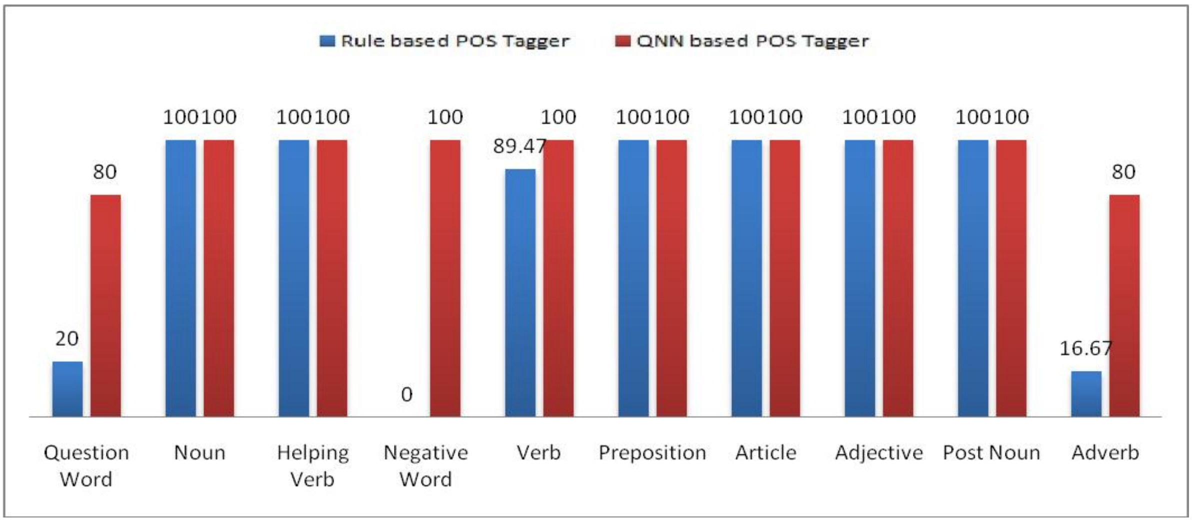


**Figure 7.1: Bar diagram for accuracy comparison between rule based POS tagging and QNN based tagging**

The accuracy based on the categories of POS is shown in the Figure 7.1. For the categories having low accuracy, such Question Word, Negative Word, Verb, Adverb. It is observed that all of them are highly ambiguous and almost invariable. Most of them are also hard to disambiguate without any semantic information.

Experiments show that during learning process with QNN based POS Tagger for English, there is decrease in indeterminacy of pattern recognition and increase in authenticity of pattern recognition of POS. Hence, by using POS tagger with QNN, the proposed system has achieved a better POS tagging with higher accuracy in comparison to other existing approaches.

### 7.2.2. Validation of QNN based Hindi POS Tagger

The proposed QNN based POS tagger for Hindi has been validated by comparing the results with rule based Hindi POS tagger. In the experiment all words of Hindi sentence are tagged with POS tag using unique numeric code as discussed in section 5.3.1.3. It is possible to use three numeric codes to encode all the words in one language. Figure 5.4 shows how this encoding scheme produced numeric codes in the input and output layer of the QNN. All the errors of words in Hindi and Devanagari-Hindi, sentence and POS are evaluated and recorded. The part of speech distribution for Devanagari-Hindi sentences according to their number and percentage are shown in Table 7.3

**Table 7.3: POS Distribution of 2600 English sentences**

| **Parts of Speech** | **Number wise POS Distribution with Hindi** | **Percentage wise POS Distribution for Hindi (%)** |
|---|---|---|
| Question Word | 500 | 4.35 |
| Noun | 2600 | 22.61 |
| Helping Verb | 2300 | 20 |
| Negative Word | 900 | 7.83 |
| Verb | 1900 | 16.52 |
| Preposition | 300 | 2.61 |
| Article | 300 | 2.61 |
| Adjective | 700 | 6.09 |
| Post Noun | 1500 | 13.04 |
| Adverb | 500 | 4.35 |
| **Total** | **11500** | |

All the errors of POS for words in Hindi sentence are evaluated and recorded. The Figure 7.2 clearly shows that the proposed Hindi POS tagger correctly disambiguates and correctly identifies the parts of speech with higher accuracy. The accuracy based on the categories of parts of speech is shown in the Figure 7.2. Experiments show that during learning process with QNN based POS tagger for Hindi, there is decrease in indeterminacy of pattern recognition and increase in authenticity of pattern recognition of POS.

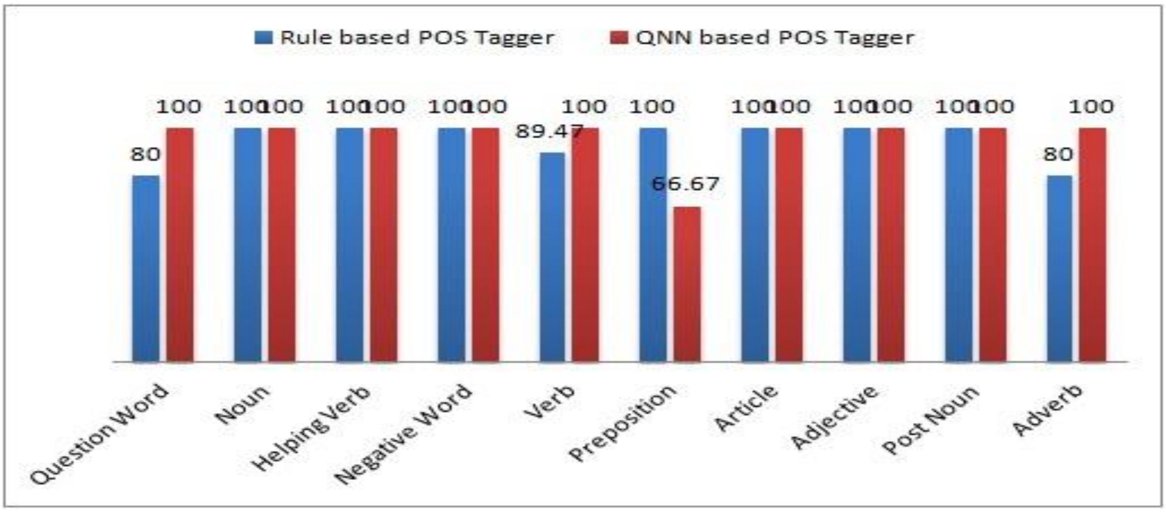


**Figure 7.2 Bar diagram for accuracy comparison between rule based Hindi POS tagging and QNN based tagging**

Hence, by using POS tagger with QNN, the proposed system has achieved a better POS tagging with higher accuracy in comparison to rule based Tagger.

### 7.2.3. Sentences tested with Google Translation, Microsoft's Bing Translation and proposed System.

The performance of proposed system is comparatively analyzed with Google translation (http://translate.google.com/) and Microsoft's Bing translation (http://www.bing.com/translator) by using various MT evaluation methods like BLEU, NIST, ROUGE-L, and METEOR. For evaluation purpose we translate the same set of input sentences by using the proposed system, Google translation and Bing translation, and then evaluate the output got from each of the system

#### 7.2.3.1. Validation of English to Hindi QNN based MT System on BLEU score

The Validation of English to Hindi QNN based MT System has been performed using BLEU score as above given Equations 7.3 and 7.4. Comparative Bar diagram between proposed system, Google, Bing and Anuvadaksh based on BLEU score is shown in Figure 7.3. The bar diagram clearly shows that the proposed system has remarkably high accuracy of 0.9809 in comparison of Anuvadksh 0.1128, Bing 0.1517 and Google 0.1686 accuracy on BLEU score.

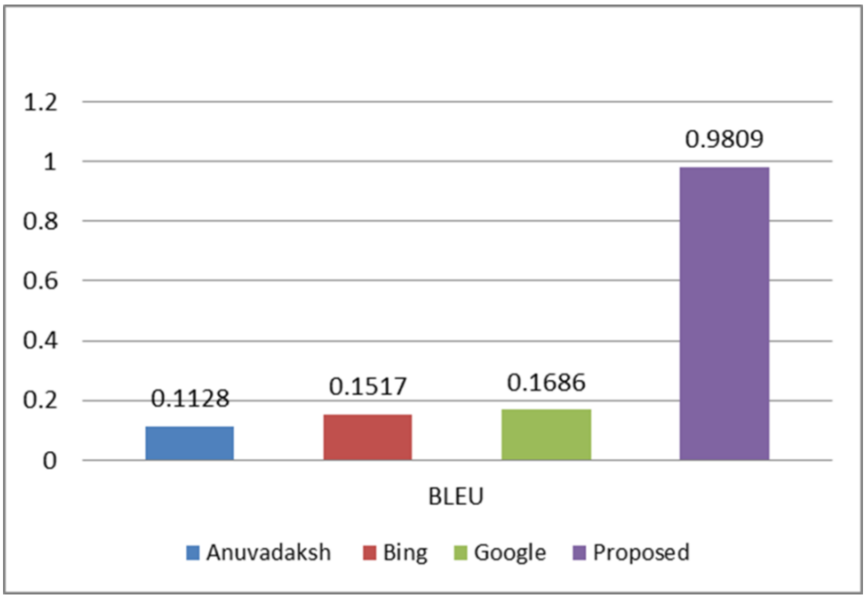


**Figure 7.3: Accuracy comparison among proposed system, Google, Bing and Anuvadaksh based on BLEU score**

### 7.2.3.2. Validation of English to Hindi QNN based MT System on NIST score

The Validation of English to Hindi QNN based MT System has been performed using NIST score as above given Equation 7.8.

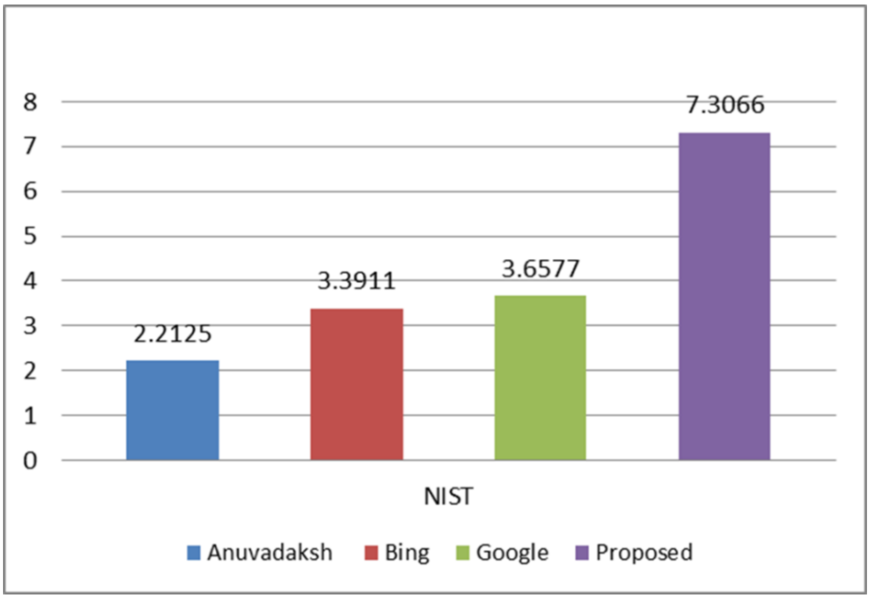


**Figure 7.4: Accuracy comparison among proposed system, Google, Bing and Anuvadaksh based on NIST score**

Comparative Bar diagram between proposed system, Google, Bing and Anuvadksh on

NIST score is shown in Figure 7.4. The bar diagram clearly shows that the proposed system has remarkably high accuracy of 7.3066 in comparison of Anuvadksh 2.2125,Bing 3.3911 and Google 3.6577 accuracy on NIST score.

#### 7.2.3.3. Validation of English to Hindi QNN based MT System on ROUGE-L score

The Validation of English to Hindi QNN based MT System has been performed using ROUGE-L score as above given Equation 7.12. Comparative Bar diagram between proposed system, Google, Bing and Anuvadksh on NIST score is shown in Figure 7.5.

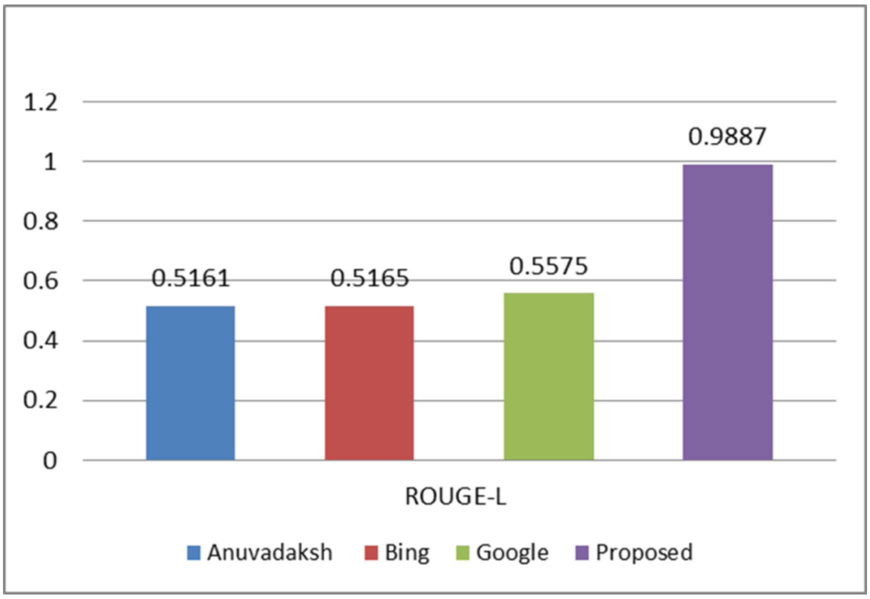


**Figure 7.5: Accuracy comparison among proposed system, Google, Bing and Anuvadaksh based on ROUGE-L score**

The bar diagram clearly shows that the proposed system has remarkably high accuracy of 0.9887 in comparison of Anuvadksh 0.5161, Bing 0.5165 and Google 0.5575 accuracy on ROUGE-L score.

#### 7.2.3.4. Validation of English to Hindi QNN based MT System on METEOR score

The Validation of English to Hindi QNN based MT System has been performed using METEOR score as above given Equation 7.17. Comparative Bar diagram between proposed system, Google, Bing and Anuvadksh on NIST score is shown in Figure 7.6.

The bar diagram clearly shows that the proposed system has remarkably high accuracy 0.7367 on METEOR score. Anuvadksh has shown accuracy 0.2289 on METEOR score, Bing 0.2399 and Google 0.2523 accuracy on METEOR score.

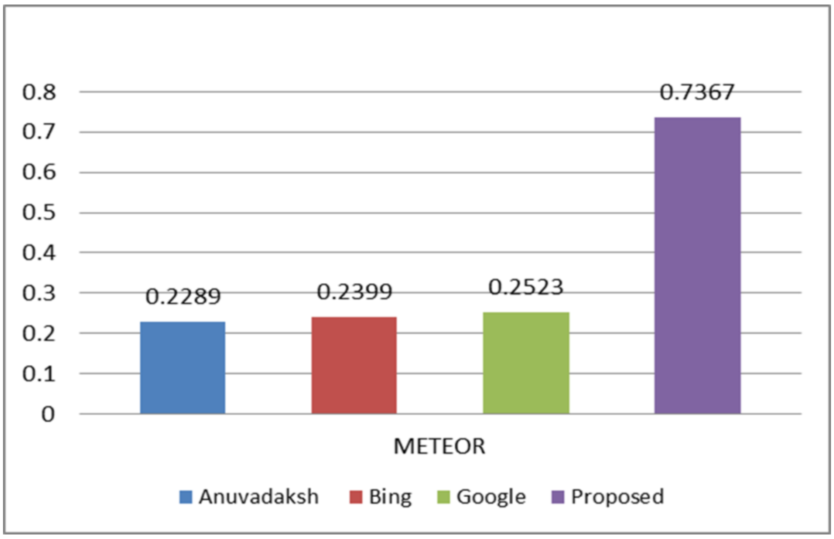


**Figure 7.6: Accuracy comparison among proposed system, Google, Bing and Anuvadaksh based on METEOR score**

### 7.2.3.5. Validation of Hindi to English QNN based MT System on BLEU score

The Validation of Hindi to English QNN based MT System has been performed using BLEU score as above given Equations 7.3 and 7.4.

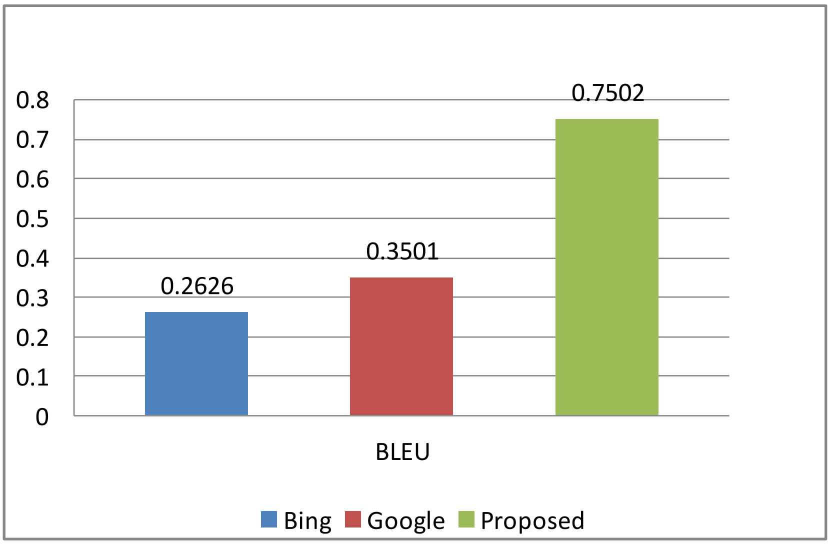


**Figure 7.7: Accuracy comparison among proposed system, Google and Bing based on BLEU score**

Comparative Bar diagram between proposed system, Google and Bing based on BLEU score is shown in Figure 7.7. The bar diagram clearly shows that the proposed system has remarkably high accuracy of 0.7502 in comparison of Bing 0.2626, and Google 0.3501 accuracy on BLEU score.

#### 7.2.3.6. Validation of Hindi to English QNN based MT System on NIST score

The Validation of Hindi to English QNN based MT System has been performed using NIST score as above given Equation 7.8. Comparative Bar diagram between proposed system, Google, Bing and Anuvadksh on NIST score is shown in Figure 7.8. The bar diagram clearly shows that the proposed system has remarkably high accuracy of 6.5773 in comparison of Bing 4.1744, and Google 4.955 accuracy on NIST score.

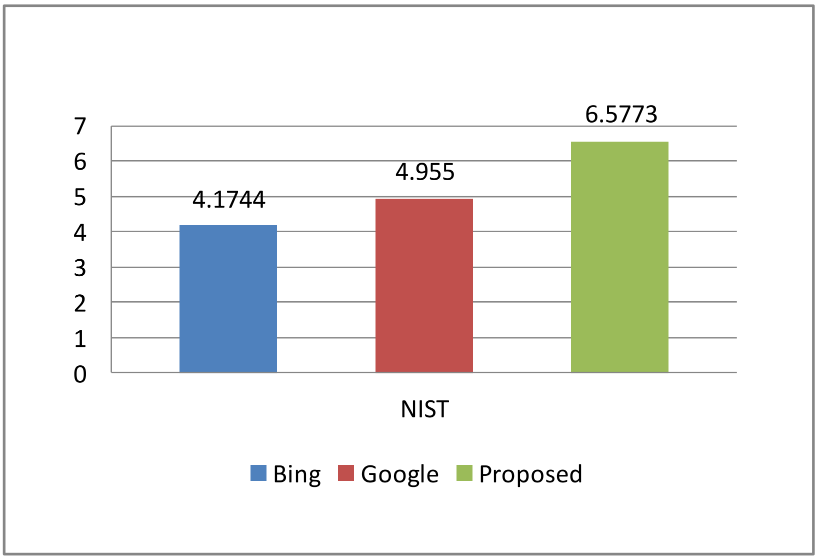


**Figure 7.8: Accuracy comparison among proposed system, Google and Bing based on NIST score**

#### 7.2.3.7. Validation of Hindi to English QNN based MT System on ROUGE-L score

The Validation of Hindi to English QNN based MT System has been performed using ROUGE-L score as above given Equation 7.12. Comparative Bar diagram between proposed system, Google, Bing and Anuvadksh on NIST score is shown in Figure 7.9.

The bar diagram clearly shows that the proposed system has remarkably high accuracy of 0.9233 in comparison of Bing 0.6475, and Google 0.7189 accuracy on ROUGE-L score.

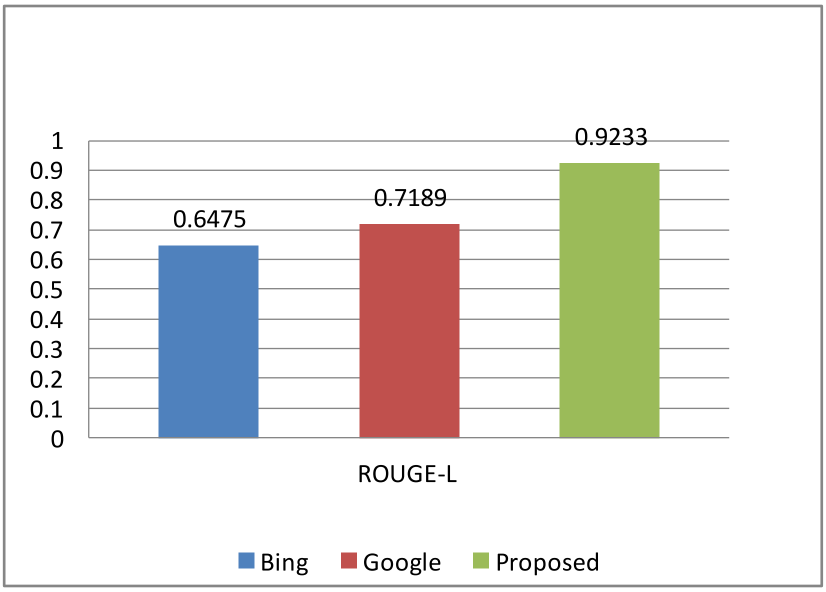


**Figure 7.9: Accuracy comparison among proposed system, Google, Bing and Anuvadaksh based on ROUGE-L score**

### 7.2.3.8. Validation of Hindi to English QNN based MT System on METEOR score

The Validation of Hindi to English QNN based MT System has been performed using METEOR score as above given Equation 7.17. Comparative Bar diagram between proposed system, Google, Bing and Anuvadksh on NIST score is shown in Figure 7.10.

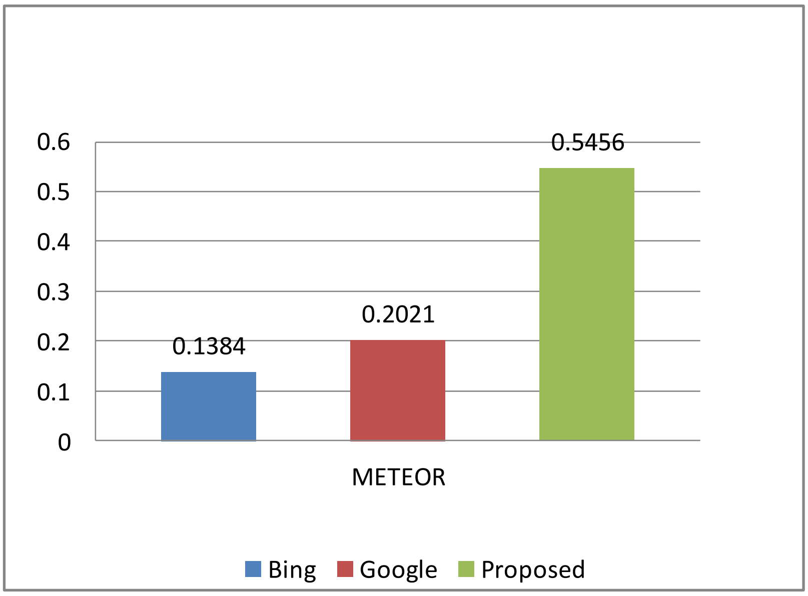


**Figure 7.10: Accuracy comparison among proposed system, Google and Bing based on METEOR score**

The bar diagram clearly shows that the proposed system has remarkably high accuracy of 0.5456 on METEOR score in comparison of Bing 0.1384, and Google 0.2021.

### 7.2.4. Human based Evaluation

On the basis of the tests performed on dataset, the accuracy percentage of ANN and QNN based MT systems have been evaluated on Human based Evaluation. Each translated sentence has been evelauated by Human manually.

#### 7.2.4.1. Validation of English to Hindi QNN based MT System

The Validation of English to Hindi QNN based MT System has been performed using Human based Evaluation. Comparative Bar diagram between proposed system and ANN based MT System based on Human based Evalution shown in Figure 7.11.

The overall accuracy with ANN is 86.667% and with QNN is 98.261% which is much higher than the equivalent ANN based MT system used by Chandola (Chandola *et al*, 1994).

Experiments confirm that the accuracy 98.261 % of MT based on quantum neural network is better than other bilingual translation methods.

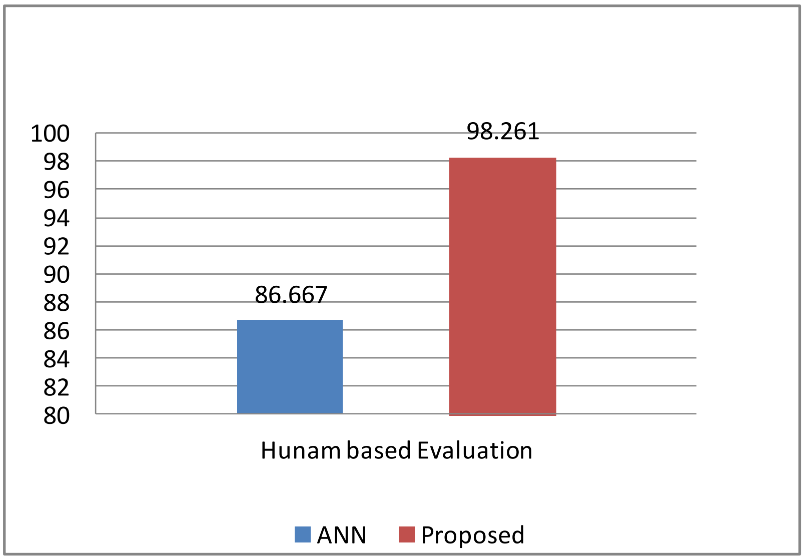


**Figure 7.11: Accuracy comparison among proposed system and ANN based MT system on Human based Evaluation**

#### 7.2.4.2. Validation of Hindi to English QNN based MT System

The Validation of Hindi to English QNN based MT System has been performed using Human based Evaluation. Comparative Bar diagram between proposed system and ANN based MT System based on Human based Evalution shown in Figure 7.12.

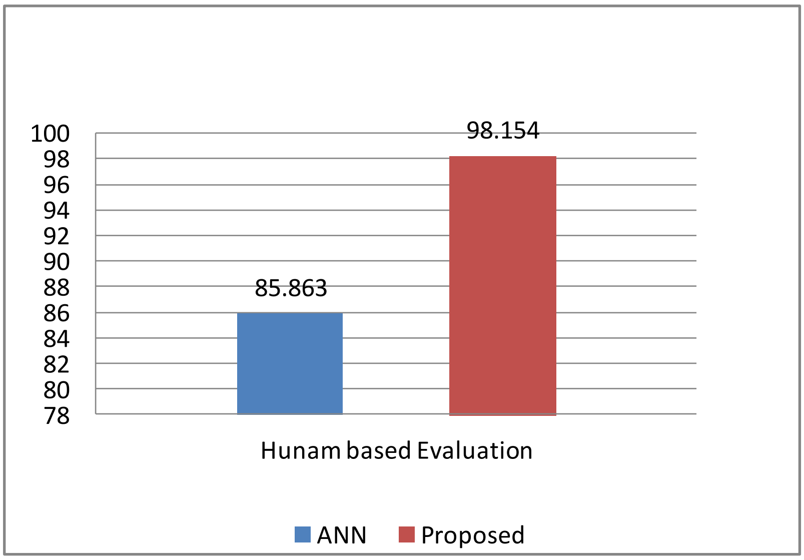


**Figure 7.12: Accuracy comparison among proposed system and ANN based MT system on Human based Evaluation**

The overall accuracy with ANN is 85.863% and with QNN is 98.154% which is much higher than the equivalent ANN based MT system.Experiments confirm that the accuracy achieved is 98.154 % of machine translation based on quantum neural network, which is better than other bilingual translation methods.

## 7.3. Conclusion

In this chapter, we have validated the QNN based MT System for Hindi to English, English to Hindi and for QNN based POS tagger for Hindi and English.

The performance of proposed system has been evaluated together with other existing systems like Bing translation, Google translation, neural network based MT and Anuvadksh. The proposed system has ben compared with existing system for validation on the score of NIST, BLEU, ROUGE-L and METEOR scores.

The accuracy of proposed quantum neural network based machine translation system for Devanagari (Hindi) to English has been compared on different scores viz. BLEU, NIST, ROUGE-L, METEOR and human based evaluation, the accuracy are respectively 0.7502 on score of 1, 6.5773 on score of 10, 0.9233 on score of 1, 0.5460 on score of 1 and 98.154 %. In case of English to Hindi MT system, the accuracy achieved on BLEU, NIST, ROUGE-L, METEOR and human based evaluation respectively are 0.9809 on score of 1, 7.3066 on score of 10, 0.9887 on score of 1, 0.9655 on score of 1 and 98.261%. The accuracy of proposed system for both Hindi to English and English to Devanagari (Hindi) are found to be significantly higher in comparison with the existing English to Devanagari (Hindi) and Devanagari (Hindi) to English MT system like Google and Bing, ANN based MT system and Anuvadaksh.

# CHAPTER 8

# CONCLUSIONS AND FUTURE SCOPE

## Chapter 8

# Conclusion and Future Scope

### 8.1. Conclusion

In this thesis, quantum neural network approach has been used for the problem of machine translation for English to Devanagari (Hindi) and vice versa.

The architecture of the proposed model has been discussed in detail. The proposed algorithm for handling the complex sentences has also been discussed. Moreover, the method used to implement the machine translation by using the QNN is also included. The coding mechanism is used to decode and encode the sentences into numeric code to pass these to QNN. This chapter also covers the QNN based pats of speech tagging in detail. The experiments have been performed for identifying the optimal learning rules, used in QNN based Hindi, English POS tagger and English to Hindi, Hindi to English Machine Translation. The optimal learning rule has been achieved by optimizing the value of Quantum Interval **($\theta^r$)** *viz*. Equation 4.6. With the different values of Quantum Interval **($\theta^r$),** the same experiment is done several times to identify the optimal value of Quantum Interval **($\theta^r$)**.

The results of different experiments have been performed with QNN and have also been compared between traditional ANN and quantum neural networks.

We have also validated the QNN based MT System for Hindi to English, English to Hindi and for QNN based POS tagger for Hindi and English.

The performance of proposed system has been evaluated together with other existing systems like Bing translation, Google translation, neural network based MT and Anuvadksh. The proposed system has been compared with existing system for validation on the score of NIST (National Institute of Standard and Technology), BLEU (Bilingual Evaluation Understudy), ROUGE-L (Recall-Oriented Understudy for Gisting Evaluation-Longest Common Subsequence) and METEOR (Metric for Evaluation of Translation with Explicit Ordering) scores.

The accuracy of proposed quantum neural network based machine translation system for Devanagari (Hindi) to English has been compared on different scores viz. BLEU, NIST, ROUGE-L, METEOR and human based evaluation, the accuracy are respectively 0.7502 on score of 1, 6.5773 on score of 10, 0.9233 on score of 1, 0.5460 on score of 1and 98.154 %.. In case of English to Hindi MT system, the accuracy achieved on BLEU, NIST, ROUGE-L, METEOR and human based evaluation respectively are 0.9809 on score of 1, 7.3066 on score of 10, 0.9887 on score of 1, 0.9655 on score of 1 and 98.261%. Similarly, accuracy of proposed system for both Hindi to English and English to Devanagari (Hindi) are found to be significantly higher in comparison with the existing English to Devanagari (Hindi) and Devanagari (Hindi) to English MT system like Google and Bing, ANN based MT system and Anuvadaksh.

The accuracy of this system has been improved significantly by incorporating QNN. Its performance is also compared with ANN based MT System. It is also shown that QNN based machine translator performs better than the ANN based MT Systems and it requires less training time than the neural network based MT systems. As such, this investigation shows that QNN based MT approach may be better than other existing approaches for English to Devanagari (Hindi) and vice versa.

## 8.2. Future Scope

Although our system is showing higher accuracy using the Quantum Neural Network based MT approach, although there is some possibility for enhancement.

The future directions are given below:

**Rich Lexicon/ Dictionary:**

The general and most appropriate method for enhancing the data driven approach, similar to the proposed work, is to use huge amount of data as knowledge base. The database containing bilingual dictionary having the complete set of POS information of any word like proper noun, surnames, titles, etc.

**Rich collection of bilingual translated sentences:**

Here hybrid approach of Quantum Neural Network based machine translation has been used to improve the accuracy. The proposed approach requires the high quality bilingual translated sentences of Hindi and its English translated Sentences, for accurate machine translation.

**Upgraded Models:**

Even though using huge data, enhanced models may perform better translation quality. Expansion to the parser for handling more grammatical structures like phrases, idioms and proverbs etc. would also be other direction of further work.

**Better Evaluation Metrics:**

Many Automatic evaluation methods are available viz. BLEU, NIST, ROUGE-L, METEOR etc, but still there is a lot of possibility to enhance the Automatic evaluation. Throughout the development and tuning phase, the quality of the machine translation system is evaluated many times. The parameters of the machine translation system are tuned to get the high score using any automatic evaluation metric. All the existing MT evaluation methods compare the n-gram MT translated sentence, word by word with the reference translation. Here the reference translation generally done by human. The outcomes of all the MT evaluation methods are regardless to the context of the sentence, so there is a need of enhanced metrics for a rapid development cycle.

**Extending to other Languages:**

As the performance of this proposed Quantum Neural Network based Machine translation System for Hindi to English and English to Hindi Translation are significantly more accurate than other existing systems. So it is worth mentioning that QNN based MT may very well be extended to other language translation problems.

**Extending to Other MT application:**

The outcome/methodology presented in this thesis may be a bench mark foundation for various other applications of machine translation. This system may be incorporated with other systems to deal with more complex tasks, like voice assisted lingual interpreter *etc.*

**Bibliography:**


[1]. Agarwal, H. and Mani A., (2006),” Part of Speech Tagging and Chunking with Conditional Random Fields” in the proceedings of NLPAI Contest.

[2]. Ahmed, Bapi Raju S., Chandrasekhar, P.,V.,S., Krishna Prasad, M., (2002), “Application of Multilayer Perceptron Network for Tagging Parts-of-Speech”, Proceedings of the Language Engineering Conference (LEC'02), pp. 57-63, 2002.

[3]. Ananthakrishnan, R., Kavitha, M. and Jayprasad J. H., (2006) “MaTra: a practical approach to fully-automatic indicative English-Hindi machine translation,” in Proceedings of the Symposium on Modeling and Shallow Parsing of Indian Languages (MSPIL '06).

[4]. Ananthakrishnan, S., Prasad, R., Stallard, D., and Natarajan, P., (2013), “Batch-mode semi-supervised active learning for statistical machine translation,” Computer Speech & Language, vol. 27, pp. 397–406.

[5]. Andres-Ferrer, J., Ortiz-Martinez, D., García-Varea, I. and Casacuberta, F., (2008), “On the use of different loss functions in statistical pattern recognition applied to machine translation”, Pattern Recognition Letters, Vol. 29(8), pp. 1072–1081.

[6]. Aronoff, M., Fudeman, K., (2011)"Book: What is Morphology?” John Wiley & Sons, Language Arts & Disciplines.

[7]. Balahur, A. and Turchi, M., (2014), "Comparative experiments using supervised learning and machine translation for multilingual sentiment analysis”, Computer Speech and Language, Vol. 28, pp.56–75.

[8]. Banerjee S. and Lavie, A., (2005), “METEOR: An automatic metric for MT Evaluation with improved correlation with human judgments”, in Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization. Ann Arbor, Michigan, US, pp. 65-72.

[9]. Behrman, Nash, L. R. and Steck, J. E., (2000), "Simulations of quantum neural networks", International Journal Information Sciences, pp. 257-269.

[10]. Bender, E., (1961), "HINDI Grammar and Reader", University of Pennsylvania Press, University of Pennsylvania South Asia Regional Studies, Philadelphia, Pennsylvania.

[11]. Bennett, W. S., (1990), "How much semantics is necessary for MT systems?", in Proceedings of the Third International Conference on Theoretical and Methodological Issues in Machine Translation of Natural Languages, Linguistics Research Center, The University of Texas, Austin, Vol. TX, pp. 261-269.

[12]. Brants, T. , Popat, A.C. , Xu, P. , Och, F.J. and Dean, J.,(2007),"Large language models in machine translation," in Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Prague, Czech Republic , pp. 858–867.

[13]. Brill, E., (1992), "A simple rule-based Parts of Speech tagger", in Proceedings of ANLP-92, 3rd Conference on Applied Natural Language Processing, Trento, pp. 152-155.

[14]. Brill, E., (1994), "Some advances in transformation-based Parts of Speech tagging", in AAAI '94: Proceedings of the twelfth national conference on Artificial Intelligence Vol. 1, American Association for Artificial Intelligence, Menlo Park, CA, USA, pp. 722-727.

[15]. Brill, E., (1995), "Transformation-Based Error Driven Learning and Natural Language Processing: A Case Study in Parts of Speech Tagging", Computational Linguistics Vol. 21(94), pp. 543-566.

[16]. Brown, P. F., (1990), "A statistical approach to Machine Translation", Journal of Computational Linguistics, Vol. 16(2), pp. 79-85.

[17]. Brown, P. F., Lai, J. C. and Mercer, R. L., (1991), "Aligning sentences in parallel corpora", in proceedings of 29th Annual Meeting of Association for Computational Linguistic, Berkeley, pp. 169-176.

[18]. Brown, P. F., Pietra, S. A. D., Pietra, V. J. D., Lafferty, J. D. and Mercer, R. L., (1992), "Analysis, statistical transfer, and synthesis in machine translation", in Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation of Natural Languages, Montreal, Canada, pp. 83-100.

[19]. Brown, P. F., Pietra, S. A., Pietra, V. J. D., Pietra, D. and Mercer, R. L., (1993), "The mathematics of statistical Machine Translation: parameter estimation", Computational Linguistics, Vol. 19(2), pp. 263-311.

[20]. Brown, R. D., (1996), "Example-Based Machine Translation in the pangloss system", in Proceedings of the 16th International Conference on Computational Linguistics (COLING-96), Copenhagen, Denmark, pp. 169-174.

[21]. Brown, R. D., (1999), "Adding linguistic knowledge to a lexical Example-Based Translation System", in Proceedings of the Eighth International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-99), Chester, UK, pp. 22-32.

[22]. Brown, R. D., (2001), "Transfer-rule induction for Example-Based Translation", in Proceedings of the MT Summit 8 Workshop on Example-Based Machine Translation, Santiago de Compostela, Spain, pp. 1-11.

[23]. Carl, M. and Hansen, S., (1999), "Linking translation memories with Example-Based Machine Translation", in Proceedings of Machine Translation Summit 8, Singapore, pp. 617-624.

[24]. Carl, M. and Way, A., (2003), "Advances in Example-Based Machine Translation Series: Text, Speech and Language Technology", Vol. 21, Kluwer Academic Publishers, Netherlands.

[25]. Chakraverty, S., Gupta, P. and Sharma, S., (2010), “Neural network-based simulation for response identification of two-storey shear building subject to earthquake motion”, Journal of Neural Computing and Applications, Vol. 3(19), pp. 367-375

[26]. Chandola and Mahalanobis, A., (1994), “Ordered rules for full sentence translation: a neural network realization and a case study for Hindi and English”, Pattern Recognition, Vol. 27(4), pp. 515–521.

[27]. Cicekli, I., and Guvenir, H. A., (1996), "Learning Translation Rules from A Bilingual Corpus", in Proceedings of the Second International Conference on New Methods in Language Processing (NeMLaP-2), Kemal Oflazer and Harold Somers (Eds.), Ankara, Turkey, pp. 90-97.

[28]. Cicekli, I., and Guvenir, H. A., (2001), "Learning Translation Templates from Bilingual Translation Examples", Applied Intelligence. Vol.15 (1), pp. 57 – 76.

[29]. Costa-jussa, M. and Fonollosa, J., (2009),"An N-gram-based reordering model", Journal of Computer Speech and Language, Vol.23 , pp. 362–375.

[30]. Cutting, Kupiec, J., Pedersen, J. and Sibun, P., (1992), “A practical part- of-speech tagger”, in Proceedings of the Third Conference on Applied Natural Language Processing, Italy, pp. 133-140.

[31]. Daelemans, W., Zavrel, J., Berck, P. and Gillis, S., (1996), “MBT: A memory-based part of speech tagger-generator”, in Proceedings of the Fourth Workshop on Very Large Corpora, E. Ejerhed and I. Dagan (eds.), Copenhagen, Denmark, pp. 14- 27.

[32]. Dalal, K. Nagaraj, U. Sawant and S. Shelke, (2006), “Hindi Part-of-Speech Tagging and Chunking: A Maximum Entropy Approach”, In Proceeding of the NLPAI Machine Learning Competition.

[33]. Daqi, Z. and Rushi, W., (2007), “A multi-layer quantum neural networks recognition system for handwritten digital recognition,” in Proceedings of the

3rd International Conference on Natural Computation, Hainan, China, pp. 718–722.

[34]. Dave, S. and Bhattacharyya, P., (2001a), “Knowledge extraction from Hindi text”, Journal of Institution of Electronic and Telecommunication Engineers, Vol. 18(4), pp. 1-12.

[35]. Dave, S., Parikh, J. and Bhattacharyya, P., (2001b), "Interlingua based English Hindi machine translation and language divergence", Journal of Machine Translation, Vol. 16(4), pp.251-304.

[36]. Deng, Y., and Byrne, W., (2008)"HMM Word and Phrase Alignment for Statistical Machine Translation", IEEE Transactions On Audio, Speech, And Language Processing, Vol. 16(3), pp. 494-507.

[37]. Doddington,G., (2002), “Automatic evaluation of machine translation quality using n-gram co-occurrence statistics,” in Proceedings of the 2nd International Conference on Human Language Technology Research.

[38]. Dorr, B. J., (1993), “Machine Translation: A View from the Lexicon”, MIT Press, Cambridge, MA.

[39]. Dorr, B. J., Pearl, L., Hwa, R. and Habash, N. Y. A., (2002), “DUSTer: A method for unraveling cross-language divergences for statistical word level alignment”, in Proceedings of the Fifth Conference of the Association for Machine Translation in the Americas, Tiburon, CA.

[40]. Dutta, K., Prakash, N. and Kaushik, S., (2010),” Probabilistic neural network approach to the classification of demonstrative pronouns for indirect anaphora in Hindi”, Expert Systems with Applications, Vol. 37, (8), pp. 5607–5613.

[41]. Dwivedi, S. K. and Sukhadeve, P., (2010), “Machine Translation System in Indian perspectives”, Journal of Computer Science, Vol. 6(10), pp. 1111-1116.

[42]. Fung, P. and McKeown, K., (1996), “A technical word-and term-translation aid using noisy parallel corpora across language groups”, The Machine

Translation Journal, Special Issue on New Tools for Human Translators, pp. 53-87.

[43]. Gale, W. A. and Church, K. W., (1991), “Identifying word correspondences in parallel texts”, in Proceedings of the Fourth DARPA Workshop on Speech and Natural Language, Morgan Kaufmann Publishers, Inc., pp. 152-157.

[44]. Gale, W. A. and Church, K. W., Yarowsky, D., (1992a),"One sense per discourse", Proceedings of the workshop on Speech and Natural Language, Association for Computational Linguistics, pp.233-237.

[45]. Gale, W. A., Church, K. W. and Yarowsky, D., (1992b), " A method for disambiguating word senses in a large corpus”, Journal of Computers and the Humanities, Kluwer Academic Publishers, vol. 26, pp. 415-439.

[46]. Gale, W. and Church, K., (1993), “A program for aligning sentences in bilingual corpora”, Computational Linguistics, vol. 19(1), pp. 75-102.

[47]. Garje, G. V., Kharate, G. K., (2013) “Survey Of Machine Translation Systems In India”, International Journal on Natural Language Computing ,Vol. 2(4), pp.47-67.

[48]. George D., (2002), "Automatic Evaluation of Machine Translation Quality Using n-Gram Co-occurrence Statistics”, in Proceedings of the Second International Conference on Human Language Technology Research (HLT).

[49]. Germann, U., (2001), “Building a Statistical Machine Translation system from scratch: How much bang for the buck can we expect?”, ACL 2001 Workshop on Data-Driven Machine Translation, Toulouse, France.

[50]. Germann, U., Jahr, M., Knight, K., Marcu, D. and Yamada, K., (2004) ,"Fast and optimal decoding for machine translation" Journal of Artificial Intelligence, Vol.154, pp. 127–143.

[51]. Gorecki, A., Bala, P. and Lesyng, B., (2006) “Parallelization of the quantum dynamics code for cluster architecture and its applications to the Gross–Pitaevskii equation” Vol. 106(3), pp 664–669.

[52]. Grishman, R. and Kosaka, M., (1992), “Combining rationalist and empiricist approaches to Machine Translation”, in Proceedings of the Fourth International Conference on Theoretical and Methodlogical Issues in Machine Translation of Natural Languages, Montreal, Canada, pp. 263-274.

[53]. Guvenir, H. A. and Cicekli, I., (1998), “Learning translation templates from examples”, Journal of Information System Vol. 23, pp. 353-363.

[54]. Habash, N. and Dorr, B. J., (2002), “Handling translation divergences: Combining statistical and symbolic techniques in generation-heavy Machine Translation”, in Proceedings of the Fifth Conference of the Association for Machine Translation in the Americas, Tiburon, CA.

[55]. Hutchins, J., (2003), “The Oxford Handbook of Computational Linguistics”, Oxford University Press, chapter Machine translation: general overview, pp. 501-511.

[56]. Indurkhya, N., Damerau, F. J.,(2010)"Handbook Of Natural Language Processing Second Edition" Chapman & Hall/Crc Taylor & Francis Group 6000 Broken Sound Parkway Nw, Suite 300 Boca Raton, Fl 33487-2742.

[57]. Kachru, Y., (1980), “Aspects of Hindi Grammar”, Manohar Publications, New Delhi.

[58]. Khalilov and Fonollosa, J. A. R., (2011), “Syntax-based reordering for statistical machine translation”, Computer Speech & Language, Vol. 25(4), pp. 761–788.

[59]. Khalilov M., Fonollosa J.A.R. and Dras M., (2009a), “A new subtree-transfer approach to syntax-based reordering for statistical machine translation”. Proceedings of the 13th Annual Conference of the European Association for Machine Translation (EAMT'09), pp. 197-204, Barcelona, Spain.

[60]. Khalilov M., Fonollosa J.A.R. and Dras M., (2009b), “Coupling hierarchical word reordering and decoding in phrase-based statistical machine translation”, Proceedings of The Third Workshop on Syntax and Structure in Statistical Translation (SSST-3) at North American Chapter of the

Association for Computational Linguistics - Human Language Technologies conference (NAACL-HLT'09), pp. 78-86, Boulder, Colorado (USA).

[61]. Kit, C., Pan, H. and Webster, J., (2002), "Example-Based Machine Translation: A New Paradigm", Translation and Information Technology, Chinese UHK Press, pp. 57-78.

[62]. Koehn, P., Och, F. J., and Marcu, D., (2003), "Statistical phrase-based translation", in proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, Morristown, NJ, USA, pp. 48–54.

[63]. Koehn, P., (2009). Statistical Machine Translation. Cambridge University Press.

[64]. Komeili, Z., Hendavalan, J. and Rahimi, A., (2011), "An Investigation of the Translation Problems Incurred by English to- Persian Machine Translations: Padideh", Pars, and Google Softwares", Journal of Procedia - Social and Behavioral Sciences, Vol. 28, pp.1079 – 1082.

[65]. Koul, O. N.,(2008), "Modern Hindi Grammar", Hyattsville, USA: Dunwoody Press.

[66]. Kretzschmar, R., Bueler, R., Karayiannis, N.B. and Eggimann, F., (2000), "Quantum neural networks versus conventional feed forward neural networks: An experimental study", IEEE Signal Processing Society Workshop Vol. 1, pp. 328 – 337.

[67]. Kumar S. G., (2005)," The EB-ANUBAD translator: A hybrid scheme", Journal of Zhejiang University Science, Vol. 6(10), pp.1047-1050

[68]. Labaka, G., Espana-Bonet, C., Màrquez, L., Sarasola, K., (2014) "A hybrid machine translation architecture guided by syntax" Machine Translation, Springer, Vol. 289(2), pp 91-125.

[69]. Lavie, A. and Denkowski, M. J., (2009),"The Meteor metric for automatic evaluation of machine translation" Journal of Machine Translation, Vol. 23(2-3), pp. 105-115.

[70]. Li, F., Zhao, S. and Baoyu, Z., (2002), "Quantum Neural Network in Speech Recognition", IEEE, 6th International Conference on Signal Processing, Beijing, China, Vol.2 pp. 1267-1270.

[71]. Li, P., Li, Y., Xiong, Q., Chai, Y. and Zhang, Y., (2014) "Application of a hybrid quantized Elman neural network in short-term load forecasting" Journal of Electrical Power and Energy Systems Vol. 55, pp. 749–759.

[72]. Li, P., Xiao, H., Shang, F., Tong, X., Li, M. and Cao, M., (2013)," A hybrid quantum-inspired neural networks with sequence inputs" Journal of Neurocomputing, Vol.117, pp.81–90.

[73]. Lin, C. Y., (2004) "ROUGEL: A package for automatic evaluation of summaries," in Proceedings of the Workshop on Text Summarization Branches Out, Barcelona, Spain, pp. 74-81.

[74]. Liu, C., Chena, C., Changc, C. and Shih, L., (2013),"Single-hidden-layer feed-forward quantum neural network based on Grover learning" Journal of Neural Networks, Vol. 45, pp.144–150.

[75]. Luitel, B. and Venayagamoorthy, G., (2010), "Quantum inspired PSO for the optimization of simultaneous recurrent neural networks as MIMO learning systems", Journal of Neural Networks, Vol. 23, pp. 583-586.

[76]. Manning, C. D. and Schutze, H., (2002), "Foundations of Statistical Natural Language Processing", MIT Press 2002.

[77]. Martinez, O. I., Sierra, J.L., Fernandez-Manjon, B. (2009), "Translating e-learning Flow-Oriented Activity Sequencing Descriptions into Rule-Based Designs," in Information Technology: New Generations, *ITNG '09.* pp.1108-1113.

[78]. Martínez, O. I., Varea, G.I., and Casacuberta, F., (2008a), “The scaling problem in the pattern recognition approach to machine translation”, Pattern Recognition Letters, Vol. 29(8), pp. 1145–1153.

[79]. Martínez, O. I., Varea, G.I., and Casacuberta, F., (2008b), “On the use of different loss functions in statistical pattern recognition applied to machine translation”, Pattern Recognition Letters, Vol. 29(8), pp. 1145–1153.

[80]. McEnery, A. M., Oakes, M. P. and Garside, R., (1994), “The use of approximate string matching techniques in the alignment of sentences in parallel corpora”, in A. Vella (ed.), in Proceedings of Machine Translation: 10 Years On, University of Cranfield, UK, pp 55-67.

[81]. Mima, H., Iida, H. and Furuse, O., (1998) “Simultaneous Interpretation Utilizing Example-based Incremental Transfer”, in Coling-ACL, pp. 855–861.

[82]. Mitsunaga, Shigeo, S., and Koji, N., (2006), “A Study on Learning with a Quantum Neural Network”, IEEE International Joint Conference on Neural Networks, Vancouver, BC, Canada, pp. 203-206.

[83]. Morel. B., (2000),"Biologically plausible learning rules for neural networks and quantum computing" Journal of Neurocomputing, Vol. 32, pp. 921-926.

[84]. Mridha, M. F., Hossain, M. Z. and Noor, S. A., (2010), " Development of Morphological Rules for Bangla Words for Universal Networking Language”, International Journal of Computer Science and Network Security, Vol. 10, pp. 235-241. 2.

[85]. Nagao, M., (1984), “A Framework of a Mechanical Translation between Japanese and English by Analogy principle”, Artificial and Human Intelligence, Chapter 11.

[86]. Narayanan, A. and Menneer, T.,(2000), "Quantum artificial neural network architectures and components", Journal of Information Sciences, Vol. 128, pp.231-255

[87]. Nieben, S., Ney, H., (2001), “Toward hierarchical models for statistical machine translation of inflected languages”, in Workshop on Data-Driven Machine Translation at 39th Annual Meeting of the Association of Computational Linguistics (ACL), pp. 47-54.

[88]. Nieben, S., Ney, H., (2004), “Statistical machine translation with scarce resources using morpho-syntactic information”, Computational Linguistics, vol. 30(2), pp. 181-204.

[89]. Nieben, S., Och, F. J., Leusch, G. and Ney, H., (2000), “An evaluation tool for machine translation: Fast evaluation for machine translation research”, in Proceedings of the Second International Conference on Language Resources and Evaluation, Athens, Greece, pp. 39-45.

[90]. Nieben, S., Vogel, S., Ney, H., Tillman, C., (1998), “A DP based search algorithm for statistical machine translation”, in Proceedings of 36th Annual Meeting of the Association of Computational Linguistics (ACL).

[91]. Och, F., J., (2006), "Challenges in Machine Translation", Chinese Spoken Language Processing of the series Lecture Notes in Computer Science, Vol. 4274, pp 15-15.

[92]. Och, F., J., Ney, H., (2004), “The alignment template approach to statistical machine translation”, in Computational Linguistics MIT Press, vol. 30(4), pp. 417-449.

[93]. Och, F.J. and Ney, H., (2001), “Statistical Multi-Source Translation”. In: "MT Summit 2001", Santiago de Compostela, Spain, pp. 253-258.

[94]. Och, F.J. and Ney, H., (2002), “Discriminative training and maximum entropy models for statistical machine translation”, in Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, pp. 295-302. 1.

[95]. Okhovvat, M. and Bidgoli, B. M., (2011), “A hidden Markov model for Persian Parts of Speech tagging”, Procedia Computer Science, Vol. 3, pp. 977-981.

[96]. Okpor, M. D.,(2014)“Machine Translation Approaches: Issues and Challenges” IJCSI International Journal of Computer Science Issues, Vol. 11(5),pp 159-165

[97]. Ortiz-Martınez, D., Garcia-Varea, I. and Casacuberta, F., (2008b), “On the use of different loss functions in statistical pattern recognition applied to machine translation”, Pattern Recognition Letters, Vol. 29(8), pp. 1145–1153.

[98]. Ortiz-Martınez, D., Garcia-Varea,I. and Casacuberta, F.,(2008a),"The scaling problem in the pattern recognition approach to machine translation", Journal of Pattern Recognition Letters, Vol. 29, pp.1145-1153.

[99]. Panda, M. K., Pillai, G. N. and Kumar, V. (2012), “An Interval Type-2 Fuzzy logic controller design for AVR system” Electric Power Components and Systems, Vol. 40(2). pp. 219-235.

[100]. Panigrahi, S.K., Chakraverty, S. and Mishra, B.K. (2009),"Vibration based damage detection in a uniform strength beam using genetic algorithm"Meccanica, Vol. 44, pp .697–710.

[101]. Papineni, K., Roukos, S., Ward ,T. and Zhu, W. J., (2002), “BLEU: A method for automatic evaluation of machine translation”, in Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, Pa, USA, pp. 311–318.

[102]. Purushothaman, G. and Karayiannis, N. B., (1994), “Fuzzy pattern classification using feed forward neural networks with multilevel hidden neurons”, IEEE International Conference on neural networks, Orlando, FL, USA, pp. 1577-1582.

[103]. Purushothaman, G. and Karayiannis, N. B., (1997), “Quantum Neural Networks (QNN's): inherently fuzzy feedforward neural networks”, IEEE Transactions on Neural Networks, Vol. 8(3), pp. 679–693.

[104]. Purushothaman, G. and Karayiannis, N. B., (1998), “Feed-forward neural architectures for membership estimation and fuzzy classification”,

International Journal of Smart Engineering System Design, Vol. 1, pp. 163-185.

[105]. Puscasu, G., (2004), “A multilingual method for clause splitting”, in Proceedings of CLUK 2004, University of Birmingham, Birmingham, UK.

[106]. Qing, Ma and Isahara, H., (1997), “Parts of Speech Tagging of Thai Corpus with the Logically Combined Neural Networks”, in Proceedings of the Natural Language Processing Pacific Rim Symposium, Linguistics and Knowledge Science Laboratory, Athens, Greece, pp. 537-540.

[107]. Raj, R.G. and Abdul-Kareem, S., (2009), “Information Dissemination and Storage for Tele-Text Based Conversational Systems' Learning”, Malaysian Journal of Computer Science, Vol. 22(2), pp. 138-159.

[108]. Raj, R.G. and Abdul-Kareem, S., (2011), “A Pattern Based Approach for the Derivation of Base Forms of Verbs from Participles and Tenses for Flexible NLP”, Malaysian Journal of Computer Science, Vol. 24(2), pp. 138-159.

[109]. Rajan, R., Sivan, R., Ravindran, R. and Soman, K. P., (2009), “Rule Based Machine Translation from English to Malayalam”, IEEE International Conference on Advances in Computing, Control, & Telecommunication Technologies, pp. 439 – 441.

[110]. Raman, S. and Reddy, N. R. K., (1997), “A transputer-based parallel machine translation system for Indian languages”, Microprocessors and Microsystems, Vol. 20(6), pp. 373–383.

[111]. Rao, D., Mohanraj, K., Hegde, J., Mehta, V. and Mahadane, P., (2000), “A practical framework for syntactic transfer of compound-complex sentences for English- Hindi Machine Translation”, in Proceedings of the Conference on Knowledge based computer systems, National Centre for Software Technology, Mumbai, pp. 343-354.

[112]. Reddy, A. and Rose, R. C., (2010) "Integration of Statistical Models for Dictation of Document Translations in a Machine-Aided Human Translation

Task", IEEE Transactions On Audio, Speech, And Language Processing, Vol. 18(8), pp. 2015-2027.

[113]. Roark, B., Saraclar, (2006) “Utterance classification with discriminative language modelling” Journal Of Speech communication, Vol. 40(3), pp 276-287

[114]. Roark, B., Saraclar, M. and Collins, M., (2004a), “Corrective language modeling for large vocabulary ASR with the perceptron algorithm", Acoustics, Speech, and Signal Processing, 2004. Proceedings.(ICASSP'04), Vol.1, pp. 749-752.

[115]. Roark, B., Saraclar, M. and Collins, M., (2007), "Discriminative n-gram language modeling", Computer Speech and Language, Vol. 21, pp. 373–392.

[116]. Roark, B., Saraclar, M. and Collins, M., Johnson, M., (2004b), “Discriminative language modeling with conditional random fields and the perceptron algorithm", Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, pp. 47-54.

[117]. Rodriguez. L., Garcia-Varea, I. and Gamez,J.,(2008),"On the application of different evolutionary algorithms to the alignment problem in statistical machine translation", Journal of Neurocomputing, Vol. 71, pp. 755–765.

[118]. Schmid, H., (1994a), “Part-of-speech tagging with neural networks”, in Proceedings of International Conference on Computational Linguistics-94, Kyoto, Japan, pp. 172–176.

[119]. Schmid, H., (1994b), “Probabilistic Part-of-Speech Tagging Using Decision Trees”, Proceedings of International Conference on New Methods in Language Processing, Manchester, UK.

[120]. Sejnowski, T. J. and Rosenberg, C. R., (1987)," Parallel Networks that Learn to Pronounce English Text", Complex Systems, Vol. 1, pp. 145-168.

[121]. Shahnawaz and Mishra, R. B., (2012), "A Neural Network based Approach for English to Hindi Machine Translation", International Journal of Computer Applications, Vol. 53(18), pp. 975 – 8887.

[122]. Shirai, S., Bond, F. and Takhashi, Y., (1997), "A Hybrid Rule and Example-Based Method for Machine Translation", in Proceedings of the 4th Natural Language Processing Pacific Rim Symposium: NLPRS-97, Phuket, Thailand, pp. 49-54.

[123]. Siddiqui, T., Tiwary, U. S., (2008)"Natural language processing and information retrieval", Oxford University Press, New Delhi.

[124]. Silvestre-Cerda, J., Andres-Ferrer, J. and Civera, J., (2012) "Explicit length modelling for statistical machine translation",Journal of Pattern Recognition, Vol. 45, pp. 3183-3192.

[125]. Singh, S. B., (2003), "English- Hindi Translation Grammar", first edition, Prabhat Prakashan, 4/19 Asaf Ali Road, New Delhi-110002.

[126]. Singh, S., Gupta, K., Shrivastava, M. and Bhattacharyya, P., (2006), "Morphological richness offsets resource demand – experiences in constructing a pos tagger for hindi", in Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions, Sydney, Australia, pp. 779–786.

[127]. Singh, V.P., Chakraverty, S., Sharma, R.K. and Sharma, G.K., (2009), "Modeling vibration frequencies of annular plates by regression based neural network", Applied Soft Computing, Vol.9, pp.439–447

[128]. Sinha, R. M. K. and Jain, A., (2003), "AnglaHindi: An English to Hindi Machine Translation system", in Proceedings of the MT Summit 9, Orleans, LA, pp. 23-27.

[129]. Sinha, R. M. K. and Thakur, A., (2005), "Machine translation of bi-lingual Hindi-English (Hinglish)", in Proceedings of the 10th Machine Translation Summit, Phuket, Thailand, pp. 149–156.

[130]. Sinha, R. M. K., (1984), "Computer processing of Indian languages and scripts -potentialities and problems". Journal of Institution of Electronic and Telecommunication Engineers, Vol. 30(6), pp. 133-149.

[131]. Sinha, R. M. K., (2004), “An engineering perspective of machine translation: anglabharti-II and anubharti-II architectures”, in Proceedings of the International Symposium on Machine Translation, NLP and Translation Support System, Tata McGraw-Hill, New Delhi, India, pp. 134–138.

[132]. Sofianopoulos, S. and Tambouratzis, G., (2010), "Multi-objective optimisation of real-valued parameters of a hybrid MT system using Genetic Algorithms", Pattern Recognition Letters, Vol.31 pp.1672–1682.

[133]. Somers, H., (1998), “Further experiments in bilingual text alignment”, International Journal of Corpus Linguistics, Vol. 3, pp. 115-150.

[134]. Somers, H., (1999), “Review article: Example-Based Machine Translation”, Journal of Machine Translation, Vol.14, pp. 113-158.

[135]. Somers, H., (2001), “EBMT seen as case-based reasoning”, MT Summit 8 Workshop on Example-Based Machine Translation, Santiago de Compostela, Spain, pp. 56-65.

[136]. Su, H., (2011),"Chaos quantum-behaved particle swarm optimization based neural networks for short-term load forecasting, Procedia Engineering, Vol.15, pp.199 – 203.

[137]. Sumita, E. and Iida, H., (1991), “Experiments and prospects of Example-Based Machine Translation”, in Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics, Berkeley, California, USA, pp. 85-192.

[138]. Takahashi, K., Kurokawa, M. and Hashimoto, M., (2014),"Multi-layer quantum neural network controller trained by real-coded genetical gorithm" Journal of Neurocomputing, Vol.134, pp. 159–164.

[139]. Theodoroua,Y., Drossos, C., and Alevizosb, P., (2007), "Correspondence analysis with fuzzy data: The fuzzy eigenvalue problem", Fuzzy Sets and Systems, Vol.158(7), pp.704–721

[140]. Tillmann, C., and Zhang, T.,(2008) “An Online Relevant Set Algorithm for Statistical Machine Translation” IEEE Transactions On Audio, Speech, and Language Processing, Vol. 16( 7), pp1274-1286.

[141]. Tufis and Mason, O., (1998), “Tagging Romanian Texts: a Case Study for QTAG, a Language Independent Probabilistic Tagger”, in Proceedings of the First International Conference on Language Resources & (LREC), Granada, Spain, pp. 589-596.

[142]. Udupa U. R., Faruquie. T. A., (2004)"An English-Hindi Statistical Machine Translation System" Natural Language Processing – IJCNLP Lecture Notes in Computer Science, Vol. 3248, pp 254-262.

[143]. Udupa, U. R., Maji, H. K., (2006) "Computational Complexity Of Statistical Machine Translation", Annual Meeting of The European Chapter of The Association of Computational Linguistics.

[144]. Veale, T. and Way, A., (1997), Gaijin, “A template-driven bootstrapping approach to Example-Based Machine Translation”, International Conference, Recent Advances in Natural Language Processing, Tzigov Chark, Bulgaria, pp. 239-244.

[145]. Vikram, S., (2013),"Morphology: Indian Languages and European Languages", International Journal of Scientific and Research Publications, Vol. 3(6).

[146]. Watanabe, H., (1992), “A similarity-driven transfer system”, in Proceedings of the 14th COLING, pp. 770-776.

[147]. Watanabe, H., Kurohashi, S. and Aramaki, E., (2000), “Finding structural correspondences from bilingual parsed corpus for Corpus-Based Translation”, in Proceedings of International Conference on Computational Linguistics - 2000, Saarbrucken, Germany.

[148]. Weaver, W., (1949), "Translation". Machine Translation of Languages. Cambridge, Massachusetts: MIT Press. pp. 15–23

[149]. Wren, P., Martin, H. and Rao, N., (1989), “High School English Grammar”, S. Chand & Co. Ltd., New Delhi.

[150]. Wu, D., (1995), “Large-scale automatic extraction of an English-Chinese translation lexicon”, Machine Translation, vol. 9(3-4), pp. 285-313.

[151]. Xiong, D., Zhang, M., and Li, H., (2011) “A Maximum-Entropy Segmentation Model for Statistical Machine Translation” IEEE transactions on audio, speech, and language processing, Vol. 19(8), pp. 2494-2505.

[152]. Zens, R., Och, F. J. and Ney, H., (2002), “Phrase-based statistical machine translation”, in KI ’02: Proceedings of the 25th Annual German Conference on AI, London, UK, Springer-Verlag, pp. 18–32.

[153]. Zhang, R. and Kikui, G., (2006), "Integration of speech recognition and machine translation: Speech recognition word lattice translation", Journal of Speech Communication, Vol. 48, pp.321–334.

[154]. Zhi-ying, L. and Yao-hong, J., (2012)"Passive sentence transformation in Chinese-English patent machine translation" The Journal of China Universities of Posts and Telecommunications, Vol. 19(2), pp.135–139.

[155]. Zhou, Gan, Q. and Krzyzak, A., (1999), “Recognition of handwritten numerals by quantum neural network with fuzzy features”, International Journal on Document Analysis and Recognition, Vol. 2(1), pp. 30-36. 1.

[156]. Zhu, D., Wu, R., (2007), “A Multi-layer Quantum Neural Networks Recognition System for Handwritten Digital Recognition”, Third International Conference on Natural Computation, Vol. 1, pp. 718-722.

[157]. Zyglarski, B. and Bała, P., (2011) “Neural Networks Aided Automatic Keywords Selection” Knowledge Discovery, Knowledge Engineering and Knowledge Management Communications in Computer and Information Science, Vol. 128, pp. 377-389.

# LIST OF PUBLICATIONS

**Following are the papers that have been published /communicated from the present research:**

**Published:**

**[1].** **Ravi Narayan,** V. P. Singh and S. Chakraverty, (2014), **"Quantum Neural Network Based Machine Translator for Hindi to English"**, The Scientific World Journal, Vol. 2014, Article ID 485737.

**[2].** **Ravi Narayan,** V. P. Singh and S. Chakraverty, (2014), **"Quantum Neural Network based Parts of Speech Tagger for Hindi"**, International Journal of Advancements in Technology, Vol 5 no 2.

**[3].** **Ravi Narayan,** S. Chakraverty and V. P. Singh, (2013),**"Machine Translation using Quantum Neural Network for Simple Sentences"**, International Journal of Information and Computation Technology, Vol. 3, no.7, pp.683 – 690.

**[4].** **Ravi Narayan,** S. Chakraverty and V. P. Singh, (2014), **"Neural Network based Parts of Speech Tagger for Hindi"**, Third International conference, Advances and control and Optimisation of dynamical systems, IIT Kanpur, , proceedings of IFAC-Elsevier, Vol 3, no.1, pp 519-524**.**

**[5].** **Ravi Narayan,** S. Chakraverty and V. P. Singh, (2015) "**Quantum Neural Network based Machine Translator for English to Hindi"**, International Journal of Applied Soft Computing, Elsevier**. (Accepted)**

**Communicated:**

**[6].** **Ravi Narayan,** V. P. Singh and S. Chakraverty, "**Quantum Neural Network based Parts of Speech Tagger for English"**, Malaysian Journal of Computer Science.

**[7].** **Ravi Narayan,** S. Chakraverty and V. P. Singh, "**Pattern Directed Hybrid machine translator based on Quantum Neural Network for English to Hindi",** Arabian Journal of Science and Engineering**.**