LANGUAGE IN INDIA

Strength for Today and Bright Hope for Tomorrow

Volume 10 : 10 October 2010

ISSN 1930-2940

Development of a Hindi to Punjabi Machine Translation System

A Doctoral Dissertation

Vishal Goyal, Ph.D.

DEVELOPMENT OF A HINDI TO PUNJABI

MACHINE TRANSLATION SYSTEM

A

THESIS

Presented to the Faculty of Physical Sciences of the

Punjabi University

in Fulfilment of the Requirements for the

Degree of

DOCTOR OF PHILOSOPHY

IN

COMPUTER SCIENCE

Vishal Goyal

Department of Computer Science
Punjabi University, Patiala
February, 2010

CERTIFICATE

This is to certify that this thesis “Development of a Hindi to Punjabi Machine Translation System” embodies the work carried out by Vishal Goyal himself under my supervision and that it is worthy of consideration for the award of the Ph.D. degree.

(Dr. Gurpreet Singh Lehal)
Professor,
Department of Computer Science,
Punjabi University, Patiala.
(Supervisor)

DECLARATION

I hereby affirm that the work presented in this thesis is exclusively my own and there are no collaborators. It does not contain any work for which a degree/diploma has been awarded by any other University/Institution. A part of this thesis has already been published in international & national journals and the proceedings of the IEEE International Conference.

(Vishal Goyal)

Countersigned

(Dr. Gurpreet Singh Lehal)
Professor,
Department of Computer Science,
Punjabi University, Patiala.

Acknowledgements

I would like to express my gratitude to my supervisor, Dr. Gurpreet Singh Lehal, Professor, Department of Computer Science, Punjabi University, Patiala. His advice, his continous interest, and his support made the thesis ultimately possible. Despite his busy schedule as the Director of the Advanced Centre for Technical Development of Punjabi Language and Culture he has been always available for discussion.

I would like to extend my thanks to all those who have helped in finishing my work on time. I will not go into listing all the names here, but I remember even the slightest instance of support provided to me. I am grateful to all the people at ACTDL LAB and my Computer Science Department, for helping me.

Last, but not the least, I thank all the members of my family for extending me much needed support and inspiration to finish this arduous task.

Vishal Goyal

Patiala, February 2010

Contents

Abstract

Machine Translation is a task of automatic translation a text from one natural language to another. Even after more than 60 years of research, Machine Translation is still an open problem. Work for the development of Machine Translation systems for Indian languages is still in infancy. This research work is an attempt to develop a Machine Translation system from Hindi to Punjabi language. A number of Machine Translation systems have already been developed though their accuracy needs to be improved. Machine Translation is not a trivial task by nature of translation process itself. But Machine Translation of closely related languages eases the task. We call a language pair to be closely related if the languages have the grammar that is close in structure, contain similar constructs having almost same semantics, and share a great deal of lexicon. By closely related languages, we also mean in□ectively and morphosyntactically similar languages. Some linguist define closeness between the languages on the basis of features viz. common root, similar alphabets, similar verb patterns, structural similarity, similar grammar, similar religio-cultural and demograpohic contexts and references, a similar clearly displayed ability to blend with foreign tongues . Generally, such languages have originated from the same source and spoken in the areas in close proximity. Hindi and Punjabi belong to same sub group of the Indo European family, thus are sibling languages. It has been analysed that Hindi and Punjabi languages share all features of closely related languages. For such closely related sibling languages, effective word for word translation can

be achieved (Hajic et al., 2000) [90]. Thus for our system, Direct Machine Translation approach which seems promising approach has been used.

The challenges in deleveloping Hindi to Punjabi Machine Translation system lie with major problems mainly related to the non-availability of lexical resources, spelling variations, word sense disambiguation, transliteration, named entity recognition and collocations.

This research work addresses the problems in the various stages of the development of a complete Hindi to Punjabi Machine Translation system and discusses potential solutions. The thesis has been divided into eight chapters.

The first chapter of the thesis introduces general concept of Machine Translation, various approaches to Machine Translation systems and key activities involved in Machine Translation. It also provides a formal description about the research question undertaken for this study. The objectives, need, and scope of the study have also been discussed. Then some of the key application areas of Machine Translation system are explored. Afterwards, the approach followed along with the reasons behind its selection to solve this research problem has been explained in brief. An overview of the design of the Machine Translation system undertaken to develop in this research work is provided later. The chapter concludes by presenting major contributions of this research work and an outline of the study.

Chapter 2 discusses the existing work in the field of Machine Translation in India and outside India. This chapter on literature survey forms the basis of our work on developing the Machine Translation system and later on helps us

in comparing our work with the existing state of the art in Machine Translation system.

Chapter 3 explains and compares Hindi and Punjabi languages with respect to orthography, grammar, and Machine Translation.

Chapter 4 and 5 provide the design and implementation details of various activities involved in the Machine Translation system. Chapter 4 describes the system architecture and preprocessing stage. The chapter starts with the choice of approach and discusses the motivation behind its selection. Then the required resources are discussed followed by description of system architecture. The details of preprocessing phase which involves text normalization, Identifying Collocations, Identifying Proper Nouns are discussed. Then tokenization process is explained. The details of the translation system involving the identifying titles, identifying surnames, lexicon lookup, word sense disambiguation module, transliteration module and post processing modules are discussed in Chapter 5.

Chapter 6 describes the post processing stage of the system. Chapter 7 provides the evaluation of the system and its results. Chapter 8 concludes this thesis by providing a summary of the research work undertaken, contributions of this research work, limitations, and some directions in which this work could be extended in the future. In appendix A, the interface designed for text translation, website translation and email translation has been discussed. Test data set for intelligibility test and accuracy test is available in Appendix B and C respectively. The system has been rigorously evaluated and its accuracy

has been found to be 94% on the basis of intelligibility test and 90.84% on the basis of accuracy test.

List of Tables

6.2 Percentage Contribution of Regular Expressions on the basis of Grammatical Error Categories

6.3 Percentage Contribution of various MT System modules during translation

7.1 Test data set for the evaluation of Hindi to Punjabi Machine Translation System

7.2 Score Sheet for Intelligibility Test

7.3 Score Sheet for Accuracy Test

7.4 Comparison of our System with other existing systems

List of Figures

A.7. Screenshot for Email Sending facility of the system

Chapter 1

Introduction

The goal of automatic translation (also called Machine Translation, or MT) is to translate text from one human language into other using computers. MT was one of the first envisioned applications of computers back in the 1950's. Even after more than 60 years of research, MT is still an open problem. Nowadays, the demand for MT is steadily growing. All over the world, documents have to be translated into all official languages. This multilingualism is considered a part of democracy. Also in the private sector, there is a large demand of MT: technical manuals have to be translated into several languages. An even large demand exists in the World Wide Web. Thus, MT can help to reduce the language barrier and enable easier communication. This research work is an attempt to develop a Machine Translation system from Hindi to Punjabi Language. A number of Machine Translation systems have already been developed though their accuracy needs to be improved. However, there is no Hindi to Punjabi Machine Translation system available at the present.

This chapter introduces general concept of Machine Translation, various approaches for Machine Translation systems and key activities involved in Machine Translation. It also provides a formal description about the research question undertaken for this study. The objectives, need, and scope of the

study have also been discussed. Then some of the key application areas of Machine Translation system are explored. Afterwards, the approach followed along with the reasons behind its selection to solve this research problem has been explained in brief. An overview of the design of the Machine Translation system undertaken to develop in this research work is provided later. The chapter concludes by presenting major contributions of this research work and an outline of the study. This work is based on the Devanagari and Gurmukhi scripts. Thus, the examples given in this thesis work are in Devanagari and Gurmukhi scripts along with their transliteration. For inline examples, transliteration will be provided in italics e.g. गहना *(gahnā*). The transliteration provided is based on transliteration software – the GTrans, which was developed in the Department of Computer Science, Punjabi University, Patiala, Punjab, India.

1.1 Machine Translation

You feed a story written in Hindi into a computer system and out comes its translation in Punjabi, Oriya, English, Tamil and other languages. It is inexpensive, immediate and simultaneous. The language barriers melt away. The richness of other literatures opens up to everyone. The world is intellectually and culturally united into one. This is the dream of people working in a fascinating area of research called Machine Translation. Thus Machine Translation system is software designed that essentially takes a text

in one language (called the source language), and translates it into another language (called the target language). The source and target languages are natural languages such as English and Hindi, as opposed to man-made languages such as C or SQL. Translation, in its full generality, is a difficult, fascinating, and intensively human endeavor, as rich as any other area of human creativity. Machine Translation is an important sub-discipline of the wider field of artificial intelligence (AI). AI (among other things) deals with getting machines to exhibit intelligent behaviour.

Though Machine Translation has been an interesting area of research since the invention of computers, in India it is relatively young. As a discipline it dates back to the early 1950. The earliest efforts in research date back to late 80s and early 90s. The complexity of the problem was originally underestimated, and some early successful demonstrations of experimental systems led to unrealistic expectations which were hard to fulfill. This led to some skepticism, and funding on MT work almost ceased. In the early eighties, the Japanese Fifth Generation Computing Project revived interest in this work and some of the prominent works in India are the projects at IIT Kanpur, University of Hyderabad, NCST Mumbai and CDAC Pune. The Technology Development in Indian Languages (TDIL), an initiative of the Department of IT, Ministry of Communications and Information Technology, Government of India, has played an instrumental role in funding these projects. Since the mid and late 90's, a few more projects have been initiated—at IIT Bombay, IIIT Hyderabad, AU-KBC Centre Chennai, Jadavpur

University Kolkata and Punjabi University Patiala. There are also a couple of efforts from the private sector - from Super Infosoft Pvt Ltd, and more recently, the IBM India Research Lab [1,2].

Machine Translation between closely related languages is easier than between language pairs that are not related with each other. Having many parts of their grammars and vocabularies in common reduces the amount of effort needed to develop a translation system between related languages. In this thesis, we will be discussing the Machine Translation system between closely related languages- Hindi and Punjabi.

1.1.1 Background:

Warren Weaver, a director of the Rockefeller Foundation, received much credit for bringing the concept of MT to the public when he published an influential paper on using computer for translation in 1949. The early 1950s were a period of intense research in MT in both the United States and Europe. 1952 saw the first conference on MT, but it was not until 1954 that a translation system was demonstrated in New York. The reaction of public to this MT system was negative because many people thought that perfect MT was close at hand and human translators would be out of their jobs. In 1959, IBM installed an MT system for the United States Air Force, followed by Georgetown University installing systems at Erratum and the United States Atomic Energy Agency. Despite some success of early MT systems, MT research funding was on the verge of serious reduction.

The growing dissatisfaction of research sponsors caused the United States National Academy of Sciences to set up the Automatic Language Processing Advisory Committee (ALPAC) in 1966. ALPAC, whose members were the major sponsors of current MT research projects, was to evaluate the effectiveness, costs, and potential future progress of MT. Their findings, known as the ALPAC Report, concluded that MT was not useful and sufficient goal. The research was rather unsatisfactory to justify further funding from the United States government. The effects of the report rippled to cause most private sponsors of MT projects in the United States to withdraw from future funding. ALPAC also suggested the complete discontinuation of MT research in the United States and the computer aids for translators should be developed instead. So, for several years, MT research was virtually at standstill. 1976 marked a positive turning point for MT research when the country of Canada made public their Mateo System, which translated weather forecasts. Later that year, the European Commission purchased SYSTRAN, a Russian-English system. MT interest and activity has increased ever since, and MT has been established as a legitimate field of research. In the 1980s, MT software for personal computers appeared; the 1990s showed MT implemented as an online service. The 2000s have shown even more research into MT and many new, efficient hybrid algorithms.

The advent of low-cost and more powerful computers towards the end of the 20th century brought MT to the masses, as did the availability of sites on the Internet. Much of the effort previously spent on MT research, however, has

shifted to the development of Computer-Assisted Translation (CAT) systems, such as translation memories, which are seen to be more successful and profitable [3-18,21,22,23].

1.1.2 Approaches used for Machine Translation [19-22]:

There are a number of approaches used for MT. But mainly three approaches are used. These are discussed below:

1. Rule-Based Approaches
2. Data-Driven Approaches
3. Hybrid Approaches

1.1.2.1. Rule-Based Approaches:

The current rule-based architecture of MT can be categorized into three areas:

1. Direct MT
2. Indirect MT
3. Interlingua MT

The Machine Translation has two generations to be considered during its development. The first generation Machine Translations are those which were done in 1960s and are called Direct Machine Translation. They used the direct approach of translation which was based on word-to-word and/or

phrase to phrase translations. Simple word-to-word translation cannot resolve the ambiguities arising in MT. A more thorough analysis of source language text is required to produce better translation. As the major problem of the first generation MT was the lack of linguistic information about source text, researchers therefore moved on to finding ways to capture this information. This gave rise to the development of the indirect MT systems which are generally regarded as second generation MT systems.

1.1.2.1.1 Direct MT System:

The direct method, also known as the "Transformer" method was the strategy adopted by the earliest MT systems. It is the most primitive method and uses a one stage process in which the systems simply translate the source language texts in to the corresponding word-to-word or phrase-to-phrase by using the bilingual lexicon. For example - direct translation from Hindi to Punjabi for राम ने श्याम को प्यार से गले लगाया । (*rām nē shyām kō pyār sē galē lagāyā*) is ਰਾਮ ਨੇ ਸ਼ਿਆਮ ਨੂੰ ਪਿਆਰ ਨਾਲ ਗਲੇ ਲਗਾਇਆ। (*rām nē shiām nūṃ piār nāl galē lagāiā).* The basic characteristic for such type of translation is that it is very simple and one needs to replace a word of source language to a word in target language using a bilingual dictionary.

They also perform some morphological analysis before looking into the bilingual lexicon for the root words. They will then perform the necessary

reordering of the words as according to the target language sentence format. The morphological processes may improve the quality of the translation but they don't really analyze the structure of the source language. An example of the direct MT system is SYSTRAN. The Direct Machine Translation was the technique developed in 1950s where the computers were in an early stage of technical development with very less speed which resulted in long processing time. Due to these constraints the direct MT used a very straightforward and easy to implement approach. It supports the translation of source language sentences to the sentences of the target language having structures similar to the structure in the source language. As very little effort is made to disambiguate the source language, this technique can't translate highly ambiguous sentences. The main problem of the direct MT approach is that it doesn't analyze the linguistic information or the meaning of source sentences before performing the translation. Without this information the resulting MT system can't always resolve the ambiguities that arise in the source sentence and /or during the translation. As a result, the direct MT systems can't provide a quality translation of the source language text.

The disadvantage of direct method is that it is unidirectional i.e. if the target is to be translated back into the source language, a different transformer must be used. It uses n^2 translation modules for translations among n languages, thus making it exponentially large for multi-language translating. Other problems with the direct method involves are in relation to the structure of

sentences if these are complex. It requires complex grammatical analyses, in the absence of which word ordering in the target language sentence can often be wrong. So direct translation is very inaccurate for complex text, but has been implemented successfully with a limited number of lexical entries. It is to be noted that this direct approach is most suitable choice for language pair that are closely related to each other.

1.1.2.1.2 Indirect MT System:

Owing to the fact that linguistic information helps an MT system to disambiguate source language sentences and to produce better quality target language translation, with the advance of computing technology, MT researchers started to develop methods to capture and use the linguistic information in the translation process. The indirect method occupies the level above direct translation in the MT pyramid and is also known as transfer or linguistic knowledge (LK) translation.

The transfer architecture not only translates at the lexical level, like the direct architecture, it also translates syntactically and sometimes semantically. The transfer method will first parse the sentence of the source language then will apply rules that map the grammatical segments of the source sentence to a representation in the target language. For example:

“Children like to play cricket” will be translated in Hindi as बच्चे क्रिकेट खेलना पसंद करते हैं (*bccē krikēṭ khēlnā pasand kartē haiṃ)*

In this example Verb Phrase ‘like’ is translated into पसंद करते (*pasand kartē),* Subject ‘Children’ is translated to बच्चे (*bccē).*

After syntactically and semantically analyzing the sentence, we can easily translate a sentence even with different structures. In this approach word reordering is also done. Suppose in English the word order in sentence is SVO when translated into Hindi, the word order of the translated sentence will be SOV.

The transfer approach uses n^2 transfer modules, n analysis components, and n synthesis components, where n is the number of languages in the translation system. Thus, one of its downfalls is the sheer size of the rules needed for its implementation.

1.1.2.1.3 Interlingua MT System:

The Interlingua or pivot approach appears at the apex of the MT pyramid. The main idea behind it is that the analysis of any language should result in a language-independent representation. The target language is then generated from that language-neutral representation. In a pure interlingua system there

are no transfer rules as a representation should be common to all languages used by the system.

This approach requires only one interlingual transfer model whereas the transfer approach requires n^2 transfer modules. The interlingual approach requires more analysis and is more abstract. It requires n analysis components, n Interlingua converters, n generation components where n is the number of languages in translation system.

There are few problems with the Interlingua approach. It requires an analyzer for each source language and a generator for each target language. Analysis of source text requires a deep semantic analysis that requires extensive word knowledge. Unfortunately, the true meaning of the sentence cannot always be extracted. Additionally, if a text is analyzed as deeply as is expected, then much of the source author's style will be lost.

1.1.2.2 Data-Driven Approach:

There are four approaches using data driven method:

1. Example Based MT
2. Knowledge Based MT
3. Statistics Based MT
4. Principle Based MT

1.1.2.2.1 Example Based MT:

Example-based translation is essentially translation by analogy. An Example-Based Machine Translation (EBMT) system is given a set of sentences in the source language (from which one is translating) and their corresponding translations in the target language, and uses those examples to translate other, similar source-language sentences into the target language. The basic premise is that, if a previously translated sentence occurs again, the same translation is likely to be correct again. EBMT systems are attractive in that they require a minimum of prior knowledge and are therefore quickly adaptable to many language pairs.

A restricted form of example-based translation is available commercially, known as a translation memory. In a translation memory, as the user translates text, the translations are added to a database, and when the same sentence occurs again, the previous translation is inserted into the translated document. This saves the user the effort of re-translating that sentence, and is particularly effective when translating a new revision of a previously-translated document (especially if the revision is fairly minor).

More advanced translation memory systems will also return close but inexact matches on the assumption that editing the translation of the close match will take less time than generating a translation from scratch.

wEBMT, ALEPH , English to Turkish, English to Japanese, English to Sanskrit and PanEBMT are some of the example based MT systems.

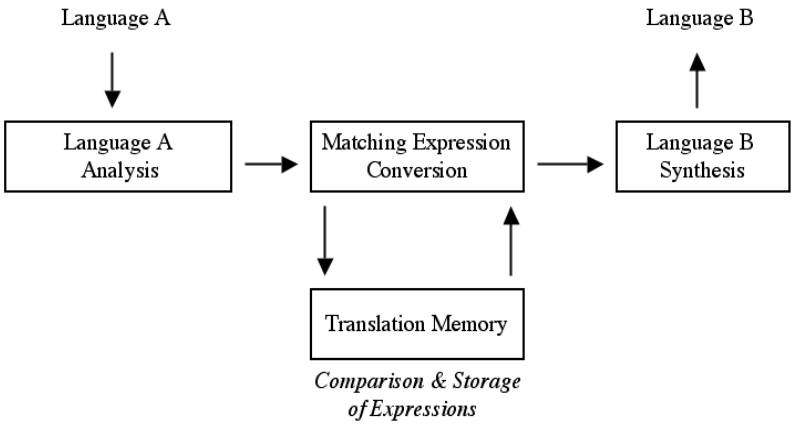

Figure 1.1: Example Based MT

1.1.2.2.2 Knowledge-Based MT:

Knowledge-Based MT (KBMT) is characterized by a heavy emphasis on functionally complete understanding of the source text prior to the translation to the target text. KBMT does not require total understanding, but assumes that an interpretation engine can achieve successful translation into several languages. KBMT is implemented on the Interlingua architecture; it differs from other interlingual techniques by the depth with which it analyzes the source language and its reliance on explicit knowledge of the world.

KBMT must be supported by world knowledge and by linguistic semantic knowledge about meanings of words and their combinations. Thus, a specific language is needed to represent the meaning of languages. Once the source language is analyzed, it will run through the augmenter. It is the Knowledge

base that converts the source representation into an appropriate target representation before synthesising into the target sentence.

KBMT systems provide high quality translations. However, they are quite expensive to produce due to the large amount of knowledge needed to accurately represent sentences in different languages.

English-Vientnamese Machine Translation system is one of the examples of KBMTS.

1.1.2.2.3 Statistics Based MT:

By the turn of the century, this newer approach based on statistical models – where in a word or phrase is translated to one of a number of possibilities based on the probability that it would occur in the current context - had achieved marked success. The best examples substantially outperform rule-based systems. Statistics-based Machine Translation (SMT) also may prove easier and less expensive to expand, if the system can be taught new knowledge domains or languages by giving it large samples of existing human-translated texts.

A string of source language, ε, can be translated into a string of target language in many different ways. Often, knowing the broader context in which ε occurs may serve to winnow the field of acceptable target language translations, but even so, many acceptable translations will remain; the choice among them is largely a matter of taste. In statistical translation, the view is

taken that every target language string, ζ, is a possible translation of ε. Every pair of strings is assigned (ε~ ζ) a number Pr(ζ | ε), which then is interpreted as the probability that a translator, when presented with ε, will produce ζ as his translation. Further the view is taken that when a native speaker of target language produces a string of target language words, he has actually conceived of a string of source language words, which he translated mentally. Given a target language string ζ, the job of the translation system is to find the string ε that the native speaker had in mind when he produced ζ. Thus the chances of error are minimized by choosing that source language string for which Pr(ε | ζ) is greatest:

$$\hat{e} = \text{argmax}_{\varepsilon} \Pr(\varepsilon \mid \zeta) \qquad \text{.....(1)}$$

The argmax operation denotes the search problem, i.e. the generation of the output sentence in the target language. The term Pr(ε | ζ) is termed as true probability distribution that target language sentence ε is translation of source language sentence ζ.

Bayes' theorem is used:

$$\Pr(\varepsilon \mid \zeta) = \frac{\Pr(\varepsilon)\Pr(\zeta \mid \varepsilon)}{\Pr(\zeta)} \qquad \text{.....(2)}$$

Since the denominator here is independent of ε, finding ê is the same as finding ε so as to make the product Pr(ε) Pr(ζ | ε) as large as possible. Thus, at the Fundamental Equation of Machine Translation is arrived at:

$$\hat{e} = \text{argmax}_{\varepsilon} \Pr(\varepsilon) \Pr(\zeta \mid \varepsilon) \qquad \text{.....(3)}$$

Equation (3) summarizes the three computational challenges presented by the practice of statistical translation: estimating the *language model probability,* Pr(ε); estimating the *translation model probability,* Pr(ζ | ε); and devising an effective and efficient suboptimal search for the input string that maximizes their product. These are known as the language modeling problem, the translation modeling problem, and the search problem.

Statistical translation systems works in two stages viz. training and translation. In training it "learns" how various languages work. Before translation, the system must be trained. Training is done by feeding the system with source language documents and their high-quality human translations in target language. With its resources, the system tries to guess at documents' meanings. Then an application compares the guesses to the human translations and returns the results to improve the system's performance. The whole process is carried out by dividing sentences into *N-gram*s. While training, statistical systems track common *N-gram*s, translations most frequently used are learnt and those meanings when finding the phrases in the future are applied. They also statistically analyze the position of *N-gram*s in relation to one another within sentences, as well as words' grammatical forms, to determine correct syntax. After their training, the systems are used to process actual phrases and produce the translation from what ever it has learnt in training phase.

Despite some success, however, severe problems still exist: outputs are often ungrammatical and the quality and accuracy of translation falls well below that of a human linguist - and well below demands of all but highly specialized commercial markets.

Moses, CASIA, Chinese-to-English, Google translate, LDV-COMBO and MARIE are some of the examples for statistical approach based MT systems.

1.1.2.2.4 Principle-Based MT:

Principle-Based MT (PBMT) Systems employ parsing methods based on the Principles & Parameters Theory of Chomsky's Generative Grammar. The parser generates a detailed syntactic structure that contains lexical, phrasal, grammatical, and thematic information. It also focuses on robustness, language-neutral representations, and deep linguistic analyses.

In the PBMT, the grammar is thought of as a set of language-independent, interactive well-formed principles and a set of language-dependent parameters. Thus, for a system that uses n languages, n parameter modules and one principles module are needed. Thus, it is well suited for use with the interlingual architecture.

PBMT parsing methods differ from the rule-based approaches. Although efficient in many circumstances, they have the drawback of language-dependence and increase exponentially in rules if one is using a multilingual translation system. It provides broad coverage of many linguistic phenomena,

but lacks the deep knowledge about the translation domain that KBMT and EBMT systems employ. Another drawback of current PBMT systems is the lack of the most efficient method for applying the different principles.

UNITRAN is one of the examples of Principle based Machine Translation system.

1.1.2.3 Hybrid Approaches:

Hybrid approaches to MT are becoming increasingly popular research subjects. The general idea behind hybrid approaches is to use a linguistic method to parse the source text, and a non-linguistic method, such as statistical-based or example-based, to assist with finding the proper interpretation.

1.1.2.3.1 Example-Based MT and Statistical-Based MT

EBMT works very well translating sentences that are already represented in its translation memory. SBMT can generate sentences with good accuracy, but is generally not successful when it handles idioms, collocations, and long-distance dependencies very well. By combining these two methods, a hybrid EBMT and SBMT system can first query the translation memory for matching sentences. If no close match is found, then a statistical analysis and interpretation of the sentence will be used.

TransEasy is one of the examples for Machine Translation based on Hybrid Approach. It uses Example and Statistical based approaches.

1.1.3 Key Activities [19-23]

Based on the above discussion of the Machine Translation techniques, it is evident that there are some common and key activities, which formulate a typical Machine Translation system.

An overview of such activities is provided below. These activities are usually executed in a sequence. However, depending upon the technique being followed, one or more of these activities may be omitted.

- **Pre-processing:** This module tokenizes the input text into words based on the list of word boundaries. Another major task performed in this phase is filtering. Filtering means detecting and marking certain special expressions like abbreviations, collocations, Named Entities, surnames, titles etc., in the input text. Text Spelling Standardization is another task in Pre-processing in which the words having spelling variations are replaced with the standard spelling words. This task helps in increasing the accuracy of the system. Filtering can be useful as the words or word sequences marked by the filter may not be required to go through the next two stages, namely, morphological analysis and part-of-speech tagging.

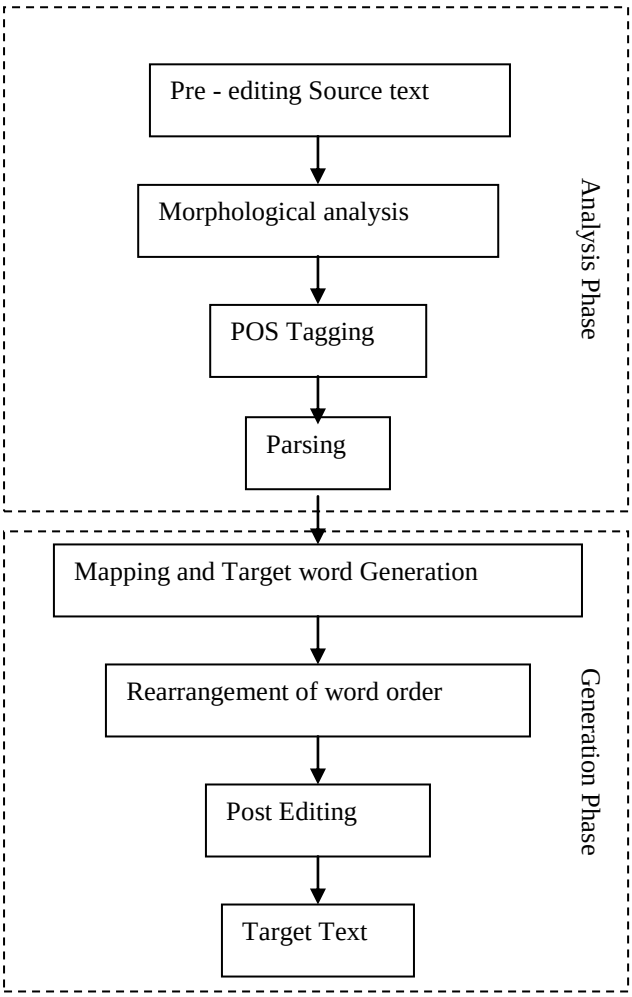

Fig 1.2 MT System General Architechture

- **Morphological analysis:** In this stage, morphological analyzer processes every unmarked token in the input text. The purpose of a morphological analyzer is to return root word and grammatical information about all the possible word classes (parts of speech) for a given word. Morphological analysis is essential for Hindi because it has a fairly rich system of inflectional morphology like other Indian languages. Morphological generator does exactly the reverse of

morphological analyzer. Given a root word and its grammatical information (including word class), a typical morphological generator will generate the word form or surface form for that root word.

- **Part-of-speech tagging:** The output of a morphological analyzer is usually ambiguous as it may return more than one POS (part-of-speech) tag for a single word. The reason being that in sentences, same word can be used as a noun or a verb, as a verb or a postposition etc. The job of a part-of-speech tagger is to disambiguate that ambiguous input by making use of the context information in which the word is being used. A part-of-speech tagger is also known as morphological disambiguator or simply a tagger.
- **Phrase chunking:** It is situated between POS tagging and a full-blown grammatical analysis, i.e. parsing. Whereas POS tagging works only on the word level and the grammatical analysis is supposed to build a tree structure of the sentence, phrase chunking assigns tags to word sequences in the sentence. There is no standardization about chunk names and their meanings, like POS tags, anyone can define his/her own chunk names and assign meanings to them. As chunking requires POS tagged text, its accuracy cannot be better than that of a POS tagger used. Chunking process is also known as shallow parsing as it simplifies the task for the next phase, i.e. parsing.
- **Parsing:** A parser is supposed to perform the full syntactic analysis of the given text. For every parsed sentence it is supposed to return a

data structure (mostly a parse tree) describing its syntactic components and their relationships with each other. It outputs the analysis based on the grammar it uses. For analyzing a sentence written in a particular language, the parser needs the grammar of that language. For specifying the grammar rules of a natural language, grammar formalism is required. Grammar formalism provides guidelines for specifying the underlying language's grammar rules. The parser will then make use of that grammar formalism. There are various grammar formalisms available for use like CFG (Context Free Grammar), GPSG (Generalized Phrase Structure Grammar), and HPSG (Head Driven Phrase Structure Grammar) etc. In simple terms, grammar formalism consists of a lexicon of words associated with their grammatical category and a set of rules specifying the sentence structure or syntax of the language. If a syntax-based parser fails to parse a sentence completely, then that sentence could be marked as incorrect or ungrammatical.

- **Translation and Transliteration:** Having all the necessary information regarding the words in a sentence, the next step is to find the equivalent in the target language. An alternative term for the computation of target texts from intermediate representations is synthesis. This is done with the help of lexicon. In Direct MT technique, this stage involves just dictionary look up. Some local reordering of words is also seen as generation in such systems. Sometime

morphological synthesizers are required to generate the word in target language. In transfer system, the generation phase is generally split into two modules, 'syntactic generation' and 'morphological generation'. In syntactic generation, the intermediate representation which is the output from analysis and transfer resembles a deep structure tree of the older type of transformational-generative grammar. It is converted by 'transformational rules' into an ordered surface structure tree, with appropriate labeling of the leaves with grammatical functions and features.

- **Rearrangement of word order:** If the source language and target language have different word order, then this step tries to reorder the words according to the grammar of target language. Any differences between languages can be dealt within the word generation and ordering stages. For example, the word order in English is Subject-Verb-Object. On the other hand, Hindi has relatively free word order. Generally a sentence in Hindi has the order Subject-Object-Verb. So to make the output according to the grammar of target language, some reordering techniques are required.
- **Post Processing:** The main factor which decides the amount of post-editing that needs to be done on a translation produced by machine is the quality of the output. Obviously enough, the difficulty of post-editing and the time required for it correlates with the quality of the raw MT output: the worse the output, the greater the post-edit effort. The post-

editor is a corrector for ill-formed sentences. It is basically tail-end of all the Machine Translation systems. It improves the translation quality by making corrections in the translation generated.

As mentioned earlier, not all of the above-mentioned activities are mandatory for a Machine Translation system. Selection of these activities depends on the approach a Machine Translation is following.

1.2 Research Questions

It has already been said that there is no Machine Translation system for Hindi to Punjabi presently. However, A number of Machine Translation systems between Indian and Non Indian languages have already been developed though their accuracy needs to be improved. Based on the brief introduction of Machine Translation given in section 1.1.1, the problem statement for the present research work has been formulated as below:

"To develop algorithms and lexical resources along with a software package to translate Hindi text to Punjabi text."

In other terms, the research question is to develop an automated Hindi to Punjabi Machine Translation System that will translate the Hindi text to Punjabi text. In this way, the richness of Hindi literatures opens up to Punjabi knowing people. This system will be helpful in reading the online Hindi newspapers in Punjabi language, Thus, removing the language barrier among

people. The users can type their email in Hindi language and the receiver can receive the email in Punjabi Language, Thus, making the communication in user's native languages possible.

1.2.1 Objectives

The objectives of this study are as follows:

1 To study Hindi and Punjabi Languages and their comparison.

2 To develop machine readable Hindi to Punjabi Dictionary for the purpose of translation.

3 To develop algorithm for generating Named Entities from the Corpus and then using this lexicon of Named Entities in translation.

4 To develop lexicon for collocations in Hindi text to be used during translation process.

5 To develop the lexicon and algorithm for handling surnames and titles in input text.

6 To adapt and use the existing lexical resources such as digital dictionary, morph etc.

7 To develop transliteration module for handling out-of-vocabulary words.

8 To develop algorithm for postprocessing tasks.

9 To develop test cases for evaluating the system critically.

1.2.2 Challenges

There are number of challenges for developing a Machine Translation system. Some of the major challenges faced in development of Hindi to Punjabi MT system are:

1. Lack of lexical resources such as digital bilingual dictionary, morphological analyzer and generator, POS tagger etc. There is no machine readable dictionary available for Hindi to Punjabi. Morphological Analyzer for Hindi has been developed by IIIT Hyderabad but this can not be used directly into the system and lots of modifications are required for making its use in the system. This is used for handling inflectional words of a word. It is not possible to store all the words including inflected words into the lexicon.
2. Multiple translations in Punjabi for Hindi words. There are many Hindi words which have different meanings depending upon the context in which the word is present in the sentence. The program has to automatically decide the exact translation. We have used n-gram technique for disambiguating the word.
3. Identifying Named Entities present in the text like the word vishal goyal, State Bank of India, S. Parkash Singh Badal, Dr. Parkash.
4. Collection of phrases that cannot be translated word by word and these have different meaning in collection than in individual.
5. Handling grammatical errors after translation i.e. grammatical agreement corrections.

1.2.3 Need and Scope

Machine Translation Systems are in great demand and are widely in use. For the past few years, number of Machine Translation Systems has been developed for Indian and foreign languages but their quality of translation is not up to mark for use in real projects. Thus, at present no such acceptable system is available for most of the Indian languages. The use of computers is gaining popularity in day-to-day tasks of word processing, writing reports, and printing official documents etc. All the documents are written in their regional official language. Thus for making these documents readable and useful for other regions, translation systems must be developed. Therefore, Machine Translation systems are an obvious requirement in such a situation. Recently, "Sampark: Machine Translation System among Indian Languages" has been funded by TDIL, Department of Information technology, Govt. of India, developed by Consortium of Institutions has released the Hindi to Punjabi Machine Translation System on trial basis on 13th August 2009 after spending three years. The translation is not promising and thus present system cannot be used for practical purposes. Indian languages have many features in common, so the present work could be well extendable to other Indian languages that are closely related to each other.

1.2.4 Potential Use [19-22]

The potential application areas of automatic Machine Translation are numerous and have the limits of imagination. Some of them are enumerated in this section.

Large Scale Translation: Large scale translation using MT is cost effective – there are many large companies saving time and money with MT. Leaving literature, sociology or legal texts aside (which require high level of publishable quality) MT is a success for technical documents especially within a particular domain. Typical texts are internal reports, operational manuals, repetitive publicity and marketing documents. Operational manuals, in particular, often represent many thousands of pages to be translated, and are extremely boring for human translators, and they are often needed in many languages (English, French, German, Japanese, etc.). But companies want fairly good quality of output as well. Manuals are repetitive, there may be frequent updates; and from one edition to another there may be very few changes. Automation is the obvious answer.

As an Aid for Translators: Machine Translation has changed the way translators work. The development of electronic termbanks, the increasing need to adhere to terminology standards, the overwhelming volumes of translation, and above all the development of facilities for using previous examples of translations have meant that translators could see the practical advantages of computerization in their work. Probably the largest users of computer aids for translation are found in the field of software and web localization. Localization means the adaptation of products for a particular

national or regional market, and the creation of documentation for those products. The incentive for computerization is the demand for the localization of publicity, promotional material, manuals for installation, maintenance and repair, etc. These must be available in the country (or countries) concerned as soon as the product itself is ready – often in a matter of days, not weeks.

Translation of Websites: A recent development is the appearance of software for translating webpages. Companies must now maintain high-profile presence on the Internet, in order to remain competitive. For multi-national companies, this also means that information on their websites must be made available in multi-languages. One solution is to refer users to online MT services but for many reasons this is unsatisfactory. Another is to engage a localization agency to translate every webpage. A third option which is increasingly adopted is to integrate one of the automatic webpage localization systems offered by many of the vendors of MT systems. Examples are ArabSite, IBM WebSphere, InterTran Website Translation Server, SDL Webflow, SystranLinks, and Worldlingo.

MT for Assimilation: Another main use of MT is assimilation, for getting the gist (essence) of the basic message of a text. The recipient does not necessarily require good quality. The main requirement is immediate use. However, the output must be readable; it must be reasonably intelligible for someone knowing the subject background. The wide availability of free translation of webpages makes it possible for companies and organizations to reach potential clients and customers who are unfamiliar with the language of

their websites; and many organizations provide links to such services for users to obtain translations of their websites.

MT as a Cross Language Information retrieval Tool: Closely related to the use of MT for translating texts for assimilation purposes is their use for aiding bilingual (or cross-language) communication and for searching, accessing and understanding foreign language information from databases and webpages. In the field of information retrieval there is much research at present on what is referred to as cross-language information retrieval (CLIR), i.e. information retrieval systems capable of searching databases in many different languages. Either they are systems which translate search terms from one language into other, and then do searching as a separate operation, with results presented en bloc to users; or, more ambitiously, translation of search terms or translation of output is conducted interactively with users.

MT as a Tool for Communication: It is probably true to say that one of the main applications of personal MT ('home') systems is the translation of correspondence (including personal e-mails) and the translation of web pages. Above all, there is oral communication involving translation. Although, we do not yet have speech translation, we do have systems with voice input and output, i.e. where users speak into the system, the spoken word or sentence is converted into text, the written text is translated into another text, and the system then produces spoken output.

MT for Summarization: Most people when faced with a foreign language text do not necessarily want the whole text translated, what they want is a

translated summary. There is a clear need for the production of summaries in languages other than the source. Summarization itself is a task which is difficult to automate; but applying MT to the task as well is an obvious expansion, either by translating the text as a whole into another language and then summarizing it, or by summarizing the original text and then translating the summary. The later has usually been the approach of researchers so far.

MT as Key technology for Cyber Revolution: Machine Translation can take information technology to the grassroots level and bring about sweeping societal changes through E-governance, E-commerce and E-entertainment leading to E-empowerment of the rural population. Local language information kiosks with computers, printers, Internet and E-mail facility are being set up to connect the Government to the citizen even at the grassroots level. At these kiosks, Machine Translation is essential so that all forms, records and information on the Government web site can be translated instantly into the local language that the people can understand. Similarly, the local language input by the citizens such as E-inquiries and E-grievances, should be machine translated at the click of the mouse, into a language that the concerned bureaucrat or minister can comprehend. [5].

1.3 Approach Applied for Our Machine Translation System

There are number of approaches discussed in the literature viz. Direct based, Transformer based, Interlingua based, Statistical etc. The choice of approach depends upon the available resources and the kind of languages involved.

Direct systems do not preclude syntactic or semantic analysis. There is a pragmatic constraint on the analysis, though, that it is subordinated to the translation task. Another difference concerns generation. A pure transfer system relies on a grammar for the target language to derive target sentences, while a direct system uses the word order of the source sentence as the point of departure for deriving a proper word order for the translation. A direct system relying on word-based analysis and transfer, will usually be able to derive some output for every input. The real issue, therefore, is empirical.

In general, if the two languages are structurally similar, in particular as regards lexical correspondences, morphology and word order, the case for abstract syntactic analysis seems less convincing. *Since the present research work deals with a pair of closely related language, so direct translation system is the obvious choice.* The overall system architecture shown in figure 1.3 is adopted for Hindi to Punjabi Machine Translation System. The system is divided into three stages: Preprocessing, Translation Engine, and Post Processing stage. Following is the description of various steps of this architecture.

1.3.1 Pre Processing

The preprocessing stage is a collection of operations that are applied on input data to make it processable by the translation engine. In our current work, we have performed following pre processing steps:

- Text Normalization
- Replacing Collocations

- Replacing Proper Nouns

1.3.1.1 Text Normalization

It works on spelling standardization issues, thereby resulting in multiple spelling variants for the same word. The major reasons for this phenomenon can be attributed to the phonetic nature of Indian languages and multiple dialects, transliteration of proper names, words borrowed from foreign languages, and the phonetic variety in Indian language alphabet. The variety in the alphabet, different dialects and influence of foreign languages has resulted in spelling variations of the same word. Such variations sometimes can be treated as errors in writing. During this phase of Pre Processing phase , rules specific to Hindi language which can handle such variations, which could result in more precise performance have been used for making the input text normalized for better accuracy.

For example we found widely used spelling variations for the Hindi word अंग्रेजी (*aṅgrējī)* as shown below:

अँग्रेजी, अंगरेजी, अन्ग्रेजी, अँगरेजी, अंग्रेजी, अंग्रेज़ी

1.3.1.2 Replacing Collocations means finding and replacing those combinations of words in Hindi that cannot be translated word to word and such combinations of words have different word in group rather than their individual. This activity helps a lot in increasing the accuracy of the system.

For example, the collocation उत्तर प्रदेश (*uttar pradēsh)*, if translated word to

word, will be translated as ਜਵਾਬ ਰਾਜ (*javāb rāj),* But it must be translated as ਉੱਤਰ ਪ੍ਰਦੇਸ਼ (*uttar pradēsh)*.

1.3.1.3 Replacing Proper Nouns means finding and replacing those combination of words in the input text that are acting as names of person, bank, river, ocean, days of week, months of year, university, cooperative society etc. For example: कमल गोयल (*kamal gōyal*) is a proper noun.

1.3.2 Tokenizer

The tokenizer takes the text generated by previous text as input. This module, using space, a punctuation mark, as delimiter, extracts tokens (word) from the text and gives it to Translation engine for analysis. This process is repeated for the whole text.

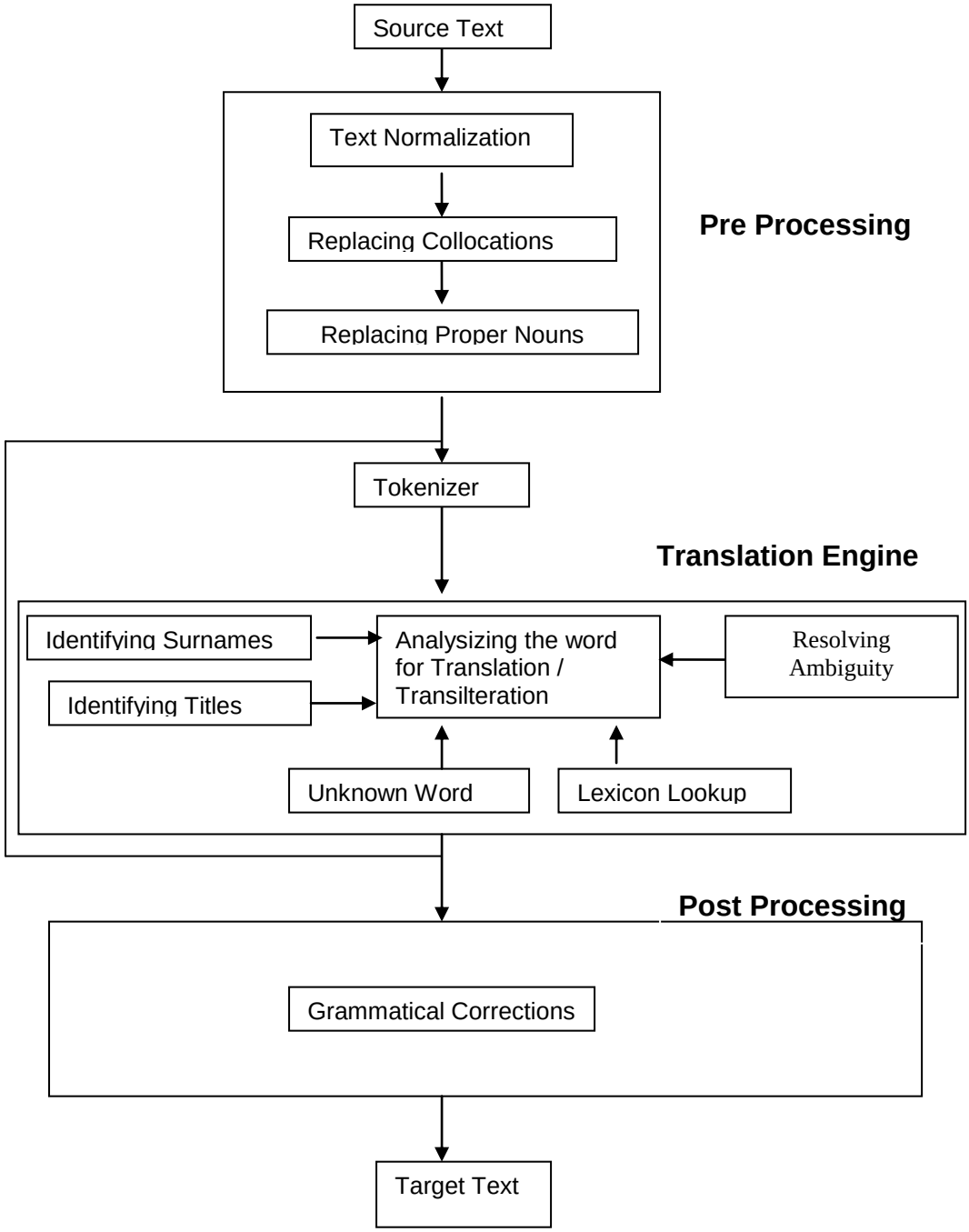

Figure 1.3: Overview of Hindi to Punjabi Machine Translation System

1.3.3 Translation Engine

The translation engine is responsible for translation of each token obtained from the previous step. It uses various lexical resources for finding the match of a given token in target language. Following is the description of how a token is passed through various modules.

1.3.3.1 Analyzing the word for Translation /Transliteration

The token obtained in the previous stage is passed through various stages.

1.3.3.1.1 Identifying Titles:

The token is checked whether it is a title like प्रो(prō), श्रीमती(shrīmtī) etc. If the current token is found to be a title, then the token next to it, should be transliterated instead of translation.

1.3.3.1.2 Identifying Surnames:

The token is checked whether it is a surname like अग्रवाल (agrvāl), ओबेरॉय (ōbērāy) etc. If the current token is found to be a surname, then the token previous to it, should be transliterated instead of translation.

1.3.3.1.3 Lexicon Lookup:

If the token does not satisfy above two steps, then it is looked into the lexicon for a match for direct word to word translation.

1.3.3.1.4 Resolving Ambiguity:

If the token is not present in the lexicon for direct translation, it is looked into the database of ambiguous words. If this token is found to be ambiguous, then disambiguity is resolved with the help of n-gram language modeling. The

system uses bigram and trigram databases, which contains one and two words respectively in the vicinity of an ambiguous word and corresponding meaning for that particular context.

1.3.3.1.5 Unknown Words:

If all the above modules fail to analyze the token, it is considered to be foreign/unknown word. Such words first pass through the morphological analysis phase based on the rules for inflections in Hindi words. Morphological generator generates the transliterated word using the inflectional rules and then checks the generated word in the Punjabi unigrams database for its genuinity. If this new generated word is found in the Punjabi unigrams, it is considered for translation otherwise the token is sent to transliteration module for transliteration.

Transliteration Module is the major module in the system that uses various rules specifically designed from the translation point of view.

1.3.4. Post Processing

After converting all the source text to target text, there are some of the grammatical errors that need to be corrected. For this purpose, we have formulated the rules for correcting the grammatical errors. Such rules have been implemented using Regular expressions and Pattern matching. This Post Processing phase is responsible for correcting grammatical errors in the generated output.

1.4 Thesis Outline

The study has been undertaken with the following chapter scheme:

In first chapter of this thesis, we introduce Machine Translation and provide details about various types of MT systems. The benefits, applications, and challenges of Machine Translation are described. After elaborating the various approaches used for Machine Translation and stages in a generic MT system we provide a formal description about the research question that we intend to undertake in this thesis work along with the major contribution and achievements of this research.

Chapter 2 discusses the existing work in the field of Machine Translation in India and outside India. This chapter on literature survey forms the basis of our work on developing the Machine Translation system and later on helps us in comparing our work with the existing state of the art in Machine Translation system.

Chapter 3 explains and compares the Hindi and Punjabi languages with respect to orthography, grammar, and Machine Translation.

Chapter 4 and 5 provide the design and implementation details of various activities involved in the Machine Translation system. Chapter 4 describes the system architecture and Pre processing stage. The chapter starts with the choice of approach and discusses the motivation behind its selection. Then the required resources are discussed followed by description of system architecture. The details of Pre processing phase which involves text

normalization, Identifying Collocations, Identifying Proper Nouns are discussed. Then tokenization process is explained. The details of the translation system involving the identifying titles, identifying surnames, lexicon lookup, word sense disambiguation module, transliteration module and post processing modules are discussed in Chapter 5.

Chapter 6 describes the post processing stage of the system.

Chapter 7 provides the evaluation of the system and its results.

Chapter 8 concludes this thesis by providing a summary of the research work undertaken, contributions of this research work, limitations, and some directions in which this work could be extended in the future.

In appendix A, the interface designed for text translation, website translation and email translation has been discussed. Test data set for Intelligibility test and accuracy test are available at Appendix B and C respectively.

1.5 Summary

In this chapter, introduction to Machine Translation, key activities involved, and various approaches for developing Machine Translation have been provided. It is followed by a formal statement for this research work along with its objectives, challenges involved, need and scope, and potential application areas of this system. Further, the approach followed to develop the Hindi to Punjabi Machine Translation System has been discussed along with an overview of the design of this system. The chapter concludes with a brief

outline of this thesis. The next chapter provides a survey of the existing literature in the field of Machine Translation.

Chapter 2

Survey of Literature

This chapter presents the state of the art in the field of Machine Translation. First part of this chapter discusses the Machine Translation systems for non Indian languages and second part discusses the Machine Translation systems for Indian languages.

2.1 Machine Translation Systems:

2.1.1 Machine Translation System for non Indian languages

Various Machine Translation systems have already been developed for most of the commonly used natural languages. This section briefly discusses some of the existing Machine Translation systems and the approaches that have been followed.

Georgetown Automatic Translation (GAT) System (1952), developed by Georgetown University, used direct approach for translating Russian texts (mainly from physics and organic chemistry) to English. The GAT strategy was simple word- for-word replacement, followed by a limited amount of transposition of words to result in something vaguely resembling English. There was no true linguistic theory underlying the GAT design. It had only six grammar rules and 250 items in its vocabulary. The translation was done using IBM 701 mainframe computer. Georgetown University and IBM jointly

conducted the Georgetown-IBM experiment in 1954 for more than sixty Russian sentences into English. The experiment was a great success and ushered in an era of Machine Translation research. The Georgetown MT project was terminated in the mid-60s.[8,23]

CETA (1961), included the linguistic theory unlike GAT, for translating Russian into French. It was developed at Grenoble University in France. It is based on Interlingua approach with dependency-structure analysis of each sentence at the grammatical level and transfer mapping from one language-specific meaning representation at the lexical level. During the period of 1967-71, this system was used to translate about 4,00,000 words of Russian mathematics and physics texts into French. It was found that it fails for those sentences for which complete analysis cannot be derived. In 1971, new and improved system GETA based on the limitations of CETA was developed. [24-27]

METAL (Mechanical Translation and Analysis of Languages) (1961), was developed at Linguistics Research Center, University of Texas for German into English. The system used indirect Machine Translation approach using Chomsky's transformational paradigm. Indirect translation was performed in 14 steps of global analysis, transfer, and synthesis. The performance and accuracy of the system was moderate.[28]

The Mark II (1964), a direct translation approach based Russian to English MT System for U.S. Air Force. It was developed by IBM Research Center. Translation was word by word, with occasional backtracking, Each Russian

item (either stem or ending) in the lexicon was accompanied by its English equivalent and grammatical codes indicating the classes of stems and affixes that could occur before and after it. In addition to lexical entries, processing instructions were also intermixed in the dictionary: 'control entries' relating to grammatical processes (forward and backward skips), and also instructions relating to loading and printing routines. There were some 25,000 such 'control entries' included in the dictionary. This contained 150,000 entries at the World's Fair demonstration, and 180,000 in the USAF version. A third of the entries were phrases, and there was also an extensive system of micro glossaries. An average translation speed of 20 words per second was claimed. The examples of Russian-English translations at the World's Fair were reasonably impressive (Bowers & Fisk (1965)). The Russian-English translations produced by Mark II were often rather crude and sometimes far from satisfactory. The limitations of word by word translation are more evident in the evaluation reports submitted by Pfafflin (1965), Orr & Small (1967), ALPAC(1966). An evaluation, MT research at the IBM Research Center ceased in 1966 (Roberts & Zarechnak 1974).

As one of the first operational MT systems, the IBM Russian-English system has a firm place in the history of MT. It was installed in the USAF's Foreign Technology Division at the Wright-Patterson Air Force Base, Dayton, Ohio, where it remained in daily operation until 1970. [29]

LOGOS (1964), a direct Machine Translation system for English-Vietnamese language pair was initially developed by US Private firm Logos Corporation.

Logos analyzes whole source sentences, considering morphology, meaning, and grammatical structure and function. The analysis determines the semantic relationships between words as well as the syntactic structure of the sentence. Parsing is only source language-specific and generation is target language-specific. Unlike other commercial systems the Logos system relies heavily on semantic analysis. This comprehensive analysis permits the Logos system to construct a complete and idiomatically correct translation in the target language. This Internet-based system allows 251 users to submit formatted documents for translation to their server and retrieve translated documents without loss of formatting. In 1971, It was used by the U.S. Air Force to translate English maintenance manuals for military equipment into Vietnamese. Eventually, LOGOS forged an agreement with the Wang computer company that allowed the implementation of the German-English system on Wang office computers. This system reached the commercial market, and has been purchased by several multi-national organizations (e.g., Nixdorf, Triumph- Adler, Hewlett-Packard). The System is also available for English-French, English-German language pairs. [30-32]

TAUM-AVIATION (1965), a transfer approach based English - French MT System for weather forecasts. It was developed at University of Montreal. After short span of time, the domain for translation shifted to translating aviation manuals by adding semantic analysis module to the system. The TAUM-AVIATION system is based on a typical second generation design (Isabelle et al. 1978, Bourbeau 1981). The translation is produced indirectly,

by means of an analysis/transfer/synthesis scheme. The overall design of the system is based on the assumption that translation rules should not be applied directly to the input string, but rather to a formal object that represents a structural description of the content of this input. Thus, the source language (SL) text (or successive fragments of it) is mapped onto the representations of an intermediate language, (also called normalized structure) prior to the application of any target language-dependent rule. In this system, the dictionaries list only the base form of the words (roughly speaking, the entry form in a conventional dictionary). In March 1981, the source language (English) dictionary included 4054 entries; these entries represented the core vocabulary of maintenance manuals, plus a portion of the specialized vocabulary of hydraulics. Of these, 3280 had a corresponding entry in the bilingual English-French dictionary. The system was evaluated and

the low accuracy of the translation by the system forced the Canadian Government to cancel the funding and thus TAUM project in 1981. [33-34]

SYSTRAN (1968) is a direct Machine Translation system developed by Huchins and Somers. The system was originally built for English-Russian Language Pair. In 1970, SYSTRAN System installation at United States Air Force (USAF) Foreign Technology Division (FTD) at Wright-Patterson Air Force Base, Ohio, replaced IBM MARK-II MT System and is still operational. Large number of Russian scientific and technical documents were translated using this system. The quality of the translations, although only approximate, was usually adequate for understanding content. In 1974, NASA also selected

SYSTRAN to translate materials relating to the Apollo-Soyuz collaboration, and in 1976, EURATOM replaced GAT with SYSTRAN. The Commission of the European Communities (CEC) purchased an English-French version of SYSTRAN for evaluation and potential use. Unlike the FTD, NASA, and EURATOM installations, where the goal was information acquisition, the intended use by CEC was for information dissemination - meaning that the output was to be carefully edited before human consumption. The quality for this purpose was not adequate but improved after adding lexicon entries specific to CEC related translation tasks. Also in 1976, General Motors of Canada acquired SYSTRAN for translation of various manuals (for vehicle service, diesel locomotives, and highway transit coaches) from English into French on an IBM mainframe. GM's English-French dictionary had been expanded to over 1,30,000 terms by 1981 (Sereda 1982). GM purchased an English-Spanish version of SYSTRAN, and began to build the necessary [very large] dictionary. Sereda (1982) reported a speed-up of 3-4 times in the productivity of his human translators. Currently, SYSTRAN System is available for translating in 29 language pairs. [35-39]

CULT(Chinese University Language Translator)(1968), is an interactive online MT System based on direct translation strategy for translating Chinese mathematics and physics journals into English. Sentences are analyzed and translated one at a time in a series of passes. After each pass, a portion of the sentence is translated into English. The CULT includes modules like source text preparation, input via Chinese keyboard, lexical analysis, syntactic and

semantic analysis, relative order analysis, target equivalence analysis, output and output refinement. CULT is a successful system but it appears somewhat crude in comparison to interactive systems like ALPS and Weidner. [40-44]

ALPS (1971), a direct approach based English into French, German, Portuguese and Spanish for Mormon ecclesiastical texts. It was developed at Brigham Young University. It was started with an aim to develop fully automatic MT System but later in 1973, it became Machine Aided System. It is an Interactive Translation System that performs global analysis of sentences with human assistance, and then performs indirect transfer again with human assistance. But this project was not successful and hence not operational. [45]

The METEO (1977) , is the world's only example of a truly fully automatic MT System for Canadian Meterological Centre's(CMC's) nation wide weather communication networks. METEO scans the network traffic for English weather reports, translates them directly into French, and sends the translations back out over the communications network automatically. This system is based on the TAUM technology as discussed earlier. It is probably the first MT system where translators had involved in all phases of the design, development and refinement. Rather than relying on post-editors to discover and correct errors, METEO detects its own errors and passes the offending input to human editors and output deemed correct by METEO is dispatched without human intervention. This system correctly translates 90-95%, shuttling the other 5-10% to the human CMC translators.[46-47]

An English Japanese Machine Translation System (1982) developed by Makoto Nagao et. al. The title sentences of scientific and engineering papers are analyzed by simple parsing strategies. Title sentences of physics and mathematics of some databases in English are translated into Japanese with their keywords, author names, journal names and so on by using fundamental structures. The translation accuracy for the specific areas of physics and mathematics from INSPEC database was about 93%.[48]

RUSLAN (1985), a direct Machine Translation system between closely related languages Czech and Russian, by Hajic J, for thematic domain, the domain of operating systems of mainframes. The system used transfer based architecture. This project started in 1985 at Charles University, Prague in cooperation with Research Institute of Mathematical Machines in Prague. It was terminated in 1990 due to lack of funds. The system was rule based, implemented in Colmerauer's Q-Systems. The system had a main dictionary of about 8,000 words, accompanied by transducing dictionary covering another 2,000 words. The typical steps followed in the system are Czech morphological analysis, syntactico semantic analysis with respect to Russian sentence structure and morphological synthesis of Russian. Due to close language pair, a transfer-like translation scheme was adopted with many simplifications. Also many ambiguities are left unresolved due to the close relationship between Czech and Russian. No deep analysis of input sentences was performed. The evaluations of results of RUSLAN showed that roughly 40% of the input sentences were translated correctly, about 40% of

input sentences with minor errors correctable by human post-editor and about 20% of the input required substantial editing or re-translation. There are two main factors that caused a deterioration of the translation. The first factor was the incompleteness of main dictionary of the system and second factor was the module of syntactic analysis of Czech. RUSLAN is a unidirectional system dealing with one pair of language Czech to Russian.[49]

PONS (1995) , an experimental interlingua system for automatic translation of unrestricted text, constructed by Helge Dyvik, Department of Linguistics and Phonetics, University of Bergen. 'PONS' is in Norwegian an acronym for "Partiell Oversettelse mellom Nærstående Språk" (Partial Translation between Closely Related Languages). PONS exploits the structural similarity between source and target language to make the shortcuts during the translation process. The system makes use of a lexicon and a set of syntactic rules. There is no morphological analysis. The lexicon consists of a list of entries for all word forms and a list of stem entries, or 'lexemes'. The source text is divided into substrings at certain punctuation marks, and the strings are parsed by a bottom-up, unification-based active chart parser. The system had been tested on translation of sentence sets and simple texts between the closely related languages Norwegian and Swedish, and between the more distantly related English and Norwegian. [50]

interNOSTRUM (1999) is a bidirectional Spanish-Catalan Machine Translation system. It was developed by Marote R.C. et al. It is a classical indirect Machine Translation system using an advanced morphological

transfer strategy. Currently it translates ANSI, RTF (Microsoft's Rich Text Format) and HTML texts. The system has eight modules: a deformatting module which separates formatting information from text, two analysis modules (morphological analyzer and part-of-speech tagger), two transfer modules (bilingual dictionary module and pattern processing module) and two generation modules (morphological generator and post-generator), and the reformatting module which integrates the original formatting information with the text. This system achieved great speed through the use of finite-state technologies. Error rates range around 5% in Spanish-Catalan direction when newspaper text is translated and are somewhat worse in the Catalan-Spanish direction. The Catalan to Spanish is less satisfactory as to vocabulary coverage and accuracy. [51]

ISAWIKA!(1999) is a transfer-based English-to-Tagalog MT system that uses ATN (Augmented Transition Network) as the grammar formalism. It translates simple English sentences into equivalent Filipino sentences at the syntactic level. [52]

English-to-Filipino MT system (2000) is a transfer based MT System that is designed and implemented using the lexical functional grammar (LFG) as its formalism. It involves morphological and syntactical analyses, transfer and generation stages. The whole translation process involves only one sentence at a time. [53]

Tagalog-to-Cebuano Machine Translation System (T2CMT)(2000) is a uni-directional Machine Translation system from Tagalog to Cebuano. It has three

stages: Analysis, Transfer and Generation. Each stage uses bilingual from Tagalog to Cebuano lexicon and a set of rules. The morphological analysis is based on TagSA (Tagalog Stemming Algorithm) and affix correspondence-based POS (part-of-speech) tagger. The author describes that a new method is used in the POS-tagging process but does not handle ambiguity resolution and is only limited to a one-to-one mapping of words and parts-of-speech. The syntax analyzer accepts data passed by the POS tagger according to the formal grammar defined by the system. Transfer is implemented through affix and root transfers. The rules used in morphological synthesis are reverse of the rules used in morphological analysis. T2CMT has been evaluated, with the Book of Genesis as input, using GTM (General Text Matcher), which is based on Precision and Recall. Result of the evaluation gives a score of good performance 0.8027 or 80.27% precision and 0.7992 or 79.92% recall. [54]

Turkish to English Machine Translation system(2000) is a hybrid Machine Translation system by combining two different approaches to MT. The hybrid approach transfers a Turkish sentence to all of its possible English translations, using a set of manually written transfer rules. Then, it uses a probabilistic language model to pick the most probable translation out of this set. The system is evaluated on a test set of Turkish sentences, and compared the results to reference translations. The accuracy comes out to be about 75.6%.[55]

CESILKO(2000), is a Machine Translation system for closely related Slavic language pairs, developed by HAJIC J, HRIC J K. and UBON V. It has been

fully implemented for Czech to Slovak, the pair of two most closely related Slavic languages. The main aim of the system is localization of the texts and programs from one source language into a group of mutually related target languages. In this system, no deep analysis had been performed and word-for-word translation using stochastic disambiguation of Czech word forms has been performed. The input text is passed through different modules namely morphological analyzer, morphological disambiguation, Domain related bilingual glossaries, general bilingual dictionary, and morphological synthesis of Slovak. The dictionary covers over 7, 00,000 items and it is able to recognize more than 15 million word-forms. The system is claimed to achieve about 90% match with the results of human translation, based on relatively large test sample. Work is in progress on translation for Czech-to-Polish language pairs.[56]

Bulgarian-to-Polish Machine Translation system (2000), has been developed by S. Marinov. This system has been developed based on the approach followed by PONS discussed above. The system needs a grammar comparison before the actual translation begins so that the necessary pointers between similar rules are created and system is able to determine where it can take a shortcut. The system has three modes, where mode 1 and 2 enable system to use the source language constructions and without making a deeper semantic analysis to translate to the target language construction. Mode 3 is the escape hatch, when the Polish sentences have to

be generated from the semantic representation of the Bulgarian sentence. The accuracy of the system has been reported to be 81.4%.[57]

Tatar (2001), a Machine Translation system between Turkish and Crimean, developed by Altintas K. et al., used finite state techniques for the translation process. It is in general disambiguated word for word translation. The system takes a Turkish sentence, analyses all the words morphologically, translates the grammatical and context dependent structures, translates the root words and finally morphologically generates the Crimean Tatar text. One-to-one translation of words is done using a bilingual dictionary between Turkish and Crimean Tatar. The system accuracy can be improved by making word sense disambiguation module more robust.[58]

Antonio M. Corbí-Bellot et. al. (2005) developed the open source shallow-transfer Machine Translation (MT) engine for the Romance languages of Spain (the main ones being Spanish, Catalan and Galician). The Machine Translation architecture uses finite-state transducers for lexical processing, hidden Markov models for part-of-speech tagging, and finite-state based chunking for structural transfer. The author claims that, for related languages such as Spanish, Catalan or Galician, a rudimentary word-for-word MT model may give an adequate translation for 75% of the text, the addition of homograph disambiguation, management of contiguous multi-word units, and local reordering and agreement rules may raise the fraction of adequately translated text above 90%. [59]

Carme Armentano-oller et. al (2005) extended the idea of A.M.Corbi-Bellot et. al. and developed an open source Machine Translation tool box which includes (a) the open-source engine itself, a modular shallow transfer Machine Translation engine suitable for related languages (b) extensive documentation specifying the XML format of all linguistic (dictionaries, rules) and document format management files, (c) compilers converting these data into the high speed format used by the engine, and (d) pilot linguistic data for Spanish—Catalan and Spanish—Galician and format management specifications for the HTML, RTF and plain text formats. They use the XML format for linguistic data used by the system. They define five main types of formats for linguistic data i.e. dictionaries, tagger definition file, training corpora, structural transfer rule files and format management files. [60]

Apertium (2005), developed by Carme Armentano-oller et. al is an open-source shallow-transfer Machine Translation (MT) system for the [European] Portuguese ↔ Spanish language pair. This platform was developed with funding from the Spanish government and the government of Catalonia at the University of Alicante. It is a free software and released under the terms of the GNU General Public License. Apertium originated as one of the Machine Translation engines in the project OpenTrad and was originally designed to translate between closely related languages, although it has recently been expanded to treat more divergent language pairs (such as English–Catalan). Apertium uses finite-state transducers for all lexical processing operations (morphological analysis and generation, lexical transfer), hidden Markov

models for part-of-speech tagging, and multi-stage finite-state based chunking for structural transfer. For Portuguese–Spanish language pair, promising results are obtained with the pilot open-source linguistic data released which may easily improve (down to error rates around 5%, and even lower for specialized texts), mainly through lexical contributions from the linguistic communities involved. [61]

ga2gd (2006), a robust Machine Translation system, developed by Scannell K.P., between Irish and Scottish Gaelic despite the lack of full parsing technology or pre-existing bilingual lexical resources. It includes the modules Irish standardization, POS Tagging, stemming, chunking, WSD, Syntactic transfer, lexical transfer, and Scottish post processing. The accuracy has been reported to be 92.72%. [62]

SisHiTra(2006) is a hybrid Machine Translation system from Spanish to Catalan. It was developed by Gonzalez et. al. This project tried to combine knowledge-based and corpus-based techniques to produce a Spanish-to-Catalan Machine Translation system with no semantic constraints. Spanish and Catalan are languages belonging to the Romance language family and have a lot of characteristics in common. SisHiTra makes use of their similarities to simplify the translation process. A SisHiTra future perspective is the extension to other language pairs (Portuguese, French, Italian, etc.). The system is based on finite state machines. It has following modules: preprocessing modules, generation module, disambiguation module and post-

processing module. The word error rate is claimed to be 12.5% for SisHiTra system.[63]

2.1.2 Machine Translation Systems for Indian languages

This section will summarize the existing Machine Translation systems for Indian languages that are as follows:

ANGLABHARTI (1991), is a machine-aided translation system specifically designed for translating English to Indian languages. English is a SVO language while Indian languages are SOV and are relatively of free word-order. Instead of designing translators for English to each Indian language, Anglabharti uses a pseudo-interlingua approach. It analyses English only once and creates an intermediate structure called PLIL (Pseudo Lingua for Indian Languages). This is the basic translation process translating the English source language to PLIL with most of the disambiguation having been performed. The PLIL structure is then converted to each Indian language through a process of text-generation. The effort in analyzing the English sentences and translating into PLIL is estimated to be about 70% and the text-generation accounts for the rest of the 30%. Thus only with an additional 30% effort, a new English to Indian language translator can be built. The attempt has been made to 90% translation task to be done by machine and 10% left to the human post-editing. The project has been applied mailnly in the domain of public health. [64]

Anusaaraka (1995) was developed at IIT Kanpur, and was later shifted to the Center for Applied Linguistics and Translation Studies (CALTS), Department of Humanities and Social Studies, University of Hyderabad. Of late, the Language Technology Research Center (LTRC) at IIIT Hyderabad is attempting an English-Hindi Anusaaraka MT System. The focus in Anusaaraka is not mainly on Machine Translation, but on Language access between Indian Languages. Using principles of Paninian Grammar (PG), and exploiting the close similarity of Indian languages, it essentially maps local word groups between the source and target languages. Where there are differences between the languages, the system introduces extra notation to preserve the information of the siurce language. The project has developed Language Accesors for Punjabi, Bengali, Telugu, Kannada and Marathi into Hindi. The output generated is understandable but not grammatically correct. For example, a Bengali to Hindi Anusaaraka can take a Bengali text and produce output in Hindi which can be understood by the user but will not be grammatically perfect. The system has mainly been applied for children's stories.[65]

Anubharati (1995), used EBMT paradigm for Hindi to English translation. The translation is obtained by matching the input sentences with the minimum distance example sentences. The system stored the examples in generalized form to contain the category/class information to a great extent. This made the example-base smaller in size and its further processing partitioning reduces

the search space. This approach works more efficiently for similar languages such as among Indian languages. [66]

The Mantra (MAchiNe assisted TRAnslation tool) (1999) translates English text into Hindi in a specified domain of personal administration specifically gazette notifications pertaining to government appointments, office orders, office memorandums and circulars. It is based on the TAG formalism from University of Pennsylvania. In addition to translating the content, the system can also preserve the formatting of input word documents across the translation. The Mantra approach is general, but the lexicon/grammar has been limited to the language of the domain. This project has also been extended for Hindi-English and Hindi-Bengali language pairs and also existing English- Hindi translation has been extended to the domain of parliament proceeding summaries.[67]

MAT (2002), a machine assisted translation system for translating English texts into Kannada, has been developed by Dr. K. Narayana Murthy at Resource Centre for Indian Language Technology Solutions, University of Hyderabad. Their approach is based on using the Universal Clause Structure Grammar (UCSG) formalism. The input sentence is parsed by UCSG parser and outputs the number, type and inter-relationships amongst various clauses in the sentence and the word groups that take on various functional roles in clauses. Keeping this structure in mind, a suitable structure for the equivalent sentence in the target language is first developed. For each word, a suitable target language equivalent is obtained from the bilingual dictionary. The MAT

System provides for incorporating syntactic and some simple kinds of semantic constraints in the bilingual dictionary. The MAT system includes morphological analyzer/generator for Kannada. Finally, the target language sentence is generated by placing the clauses and the word groups in appropriate linear order, according to the constraints of the target language grammar. Post Editing tool has been provided for editing the translated text. MAT System 1.0 had shown about 40-60% of fully automatic accurate translations. It has been applied to the domain of government circulars, and funded by the Karnataka government. [68]

An English–Hindi Translation System (2002) with special reference to weather narration domain has been designed and developed by Lata Gore et. al. The system is based on transfer based translation approach. MT system transfers the source sentence to the target sentence with the help of different grammatical rules and also a bilingual dictionary. The translation module consists of sub modules like Pre-processing of input sentence, English tree generator, post-processing of English tree, generation of Hindi tree, Post-processing of Hindi tree and generating output. The translation system gives domain specific translation with satisfactory results. By modifying the database it can be extended to other domains.[69]

VAASAANUBAADA (2002), an Automatic Machine Translation of Bilingual Bengali-Assamese News Texts using Example-Based Machine Translation technique, has been developed by Kommaluri Vijayanand et. al. It involves Machine Translation of bilingual texts at sentence level. In addition, it also

includes preprocessing and post-processing tasks. The bilingual corpus has been constructed and aligned manually by feeding the real examples using pseudo code. The longer input sentence is fragmented at punctuations, which results in high quality translation. Backtracking is used when the exact match is not found at the sentence/fragment level, leading to further fragmentation of the sentence. The results when tested by authors are fascinating with quality translation. [70]

ANGLABHARTI-II (2004) addressed many of the shortcomings of the earlier architecture. It uses a generalized example-base (GEB) for hybridization besides a raw example-base (REB). During the development phase, when it is found that the modification in the rule-base is difficult and may result in unpredictable results, the example-base is grown interactively by augmenting it. At the time of actual usage, the system first attempts a match in REB and GEB before invoking the rule-base. In AnglaBharti-II, provisions were made for automated pre-editing & paraphrasing, generalized & conditional multi-word expressions, recognition of named-entities. It incorporated an error-analysis module and statistical language-model for automated post-editing. The purpose of automatic pre-editing module is to transform/paraphrase the input sentence to a form which is more easily translatable. Automated pre-editing may even fragment an input sentence if the fragments are easily translatable and positioned in the final translation. Such fragmentation may be triggered by in case of a failure of translation by the 'failure analysis' module. The failure analysis consists of heuristics on speculating what might have

gone wrong. The entire system is pipelined with various sub-modules. All these have contributed significantly to greater accuracy and robustness to the system. [71]

The MaTra system (2004), a tool for human aided Machine Translation from English to Indian languages currently Hindi, has been developed by the Natural Language group of the Knowledge Based Computer Systems (KBCS) division at the National Centre for Software Technology (NCST), Mumbai (currently CDAC, Mumbai). The system has been applied mainly in the domain of news, annual reports and technical phrases. This system used transfer approach using a frame-like structured representation. The system used rule-bases and heuristics to resolve ambiguities to the extent possible. It has a text categorization component at the front, which determines the type of news story (political, terrorism, economic, etc.) before operating on the given story. Depending on the type of news, it uses an appropriate dictionary. It requires considerable human assistance in analyzing the input. Another novel component of the system is that given a complex English sentence, it breaks it up into simpler sentences, which are then analyzed and used to generate Hindi. The system can work in a fully automatic mode and produce rough translations for end users, but is primarily meant for translators, editors and content providers. [72]

ANUBHARTI-II (2004) has been generalized to cater to Hindi as source language for translation to any other Indian language, The system used hybrid Example-based Machine Translation approach which is a combination of

example-based approach and traditional rule-based approach. The example-based approaches emulate human-learning process for storing knowledge from past experiences to use it in future. It also uses a shallow parsing of Hindi for chunking and phrasal analysis. The input Hindi sentence is converted into a standardization form to take care of word-order variations. The standardized Hindi sentences are matched with a top level standardized example-base. In case no match is found then a shallow chunker is used to fragment the input sentence into units that are then matched with a hierarchical example-base. The translated chunks are positioned by matching with sentence level example base. Human post-editing is performed primarily to introduce determiners that are either not present or difficult to estimate in Hindi. [71]

Shakti (2004), is a Machine Translation system from English to any Indian language currently being developed at Language Technologies Research Centre, IIIT-Hyderabad. It has already produced output from English to three different Indian languages – Hindi, Marathi, and Telugu. It combines rule based approach with statistical approach. The rules are mostly linguistic in nature and the statistical approach tries to infer or use linguistic information. Although the system accommodates multiple approaches, the backbone of the system is linguistic analysis. The system consists of 69 different modules. About 9 modules are used for analyzing the source language (English), 24 modules are used for performing bilingual tasks such as substituting target language roots and reordering etc., and the remaining modules are used for

generating target language. The overall system architecture is kept extremely simple. All modules operate on a stream of data whose format is Shakti standard format (SSF). [73]

Shiva (2004), is an example based Machine Translation system from English to Hindi developed at IIIT Hyderabad.[73,74]

English-Telugu Machine Translation System has been developed jointly at CALTS with IIIT, Hyderabad, Telugu University, Hyderabad and Osmania University, Hyderabad. This system uses English-Telugu lexicon consisting of 42,000 words. A word form synthesizer for Telugu is developed and incorporated in the system. It handles English sentences of a variety of complexity.[74]

Telugu-Tamil Machine Translation System has also been developed at CALTS using the available resources here. This system uses the Telugu Morphological analyzer and Tamil generator developed at CALTS. The backbone of the system is Telugu-Tamil dictionary developed as part of MAT Lexica. It also used verb sense disambiguator based on verbs argument structure. [74]

ANUBAAD (2004) , an example based Machine Translation system for translating news headlines from English to Bengali, has been developed by Sivaji Bandyopadhyay at Jadavpur University Kolkata. During translation, the input headline is initially searched in the direct example base for an exact match. If a match is obtained, the Bengali headline from the example base is produced as output. If there is no match, the headline is tagged and the

tagged headline is searched in the Generalized Tagged Example base. If a match is obtained, the output Bengali headline is to be generated after appropriate synthesis. If a match is not found, the Phrasal example base will be used to generate the target translation. If the headline still cannot be translated, the heuristic translation strategy applied is - translation of the individual words or terms in their order of appearance in the input headline will generate the translation of the input headline. Appropriate dictionaries have been consulted for translation of the news headline. [75]

Hinglish (2004) , a Machine Translation system for pure (standard) Hindi to pure English forms developed by R. Mahesh K. Sinha and Anil Thakur. It had been implemented by incorporating additional layer to the existing English to Hindi translation (AnglaBharti-II) and Hindi to English translation (AnuBharti-II) systems developed by Sinha. The system claimed to be produced satisfactory acceptable results in more than 90% of the cases. Only in case of polysemous verbs, due to a very shallow grammatical analysis used in the process, the system is unable to resolve their meaning. [76]

Tamil-Hindi Machine-Aided Translation system has been developed by Prof. C.N. Krishnan at AU-KBC Research Centre, MIT Campus, Anna University Chennai. This system is based on Anusaaraka Machine Translation System architecture. It uses a lexical level translation and has 80-85% coverage. Stand-alone, API, and Web-based on-line versions have been developed. Tamil morphological analyser and Tamil-Hindi bilingual dictionary (~ 36k) are the by products of this system. They also developed a prototype of

English - Tamil MAT system. It includes exhaustive syntactical analysis. Currently, it has limited vocabulary (100-150) and small set of Transfer rules. [77]

AnglaHindi (2003) , a pseudo –interlingual rule-based English to Hindi Machine-Aided Translation System, developed by Sinha et. al. at IIIT, Kanpur. It is a derivative of AnglaBharti MT System for English to Indian languages. AnglaHindi besides using all the modules of AnglaBharti, also makes use of an abstracted example-base for translating frequently encountered noun phrases and verb phrasals. The system generates approximately 90% acceptable translation in case of simple, complex and compound sentences upto a length of 20 words. [78]

IBM-English-Hindi Machine Translation System has been initially developed by IBM India Research Lab at New Delhi with EBMT approach. Now, the approach has been changed to statistical Machine Translation between English and Indian languages. [79-84]

English to {Hindi, Kannada, Tamil} and Kannada to Tamil Language-Pair Example Based Machine Translation (2006) has been developed by Prashanth Balajapally. It is based on a bilingual dictionary comprising of sentence-dictionary, phrases-dictionary, words-dictionary and phonetic-dictionary and is used for the Machine Translation. Each of the above dictionaries contains parallel corpora of sentences, phrases and words, and phonetic mappings of words in their respective files. Example Based Machine Translation (EBMT) has a set of 75000 most commonly spoken sentences

that are originally available in English. These sentences have been manually translated into three of the target Indian languages, namely Hindi, Kannada and Tamil. [79-83]

Google Translate (2007), is based on statistical Machine Translation approach, and more specifically, on research by Franz-Josef Och. Before using statistical approach, Google translate was using SYSTRAN for its translation till 2007. Currently, it is providing the facility of translation among 51 language pairs. It includes only one Indian language Hindi. The accuracy of translation is good enough to understand the translated text. [Internet Source: http://translate.google.com/]

Punjabi to Hindi Machine Translation System (2007) has been developed by Gurpreet Singh Joshan et. al. at Punjabi University Patiala. This system is based on direct word-to-word translation approach. This system consists of modules like pre-processing, word-to-word translation using Punjabi-Hindi lexicon, morphological analysis, word sense disambiguation, transliteration and post processing. The system has reported 92.8% accuracy. [84]

Sampark: Machine Translation System among Indian languages (2009), developed by the Consortium of Institutions. Consortium of institutions include IIIT Hyderabad, University of Hyderabad, CDAC(Noida,Pune), Anna University, KBC, Chennai, IIT Kharagpur, IIT Kanpur, IISc Bangalore, IIIT Alahabad, Tamil University, Jadavpur University. Currently experimental systems have been released namely {Punjabi,Urdu, Tamil, Marathi} to Hindi

and Tamil-Hindi Machine Translation systems. The accuracy of the translation is not up to the mark.[Internet Source:http://sampark.iiit.ac.in]

Yahoo! Bable Fish (2008), developed by AltaVista, is a web-based application on Yahoo! that machine translates text or web pages from one of several languages into another. The translation technology for Babel Fish is provided by SYSTRAN. It translates among English, Simplified Chinese, Traditional Chinese, Dutch, French, German, Greek, Italian, Japanese, Korean, Portuguese, Russian, Swedish, and Spanish. [Internet Source: http://babelfish.yahoo.com/]

Microsoft Bing Translator (2009) is a service provided by Microsoft as part of its Bing services which allow users to translate texts or entire web pages into different languages. All translation pairs are powered by Microsoft Translation (previously Systran), developed by Microsoft Research, as its backend translation software. The translation service is also using statistical Machine Translation strategy to some extent [Internet Source: http://www.microsofttranslator.com/]

Bengali to Hindi Machine Translation System (2009) is a hybrid Machine Translation system, developed at IIT Kharagpur. This system uses multi-engine Machine Translation approach. It is based on the unfactored Moses SMT system with Giza++ (Josef,2000) derived phrase table as a central element. This system uses dictionary consisting of 15,000 parallel sysnets, Gazeteer list consisting of 50,000 parallel name list, monolingual corpus of 500K words both from source and target languages, suffix list of 100 Bengali

linguistic suffixes. The BLUE score obtained during system evaluation is 0.2318. [85]

2.2 Summary

As we have seen in the above discussion the English to Japanese, GAT (English-Russian), Mark-II (Russian-English), LOGOS (English-Vietnamese), SYSTRAN (English-Russian) ,CULT (Chinese mathematics and physics journals into English), ALP (English into French, German, Portuguese and Spanish), RUSLAN (Czech and Russian), CESILKO(Czech to Slovak), English-Arabic and Punjabi to Hindi Machine Translation Systems have been developed using direct MT approach for closely related language pairs. Some of these are very successful and popular Machine Translation systems which are still operational. Thus, it is concluded that direct Machine Translation approach is the most appropriate for closely related languages.

Hindi and Punjabi is a case of closely related but distinct languages as these languages are not mutually intelligible, having distinct orthographies, independent lexica and number of important structural differences in terms of syntax. Hindi and Punjabi, being one of the closest pairs of Indo-Iranian languages, are chosen in this study as a model for translation between any pair of close languages. They have most parts of their grammar in common although morphemes and expressions may differ. The use of narration in both languages is almost the same and a narration can directly be translated. But it

is not straightforward to translate some phrases, idioms and even some grammatical structures.

Hence, direct approach is most suitable approach for developing Hindi to Punjabi Machine Translation System. In the next chapter we will discuss about the comparative study of Hindi and Punjabi languages in detail.

Chapter 3

Comparative Study of Hindi and Punjabi

3.1 Introduction

India is a linguistically rich country having eighteen constitutional languages, which are written in ten different scripts. Indian languages can be broadly classified into five groups according to their origin and similarity. These are Indo-Aryan family (Hindi, Bangla, Assami, Punjabi, Marathi, Oriya and Gujarati); Dravidian family (Tamil, Telugu, Kannada and Malayalam); Austro-Asian family and Tibetan-Burmese family and Andamanese (Jha, 2005). Many of them are structurally similar called sibling languages. Within each group, there is high degree of structural similarity. With some efforts effective mapping rules can be created amongst languages within the same group. Indian languages are inflectional with a rich morphology, relatively free word order, and default sentence structure as SOV (Subject Object Verb). It is believed that Machine Translation systems can be developed with less effort and using direct approach between sibling language pairs. [85]

In this chapter, we will discuss the comparative study of the language pair of our Machine Translation system i.e. Hindi and Punjabi. Our motive of comparative analysis is to sort out the closeness between Hindi and Punjabi from Machine Translation point of view and to make the base for deciding about the appropriate approach to be followed for development of our

Machine Translation system. By analysis we mean the identification of bilingual rules for source language and target language so that the transfer of source language to target language can be performed by computers successfully. In order for MT systems to work, source and target languages must be fully analyzed. This kind of study, however, is not adequately covered by theoretical linguistics. V. Geethakumary [86] states that if the source language and the target language both have significantly similar linguistic features on all the levels of their structures then the first step to be adopted is that both languages should be analyzed independently. After the independent analysis, to sort out the different features of the two languages, comparison of the two languages is necessary. The present study has been undertaken keeping in view the Machine Translation system being developed for languages from Hindi to Punjabi. This is not a complete analysis, but rather a comparison to give some idea about Hindi and Punjabi grammar. It covers main aspects of Hindi and Punjabi languages. Details of both Hindi and Punjabi grammar can be found in Michel [87] and Singh and Singh [88] respectively. Following sections will discuss about the comparison between the Hindi and Punjabi Language on the basis of orthography and grammar. This chapter also discusses these languages from Machine Translation point of view.

3.2 Comparison between Hindi and Punjabi Language On the basis of Orthography: [87-102]

3.2.1 Family and Status:

Hindi and Punjabi languages belong to the same subgroup of the Indo-European family i.e. Indo-Aryan family of the languages. Hindi and Punjabi are spoken by about 577 million people and 100 million people all over the world respectively. Hindi and Punjabi have been ranked 4th and 11th widely spoken language in the world respectively (Ethnologue, 2009). In India, Hindi has been accorded the status of 'official language' by the central government for use for most administrative purposes, and Punjabi being the official language of the state the Punjab and has been accorded the status of 'official language' by the Punjab government for use for most administrative purposes. Both the languages have originated from Sanskrit (Masica 1991). Punjabi language is mostly used in the region of Punjab, Haryana, Delhi, Himachal Pardesh, Jammu & Kashmir and in some areas of Pakistan namely Punjab, Sindh and Blochistan. On the other hand, Hindi is a national language of India and is spoken and used by the people all over the country. But the main regions are Haryana, Uttar Pardesh, Rajasthan, Bihar and Chattisgarh.

3.2.2 Script

3.2.2.1 Devanagari script:

Hindi Language is written in Devanagari Script. It is written Left-to-Right. The Devanagari script, used for writing Sanskrit and other Indian languages had evolved over a period of more than two thousand years. Devanagari emerged around 1200 AD out of the Siddham script, gradually replacing the earlier,

closely related Sharada script (which remained in parallel use in Kashmir). Both are immediate descendants of the Gupta script, ultimately deriving from the Brāhmī script attested from the 3rd century BC; Nagari appeared in approx. the 8th century as an eastern variant of the Gupta script, contemporary to Sharada, its western variant. The descendants of Brahmi form the Brahmic family, including the alphabets employed for many other South and South-East Asian languages.

Nāgarī is in Sanskrit the feminine of *nāgara.* The feminine form is used because of its original application to qualify the feminine noun *lipi* "script". There were several varieties in use, one of which was distinguished by affixing *deva* "divine, deity" to form a tatpurusha compound meaning the "divine urban(e) [script]". However, the widespread use of "Devanagari" is a relatively recent phenomenon; well into the twentieth century, and even today, simply "Nagari" was (and is) also in use for this same script. The rapid spread of the usage of "Devanagari" seems also to be connected with the almost exclusive use of this script in colonial times (particularly by European scholars) to publish works in Sanskrit (held by many to be the language of the gods), even though traditionally nearly all indigenous scripts have actually been employed for this language. This has led to the establishment of such a close connection between the script and Sanskrit that it is erroneously widely regarded as "the Sanskrit script" today.

3.2.2.2 Gurmukhi Script:

A unique feature of Punjabi is that it is written in two mutually incomprehensible scripts. In India Punjabi language is written in Gurmukhi script, while in Pakistan it is written in Shamukhi (Urdu) script. Gurmukhi script is written Left-to-Right and Shahmukhi is written right-to-left. Gurmukhi Script derived from the Sharada script and standarized by Guru Angad Dev in the 16th century, was designed to write the Punjabi Language (Gill, Gleason, 1963). The word Gurmukhi is commonly translated as "from the mouth of Guru". However, the term used for the Punjabi script has somewhat different connotations. The opinion given by traditional scholars is that as the Sikh holy writings, before they were scribed, were uttered by the Gurus, they came to be known as Gurmukhi or the "Utterence of the Guru". And consequently, the script that was used for scribing the utterence was also given the same name. However, the prevalent view among Punjabi linguists is that as in the early stages the Gurmukhi letters were primarily used by Gurmukhs, or the Sikhs devoted to the Guruy, the script came to be associated with them. Another view is that as the Gurmukhs, in accordance with the Sikh belief, used to meditate on the letter ਵ, ਹ, ਗ, ਰ which jointly forms ਵਾਹਿਗੁਰੂ or God in Sikhism, these letters were called Gurmukhi or the "Speech of the Gurmukhs". Subsequently, the whole script came to be known as Gurmukhi.

Like most of the north Indian writing systems, the Gurmukhi script is a descendent of the Brahmi script. It is believed that Gurmukhi script was invented by the second Sikh Guru, Guru Angad Dev, However, it would be correct to say that script was standarized rather than invented, by the Sikh

Gurus. E.P. Newton (Panjabi Grammar, 1898) writes that at least 21 Gurmukhi characters are found in ancient manuscripts: 6 from 10th century, 12 from 3rd century BC and 3 from 5th century BC. Apparently, the first Sikh Guru, Guru Nanak Dev also used the Gurmukhi script for his writings. The usage of Gurmukhi letters in Guru Granth Sahib meant that the script developed its own orthographical rules. In the following epochs, Gurmukhi became the prime script applied for literary writings of the Sikhs. Later in the 20th century, the script was given the authority as the official script of the Eastern Punjabi Language. Meanwhile, in western Punjab, a form of the Urdu script, known as Shahmukhi is still in use.

3.2.3 Consonants:

3.2.3.1 Basic Consonants

There are thirty three basic consonants or consonant-like graphs in Devanagari script and thirty- five in Gurmukhi scripts which are as follows.

Table 3.1: Basic Consonants in Devanagari

क *k*	ख *kh*	ग *g*	घ *gh*	ङ *ṅ*
च *c*	छ *ch*	ज *j*	झ *jh*	ञ *ñ*
ट *ṭ*	ठ *ṭh*	ड *ḍ*	ढ *ḍh*	ण *ṇ*
त *t*	थ *th*	द *d*	ध *dh*	न *n*
प *p*	फ *ph*	ब *b*	भ *bh*	म *m*
य *y*	र *r*	ल *l*	व *v*	

श *sh*	ष *sh*	स *s*	ह *h*	

Table 3.2: Basic Consonants in Gurmukhi

ਕ *k*	ਖ *kh*	ਗ *g*	ਘ *gh*	ਙ *ṅ*
ਚ *c*	ਛ *ch*	ਜ *j*	ਝ *jh*	ਞ *ñ*
ਟ *ṭ*	ਠ *ṭh*	ਡ *ḍ*	ਢ *ḍh*	ਣ *ṇ*
ਤ *t*	ਥ *th*	ਦ *d*	ਧ *dh*	ਨ *n*
ਪ *p*	ਫ *ph*	ਬ *b*	ਭ *bh*	ਮ *m*
ਯ *y*	ਰ *r*	ਲ *l*	ਵ *v*	ਸ *s*
ੜ *ṛ*	ੳ	ਅ *a*	ੲ	ਹ *h*

In addition to basic consonants, there are other consonants that are formed with some of the basic consonants supplemented with a dot diacritic. In Devanagari script these are क़ (*k̲*), ख़ (*k̲h̲*), ग़ (*ġ*), ज़ (*z*), फ़ (*f*), ड़ (*ṛ*) and in Gurmukhi script, these are ਖ਼ (*k̲h̲*), ਗ਼ (*ġ*), ਜ਼ (*z*), ਫ਼ (*f*), ਸ਼ (*sh*). There is one more such consonant ਲ਼ (*l̤*) in Gurmukhi script. But it is not much frequent in clusters. It was a proposal to distinguish consonant ਲ (*l*) □ from ਲ਼ (*l̤*) by adding a dot diacritic like that used to distinguish ਸ (*s*) from ਸ਼ (*sh*). This however has met with no acceptance and is seldom if ever used.

3.2.3.2 Dead and Live Consonants:

Devanagari employs a sign known in Sanskrit as the *virama* or vowel omission sign. In Devanagari and Gurmukhi both, it is called *hal* or *halant*, and that term is used in referring to the virama or to a consonant with its vowel suppressed by the virama. The virama sign (◌्) nominally serves to cancel (or kill) the inherent vowel of the consonant to which it is applied. When a consonant has lost its inherent vowel by the application of virama, it is known as a *dead consonant*; in contrast, a *live consonant* is one that retains its inherent vowel or is written with an explicit dependent vowel sign.

3.2.3.3 Consonant Conjuncts:

The Indic scripts are noted for a large number of consonant conjunct forms that serve as orthographic abbreviations (ligatures) of two or more adjacent letterforms (Michael, 1986). This abbreviation takes place only in the context of a *consonant cluster*. An orthographic consonant cluster is defined as a sequence of characters that represents one or more dead consonants followed by a normal, *live* consonant letter.

In Devanagari, we have four consonant conjucts namely ज्ञ (ज् +ञ), क्ष (क् +ष), श्र (श् +र), त्र(त् +र).

In Gurmukhi, only three types of conjunct consonants are used. In all bases, a modified form of the second consonant is subjoined to the unaltered form of the first. In the first type, a form of ਹ(*h)* is subjoined. The following table shows the common combinations.

Table 3.3: Conjuct Consonants

Base	Form	Devanagari Equivalent	Example
ੜ (*ṛ)*	ੜ੍ਹ (*ṛh)*	ड़ (*ṛ)*	ਪੜ੍ਹ (*paṛh)*
ਨ (*n)*	ਨ੍ਹ (*nh)*	न्हं *nhaṃ*	ਨ੍ਹੇਰ (*nhēr)*
ਲ (*l)*	ਲ੍ਹ (*lh)*	ल्ह (*lh)*	ਲ੍ਹਾ (*lhā)*
ਮ (*m)*	ਮ੍ਹ (*mh)*	म्ह (*mh)*	ਮ੍ਹੈਂਸ (*mhaiṃs)*

In second type of conjunct, a form of ਰ (*r)* is subjoined to certain consonants, most commonly stops. These occur only in tatsamas (Those words that are directly borrowed from Sanskrit with little or no phonetic alteration) like ਪ੍ਰ , ਕ੍ਰ , ਸ਼੍ਰ etc. In Devanagari, when र is served as the second member of a cluster, it is indicated by a small diagonal slash (going in the opposite direction from that of the virama) written under the sign for the first member of a conjunct: क्र, प्र,द्र,ब्र

Similarly, in Devanagari, when र is served as the first member of a conjunct, the sound is indicated by a small hook placed on the top of the rekha for the second consonant: र्क, र्ह, र्श, र्म. This hook is deferred until after any matra written to the right side of the conjunct like र्थी, र्मो.

In third type of conjunct, a form of ਵ is subjoined. For example: ਸ੍ਵ in Gurmukhi is written as स्व in Devanagari, Similarly ਸ੍ਵਰ (*svar)* in Gurmukhi is written as स्वर (*svar)* in Devanagari.

Several Devanagari conjuncts are so irregular as to prelude the immediate recognition of their components. The most important of these are क्त, क्ष, ज्ञ,द्द,द्ध,द्म. The consonant श has a special combining form श्र that is often used in place of श् in some clusters.(e.g. श्र, श्व) . Slightly irregular conjuncts exist in which ह stand as the first element (e.g. ह्न ,ह्म, ह्य ,ह्ल , ह्व).

3.2.3.4 Geminate (Doubled) Consonants:

In Gurmukhi, gemination is written by the sign ੱ(addak) above and before the consonant to be doubled. In Devanagri, doubled consonant cluster, gemination is written by writing the first component of the consonant cluster as the truncate form of the consonant (which is frequently built from the independent version of the latter consonant by the deletion of the vertical bar that appears on the right side of many Devanagri characters and the second component of the consonant cluster is, the unaltered full symbol for the second consonant. For example: पक्की (*pakkī)* (ਪੱਕੀ (*packī)* in Gurmukhi), कच्चा (*kaccā)* (ਕੱਚਾ (*kaccā)* in Gurmukhi). Similarly, in Gurmukhi, clusters of unaspirated stop plus homorganic aspirate stops are written by use of

ੱ(addak) before the letter for the aspirate. In Devanagari, this cluster is written with the short form of unaspirated stop plus full form of homorganic aspirate stop. For example: अच्छा (ਅੱਛਾ in Gurmukhi), पक्खी (ਪੱਖੀ in Gurmukhi).

In a small number of cases, the components of a consonant are sting out in a horizontal line (e.g. न्न) , arranged vertically or juxtaposed in some less regular manner (द्ध, ह्म) . Similarly in the Gurmukhi two geminates /nn/ and /mm/ are written with /tippi/ (ੰ). For example: ਪੰਨਾ (पन्ना in Devanagari), ਪੰਮਾ (पम्मा in Devanagari). It must be noted that there are no short forms in Gurmukhi like in Devanagri for consonants. So, while transliterating the short from of Hindi consonant, it is transliterated into full form of that consonant in Gurmukhi like मग्न (*magn)* in Devanagari will be transliterated into ਮਗਨ *(magn).*

3.2.4 Vowels:

Both the Scripts possess two different forms for each of the vowels- Full form and short form.

3.2.4.1 Full form:

In Devanagari, a full form is employed for a vowel that does not immediately follow a consonant or consonant cluster, i.e. in word-initial position or when the second of a sequence of vowels. Whereas in Gurmukhi, when a vowel is not preceeded by a consonant, it is written with one of the three vowel bearers - consonant like sign – ੳ , ਅ, ੲ indicating the absence of consonant.

3.2.4.2 Short form (or *matra*):

In Devanagari, short form is used when the vowel immediately follows a consonant or consonant cluster. These short forms consist of lines, hooks or combination of both above, below or to the side of the consonantal characters. These vowels are written around (that is, below, above, to the right, and to the left) the consonant signs.

In Gurmukhi, there are 10 vowel characters, 9 vowel symbols, 2 symbols for nasal sounds and 1 symbol that duplicates the sound of a consonant (Malik 2006, Malik 2005) Whereas in Devanagari, there are 11 vowel characters, 10 vowel symbols, 2 symbols for nasal sounds.

Following table shows both the above form of vowels for both the scripts and their correspondence in the Devanagari and Gurmukhi scripts:

Table 3.4: Vowels in Devanagari and Gurmukhi

Devanagari		**Gurmukhi**	
Short Form	**Full Form**	**Short Form**	**Full Form**
No Sign	अ (*a*)	No Sign	ਅ(*a*)
◌ा (*ā*)	आ(*ā*)	◌ਾ(*ā*)	ਆ (*ā*)
◌ि(*i*)	इ (*i*)	◌ਿ(*i*)	ਇ (*i*)
◌ी(*ī*)	ई (*ī*)	◌ੀ (*ī*)	ਈ (*ī*)

ੁ(*u*)	उ (*u*)	ੁ (*u*)	ਉ (*u*)
ੂ(*ū*)	ऊ (*ū*)	ੂ (*ū*)	ਊ (*ū*)
ੇ(*ē*)	ए (*ē*)	ੇ (*ē*)	ਏ (*ē*)
ੈ(*ai*)	ऐ (*ai*)	ੈ (*ai*)	ਐ (*ai*)
ੋ(*ō*)	ओ(*ō*)	ੋ (*ō*)	ਓ (*ō*)
ੌ(*au*)	औ(*au*)	ੌ (*au*)	ਔ (*au*)
ृ (*r*)	ऋ (*ri*)	--	----
ं(*ṃ*)	---	ं / ੰ (*ṃ*)	---
ँ(*ṃ*)	---	ं / ੰ (*ṃ*)	---
Conjunct	---	ੱ	---

3.2.4.3 Inherent ‘a’:

One vowel, ‘a’ has no special short form. The absence of a matra adjacent to a consonant suffices to indicate the presence of this vowel. At the end of a word, the inherent ‘a’ is not normally vocalized.

3.2.4.4 Nasalized vowels:

The two signs are used for nasalization. In Devanagari, *anusvara* (ं *ṃ*) and *anunasika* (ँ *ṃ*) also called *candrabindu*. Indian grammarians have formulated elaborated rules describing when each of these is used.In practice, the distinction between the two notations is often not observed. The first of these, *anusvara* is always used when the vowel marking (whether short or long form) protrudes above the rekha (e.g. ईं, ऐं,कों, मौं). With other vowel

signs, both *anusvara* and *anunasika* can be used (e.g. मुंह (*muṃh)* / मुँह (*muṃh)*, आंख (*āṅkh)* /आँख (*āṅkh*)), although some writers take care to consistently employ only anusvara in all contexts. Whereas In Gurmukhi, bindu (ਂ) is used with ਆ, ਈ, ਐ, ਔ, ਅੈ, ਅੌ, ਾ, ੀ, ੇ, ੈ, ੋ, ੌ, ਓ and tippi (ੰ) is used with ਉ, ਊ, ਏ, ਅ, ੁ, ੂ, ਇ, ਿ.

3.2.5 Punctuation Marks:

Only viraama (।) or a double vertical line (॥) was used in traditional writing for marking end of sentence and the end of a verse respectively for both Devanagari and Gurmukhi scripts. In modern writings, period, comma, hyphen, semicolon, exclamation sign, question mark and dash have also been used. In the ancient Punjabi, the use of double dandi was customary at the end of the sentences but in contemporary Punjabi, only single Dandi is used.

3.2.6 Abbreviation:

Abbreviations are formed in Hindi by the use of either a small circle (॰)or a dot after the first syllable of the word to be abbreviated: प्रो॰ (*prō॰)*, डा॰ (*ḍā॰)*, ई॰ (*ī॰)*,

ਪੂੰ (*pūṁ*) whereas in Gurmukhi Script, sign (:) is used to mark abbreviation like ਪ੍ਰੋ:(*prō:*), ਡਾ: (*ḍā*).

3.2.7 Numerals:

Following chart shows the correspondence between the numerals of both the scripts:

Table 3.5: Numerals in Devanagari and Gurmukhi

Devanagari	Gurmukhi
०	੦
१	੧
२	੨
३	੩
४	੪
५	੫
६	੬
७	੭
८	੮
९	੯

3.2.8 Alphabetic Order:

The alphabetic order of Devanagari is a model of logic and rational design, reflecting a keen understanding of the phonetic properties of the sounds designated by the various characters in the system. In Devanagari, vowels

precede consonants with the latter divided up into groups containing stops and nasals, semi vowels, sibilants, and h respectively.

The full alphabetic order of Devanagari as used for Hindi is as follows:

अ आ इ ई उ ऊ ऋ ए ऐ ओ औ क (क़) ख (ख़) ग (ग़) घ ङ च छ ज ज़ झ ञ ट ठ ड ढ ण त त्र

थ द ध न प फ ब भ म य र ल व श ष स ह

The full alphabetic order of Gurmukhi as used for Punjabi is as follows:

ਅ ਆ ਇ ਈ ਉ ਊ ਏ ਐ ਓ ਔ ਸ ਸ਼ ਹ ਕ ਖ ਖ਼ ਗ ਗ਼ ਘ ਙ ਚ ਛ ਜ ਜ਼ ਝ ਞ ਟ ਠ ਡ ਢ ਣ ਤ ਥ ਦ ਧ ਨ ਪ

ਫ ਫ਼ ਬ ਭ ਮ ਯ ਰ ਲ ਲ਼ ਵ ੜ

In Hindi, sequence under each consonants is the letter without any symbol, then followed by vowel symbols ा, ि, ी, ु, ू, े, ै, ो, ौ

In Punjabi, Sequence under each consonants is the letter without any symbol, then followed by vowel symbols ਾ, ਿ, ੀ, ੁ, ੂ, ੇ, ੈ, ੋ, ੌ

3.3. Comparison between Hindi and Punjabi on the basis of grammar [87-102]

3.3.1 Nouns

Nouns in Hindi and Punjabi are highly inflected. Hindi and Punjabi both have two genders (masculine and feminine), two numbers (singluar and plural) whereas Hindi has three cases (direct, oblique, and vocative) and Punjabi has five cases (direct, oblique, vocative, ablative, and locative/instrumental). The

latter two cases in Punjabi are essentially now vestigial: the ablative occurs only in the singular, in free variation with oblique case plus ablative postposition, and the locative/instrumental is confined to set adverbial expressions.

Nouns in Hindi can be further divided into declensional subtypes, Class I (marked/definite) and Class II (unmarked/indefinite), with the basic difference being that the former has characteristic terminations in the direct singular while the later does not. While Punjabi Nouns may be further divided into extended and unextended declensional subtypes, with the former characteristically consisting of masculines ending in unaccented *-ā* and feminines in *-ī*.

3.3.2 Adjectives

In Hindi and Punjabi both, adjectives are of two basic kinds, declinable/inflected and indeclinable/uninflected. Declinable adjectives agree with the nouns they modify in gender (masculine vs. feminine), number (singular vs. plural), and case (direct vs. oblique). Indeclinable adjectives possess but a single form when modifying nouns of different genders, numbers, or cases. Indeclinable adjectives are completely invariable, and can end in either consonants or vowels (including ā and ī). These adjectives do not end in any characteristics sound or series of sounds.

Table 3.6 : Declinabale and Indeclibale Hindi Adjectives

	Declinable	**Indeclinable**

Hindi	काला(*kālā)*, अच्छा(*acchā)*, ठंडा(*ṭhaṇḍā)* etc.	सुंदर(*sundar)*,खराब(*kharāb)*, भारी(*bhārī)* etc.
Punjabi	ਕਾਲਾ(*kālā)*, ਚੰਗਾ(*caṅgā)*, ਠੰਢਾ (*ṭhaṇḍhā)* etc.	ਸੋਹਣਾ(*sōhṇā)*,ਖਰਾਬ(*kharāb)*, ਭਾਰੀ(*bhārī)* etc.

3.3.3 Postpositions

Postpositions denote the relation of noun, pronoun, or verb with the other components of sentence. It is the use of postpositions with a noun or verb that necessitates the noun or verb taking the oblique case. Hindi and Punjabi both have core and compound postpositions. Core postpositions are also known as one word primary postpositions. For example: Some of the core postpositions in Hindi are का, की, के, को, ने, पर, में, तक, से and in Punjabi are ਦਾ, ਦੇ, ਨੂੰ, ਨੇ, ਉੱਤੇ, ਵਿੱਚ, ਤੱਕ, ਤੋਂ. Compound postpositions are composed of the genitive primary postposition plus an adverb. These postpositions follow their oblique targets either directly or with the inflected genitive linker. For example: Some of the compound postpositions in Hindi are के लिए, के साथ, के सामने, से पहले and in Punjabi are ਦੇ ਵਿੱਚ, ਦੇ ਨਾਲ etc.

3.3.4 Pronouns

Hindi and Punjabi languages both have personal pronouns for the first and second persons, while for the third person demonstratives are used, which

can be categorized as proximate and non-proximate. Pronouns distinguish three persons (first, second, and third), two numbers (singular and plural), and two cases (direct and oblique), though not gender.

Table 3.7: Hindi and Punjabi Pronouns

Pronouns	**First Person**	**Second Person**	**Third Person (Proximate)**	**Third Person (non-proximate)**
Hindi	मैं(*maiṃ)*, हम (*ham)*	तू(tū), तुम *(tum)*, आप (āp)	यह(*yah)*, ये (*yē)*	वह(*vah)*, वे (*vē)*
Punjabi	ਮੈਂ(*maiṃ)*, ਅਸੀਂ (*asīṃ)*	ਤੂੰ(*tūṃ)*,ਤੁਸੀਂ(tusīṃ)	ਇਹ (*ih)*	ਉਹ(*uh)*

3.3.5 Verbs

In both Hindi and Punjabi, the major grammatical categories that structure the verbal system are those of aspect and tense. The term aspect is to be understood as indicating the nature of the action of a verb as to its beginning, duration, completion, or repetition, but without reference to its position in time. There are three grammatical aspects, the habitual, the progressive (or continuous), and the perfective. Verbal forms indicating one of these aspects is usually further specified for one of four tenses, i.e., the present, past, presumptive, and subjunctive. Like the nominal system, the Hindi and Punjabi

verbs involve successive layers of (inflectional) elements to the right of the lexical base.

Compound verbs, a highly visible feature of Punjabi and Hindi grammar, consist of a verbal stem plus an auxiliary verb. The auxiliary (variously called "subsidiary", "explicator verb", and "vector") loses its own independent meaning and instead "lends a certain shade of meaning" to the main/stem verb, which "comprises the lexical core of the compound". While most verb can act as a main verb, there is a limited set of productive auxiliaries. For example, Some of verbs in Hindi are रहना(*rahnā)*, होना(*hōnā)*, जाना(*jānā)*, देना (*dēnā)* and in Punjabi are ਰਹਿਣਾ (*rahiṇā)*, ਹੋਣਾ (*hōṇā)*, ਜਾਣਾ (*jāṇā)*, ਦੇਣਾ(*dēṇā)* etc.

3.3.6 Sentence Structure

Hindi and Punjabi both are SOV (Subject Object Verb) and free order languages. Structurally both Hindi and Punjabi languages are same. In both languages, sentence is comprised of Subject and Predicate. In both languages, the basic elements are Kaaraka. Both have eight numbers of Kaaraka which by combining with each other create a sentence. The general sequence for transitive Sentence is Karta, Karam , Kria e.g गणेश खेत में सोता है (*gaṇēsh khēt mēṃ sōtā hai)* and for intransitive sentence is karta, kriya e.g. गणेश भागा (*gaṇēsh bhāgā)*. In both languages the relation between kaarka's

are shown by postpositions. Total eight part-of-speeches are recognized in both Hindi and Punjabi. Beside this, both have same types of Nouns, Genders, Number, Persons, Tenses and Cases.

3.3.7 Vocabulary

Joshan and Lehal [84] carried out an experiment to find out the total number of words which use the same alphabets and vowel/vowel sounds and convey the same meaning in both languages. Results showed that about 8% of source language words come under this category. This provides an idea of the overlap of vocabulary across languages. Hence for this study, it strengthens the fact of close relationship between Hindi and Punjabi languages. Moreover, it gives boost to the idea of using transliteration of source text as last option.

3.4 Comparison of Hindi and Punjabi from Machine Translation point of view [87-102]

3.4.1 Language Structure (Syntactic Vs Analytic)

Hindi is both analytic and syntactic in nature. Thus, it is not a purely analytic in nature. It may cause a problem while translating text from Hindi to Punjabi. It can lead to an unacceptable output if left un-dealt.

3.4.2 Ambiguity

Ambiguity is one of the major NLP problems which have been a great challenge for computational linguists. In general, people are unaware of the ambiguities in the language they use because they are very good at resolving them using context and their knowledge of the world. But computer systems do not have this knowledge, and consequently do not do a good job of making use of the context.

Something is ambiguous when it can be understood in two or more possible ways or when it has more than one meaning. If the ambiguity is in a sentence or clause, it is called structural (syntactic) ambiguity. Following example shows the structural ambiguity in Hindi:

परमोद ने खाते हुए चोर को पकड़ा (*parmōd nē khātē huē cōr kō pakṛā)*

This sentence can be interpreted in two ways viz. Parmod caught the thief while eating or Parmod caught the thief when the thief was eating.

Lexical ambiguity also known as word level ambiguity is a problem in translating Hindi to Punjabi. In Hindi, lexical ambiguity has been found in Nouns, Verbs, and Postpositions etc. The postposition से in Hindi can be translated into number of Punjabi postpositions like ਤੋਂ, ਨੂੰ, ਜਿਹੇ, ਕਰਕੇ and ਨਾਲ depending upon the usage of से in the sentence. Similarly in Verb, like जाना can be translated into ਜਾਣਾ and ਜਾਣਿਆ. Similarly in case of proper nouns, like प्रकाश (*prakāsh)* can be translated into ਪ੍ਰਕਾਸ਼ (*prakāsh)* or ਚਾਨਣ (*cānaṇ)*.

To illustrate more, consider the following sentence:

राम आम खा रहा है । (*rām ām khā rahā hai*)

In the above example, word आम in the sentence is lexically ambiguous. Its meaning can be interpreted in two ways – mango (a fruit) and usual (an adjective) as in following examples:

Usage as Noun: तोता पेड़ पर बैठकर आम खा रहा है *(tōtā pēḍa par baiṭhakar ām khā rahā hai)*

Usage as Adjective: ऐसे चोरों से मिलना आम बात है जो चोरी के ख़िलाफ़ उपदेश देते हैं (*aisē cōrōṃ sē milnā ām bāt hai jō cōrī kē khailāpha updēsh dētē haiṃ)*

3.4.3 Gender disgreement

During translation, sometimes correct gender of a word is not reflected in the translated language and it causes gender disagreement with verb/postposition in the target language. For example,If we translate the sentence उसको किताब चाहिए (*uskō kitāb cāhiē)* using direct approach, it will be translated to ਉਸਨੂੰ ਕਿਤਾਬ ਚਾਹੀਦਾ ਹੈ (*usnūṃ kitāb cāhīdā hai)*. Here the word किताब (*kitāb)* is feminine in nature and thus translation of verb चाहिए (*cāhiē)* in the sentence

must agree with the feminine nature of ਕਿਤਾਬ (*kitāb)* and thus be translated into ਚਾਹੀਦੀ ਹੈ (*cāhīdī hai)*.

3.4.4 Problems in Identifying Proper Nouns

The problem arises when a word in Hindi Sentence which is used as proper name of a person, is translated by the system instead of transliterating it. Such words are required to transliterate rather than translation. For example consider following sentences

दीपक गोयल कहाँ है? (*dīpak gōyal kahāṃ hai?)*

The word दीपक (*dīpak)* can be translated to ਦੀਵਾ (*dīvā)*. But in this sentence, the word दीपक (*dīpak)* has been used as a proper noun and thus, must be transliterated to ਦੀਪਕ (*dīpak)* instead of translated to ਦੀਵਾ *(dīvā)*. This problem is also known as Named Entity Recognition. Thus, Named Entity Recognition(NER) problem is a subtask of information extraction that seeks to locate and classify atomic elements in text into predefined categories such as the names of persons, organizations, locations, expressions of times, quantities, monetary values, percentages, etc.

3.4.5 Problem related to Collocations

Collocation is two or more consecutive words with a special behavior. (Choueka: 1988). Collocation means those combinations of words in Hindi that cannot be translated word to word and such combinations of words have different word in group rather than their individual. These groups of words have a special behavior. The meaning of the collocation can not be predicted from its parts, there is usually an element of meaning added to the parts of collocation. For example, the collocation उत्तर प्रदेश (*uttar pradēsh)* if translated word to word, will be translated as ਜਵਾਬ ਰਾਜ (*javāb rāj)* But it must be translated as ਉੱਤਰ ਪ੍ਰਦੇਸ਼ (*uttar pradēsh)*. Thus, special attention is needed for such combinations of words in Hindi Langauge.

3.4.6 Problems related to Foreign Words

Modern Hindi includes number of foreign words that are adopted from other languages. These words do not have any meaning in Hindi language and is propagated as such to Punjabi language while translating. So, these words are treated as unknown words and must be transliterated. For example: क्रिकेट (*krikēṭ)*, मैच (*maic)*, जाकेट (*jākēṭ)* etc.

3.4.7 Spelling variations

The Cambridge Dictionary defines spelling as 'forming words with the correct letters in the correct order', or the ability to do this where variation is 'difference' or 'deviation' in the structure. The existence of the variants does not make much of the difference to the common person who is using the language because it does not come on the way of proper communication of the message but it is much important in case of Machine Translation. The major reasons for spelling variations in language can be attributed to the phonetic nature of Indian languages and multiple dialects, transliteration of proper names, words borrowed from foreign languages, and the phonetic variety in Indian language alphabet. [105]

For example, Following are the possible spelling variations for the Hindi word अंग्रेजी (*aṅgrēji*):

अँग्रेजी, अंगरेजी, अन्ग्रेजी, अँगरेजी, अंग्रेजी, अंग्रेज़ी

3.5 Conclusion

In this chapter we have tried to compare Hindi and Punjabi language from the point of view of orthography, grammar and Machine Translation. This study is by no means an exhaustive one. This study was primarily aimed at knowing the closeness between both the languages and thus, to find the appropriate approach for the development of Machine Translation.

We call a language pair to be closely related if the languages have the grammar that is close in structure, contain similar constructs having almost

same semantics, and share a great deal of lexicon. By closely related languages, we also mean in□ectively and morphosyntactically similar languages. Some linguist define closeness between the languages on the basis of features viz. common root, similar alphabets, similar verb patterns, structural similarity, similar grammar, similar religio-cultural and demograpohic contexts and references, a similar clearly displayed ability to blend with foreign tongues . Generally, such languages have originated from the same source and spoken in the areas in close proximity.

Hindi and Punjabi belong to same sub group of the Indo European family, thus are sibling languages. We have also observed that Hindi and Punjabi languages share all features of closely related languages. For such closely related sibling languages, effective translation can be achieved by word-for-word translation (Hajic et al., 2000) [90]. Thus, it is concluded that direct Machine Translation approach is promising for closely related languages Hindi and Punjabi.

Chapter 4

Pre Processing Phase

The present and the next chapter discuss the design and implementation of the algorithms and structures that formulate our Hindi to Punjabi Machine Translation system. For all the activities, the design of the databases used, if any, along with some sample entries from the databases and the approach followed for that activity have been discussed in detail. While describing the design of the databases used, only the fields or databases directly concerned with performing the activity under consideration have been provided. There may be some additional fields or databases used for proper functioning of this Machine Translation system but have virtually no impact on describing the approach, thus, description of such databases or fields have been avoided. All the activities of this Machine Translation system have been implemented in ASP.Net and their databases are in the MS-Access with Hindi and Punjabi text in Unicode format. This Machine Translation system accepts Hindi text as input and provides output in Gurmukhi script in Unicode.

This chapter provides first activity pre-processing of our Machine Translation system. The remaining activities have been detailed in the next chapters. Chapter 1 has already presented the complete design of this Machine Translation system.

4.1 Introduction

The preprocessing stage is a collection of operations that are applied on input data to make it processable by the translation engine. In the first phase of Machine Translation system, various activities incorporated include text normalization, replacing collocations and replacing proper nouns. Figure 4.1 presents the design of this pre-processing system in more detail.

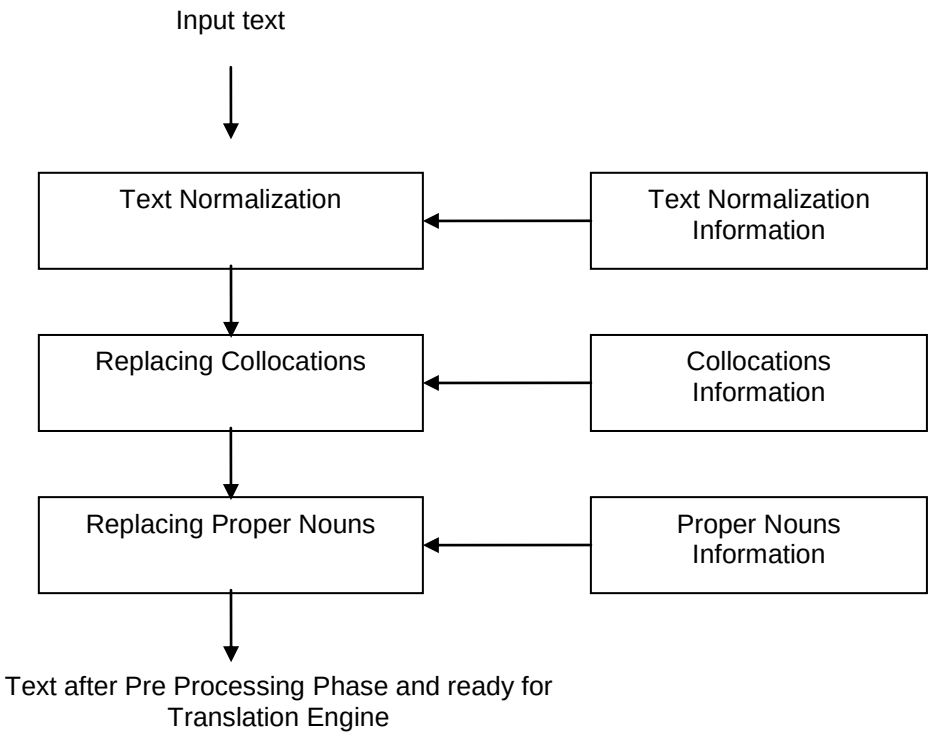

Figure 4.1: Pre-processing System Design

The four sub-activities of pre-processing system shown in Figure 4.1 are explained in the following sub-sections.

4.2 Text Normalization

Spelling conventions are an important feature of any language that is written.

The Cambridge Dictionary defines spelling as 'forming words with the correct

letters in the correct order', or the ability to do this where variation is 'difference' or 'deviation' in the structure. The existence of the variants does not make much of the difference to the common person who is using the language because it does not come on the way of proper communication of the message but it is much important in case of Machine Translation. This sub phase works on spelling standardization issues, thereby resulting in multiple spelling variants for the same word. The major reasons for this phenomenon can be attributed to the phonetic nature of Indian languages and multiple dialects, transliteration of proper names, words borrowed from foreign languages, and the phonetic variety in Indian language alphabet. The variety in the alphabet, different dialects and influence of foreign languages has resulted in spelling variations of the same word. Such variations sometimes can be treated as errors in writing. For example, Following are the possible spelling variations for the Hindi word अंग्रेजी *(aṅgrējī)*:

अँग्रेजी, अंगरेजी, अन्ग्रेजी, अँगरेजी, अंग्रेजी, अंग्रेज़ी

But out of these above possible spelling variants, only following are found in the Hindi corpus along with their frequency of occurrence:

Table 4.1: Frequency of Occurrence for Possible Spelling Variants of Word अंग्रेजी

अंग्रेज़ी *(aṅgrējī)*	87.017%
अंग्रेजी *(aṅgrējī)*	8.037%

अँगरेजी *(aṅgrējī)*	4.945%

Following rules specific to Hindi language have been framed which can handle such variations, which could result in more precise performance and for making the input text normalized for better accuracy:

Table 4.2: Text Normalization Rules

Rule No.	Rule	Example
1.	Chandrabindu (a half-moon with a dot) and bindu (a dot on top of alphabet) can be used interchangeably.	(i) अँगरेज़ (*aṅgrēja)*, अंगरेज़ (*aṅgrēja)* (ii) लाँच (*lāñc)*, लांच (*lāñc)*
2.	There are five consonant characters with nukta (a dot under consonant) viz. क़, ख़, ग़, ज़, फ़. With this rule, all consonants with nuktas and these consonants without nukta will be considered same.	(i)अंग्रेज़(*aṅgrēja)*, अंग्रेज (*aṅgrēj)* (ii) फोटो (*phōṭō)*, फ़ोटो (*phaōṭō)* (iii) तेज (*tēj)*, तेज़ (*tēja)*
3.	Hindi and many other Indian languages face the problems of 'schwa' (the default vowel 'a' that occurs with every consonant) deletion. Lots of spelling variations occur due to 'schwa' deletion. In order to normalize such words we delete all the halanth characters in the given word to generate spelling variant.	(i)भगवान्(*bhagvān)*, भगवान (*bhagvān)* (ii)अगरज (*agaraj)*, अग्रज (*agraj)* (iii)अक्सर (*aksar)*, अकसर(*akasar)*

		(iv)नारकोटिक (*nārkōṭik)*,नार्कोटिक (*nārkōṭik)*
4.	'Bindu' and 'न्' can be used interchangeably.	(i) कन्ठ(*kanṭh)*, कंठ (*kaṇṭh)*
5.	'Bindu' and 'म्' can be used interchangeably for words having 'म्' before the labial consonants like प,ब,फ,म,व in the word.	(i) अम्बु(*ambu)*, अंबु (*ambu)* (ii) पम्प (*pamp)*, पंप (*pamp)*
6.	There is one supplemental sound occasionally encountered in Hindi. This is the 'Visarga', noted in devanagari by the sign ('ः'). This sign appears only in tatsama vocabulary items. The words having sign ('ः') can also be written without it and is treated equivalent.	(i) अक्रमतः(*akrmat)*, अक्रमत (*akrmat)* (ii) अंततः (*antat:)*, अंतत (*antat)*
7.	Sometimes in place of 'ङ्'/'ण्'/ 'ञ' in the words, Chandrabindu (a half-moon with a dot) / bindu (a dot on top of alphabet) can be used and are equally correct. But it is very rare.	(i) गंङ्गा(*gaṅṅgā)*, गंगा(*gaṅgā)* (ii) ब्राण्ड (*brāṇḍ)*, ब्राँड *(brāṇḍ)* (iii) पञजा (*pañjā)*,पंजा *(pañjā)*
8.	'ई' and 'यी' cane be used	(i) नई (*naī)*, नयी(*nayī)*

	interchangeably in words.	
9.	'ए' and 'ये' can be used interchangeably in words.	(i) लिए (*liē*), लिये (*liyē*)

Analysis:

An exhaustive analysis has been done on large Hindi corpus collected from number of online resources for finding most useful rules among above mentioned rules. The Hindi Corpus used for analysis consists of about 1,00,000 words.

As it has been mentioned earlier that there can be a large number of possible spelling variations for a particular word depending upon the above rules, but in real data, among these variations, very less spelling variations are found. Only 1.492% words show the variations in their spellings. Following Table shows that out of these 1.492% words, percentage of words having one, two or three variations:

Table 4.3: % Word Occurrence with Spelling Variation Count

Number of variants	Words (%)	Example
1	99.985	जरूरत (*jarūrat)*, ज़रुरत (*jarurat)*
2	0.010	अँगरेजी (*aṅgrējī)*,अंग्रेजी (*aṅgrējī)*,अंग्रेज़ी (*aṅgrējī)*
3	0.005	फ़र्क (*phark)*, फर्क (*phark)*, फ़र्क़ (*phark)*, फ़रक (*phark)*

Thus, above table represents that, the variations found for majority of the words is just 1 and in worst case, it can go up to 3. And no case has been found with more than three spelling variants.

Following graph represents the importance and usage of different rules during analysis:

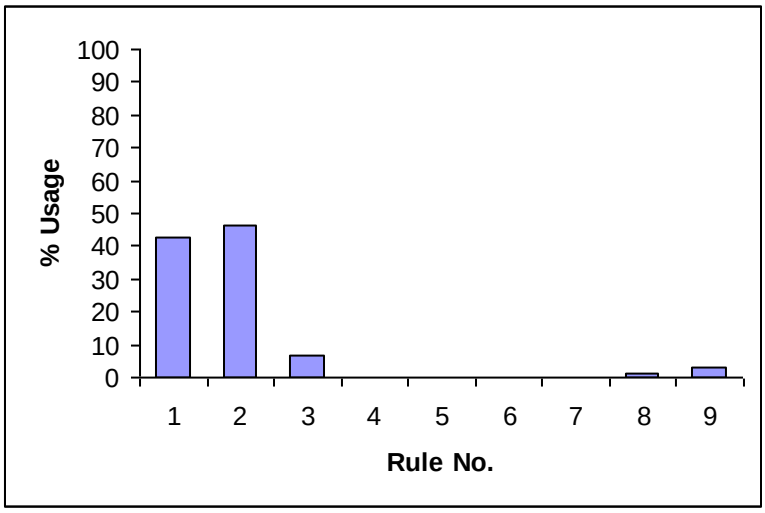

Figure 4.2: Analysis of % Usage of Various Text Normalization Rules

The above graph shows that Rule No 1 and 2 have maximum applicability and rests of the rule are seldom used. Rules other than 1 and 2 are also contributing in standardization but their role is limited.

It is found that only 7.45% text was standardized using the above rules. Following graph shows the analysis of the contribution of various rules Vs the number of words standardized:

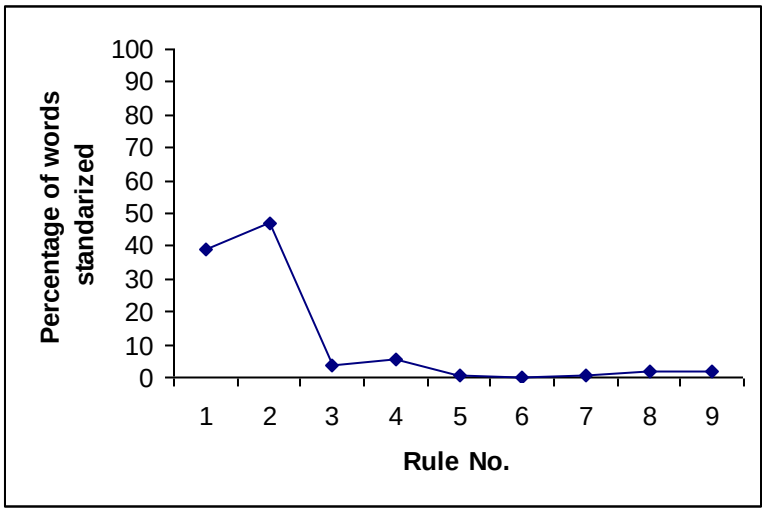

Figure 4.2: Analysis of Contribution of Text Normalization Rules

Majority of the standardization is done on the basis of the rules 1 and 2. Rest of the rules play very limited roles.

Database design:

Table 4.4 carries the design of the database used for storing information about text normalization.

Table 4.4: Text Normalization Database Design

Field Name	Description
nonstandardWord	Stores the non standard Hindi words
nswFreq	The frequency of the non standard word in the corpus analysed
standardWord	Hindi Word with standard spellings
swFreq	The frequency of the standard word in the corpus analysed

Sample database entries:

Table 4.5: Sample Entries of Text Normalization Database

nonstandardWord	nswFreq	standardWord	swFreq
फ़िल्म (*phailm)*	104	फिल्म (*philm)*	2165
हालांकि (*hālāṅki)*	1486	हालाँकि (*hālāṅki)*	3120
हाँ (*hāṃ)*	2045	हां (*hāṃ)*	4513
मौक़ा (*maukaā)*	700	मौका (*maukā)*	1580
किराए (*kirāē)*	600	किराये (*kirāyē)*	3411
हूं (*hūṃ)*	985	हूँ (*hūṃ)*	24910
एण्ड (*ēṇḍ)*	100	एंड (*ēṇḍ)*	1853
स्थाई (*sthāī)*	6	स्थायी (*sthāyī)*	20
इंटरनेश्नल (*iṇṭranēshnal)*	2	इंटरनेशनल (*iṇṭranēshnal)*	16
रविन्द्र (*ravindr)*	1	रविंद्र (*ravindr)*	4

Our Approach: The small offline module has been developed to generate the database for standardization. The module starts applying the rules discussed above, to the Hindi corpus collected from various sources like Hindi newspaper websites, various literatures available online etc. Thus, storing standard and non standard words extracted during corpus analysis along with their frequency into database. Then, the word having the maximum frequency among its spelling variant words is considered to be standard one. In future, this standard word may also be replaced with some of its other variants if frequency of the new spelling variant exceeds the current standard one. For example: the spelling variations हालांकि (*hālāṅk)* and हालाँकि (*hālāṅk)* are equally correct. If in some input text one variation is present more number of times than other, it can become standard one and vice versa. Thus, the

database is always in updated mode to accept changes for the existing entries also. The spelling variant(s) among non standards having frequency zero is omitted as they do not have existence in the real text. In this way, only those spelling variations are kept in the database that actually exists in Hindi Vocabulary. In this way, database is generated and presently database consists of 2,00,450 entries. Once this database is generated, during the preprocessing phase, the table lookup is done to replace the non standards words present in the database with the standard ones.

4.3 Replacing Collocations

After passing the input text through text normalization, the text passes through this Collocation replacement sub phase of Pre-processing phase. Collocation is two or more consecutive words with a special behavior. (Choueka :1988). Collocation means those combinations of words in Hindi that cannot be translated word to word and such combinations of words have different word in group rather that their individual. These groups of words have a special behavior. The meaning of the collocation can not be predicted from its parts, there is usually an element of meaning added to the parts of collocation. For example, the collocation उत्तर प्रदेश (*uttar pradēsh)* if translated word to word, will be translated as ਜਵਾਬ ਰਾਜ (*javāb rāj)* but it must be translated as ਉੱਤਰ ਪ੍ਰਦੇਸ਼ (*uttar pradēsh)*.

Related works:

Collocation has long been studied by lexicographers and linguists in various ways. Most collocation extraction methods are based on exploiting the various idiosyncrasies exhibited by collocations. The variation in statistical distributional characteristics has been widely employed to test for evidence of a collocation. Point wise Mutual Information is one of the earliest measures of association used for collocations [104]. Word association has also been measured using measures like Jaccard, Odds Ratio, etc [105]. Classical statistical hypothesis tests like Chisquare test, t-test, z-test, Log Likelihood Ratio [106] have also been employed to decide whether the constituents of a collocation are independent of each other. The variation in positional distribution of words in a collocation has also been used to identify significant collocations [107]. Lin [108] and Cruys et.al. [109] have used the principle of substitution to extract institutionalized collocations. They measure the difference between the distributional characteristics of the collocation and other similar collocations obtained by lexical substitution. While Lin uses PMI as the base association score, Cruys et.al. [109] use a strength of association measure motivated by the idea of selectional preference of a constituent word for another. Fazly et.al. [110] extract collocation by exploiting their syntactic fixedness. Katz [111] and Baldwin [112] use the context as a bag of words and build context vectors for representing collocations and their constituents. Comparison of the collocation and constituent vectors helps determine if the collocation is non-compositional. Moiron et.al. [113] have used the idea of

translation ambiguity to extract non-compositional MWEs. The noncompositional collocations will have more translation candidates on account of more uncertainty in translation. This uncertainty is measured as translational entropy. Language modeling has been used to extract domain specific phrases, by comparing the distribution of collocations in a general and domain-specific corpus [114]. All the measures mentioned above have modeled the problem as a ranking problem, where the collocations more likely to be MWEs are ranked higher. If an annotated training set is available, the MWE extraction problem can be set up as a classification problem [115]. For Indian languages, automated collocation extraction work has been limited. In fact, both of the existing works [115-117] use some kind of English translation for extracting Hindi collocations. Mukerjee et.al. [116] have used parallel corpus alignment and POS tag projection with parallel English corpus to extract complex predicates. Venkatapathy et.al. [115] use a classification based approach for extracting N-V collocations for Hindi. They use identity of the verb, semantic type of the object, case marker with the object, similarity of the verb form of the object with the verb-object pair under consideration etc. as features in a MaxEnt classifier. Thus, there are number of approaches for extracting Collocations from the corpus Like Frequency Method, Mean and Variance, Hypothesis Testing, t-test, Pearson's Chi-Square Test , Likelihood Ratio and Point wise Mutual Information.

Our Approach for Extracting Collocations:

Our focus is on extracting collocations which can be used for translation into Punjabi and above mentioned approaches are not suitable in our case. We have developed an offline module using t-test for automatically extracting the collocations from the Hindi Corpus. The steps performed for extracting the collocations using the t-test are as follows:

1. Extract all the unigrams, bigrams and trigram from the corpus along with their frequencies of their occurrence in the corpus and store into a database table tbl_Unigram, tbl_bigram, tbl_trigram respectively.
2. Combine all bigrams and their frequencies with their corresponding unigrams and their particular frequencies into the database table tbl_unibi.
3. Combine all trigrams and their frequencies with their corresponding unigrams and their particular frequencies into the database table tbl_unitri.
4. For each entry in table tbl_unibi, Expected mean (μ) is calculated using the formula P(bigram) = P(unigram1)P(unigram2). Where P(unigrami) = Frequency of unigram / total no of tokens in analyzed corpus.
5. For each entry in table tbl_unibi, Observed mean is calculated by dividing the frequency of the particular bigram with the total number of bigrams found during corpus analysis.
6. The variance (s2) is equal to the observed frequency.
7. Now Apply the formula t = (x -μ)/sqrt(s2/N). Where N is the total number of bigrams found during corpus analysis.

8. Apply the steps 4 to 7 for trigrams.
9. After applying t-test to all bigrams and trigrams, there are many bigrams and trigrams which are not good candidates for collocations. We removed all the analyzed bigrams and trigrams whose t-value is less than 2.576 (standard value provided by t-test).

The accuracy of the results for collocation extraction using t-test is not accurate and includes number of such bigrams and trigrams that are not actually collocations. Thus, manually such entries were removed and actual collocations were further extracted. The correct corresponding Punjabi translation for each extracted collocation is stored in the collocation table of the database. The collocation table of the database consists of 5000 such entries.

Database design: Table 4.6 carries the design of the database used for storing information about collocations.

Table 4.6: Collocation Database Design

Field Name	**Description**
Collocation	Stores the Hindi collocation
punjabiTranslation	Stores Punjabi translation for corresponding collocation

Sample database entries:

Table 4.7: Sample Entries in Collocation Database

Collocation	**punjabiTranslation**
आप को *(āp kō)*	ਤੁਹਾਨੂੰ *(tuhānūṃ)*

उत्तर प्रदेश (*uttar pradēsh)*	ਉੱਤਰ ਪ੍ਰਦੇਸ਼ (*uttar pradēsh)*
जाने जान (*jānē jān)*	ਜਾਣੇ ਜਾਣ (*jāṇē jāṇ)*
जोर-शोर (*jōr-shōr)*	ਜੋਰ-ਸ਼ੋਰ (*jōr-shōr)*
दैनिक जागरण (*dainik jāgraṇ)*	ਦੈਨਿਕ ਜਾਗਰਣ (*dainik jāgraṇ)*
नाग पंचमी (*nāg pañcmī)*	ਨਾਗ ਪੰਚਮੀ (*nāg pañcmī)*

Our approach for replacement of collocation:

In this sub phase, the normalized input text is analyzed. Each collocation in the database found in the input text will be replaced with the Punjabi translation of the corresponding collocation. This step helps a lot in increasing the translation accuracy of the system. It is found that when tested on a corpus containing about 1,00,000 words, only 0.001% collocations were found and replaced during the translation.

4.4 Replacing Proper Nouns

A great proposition of unseen words includes proper nouns like personal, days of month, days of week, country names, city names, bank names, organization names, ocean names, river names, university names etc. and if translated word to word, their meaning is changed. If the meaning is not affected, even though this step fastens the translation process. Once these words are recognized and stored into the proper noun database, there is no need to decide about their translation or transliteration every time in the case of presence of such words in input text for translation. This gazetteer makes

the translation accurate and fast. This list is self growing during each translation. Thus, to process this sub phase, the system requires a proper noun gazetteer that has been complied offline. For this task, we have developed an offline module to extract proper nouns from the corpus based on some rules. Following sections will explain the process of preparing the proper noun gazetteer and then the use of this gazetteer in pre-processing phase.

4.4.1 Compilation of Proper Nouns Gazetteer:

The gazetteer has been prepared using two approaches. One approach is through an offline module and another is through manual collection from various sources available online. The offline module further needs two databases containing titles like श्री (*shrī)*, श्रीमती (*shrīmtī)*, प्रो (*prō)* etc. and surnames like अवस्थी (*avsthī)*, आहूजा (*āhūjā)* etc. The database design of these databases has been explained in following sections. These databases have been prepared manually by collecting the data from various resources. The offline module accepts the Hindi text, applies various rules on it, extracts the proper names, and stores it in proper noun database. Following are the rules for extraction of proper nouns through offline module:

Rule 1: It checks whether the token from input text is matched with any entry in titles database, then the token next to current one is a proper noun like

श्रीमान कमल गोयल (*shrīmān kamal gōyal)*. Here श्रीमान (*shrīmān)* is a title and thus, कमल (*kamal)* is a proper noun.

Rule 2: It checks whether the token from input text is matched with any entry in surname database, then the token previous to current one is a proper noun like कमल गोयल (*kamal gōyal)*. Here, गोयल (*gōyal)* is a surname and thus, कमल (*kamal)* is a proper noun.

Using above two rules, initial proper nouns gazetteer is prepared from a large Hindi Corpus. Then manual entries are also added into this gazetteer for making it more robust for use by the translations system. After generating this gazetteer, there is need to call transliteration module (explained in the next chapter) for storing the equivalent Punjabi version of this Hindi entry. The database consists of 8000 such entries.

4.4.2 Replacing Proper Nouns:

After passing the input text through text normalization and collocation replacement sub phase of pre-processing, the output text from collocation phase becomes input text for this proper noun replacement sub phase of preprocessing. If there are any tokens in the input text that gets matched with the entries of the proper nouns database, are replaced with the corresponding equivalent Punjabi proper nouns.

Database design: Table 4.8 carries the design of the database used for storing information about proper nouns.

Table 4.8: properNoun Database Design

Field Name	Description
hindiPropernoun	Stores the Hindi version of proper noun
punjabiProperNoun	Stores equivalent Punjabi version of the proper noun.

Sample database entries:

Table 4.9: Sample Entries of properNoun Database

hindiProperNoun	punjabiProperNoun
अमर सिंह (*amar siṃh)*	ਅਮਰ ਸਿੰਘ (*amar sirigh)*
आजाद नगर (*ājād nagar)*	ਆਜ਼ਾਦ ਨਗਰ (*āzād nagar)*
इंग्लैंड (*iṅglaiṇḍ)*	ਇੰਗਲੈਂਡ (*iṅglaiṇḍ)*
इंदिरा गांधी (*indirā gāndhī)*	ਇੰਦਰਾ ਗਾਂਧੀ (*indrā gāndhī)*
उत्तर भारत (*uttar bhārat)*	ਉੱਤਰ ਭਾਰਤ (*uttar bhārat)*
जमना बाई स्कूल (*jamnā bāī skūl)*	ਜਮਨਾ ਬਾਈ ਸਕੂਲ (*jamnā bāī sakūl)*

4.5 Summary

In this chapter, pre-processing activity of our Machine Translation system has been provided. Design and implementation details of these activities have been discussed. Along with the database design, some excerpts from the respective databases have been provided to make the design more clear. In the next chapter, the remaining activities of our Machine Translation system, i.e. tokenizer and translation engine are discussed.

Chapter 5

Tokenizer and Translation Engine

5.1 Tokenizer

Tokenizers (also known as lexical analyzers or word segmenters) segment a stream of characters into meaningful units called tokens. The tokenizer takes the text generated by pre processing phase as input. Individual words or tokens are extracted and processed to generate its equivalent in the target language. This module, using space, a punctuation mark, as delimiter, extracts tokens (word) one by one from the text and gives it to translation engine for analysis till the complete input text is read and processed.

5.2 Translation Engine

The translation engine is the main component of our Machine Translation system. It takes token generated by the tokenizer as input and outputs the translated token in the target language. These translated tokens are concatenated one after another along with the delimiter. Then this generated text is passed on to the postprocessing phase. Translation Engine Phase of the system involves various sub phases that are Identifying titles, Identifying surnames, word-to-word translation using lexicon lookup, Word sense disambiguation and handling out-of-vocabulary words. All the modules have equal importance in improving the accuracy of the system. In this chapter,

these modules are described in detail followed by an example. This phase comprises of following sub phases:

1. Identifying titles
2. Identifying surnames
3. Word-to-word translation using lexicon lookup
4. Word sense disambiguation
5. Handling out-of-vocabulary words
   5 (a) Word Inflectional analysis and generation
   5 (b) Transliteration

5.2.1 Identifying Titles

Title may be defined as a formal appellation attached to the name of a person or family by virtue of office, rank, hereditary privilege, noble birth, or attainment or used as a mark of respect. Thus word next to title is usually a proper noun. And sometimes, a word used as proper name of a person has its own meaning in target language. When this word is passed through the translation engine, it is translated by the system. This cause the system failure as these proper names should be transliterated instead of translation. For example consider the Hindi sentence श्रीमान हर्ष जी हमारे यहाँ पधारे। (*shrīmān harsh jī hamārē yahāṃ padhārē)*. In this sentence, ਹਰਖ (*harsh)* has the meaning “joy”. The equivalent translation of ਹਰਖ (*harsh)* in target language is ਖੁਸ਼ੀ (*khushī)*. Thus, the sentence will be translated as ਸ਼੍ਰੀਮਾਨ ਖੁਸ਼ੀ ਜੀ ਸਾਡੇ ਇੱਥੇ ਪਧਾਰੇ । (*shrīmān khushī jī sāḍē itthē padhārē).* But actually it must be

translated as ਸ਼੍ਰੀਮਾਨ ਹਰਸ਼ ਜੀ ਸਾਡੇ ਇੱਥੇ ਪਧਾਰੇ । (*shrīmān harash jī sāḍē itthē padhārē).* The reason is straightforward that in this sentence हर्ष (*harsh)* word is acting as proper noun and it must be transliterated and not translated.

In this system, a small module has been developed for locating such proper nouns where titles are present as their previous word like श्री (*shrī)*, श्रीमान (*shrīmān)*, श्रीमती (*shrīmtī)* etc. There is one special character '॰' in Devanagari script to mark the symbols like डा॰, प्रो॰. If tokenizer found this symbol during reading the text, the word containing it, will be marked as title by setting the IsTitle Flag to true. If isTitle flag has been set to true, the next word generated by tokenizer will be transliterated and not processed for translation. After the word next to title will be transliterated, isTitle flag is again reset to False. The named entities found from the text through this module are also added to the proper nouns database automatically. It improves the systems in two ways – one, it helps in continuously increasing the proper noun coverage, Second, the expansion of proper noun database will increase the speed of translation.

Database design: Table 5.1 carries the design of the database used for storing information about titles.

Table 5.1: Titles Database Design

Field Name	Description
titleInHindi	Stores the titles in Hindi
titleInPunjabi	Stores the corresponding translated titles

	in Punjabi

Sample Database Entries:

The title database consists of 14 entries. Following table shows some of the database entries for titles database:

Table 5.2: Sample Entries of Titles Database

titleInHindi	titleInPunjabi
प्रो (*prō*)	ਪ੍ਰੋ (*prō*)
श्रीमती (*shrīmtī*)	ਸ਼੍ਰੀਮਤੀ (*shrīmtī*)
श्रीमान (*shrīmān*)	ਸ਼੍ਰੀਮਾਨ (*shrīmān*)
श्री (*shrī*)	ਸ਼੍ਰੀ (*shrī*)

This database can be extended at any time to allow new titles to be added.

5.2.2 Identifying Surnames

Surname may be defined as a name shared in common to identify the members of a family, as distinguished from each member's given name. It is also called family name or last name. Thus the word previous to surname is usually a proper noun. And sometimes, a word used as proper name of a person has its own meaning in target language. When this word is passed through the translation engine, it is translated by the system. This causes the system failure as these proper names should be transliterated instead of translation. For example consider the Hindi sentence प्रकाश सिंह हमारे यहाँ

ਧਧਾਰੇ।. *(prakāsh siṃh hamārē yahāṃ padhārē)* In this sentence, ਪ੍ਰਕਾਸ਼ (*prakāsh)* is a noun having sense “light”. The equivalent in target language is ਚਾਨਣ (*cānaṇ)*. Thus, the sentence will be translated as ਚਾਨਣ ਸਿੰਘ ਸਾਡੇ ਇੱਥੇ ਪਧਾਰੇ । (*cānaṇ siṅgh sāḍē itthē padhārē)*. But actually it must be translated as ਪ੍ਰਕਾਸ਼ ਸਿੰਘ ਸਾਡੇ ਇੱਥੇ ਪਧਾਰੇ । (*prakāsh siṅgh sāḍē itthē padhār)ē*. The reason is straightforward that in this sentence प्रकाश (*prakāsh)* word is acting as proper noun and it must be transliterated and not translated.

A small module has been developed for locating such proper nouns where word under consideration is a surname. If it is found to be surname then the word previous to this word is transliterated. If in any case, the previous word has been translated, now it has been corrected by transliteration. This module was also tested on a large Hindi corpus and showed that about 2-5 % text of the input text depending upon its domain is proper noun. Thus, this module plays an important role in translation. But it has also been observed that there were some cases where this module fails on following examples:

(i) आप कुमार से पूछ लें । (*āp kumār sē pūch lēṃ).*

(ii) उन्होंने सिंह परिवार से रिश्ता जोड़ा । (*unhōnnē siṃh parivār sē rishtā jōṛā).*

(iii) मैंने गोयल को कहा था कि वो यहाँ ना आये । (*mainnē gōyal kō kahā thā ki vō yahāṃ nā āyē)*

(iv) राम ने सिंह की क़ुरबानी को सराहा । (*rām nē siṃh kī kurbānī kō sarāhā).*

In the above examples, before the surnames कुमार (*kumār)*, सिंह (*siṃh)*, गोयल (*gōyal)* the tokens are आप (*āp)*, उन्होंने (*unhōnnē)*, मैंने (*mainnē)* respectively.

These token were transliterated rather than translated according to this module. Now, this module has been made intelligent to differentiate between proper nouns and other tokens like pronouns, prepositions, adjectives etc and thus only proper nouns will be transliterated. List of such approx. 50 tokens has been prepared manually so that whenever these tokens are found before the surnames, these must not be transliterated and will be translated.

It is not possible to store all the possible proper nouns directly into the database. Thus, the proper nouns found from the input text through this module are automatically added to the proper nouns gazeteer. Hence, through this self learning approach, the system's accuracy and speed keep on increasing with use.

Database design: Table 5.3 carries the design of the database used for storing information about surnames.

Table 5.3: Surnames Database Design

Field Name	**Description**
surnameInHindi	Stores the surname in Hindi
surnameInPunjabi	Stores the corresponding transliterated surnames in Punjabi

Sample database entries:

The surnames database consists of 654 entries. Following table shows some of the database entries for surnames database:

Table 5.4: Sample Entries of Surname Database

surnameInHindi	surnameInPunjabi
अरोड़ा (*arōṛā)*	ਅਰੋੜਾ (*arōṛā)*
कक्ड़ (*kakṛ)*	ਕੱਕੜ (*kakkaṛ)*
खुराना (*khurānā)*	ਖੁਰਾਨਾ (*khurānā)*
जिंदल (*jindal)*	ਜਿੰਦਲ (*jindal)*

5.2.3 Word-to-Word translation using lexicon lookup

If token is not a title or a surname, it is looked up in the HPDictionary database containing Hindi to Punjabi direct word to word translation. If it is found, it is used for translation. If no entry is found in HPDictionary database, it is sent to next sub phase for processing. For example, token is अड़तीसवाँ (*aṛtīsvāṃ)*, it is looked up in the database and the entry for it is found in the database. Then its translated version is used in the output text i.e. ਅਠੱਤੀਵਾਂ (*aṭhttīvāṃ)*. And no other phase is required for this token. Tokenizer will start generating next token for processing by the translation engine.

Database design: Table 5.5 carries the design of the database used for storing entries for Hindi words to Punjabi words direct translation.

Table 5.5: HPDictionary Database Design

Field Name	Description
hindiWord	Stores the Hindi Word

punjabiWord	Stores the corresponding translated word in Punjabi

Sample database entries:

The HPDictionary database consists of 54,127 entries.Following table shows some of the database entries for HPDictionary database:

Table 5.6: Sample Entries of HPDictionary Database

hindiWord	**punjabiWord**
अथवा (*athvā)*	ਅਤੇ (*atē)*
छोड़ी (*chōḍaī)*	ਛੱਡੀ (*chaḍḍī)*
जायेंगे (*jāyēṅgē)*	ਜਾਣਗੇ (*jāṇgē)*
सीखा (*sīkhā)*	ਸਿੱਖਿਆ (*sikkhiā)*

This database can be extended at any time to allow new entries in the dictionary to be added.

5.2.4 Resolving Ambiguity

Ambiguity is one of the NLP problems which have been a great challenge for computational linguists. In general, people are unaware of the ambiguities in the language they use because they are very good at resolving them using context and their knowledge of the world. But computer systems do not have this knowledge, and consequently do not do a good job of making use of the context.

Something is ambiguous when it can be understood in two or more possible ways or when it has more than one meaning. If the ambiguity is in a

sentence or clause, it is called structural (syntactic) ambiguity. If it is in a single word, it is called lexical ambiguity.

For the structural ambiguity, consider the sentence “The man saw the girl with the telescope”. This sentence is ambiguous since it can be interpreted in two ways: The man saw the girl who possessed the telescope or, the man saw the girl with the aid of the telescope. However, the sentence “The man saw the girl with a red hat” is not ambiguous for a human reader (people have the knowledge that a hat cannot be used to see), while it has the same ambiguity as the previous example for a computer.

In a Machine Translation application, different senses of a word may be represented with different words in the target language. Consider the following sentence :

राम आम खा रहा है । *(rām ām khā rahā hai)*

In the above example, word आम in the sentence is lexically ambiguous. Its meaning can be interpreted in two ways – mango (a fruit) and usual (an adjective) as in following examples:

Usage as Noun: तोता पेड़ पर बैठकर आम खा रहा है (*tōtā pēḍa par baiṭhakar ām khā rahā hai)*

Usage as Adjective: ऐसे चोरों से मिलना आम बात है जो चोरी के ख़िलाफ़ उपदेश देते हैं (*aisē cōrōṃ sē milnā ām bāt hai jō cōrī kē khailāpha updēsh dētē haiṃ)*

In order to correctly translate a text in one language to another, firstly we have to know the senses of the words and then find the best translation equivalent in the target language.

Lexical ambiguity can refer to both homonymy and polysemy. Homonyms are words that are written the same way, but are (historically or conceptually) really two different words with different meanings which seem unrelated. Examples are *suit* ("lawsuit" and "set of garments") and *bank* ("river bank" and "financial institution"). If a word's meanings are related, it is called a polyseme. The word *party* is polysemous because its senses can be generalized as "group of people", that is they are related.

Now let us consider the meaning of the noun *party* in the following sentence:

Mr. Smith's party took 38% of the votes in the last election.

It is clear to a human reader that the noun *party* is in the sense "an organization to gain political power" in the above sentence. Most people are not even aware of the ambiguity contained in the sentence. Humans are so skilled at resolving potential ambiguities that they do not realize they are doing it. There has been research on how people resolve ambiguities; however we still do not know exactly how humans do lexical disambiguation. Therefore, it is a difficult task to teach a computer to do the same thing. The most prominent way to disambiguate a word is examining its context. The context

can be considered as the words surrounding the ambiguous word, which is the noun *party* in our case. Words as *vote* and *election* might be a good clue for the sense of the noun *party*. But context is not the only information available for disambiguation. Syntactic classes of the words in the ambiguous word's context (whether they are noun, verb or adjective, etc.), whether the ambiguous word plays the role of object or subject in the syntactic structure of the sentence may also be used in the disambiguation process.

In our research problem, we have determined the correct meaning of an ambiguous word which comes across during translation process, namely Word Sense Disambiguation using the context information.

WSD algorithms can be divided into two based on the corpora used for training. These approaches are:

i. Supervised Word Sense Disambiguation

ii. Unsupervised Word Sense Disambiguation

In supervised WSD the training data is sense-tagged whereas in unsupervised WSD the training data is raw corpora which have not been semantically disambiguated. In the following sections these approaches will be explained in detail.

Supervised Disambiguation

Supervised disambiguation is an application of the supervised learning approach for creating a classifier. A disambiguated corpus where each occurrence of an ambiguous word is annotated with a contextually appropriate

sense is available for training. The aim in supervised disambiguation is to build a classifier which correctly classifies new cases based on their context of use.

Machine learning algorithms such as Bayesian classifiers [118], decision lists [119], decision trees [120], k-nearest neighbor and neural networks [121] are examples of supervised learning algorithms.

An example of probabilistic algorithms is Naïve Bayes [122] which has been frequently applied in WSD with good results [123]. Gale, Church and Yarowsky [124,125] uses a variant of Bayes ratio on six ambiguous nouns, namely *drug*, *duty*, *land*, *language*, *position*, and *sentence*, and reports 90% accuracy in discriminating between two senses of these words. Mooney [126] reports that Naïve Bayes and neural networks achieved the highest performance with an accuracy of 73% in assigning the correct senses to a corpus of examples of word *line* which has six senses. The other algorithms in Mooney's survey were 3-nearest neighbors, perceptron, decision tree, decision list and logic programming variants. Combining various classifiers has also been tested. Florian et al. [127] combined four classifiers namely feature-enhanced Naïve Bayes, Cosine, bag-of-words Naïve Bayes and non-hierarchical decision lists.

Decision lists search for discriminatory features in the training corpus and build a set of rules for disambiguation. Yarowsky [128] makes use of hierarchical decision lists and achieves top performance in the SENSEVAL-1 framework on the 36 test words for which tagged training data was available.

Agirre and Martinez [129] reports that decision lists provide state-of-the-art results with simple and very fast means. This approach is reported to learn with low amounts of data.

Decision lists and Bayesian classifiers are the most popular algorithms in supervised disambiguation. For neural networks Towell and Voorhees [130], for decision trees Black [131] and Pedersen [132], for k-nearest neighbor Ng and Lee [133] and for information-theoretic approaches Brown et al. [134] are some examples of the work done on WSD.

A major problem with supervised approaches is the need for a large sense-tagged training set. Despite the availability of large corpora, manually sense-tagging of a corpus is very difficult and very few sense-tagged data are available now.

The two largest corpora that are available are the SemCor corpus [135] and the SENSEVAL corpus [136-138]. The SemCor corpus, created by the Princeton University, is a subset of the English Brown corpus containing almost 700,000 running words. In SemCor, all the words are tagged by part of speech and more than 200,000 content words are also lemmatized and sense-tagged according to Princeton WordNet 1.6 (mappings for later versions of WordNet are also available). SENSEVAL corpus is derived from the HECTOR corpus and dictionary project. It is a joint Oxford University Press and Digital project which took place in the early 1990s. Another sense-tagged corpus available is the DSO Corpus of Sense-Tagged English (Ng and Lee, 1996) [133]. This corpus contains sense-tagged word occurrences for

121 nouns and 70 verbs which are among the most frequently occurring and ambiguous words in English. These occurrences are provided in about 192,800 sentences taken from the Brown Corpus and the Wall Street Journal and have been hand tagged by students at the Linguistics Program of the National University of Singapore. WordNet 1.5 sense definitions of these nouns and verbs were used to identify a word sense for each occurrence of each word.

There have been several efforts for finding a way to avoid the use of hand-tagged data. Bootstrapping is the most frequently used method for this purpose. Bootstrapping relies on a small number of instances of each sense for each lexeme of interest. These sense-tagged instances are used as seeds to train an initial classifier. This initial classifier is then used to extract a larger training set from the remaining untagged corpus. With each iteration of this process, the training corpus grows and the untagged corpus shrinks.

Hearst [139] generates a seed set by simply hand-tagging a small set of examples from the untagged corpus. However, during the training phase each occurrence of a set of nouns to be disambiguated is manually sense-tagged in several occurrences. Schütze [140-141] proposes a method that avoids tagging each occurrence in the training corpus. Yarowsky [142] proposes an alternative technique by using two constraints named as “One sense per collocation” and “One sense per discourse” and reports an accuracy of 96% on twelve words. “One sense per collocation” argues that nearby words provide strong and consistent clues to the sense of a target word, conditional

on relative distance, order and syntactic relationship. Also, “One sense per discourse” constraint argues that the sense of a target word is highly consistent within any given document. Different bootstrapping techniques are also presented in Mihalcea and Moldovan [143] and Mihalcea [144]. Mihalcea [144] makes a comparison between the results when training is performed on hand-tagged data and the results when training is done using the generated corpus by bootstrapping. She reports that the precision achieved with the generated corpus is comparable, and sometimes better than the precision achieved with hand-tagged corpora.

Another method for avoiding hand-tagged data is using parallel corpora [145]. In this method, bilingual corpora are used since different senses of some words translate differently in another language. By using a parallel aligned corpus, the translation of each occurrence of such words can be used to determine their correct senses automatically. In Dagan and Itai [146], Ide et al. [147] and Ng et al. [148], various uses of parallel corpora for WSD and its disadvantages can be found.

The main problem that supervised disambiguation methods face with is data sparseness. Since the sense-tagged training corpus is finite and very small for WSD, some senses of polysemous words are very likely to be missing and most of them have few examples. For a supervised algorithm to be successful, the training data must ensure that all senses of a polysemous word are covered. Smoothing is used to solve the data sparseness problem. The task of reevaluating some of the zero-probabilities or low-probabilities

and assigning them non-zero values is called smoothing. Some of the smoothing methods are add-one smoothing, Witten-Bell smoothing [149], and Good-Turing smoothing [150]. Gale [151], presented a Good-Turing method for estimating the probabilities of seen and unseen objects in linguistic applications named as Simple Good-Turing method.

Unsupervised Disambiguation

In machine learning the distinction between supervised and unsupervised algorithms rests on whether a set of classifications exists. In unsupervised word sense disambiguation, information is gathered from raw corpora which have not been semantically disambiguated.

Yarowsky [152] proposed an approach for marking words with their categories from a thesaurus. He used Roget's Thesaurus [153]. Training was carried out on an untagged corpus of 10 million words obtained from the electronic version of the Grollier's Encyclopedia. The important aspect of the approach was that he used a context of 50 words either side so that 100 words were considered in the training examples for each ambiguous word. This method was tested on 12 ambiguous words and reported to achieve 92% accuracy. Yarowsky notes that this method is best for extracting topical information, most successful for nouns. The algorithm presented in Yarowsky [142] is also an unsupervised algorithm making use of a bootstrapping procedure.

McCarthy et al. [154], presents an algorithm that makes use of a thesaurus acquired from raw textual corpora and the WordNet similarity package to find predominant noun senses automatically. The acquired predominant senses

gave a precision of 64% on the nouns of the SENSEVAL-2 all-words task which is a promising result regarding that no hand-tagged data is used.

Some of the unsupervised methods correspond to clustering tasks rather than sense tagging tasks because they do not label words to predefined senses. These algorithms do not make use of an outside source of knowledge to define senses. This is called *Word Sense Discrimination* rather than disambiguation. They divide the occurrences of a word into a number of classes by determining for any two occurrences whether they belong to the same sense or not [155-157]. Schütze's [155] results indicate that for coarse binary distinctions, unsupervised techniques can achieve results approaching those of supervised and bootstrapping methods. Purandere and Pedersen [156] present a systematic comparison of discrimination techniques proposed by Pedersen and Bruce [156,158,159] and by Schütze [157].

Knowledge Bases for WSD

In this section, different kinds of knowledge bases are presented. These knowledge bases can be used in any WSD system, whether it is supervised or unsupervised.

Machine Readable Dictionaries

Machine readable dictionaries (MRD) provide a ready-made information source of word senses. The first attempt to use MRD's came from Lesk (1986)[160]. He starts from the simple idea that a word's dictionary definitions are likely to be good indicators of the senses they define. By using Oxford Advanced Learner's Dictionary (OALD), he counts overlapping content words

in the sense definitions of the ambiguous word and in the definitions of context words occurring nearby and selects the sense that achieves the maximum number of overlaps. The accuracy of the method is reported to be 50-70% on short samples of the Jane Austen novel Pride and Prejudice and an Associated Press news story based on very brief experimentation with the program.

Cowie et al. [161] tried to improve Lesk's approach by optimizing the overlap of all words in a single sentence simultaneously. However, it was found computationally very expensive. Therefore, Cowie et al. [161] used simulated annealing[162] for the first time in natural language processing. They evaluated this approach using a corpus consisting of 50 example sentences taken from Longman Dictionary of Contemporary English (LDOCE) which were disambiguated by hand. 47% of the words were reported to be correctly disambiguated to sense level and 72% to more rough grained senses.

Stevenson and Wilks [163] computed the overlap by normalizing the contribution of a word to the overlap count. Pedersen and Banerjee [164] described a different version of the Lesk's algorithm by employing glosses contained in WordNet [165]. Because of the fact that dictionaries are created for human use, not for computers, there are some inconsistencies [166-168]. Although they provide detailed information at the lexical level, they lack pragmatic information used for sense determination. For instance, the relation between *ash* and *tobacco*, *cigarette* or *tray* is very indirect in a dictionary

whereas the word *ash* co-occurs very frequently with these words in a corpus [169].

Thesauri

Thesauri provide information about relationships among words. Thesaurus based disambiguation makes use of the semantic categorization provided by a thesaurus or a dictionary with subject categories. The most frequently used thesaurus in WSD is Roget's International Thesaurus (Roget, 1946) which was put into machine-tractable form in 1950's [153].

Walker [170] proposed an algorithm as follows: each word is assigned to one or more subject categories in the Thesaurus. If the word is assigned to several subjects, then it is assumed that they correspond to different senses of the word.

Similar to machine readable dictionaries, a Thesaurus is a resource for humans, so there is not enough information about word relations.

Computational Lexicons

The usefulness of lexical relations in linguistic, psycholinguistic and computational research has led to a number of efforts to create large electronic databases of such relations. Beginning from the mid-1980's, construction of semantic lexicons by hand has emerged. Some examples of these lexicons are WordNet [164], CyC [171], ACQUILEX [172], and COMLEX [173]. Each of these lexicons contains different kinds of information.

WordNet

WordNet is an online lexical reference system which was developed at Princeton University under the direction of Professor George A. Miller. It combines many features used for WSD in one system. It includes definitions of word senses as in a dictionary; it defines “synsets” of synonymous words representing a single lexical concept; and it includes word-to-word relations.

WordNet consists of three databases: noun database, verb database and one database for adjectives and adverbs. Each database consists of lexical entries corresponding to unique orthographic forms.

The earliest attempts to use WordNet in WSD were in information retrieval field. Voorhees [174] and Richardson and Smeaton [175] created knowledge bases using WordNet’s hierarchy. Li et al. [176] proposed a WordNet-based algorithm for WSD. Disambiguation was done by semantic similarity between words and heuristic rules. Heuristic rules were based on the semantic similarity and the WordNet hierarchy. Leacock et al. [177] used WordNet to counter data sparseness problem. Hawkins [178] built up a WSD system that works with frequency and contextual information that is based on WordNet. Fellbaum et al. [179] proposed a system that made use of syntactic clustering and semantic distinctions extracted from WordNet.

WordNet is mostly used to determine semantic similarity between senses. Resnik [180] computed information content of words which is a measure of the specificity of the concept that subsumes the words in the WordNet hypernym hierarchy. Agirre and Rigau [181] employ WordNet to determine the

conceptual distance among concepts whereas Mihalcea and Moldovan [182] exploit semantic density and WordNet glosses in an all words word sense disambiguation. Lin [183-185] described a semantic similarity measure where similarity between two objects is defined to be the amount of information contained in the commonality between the objects divided by the amount of information in the description of the objects.

Other approaches using WordNet are Jiang [186], Agirre and Agirre et al. [187], Haynes [188] and Banerjee et al. [189]. A combination of MRDs and WordNet has also been tried with some success [190-192].

WSD in Indian Languages

Robust Standalone Systems for word sense disambiguation in Indian language are very few. WSD is mostly tackled at the POS tagging and Morphological analysis phase and what ever left is handled with the help of rules. Recently some standalone algorithms have also been developed for WSD in Indian languages.

Anusaaraka is one of the oldest MT systems available in India. It is more a language accessor rather than an MT system [65]. It is based on the assumption that most Indian languages have same origin so most of the words in source language have one meaning in target language. Based on this, it just provides the glosses of source language in the target language. There are cases where the meaning is too general or too specific. Such cases are handled by introducing some special notation to either narrow down or

widen the meaning. An attempt is made to find the underlying thread that connects different senses of the polysemous word. A kind of formula is then evolved that faithfully and unambiguously represents the connection between these different senses. For the English – Hindi system, the current version of Anusaaraka uses a dictionary called Shabdanjali. POS tagger and wasp workbench are used for developing word sense disambiguation rules semi-automatically.

Similarly in AnglaBharati approach a rule base is used for picking up the correct sense of each word in the source language to the extent feasible using interleaved semantic interpreter [65]. Further disambiguation and choice of right construct and lexical preference are generated by the target language text generator module. Many a time, multiple rules may get invoked leading to the multiple interpretation of the input sentence. The rules are ordered in terms of their preference and an upper limit is put on the number of alternatives produced. Most of the disambiguation rules are in the form of syntacto-semantic constraints. Semantics are used to resolve most of the intra-sentence anaphora/pronoun references. Alternative meanings for the unresolved ambiguities are retained in the pseudo target language. The lexical database is hierarchically organized to allow domain specific meanings and also prioritize meanings as per user requirement.

In the example based approach developed by [66] and known as ANUBHARTI, ambiguities in the meaning of the verb phrasal are also resolved using an appropriate distance function in the example base. The

alternate translations are being ranked with respect to the ordering of the rule base.

In ANUBAAD system, sense disambiguation is carried out at various levels [75]. It starts with POS of a word. Some semantic categories are associated with words to identify the inflections to be attached with corresponding words in Indian languages as well as to identify the context in the sentence. Context identification is also done by the recognition of idiomatic expression and using context templates for each word. The context templates have been designed on the basis that meaning of the word may be independent of the context, may depend upon the occurrence of a sequence of words or words with certain semantic categories or may depend on the occurrence of certain keywords or keyword with certain semantic category.

In Matra, rule bases and heuristic approaches are used for word sense disambiguation. A method has been described by Durgesh Rao et. al.[1] for mapping prepositions from English to Hindi. Similarly in Saarthak, emphasis is on sentence-level word sense disambiguation, which makes it different from general statistical techniques that use contextual information for the same. At AU-KBC research centre, S. Baskaran [193] presents an approach in which all the occurrences of the ambiguous words are classified into different clusters in such a way that all the occurrences are in the same sense within a cluster. Development of a Prototype of a Frame-based System for the Understanding of Malayalam Language has been carried out by Sumam M. Idicula and David Peter S [194]. In this system, three types of information are

used for word sense disambiguation. They are local word grouping (grouping of words which can collectively perform a syntactic role in a sentence), syntactic information and semantic tags. Prabhakar Pandey et. al. [195] makes use of the Wordnet for Hindi developed at IIT Bombay, for WSD. The accuracy values are reported to be in range from about 40% to about 70%. The system currently deals with only nouns. Ganesh Ramakrishnan [196] introduces the notion of soft word sense disambiguation which states that given a word, the sense disambiguation system should not commit to a particular sense, but rather, to a set of senses which are not necessarily orthogonal or mutually exclusive.

Information Sources for WSD

There are various information sources or feature types used in WSD regardless of the type of the approach. To disambiguate a word, various kinds of information, including syntactic tags, word frequencies, collocations, semantic context, role-related expectations, and syntactic restrictions can be considered.

In Agirre and Martinez [197], a comparison of WSD systems has been made based on the information source they used. Some of these sources are as follows:

Frequency of Senses: Frequency information is used to measure the likelihood of each possible sense appearing in the text. Therefore this information is generally used in statistical approaches and it is generally

learned from hand-tagged data such as SemCor corpus. Interestingly very few WSD approaches outperform the “most frequent sense” heuristic. WordNet senses are ordered according to the frequencies of the senses in the SemCor corpus.

Part of Speech (POS): Part of speech tagging is regarded as the first step of the disambiguation process if the lemmas have the same orthographic forms but different syntactic classes. It is useful because it reduces the number of possible senses a word can belong to. An orthographic form may even be unambiguous in one syntactic class whereas it has more than one sense in another. For instance, in WordNet 2.0 *handle* has 5 senses as a verb, but only one sense as a noun. The impact of knowledge resources on WSD in examined in Gaustad [198]. The results show that accurate POS information is beneficial for WSD and that including the POS of the ambiguous word itself as well as POS of the context increases the disambiguation accuracy.

Morphology: It is defined as the relation between derived words and their roots. For instance, the noun *agreement* has 6 senses, its verbal root *agree* 7. A stemmer tries to reduce various forms of a word to a single stem. Since English is a language with little inflectional morphology, it is not certain that using morphology will lead to significant improvements in WSD. With other languages, such as German or Italian, morphology is of greater influence.

Collocations: Collocation is the relationship among any group of words that tend to co-occur in a predictable configuration. Disambiguation relies heavily on collocational information. For example, the noun *match* has 9

senses. However, it has only one possible sense in "*football match*". It is observed that collocations are strong indicators if they are learned from hand-tagged corpora. Although they are strong, they should be used with other sources. They should not be treated as rules for sense-filtering alone [199,200].

Semantic word associations: These can be classified as follows:

i. *Taxonomial organization:* This refers to the classification of words in a hierarchy and the lexical-semantic relationships holding between words such as a *dog* is a kind of *animal*. This kind of information can be extracted from ontologies like WordNet.

ii. *Situation and Topic:* Information about the situation or topic enables a WSD system to see the ambiguous word in a broader context. For example, if the word *mouse* is used in an office situation and the topic is computer use, the most probable sense of the word *mouse* will be "computer tool", not "animal". Semantic word associations around topic and situation are powerful when learned from hand-tagged corpora. Associations learned from MRDs can also be useful.

iii. *Argument-head relations:* These relations provide important clues for disambiguation such as the relationship between *dog* and *bite* in the sentence "the dog bit the man."

Subcategorization information: Subcategorization refers to certain kinds of relations between words and phrases. For example the verb *want* can be

followed by an infinitive, as in *"I want to fly to Istanbul"*, or a noun phrase, as in *"I want a flight to Istanbul"*. But the verb find cannot be followed by an infinitive. For example *"I found to fly to Istanbul."* is not a correct sentence. Verbs have several possible patterns of arguments. A particular set of arguments that a verb can appear with is referred to a subcategorization frame. Subcategorization frames capture syntactic regularities about complements.

Agirre and Martinez [187] made a comparison between the contributions of the above resources to WSD. According to their observations, if learned from hand-tagged corpora, collocations and semantic word associations are the most important knowledge types for WSD, but they also mentioned that syntactic cues are equally reliable. On the other hand, taxonomical information was found to be very weak.

Our Approach

While dealing with related languages like Hindi and Punjabi, structural ambiguity is not a problem at all because the ambiguity in the source sentence is transferred to the target sentence without affecting the underlying meaning. We are not claiming that there is no structural ambiguity in the Hindi language that do not carry over as such in Punjabi language, but we did not come across with any. So, structural ambiguity has not been touched in this research work. To start with, all we have is a raw corpus of Hindi text. So the N-Gram statistical approach is the obvious choice for our purpose. The

following section provides the theory of N-Gram approach and our approach for WSD.

N-Gram Approach:

An *n-gram* is simply a sequence of successive n words along with their count i.e. number of occurrences in training data [201,202]. An *n-gram* of size 2 is a bigram; size 3 is a trigram; and size 4 or more is simply called an *n-gram* or (n − 1)-order Markov model. An *n-gram* model models sequences of natural languages using the statistical properties of *n-gram*s. More concisely, an *n-gram* model predicts x_i based on x_{i-1}, x_{i-2}, ..., x_{i-n}. *n-gram*s models are widely used in statistical natural language processing.

The number of words in the local context of ambiguous word makes a window. The size of this window i.e. the value of N depends on various factors.

a) Larger the value of n, higher is the probability of getting correct word sense i.e. for the general domain; more training data will always improve the result. But on the other hand most of the higher order *n-gram*s do not occur in training data. This is the problem of sparseness of data.
b) As training data size increases, the size of model also increases which can lead to models that are too large for practical use. The total number of potential *n-gram*s scales exponentially with n. Computer up to present could not calculate for a large n because it requires huge amount of memory space and time.

c) Does the model get much better if we use a longer word history for modeling an *n-gram*?

d) Do we have enough data to estimate the probabilities for the longer history?

Claude E. Shannon [203] established the information theory for finding the value of n in 1951. This theory included the concept that a language could be approximated by an nth order Markov model by n to be extended to infinity. Shannon computed the per letter entropy rather than per word entropy. He gives entropy of English text as 1.3 bits per letter. Since his proposal there were many trials to calculate *n-gram*s for a big text data of a language. Brown et. al.[204] performs a test on much larger text and give an upper bound of 1.75 bits per character for English language by using trigram model. Iyer et al. [205] investigate the prediction of speech recognition performance for language model in the switchboard domain, for trigram model built on different amounts of in domain and out of domain training data. Over the ten models they constructed, they find that perplexity predicts word error rate well when only in domain training data is used, but poorly when out of domain text is added. They find that trigram coverage or the fraction of trigram in the test data present in training data is a better predictor of word error rate than perplexity.

Chen et al. [206] investigate their language model for speech recognition performance in the Broadcast news domain and concluded that perplexity

correlates with word error rate remarkably well when only considering *n-gram* model trained on in domain data.

Manin [207] performs a study on predictability of word in context and found that unpredictability of a word depends upon the word length. Marti et. al. [208] tested different vocabulary size and concluded that language models become more powerful in recognition tasks with larger vocabulary size. Resnik et. al. [209-210] made several observations about the state of the art in automatic word sense disambiguation and offer several specific proposals to the community regarding improved evaluation criteria, common training and testing resources, and the definitions of sense inventories.

While Kaplan [211] Choueka and Lusignan [212], based on the observation that people don't seem to need very much context, claims that only 5 words to the left and 5 words to the right of the polysemous word are sufficient for WSD but William A. Gale et. al. [213] use a very wide context, 100-words surrounding the polysemous word in question. They find that there are often very useful clues even quite far away from the polysemous word in question. They demonstrated that information is *measurable* out to 10,000 words away from the polysemous word. They also observed that although contextual clues are measurable at surprisingly large distances, much of this information might not be very useful. In particular, it might have been possible to find the same information at smaller distances. In their words:

"The contribution is largest, not surprisingly, for smaller *d,* but nevertheless, the contribution continues to grow out to at least twenty words,

perhaps fifty words, well beyond the ±6 word contexts typically found in many disambiguation studies. Increasing the context from ±6 words to ±50 words improves performance from 86% to 90%."

Among number of approaches for disambiguation, the most appropriate approach to determine the correct meaning of a Hindi word in a particular usage for our Machine Translation system is to examine its context using N-gram approach. After analyzing the past experiences of various authors explained above, we have chosen the value of n to be 3 and 2 i.e. trigram and bigram approaches respectively for our system. Trigrams are further categorized into three different types. First category of trigram consists of context one word previous to and one word next to the ambiguous word. Second category of trigram consists of context of two adjacent previous words to the ambiguous word. Third category of the trigram consists of context of two adjacent next words to the ambiguous word. Bigrams are also categorized into two categories. First category of the bigrams consists of context of one previous word to ambiguous word and second category of the bigrams consists of one context word next to ambiguous word. The disambiguation algorithm starts with the look up in the trigrams databases. All the three trigrams databases are looked up for the some entry corresponding to ambiguous word. If the entry is matched in more than one trigrams databases, the entry of the database with maximum frequency will be considered to be the best match. If the entry is matched only in any one of the three trigrams databases, then that entry is used regardless of the frequency.

If in case, no trigrams database is able to disambiguate the word, then bigram databases are used for disambiguation. Entry is looked up in both of the bigrams databases and if found in both the databases, the entry with maximum frequency will be considered. If the entry is found only in any one of the bigram databases, then that entry is used for disambiguation. In the worst case, if no entry is matched from any of entries in both trigrams and bigrams databases, then this word is assumed to be unknown and out-of-vocabulary module will handle such words. For this purpose, the Hindi corpus consisting of about 2 million words was collected from different sources like online newspaper daily news, blogs, Prem Chand stories, Yashwant jain stories, articles etc. The most common list of ambiguous words was found. We have found a list of 75 ambiguous words out of which the most frequent are से *sē* and और *aur*. Following table shows a summary of different lexical categories for these ambiguous words:

Table 5.7: Lexical Category % distribution of ambiguous words

S.No.	Lexical Category	Ambiguos Words
1.	Noun	38%
2.	Verb	13.5%
3.	Adjective	6.5%
4.	Adverb	1.38%
5.	Postposition	1.38%
6.	Noun and Verb	8.17%

7.	Noun and Adjective	21.39%
8.	Adjective and Conjunction	1.38%
9.	Adjective and Verb	2.76%
10.	Noun,Postposition and Conjuction	1.38%
11.	Noun,adjective and adverb	4.16%

Through Corpus analysis and taking this ambiguous word list as base, above mentioned three types of trigrams databases (one word to the left and one word to the right of the ambiguous word, two consecutive words to the left of the ambiguous word, two consecutive words to right of the ambiguous words) and bigrams (one word previous to the ambiguous word, one word to the right of the ambiguous word) were generated from the corpus along with their frequency in the corpus and stored into the databases trigramsMiddle, trigramsLeft, trigramsRight, bigramsLeft and bigramsRight respectively. On analysis, it has been found that in Hindi language, most common words that are ambiguous are post positions like से (*sē)*, पर (*par)* etc and the conjunction

और (*aur)*.

Consider the postposition 'से' that can be translated most commonly into ਤੋਂ , ਨੂੰ ,

ਜਿਹੇ , ਕਰਕੇ and ਨਾਲ . Let us take the example:

मैंने राम से पुछा आप कहाँ जा रहे हो । (*mainnē rām sē puchā āp kahāṃ jā rahē hō)*

It is the task of word sense disambiguator module to find the appropriate meaning/sense for the ambiguous word 'से'. The algorithm starts with collecting the words surrounding the ambiguous word 'से' in the sentence. Thus, it forms following three context bags with window size 3 in which Context Bag 1 contains two context words previous to the ambiguous word से *sē*, Context Bag 2 contains one context word previous to and one context word next to the ambiguous word से and Context Bag 3 contains two context words next to the ambiguous word से *sē*.

Context Bag 1: (मैंने) (राम) से

Context Bag 2: (राम) से (पुछा)

Context Bag 3: से (पुछा) (आप)

Then the algorithm searches the entry match for Context Bag1, Context Bag2 and Context Bag3 in trigrams databases - trigramsRight, trigramsMiddle,and trigramsLeft respectively. The entries are found in different trigrams databases as shown below:

Table 5.8: Example demonstrating the ambiguity resolution

Database	Context Bag	PunjabiMeaning	Frequency

trigramsMiddle	(राम) से (पुछा)	ਨੂੰ	1209
trigramsRight	(मैंने) (राम) से	ਨਾਲ	845
trigramsLeft	से (पुछा) (आप)	ਤੋਂ	286

Above table shows that, the Punjabi meanings for all the three entries differ, hence the entry of the database with maximum frequency among three, will be used for disambiguation i.e. the entry with frequency 1209 will be used. Reason for using the entry with maximum frequency among these is to further find the most appropriate meaning for that ambiguous word.

Finally, the translated text in target language will be

ਮੈਂ ਰਾਮ ਨੂੰ ਪੁੱਛਿਆ ਤੁਸੀ ਕਿੱਥੇ ਜਾ ਰਹੇ ਹੋ । (*maiṃ rām nūṃ pucchiā tusī kitthē jā rahē hō)*

Database design: Table 5.9, Table 5.10, Table 5.11, Table 5.12 and Table 5.13 carries the design of the database used for storing entries for word sense disambiguation.

Table 5.9: triGramsMiddle Database Design

Field Name	**Description**
ambHindiWord	Stores ambiguous word in Hindi
previousWord1	Stores possible word previous to ambiguous word in text
nextWord1	Stores possible word next to ambiguous word in text
punjabiWord	Stores the correct translation for this ambiguous word for this instance
Freq	Stores the frequency of this trigram in the analyzed corpus

Table 5.10: triGramLeft Database Design

Field Name	Description
ambHindiWord	Stores ambiguous word in Hindi
previous1Word	Stores possible word previous to ambiguous word in text
previous2Word	Stores possible word previous to previous1Word mentioned above in the text
punjabiWord	Stores the correct translation for this ambiguous word for this instance
Freq	Stores the frequency of this trigram in the analyzed corpus

Table 5.11: triGramsRight Database Design

Field Name	Description
ambHindiWord	Stores ambiguous word in Hindi
next1Word	Stores possible word next to ambiguous word in text
next2Word	Stores possible word next to next1Word mentioned above in the text
punjabiWord	Stores the correct translation for this ambiguous word for this instance
Freq	Stores the frequency of this trigram in the analyzed corpus

Table 5.12: biGramsLeft Database Design

Field Name	Description
ambHindiWord	Stores ambiguous word in Hindi
previousWord	Stores possible word previous to ambiguous word in text
punjabiWord	Stores the correct translation for this ambiguous word for this instance
Freq	Stores the frequency of this bigram in the analyzed corpus

Table 5.13: biGramsRight Database Design

Field Name	Description
ambHindiWord	Stores ambiguous word in Hindi
nextWord	Stores possible word next to ambiguous word in text
punjabiWord	Stores the correct translation for this ambiguous word for this instance
Freq	Stores the frequency of this bigram in the analyzed corpus

Sample database entries:

There is no Hindi-Punjabi parallel corpus available and even no machine readable Hindi-Punjabi dictionary is available. Thus, there is no way to generate the data for word sense disambiguation databases automatically. For this purpose, we have developed a small module for finding the bigrams and trigrams for the ambiguous words from the Hindi Corpus. But the corresponding equivalent meaning in Punjabi for the ambiguous word based on its context is found manually and stored into the database for each entry. It is very time consuming and tedious task. The triGramMiddle database consists of 48,285 entries. The triGramLeft database consists of 46,735 entries. The triGramRight database consists of 49,217 entries. The biGramLeft database consists of 52,456 entries. The biGramLeft database consists of 51,129 entries.

Following tables show some of the database entries for fiveGrams, trigrams and biGrams databases:

Table 5.14: Sample Entries of triGramsMiddle Database

previous1Word	ambHindiWord	next1Word	punjabiWord	Freq
तरह (*tarah)*	से (*sē)*	समझा (*samjhā)*	ਨਾਲ (*nāl)*	5291
आराम (*ārām)*	से (*sē)*	कट (*kaṭ)*	ਨਾਲ(*nāl)*	1683
देर (*dēr)*	से (*sē)*	आना(*ānā)*	ਨਾਲ (*nāl)*	4720
कम (*kam)*	से (*sē)*	कम (*kam)*	ਤੋਂ (*tōṃ)*	3825
रखने (*rakhnē)*	से (*sē)*	उन्हें (*unhēṃ)*	ਨਾਲ (*nāl)*	853

Table 5.15: Sample Entries of triGramsLeft Database

previous1Word	previous2Word	ambHindiWord	punjabiWord	Freq
मोह (*mōh)*	माया (*māyā)*	और (*aur)*	ਅਤੇ (*atē)*	4138
अलावा (*alāvā)*	कोई (*kōī)*	और (*aur)*	ਹੋਰ (*hōr)*	2960
अंदर (*andar)*	ही (*hī)*	से (*sē)*	ਤੋਂ (*tōṃ)*	2105
टूट (*ṭūṭ)*	गया (*gayā)*	और (*aur)*	ਅਤੇ (*atē)*	3198

Table 5.16: Sample Entries of triGramsRight Database

ambHindiWord	next1Word	next2Word	punjabiWord	Freq
और (*aur)*	मोह (*mōh)*	माया (*māyā)*	ਅਤੇ (*atē)*	2418
से (*sē)*	समर्थन (*samrthan)*	वापस (*vāpas)*	ਤੋਂ (*tōṃ)*	3185
से (*sē)*	मुलाक़ात (*mulākāt)*	की (*kī)*	ਨਾਲ (*nāl)*	4190
कर (*kar)*	अंकित (*aṅkit)*	अपनी (*apnī)*	ਕਰ (*kar)*	964

Table 5.17: Sample Entries of biGramsLeft Database

previousWord	ambHindiWord	punjabiWord	Freq

मोह (*mōh)*	और (*aur)*	ਅਤੇ (*atē)*	1684
अलावा (*alāvā)*	और (*aur)*	ਹੋਰ (*hōr)*	1590
अंदर (*andar)*	से (*sē)*	ਤੋਂ (*tōṃ)*	3753

Table 5.18: Sample Entries of biGramsRight Database

ambHindiWord	nextWord	punjabiWord	Freq
से (*sē)*	एक (*ē)k*	ਤੋਂ (*tōṃ)*	1785
और (*aur)*	पुलिस (*pulis)*	ਹੋਰ (*hōr)*	1049
से (*sē)*	मुलाक़ात (*mulākāt)*	ਤੋਂ (*tōṃ)*	2916
और (*aur)*	अब (*ab)*	ਅਤੇ (*atē)*	2008

This database can be extended at any time to allow new entries in the dictionary to be added.

Analysis:

The analysis was done on a document of 100 pages consisting of 3,58,874 words. It was found that about 8.4% words were ambiguous among these. Out of these 8.4%, approximately 70% words were correctly disambiguated. Following table shows the contributions of various bigrams and trigrams databases mentioned above in disambiguating these words:

Table 5.19: Contribuation of various N-Grams in resolving ambiguity

S.No.	Table Name	Contribution
1.	bigramsRight	28.15%
2.	bigramsLeft	38.95%

3.	TrigramsMiddle	4.52%
4.	TrigramsRight	14.52%
5.	TrigramsLeft	13.86%

Thus, it is clear from above table that context of next two words for an ambiguous word helps the most in disambiguating the sense of the word.

5.2.5 Handling Unknown Words

5.2.5.1 Word Inflectional Analysis and generation

In linguistics, a suffix (also sometimes called a *postfix* or *ending*) is an affix which is placed after the stem of a word. Common examples are case endings, which indicate the grammatical case of nouns or adjectives, and verb endings. Hindi is a (relatively) free word-order and highly inflectional language. Following table shows the Hindi Suffix List:

Table 5.20: Inflections in Hindi

आ	आएं	अता	आने	एगा
इ	आओं	अती	ऊंगा	एगी
ई	इयां	ईं	ऊँगी	आएगा
उ	इयों	अतिं	आऊंगा	आएगी
ऊ	आइयां	अते	आऊंगी	आया
ए	आइयों	आता	एंगे	आए
ओ	आँ	आती	एंगी	आई
एं	इयाँ	आतीं	आएंगे	आईं
ओं	आइयाँ	आते	आएंगी*ा*	इए
आं	अताएं	अना	ओगे	आओ
उआं	अताओं	अनी	ओगी	आइए

उएं	अनाएं	अने	आओगे	अकर
उओं	अनाओं	आना	आओगी	आकर

A detailed analysis of noun, adjective, and verb inflections that were used to create this list can be found in McGregor [214] and Rao[215]. A few examples of each type are given below:

Noun Inflections: Nouns in Hindi are inflected based on the case (direct or oblique), the number (singular or plural), and the gender (masculine or feminine3). For example, लड़का (*aḍkā*) becomes लड़के (*laḍkē)* when in oblique case, and the plural लड़के (*laḍkē)* becomes लड़कों (*laṛkōṃ)*. The feminine noun लड़की (*laḍkī)* is inflected as लड़कियां (*laḍkiyāṃ)* and लड़कियों (*laḍkiyōṃ)*, but it remains uninflected in the singular direct case.

Adjective Inflections: Adjectives which end in आ (*ā)* or आं (*āṃ)* in their direct singular masculine form agree with the noun in gender, number, and case. For example, the singular direct अच्छा(*acchā)* is inflected as अच्छे(*acchē)* in all other masculine forms, and as अच्छी (*acchē)* in all feminine forms. Other adjectives are not inflected.

Verb Inflections: Hindi verbs are inflected based on gender, number, person, tense, aspect, modality, formality, and voice. Rao [215] provides a complete list of verb inflection rules.

Because of same origin, both languages have very similar structure and grammar. The difference is only in words and in pronunciation e.g. in Hindi it is लड़का (*laḍkā)* and in Punjabi the word for boy is ਮੁੰਡਾ (*muṇḍā)* and even sometimes that is also not there like घर (*ghar)* and ਘਰ (*ghar)*. The inflection forms of both these words in Hindi and Punjabi are also similar. In this activity, inflectional analysis without using morphology has been performed for all those tokens that are not processed in the previous activities of pre -processing and translation engine phases. Thus, for performing inflectional analysis, rule based approach has been followed. For this purpose, inflectional rules are also derived from the morphological analysis developed by IIIT, Hyderabad. This morphological analyzer works for Linux platform. First it was converted to work on Windows platform and then inflection rules were extracted from it for Hindi language. These rules were used for writing the rules for equivalent Punjabi inflections. These inflection rules resulted for Hindi to Punjabi translation purpose are implemented using regular expressions. The suffix separation module is based on the Hindi stemmer presented in Ananthakrishnan and Rao [216], and works by separating from each word the longest possible suffix Hindi Suffix List. When the token is passed to this sub phase for inflectional analysis, If any pattern of the regular expression (inflection rule) matches with this token, that rule is applied on the token and its equivalent translation in Punjabi is generated based on the matched rule(s). There is also a check on the generated word

for its correctness. We are using correct Punjabi words database for testing the correctness of the generated word. This generated Punjabi word is matched with some entry in punjabiUnigrams database. The database punjabiUnigrams is a collection of about 2,00,000 Punjabi words from large Punjabi corpus analysis. Punjabi corpus has been collected from various resources like online Punjabi newspapers, blogs, articles etc. If there is a match, the generated Punjabi word is considered a valid Punjabi word. If there is no match, this input token is forwarded to the transliteration activity.

The advantage of using punjabiUnigrams database is that ingenuine Punjabi words will not become the part of translation. If the wrong words are generated by inflectional analysis module, it will not be passed to translation rather it will be treated as out-of vocabulary and will be transliterated.

It has been analyzed that when this module was tested on the Hindi corpus of about 50,000 words, approx. 10,000 distinct words passed through this phase. And out of these 10,000 words, approx. 7,000 words were correctly generated and even accepted by Punjabi unigrams database. But rest was either generated wrong and was simply transliterated. Following table shows the correct accepted words generated by the inflectional analysis:

Table 5.21: List of Correct accepted words in translation after inflectional analysis and generation

रिवाजों (*rivājōṃ*)	ਰਿਵਾਜਾਂ (*rivājāṃ*)
समाजवादियों (*samājvādiyōṃ*)	ਸਮਾਜਵਾਦੀਆਂ (*samājvādīāṃ*)

धमकियाँ (*dhamkiyāṃ*)	ਧਮਕੀਆਂ (*dhamkīāṃ*)
उपलब्धियां (*uplabdhiyāṃ*)	ਉਪਲਬਧੀਆਂ (*uplabdhīāṃ*)
ताकतें (*tāktēṃ*)	ਤਾਕਤਾਂ (*tāktāṃ*)
जाओगी(*jāōgī*)	ਜਾਏਂਗੀ (*jāēṅgī*)
निकलेंगे (*niklēṅgē*)	ਨਿਕਲਣਗੇ (*niklaṇgē*)
सीमाओं (*sīmāōṃ*)	ਸੀਮਾਵਾਂ (*sīmāvāṃ*)
उठाएँ (*uṭhāēṃ*)	ਉਠਾਵਾਂ (*uṭhāvāṃ*)
बचाता (*bacātā*)	ਬਚਾਂਦਾ (*bacāndā*)
पहुँचते (*pahuñctē*)	ਪਹੁੰਚਦੇ (*pahuñcdē*)
दिखाइए (*dikhāiē*)	ਦਿਖਾਓ (*dikhāō*)
प्रतियोगियों (*pratiyōgiyōṃ*)	ਪ੍ਰਤੀਯੋਗੀਆਂ (*pratīyōgīāṃ*)
जाऊँगी (*jāūṅgī*)	ਜਾਵਾਂਗੀ (*jāvāṅgī*)
नवाजा (*navājā*)	ਨਵਾਜਿਆ (*navājiā*)
जलाकर (*jalākar*)	ਜਲਾਕੇ (*jalākē*)
धुआँ (*dhuāṃ*)	ਧੁਆਂ (*dhuāṃ*)
टटोलने (*ṭaṭōlnē*)	ਟਟੋਲਣ (*ṭaṭōlaṇ*)
आँका (*āṅkā*)	ਆਂਕਿਆ (*āṅkiā*)

Following table shows the failure cases through inflectional analysis:

Table 5.22: Failure cases during inflectional analysis and generation

Input Token	**Translation generated**

	by inflectional analysis
लडकियां (*laḍkiyāṃ*)	ਲਡਕਿਆਂ (*laḍkiāṃ*)
तारेगना (*tārēgnā*)	ਤਾਰੇਗਨਿਆ (*tārēgniā*)
शुजा (*shujā*)	ਸ਼ੁਜਿਆ (*shujiā*)
मुग्धा (*mugdhā*)	ਮੁਗਧਿਆ (*mugdhiā*)
किराना (*kirānā*)	ਕਿਰਾਨਿਆ (*kirāniā*)
उतरवा (*utravā*)	ਉਤਰਵਿਆ (*utraviā*)
तरेगना (*tarēgnā*)	ਤਰੇਗਨਿਆ (*tarēgniā*)
पटाखा (*paṭākhā*)	ਪਟਾਖਿਆ (*paṭākhiā*)

The above Punjabi words are not correct and are not present in the Punjabi language vocabulary. Thus, these words have not been passed by punjabiUnigrams database and thus rejected. This step helps in improving the accuracy of the translation system.

Following flow chart presents its working:

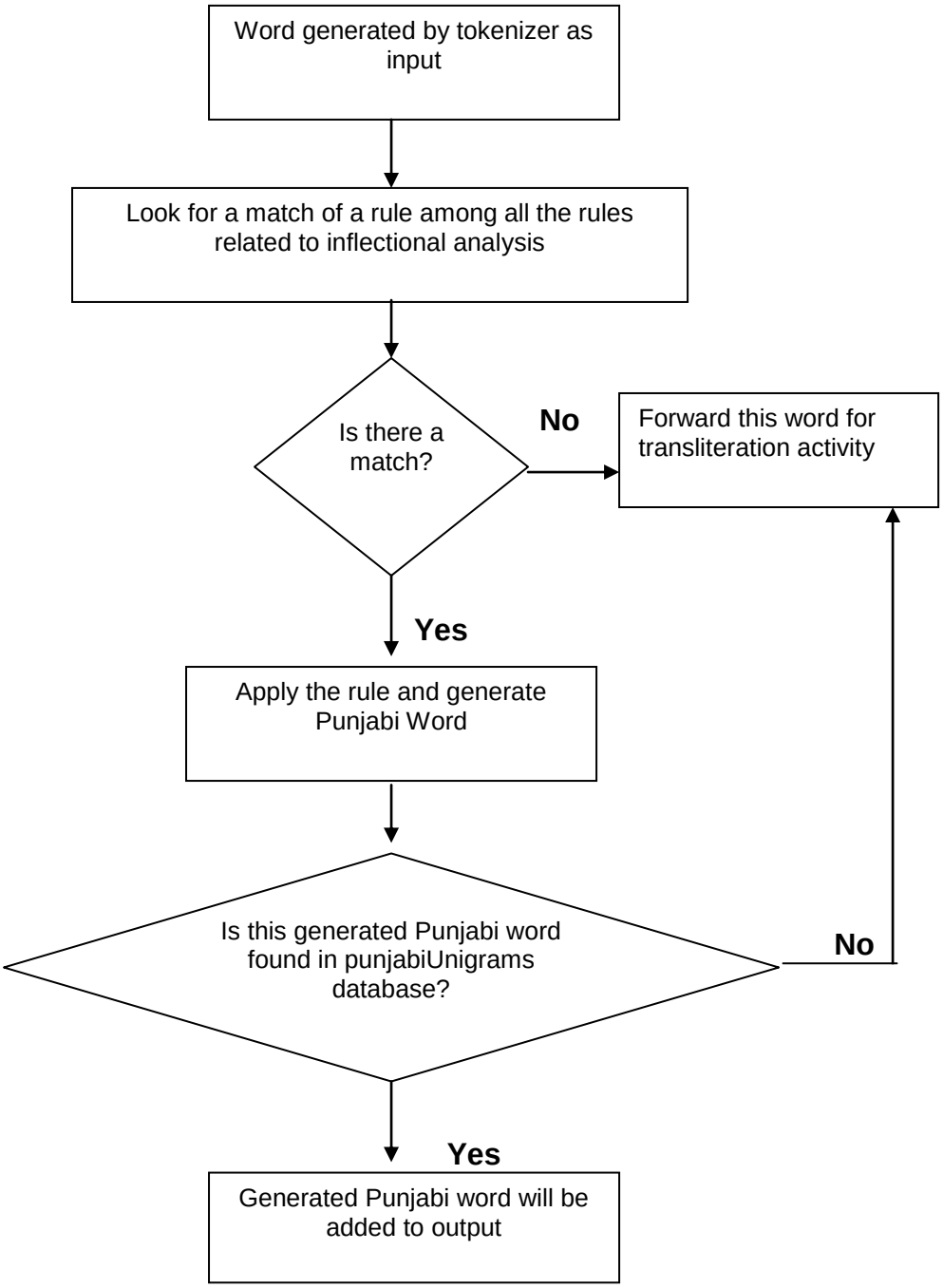

Figure 5.1: Flow Chart for Word Inflectional Analysis and generation

Following table shows various inflectional rules, each illustrated with example:

Table 5.23: Inflection Rules

Rule No.	Substring at the end of Hindi Word	Hindi Example	String to be replaced	Punjabi Example
1.	ियूँगा (*iyūṅgā*)	पियूँगा (*piyūṅgā*)	ीवांगा (*īvāṅgā*)	ਪੀਵਾਂਗਾ (*pīvāṅgā*)
2.	ियूँगी (*iyūṅgī*)	पियूँगी (*piyūṅgī*)	ीवांगी (*īvāṅgī*)	ਪੀਵਾਂਗੀ (*pīvāṅgī*)
3.	ीयूँगा (*īyūṅgā*)	जीयूँगा (*jīyūṅgā*)	ीवांगा (*īvāṅgā*)	ਜੀਵਾਂਗਾ (*jīvāṅgā*)
4.	ीयूँगी (*īyūṅgī*)	जीयूँगी (*jīyūṅgī*)	ीवांगी (*īvāṅgī*)	ਜੀਵਾਂਗੀ (*jīvāṅgī*)
5.	ीयूंगा (*īyūṅgā*)	जीयूंगा (*jīyūṅgā*)	ीवांगा (*īvāṅgā*)	ਜੀਵਾਂਗਾ (*jīvāṅgā*)
6.	ीयूंगी (*īyūṅgī*)	जीयूंगी (*jīyūṅgī*)	ीवांगी (*īvāṅgī*)	ਜੀਵਾਂਗੀ (*jīvāṅgī*)
7.	ीयेंगी (*īyēṅgī*)	जीयेंगी (*jīyēṅgī*)	ीणगीअं (*īṇgīāṃ*)	ਜੀਣਗੀਅਂ (*jīṇgīāṃ*)
8.	ीयेंगे (*īyēṅgē*)	जीयेंगे (*jīyēṅgē*)	ीणगे (*īṇgē*)	ਜੀਣਗੇ (*jīṇgē*)
9.	ियूंगा (*iyūṅgā*)	पियूंगा (*piyūṅgā*)	ीवांगा (*īvāṅgā*)	ਪੀਵਾਂਗਾ (*pīvāṅgā*)
10.	ियूंगी (*iyūṅgī*)	पियूंगी (*piyūṅgī*)	ीवांगी (*īvāṅgī*)	ਪੀਵਾਂਗੀ (*pīvāṅgī*)
11.	ियेंगी (*iyēṅgī*)	पियेंगी (*piyēṅgī*)	ीणगीअं (*īṇgīāṃ*)	ਪੀਣਗੀਅਂ (*pīṇgīāṃ*)
12.	ियेंगे (*iyēṅgē*)	पियेंगे (*piyēṅgē*)	ीणगे (*īṇgē*)	ਪੀਣਗੇ *pīṇgē*)
13.	ाऐंगीं (*āaiṅgīṃ*)	पाऐंगीं (*pāaiṅgīṃ*)	ावांगीअं (*āvāṅgīāṃ*)	ਪਾਵਾਂਗੀਅਂ (*pāvāṅgīāṃ*)
14.	ायेगा (*āyēgā*)	पायेगा (*pāyēgā*)	ावेगा (*āvēgā*)	ਪਾਵੇਗਾ (*pāvēgā*)
15.	ायेगी (*āyēgī*)	पायेगी (*pāyēgī*)	ावेगी (*āvēgī*)	ਪਾਵੇਗੀ (*pāvēgī*)
16.	ईयेगा (*īyēgā*)	पाईयेगा (*pāīyēgā*)	ोगे (*ōgē*)	ਪਾਓਗੇ (*pāōgē*)
17.	इयेगा (*iyēgā*)	पाइयेगा (*pāiyēgā*)	ोगे (*ōgē*)	ਪਾਓਗੇ (*pāōgē*)
18.	येंगे (*yēṅgē*)	पायेंगे (*pāyēṅgē*)	ोगे (*ōgē*)	ਪਾਓਗੇ (*pāōgē*)

19.	जिएगा (*jiēgā*)	दीजिएगा (*dījiēgā*)	ਓ (*ō*)	दिओ (*diō*)
20.	जीएगा (*jīēgā*)	दीजीएगा (*dījīēgā*)	ਓ (*ō*)	दिओ (*diō*)
21.	जियेगा (*jiyēgā*)	दीजियेगा(*dījiyēgā*)	ਓ (*ō*)	दिओ (*diō*)
22.	येंगी (*yēṅgī*)	जीयेंगी (*jīyēṅgī*)	ਓਗੇ (*ōgē*)	जीओगे (*jīōgē*)
23.	येंगे (*yēṅgē*)	जीयेंगे (*jīyēṅgē*)	ਓਗੇ (*ōgē*)	जीओगे (*jīōgē*)
24.	ਿਊँगा (*iūṅgā*)	पिऊँगा (*piūṅgā*)	ੀਵਾਂਗਾ (*īvāṅgā*)	पीवांगा(*pīvāṅgā*)
25.	ਿਊंगा (*iūṅgā*)	पिऊंगा (*piūṅgā*)	ੀਵਾਂਗਾ (*īvāṅgā*)	पीवांगा (*pīvāṅgā*)
26.	ੀਊँगा (*īūṅgā*)	पिऊँगा (*piūṅgā*)	ੀਵਾਂਗਾ (*īvāṅgā*)	पीवांगा (*pīvāṅgā*)
27.	ਿਊँगी (*iūṅgī*)	पिऊँगी (*piūṅgī*)	ੀਵਾਂਗੀ (*īvāṅgī*)	पीवांगी (*pīvāṅgī*)
28.	ੀਊँगी (*īūṅgī*)	पिऊँगी (*piūṅgī*)	ੀਵਾਂਗੀ (*īvāṅgī*)	पीवांगी (*pīvāṅgī*)
29.	ੀਊंगी (*īūṅgī*)	पिऊंगी (*piūṅgī*)	ੀਵਾਂਗੀ (*īvāṅgī*)	पीवांगी (*pīvāṅgī*)
30.	ੀਜिये (*ījiyē*)	दीजिये (*dījiyē*)	ਓ (*ō*)	दिओ (*diō*)
31.	ੀਏँगी (*īēṅgī*)	जीएँगी (*jīēṅgī*)	ੀਣਗੀਆਂ (*īṇgīāṃ*)	जीणगीआं (*jīṇgīāṃ*)
32.	ੀਏंगी (*īēṅgī*)	जीएंगी (*jīēṅgī*)	ੀਣਗੀਆਂ (*īṇgīāṃ*)	जीणगीआं (*jīṇgīāṃ*)
33.	ੀਏँगे (*īēṅgē*)	जीएँगे (*jīēṅgē*)	ੀਣਗੇ (*īṇgē*)	जीणगे (*jīṇgē*)
34.	ਾਉँगा (*āuṅgā*)	पाउँगा (*pāuṅgā*)	ਾਵਾਂਗਾ (*āvāṅgā*)	पावांगा (*pāvāṅgā*)
35.	ਾਏंगी (*āaiṅgī*)	पाऐंगी (*pāaiṅgī*)	ਾਉਣਗੀਆਂ (*āuṇgīāṃ*)	पाउणगीआं (*pāuṇgīāṃ*)
36.	ਾਏंगे (*āaiṅgē*)	पाऐंगे (*pāaiṅgē*)	ਾਵਾਂਗੇ (*āvāṅgē*)	पावांगे (*pāvāṅgē*)
37.	येंगी (*yēṅgī*)	पायेंगी (*pāyēṅgī*)	ਓਗੇ (*ōgē*)	पाओगे (*pāōgē*)
38.	ੂगां (*ūgāṃ*)	पहनूगां (*pahnūgāṃ*)	ਾਗਾਂ (*āgāṃ*)	पहिनागां (*pahināgāṃ*)

39.	वाना (*vānā*)	पकवाना (*pakvānā*)	ਵਾਉਣਾ (*vāuṇā*)	ਪਕਵਾਉਣਾ (*pakvāuṇā*)
40.	ाएगा (*āēgā*)	पकाएगा (*pakāēgā*)	ाऐगा (*āēgā*)	ਪਕਾਏਗਾ (*pakāēgā*)
41.	ाऐगा (*āaigā*)	पकाएगा (*pakāēgā*)	ावेगा (*āvēgā*)	ਪਕਾਵੇਗਾ (*pakāvēgā*)
42.	ाऐगी (*āaigī*)	पकाएगा (*pakāēgā*)	ावेगी (*āvēgī*)	ਪਕਾਵੇਗੀ (*pakāvēgī*)
43.	ाओगी (*āōgī*)	पकाएगा (*pakāēgā*)	ाऐंगी (*āēṅgī*)	ਪਕਾਏਂਗੀ (*pakāēṅgī*)
44.	ाओगे (*āōgē*)	पकाएगा (*pakāēgā*)	ाओगे (*āōgē*)	ਪਕਾਓਗੇ (*pakāōgē*)
45.	ातीं (*ātīṃ*)	पकाएगा) (*pakāēgā*)	ाउंदीआं) (*āundīāṃ*	ਪਕਾਉਂਦੀਆਂ (*pakāundīāṃ*)
46.	ानीं (*ānīṃ*)	पकानीं (*pakānīṃ*)	ाउणीआं (*āuṇīāṃ*)	ਪਕਾਉਣੀਆਂ (*pakāuṇīāṃ*)
47.	ूंगा (*ūṅgā*)	मारूंगा (*mārūṅgā*)	ारांगा (*āṅgā*)	ਮਾਰਾਂਗਾ (*mārāṅgā*)
48.	ूंगी (*ūṅgī*)	मारूंगी (*mārūṅgī*)	ारांगी (*āṅgī*)	ਮਾਰਾਂਗੀ (*mārāṅgī*)
49.	ेंगी (*ēṅgī*)	मारेंगी (*mārēṅgī*)	ਣਗੀਆਂ (*ṇagīāṃ*)	ਮਾਰਣਗੀਆਂ (*mārṇagīāṃ*)
50.	ेंगे (*ēṅgē*)	मारेंगे (*mārēṅgē*)	ਣਗੇ (*ṇagē*)	ਮਾਰਣਗੇ (*mārṇagē*)
51.	हरों (*harōṃ*)	परिहरों (*parihrōṃ*)	ਹਿਰਾਂ (*hirāṃ*)	ਪਰਿਹਿਰਾਂ (*parihirāṃ*)
52.	भाओं (*bhāōṃ*)	प्रतिभाओं (*pratibhāōṃ*)	ਭਾਵਾਂ (*bhāvāṃ*)	ਪ੍ਰਤਿਭਾਵਾਂ (*pratibhāvāṃ*)
53.	ूँगा (*ūṁgā*)	मारूँगा (*mārūṁgā*)	ारांगा (*āṅgā*)	ਮਾਰਾਂਗਾ (*mārāṅgā*)
54.	ेंगे (*ēṅgē*)	मारेंगे (*mārēṅgē*)	ਣਗੇ (*ṇagē*)	ਮਾਰਣਗੇ (*mārṇagē*)
55.	ारांगे (*ārāṅgē*)	मारांगे (*mārāṅgē*)	ांगे (*ṅgē*)	ਮਾਰਾਂਗੇ (*mārāṅgē*)

56.	�इएगा (*iēgā)*	पाइएगा(*pāiēgā)*	ਓਗੇ (*ōgē)*	ਪਾਓਗੇ (*pāōgē)*
57.	ऊँगा (*ūṅgā)*	पाऊँगा (*pāūṅgā)*	ਵਾਂਗਾ (*vāṅgā)*	ਪਾਵਾਂਗਾ (*pāvāṅgā)*
58.	ऊँगी (*ūṅgī)*	पाऊँगी (*pāūṅgī)*	ਵਾਂਗੀ (*vāṅgī)*	ਪਾਵਾਂਗੀ (*pāvāṅgī)*
59.	ऊंगा (*ūṅgā)*	पाऊंगा (*pāūṅgā)*	ਵਾਂਗਾ (*vāṅgā)*	ਪਾਵਾਂਗਾ (*pāvāṅgā)*
60.	ऊंगी (*ūṅgī)*	पाऊंगी (*pāūṅgī)*	ਵਾਂਗੀ (*vāṅgī)*	ਪਾਵਾਂਗੀ (*pāvāṅgī)*
61.	एँगी (*ēṅgī)*	पाएँगी (*pāēṅgī)*	ਓਗੇ (*ōgē)*	ਪਾਓਗੇ (*pāōgē)*
62.	एंगी (*ēṅgī)*	पाएंगी (*pāēṅgī)*	ਓਗੇ (*ōgē)*	ਪਾਓਗੇ (*pāōgē)*
63.	एंगे (*ēṅgē)*	पाएंगे (*pāēṅgē)*	ਓਗੇ (*ōgē)*	ਪਾਓਗੇ (*pāōgē)*
64.	एँगे (*ēṅgē)*	पाएँगे (*pāēṅgē)*	ਓਗੇ (*ōgē)*	ਪਾਓਗੇ (*pāōgē)*
65.	ऊंगी (*ūṅgī)*	पाऊंगी (*pāūṅg)ī*	ਵਾਂਗੀ (*vāṅgī)*	ਪਾਵਾਂਗੀ(*pāvāṅgī)*
66.	येगा (*yēgā)*	पायेगा (*pāyēgā)*	ਵੇਗਾ (*vēgā)*	ਪਾਵੇਗਾ (*pāvēgā)*

Database Design for punjabiUnigrams:

Following table shows the database design for punjabiUnigrams:

Table 5.24: punjabiUnigram Database Design

punjabiUnigram	Stores the Punjabi word
Frequency	Stores the frequency of this word in the analyzed Punjabi corpus.

Sample Entries for punjabiUnigrams database:

Following table shows the sample entries for punjabiUnigrams database:

Table 5.25: Sample Entries for punjabiUnigram database

punjabiUnigram	**Frequency**
ਅਤੇ (*atē)*	81897

ਪੁਲਿਸ (*pulis)*	5112
ਬਚਾਉਣ (*bacāuṇ)*	518
ਇਲੈਕਟ੍ਰੀਸ਼ੀਅਨ (*ilaikṭrīsaīan)*	3
ਪ੍ਰੌਵੀਜ਼ਨਲ (*prauvījnal)*	1

5.1.5.2 Transliteration

With the advent of new technology and the food of information through the Web, it has become increasingly common to adopt foreign words into one's language. This usually entails adjusting the adopted word's original pronunciation to follow the phonological rules of the target language, along with modification of its orthographical form. This phonetic “translation" of foreign words is called *transliteration*. Transliteration is a process that takes a character string in a source language and generates equivalent mapped character string in the target language. One of the most frequent problems translators must deal with is translating proper names and technical terms. Such terms are not translated rather are transliterated. Transliteration maps the letters of source script to letters of pronounced similarly in target script. Transliteration is particularly used to translate proper names and technical terms from languages. For example the word विशाल (*vishā)l* is transliterated as ਵਿਸ਼ਾਲ (*vishāl)* whereas translated as ਵੱਡਾ (*vaḍḍ)ā*. There must be some method in every Machine Translation system for words like technical terms and proper names of persons, places, objects etc. that cannot be found in

translation resources such as Hindi-Punjabi bilingual dictionary, surnames database, titles database etc and transliteration is an obvious choice for such words. It is the process of converting characters in one alphabet into another alphabet.

Principles for Transliteration

Following are some of the principles for a transliteration system:

- **Partial Reversibility:** Two segments of text in the target script, arising from the same source script, are to be the same if, and only if, the segments in the source script are either identical or use alternative orthography.

- **Uniformity:** Two segments of text in the target script, arising from different source scripts, are to be the same if, and only if, the two segments in the source scripts correspond precisely, according to comparative linguistics.

- **Compromise:** One symbol may have different meanings if its interpretation is never in doubt. Compromise is also necessary whenever two of these principles conflict.

- **Readability:** Even if the text is meant for computer processing, it needs to be read easily.

- **Economy:** This will also improve readability.

- **Approximation:** The symbol used should remind one of sound, and of the transliteration scheme used for printing.

Transliteration Guidelines

The following lists the general guidelines for transliterations:

Complete: Every well-formed sequence of characters in the source script should transliterate to a sequence of characters from the target script.

Predictable: The letters themselves (without any knowledge of the languages written in that script) should be sufficient for the transliteration, based on a relatively small number of rules. This allows the transliteration to be performed mechanically.

Pronounceable: Transliteration is not as useful if the process simply maps the characters without any regard to their pronunciation. Simply mapping by alphabetic order could yield strings that might be complete and unambiguous, but the pronunciation would be completely unexpected.

Unambiguous: It is possible to recover the text in the source script from the transliteration in the target script. That is, someone that knows the transliteration rules would be able to recover the precise spelling of the original source text.

Partial Reversibility: In script transliteration, there are cases where all characters in the source script may not have one-to-one mapping for transliteration in the target script. To preserve pronunciation these characters

may mapped to some character or sequence of characters that may produce a similar sound. In such cases reversibility will be incomplete.

History of Transliteration

- 1885 — The American Library Association [ALA] creates a system for representing Cyrillic characters. No diacritics are used. (e.g. zh, kh, tch, sh, shtch, ye [for jat], yu, ya) Reverse transliteration is not considered.
- 1898 — The Prussian Instructions (Preussische Instruktionen [PI]) are created, which use a system of transliteration based on the Croatian model (with diacritics).
- 1909 — The ALA and British Library Association [BLA] allow for two systems, the ALA system and one based on Croatian.
- 1905 — Library of Congress creates their system, which is virtually identical to what is used today.
- 1917— The British Academy creates its own system. Like many other systems. It does not take into account reverse transliteration.
- 1930s— Central European and Scandinavian countries adopt the Prussian Instructions [PI]. This system was based on the Croatian model. Exceptions were made for German speaking countries, where "ch" was used instead of "h" for Cyrillic "x"
- In France the Bibliotheque Nationale adopts a purely phonetic rendering following French spelling conventions (transcription rather than transliteration).
- 1953 — The British Royal Society [BRS] creates another system,

covering Russian, Serbian & Bulgarian (but not Ukrainian, Macedonian or Belorusian).

- 1954 — The International Organization for Standardization [ISO] creates IS0/R9. Based on Croatian, this transliteration system is very close to the PI system.
- 1959 — The British Standards Institution [BS/BSI] rejects ISO/R9 (because of its reliance on multiple diacritics) and comes up with its own system: BS 2979. Very close to the British Royal Society system. (This system is used by Chemical Abstracts).
- 1976 — The American National Standards Institute [ANSI] publishes their system, nearly identical to the BSI system.
- 1968 — ISO/R9:1968 is relaxed to allow for the ANSI and BS 2979 systems (in certain countries).
- 1995 — ISO/R9:1995 reverts to to its initial standards, doing away with allowing "ch" or "kh" for Cyrillic "x."

Transliteration Models:

Four machine transliteration models have been proposed by several researchers: grapheme-based transliteration model (Ψ_G), phoneme-based transliteration model (Ψ_P), hybrid transliteration model (Ψ_H) and correspondence-based transliteration model (Ψ_C). These models are classified in terms of the units to be transliterated. The Ψ_G is sometimes referred to as the *direct method* because it directly transforms source language graphemes into target language graphemes without any phonetic

knowledge of the source language words. The Ψ_P is sometimes referred to as the *pivot method* because it uses source language phonemes as a pivot when it produces target language graphemes from source language graphemes.

The Ψ_H and Ψ_C make use of both source language graphemes and source language phonemes when producing target language transliterations. Hereafter, we refer to a source language grapheme as a *source grapheme*, a source language phoneme as a *source phoneme*, and a target language grapheme as a *target grapheme*.

The transliterations produced by the four models usually differ because the models use different information. Generally, transliteration is a phonetic process, as in Ψ_P , rather than an orthographic one, as in Ψ_G. However, standard transliterations are not restricted to phoneme-based transliterations.

A review of the archives of Indian language documents on the Internet reveals several other schemes of Transliteration and fonts. The Indology site in England has electronic texts of Sanskrit Documents prepared in CSX format, a special input method recommended in 1990 for Sanskrit data entry using a DOS feature called Code page switching. ITRANS which is more recent offers conversion facilities to convert from CSX to the ITRANS format. The Tamil archives of the Institute of Indology and Tamil Studies in Germany (IITS) has an archive of texts of Tamil Sangam literature and many Sanskrit documents. These archives are based on the transliteration scheme recommended by the University of Madras, a fairly well known and accepted standard.

Transliteration among Indian scripts is easily achieved using ISCII (Indian Script Code for Information Interchange). ISCII has been designed using the phonetic property of Indian scripts and caters to the superset of all Indian scripts. By attaching an appropriate script rendering mechanism to ISCII, transliteration from one Indian script to another is achieved in a natural way. Transliteration schemes have to face the problem of letters present in one language and not in the other. Thus, unless a superset of letters from all the Indian Languages is formed, uniform transliteration is ruled out. Ram Viswanadha [217] has the view that when characters do not have any appropriate transliteration they should be consumed and not replaced with any other character. This results in partial loss of reversibility.

Our Approach

Although Hindi and Punjabi are closely related languages and for except few cases all alphabets of Devanagri script are present in Gurmukhi script, the task of transliteration from Hindi to Punjabi is not trivial. The Unicode encoding has eased the problem to some extent. In our system besides direct character mappings from alphabet in one script to another, rule based transliteration useful for a translation system is also employed to improve its accuracy. Using only direct character mapping, it shows that this word is out- -of-vocabulary for our system and has been displayed in the output by changing the script and is unknown to our system. Both direct character

mapping and complex rules employed for transliteration are explained in the following sections.

Direct Character Mappings:

Both Hindi and Punjabi languages are phonetic languages and their scripts represent the phonetic repository of their respective languages. These phonetic sounds are used to determine the relations between the characters of two scripts. On the basis of this idea, character mappings are determined. With this system every alphabet can be uniquely mapped to the corresponding alphabet as shown in following table. Taking into account the similarity of both the scripts, letter to letter mapping is the obvious choice for baseline. Following table 5.26 shows the direct mapping of Hindi to Punjabi alphabets:

Table 5.26: Direct Hindi to Punjabi Character Mapping

Hindi Character	Decimal Code	Punjabi Character	Decimal Code
ँ	2305	ਁ or ਂ	2562 or 2672
ं	2306	ਁ or ਂ	2562 or 2672
ः	2307	:	58
अ	2309	ਅ	2565
आ	2310	ਆ	2566
इ	2311	ਇ	2567
ई	2312	ਈ	2568

उ	2313	ੁ	2569
ऊ	2314	ੂ	2570
ऋ	2315	ਰਿ	2608+2623
ऍ	2317	ੇ	2575
ऐ	2318	ੇ	2575
ए	2319	ੇ	2575
ऐ	2320	ਐ	2575
ऑ	2321	ਆ	2566
ओ	2322	ਅੰ	2580
ओ	2323	ਓ	2579
औ	2324	ਅੰ	2580
क	2325	ਕ	2581
ख	2326	ਖ	2582
ग	2327	ਗ	2583
घ	2328	ਘ	2584
ङ	2329	ਙ	2585
च	2330	ਚ	2586
छ	2331	ਛ	2587
ज	2332	ਜ	2588
झ	2333	ਝ	2589
ञ	2334	ਞ	2590

ट	2335	ਟ	2591
ठ	2336	ਠ	2592
ड	2337	ਡ	2593
ढ	2338	ਢ	2594
ण	2339	ਣ	2595
त	2340	ਤ	2596
थ	2341	ਥ	2597
द	2342	ਦ	2598
ध	2343	ਧ	2599
न	2344	ਨ	2600
ऩ	2345	ਨ	2600
प	2346	ਪ	2602
फ	2347	ਫ	2603
ब	2348	ਬ	2604
भ	2349	ਭ	2605
म	2350	ਮ	2606
य	2351	ਯ	2607
र	2352	ਰ	2608
ऱ	2353	ਰ	2608
ल	2354	ਲ	2610
ळ	2355	ਲ਼	2611

क़	2356	ਲ਼	2611
व	2357	ਵ	2613
श	2358	ਸ਼	2614
ष	2359	ਸ਼	2614
स	2360	ਸ	2616
ह	2361	ਹ	2617
़	2364	਼	2620
ा	2366	ਾ	2622
ि	2367	ਿ	2623
ी	2368	ੀ	2624
ु	2369	ੁ	2625
ू	2370	ੂ	2626
े	2375	े	2631
ै	2376	ੈ	2632
ॉ	2377	ਾ	2622
ो	2378	ੋ	2635
ो	2379	ੋ	2635
ौ	2380	ੌ	2636
्	2381	੍	2637
ॐ	2384	ਓਮ	2579+2606
क़	2392	ਕ	2581

ख़	2393	ਖ਼	2649
ग़	2394	ਗ਼	2650
ज़	2395	ਜ਼	2651
ड़	2396	ੜ	2652
ढ़	2397	ੜ੍ਹ	2652+2637+2617
फ़	2398	ਫ਼	2654
य़	2399	ਯ਼	2607

Following flowchart explains the working of the transliteration phase:

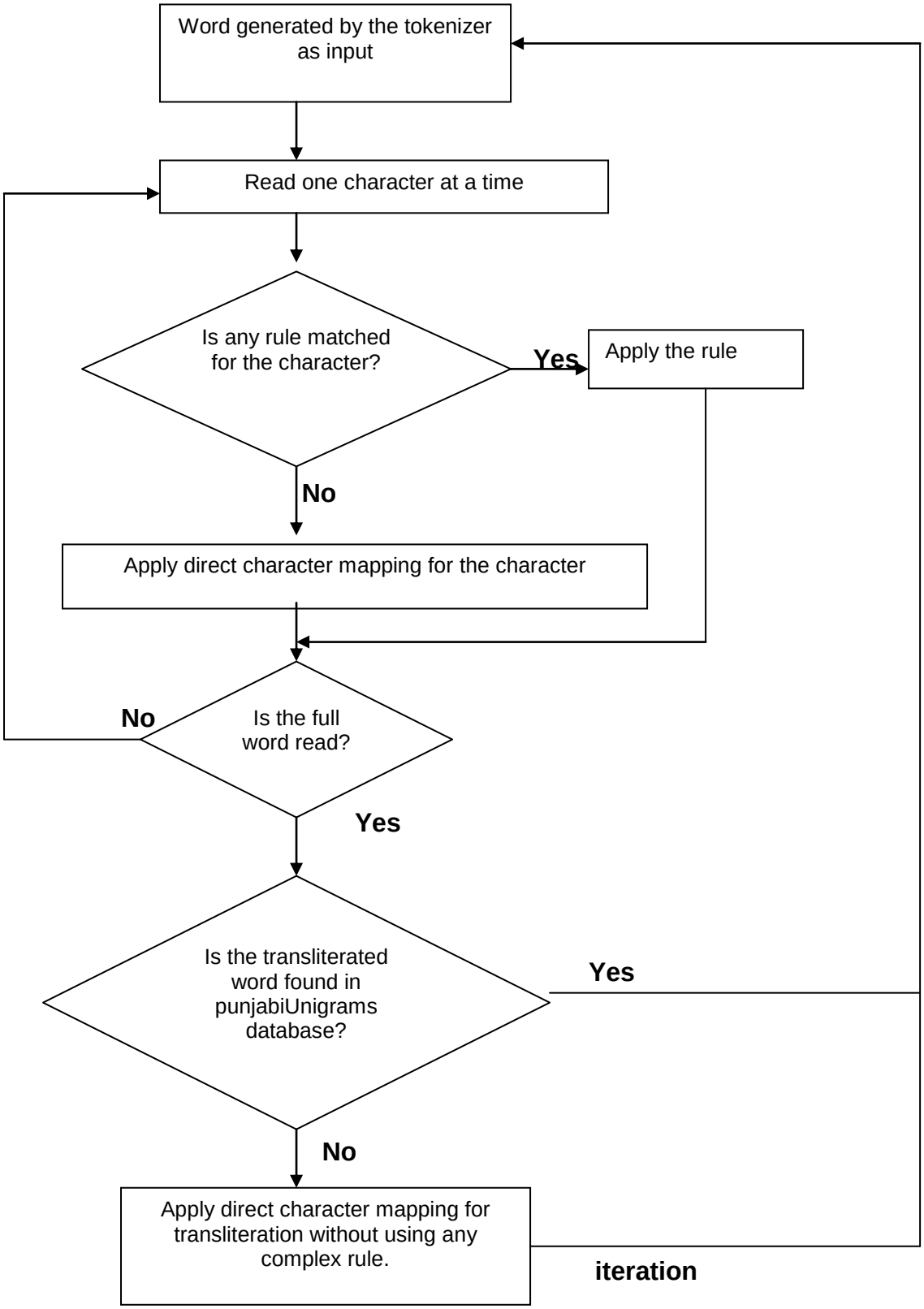

Figure 5.2: Flow Chart for Transliteration Module

Rule Based Mapping:

Although direct character mapping can produce successfully the transliterated output in Punjabi which can represent the source word in target language, but we can improve the results by making them nearer to target language in term of spellings and choice of alphabets by using some set of rules. A quite reasonable improvement can be achieved by small amount of dependency or contextual rules. Following are the rules for alleviating some of the problems not solved by direct character mapping.

1. या at the end of the words of length greater than 3 will be transliterated into ਆ. For example: विकिपीडिया (*vikipīḍiyā)* : ਵਿਕੀਪੀਡਿਆ (*vikīpīḍiā),* वेबदुनिया (*vēbduniyā)*: ਵੇਬਦੁਨਿਆ(*vēbduniā)*
2. Substring आया in the word of length greater than 3 will be transliterated to ਆਯਾ. For example: आयाम (*āyām)*: ਆਯਾਮ(*āyām)*, आयातित(*āyātit)*: ਆਯਾਤਿਤ(*āyātit) .*
3. यी at the end of the words will be transliterated into ਈ. For example: वाजपेयी (vājpēyī): ਵਾਜਪੇਈ (vājpēī), एंप्लॉयी (ēmplāyī):ਅੰਪਲਾਈ (amplāī).

4. य at the end of the word will be transliterated into ਏ. For example: अक्षय (*akshay)* : ਅਕਸ਼ਏ (*akshaē*), समुद्रपारीय (*samudrpārīy)* : ਸਮੁਦਰਪਾਰੀਏ (*samudrapārīē).*

5. या preceded by consonant or halant in the word, will be transliterated to ਿ + ਅ + ਾ. For example: प्यास (*pyās)* : ਪਿਆਸ(*piās),* ब्यास(*byās)*: ਬਿਆਸ(*biās).*

6. यू preceded by consonant or halant in the word, will be transliterated into ਿ + ਊ. For example: अभिमन्यू (*abhimnyū)*: ਅਭਿਮਨਿਊ(*abhimniū),* एविन्यू (*ēvinyū)* : ਏਵਿਨਿਊ(*ēviniū).*

7. यु preceded by consonant or halant in the word, will be transliterated into ਿ + ਉ. For example: इम्म्युनोलॉजी(*immyunōlājī)* : ਇੰਮਿਉਨੋਲਾਜੀ (*immiunōlājī)*, मैक्युलम(*maikyulam*) : ਮੈਕਿਉਲਮ(*maikiulam).*

8. ये preceded by consonant or halant in the word, there is a consonant, will be transliterated to ਿ + ਏ. For example: क्रियेटिव(*kriyēṭiv)* : ਕਰਿਏਟੀਵ (*kariēṭiv),* एनडीब्येर(*ēnḍībyēr)*: ਏਨਡੀਬਿਏਰ(*ēnḍībiēr).*

9. यै preceded by consonant or halant in the word, will be transliterated into ਿ + ਐ. For example: कहिल्यै(*kahilyai)*: ਕਹਿਲਿਐ(*kahiliai),* मोदगप्यै (*mōdgapyai)*: ਮੋਦਗਪਿਐ(*mōdgapiai).*

10. ये within the words of length greater than 2 and not at the beginning of the word, will be transliterated into ਏ. For example: थुयेर (*thuyēr)*: ਥੁਏਰ(*thuēr).*

11. यू within the words and not at the beginning of the word, will be transliterated into ਿ and ਊ. For example: बयूरा (*bayūrā)*: ਬਿਊਰਾ(*biūrā),* कंप्यूटर(*kampyūṭar)* : ਕੰਪਿਊਟਰ(*kampiūṭar).*

12. य preceded by matra ਾ and followed by consonant, will be transliterated into ਇ. For example: रसायनिक (*rasāynik)*: ਰਸਾਇਨਿਕ (*rasāiṇik),* डायबिटीज(*ḍāybiṭīj):*ਡਾਇਬਿਟੀਜ(*ḍāibiṭīj)*.

13. ाया at the end of the word will be transliterated into ਾਇਆ. For example: समझाया (*samjhāyā)*: ਸਮਝਾਇਆ(*samjhāiā),* ठहराया(*ṭhahrāyā)*: ਠਹਰਾਇਆ**(***ṭhahrāiā).*

14. ਾਯਾ within the word and not at the end ਾਯਾ. For example: सजायाफ्ता (*sajāyāphtā)***:** ਸਜਾਯਾਫ਼ਤਾ (*sajāyāphātā),* नायाब(*nāyāb)***:**ਨਾਯਾਬ(*nāyāb).*

15. य preceded by the matra ि and followed by consonant in the word, will be transliterated ਅ. For example: कैल्शियम (*kailshiyam)***:** ਕੈਲਸ਼ਿਅਮ (*kailshiam),* स्टेडियम (*sṭēḍiyam)***:**ਸਟੇਡਿਅਮ(*saṭēḍiam).*

16. य preceded by the matra ै and followed by the matra ा in the word, will be transliterated into ਿ + ਆ. For example: मुथैया (*muthaiyā)* : ਮੁਥਿਆ(*muthiā),* फैयाज़ (*phaiyāja)*: ਫਿਆਜ਼(*phiāj).*

17. Dead Consonant क (क्) followed by live consonant ख in the word, will be transliterated into ੱ. For example: मक्खन(*makkhan)*: ਮੱਖਣ (*makkhaṇ*), मधुमक्खी(*madhumkkhī):* ਮਧੁਮੱਖੀ*(madhumkkhī).*

18. Dead Consonant च(च्) followed by live consonant छ in the word, will be transliterated into ੱ. For example: इच्छा (*icchā)*: ਇੱਛਾ(*icchā),* अकच्छ(*akacch)*: ਅਕੱਛ*(akacch*).

19. Dead consonant ट(ट्) followed by live consonant ठ, will be transliterated into ੱ. For example: मट्ठी(*maṭṭhī)* : ਮੱਠੀ(*maṭṭhī),*भट्ठा(*bhaṭṭhā)*: ਭੱਠਾ(*bhaṭṭhā).*

20. Dead consonant ग(ग्) followed by live consonant घ in the word, will be transliterated into ੱ. For example: मग्घर (*magghar)* : ਮੱਘਰ(*magghar),* लक्कड़बग्घा(*lakkṛabgghā)*: ਲੱਕੜਬੱਘਾ(*lakkṛabgghā).*

21. Dead consonant ज(ज्) followed by live consonant झ in the word, will be transliterated into ੱ. निज्झर (*nijjhar)* : ਨਿੱਝਰ*(nijjhar),* उज्झड़ (*ujjhaṛ)* : ਉੱਜੜ(*ujjaṛ*).

22. Dead consonant त (त्) followed by live consonant थ in the word, will be transliterated into ੱ. For example: पत्थर (*patthar)*:ਪੱਥਰ *(patthar),* कत्था (*katthā)*:ਕੱਥਾ(*katthā).*

23. Dead consonant द(द्) followed by live consonant ध in the word, will be transliterated into ੱ. For example: सिद्धार्थ (*siddhārth)*: ਸਿੱਧਾਰਥ (*siddhārath),* अनिरुद्ध(*Aniruddh)* : ਅਨਿਰੁੱਧ(*aniruddh).*

24. Dead consonant ड(ड्) followed by live consonant ढ in the word, will be transliterated into ◌ੱ. For example: गड्ढा (*gaḍḍhā)*: ਗੱਢਾ(*gaḍḍhā),* बुड्ढा(*buḍḍhā)* : ਬੁੱਢਾ(*buḍḍhā).*

25. Conjuct consonant ज्ञ in the word will be transliterated in Gurmukhi into ◌ਿ +ਗ + ਅ. For example:खगोलवैज्ञानिक (*khagōlvaigiānik)* : ਖਗੋਲਵੈਗਿਆਨੀਕ(*khagōlvaigiānīk),* मनोविज्ञानी(*manōvigiānī)*: ਮਨੋਵਿਗਿਆਨੀ (*manōvigiānī).*

26. Dead consonant म (म्) followed by any labial consonant (प,ब,फ,म,व), will be transliterated into ◌ੰ. पम्मी (*pammī)*: ਪੰਮੀ(*pammī),* पम्प(*pamp)*: ਪੰਪ(*pamp).*

27. Dead Consonant followed by the live consonant of its preceding dead consonant in the word, will be transliterated to ◌ੱ. This rule is not applicable for consonants म and न. For example: दिग्गज(*diggaj)* : ਦਿੱਗਜ(*diggaj),* छज्जा(*chajjā)*: ਛੱਜਾ(*chajjā).*

28. Dead Consonant न (न्) followed by live consonant न in the word,will be transliterated to ं. For example: पन्ना(*pannā)* : ਪੰਨਾ(*pannā),* अन्नामलाई(*annāmlāī(annāmlāī)*,ਅੰਨਾਮਲਾਈ (*annāmlāī).*

29. ृ or (्र) preceded by consonant क in the word will be transliterated ्रि. For Example: क्रिकेट(*krikēṭ)*: ਕ੍ਰਿਕੇਟ(*krikēṭ),*संस्कृत(*saṃskrit)* : ਸੰਸਕ੍ਰਿਤ(*saṃskrit).*

30. (्र) preceded by consonant ष in the word will be transliterated ्रु. For Example: प्रयोगशाला(*prayōgshālā)*: ਪ੍ਰਯੋਗਸ਼ਾਲਾ(*prayōgshālā),* प्रिंटिंग (*priṇṭiṅg)* :ਪ੍ਰਿੰਟਿੰਗ(*priṇṭiṅg).*

31. Conjuct consonant क्ष in the word will be transliterated to कश. For example: क्षेमेन्द्र राव(kshēmēndr rāv): ਕਸ਼ੇਮੇਂਦਰ ਰਾਵ (kashēmēndar *rāv),* कुरुक्षेत्र(*kurukshētr)*,ਕੁਰੁਕਸ਼ੇਤਰ(*kurukshētar).*

32. Conjuct consonant त्र in the word will be transliterated to तर. For example: त्रिफला(*triphlā)*: ਤ੍ਰਿਫਲਾ(*triphlā)*, तंत्रिका(*tantrikā)*:ਤੰਤਰਿਕਾ(*tantrikā).*

33. Conjuct consonant श्र in the word will be transliterated to ਸ਼ਰ. For example: श्रवण(*shravaṇ)* : ਸ਼ਰਵਣ(*sharvaṇ),* श्रीलंकाई(*shrīlṅkāī*) : ਸ਼ਰੀਲੰਕਾਈ (*sharīlṅkāī).*

34. ं or ँ preceded by either आ or ई or ऐ or ओ or औ or ओ or उ or ऊ or ा or ी or े or ै or ो or ौ or ो then ं or ँ in the word will be transliterated to ਂ. For example: डिपार्टमेंट (*ḍipārṭmēṇṭ*): ਡਿਪਾਰਟਮੇਂਟ (*ḍipārṭamēṇṭ),* ऐंट्री(*aiṇṭrī)*:ਐਂਟਰੀ(*aiṇṭrī).*

35. ं or ँ preceded by either ए or अ or ु or ू or इ or ि in the word will be transliterated to ੰ. For example: इंटरनैशनल(*iṇṭranaishnal)* : ਇੰਟਰਨੈਸ਼ਨਲ (*iṇṭranaishnal),* एंटनी(*ēṇṭnī)*:ਏਂਟਨੀ(*ēṇṭnī).*

36. ं or ँ between two consecutive consonants in the word, will be transliterated to ੰ. For example: संदीप(*sandīp)*:ਸੰਦੀਪ(*sandīp),* स्टूडंट्स (*sṭūḍaṇṭs)*,ਸਟੂਡੰਟਸ(*saṭūḍṇṭas).*

When translation was performed on a document of 100 pages consisting of about 3,58,874 words, We found that about 24% of total words gets transliterated during translation process. Thus, above transliteration module plays a major role in translation.

5.3 Summary

This chapter presented the detailed working of tokenizer and the working of sub phases of translation engine phase of our Machine Translation system. The translation engine plays an important role in selecting the correct target language. Identifying surnames and titles modules detect the proper names of a person and pass them to the transliteration module. *N-gram* approach is employed for word sense disambiguation. The solution for handling out of vocabulary words using inflectional analysis without using morphology and transliteration activities are discussed in depth. The output generated by this phase is further refined by the post-processing phase. This phase is discussed in detail in next chapter.

Chapter 6

Post-Processing

6.1 Grammar Corrections

In spite of the great similarity between Hindi and Punjabi languages, there are still a number of important grammatical divergences: gender and number divergences which affect agreement. The grammar is incorrect or the relation of words in their reference to other words, or their dependence according to the sense is incorrect and needs to be adjusted. This phase is the tail end of our Machine Translation system. It is a sentence level post-processing module that improves the translation quality by making corrections in the translation generated. In other words, it can be said that it is a system of correction for ill-formed sentences. The output generated by the translation engine phase becomes the input for post-processing phase. This phase will correct the grammatical errors based on the rules implemented in the form of regular expressions discussed in the next section. In this section, we will discuss error categories which include those mistakes that lead to ungrammaticality, and thus need to be corrected. It is not possible to fully remove all the grammatical errors but to some extent. In each of the examples given in each error category, sentence marked with the asterisk (*) is

ungrammatical sentence and the other sentence without the asterisk (*) is the one corrected grammatically by the post processing phase.

1. **Within Verb Phrase Agreement:**

In a typical Punjabi sentence, within verb phrase, all the verbs must agree in gender and number.

1(a) * ਨਿਰਮਲਾ ਦੀ ਅਵਾਜ ਸੁਣਦੇ ਹੀ ਭੱਜਦੀ ਹਨ।

(nirmalā dī avāj suṇdē hī bhajjdī han.)

ਨਿਰਮਲਾ ਦੀ ਅਵਾਜ ਸੁਣਦੇ ਹੀ ਭੱਜਦੀਆਂ ਹਨ।

(nirmalā dī avāj suṇdē hī bhajjdīāṃ han.)

1(b) * ਉਨ੍ਹਾਂ ਕੰਪਨੀਆਂ ਨੂੰ ਜਿਆਦਾ ਵਿਆਕੁਲ ਕਰੇਗਾ ਜੋ ਭਾਰਤ ਵਿੱਚ ਸਥਿਤ ਆਪਣੀ ਅਨੁਸ਼ੰਗੀਆਂ

ਦੁਆਰਾ ਕੰਮ-ਕਾਜ ਕਰਾਂਦੀ ਹਨ ।

(unhāṃ kampnīāṃ nūṃ jiādā viākul karēgā jō bhārat vicc sathit āpṇī anushṅgīāṃ)

ਉਨ੍ਹਾਂ ਕੰਪਨੀਆਂ ਨੂੰ ਜਿਆਦਾ ਵਿਆਕੁਲ ਕਰੇਗਾ ਜੋ ਭਾਰਤ ਵਿੱਚ ਸਥਿਤ ਆਪਣੀ ਅਨੁਸ਼ੰਗੀਆਂ

ਦੁਆਰਾ ਕੰਮ-ਕਾਜ ਕਰਾਂਦੀਆਂ ਹਨ ।

(unhāṃ kampnīāṃ nūṃ jiādā viākul karēgā jō bhārat vicc sathit āpṇī anushṅgīāṃ duārā kamm-kāj karāndīāṃ han)

1(c) *ਪਿਛਲੇ ਸਾਲ ਇਸੇ ਮਹੀਨੇ ਕੰਪਨੀ ਨੇ 2,76,580 ਮੋਟਰਸਾਈਕਲਾਂ ਵੇਚੀ ਸਨ ।

(pichlē sāl isē mahīnē kampnī nē 2,76,580 mōṭarsāīklāṃ vēcī san)

ਪਿਛਲੇ ਸਾਲ ਇਸੇ ਮਹੀਨੇ ਕੰਪਨੀ ਨੇ 2,76,580 ਮੋਟਰਸਾਈਕਲਾਂ ਵੇਚੀਆਂ ਸਨ ।

(pichlē sāl isē mahīnē kampnī nē 2,76,580 mōṭarsāīklāṃ vēcīāṃ san)

2. Noun's Oblique Form before Postpositions

In the Punjabi Sentence, the noun before the postpositions take the oblique form.

2(a) *ਸਭ ਸਮਾਂ ਤੇ ਆਓ ।

(sabh samāṃ tē āō)

ਸਭ ਸਮੇਂ ਤੇ ਆਓ ।

(sabh samēṃ tē āō)

2(b) *ਰਾਜਾ ਨੇ ਮੰਤਰੀਆਂ ਨੂੰ ਆਦੇਸ਼ ਦਿੱਤਾ ।

(rājā nē mantrīāṃ nūṃ ādēsh dittā .)

ਰਾਜੇ ਨੇ ਮੰਤਰੀਆਂ ਨੂੰ ਆਦੇਸ਼ ਦਿੱਤਾ

(rājē nē mantrīāṃ nūṃ ādēsh dittā)

3. Subject Verb Agreement

In Punjabi Sentence, the subject must agree with verb.

3(a) *ਤੁਸੀ ਕੋਈ ਵੀ ਮੁਸੰਮੀ ਫਲ ਲੈ ਸਕਦੇ ਹਨ ।

(tusī kōī vī musmmī phal lai sakadē han)

ਤੁਸੀ ਕੋਈ ਵੀ ਮੁਸੰਮੀ ਫਲ ਲੈ ਸਕਦੇ ਹੋ ।

(tusī kōī vī musmmī phal lai sakadē hō)

3(b) *ਅਸੀਂ ਵਾਲਾਂ ਦੀ ਠੀਕ ਦੇਖਭਾਲ ਕਰ ਸੱਕਦੇ ਹਨ ।

(asīṃ vālāṃ dī ṭhīk dēkhbhāl kar sakkdē han .)

ਅਸੀਂ ਵਾਲਾਂ ਦੀ ਠੀਕ ਦੇਖਭਾਲ ਕਰ ਸੱਕਦੇ ਹਾਂ ।

(asīṃ vālāṃ dī ṭhīk dēkhbhāl kar sakkdē hāṃ)

4. Verb Object noun phrase agreement if there is ਚਾਹੀਦਾ *cāhīdā* in verb phrase

In Punjabi Sentence, verb must agree with the object noun phrase if there is ਚਾਹੀਦਾ in the verb phrase.

4(a) * ਪ੍ਰੀਖਿਆ ਵਿੱਚ ਇਹ ਗੱਲ ਸਾਹਮਣੇ ਆਈ ਹੈ ਕਿ ਸੰਕਟਗਰਸਤ ਬੈਂਕ ਆਫ ਅਮਰੀਕਾ ਨੂੰ ਸਭਤੋਂ

ਜਿਆਦਾ 33 . 9 ਅਰਬ ਡਾਲਰ ਦੀ ਪੂੰਜੀ ਚਾਹੀਦਾ ਹੈ ।

(prīkhiā vicc ih gall sāhmaṇē āī hai ki sañkṭagrasat bañk āph amrīkā nūṃ sabhtōṃ jiādā 33 . 9 arab ḍālar dī pūñjī cāhīdā hai .)

ਪ੍ਰੀਖਿਆ ਵਿੱਚ ਇਹ ਗੱਲ ਸਾਹਮਣੇ ਆਈ ਹੈ ਕਿ ਸੰਕਟਗਰਸਤ ਬੈਂਕ ਆਫ ਅਮਰੀਕਾ ਨੂੰ ਸਭਤੋਂ

ਜਿਆਦਾ 33 . 9 ਅਰਬ ਡਾਲਰ ਦੀ ਪੂੰਜੀ ਚਾਹੀਦੀ ਹੈ ।

(prīkhiā vicc ih gall sāhmaṇē āī hai ki sañkṭagrasat bañk āph amrīkā nūṃ sabhtōṃ jiādā 33 . 9 arab ḍālar dī pūñjī cāhīdī hai .)

6.2 Pattern matching and Regular Expressions:

Pattern matching is a searching technique employed normally on a string containing text in order to locate a portion or all of the specified data based on a specific search pattern criterion. A regular expression is a special text string for describing a search pattern. Regular expressions are a 'way to describe text through pattern matching' (Stubblebine 2003: 1). Regular expressions provide a powerful, flexible, and efficient method for processing text. The extensive pattern-matching notation of regular expressions allows to quickly parsing large amounts of text to find specific character patterns; to extract, edit, replace, or delete text substrings. The idea of using regular expressions for natural language processing is widely known. By using them, the most complex and repetitive linguistic errors can be identified and replaced with the right text in the MT output. The syntax of regular expressions can be simple or highly complex, depending on the pattern.

6.2.1 Related Works

The concept of automated Post-Editing was first introduced by Knight and Chander[218] and further explored by Allen and Hogan with a view to fix systematic errors committed by an MT system [219]. When these MT errors cannot be fixed with advanced User Dictionary coding techniques, they may be fixed using powerful global search and replace patterns. Roturier [220-223] used regular expressions for post-processing module of their Machine

Translation system. Kartunnen [224] suggests applying finite automata and transducers that represent regular expressions, for natural language texts. Oflazer [225] shows the use of regular expressions for tokenization, shallow parsing or morphology analysis. Hasan [226] describes the use of regular expressions for sentence clustering in SMT. Number of hybrid experiments have been conducted by combining rule-based MT (RBMT) systems with Statistical Post-Editing (SPE) systems. Two experiments were carried out for the shared task of the ACL 2007 Workshop on Statistical Machine Translation, combining a raw SYSTRAN system with a statistical post-editing (SPE) system. One experiment was run by NRC using the language pair English<>French in the context of Automatic Post-Edition systems using the PORTAGE system. The second experiment based on the same principle was run on the German > English and Spanish > English language pairs using the Moses system. The objective was to train a SMT system on a parallel corpus composed of SYSTRAN translations with the referenced source aligned with its referenced translation. A detailed evaluation of these experiments was then conducted and presented in [227,228]. They concluded that the SYSTRAN+SPE experiments demonstrated very good results – both on automatic scoring and on linguistic analysis. Their detailed comparative analysis provided directions on how to further improve these results by adding "linguistic control" mechanisms.

6.2.2 Our Approach

Here, we present the use of regular expressions in a translation system for doing post editing. The grammatical categories discussed in Section 6.1 are corrected using pattern matching through regular expressions in the MT output. It is a two step process:

(i) Pattern mentioned in regular expressions are matched with text.

(ii) If some pattern(s) matches with the strings in the text, it is replaced with the required one mentioned in pattern matched regular expression.

We have formulated 28 regular expressions for correcting such grammatical errors. Following table shows the distribution of regular expressions on the basis of error categories discussed above:

Table 6.1: Grammatical Error Category wise Regular Expression Distribution

S.No.	Grammatical Error Category	Regular Expression Count
1.	Within Verb Phrase agreement	12
2.	Noun's Oblique Form before Postpositions	02
3.	Subject Verb Agreement	13
4.	Verb Object noun phrase agreement if there is ਚਾਹੀਦਾ (*cāhīdā)* in verb phrase	01

For instance, the following example shows the subject verb agreement through regular expression:

MT output before post-processing : ਅਸੀਂ ਵਾਲਾਂ ਦੀ ਠੀਕ ਦੇਖਭਾਲ ਕਰ ਸੱਕਦੇ ਹਨ ।

(*asīṃ vālāṃ dī ṭhīk dēkhbhāl kar sakkdē han* .)

Search pattern : (ਅਸੀਂ)(((?!\u0A05\u0A38\u0A40).)+)(ਹਨ)

Replace pattern : $1$2ਹਾਂ

MT output after post-processing : ਅਸੀਂ ਵਾਲਾਂ ਦੀ ਠੀਕ ਦੇਖਭਾਲ ਕਰ ਸੱਕਦੇ ਹਾਂ ।

(asīṃ vālāṃ dī ṭhīk dēkhbhāl kar sakkdē hāṃ)

In the above example, the pattern is matched for a sentence in which subject is ਅਸੀਂ and verb is ਹਨ. Even the regular expression has been written in complex way to handle similar nested patterns also. $1 and $2 are the environment variables to store the intermediate substrings matched within the pattern.

The analysis was done on a document consisting of 35500 words. It was found that 6.197% of the output text has been corrected grammatically using these regular expressions. Following table shows the contributions of various regular expression categories in correcting the grammatical errors:

Table 6.2: % Contribution of Regular Expressions on the basis of Grammatical Error Categories

S.No.	Grammatical Error Category	Regular Expression

		Count
1.	Within Verb Phrase agreement	38.67%
2.	Noun's Oblique Form before Postpositions	3.20%
3.	Subject Verb Agreement	35.63%
4.	Verb Object noun phrase agreement if there is ਚਾਹੀਦਾ (*cāhīdā)* in verb phrase	9.84%

6.3 Sample Translations:

Following are some sample translations obtained from the system:

Input: संभावना है कि जल्दी ही कर्नाटक के राजनीतिक भविष्य का कोई फ़ैसला हो जाएगा

sambhāvnā hai ki jaldī hī karnāṭak kē rājnītik bhavishy kā kōī phaaislā hō jāēgā

Output: ਸੰਭਾਵਨਾ ਹੈ ਕਿ ਜਲਦੀ ਹੀ ਕਰਨਾਟਕ ਦੇ ਰਾਜਨੀਤਿਕ ਭਵਿੱਖ ਦਾ ਕੋਈ ਫੈਸਲਾ ਹੋ ਜਾਵੇਗਾ।

(sambhāvnā hai ki jaldī hī karnāṭak dē rājnītik bhavikkh dā kōī phaislā hō jāvēgā).

Input: लाला साहब ने अपना जीवन इसी काम के हेतु अर्पण कर दिया है।

(lālā sāhab nē apnā jīvan isī kām kē hētu arpaṇ kar diyā hai.)

Output: ਲਾਲਾ ਸਾਹਿਬ ਨੇ ਆਪਣਾ ਜੀਵਨ ਇਸ ਕੰਮ ਦੇ ਲਈ ਅਰਪਣ ਕਰ ਦਿੱਤਾ ਹੈ ।

(lālā sāhib nē āpaṇā jīvan is kamm dē laī arpaṇ kar dittā hai).

Input: अपना बिस्तर खिड़की से सटाकर कभी भी न लगाएँ।

(apnā bistar khiṛkī sē saṭākar kabhī bhī na lagāēṃ.)

Output: ਆਪਣਾ ਬਿਸਤਰਾ ਖਿੜਕੀ ਤੋਂ ਸਟਾਕਰ ਕਦੇ ਵੀ ਨਹੀਂ ਲਗਾਓ ।

(āpaṇā bisatrā khiḍ◌ kī tōṃ saṭākar kadē vī nahīṃ lagāō .)

Input: शरीफ लाहौर से इस्लामाबाद के लिए निकले थे ।

(sharīph lāhaur sē islāmābād kē liē niklē thē .)

Output: ਸ਼ਰੀਫ ਲਾਹੌਰ ਤੋਂ ਇਸਲਾਮਾਬਾਦ ਲਈ ਨਿਕਲੇ ਸਨ ।

(sharīph lāhaur tōṃ islāmābād laī niklē san).

6.4 Illustrative Example:

For illustration purpose of how the input is passed through various phases consider the following sentence in Hindi:

Input: हम अपने दोस्त दीपक शर्मा से पूछेंगे कि क्या वो हमारे साथ वन डे मैच ख़ेल कर गरीब बच्चों की मदद करना चाहेगा जैसे हम करना चाहते हैं ।

(ham apnē dōst dīpak sharmā sē pūchēṅgē ki kyā vō hamārē sāth van ḍē maic khaēl kar garīb baccōṃ kī madad karnā cāhēgā jaisē ham karnā cāhtē haiṃ).

Pre Processing Phase:

- During the Text Normalization, ख़ेल (*khaēl)* will be replaced with खेल *(khēl)*
- No named Entity is found.
- वन डे (*van ḍē)* is collocation , thus it will be replaced with ਵਨ ਡੇ (*van ḍē)* rather than it's actual word to word translation ਜੰਗਲ ਡੇ (*jaṅgal ḍē)*

Intermediate Output: हम अपने दोस्त दीपक शर्मा से पूछेंगे कि क्या वो हमारे साथ ਵਨ ਡੇ मैच खेल कर गरीब बच्चों की मदद करना चाहेगा जैसे हम करना चाहते हैं

(ham apnē dōst dīpak sharmā sē pūchēṅgē ki kyā vō hamārē sāth van ḍē maic khēl kar garīb baccōṃ kī madad karnā cāhēgā jaisē ham karnā cāhtē haiṃ)

Tokenizer: It will generate हम, अपने, दोस्त, दीपक, शर्मा, से, पूछेंगे, कि, क्या, वो, हमारे, साथ, वन, डे, मैच, खेल, कर, गरीब, बच्चों, की, मदद, करना, चाहेगा, जैसे, हम, करना, चाहते, हैं,।. These tokens are generated one by one and passed to translation engine for processing one after another.

Translation Engine:

- No Title is found
- Surname शर्मा is found and thus दीपक शर्मा will be translated to ਦੀਪਕ ਸ਼ਰਮਾ rather than ਦੀਵਾ ਸ਼ਰਮਾ
- हम, अपने, दोस्त, पूछेंगे, कि, क्या, वो, हमारे, साथ, खेल, गरीब, बच्चों, की, मदद, करना, चाहेगा, जैसे, हम, करना, चाहते, हैं,। tokens will be translated using lexicon lookup and will be translated to ਅਸੀਂ, ਆਪਣੇ , ਦੋਸਤ , ਪੁੱਛਾਂਗੇ, ਕਿ, ਕੀ, ਉਹ, ਸਾਡੇ, ਨਾਲ, ਖੇਲ, ਗਰੀਬ, ਬੱਚਿਆਂ, ਕੀਤੀ, ਮਦਦ, ਕਰਨਾ, ਚਾਹੇਗਾ, ਜਿਵੇਂ, ਅਸੀਂ, ਕਰਨਾ, ਚਾਹੁੰਦੇ, ਹਨ,। respectively.
- Tokens से and कर are ambiguous words and hence using word sense disambiguation approach, these will be translated to ਤੋਂ and ਕੇ respectively.
- मैच will be transliterated by the transliteration module to ਮੈਚ.

Intermediate Output: ਅਸੀਂ ਆਪਣੇ ਦੋਸਤ ਦੀਪਕ ਸ਼ਰਮਾ ਤੋਂ ਪੁੱਛਾਂਗੇ ਕਿ ਕੀ ਉਹ ਸਾਡੇ ਨਾਲ ਵਨ ਡੇ ਮੈਚ ਖੇਲ ਕੇ ਗਰੀਬ ਬੱਚਿਆਂ ਦੀ ਮਦਦ ਕਰਨਾ ਚਾਹੇਗਾ ਜਿਵੇਂ ਅਸੀਂ ਕਰਨਾ ਚਾਹੁੰਦੇ ਹਨ।

(asīṃ āpaṇē dōsat dīpak sharmā tōṃ pucchāṅgē ki kī uh sāḍē nāl van ḍē maic khēl kē garīb bacciāṃ dī madad karnā cāhēgā jivēṃ asīṃ karnā cāhundē han.)

Post Processing:

Pattern matching is done using regular expressions for correcting grammar. Thus, this output will be made grammatically correct using the regular expressions implementing the grammatical agreements. In this example, subject verb agreement will be done and ਹਨ will be replaced with ਹਾਂ.

Final Output: ਅਸੀਂ ਆਪਣੇ ਦੋਸਤ ਦੀਪਕ ਸ਼ਰਮਾ ਤੋਂ ਪੁੱਛਾਂਗੇ ਕਿ ਕੀ ਉਹ ਸਾਡੇ ਨਾਲ ਵਨ ਡੇ ਮੈਚ ਖੇਲ ਕੇ ਗਰੀਬ ਬੱਚਿਆਂ ਦੀ ਮਦਦ ਕਰਨਾ ਚਾਹੇਗਾ ਜਿਵੇਂ ਅਸੀਂ ਕਰਨਾ ਚਾਹੁੰਦੇ ਹਾਂ ।

(asīṃ āpaṇē dōsat dīpak sharmā tōṃ pucchāṅgē ki kī uh sāḍē nāl van ḍē maic khēl kē garīb bacciāṃ dī madad karnā cāhēgā jivēṃ asīṃ karnā cāhundē hāṃ .)

Evaluation:

The evaluation document set consisted of documents from various online newspapers news, articles, blogs, biographies etc. This test bed consisted of 35500 words and was translated using our Machine Translation system. Following table shows the contribution of various important modules of the system during translation:

Table 6.3 : % Contribution of Various MT System Modules during Translation

Module	Submodule	Contribution (%)
Preprocessing	Text Normalization	1.121%

	Replacing Collocations	0.281%
	Replacing Proper Nouns	1.408%
Translation Engine	Identifying Titles	0.005%
	Identifying Surnames	3.380%
	Word-to-word translation using Lexicon Lookup	50.949%
	Word Sense disambiguation	7.140%
	Word Inflectional analysis and generation	6.45%
	Transliteration	23.239%
Post Processing	Grammar Correction	6.197%

Following is the output translation for the text downloaded from website - http://www.bbc.co.uk/hindi/india/2010/01/100104_india_cold_va.shtml accessed on dated 04-01-2010 :

Input Text:

उत्तरी भारत में ठंड का प्रकोप जारी है. सरकारी टेलीविज़न के मुताबिक ठंड के कारण अब तक 100 लोगों की मौत हो चुकी है. हिमाचल प्रदेश और कश्मीर घाटी में बर्फ़बारी हुई है और कई इलाक़ों में घना कोहरा छाया हुआ है. पंजाब, हरियाणा और उत्तर प्रदेश में भी कड़ाके के ठंड पड़ रही है. पिछले कुछ दिनों से छाए कोहरे की वजह से कई उड़ानें और ट्रेनें रद्द करनी पड़ी हैं. उत्तर प्रदेश में तापमान सामान्य से दो से 10 डिग्री कम चल रहा है.

गिरता पारा

वहीं राजधानी दिल्ली में भी काफ़ी सर्दी है. दिल्ली में करीब 20 ट्रेनें रद्द कर दी गई हैं. रविवार को न्यूनतम तापमान 9.5 डिग्री सेल्सियस था जबकि शनिवार को तापमान 8.4 रहा. उधर अमृतसर में रविवार रात को पारा मात्र 0.8 डिग्री सेल्सियस रहा जबकि चंडीगढ़ में तापमान 9.4 डिग्री सेल्सियस था. ठंड की वजह से कई जगह स्कूलों में छुट्टियाँ बढ़ा दी गई हैं. कोहरे के कारण कई जगह दुर्घटनाएँ भी हो रही हैं.शनिवार को दो रेल हादसों में कम से कम 10 लोग मारे गए थे. कोहरे से बिजली आपूर्ति भी प्रभावित हुई थी और उत्तरी ग्रिड फेल हो गया था.

uttarī bhārat mēṃ ṭhaṇḍ kā prakōp jārī hai. sarkārī ṭēlīvija◌ॅn kē mutābik ṭhaṇḍ kē kāraṇ ab tak 100 lōgōṃ kī maut hō cukī hai. himācal pradēsh aur kashmīr ghāṭī mēṃ barphbārī huī hai aur kaī ilākōṃ mēṃ ghanā kōhrā chāyā huā hai. pañjāb, hariyāṇā aur uttar pradēsh mēṃ bhī kaḍaākē kē ṭhaṇḍ paḍa◌ॅ rahī hai. pichlē kuch dinōṃ sē chāē kōhrē kī vajah sē kaī uḍaānēṃ aur ṭrēnēṃ radd karnī paḍaī haiṃ. uttar pradēsh mēṃ tāpmān sāmāny sē dō sē 10 ḍigrī kam cal rahā hai.

girtā pārā

vahīṃ rājdhānī dillī mēṃ bhī kāphī sardī hai. dillī mēṃ karīb 20 ṭrēnēṃ radd kar dī gaī haiṃ. ravivār kō nyūntam tāpmān 9.5 ḍigrī sēlsiyas thā jabki shanivār kō tāpmān 8.4 rahā. udhar amrtasar mēṃ ravivār rāt kō pārā mātr 0.8 ḍigrī sēlsiyas rahā jabki caṇḍīgḍha◌ॅ mēṃ tāpmān 9.4 ḍigrī sēlsiyas thā. ṭhaṇḍ kī vajah sē kaī jagah skūlōṃ mēṃ chuṭṭiyāṃ baḍhā dī gaī haiṃ. kōhrē

kē kāraṇ kaī jagah durghṭanāēṃ bhī hō rahī haiṃ.shanivār kō dō rēl hādsōṃ mēṃ kam sē kam 10 lōg mārē gaē thē. kōhrē sē bijlī āpūrti bhī prabhāvit huī thī aur uttarī griḍ phēl hō gayā thā.

Output text:

ਉੱਤਰੀ ਭਾਰਤ ਵਿੱਚ ਠੰਡ ਦਾ ਕਹਿਰ ਜਾਰੀ ਹੈ . ਸਰਕਾਰੀ ਟੇਲੀਵਿਜਨ ਦੇ ਮੁਤਾਬਕ ਠੰਡ ਦੇ ਕਾਰਨ ਹੁਣ ਤੱਕ 100 ਲੋਕਾਂ ਦੀ ਮੌਤ ਹੋ ਚੁੱਕੀ ਹੈ . ਹਿਮਾਚਲ ਪ੍ਰਦੇਸ਼ ਅਤੇ ਕਸ਼ਮੀਰ ਘਾਟੀ ਵਿੱਚ ਬਰਫਬਾਰੀ ਹੋਈ ਹੈ ਅਤੇ ਕਈ ਇਲਾਕੀਆਂ ਵਿੱਚ ਸੰਘਣਾ ਕੋਹਰਾ ਛਾਇਆ ਹੋਇਆ ਹੈ . ਪੰਜਾਬ , ਹਰਿਆਣਾ ਅਤੇ ਉੱਤਰ ਪ੍ਰਦੇਸ਼ ਵਿੱਚ ਵੀ ਕੜਾਕੇ ਦੇ ਠੰਡ ਪੈ ਰਹੀ ਹੈ . ਪਿਛਲੇ ਕੁੱਝ ਦਿਨਾਂ ਤੋਂ ਛਾਏ ਕੋਹਰੇ ਦੀ ਵਜ੍ਹਾ ਕਰਕੇ ਕਈ ਉਡ਼ਾਨੇਂ ਹੋਰ ਟਰੇਨਾਂ ਰੱਦ ਕਰਣੀ ਪਈਆਂ ਹਨ . ਉੱਤਰ ਪ੍ਰਦੇਸ਼ ਵਿੱਚ ਤਾਪਮਾਨ ਇੱਕੋ ਜਿਹੇ ਤੋਂ ਦੋ ਤੋਂ 10 ਡਿਗਰੀ ਘੱਟ ਚੱਲ ਰਿਹਾ ਹੈ .

ਡਿੱਗਦਾ ਪਾਰਾ

ਉਥੇ ਹੀ ਰਾਜਧਾਨੀ ਦਿੱਲੀ ਵਿੱਚ ਵੀ ਕਾਫ਼ੀ ਸਰਦੀ ਹੈ . ਦਿੱਲੀ ਵਿੱਚ ਕਰੀਬ 20 ਟਰੇਨਾਂ ਰੱਦ ਕਰ ਦਿੱਤੀ ਗਈਆਂ ਹਨ . ਐਤਵਾਰ ਨੂੰ ਹੇਠਲਾ ਤਾਪਮਾਨ 9.5 ਡਿਗਰੀ ਸੇਲਸਿਅਸ ਸੀ ਜਦੋਂ ਕਿ ਸ਼ਨੀਵਾਰ ਨੂੰ ਤਾਪਮਾਨ 8.4 ਰਿਹਾ . ਉੱਧਰ ਅਮ੍ਰਿਤਸਰ ਵਿੱਚ ਐਤਵਾਰ ਰਾਤ ਨੂੰ ਪਾਰਾ ਸਿਰਫ 0.8 ਡਿਗਰੀ ਸੇਲਸਿਅਸ ਰਿਹਾ ਜਦੋਂ ਕਿ ਚੰਡੀਗੜ ਵਿੱਚ ਤਾਪਮਾਨ 9.4 ਡਿਗਰੀ ਸੇਲਸਿਅਸ ਸੀ . ਠੰਡ ਦੀ ਵਜ੍ਹਾ ਕਰਕੇ ਕਈ ਜਗ੍ਹਾ ਸਕੂਲਾਂ ਵਿੱਚ ਛੁੱਟੀਆਂ ਵਧਾ ਦਿੱਤੀ ਗਈਆਂ ਹਨ . ਕੋਹਰੇ ਦੇ ਕਾਰਨ ਕਈ ਜਗ੍ਹਾ ਦੁਰਘਟਨਾਵਾਂ ਵੀ ਹੋ ਰਹੀ ਹਨ . ਸ਼ਨੀਵਾਰ ਨੂੰ ਦੋ ਰੇਲ ਹਾਦਸਿਆਂ ਵਿੱਚ ਘੱਟ ਤੋਂ ਘੱਟ 10 ਲੋਕ ਮਾਰੇ ਗਏ ਸਨ . ਕੋਹਰੇ ਤੋਂ ਬਿਜਲੀ ਆਪੂਰਤੀ ਵੀ ਪ੍ਰਭਾਵਿਤ ਹੋਈ ਸੀ ਅਤੇ ਉੱਤਰੀ ਗਰਿਡ ਫੇਲ ਹੋ ਗਿਆ ਸੀ .

uttarī bhārat vicc ṭhaṇḍ dā kahir jārī hai . sarkārī ṭēlīvijan dē mutābak ṭhaṇḍ dē kāran huṇ takk 100 lōkāṃ dī maut hō cukkī hai . himācal pradēsh atē kashmīr ghāṭī vicc barphabārī hōī hai atē kaī ilākīāṃ vicc saṅghaṇā kōhrā chāiā hōiā hai . pañjāb , hariāṇā atē uttar pradēsh vicc vī kaṛākē dē ṭhaṇḍ pai rahī hai . pichlē kujjh dināṃ tōṃ chāē kōhrē dī vajhā karkē kaī uṛānēṃ hōr ṭarēnāṃ radd karṇī paīāṃ han . uttar pradēsh vicc tāpmān ikkō jihē tōṃ dō tōṃ 10 ḍigrī ghaṭṭ call rihā hai .

ḍiggdā pārā

uthē hī rājdhānī dillī vicc vī kāfī sardī hai . dillī vicc karīb 20 ṭarēnāṃ radd kar dittī gaīāṃ han . aitvār nūṃ hēṭhlā tāpmān 9.5 ḍigrī sēlsias sī jadōṃ ki shanīvār nūṃ tāpmān 8.4 rihā . uddhar amritsar vicc aitvār rāt nūṃ pārā siraph 0.8 ḍigrī sēlsias rihā jadōṃ ki caṇḍigaṛ vicc tāpmān 9.4 ḍigrī sēlsias sī . ṭhaṇḍ dī vajhā karkē kaī jaghā sakūlāṃ vicc chuṭṭīāṃ vadhā dittī gaīāṃ han . kōhrē dē kāran kaī jaghā durghaṭnāvāṃ vī hō rahī han . shanīvār nūṃ dō rēl hādsiāṃ vicc ghaṭṭ tōṃ ghaṭṭ 10 lōk mārē gaē san . kōhrē tōṃ bijlī āpūrtī vī prabhāvit hōī sī atē uttarī gariḍ phēl hō giā sī .

6.5 Summary

This chapter discusses in detail the post-processing phase. This phase involves the rules that are applied on the output produced by previous phases. The various grammatical errors corrected by this phase are also discussed. The implementation of whole system is also discussed along with illustrative example.

Translations for text from various sources like news items, stories, blogs, office orders, articles etc. are obtained from this system and made available to the evaluators for the evaluation purpose. In the next chapters, we will discuss the evaluation and results of our system, for the language pair of Hindi and Punjabi.

Chapter 7

Evaluation and Results

7.1 Introduction

Evaluation of a MT system is as important as the MT itself, answering the questions about the accuracy, fluency and acceptability of the translation and thus artifying the underlying MT algorithm. Evaluation has long been a tough task in the development of MT systems because there may exist more than one correct translations of the given sentence. The problem with natural language is that language is not exact in the way that mathematical models and theories in science are. While there is general agreement about the basic features of Machine Translation (MT) evaluation (as reflected in general introductory texts Lehrberger & Bourbeau, 1988; Hutchins & Somers, 1992; Arnold et al., 1994), there are no universally accepted and reliable methods and measures, and evaluation methodology has been the subject of much discussion (e.g. Arnold et al., 1993; Falkedal, 1994, AMTA, 1992). As in other areas of NLP, three types of evaluation are recognised:

- Adequacy evaluation to determine the fitness of MT systems within a specified operational context. It is typically performed by potential users and/or purchasers of systems (individuals, companies, or agencies). Adequacy evaluations usually include the testing of systems with sets

of *typical* documents. But these are necessarily restricted to specific domains.

- Diagnostic evaluation to identify limitations, errors and deficiencies, which may be corrected or improved (by the research team or by the developers). It is the concern mainly of researchers and developers.
- Performance evaluation to assess stages of system development or different technical implementations. It may be undertaken by either researchers/developers or by potential users.

MT evaluations typically include features not present in evaluations of other NLP systems: the quality of the *raw* (unedited) translations, e.g. intelligibility, accuracy, fidelity, appropriateness of style/register; the usability of facilities for creating and updating dictionaries, for post-editing texts, for controlling input language, for customisation of documents, etc.; the extendibility to new language pairs and/or new subject domains; and cost-benefit comparisons with human translation performance.

7.2 Related Works:

Several researchers have worked on evaluation techniques of Machine Translation systems and many measures and methods have been developed for this purpose. Attempts have been made to produce well designed and well founded evaluation schemes. Initially, MT evaluation was seen primarily in terms of comparisons of unedited MT output quality and human translations, e.g. the ALPAC evaluations [3] and those of the original Logos system

[30,31]. Later, systems were assessed for quality of output and usefulness in operational contexts, e.g., the influential evaluations of Systran by the European Commission [102]. SYSTRAN [35,227,228] and Logos [30,31] have developed internal evaluation methods to compare results given by different versions of their own systems. Palmira Marrafa and Antonio Ribero [92] proposed quantitative metrics for evaluations based on the number of errors in an evaluation and the total number of possible errors. Rita Nüebel [229] presents a blueprint for a strictly user-driven approach to MT evaluation within a net-based MT scenario, which can also be adapted to developer-driven evaluations. The Van Slype report for the European Commission [102] provided a very thorough critical survey of evaluations. Eagles Evaluation Group [230] also worked to establish standards in the field to come up with a theoretically sound framework for evaluation of a Machine Translation system. However, no consensus has ever been reached in defining one single evaluation procedure, applicable to a Machine Translation system in all circumstances. Valuable contributions to MT evaluation methodology have been made by Rinsche (1993) in her study for the European Commission, and by the JEIDA committee (Nomura & Isahara, 1992), which proposed evaluation tools for both system developers and potential users .The evaluation exercise by ARPA (White et al., 1994) compared the unedited output of the three APRA-supported experimental systems (Pangloss, Candide, Lingstat) with the output from 13 production systems from Globalink, PC-Translator, Microtac, Pivot, PAHO, Metal, Socatra XLT, Systran, and

Winger. The initial intention to measure the productivity of systems for potential users was abandoned because it introduced too many variables. Evaluation, therefore, has concentrated on the performance of the core MT engines of systems, in comparison to human translations, using measures of adequacy (how well a text fragment conveys the information of the source), fluency (whether the output reads like good English, irrespective of accuracy), and comprehension or informativeness (using SAT-like multiple choice tests covering the whole text). Roudaud [231] discusses in detail the procedure for the evaluation and improvement of an MT system by the end users. He describes the different types of problems encountered and categories them. Simone Wagner [232] suggested four methods viz. percentage of correct sentences, no. of errors, Intelligibility, Accuracy, and time taken to do post editing, which concentrates on linguistic performance of the system. He claims that these evaluation methods do not require expert linguistic knowledge and can be performed in quite short time. However, not all of them were equally suited for a comparative evaluation. Keiji *et al*. [233] evaluates the translation output by measuring the similarity between the translation output and translation answer candidates from a parallel corpus. Yasuhiro *et al*. [234] use multiple edit distances to automatically rank Machine Translation output by translation examples. While the IBM BLEU method Papineni *et al.* [235] and the NIST MT evaluation [236] compare MT output with expert reference translations in terms of the statistics of word *N-gram*s. Melamed *et al.* [237] adopt the maximum matching size of the translation and the

reference as the similarity measure for the score. Nieben and Och [238] score a sentence on the basis of scores of translations in a database with the smallest edit distance. Yokoyama *et al.* [239] propose a two-way MT based evaluation method, which compares output Japanese sentences with the original Japanese sentence for word identification and the correctness of the modification.

7.3 Our Approach:

For our purpose following steps have been performed for evaluating the system that is discussed in detail as follows:

7.3.1 Selection Set of Sentences:

It is very important aspect in MT evaluation to make appropriate selection of the sentences for evaluating the Machine Translation system. According to Lorna Balkan [240,241], There are basically three types of test materials:

Test Corpora: It is a collection of naturally occurring text, increasingly in electronic form.

Test Suites: It is a collection of usually artificially constructed inputs, where each input is designed to probe a system's treatment of a specific phenomenon or set of phenomena. Inputs may be in the form of sentences, sentence fragments, or even sequences of sentences. Test suites are useful for presenting language phenomena and combinations of phenomena in an exhaustive and systematic way. Furthermore, negative data can be derived systematically from positive data by violating grammatical constraints associated with the positive data item.

Test Collections: It is a set of inputs associated with a corresponding set of expected outputs. This type of test material is increasingly common and has been used in the evaluation of parsers and other Natural Language Processing applications. The problem with test collections is that of being able to specify an appropriate output for a system. Output from parsers can be many and varied. The Parseval project, in common with other parser evaluation projects, uses hand-produced ideal parses of sentences from the Penn Treebank, a parsed corpus, to compare parser output against. Machine Translation shares a similar problem - there is no one correct output. While at present no test collections exist for MT, it is possible to imagine producing an ideal translation, in the same way as an ideal parse.

There are several issues involved in the selection of set of sentences for a comprehensive evaluation. For example, the set could be constant, variable or a mixed one; the number of sentence may be small or large, the collection of sentences may be domain specific or generic. It is obvious that there is no guarantee that even the bulkiest sample will include all the possible syntactic structures of the source language. Elliott et al. [242] describes the text limit to include in a corpus for MT evaluation, given the general hypothesis that more text would lead to more reliable scores. The author, on the basis of an empirical assessment of score variation, estimates that systems could be reliably ranked with around 40 texts (ca. 14,000 words). Zhang and Vogel [243] also studied the influence of the amount of test data on the reliability of automatic metrics, focusing on confidence intervals for

BLEU and NIST scores. Estrella P. et. al. [244] show that for human or automatic evaluation about five documents from the same domain—with ca. 250 segments or 6,000 words—seem sufficient to establish the ranking of the systems and about ten documents are sufficient to obtain reliable scores.

For our Machine Translation system evaluation, we have used benchmark sampling method for selecting the set of sentences. Input sentences are selected from randomly selected news (sports, politics, world, regional, entertainment, travel etc.), articles (published by various writers, philosophers etc.), literature (stories by Prem Chand, Yashwant jain etc.), Official language for office letters (The Language Officially used on the files in Government offices) and blogs (Posted by general public in forums etc.). Care has been taken to ensure that sentences use a variety of constructs. All possible constructs including simple as well as complex ones are incorporated in the set. The sentence set also contains all types of sentences such as declarative, interrogative, imperative and exclamatory. Sentence length is not restricted although care has been taken that single sentences do not become too long. Following table shows the test data set:

Table 7.1: Test data set for the evaluation of Hindi to Punjabi Machine Translation System

	Daily News	**Articles**	**Official Language Quotes**	**Blog**	**Literature**
Total Documents	100	50	01	50	20

Total Sentences	10,000	3,500	8,595	3,300	10,045
Total Words	93,400	21,674	36,431	15,650	95,580

7.3.2. Selection of Tests for Evaluation

There are number of tests available for evaluating the Machine Translation systems. Van Slype [102] describes that the selection of tests for MT evaluation depends upon the target users of the MT system. The main aim of our system is effective transfer of information from Hindi to Punjabi language. Thus, Subjective tests and Error diagnosis/analysis have been selected for our MT System evaluation. Subjective Tests include intelligibility test, Accuracy Test / Fidelity rating and BLUE Scoring. Some Quantitative Metrics have also been evaluated through error analysis / diagnosis by calculating Sentence Error Rate (SER) and Word Error Rate (WER). These tests are discussed in detail in following sections.

7.3.2.1 Subjective Tests

7.3.2.1.1 Intelligibility Test

This test is used to check the intelligibility of the MT System. Van Slype Georges [102] describes intelligibility as a measure of how understandable the sentence is. Intelligibility is measured without the reference to the original sentence. It tells the degree of comprehensibility and clarity of the translation. Intelligibility is effected by grammatical errors, mis-translations, and un-translated words. The scoring methodology for intelligibility test has been adopted described by Van Slype Georges [102]. Each evaluator receives a

series of sentences in sequence i.e. sentence in their context. Literature shows variations in selecting the point scales. It has been observed that a scale comprising a very low number of points seems insufficiently discriminatory. On the other hand, a scale comprising a high number of points, assessment of which remains in the final analysis subjective, involves too wide a scatter of the ratings. Furthermore, to clarify in detail each of the possible values of the scale, there is a risk of introducing elements not germane to intelligibility. Thus, it is concluded that four points scale is most adequate, in that it measures intelligibility only, has a low scatter and is of a sufficiently discriminatory character since the evaluation covers several hundreds of sentences and the average calculated as a percentage is sufficiently precise. Hence, a four point scale is made in which highest point is assigned to those sentences that look perfectly clear and intelligible and lowest point is assigned to the sentence which is non understandable. The scale looks like:

Table 7.2 Score Sheet for Intelligibility Test

Score	Significance
3	The sentence is perfectly clear and intelligible. It is grammatically correct.
2	The sentence is generally clear and intelligible. Despite some inaccuracies, one can understand the information to be conveyed.
1	The general idea is intelligible only after considerable study. The sentence contains grammatical errors and/or poor word choice.

0	The sentence is unintelligible. The meaning of the sentence is not understandable.

7.3.2.1.2 Accuracy Test / Fidelity Measure

Accuracy Test or Fidelity measure is a measure of how much information the translated sentence retained compared to the original. It is measured indirectly. The evaluator is asked to gather whatever meaning he could from the translation sentence and then evaluate the original sentence for its "informativeness" in relation to what he had understood from the translation sentence. Thus, a rating of the original sentence as "highly informative" in relation to the translated sentence would imply that the latter was lacking in fidelity/accuracy. Halliday [245] define it as the Measurement of the correctness of the information transferred from the source language to the target language. Van Slype Georges [102] describes it as a subjective evaluation of the measure in which the information contained in the sentence of the original text reappears without distortion in the translation. Analogous to the Intelligibility test, the methodology described by Van Slype Georges [102] is adopted for the accuracy test also. A Four point scale is made in which highest point is assigned to those sentences that are completely faithful and lowest point is assigned to the sentence which is un-understandable and unacceptable. The scale looks like:

Table 7.3 Score Sheet for Accuracy Test

Score	Significance
3	Completely faithful
2	Fairly faithful: more than 50 % of the original information passes in the translation.
1	Barely faithful: less than 50 % of the original information passes in the translation.
0	Completely unfaithful. Doesn't make sense.

These both scales i.e. Intelligibility and Accuracy test scales have already been used in the evaluation of the SYSTRAN English-French MT system acquired by the Commission of the European Communities.

7.3.2.1.3 BLEU Score

Bilingual Evaluation Understudy or BLEU [246] is one of the most popular metric for automatically evaluating Machine Translation system output quality. The central idea behind this metric is the closer a Machine Translation is to a professional human translation, the better it is. The primary programming task in a BLEU implementation is to compare n-grams of the candidate with the n-grams of the reference translation and count the number of matches. These matches are position-independent. The more the matches, the better is the candidate translation. The metric calculates scores for individual segments, generally sentences, and then averages these scores over the whole corpus in order to reach a final score. It has been shown to correlate highly with human judgments of quality at the corpus level. The quality of translation is

indicated as a number between 0 and 1 and is measured as statistical closeness to a given set of good quality human reference translations. Therefore, it does not directly take into account translation intelligibility or grammatical correctness. The metric works by measuring the n-gram co-occurrence between a given translation and the set of reference translations. Then the weighted geometric mean is calculated.

7.3.3 Evaluation based on Quantitative Metrics

Rather than using broad indicators as guides to score assignments, we must also focus on the errors made by the MT system. Quantitative metrics play major role in it. It includes the technique of error analysis that tries to establish how seriously errors affect the translation output. The error analysis includes calculating Word Error Rate (WER) and Sentence Error Rate (SER). Word Error Rate (WER) is defined as percentage of words which are to be inserted, deleted, or replaced in the translation in order to obtain the sentence of reference. Sentence Error Rate (SER) is defined as percentage of sentences, whose translations have not matched in an exact manner with those of reference.

7.3.4 Experiments

It is also important to choose appropriate evaluators for our experiments. Thus, depending upon the requirements and need of the above mentioned tests, 50 People of different professions were selected for performing

experiments. 20 Persons were from villages that only knew Punjabi and did not know Hindi and 30 persons were from different professions having knowledge of both Hindi and Punjabi. Average ratings for the sentences of the individual translations were then summed up (separately according to intelligibility and accuracy) to get the average scores. Percentage of accurate sentences and intelligent sentences was also calculated separately by counting the number of sentences.

7.3.4.1 Intelligibility Evaluation

The evaluators do not have any clue about the source language i.e. Hindi. They judge each sentence (in target language i.e. Punjabi) on the basis of its comprehensibility. The target user is a layman who is interested only in the comprehensibility of translations. Intelligibility is effected by grammatical errors, mis-translations, and un-translated words.

7.3.4.1.1 Scoring

The scoring is done based on the degree of intelligibility and comprehensibility. A four point scale is made in which highest point is assigned to those sentences that look perfectly alike the target language and lowest point is assigned to the sentence which is un-understandable. Detail is as follows:

Score 3: The sentence is perfectly clear and intelligible. It is grammatical and reads like ordinary text.

Score 2: The sentence is generally clear and intelligible. Despite some inaccuracies, one can understand immediately what it means.

Score 1: The general idea is intelligible only after considerable study. The sentence contains grammatical errors &/or poor word choice.

Score 0: The sentence is unintelligible. Studying the meaning of the sentence is hopeless. Even allowing for context, one feels that guessing would be too unreliable.

7.3.4.1.2 Results

The response by the evaluators were analysed and following are the results:

- 70.3 % sentences got the score 3 i.e. they were perfectly clear and intelligible.
- 25.1 % sentences got the score 2 i.e. they were generally clear and intelligible.
- 3.5 % sentences got the score 1 i.e. they were hard to understand.
- 1.1 % sentences got the score 0 i.e. they were not understandable.

So we can say that about 95.40 % sentences are intelligible. These sentences are those which have score 2 or above. Thus, we can say that the direct approach can translate Hindi text to Punjabi Text with a consideably good accuracy.

7.3.4.1.3 Percentage Intelligibility:

Following graph shows that percentage intelligibility of individual documents:

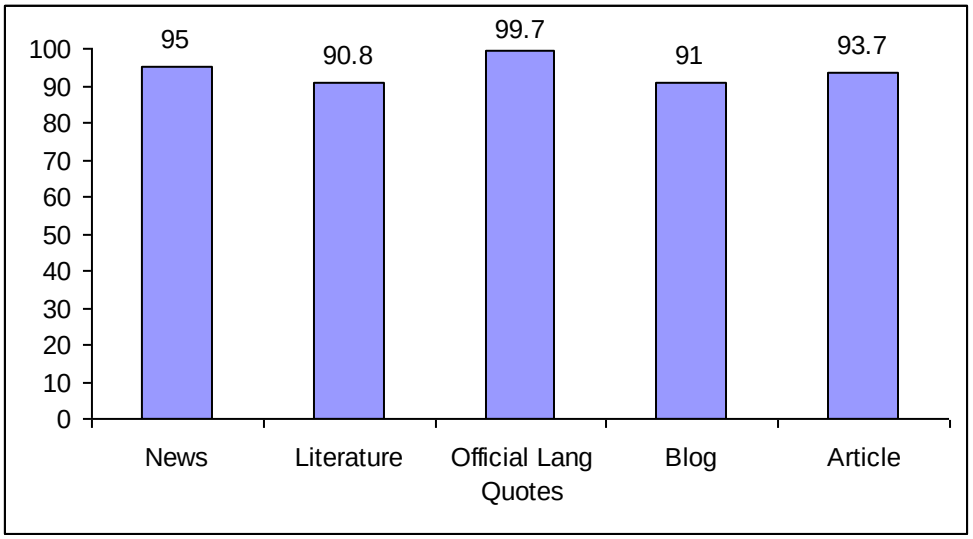

Figure 7.1: Percentage Intelligibility for Different Documents

7.3.4.1.4 Analysis

The main reason behind less accuracy for literature documents is due to the language dialect used by the writer of the stories. Some writers use Rajasthani language, some use Haryanavi dialect. And this resulted in less translation accuracy for this category. Otherwise for rest of the four categories, the quality of translation is better than other systems which will be discussed in following sections.

7.3.4.2 Accuracy Evaluation / Fidelity Measure

The evaluators are provided with source text along with translated text. A highly intelligible output sentence need not be a correct translation of the source sentence. It is important to check whether the meaning of the source

language sentence is preserved in the translation. This property is called accuracy.

7.3.4.2.1 Scoring:

The scoring is done based on the degree of intelligibility and comprehensibility. A four point scale is made in which highest point is assigned to those sentences that look perfectly like the target language and lowest point is assigned to the sentence which is not understandable and unacceptable. The scale looks like:

Score 3 : Completely faithful

Score 2: Fairly faithful: more than 50 % of the original information passes in the translation.

Score 1: Barely faithful: less than 50 % of the original information passes in the translation.

Score 0: Completely unfaithful. It doesn't make any sense.

7.3.4.2.2 Results

Initially Null Hypothesis is assumed i.e. the system's performance is NULL. The author assumes that system is dumb and does not produce any valuable output. By the intelligibility of the analysis and Accuracy analysis, it has been proved wrong.

The accuracy percentage for the system is found out to be 87.60%

Further investigations reveal that out of 13.40%:

- 80.6 % sentences achieve a match between 50 to 99%
- 17.2 % of remaining sentences were marked with less than 50% match against the correct sentences.
- Only 2.2 % sentences are those which are found unfaithful.

A match of lower 50% does not mean that the sentences are not usable. After some post editing, they can fit properly in the translated text.

7.3.4.2.3 Percentage Accuracy:

Following graph shows that percentage accuracy of individual documents:

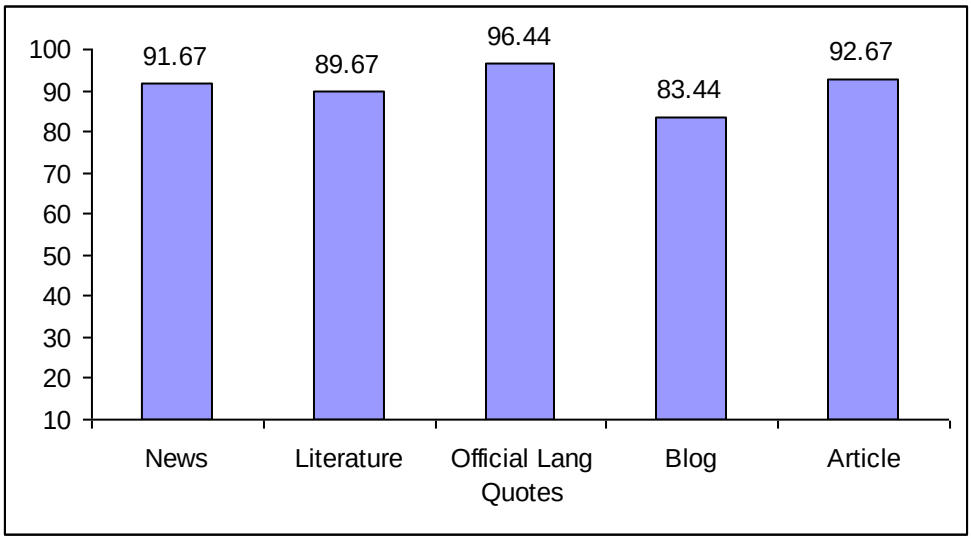

Figure 7.2: Percentage Accuracy for Different Documents

7.3.4.2.4 Analysis

The overall performance accuracy test of the system is quite good. But for Blog it is less than others. The reason is the use of slang which causes the failure of the translation software as the slang available in one language is not present in the other language. Also un-standardized language causes more ambiguities.

7.3.5 Error Analysis

Error analysis is done against pre classified error list. All the errors in translated text were identified and their frequencies were noted. Errors were just counted and not weighted. In the following sections, the experiments conducted for Word Error Analysis and Sentence Error Analysis will be explained.

7.3.5.1 Word Error Analysis

After robust analysis, Word Error rate is found to be 4.58% which is comparably lower than that of general systems, where it ranges from 9.5 to 12%[231,237,238]. Following figure shows the percentage type of errors out of the errors found:

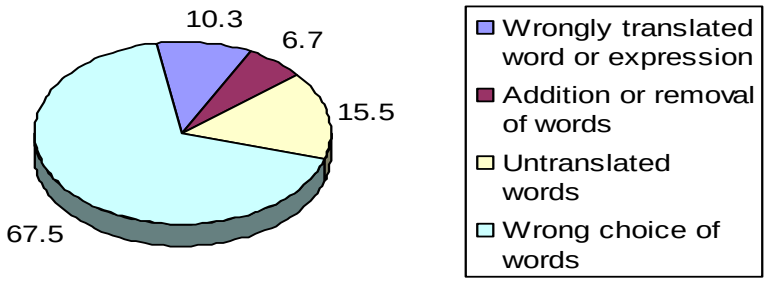

Figure 7.3: Percentage Distribution of Errors

From the above figure, it is concluded that majority of the errors are due to wrong choice of words, means the WSD module of the system must be

improved. Further, the bilingual dictionary improvements can reduce the wrongly translated and untranslated words errors.

Word Error Rate Percentage:

Following graph shows the Word Error Rate for different articles:

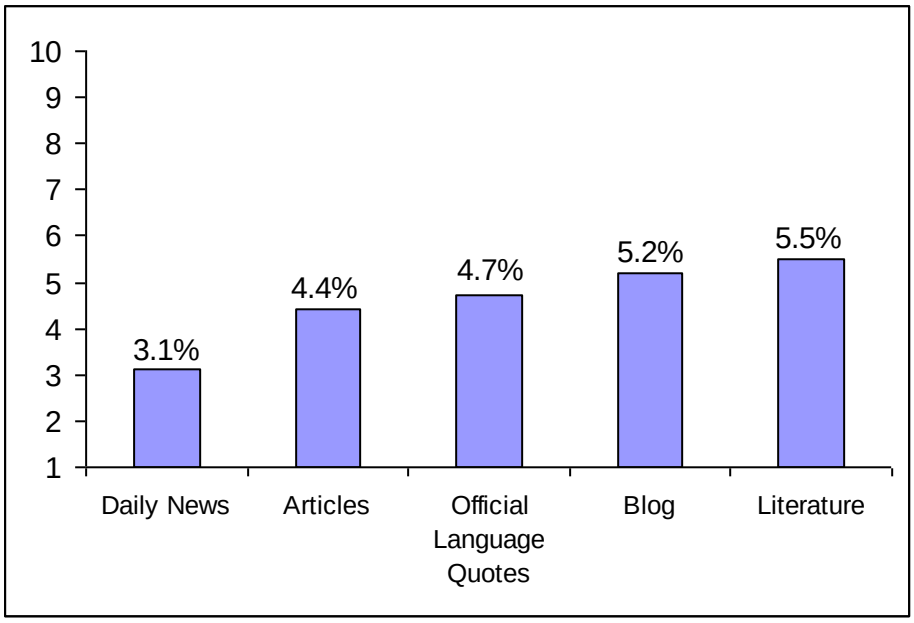

Figure 7.4: Word Error Rate for Different Documents

7.3.5.2 Sentence Error Analysis:

The Sentence error rate comes out to be 28.82%. Following graph shows the Word Error for different arcticles:

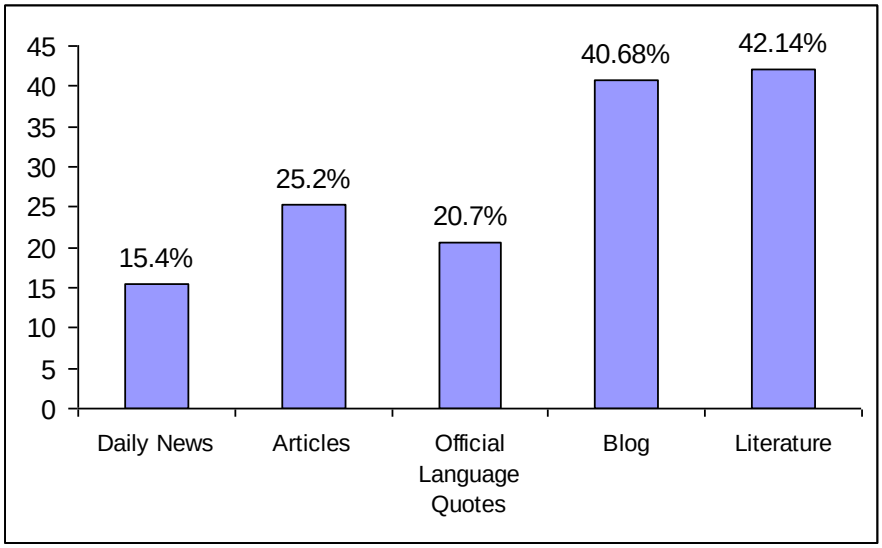

Figure 7.5: Sentence Error Rate for Different Documents

7.3.5.3 Error Analysis Conclusion

As discussed earlier, the WER and SER of un-standardized matter i.e.Blog and Literature is higher than the standardized matter. It strengthens the fact that better input gives the better output. If some pre editing of the text is performed then better results may be expected.

7.4 Comparison with Other Existing Systems:

Following table shows the comparison among various existing systems with our system:

Table 7.4: Comparison of our System with other existing systems

MT SYSTEM	Accuracy	Test Used
RUSLAN	40% correct 40% with minor errors. 20% with major error.	Intelligibility Test

CESILKO (Czech-to-Slovak)	90%	Intelligibility Test
Czech-to-Polish	71.4%	Accuracy Test
Czech-to-Lithuanian	69%	Accuracy Test
Punjabi-to-Hindi	92%	Intelligibility Test
Hindi-to-Punjabi	*94%* *90.84%*	Intelligibility Test Accuracy Test

From the above table, it is clear that the system is outperforming in comparision to others. Thus system can be acceptable for practical use.

7.5 Conclusion

Human evaluation is Holy Grail for MT evaluation, but due to lack of time and money it is becoming impractical. Thus, many automatic MT evaluation techniques have been developed. We have evaluated our system based on the subjective tests and quantitative metrics. From the above analysis, it is concluded the overall accuracy of Hindi to Punjabi Machine Translation system is found to be 94% on the basis of intelligibility test and 90.84% on the basis of accuracy test. The accuracy can be improved by improving and extending the bilingual dictionary. Even robust Word Sense Disambiguation module and Post Processing of the system can improve the system to greater extent. This system is comparable with other existing systems and its accuracy is better than those.

Chapter 8

Summary

We have developed robust Hindi to Punjabi Machine Translation system. It is available to use for free at website http://h2p.learnpunjabi.org. With online version, a user can translate a text by typing it in a box provided at webpage or one can submit a file containing text in Unicode. A user can also translate any Hindi website like http://bbc.co.uk/hindi/ and can view it in Punjabi. An E-mail option is also included whereby a user can type his message in Hindi and send the translated text or typed text to an email id submitted by him. To the best of the knowledge, the current system is one of the best Machine Translation System from one Indic language to another.

In this chapter, we will summarize the achievements and limitations of the present research work. Directions for further research that can help to enhance this Machine Translation system have also been included.

8.1 Contributions

- The survey of various existing Machine Translation systems has been presented. Based on this survey, it has been concluded that direct Machine Translation approach is suitable for closely related language pair. We call a language pair to be closely related if the languages have the grammar that is close in structure, contain similar constructs having almost same semantics, and share a great deal of lexicon. By

closely related languages, we also mean in□ectively and morphosyntactically similar languages. Generally, such languages have originated from the same source and spoken in the areas in close proximity. Thus, being Hindi and Punjabi closely related language pair [250], direct approach has been used for developing Machine Translation system for this language pair.

- The closeness between Hindi and Punjabi has been devised by comparing these languages on the basis of orthogonality, grammar and from machine translation point of view. It has also been proved using corpus based measures by Anil and Harshit [250].
- As the Statistical Machine Translation approach is actively used among researchers nowadays, the scarcity of the resources of language pair like non availability of any annotated or parallel corpus in question limited the choice of translation approach to conventional direct method. The required resources are developed from scratch and used to develop a Machine Translation system.
- The system has to tackle the named entity recognition problem as there are the chances when a token in input text having its translated meaning in target language need to be transliterated rather than translated because it acts as proper noun. Thus, module has been developed for handling Proper Nouns successfully.

- As there is no dictionary available for the language pair for Machine Translation purpose, Hence, Hindi to Punjabi lexicon for Machine Translation has been developed.
- Word sense disambiguation is done by using language modeling techniques. *N-gram*s can successfully model the disambiguation of Hindi language.
- Transliteration is the option for the out-of-vocabulary words. A successful transliteration module has been developed that uses large number of developed rules in addition to direct mapping of characters.
- Transfer rules are desirable for handling the grammatical and some structural deviations.
- The development is aimed to make a robust system for translating the input text without failure or going blank. The system was evaluated formally and informally both ways. In informal evaluation, the system has been made online at website http://h2p.learnpunjabi.org. The system was introduced to all the researchers working in this area through emails. Even the announcement of this Machine Translation system was also done through media (newspapers, Television and FM Radio). All the Major newspapers like The Tribune, Indian Express, Hindustan Times, Ajit daily, Jagbani, Dainik Jagran, Dainik Bhaskar, Amar Ujala, Rozana Rashtriya Sahara (Urdu Newspaper), Punjab Newsline etc. have published the news of launch of this system at prominent positions of their newspaper. Number of readers have used

the system and sent us the feedback about the quality of the system. Now, it is regularly being used by several newspaper publishers for translating their news, book publishers etc. In formal evaluation, the system is evaluated by both objective and subjective tests. The accuracy is figured out as 94% on the basis of Intelligibility test and 90.84% on the basis of accuracy test. In the quantitative tests the Word Error Rate is found out to be 4.58% whereas Sentence Error Rate is 28.82%.

- The development of this system is an effort to bring the Punjabi on the map of Machine Translation. The system can be integrated to other existing translation system like English to Hindi (facility provided by Google) to produce a system that will translate the text from English to Punjabi. In fact the integration of our system with Urdu to Hindi transliteration system is on chart where the Hindi text produced by transliteration system is fed into our translation system thereby producing Punjabi text from the text in Urdu.

8.2. Limitations

Although system shows good accuracy but the system still fails at some points. Some common errors are explained with examples:

8.2.1 Named Entity Recognition Failure:

a. There are foreign names in the text like बाश की लीनी (*bāsh kī līnī)*. It will be translated into ਬਾਸ਼ ਦੀ ਲੀਨੀ (*bāsh dī līnī).*

b. There are proper nouns having multiple translations in Punjabi which do not have title or surname surrounding them in the sentence. For example:

दीपक कहाँ है (*dīpak kahāṃ hai)*

ਦੀਵਾ ਕਿੱਥੇ ਹੈ (*dīvā kitthē hai)*

8.2.2 Modifier and Noun Agreement: All the modifiers must agree with the word that they modify in a noun phrase. But it fails in some of the cases as shown in following example:

बैंक खेतीबाड़ी के कर्जे के लिए घटी दरें लागू करेगा (baiṅk khētībāṛī kē karjē kē liē ghaṭī darēṃ lāgū karēgā)

ਬੈਂਕ ਖੇਤੀਬਾੜੀ ਦੇ ਕਰਜ਼ੇ ਲਈ ਘਟੀ ਦਰਾਂ ਲਾਗੂ ਕਰੇਗਾ। (*baiṅk khētībāṛī dē karzē laī ghaṭī darāṃ lāgū karēgā)*

8.2.3 Subject/Object and Verb Agreement: All the verbs must agree with the Subject/Object in the sentence. But it fails in some of the cases as shown in following example:

हम नयी भाषा कर नौकरी में उन्नती कर सकते हैं (*ham nayī bhāshā kar naukrī mēṃ unntī kar saktē haiṃ)*

ਅਸੀਂ ਨਵੀਂ ਭਾਸ਼ਾ ਸਿੱਖ ਕੇ ਨੌਕਰੀ ਵਿੱਚ ਤਰੱਕੀ ਕਰ ਸਕਦੇ ਹਨ। (*asīṃ navīṃ bhāshā sikkh kē naukrī vicc tarkkī kar sakadē han.)*

8.2.4 Resolving meaning of ambiguous words: For some of the cases, the system fails to resolve the meaning of the word among its multiple meanings. For example:

दो हफ्तों के लिए बढा दिया है (*dō haphtōṃ kē liē baḍhā diyā hai)*

ਦੋ ਹਫਤੀਆਂ ਲਈ ਵਧਿਆ ਦਿੱਤਾ ਹੈ (*dō haphtīāṃ laī vadhiā dittā hai)*

8.2.5 Noun phrase in oblique case form before postposition: In the sentence, if Noun phrase is present before the postposition, then it will come in oblique case. In some cases, it fails as in the following example:

वह खुश होकर अपने पति के पास चली गयी (*Vah khush hōkar apnē pati kē pās calī gayī)*

ਉਹ ਖੁਸ਼ ਹੋ ਕੇ ਆਪਣੇ ਘਰਵਾਲਾ ਦੇ ਕੋਲ ਚਲੀ ਗਈ। (*uh khush hō kē āpaṇē gharvālā dē kōl calī gaī)*

8.2.6 Agreement of subject noun phrase having ੂੰ *with* verb phrase: In Punjabi, all the Verb phrases in the sentence must agree with the Subject Noun phrases having ੂੰ like ਮੈਨੂੰ. Sometimes, it fails in cases as shown in following example:

मुझे दवाई चाहिए (*mujhē davāī cāhiē)*

ਮੈਨੂੰ ਦਵਾਈ ਚਾਹੀਦਾ ਹੈ। (*mainūṃ davāī cāhīdā hai)*

8.2.7 ਦਾ postposition agreement before Verb phrase: In Punjabi, all the Verb phrases in the sentence must agree with the postposition ਦਾ. But in some cases, it fails as shown in following example:

यह शव कंवलजीत सिंह का है (*yah shav kaṃvlajīt siṃh kā hai)*

ਇਹ ਲਾਸ਼ ਕੰਵਲਜੀਤ ਸਿੰਘ ਦਾ ਹੈ। (*ih lāsh kaṃvlajīt siṅgh dā hai.)*

8.2.8 ਵਾਲਾ postposition and following Noun phrase agreement: In Punjabi, the postposition ਵਾਲਾ must agree with the following Noun phrase in the sentence. But in some cases, it fails as shown in following example:

रात वाला झगडा चल रहा है (*rāt vālā jhagḍā cal rahā hai)*

ਰਾਤ ਵਾਲ਼ਾ ਲੜਾਈ ਚੱਲ ਰਿਹਾ ਹੈ। (*rāt vālā laṛāī call rihā hai.)*

8.2.9 Noun Verb Agreement: In Punjabi, the postposition ਵਾਲਾ must agree with the following Noun phrase in the sentence. But in some cases, it fails as shown in following example:

वे कोई भड़काऊ भाषण नहीं दूंगे (*vē kōī bhaḍkāū bhāshaṇ nahīṃ dēṅgē)*

ਉਹ ਕੋਈ ਭੜਕਾਊ ਭਾਸ਼ਣ ਨਹੀਂ ਦੇਵਾਂਗੇ (*uh kōī bhaṛkāū bhāshaṇ nahīṃ dēvāṅgē)*

Similarly, It is not possible to cover all the hindi words in the dictionary and as when we use the application for use, we come across words that are missing and can be added in parallel to its use.

8.3 Future Directions

Although our system is showing good results using the direct translation approach but still there is lot of scope for improvement. Following are some of the future directions:

- **More Data**

The most obvious way to improve a data-driven approach like presented here is of course to utilize more data. Database entries for bilingual dictionary, proper noun gazetteers, surnames, titles, bigrams and trigrams for WSD need to be extended.

- **Resource Development**

Statistical Machine Translation approach has now often been used by the researchers. The only requirement for this approach is the availability of high quality parallel corpus. Thus, with the development of this system, parallel corpus for Hindi-Punjabi Language pair must be developed for use in future researches in these languages.

- **Better Models**

Despite using more data, improved models can lead to better translation quality. Using parallel corpus for the language pair, it is of great interest to combine automatic techniques for various tasks with direct approach to

develop a more robust and accurate Machine Translation system. Even use of full parsers in the Machine Translation Systems can show better results.

- **Public Corpora and Tools**

There are initiatives by various NLP research groups for releasing the corpora publicly. Some of NLP tools are also available for various tasks. Using such corpora and tools will help in reducing the development time and effort of the system. Such practice will also help the researchers' efforts in redoing the tasks that have already been done.

- **Better Evaluation Metrics**

Automatic evaluation metrics are important for a rapid development cycle. During the development and tuning phase, the quality of the MT system is evaluated several (hundred) times. The parameters of the MT system are adjusted to achieve a high score of a given automatic evaluation metric. Nevertheless, the ultimate goal is to improve the translation quality using this parameter tuning. Therefore, automatic evaluation metrics should have a high correlation with human judgment of translation quality. Furthermore, it should not be possible to cheat the metric, i. e. to improve the score without improving translation quality. Current metrics have their limitations as pointed out in [Callison-Burch & Osborne+ 06] for the BLEU score. As MT systems are tuned toward a specific metric, improved MT evaluation metrics will lead to better Machine Translation quality.

- **Integration with other systems**

The system developed can be integrated with other systems to deal with more complicated tasks. The system can be integrated to translate any Language to Hindi and further to Punjabi. Thus, we can say that any language text can be translated to Punjabi language text using this system. For this purpose, Google translation APIs can also be used.

References

[1] Rao D. 2001. Machine Translation in India: A Brief Survey. *In proceedings of SCALLA2001 Conference*, November 21-23, NCST, Bangalore, India. [Internet Source: http://elda.org/en/proj/scalla/SCALLA2001/SCALLA2001Rao.pdf]

[2] Naskar S. and Bandyopadhyay Sivaji. 2005. Use of Machine Translation in India: Current Status. *In proceedings of MT SUMMIT X*, September 13-15, Phuket, Thailand. pp. 465-470.

[3] ALPAC. 1966. Languages and Machines: Computers in Translation and Linguistics. *Report of the Automatic Language Processing Advisory Committee*, Division of Behavioral Sciences, National Academy of Sciences. National Research Council Publication 1416, Washington, D.C.

[4] George and Kumar. 2002. Machine Translation - An Evolving Platform For Social Change. *In proceedings of Symposium on Translation Support Systems, IIT Kanpur.* pp130-37.

[5] Hutchins, W.J. 1978. Machine Translation and machine-aided translation. *Journal of Documentation* 34(2), pp 119-159.

[6] Hutchins, W.J. 1986. Machine Translation: Past, Presence, Future. *Ellis Horwood Series in Computers and their Applications*. Chichester. New York. pp 382.

[7] Hutchins, W.J. 1997. Looking back to 1952: the first MT conference TMI-97. *In proceedings of the 7th International Conference on*

Theoretical and Methodological Issues in Machine Translation, July 23-25, St.John's College, Santa Fe, New Mexico, USA. pp.19-30.

[8] Hutchins, W. J. 1994. The Georgetown-IBM Demonstration, *MT News International*, pp. 8-10.

[9] Hutchins and Somer. 1992. An Introduction to Machine Translation, London: Academic Press.

[10] Ke Ping, 1996. A Socio-semiotic Approach to Meaning in Translation. *BABLE (42) 2*. pp 74-83.

[11] Kristin Demos and Mark Frauenfelder. 2000. Machine Translation's past and future. *Wired* 8 (5).

[12] Panov, D.Y. 1960. Machine Translation and Human being. *Impact of Science on Society Vol 10.* pp. 16-25.

[13] Reifler, E. 1961. MT linguistics and MT lexicography. *Kent*, University of Washington. pp. 841-852.

[14] Richards, I. A. 1953. Towards a theory of translation. In *Studies in Chinese Thought,* University of Chicago Press, Chicago.

[15] Toma, P. 1974. Computer translation: in its own right. In: *Kommunikationsforschung und Phonetik* (Hamburg: Buske), pp. 155-164.

[16] Toma, P. 1976. An operational Machine Translation system. *Brislin, R.W. ed. Translation: applications and research* (New York: Gardner P.), pp. 247-259.

[17] Yehoshua Bar-Hillel. 1964. The present state of research on mechanical translation, *American Documentation* 2 (4), pp. 229-237.

[18] Brown, P., Cocke, J., Della Pietra, S., Jelinek, F., Lafferty, J., Mercer, R., and Roossin, V., Della Pietra, P. 1990. *A Statistical Approach to Machine Translation Computational Linguistics 16(2)*, pp. 79-85

[19] Allen, J. 1995. *Natural Language Understanding. Second Edition.* Benjamin Cummings.

[20] Ashkar Bharati, Vineet Chaitanya, Rajeev Sangal. 2004. *Natural Language Processing: A Paninian Perspective*, PHI.

[21] Daniel Jurafsky, James H. Martin. 2000. *Speech and Language Processing*. Second Edition, PHI.

[22] John. H. 2009. Multiple uses of Machine Translation and computerized translation tools*. ISMTCL: International Symposium on Data Mining and Sense Mining, Machine Translation and Controlled Languages, and their application to emergencies and safety critical domains*, July 1-3, Centre Tesnière, University of Franche-Comté, Besançon, France .pp. 13-20.

[23] Dostert, L.E. 1955. The Georgetown-IBM experiment. *In Locke & Booth :1955*. pp.124-135.

[24] Vauquois, B. and Boitet, C. 1985. Automated translation at Grenoble University. *Computational Linguistics 11.* pp. 28-36.

[25] Veillon, G. 1968. Description du langage pivot du système de traduction automatique du C.E.T.A. *T.A.Informations1968*, pp. 8-17.

[26] Vauquois, B. 1976. Automatic translation - a survey of different approaches. *Statistical Methods in Linguistics (Stockholm).* pp.127-135.

[27] Vauquois, B. and Boitet, C. 1984. Automated translation at GETA. *Grenoble: GETA.* pp. 130-36.

[28] Thurmair, G. 1990. Complex Lexical Transfer in Metal. *In proceedings of the third International. Conference on Theoretical and Methodological Issues in Machine Translation of Natural Languages (Austin, TX).* pp. 91-107.

[29] David B. Orr and Victor H. Small. 1967. Comprehensibility of machine-aided translations of Russian scientific documents. *Mechanical Translation and Computational Linguistics, vol. 10, nos.1 and 2.* pp.1-10

[30] Lale Yurtseven. 1997. Logos Machine Translation system. *In proceedings of MT Summit VI. Machine Translation: Past, Present, Future, 29 October – 1 November 1997, San Diego, California, USA.* pp. 251-252.

[31] Lale Yurtseven. 1997. Logos Machine Translation system. *Fifth conference on Applied Natural Language Processing [of] Association for Computational Linguistics. Description of system*

demonstrations and videos, 31 March – 3 April 1997, Washington Marriott Hotel, Washington, DC, USA. p.17.

[32] Lale Yurtseven. 1997. Logos Translation System. *Exhibit at MT Summit VI. Machine Translation: Past, Present, Future*, 29 October – 1 November 1997, San Diego, California, USA. p. 279.

[33] Pierre Isabelle and Laurent Bourbeau. 1985 TAUM-AVIATION: its technical features and some experimental results. *Computational Linguistics* 11 (1). pp. 18-27.

[34] John Chandioux. 1977. Creation of a second-generation system for Machine Translation of technical manuals. *Overcoming the language barrier: third European Congress on Information Systems and Networks*, Luxembourg, 3-6 May 1977, organised by the Commission of the European Communities (München: Verlag Dokumentation, 1977). Vol. 1. pp. 613-621.

[35] Joann P. Ryan. 1987. Systran: a Machine Translation system to meet user needs. *MT Summit: Machine Translation Summit.* Manuscripts & Program, September 17-19, 1987, Hakone Prince Hotel, Japan. pp. 99-103.

[36] Dorothy Senez. 1995. The use of Machine Translation in the Commission *MT Summit V* Proceedings, Luxembourg, July 10-13, 1995. pp. 11.

[37] Jean Senellart, Jin Yang, & Anabel Rebollo. 2003. SYSTRAN intuitive coding technology. *In proceedings of MT Summit IX*, New Orleans, USA, 23-27 September 2003. pp.346-353.

[38] Loic Dugast, Jean Senellart, and Philipp Koehn. 2007. Statistical Post-Editing on SYSTRAN's Rule-Based Translation System. *In Proceedings of WMT-2007*. pp. 220-223.

[39] Pierre Senellart and Jean Senellart. 2005. SYSTRAN Translation Stylesheets: Machine Translation driven by XSLT. *In proceedings XML Conference & Exposition*. Atlanta. USA. pp.167-178

[40] Shiu-Chang Loh & Luan Kong. 1979. An interactive on-line Machine Translation system (Chinese into English). *Translating and the Computer: proceedings of a seminar*, London, 14th November, 1978. pp. 135-148.

[41] Shiu-Chang Loh, Luan Kong, & Hing-Sum Hung. 1978. Machine Translation of Chinese mathematical articles. *ALLC Bulletin*, Vol.6, 1978. pp. 111-120.

[42] P.H.Nancarrow. 1978. The Chinese University Language Translator (CULT) – a report. *ALLC Bulletin*, vol.6, 1978; p. 121.

[43] S.C.Loh and L.Kong. 1977. Computer translation of Chinese scientific journals. *Overcoming the language barrier: third European Congress on Information Systems and Networks*, Luxembourg, 3-6 May 1977, organised by the Commission of the European Communities. Vol.1. pp.631-645.

[44] Shiu-Chang Loh. 1976. CULT, Chinese University Language Translator [abstract]. In proceedings of FBIS Seminar on Machine Translation, 8-9 March 1976, Rosslyn, Virginia. *American Journal of Computational Linguistics*, microfiche 46; pp.46-50.

[45] Patrick Corness. 1985. The ALPS computer-assisted translation system in an academic environment. *Translating and the Computer 7*. pp.118-127.

[46] Benoît Thouin. 1981. The METEO system. *In Proceedings of a conference on Practical experience of Machine Translation.* London, 5-6 November 1981. pp. 39-44.

[47] John Chandioux. 1976. METEO: An operational system for the translation of public weather forecasts. *In proceedings of FBIS Seminar on Machine Translation*, 8-9 March 1976, Rosslyn, Virginia. pp.27-36.

[48] Makoto Nagao, Jun-ichi Tsujii , Koji Yada , Toshihiro Kakimoto, 1982. An English Japanese Machine Translation system of the titles of scientific and engineering papers. *In proceedings of the 9th conference on Computational linguistics*. July 05-10. Prague. Czechoslovakia. pp.245-252.

[49] Hajic J., 1987. Ruslan-An MT System between closely related languages, *In Proceedings of the 3rd Conference of The European Chapter of the Association for Computational Linguistics*, Copenhagen. Denmark. pp.113-117.

[50] Dyvik, Helge. 1995. Exploiting Structural Similarities in Machine Translation. *Computers and the Humanities*. pp.225 - 234.

[51] Marote R. C, Guillen E., Alenda A.G., Savall M.I.G., Bellver A.I., Buendia S.M., Rozas S.O., Pina H.P., Anton P.M.P., Forcada M.L. 2001. The Spanish-Catalan Machine Translation system interNOSTRM. *In proceedings of MT Summit VIII*, September 18-22 Sept, Santiago de Compostela, Galicia, Spain. pp. 156-162.

[52] Roxas, R., Devilleres, E., Giganto, R. 2000. Language Formalisms for Multi-lingual Machine Translation of Philippine Dialects. *MS Thesis.* De La Salle University, Manila, 2000.

[53] Borra, A. 2000. A Transfer-based Analysis Engine English to Filipino Machine Translation. *Ph.D. Thesis.* Institute of Computer Science, UPLB.

[54] Fat, J.G. 2004. T2CMT: Tagalog-to-Cebuano Machine Translation, *MS Thesis*, De La Salle University, Manila,. 2004.

[55] Cigdem Keyder Turhan. 1997.An English to Turkish Machine Translation system using structural mapping. *In proceedings of Fifth conference on Applied Natural Language Processing [of] Association for Computational Linguistics*. pp.320-323.

[56] HAJIC J, HRIC J, KUBON V. 2000. CESILKO– an MT system for closely related languages. *In Tutorial Abstracts and Demonstration Notes of ACL2000. Washington.* pp. 7-8.

[57] S. Marinov. 2000. Structural Similarities in MT: A Bulgarian-Polish Case. [Internet Source: http://www.gslt.hum.gu.se/~svet/courses/mt/termp.pdf.]

[58] K. Altintas. 2001 Turkish To Crimean Tatar Machine Translation System. *Master Thesis*. Department Of Computer Engineering And The Institute Of Engineering And Science. Bilkent University, Turkey.

[59] M. Corbí-Bellot, Mikel L. Forcada, Sergio Ortiz-Rojas, Juan Antonio Pérez-Ortiz, Gema Ramírez-Sánchez, Felipe Sánchez-Martínez, Iñaki Alegria, Aingeru Mayor, Kepa Sarasola. 2005. An open-source shallow-transfer Machine Translation engine for the Romance languages of Spain. *In Proceedings of the Tenth Conference of the European Association for Machine Translation*. May 30-31. Budapest.Hungary. pp 79-86.

[60] Carme A., Rafael C., Antonio M., Mikel L., Mireia G., Sergio O., Juan A., Gema R., Felipe S., Miriam A. Open-source Portuguese-Spanish Machine Translation. 2006. *In Lecture Notes in Computer Science 3960 (Computational Processing of the Portuguese Language, Proceedings of the 7th International Workshop on Computational Processing of Written and Spoken Portuguese, PROPOR 2006)*. May 13-17. ME - RJ/Itatiaia, Rio de Janeiro, Brazil, pp. 50-59.

[61] Mikel L.Forcada, Francis M.Tyers, and Gema Ramírez-Sánchez. 2009. The Apertium Machine Translation platform: five years on. *In Proceedings of the First International Workshop on Free/Open-Source Rule-Based Machine Translation*, 2-3 November 2009, Universitat d'Alacant, Alacant, Spain. pp. 3-10.

[62] Scannell K.P. Machine Translation for Closely Related language Pair. 2006. *In proceedings of the Workshop on Strategies for developing Machine Translation for minority languages*. LREC 2006, Genoa, Italy. pp.103-107.

[63] J.González, A.L.Lagarda, J.R.Navarro, L.Eliodoro, A.Giménez, F.Casacuberta, J.M.de Val, & F.Fabregat. 2006. SisHiTra: a Spanish-to-Catalan hybrid Machine Translation system. *LREC-2006: Fifth International Conference on Language Resources and Evaluation*. 5th SALTMIL Workshop on Minority Languages: "Strategies for developing Machine Translation for minority languages", Genoa, Italy, 23 May 2006. pp. 69-73.

[64] R. M. K. Sinha, Jain R., Jain A. 2001. Translation from English to Indian languages: ANGLABHARTI Approach. *In proceedings of Symposium on Translation Support System STRANS 2001*. February 15-17, IIT Kanpur, India. pp.167-172.

[65] Bharati, Akshar, Chaitanya, Vineet, Kulkarni, Amba P., Sangal, Rajeev. 1997. Anusaaraka: Machine Translation in stages. Vivek, A

Quarterly in Artificial Intelligence, Vol. 10, No. 3. ,NCST, Banglore. India, pp. 22-25.

[66] Renu Jain, R.M.K.Sinha, and Ajai Jain, ANUBHARTI.2001. Using Hybrid Example-Based Approach for Machine Translation. *In proceedings of Symposium on Translation Support Systems (SYSTRAN2001)*, February 15-17,2001. Kanpur. pp.123-130.

[67] Hemant Darbari. 1999. Computer-assisted translation system – an Indian perspective. *Machine Translation Summit VII,* 13th-17th September 1999, Kent Ridge Digital Labs, Singapore. *In Proceedings of MT Summit VII : MT in the Great Translation Era*. pp.80-85.

[68] Murthy. K. 2002. MAT: A Machine Assisted Translation System. *In Proceedings of Symposium on Translation Support Systems, STRANS-2002, IIT Kanpur, India.* pp.134-139.

[69] Lata Gore and Nishigandha Patil. English To Hindi - Translation System. In *proceedings of Symposium on Translation Support Systems STRANS-2002.* IIT Kanpur. pp. 178-184.

[70] Kommaluri Vijayanand, Sirajul Islam Choudhury, Pranab Ratna. 2002. VAASAANUBAADA - Automatic Machine Translation of Bilingual Bengali-Assamese News Texts. *Language Engineering Conference*. Hyderabad, India. [Internet Source: *http://*portal.acm.org/citation.cfm?id=788716]

[71] R.M.K. Sinha. 2004. An Engineering Perspective of Machine Translation: AnglaBharti-II and AnuBharti-II Architectures. *In proceedings of International Symposium on Machine Translation, NLP and Translation Support System (iSTRANS- 2004).* November 17-19. Tata Mc Graw Hill, New Delhi. Pp. 134-38.

[72] Ananthakrishnan R, Kavitha M, Jayprasad J Hegde, Chandra Shekhar, Ritesh Shah, Sawani Bade, Sasikumar M. 2006. MaTra: A Practical Approach to Fully- Automatic Indicative English-Hindi Machine Translation. In the proceedings of MSPIL-06.

[73] Bharati, R. Moona, P. Reddy, B. Sankar, D.M. Sharma, R. Sangal. 2003. Machine Translation: The Shakti Approach. *Pre-Conference Tutorial*. ICON-2003.

[74] Sivaji Bandyopadhyay. 2004. Use of Machine Translation in India. *AAMT Journal*, 36. pp. 25-31.

[75] Bandyopadhyay S. 2000. ANUBAAD - The Translator from English to Indian Languages. *In proceedings of the VIIth State Science and Technology Congress. Calcutta. India. pp. 43-51.*

[76] R. Mahesh K. Sinha and Anil Thakur. 2005. Machine Translation of bi-lingual Hindi-English (Hinglish) text. in proceedings of the tenth Machine Translation Summit. MT Summit X, Phuket, Thailand, September 13-15. pp.149-156.

[77] Sobha L, Pralayankar P, and Kavitha V. 2009. Case Marking Pattern from Hindi to Tamil MT. *In proceedings of 3rd National*

Conference on Recent Advances and Future Trends in IT (RAFIT), *Punjabi University, Patiala, Punjab.* pp.156-59.

[78] R.M.K. Sinha, Jain A. 2003. AnglaHindi: an English to Hindi machine-aided translation system, MT Summit IX, New Orleans. USA. September 23-27. pp.494-497.

[79] D. Gupta and N. Chatterjee. 2003. Identification of Divergence for English to Hindi EBMT. *In proceedings of MT SUMMIT IX.* New Orleans, Louisiana, USA. pp.157-162.

[80] Sivaji Bandyopadhyay. 2002. Teaching MT – An Indian Perspective. *In proceedings of the 6th EAMT Workshop on Teaching Machine Translation*. Manchester. UK. pp.13-22.

[81] Shachi Dave, Jignashu Parikh and Pushpak Bhattacharyya. 2001. Interlingua-based English- Hindi Machine Translation and Language Divergence. *Journal of Machine Translation, 16 (4)*. pp. 251-304.

[82] Sivaji Bandyopadhyay. 2000. State and Role of Machine Translation in India. *Machine Translation Review*, 11. pp. 25-27.

[83] Murthy, B. K., Deshpande, W. R. 1998. Language technology in India: past, present and future. *In proceedings of MLIT Symposium 3.* GII/GIS for Equal Language Opportunity. Vietnam. October 6-7. pp. 134-137.

[84] G. S. Josan and G. S. Lehal. 2008. A Punjabi to Hindi Machine Translation System. *Coling 2008: Companion volume: Posters and Demonstrations*, Manchester, UK, pp. 157-160.

[85] Sanjay Chatterji,Devshri Roy,Sudeshna Sarkar, Anupam Basu. 2009. A Hybrid Approach for Bengali to Hindi Machine Translation. *In proceedings of ICON 2009, 7th International Conference on Natural Language Processing.* pp. 83-91.

[86] V. Geethakumary. 1997. A Contrastive Analysis of Hindi and Malayalam", *Ph.D. thesis in Linguistics*, University of Kerala.

[87] Shapiro Michael C. 1986. *A primer of modern standard hindi*. New Delhi: Moti Lal Banarsi Dass Publishers.

[88] Gill Harjeet , Gleason Henry A. 1963. *A Reference Grammar of Punjabi*. Patiala: Punjabi University Publication Bureau.

[89] Bhatia, Tej K. 1996. *Colloquial Hindi.* New York : Routledge. Günther Hartmut, Ludwig Otto. 1996. *Writing and its use, an Interdisciplinary Handbook of international Research.* Walter de Gruyter.

[90] Hajic J, Hric J, Kubon V. 2000. Machine Translation of Very Close Languages. *In proceedings of the 6th Applied Natural Language Processing Conference*. April 29 - May 4, 2000, Seattle, Washington, USA. pp 7-12.

[91] Joshi S.S. 1978. *Punjabi-English Dictionary*, Patiala: Punjabi University Publication Bureau.

[92] Marrafa, Palmira and Ribeiro A. 2001. Quantitative Evaluation of Machine Translation Systems: Sentence level. In Proceedings of the MT Summit VIII Fourth ISLE workshop 2001, Spain, pp. 39-43.

[93] Masica, Coline P. 1991. *Indo-Aryan languages*. Cambridge: Cambridge University Press.

[94] Newton, E.P., 1898. *Panjabi Grammar.* Ludhiana mission press

[95] Singh Jodh, Badan Baldev Singh, Singh Maninder, Joshi Ramsharan, Singh Rajinder. 1990. *Hindi to Punjabi Dictionary*, New Delhi: National Book Shop.

[96] Singh H. 1991. *Saadian Bhashawan,* Ed. 1, New Delhi: Punjabi Academy.

[97] Singh H. Singh L. 1986. *College Punjabi Viakaran* ,Chandigarh: Punjab State University text Book Board.

[98] Chaturvedi, M.G. A Constrastive Study of Hindi-English Phonology. National publishing House. Delhi.

[99] Frank Van Eynde, “Machine Translation and Linguistic Motivation”, in Frank Van Eynde, ed., *Linguistic Issues in Machine Translation*, London: Pinter Publishers, 1993, p. 73.

[100] Gill H. S. 1996. The Gurmukhi Script. *In Daniels and Bright, The World's Writing Systems.* p. 397.

[101] Krishnaswamy, N. and Verma. S.K.1992. Modern Applied Linguistics: An introduction. Macmillion India Press, Madras, 1992.

[102]Slype V. 1979. Critical Methods for Evaluating the Quality of Machine Translation. *Prepared for the European Commission Directorate General Scientific and Technical Information and Information Management. Report BR-19142*. pp. 321-350.

[103]Alan Black, Mari Ostendorfand Christopher Richards, Richard Sproat, Stanley Chen, Shankar Kumar. 2001. Normalization of Non-Standard Words. *Computer Speech and Language, 15(3)*: pp. 287-333

[104] K. Church and P. Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics. 16(1).*

[105] P. Pecina. 2005. An extensive empirical study of collocation extraction methods. *In proceedings of ACL Student Research Workshop*- 2005. pp 123-128.

[106]T. Dunning. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics.19(1).*

[107]F. Smadja.1993. Retrieving collocations from text: Xtract. *Computational Linguistics. 19(1).*

[108]Lin. Automatic identification of noncompositional phrases. *In proceedings of ACL 1999*.

[109] T. de Cruys and B. V. Moiron. 2007. Semantics-based multiword expression extraction. *In proceedings of ACL-2007 Workshop on Multiword Expressions.* pp.134-139.

[110] Fazly and S. Stevenson. 2006 Automatically constructing a lexicon of verb phrase idiomatic combinations. *In proceedings of EACL-2006.* pp. 156-178.

[111] G. Katz and E. Giesbrechts. 2006. Automatic identification of noncompositional multi-word expressions using Latent Semantic Analysis. *In proceedings of ACL- 2006 Workshop on Multiword Expressions.* pp. 145-167.

[112] T. Baldwin, C. Bannard, T. Tanaka, and D.Widdow. 2003. An empirical model of multiword expressions decomposability. ACL-2003 Workshop on Multiword Expressions. pp.178-190.

[113] B.V. Moiron and J. Tiedemann. 2006. Identifying idiomatic expressions using automatic word alignment. *In proceedings of EACL 2006 Workshop on Multiword Expressions in a multilingual context*. pp 140-145.

[114] T. Tomokiyo and M. Hurst. 2003. A language model approach to keyphrase extraction. *In proceedings of ACL-2003 Workshop on Multiword Expressions*. pp 178-190.

[115] S. Venkatapathy and A. Joshi. 2005. Relative Compositionality of Noun+Verb Multi-word Expressions in Hindi. *In proceedings of ICON-2005.* IIIT Hyderabad. Pp 123-128.

[116] Mukerjee, A. Soni, and A. Raina. 2006. Detecting Complex Predicates in Hindi using POS Projection across Parallel corpora. *In*

proceedings of the Workshop on Multiword Expressions at ACL-2006. pp. 156-163.

[117] Plag. 2003. Word Formation in English. Cambridge University Press.

[118] Duda, R. and Hart, P. 1973. *Pattern Classification and Scene Analysis*, John Wiley and Sons, New York.

[119] Rivest, Ronald L. 1987. Learning Decision Lists, *Machine Learning*, 2(3). pp.229-246.

[120] Quinlan, J.R. 1986. Induction of Decision Trees, *Machine Learning,* 1. pp. 81-106.

[121] Rumelhart, D.E., Hinton, G.E., and Williams, R.J., 1986. Learning Internal Representations by Error Propagation. *Parallel Distributed Processing*, Vol. 1. pp. 318-362.

[122] Mitchell, Tom, 1997. *Machine Learning : Chapter 6*, McGraw Hill.

[123] Pedersen, T. 2000. A simple approach to building ensembles of naïve bayesian classifiers for word sense disambiguation. *In Proceedings of the First Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2000).* Seattle .pp. 63-69.

[124] Gale, B., Church, K., and Yarowsky, D. 1992. A method for disambiguating word senses in a large corpus. *Computers and the Humanities:* 26. pp.415-439.

[125] Gale, B., Church, K., and Yarowsky. 1992. One sense per discourse. *In proceedings of the ARPA Workshop on Speech and Natural Language Processing.* pp. 233-237.

[126] Mooney, R. 1996. Comparative experiments on disambiguating word senses: An illustration of the role of bias in machine learning. *In proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-96)*, University of Pennsylvania, pp. 82-91.

[127] Florian, R., Cucerzan, S., Schafer, C. and Yarowsky, D, 2002. Combining classifiers for word sense disambiguation. *Natural Language Engineering*, Special Issue on Word Sense Disambiguation Systems, 8(4). pp.327-341.

[128] Yarowsky, D.. 2000. Hierarchical decision lists for word sense disambiguation. *Computers and the Humanities*. 34(1-2). pp. 179-186.

[129] Agirre, E., and Martinez, D., 2000. Exploring automatic word sense disambiguation with decision lists and the web. *In Proceedings of the Coling 2000 Workshop: Semantic Annotation and Intelligent Annotation.,* Centre Universitaire.Luxembourg. pp. 201-208.

[130] Towell, G. and, Voorhees, E. 1998. Disambiguating Highly Ambiguous Text. *Computational Linguistics.* 24(1). pp.125-145.

[131] Black, E. 1988. An Experiment in Computational Discrimination of English Word Senses. *IBM Journal of Research and Development.* 32. pp.185-194.

[132] Pedersen, T. 2002. Evaluating the effectiveness of ensembles of decision trees in disambiguating Senseval lexical samples. *In Proceedings of the ACL 2002 Workshop on Word Sense Disambiguation: Recent Successes and Future Directions*. Philadelphia. pp. 145-51.

[133] Ng, H. T. and Lee, H. B. 1996. Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach. *In 34th Annual Meeting of the Association for Computational Linguistics (ACL 1996)*. Santa Cruz. pp. 40-47.

[134] Brown, Peter F., Stephen, D.P., Vincent, J.D.P., and Robert, L.M. 1991. Word sense disambiguation using statistical methods. *In Proceedings of the 29th Annual Meeting*. Berkeley. pp.264-270.

[135] Landes, S., Leacock, C., and Tengi, R. 1998. Building a semantic concordance of English, *In C. Fellbaum, ed., WordNet: An electronic lexical database and some applications.* Cambridge. MA.: MIT Press. pp. 198-203.

[136] Kilgarriff, A. 1998. Senseval: An exercise in evaluating word sense disambiguation programs. *In proceedings of the First International Conference on Language Resources and Evaluation (LREC 1998)*, pp. 581-588.

[137] Kilgarriff, A. and Rosenzweig, J. 2000. Framework and results for English Senseval. *Computers and the Humanities, 34(1-2).* pp.15-48.

[138] Kilgarriff, A. 2001. English lexical sample task description. *In proceedings of Senseval-2*. Second International Workshop on Evaluating Word Sense Disambiguation Systems, Toulouse. pp.123-128.

[139] Hearst, M. 1991. Noun homograph disambiguation using local context in large text corpora. *In proceedings of the 7th Annual Conference of the UW Centre for the New OED and Text Research: Using Corpora*. Oxford. pp. 178-184.

[140] Schütze, H. 1992. Dimensions of Meaning. *In Proceedings of Supercomputing*. pp. 787-796.

[141] Schütze, H.. 1993. Word Space. *Advances in Neural Information Processing Systems 5*. pp. 895-902.

[142] Yarowsky, D. 1995. Unsupervised word sense disambiguation rivaling supervised methods. *In 33th Annual Meeting of the Association for Computational Linguistics (ACL 1995).* Cambridge. pp. 189-196.

[143] Mihalcea, R. and Moldovan, D. 2001. A highly accurate bootstrapping algorithm for word sense disambiguation. *International Journal on Artificial Intelligence Tools*. 10(1-2). pp. 5-21.

[144] Mihalcea, R. 2002. Bootstrapping Large Sense Tagged Corpora. *In proceedings of the 3rd International Conference on Languages Resources and Evaluations (LREC 2002). Las Palmas. Spain.* pp. 205-211.

[145] Brown, P., Lai, J. C., and Mercer, R. 1991. Aligning Sentences in Parallel Corpora. *In Proceedings of ACL-91,* Berkeley. CA. pp. 167-174.

[146] Dagan, I. and Itai, A. 1994. Word sense disambiguation using a second language monolingual corpus. *Computational Linguistics.* 20(4). pp. 563-596.

[147] Ide, N., Erjavec, T., and Tufis, D. 2002. Sense discrimination with parallel corpora. *In proceedings of the ACL 2002 Workshop on Word Sense Disambiguation: Recent Successes and Future Directions*. pp. 56-60.

[148] Ng, H. T., Wang, B., and Chan, Y. S. 2003. Exploiting parallel texts for word sense disambiguation. In 41th Annual Meeting of the Association for Computational Linguistics (ACL 2003), pp. 455-462.

[149] Witten, I. H., and Bell, T.C. 1991. The zero-frequency problem: Estimating the probabilities of novel events in adaptive text compression. *IEEE Transactions on Information Theory.* 37(4). pp. 1085-1094.

[150] Good, I. 1953. The populations of frequencies of species and the estimation of population parameters. *Biometrika.* pp. 237-264.

[151] Gale, W., 1994. Good Turing smoothing without tears. *Technical report*. AT&T Bell Laboratories. Murray Hill, NJ.

[152] Yarowsky, D. 1992. Word-sense disambiguation using statistical models of Roget's categories trained on large corpora. *In proceedings of 14th International Conference on Computational Linguistics, COLING-92*. Nantes. pp. 454-460.

[153] Chapman, R. 1977. *Roget's International Thesaurus (Fourth Edition).* Harper and Row. New York.

[154] McCarthy, D., Koeling, R., Weeds, J. and Carroll, J. 2004. Finding predominant senses in untagged text. *In proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics.* Barcelona. Spain. pp. 156-161.

[155] Schütze, H. and Pedersen, J. 1995. Information retrieval based on word senses. *In proceedings of Fourth Annual Symposium on Document Analysis and Information Retrieval*. pp.161-175.

[156] Pedersen, T. and Bruce, R. 1997. Distinguishing word senses in untagged text. *In proceedings of the Second Conference on Empirical Methods in Natural Language Processing*. pp.197–207.

[157] Schütze, H. 1998. Automatic word sense discrimination. *Computational Linguistics.24(1)*. pp.97–123.

[158] Purandare, A. and Pedersen, T. 2004. Word Sense Discrimination by Clustering Contexts in Vector and Similarity Spaces. *In*

Proceedings of the Conference on Computational Natural Language Learning (CoNLL). pp. 164-170.

[159] Pedersen, T., and Bruce, R. 1998. Knowledge lean word sense disambiguation. *In Proceedings of the Fifteenth National Conference on Artificial Intelligence.* pp.800–805.

[160] Lesk, M. 1986. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. *In proceedings of ACM SIGDOC Conference,* pp. 24-26.

[161] Cowie, J., Guthrie, J., and Guthrie, L. 1992. Lexical disambiguation using simulated annealing. *In proceedings of the 15th [sic] International Conference on Computational Linguistics (Coling 1992)*, pp. 359-365.

[162] Sampson, G. 1986. A stochastic approach to parsing. *In proceedings of the 11th International Conference on Computational Linguistics (COLING-86)*. pp.151-155.

[163] Stevenson, M. and Wilks, Y., 2001. The interaction of knowledge sources in word sense disambiguation. *Computational Linguistics*: 27(3). pp.321-349.

[164] Pedersen, T. and Banerjee, S. 2002. An adapted lesk algorithm for word sense disambiguation using WordNet. *In proceedings of the Third International Conference on Intelligent Text Processing and Computational Linguistics (CICLing-02).* pp. 146-151.

[165] Miller, G., Leacock, C., Tengi, R., Bunker, R., and Miller, K. 1990. Five papers on WordNet. *Special Issue of International Journal of Lexicography :3(4)*.

[166] Véronis, J., and Ide, N.1991. An assessment of information automatically extracted from machine readable dictionaries. *In proceedings of the fifth conference of the European chapter of the Association for Computational Linguistics.* pp. 227-232.

[167] Ide, N., and Véronis, J. 1993. Refining taxonomies extracted from machine readable dictionaries. *Research in Humanities Computing II .* Oxford University Press. pp. 45-59.

[168] Ide, N., and Véronis, J. 1993. Knowledge extraction from machine readable dictionaries: an evaluation. *Third International EAMT Workshop "Machine Translation and the Lexicon", Heidelberg (Germany), 1993b.* pp.201-206.

[169] Ide, N. and Véronis, J. 1998. Introduction to the special issue on word sense disambiguation: The state of the art. *Computational Linguistics*:24(1). pp.1-40.

[170] Walker, D. E. 1957. Knowledge resource tools for accessing large text files. *Machine Translation: Theoretical and methodological issues*. Cambridge University Press. pp. 254.

[171] Lenat, Douglas B. and Guha, Ramanathan V. 1990. Building large knowledge-based systems. *Reading*,Addison-Wesley Massachusetts.

[172] Briscoe, Edward J. 1991. Lexical issues in natural language processing. *In proceedings of the Symposium on Natural Language and Speech.* November 26-27. Brussels, Belgium. pp. 39-68.

[173] Grishman, R., MacLeod, C., and Meyers, A., COMLEX. 1994. Syntax: Building a computational lexicon. *In proceedings of the 15th International Conference on Computational Linguistics.* COLING'94*,* August 5-9, Kyoto, Japan, pp. 68-272.

[174] Voorhes, Ellen M. 1993. Using WordNet to disambiguate word senses for text retrieval. *In proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* 27 June-1 July. Pittsburgh, Pennsylvania. pp.171-180.

[175] Richardson, R., and Smeaton, Alan F. 1994. Automatic word sense disambiguation in a KBIR application*. Working paper CA-0595*. School of Computer Applications, Dublin City University, Dublin, Ireland.

[176] Li, X., Szpakowics, S., and Matwin, S. 1995. A WordNet-based algorithm for word sense disambiguation. *In proceedings of the 14th International Joint conference on Artificial Intelligence*, Montreal. pp. 1368-1374.

[177] Leacock, C., Chodorow, M., and Miller, G. 1998. Using corpus statistics and WordNet relations for sense identification. *Computational Linguistics, 24(1)*. pp.147-165.

[178] Hawkins, P., DURHAM. 1999. A Word Sense Disambiguation System. *PhD thesis*. Laboratory for Natural Language Engineering, Department of Computer Science, University of Durham, Durham.

[179] Fellbaum, C., Palmer, M., Trang Dang, H., Delfs, L., Wolff, S. 2001. Manual and Automatic Semantic Annotation with WordNet. *In proceedings of the NAACL Workshop on WordNet and Other Lexical Resources: Applications, Customizations*. Carnegie Mellon University, Pittsburg, PA. pp. 310-314.

[180] Resnik, P. 1995. Disambiguating Noun Groupings with Respect to WordNet Senses. *In proceedings of the Third Workshop on Very Large Corpora*. Cambridge. Massachusetts. pp. 54-68.

[181] Agirre, E. and Rigau, G. 1996. Word sense disambiguation using conceptual density. *In proceedings of the 16th International Conference on Computational Linguistics (Coling 1996).* pp. 16-22.

[182] Mihalcea, R. and Moldovan, D. 1999. A method for word sense disambiguation of unrestricted text. *In proceedings of 37th Annual Meeting of the Association for Computational Linguistics (ACL 1999).* Maryland. pp.152-158.

[183] Lin, D. 1997. Using syntactic dependency as local context to resolve word sense ambiguity. *In proceedings of 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics (ACL/EACL 1997)* . Madrid. pp.64-71.

[184] Lin, D. 1998. Automatic retrieval and clustering of similar words. *In proceedings of COLINGACL'98.* Montreal. pp.132-136.

[185] Lin, D. 2000. Word sense disambiguation with a similarity-smoothed case library. *Computers and the Humanities, 34(1-2)*. pp.147-152.

[186] Jiang, Jay J., and Conrath, David W. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. *In proceedings of ROCLING X.* Taiwan. pp. 145-49.

[187] Agirre, E. and Martinez, D. 2001. Knowledge sources for word sense disambiguation. *In proceedings of the Fourth International Conference TSD 2001*, *Notes in Computer Science.* Berlin. pp.1-10.

[188] Haynes, S., Semantic tagging using WordNet examples. 2001. *In proceedings of Senseval-2, Second International Workshop on Evaluating Word Sense Disambiguation Systems*. Toulouse. pp. 79-82.,

[189] Banerjee, Satanjeev, and Pedersen, 2003. Extended Gloss Overlaps as a Measure of Semantic Relatedness. *In proceedings of the Eighteenth International Joint Conference on Artificial Intelligence IJCAI-2003*, Acapulco, Mexico. Pp.143-47.

[190] Litkowski, K. 2000. Senseval: The CL research experience. *Computers and the Humanities, 34(1-2)*. pp.153-158.

[191] Litkowski, K. 2001. Use of machine readable dictionaries for word-sense disambiguation in Senseval-2. *In proceedings of Senseval-2,*

Second International Workshop on Evaluating Word Sense Disambiguation Systems. Toulouse. pp.107-110.

[192] Mihalcea, R. and Moldovan, D. 1998. Word sense disambiguation based on semantic density. *In proceedings of the Coling-ACL'98 Workshop : Usage of WordNet in Natural Language Processing Systems.* Montreal. pp. 16-22.

[193] S Baskaran, V Vaidehi. 2002. *Role of Collocations and Case-Markers in Word Sense Disambiguation: A Clustering-Based Approach.* IEEE-NLPKE 2002, Hammamet, Tunisia. pp.156-61.

[194] Sumam M. Idicula and David Peter S. 2007. A Morphological Processor for Malayalam Language. *South Asia Research*. SAGE Publications, Vol. 27, No. 2, 2007. pp. 173-186.

[195] Prabhakar Pandey, Laxmi Kashyap, Manish Sinha, Mahesh Kumar Reddy .R, Pushpak Bhattacharyya. 2004. Hindi Word Sense Disambiguation. *In proceedings of International Symposium on Machine Translation, Natural Language Processing and Translation Support Systems*, Delhi, India. pp.129-134.

[196] Ganesh Ramakrishnan, B. P. Prithviraj, A. Deepa, Pushpak Bhattacharyya, and Soumen Chakrabarti. 2004. "Soft Word Sense Disambiguation", *In proceedings of The Second Global Wordnet Conference 2004 Brno*, Czech Republic. pp. 291-297.

[197] Martinez, D., Agirre, E., and Marquez, L. 2002. Syntactic features for high precision word sense disambiguation. *In proceedings of the*

19th International Conference on Computational Linguistics (Coling 2002). Taipei. pp.626-632.

[198] Gaustad, T., 2004. Linguistic Knowledge and Word Sense Disambiguation. *PhD Thesis.*

[199] Hirst, G. 1987. *Semantic Interpretation and the Resolution of Ambiguity.* Cambridge University Press, Cambridge.

[200] Dahlgren, K., MeDowell, J., and Stabler, Edward P. 1989. Knowledge representation for commonsense reasoning with text. *Computational Linguistics, 15(3)*. Pp.149-170.

[201] Gotoh Y. & Renals S. 2003. Statistical Language Modeling. *Text and speech Triggered Information Access*, S. Renals and G. Grefenstette(eds.), Springer,2003

[202] Jurafsky D. & Martin J. *Speech and Language Processing: An Introduction to speech recognition, computational linguistics and natural language processing*, Prentice-Hall, New Jersey 2003.

[203] Shannon C.E. 1951. Prediction and entropy of printed English. *The bell system technical journal*. January 1951, pp. 50-65

[204] Brown, Peter E, Della Pietra, Stephen A., Della Pietra, Vincent J. and Mercer, Robert L. 1991. Word-sense disambiguation using statistical methods. In *proceedings of 29th Annual Meeting of the Association for Computational Linguistics,* Berkeley. pp 264-270.

[205] Iyer R., Ostendorf M., Meteer M. 1997. Analysing and predicting language model improvements. *In proceedings of the IEEE*

workshop on Automatic Speech Recognition and Understanding Santa Barbara, CA. pp. 254-261.

[206] Chen S., Beeferman D., & Rosenfeld R. 1998. Evaluation Metrics for language models. *Broadcast News Transcription and Understanding Workshop,* February 1998.

[207] Manin D.Y. 2006. Experiments on predictability of word in context and information rate in natural language. *INFORMATION PROCESSES, Electronic Scientific Journal.* pp 229-236.

[208] Marti U.V. and Bunke H. 2001. On the influence of vocabulary size and language models in unconstrained handwritten text recognition. *In Proceedings of the 6th International Conference on Document Analysis and Recognition*, IEEE Computer Society Washington, DC, USA. pp 260 – 265

[209] Resnik P. and Yarowsky D. 1997. A Perspective on word sense disambiguation methods and their evaluation. *In proceedings of ACL-SIGLEX Workshop on Tagging Text with Lexical Semantics: Why, What, and How?. .*Washington, D.C. pp. 79-86.

[210] Resnik P. & Yarowsky D. 1998. Distinguishing Systems and distinguishing senses: New evaluation methods for word sense disambiguation. *Natural language engineering 1(1).* Cambridge University Press.

[211] Kaplan, A. 1955. An experimental study of ambiguity and context. *Mechanical Translation 2(2).* pp. 39-46.

[212] Choueka, Yaacov and Lusignan, Serge. 1985. Disambiguation by short contexts. *Computers and the Humanities, 19*. pp. 147-158.

[213] William A. Gale, Kenneth W. Church, & David Yarowsky. 1992. Using bilingual materials to develop word sense disambiguation methods. *Fourth International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-92), Empiricist vs. rationalist methods in MT*, June 25-27, 1992, Montreal, CCRIT-CWARC. pp.101-112.

[214] McGregor, R. S. 1974. *Outline of Hindi Grammar.* Oxford University Press, Delhi, India.

[215] Durgesh Rao. 1996. Natural Language Generation for English to Hindi Human-Aided Machine Translation of News Stories. *Master's Thesis.* Indian Institute of Technology, Bombay.

[216] Ananthakrishnan Ramanathan and Durgesh Rao. 2003. A Lightweight Stemmer for Hindi. *In proceedings of Workshop on Computational Linguistics for South-Asian Languages, EACL*. pp.201-206.

[217] Ram Viswanadha. 2002. Transliteration of Tamil and Other Indic Scripts. *In proceedings of Tamil Internet 2002, California, USA.* pp 277-285.

[218] Kevin Knight and Ishwar Chander. 1994. Automated Post-Editing of Documents. *In Proceedings of the 12th National Conference on Artificial Intelligence*, Seattle, WA, pp. 779–784.

[219] Jeff Allen and Christopher Hogan. 2000. Toward the Development of a Post-Editing Module for Raw Ma-chine Translation Output: A Controlled Language Perspective. *In proceedings of the Third International Workshop on Controlled Language Applications*. Seattle, WA, pp. 62–71.

[220] Johann Roturier. 2009. Controlled Language for MT in Action. *Presentation given at Translingual Europe 2009, Prague. [Internet Source:* http://ufal.mff.cuni.cz/tle2009/presentations/roturier-controlled-language-for-mt-in-action.pptx]

[221] Johann Roturier and Jean Senellart. 2008. Automatic Post-Editing: Review of Translation Quality Gains. *Presentation given at LISA Forum 2008.*

[222] Dublin. Johann Roturier. 2006. An Investigation into the Impact of Controlled English Rules on the Comprehensibility, Usefulness, and Acceptability of Machine-Translated Technical Documentation for French and German Us-ers. *Unpublished PhD thesis*. Dublin City University, Ireland

[223] Johann Roturier, Sylke Krämer, and Heidi Düchting. 2005. Machine Translation: The translator's choice. *In Proceedings of the 10th LRC conference, Limerick, Ireland.*

[224] Karttunen L., Chanod J-P., Grefenstette G., Schiller A. 1996. Regular Expressions for Language Engineering, *Natural Language Engineering*, Cambridge University Press, pp. 305-328.

[225] Oflazer K., Yilmaz Y, Vi-xfst. 2004. A Visual Regular Expression Development Environment for Xerox Finite State Tool, *In proceedings of the Seventh Meeting of the ACL Special Interest Group in Computational Phonology, Association for Computational Linguistics*, Barcelona, Spain, July 2004, pp 86-93.

[226] Hasan S., Ney H. 2005. Clustered language models based on regular expressions for SMT. *In proceedings of 10th EAMT conference Practical applications of Machine Translation*, 30-31 May 2005, Budapest; pp. 119-125

[227] Jean Senellart, Jin Yang, and Anabel Rebollo. 2003. SYSTRAN intuitive coding technology. *In proceedings of MT Summit IX*, New Orleans, USA, 23-27 September. pp. 346-353.

[228] Pierre Senellart and Jean Senellart. 2005. SYSTRAN Translation Stylesheets: Machine Translation driven by XSLT. *In proceedings XML Conference & Exposition*, Atlan-ta, USA, November 2005.

[229] Nuebel, Rita. 1998. MT Evaluation in Research and Industry: Two Case Studies. *In proceedings of 14th Twente Workshop on Language Technology in Multimedia Information Retrieval*, December 1998, University of Twente, The Netherlands.

[230] EAGLES.1996. Expert Advisory Group on Language Engineering. *Evaluation of Natural Language Processing Systems (Final Report).* Prepared for DG XIII of the European Commission.

[231] Roudaud B., Puerta M. C., and Gamrat O. 1993. A procedure for the evaluation and improvement of an MT system by the end-user. *Machine Translation, Volume 8 Number 1-2*, March 1993, pp 109-116.

[232] Wagner S. 1998. Small Scale Evaluation Methods. In: Rita Nübel; Uta Seewald-Heeg (eds.): *Evaluation of the Linguistic Performance of Machine Translation Systems. Proceedings of the Workshop at the KONVENS-98.* Bonn: 1998, 93-105.

[233] Keiji Yasuda, Fumiaki Sugaya, Toshiyuki Takezawa, Seiichi Yamamoto & Masuzo Yanagida. 2001. An automatic evaluation method of translation quality using translation answer candidates queried from a parallel corpus. *In proceedings of MT Summit VIII: Machine Translation in the Information Age*, Santiago de Compostela, Spain, 18-22 September 2001. pp.373-378.

[234] Yasuhiro A., K. Imamura and E. Sumita. 2001. Using multiple edit distances to automatically rank Machine Translation output. *In proceedings of MT Summit VIII*, *2001.* pp. 15–20.

[235] Papineni, K., S. Roukos, T. Ward, W.-J. Zhu. 2002. BLEU: a method for automatic evaluation of Machine Translation. *In proceedings of the 40th Annual Meeting of the Association for the Computational Linguistics (ACL),* Philadelphia, July 2002, pp. 311-318.

[236] NIST Report. 2002. Automatic Evaluation of Machine Translation Quality Using *N-gram* Co-Occurrence Statistics. [Internet Source: http://www.nist.gov/speech/tests/mt/doc/ngramstudy.pdf]

[237] Melamed I. D., Joseph P. T., Luke S. 2003. Evaluation of Machine Translation and its Evaluation. *In the proceedings of MT Summit IX, New Orleans, USA, 23-28 September 2003.* pp.175-181.

[238] Nieben, S., Och, F.J., Leusch, G., and Ney, H. 2000. An Evaluation Tool for Machine Translation: Fast Evaluation for MT Research. *In Proceedings of the Second International Conference on Language Resources and Evaluation,* LREC 2000, Athens, Greece, vol. 1, pp. 39-45.

[239] Yokoyama, S., Kumano, A., Matsudaira, M., Shirokizawa, Y., Kawagoe, M., Kodama, S., Kashioka, H., Ehara, T., Miyazawa, S., Nakajima, Y. 1999. Quantitative Evaluation of Machine Translation using Two-way MT. *In proceedings of MT Summit-VII*, September 13-17, Kent Ridge Digital Labs, Singapore 1999. pp. 132-140.

[240] Balkan L. 1998. Test suites: some issues in their use and design. *In proceedings of the International conference on Machine Translation: ten years on,* Cranfield University, England, 12-14 November 1994. pp.214-222.

[241] Arnold, D., Balkan, L., Humphreys, R. Lee, Meijer, S., and Sadler, L. 1995. *Machine Translation: An Introductory Guide*. NCC Blackwell, Manchester, Oxford.

[242] Elliott D., Hartley A., and Atwell E. 2003. Rationale for a multilingual aligned corpus for Machine Translation evaluation. *In proceedings of International Conference on Corpus Linguistics (CL2003)*, Lancanster, UK. pp. 191-200.

[243] Zhang Y. and Vogel S. 2004. Measuring Confidence Intervals for the Machine Translation Evaluation Metrics. *In International Conference on Theoretical and Methodological Issues in Machine Translation (TMI 2004). Baltimore, MD*. pp. 189-194.

[244] Paula Estrella, Olivier Hamon, & Andrei Popescu-Belis, 2007. How much data is needed for reliable MT evaluation? Using bootstrapping to study human and automatic metrics. *In proceedings of MT Summit XI*, 10-14 September 2007, Copenhagen, Denmark. pp.167-174.

[245] Halliday, T. & Briss, E. 1977. The Evaluation and Systems Analysis of the Systran Machine Translation System. *Report RADC-TR-76-399, January, 1977.* Rome Air Development Center, Griffiss Air Force base, New York.

[246] Chris Callison-Burch, Miles Osborne, and Philipp Koehn, 2006. Re-evaluating the Role of BLUE in Machine Translation Research. *In Proceedings of EACL-2006.*pp.145-151.

[247] Ananthakrishnan R, Pushpak Bhattacharyya, M. Sasikumar, Ritesh Shah. 2007. Some Issues in Automatic Evaluation of English-Hindi MT: more blues for BLEU. *In proceeding of 5th International*

Conference on Natural Language Processing (ICON-07), Hyderabad, India. pp.135-39.

[248] Vamshi Ambati and Rohini U. 2007. A Hybrid approach to example based Machine Translation for Indian Languages. I*n Proceedings of 5th International conference on natural language processing (ICON-2007)*, January 4-6, 2007, IIIT Hyderabad. pp. 146-151.

[249] Gangadharaia R and Balakrishanan N. 2006. Application of linguistic rules to generalized example based Machine Translation for Indian languages. *In proceedings of first National symposium oon modeling and shallow parsing of Indian languages, (MSPIL),* India, Mumbai, April 2006. pp-34-39

[250] Singh Anil K. and Surana Harshit. 2007. Can Corpus Based Measures be Used for Comparative Study of Languages?. *In Proceedings of Ninth Meeting of the ACL Special Interest Group in Computational Morpholgy and Phonology, Prague, June 2007.* pp. 40-47.

Publications Based on the Work Presented in this Thesis

Journals

- Vishal Goyal, G S Lehal, “Advances in Machine Translation Systems”, Language In India, Volume 9, November 2009 Issue, pp. 138-150(2009)
- V Goyal and G S Lehal, “Evaluation of Hindi to Punjabi Machine Translation System”, International Journal of Computer Science Issues, France, Volume 4, September 2009.
- V Goyal and G S Lehal, “A Machine Transliteration System for Machine Translation System : An Application on Hindi-Punjabi Language Pair”, *Atti Della Fondazione Giorgio Ronchi (Italy)*, Volume LXIV, No. 1, pp. 27-35 (2009)
- V Goyal and G S Lehal “Comparative Study of Hindi and Punjabi Language Scripts”, *Nepalese Linguistics,* Journal of the Linguistics Society of Nepal, Volume 23, pp. 67-82 (2008).
- Vishal Goyal and Gurpreet Singh Lehal, “Web Based Hindi to Punjabi Machine Translation System”, Journal of Emerging Technologies in Web Intelligence, Volume 2, Number 2, May 2010 (Accepted , to be Published).

Conference

- V. Goyal and G. S. Lehal, "Hindi Morphological Analyzer and Generator", *Proceedings First International Conference on Emerging Trends in Engineering and Technology*, Nagpur, IEEE Computer Society Press, California, USA, pp. 1156-1159 (2008).

Appendix A

Graphic User Interface and Extended Features

Information technology in the current scenario is evolving as an effective tool for making information wide spread and available *on-line* to several communities at large. On one hand, the increased use of ICT is enabling people across the globe to participate in the knowledge network; at the same time larger populations in the rural areas of developing country like India are being deprived of the benefits of the use of ICT. One of the main reasons behind this seems to be the *language barrier*. For such cause, Hindi to Punjabi Machine Translation system can play an important role to reduce digital divide due to the language barrier. This lessening of digital divide and increasing the accessibility of information present in the Internet happens to be one of the objectives of our work among the various aims of this research.

We have made our Hindi to Punjabi Machine Translation available online free of cost for use world wide. Our System is capable of doing following tasks:

1. Text translation from Hindi to Punjabi
2. Text transliteration from Hindi to Punjabi
3. Translating Websites
4. Sending Email in Punjabi Language originally written in Hindi language.

Above tasks will be discussed in detail in following sections. Following Screenshot shows the GUI for the Hindi to Punjabi Machine Translation System:

Figure A.1: GUI for Hindi to Punjabi Machine Translation System

Text translation from Hindi to Punjabi

This facility enables the users to translate the input text into Punjabi language text on clicking the Translate button. The text can be input in the textarea there through various modes viz. browsing and reading the text file, typing the text using keyboard and by using the typing pad provided on the interface. It is also an added feature that the text can be entered in a mixed way i.e. English text can also be embedded in between the Hindi text. For the ease of users,

the text can also be typed in four different ways. Care has been taken for professionals or users who are habitual of typing Hindi text in Krutidev or AnmolLipi Font. As it is clear from the above GUI, there is an option of choosing keyboard mapping for typing showing three values in the drop down besides it. The dropdown has the values Krutidev, AnmolLipi and Roman. For instance, if the user chooses the Krutidev option for keyboard mapping style, it enables the user to type the text using the keyboard mapping similar to Krutidev character mapping on the keyboard. For example – on pressing the key 'v' , the character 'अ' will be typed. The difference is the typed text will be in Unicode encoding rather in Krutidev font. It is very interesting that if the user chooses the Roman option, it facilitates the user to type a word just the way it sounds in Hindi language using English letters and once the typing of a word is finished, hit the SPACE bar, the word will be converted to Hindi language script. For example, typing "hamesha" transliterates into Hindi as: "हमेशा ". We have enabled it on our System interface using the Google Indic Transliteration APIs available at the Google website. Another way for entering in input text is by using the typing pad available on the interface. User needs to just click the appropriate buttons corresponding to each character to be typed. Now the text has been entered using any one of the options mentioned above and on pressing the "Translate" button, the input text is translated into Punjabi language. Following screen shot demonstrates text translation facility in our system:

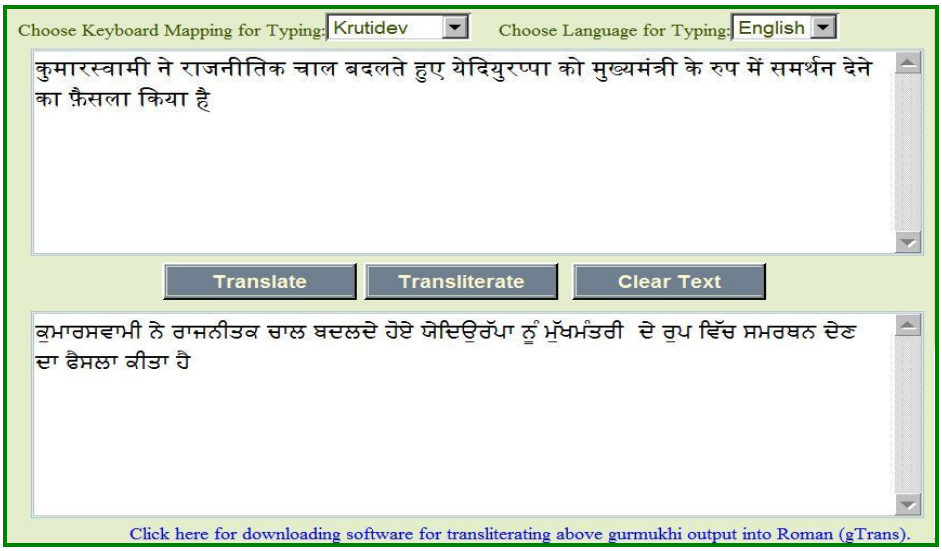

Figure A.2: Screenshot for translation facility of the system

Text transliteration from Hindi to Punjabi:

Transliteration is the process of converting a word written in one language into another language. Transliteration is distinct from translation, which involves a change in language while maintaining the meaning of the word; transliteration instead converts the sound of the word from one language to another. The option of a transliteration component is to enable the well developed poetic verse in the Hindi language to be available to the Punjabi literate public. The transliteration facility in our system can be used in similar manner as explained above for translation facility. The only difference is that user will click the transliterate button for transliterating the text from Hindi to Punjabi text. Following screen shot demonstrates the transliteration feature of the system:

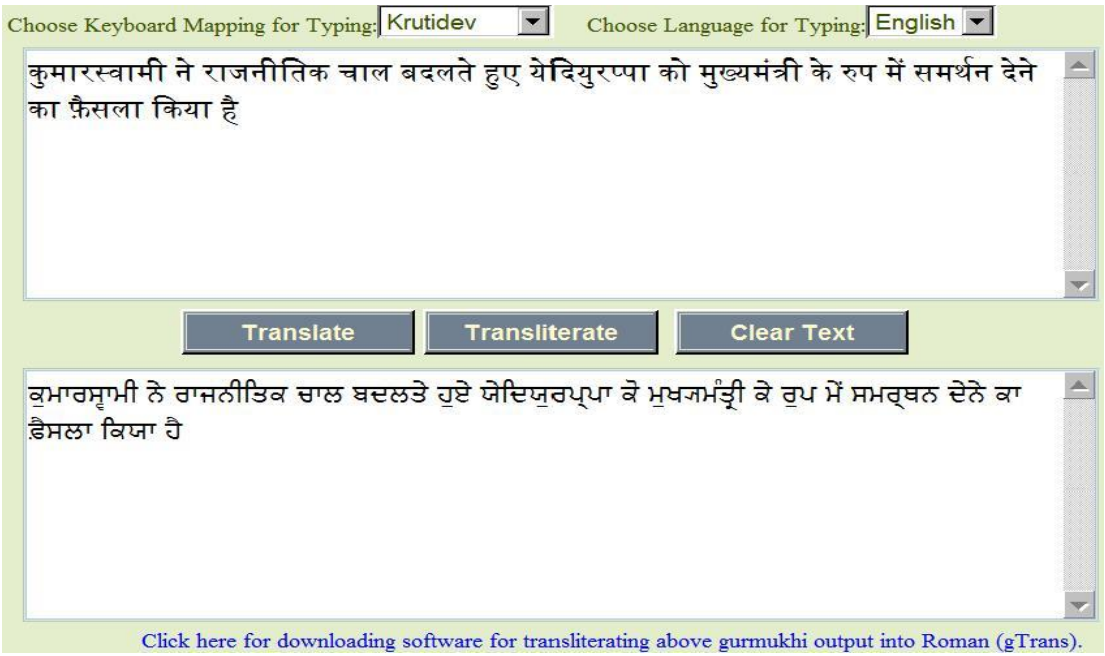

Figure A.3: Screenshot for transliteration facility of the system

Website Translation

Using this facility, user can translate an entire web page directly, simply by entering the URL and clicking Translate button. This facility is available on the home page of our system displayed at the lower right most corner of the GUI. The user can submit a URL of a Hindi website of his/her interest for translation, then clicks the translate button present besides the textbox where user has entered the URL. Then the translation request is processed at the web server and after translation, translated website is displayed to the user. The important aspect in this feature is that the format of the website is retained after translation. On translated webpage, the links can be further clicked to process them similar to the one that has been translated. It gives the user a feeling that they are browsing Punjabi website. The implementation

of this task includes modules - retrieving and parsing HTML Page, translating, combining the translation unit with HTML Codes, altering the links in webpage, displaying the result. Retrieving and Parsing HTML Page includes first downloading the html source code from the server and then extracting the text out of the html tags for processing. Then the text present in the HTML tags is translated using the text translation module mentioned above. Altering the links in webpage is very important process in it. Here, all the links in webpage are replaced, so that the next links must redirect the request through our translation service. By this step, user does not need to enter URLs or take any other action if user wants to translate the linked page. The user simply needs to click on the given link. Translated webpage is then forwarded to the client in the same format in which the original page had appeared. Following screen shots demonstrate the website translation feature:

Figure A.4: Screenshot for website translation facility of the system

Figure A.5: Screen shot of Original Website http://www.webdunia.com/ accessed on 27/12/2009 at 08:40 PM IST

Figure A.5: Screen shot of translated version by the system for website shown in Figure A.4

Sending Email in Punjabi originally written in Hindi language:

This facility enables the user to write the email (text) in Hindi language and this text can be emailed to the recipient either in same language or in Punjabi language after translating the original text. It has very real application in sense that sender knows Hindi and wants to communicate some information to target recipient who knows only Punjabi. For this purpose, the sender can write the email in Hindi language and while sending the email, can send the email in Punjabi by clicking the option of sending the email after translating into Punjabi. Thus, recipient will receive the email in Punjabi. The message is communicated as per the

convenience of the sender and recipient both. Following screen shot demonstrates this feature:

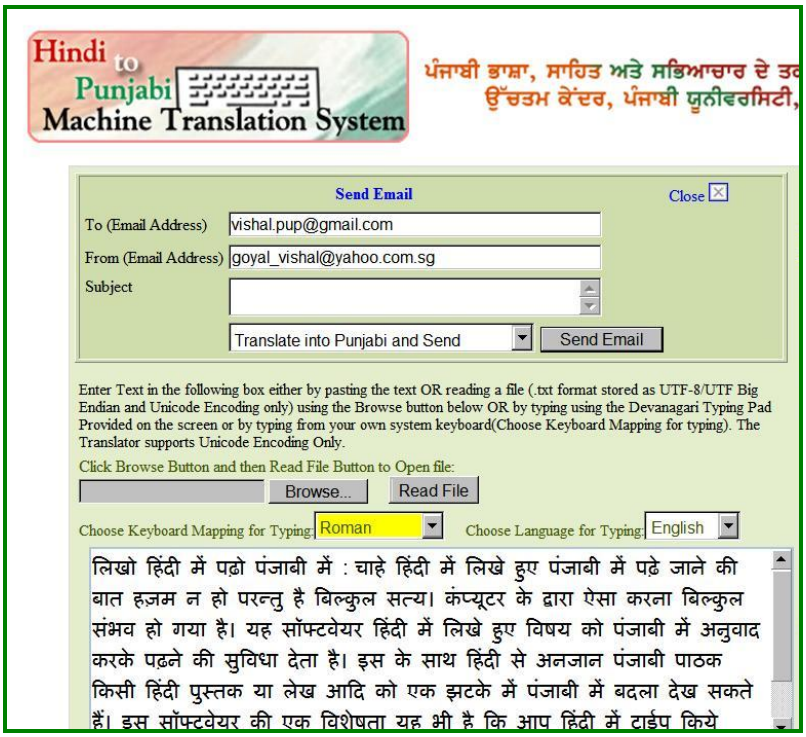

Figure A.7: Screenshot for Email Sending facility of the system

Appendix B

Test Data Set for Intelligibility Test

Intelligibility Evaluation:

The evaluators do not have any clue about the source language i.e. Hindi Language. They judge each sentence (in target language i.e. Punjabi) on the basis of its comprehensibility. The target user is a layman who is interested only in the comprehensibility of translations. Intelligibility is effected by grammatical errors, miss-translations, and un-translated words.

Scoring:

The scoring is done based on the degree of intelligibility and comprehensibility. A Four point scale is made in which highest point is assigned to those sentences that look perfectly alike the target language and lowest point is assigned to the sentence which is un-understandable. Detail is a follows:

Score 3 : The sentence is perfectly clear and intelligible. It is grammatical and reads like ordinary text.

Score 2: The sentence is generally clear and intelligible. Despite some inaccuracies, one can understand immediately what it means.

Score 1: The general idea is intelligible only after considerable study. The sentence contains grammatical errors &/or poor word choice.

Score 0: The sentence is unintelligible. Studying the meaning of the sentence is hopeless. Even allowing for context, one feels that guessing would be too unreliable.

Intelligibity Test -News

S.No.	Sentence	0	1	2	3
1.	ਮੁੰਬਈ । ਰਿਜਰਵ ਬੈਂਕ ਨੇ ਸ਼ੁੱਕਰਵਾਰ ਨੂੰ ਕਿਹਾ ਕਿ ਸੰਸਾਰਿਕ ਆਰਥਕ ਸੰਕਟ ਦੇ ਪ੍ਰਭਾਵ ਨੂੰ ਦੇਸ਼ ਦੀ ਮਾਲੀ ਹਾਲਤ ਦੇ ਉੱਬਰਣ ਦੇ ਬਾਅਦ ਉਹ ਮੁਦਰਾਸਫੀਤੀ ਦੇ ਅੰਦਾਜੀਆਂ ਅਤੇ ਮੱਧ ਕਾਲ ਵਿੱਚ ਇਸਦੇ ਨਤੀਜੀਆਂ ਦੇ ਪਰਬੰਧਨ ਉੱਤੇ ਧਿਆਨ ਦੇਵੇਗਾ ।				
2.	ਵਾਸ਼ਿੰਗਟਨ । ਅਮਰੀਕੀ ਬੈਂਕਾਂ ਦੇ ਸਟਰੇਸ ਟੇਸਟ ਦਾ ਨਤੀਜਾ ਆਖ਼ਿਰਕਾਰ ਆ ਹੀ ਗਿਆ ।				
3.	ਜਿਨ੍ਹਾਂ ਬੈਂਕਾਂ ਨੂੰ ਪੂੰਜੀ ਦੀ ਲੋੜ ਹੈ , ਉਨ੍ਹਾਂਨੂੰ ਯੋਜਨਾ ਬਣਾਉਣ ਲਈ 8 ਜੂਨ ਤੱਕ ਦਾ ਸਮਾਂ ਦਿੱਤਾ ਗਿਆ ਹੈ ।				
4.	ਬੈਂਕਾਂ ਨੂੰ ਇਹ ਯੋਜਨਾ ਆਪਣੇ ਨਿਆਮਕਾਂ ਤੋਂ ਮਨਜ਼ੂਰ ਕਰਵਾਣੀ ਹੋਵੇਗੀ ।				
5.	ਅਧਿਕਾਰੀਆਂ ਦਾ ਕਹਿਣਾ ਹੈ ਕਿ ਆਰਥਕ ਹਾਲਤ ਵਿੱਚ ਸੁਧਾਰ ਲਈ ਮਜਬੂਤ ਬੈਂਕਿੰਗ ਤੰਤਰ ਜਰੂਰੀ ਹੈ ।				
6.	ਇਸ ਟੇਸਟ ਤੋਂ ਨਿਵੇਸ਼ਕਾਂ ਵਿੱਚ ਇਹ ਭਰੋਸਾ ਪਰਤੇਗਾ ਕਿ ਸਾਰੇ ਬੈਂਕ ਕਮਜੋਰ ਨਹੀਂ ਹਨ ।				
7.	ਨਾਲ ਹੀ ਕਮਜੋਰ ਬੈਂਕਾਂ ਵਿੱਚ ਵੀ ਸੁਧਾਰ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ ।				
8.	ਵਿਕਰਮ ਪੰਡਤ ਦੀ ਅਗੁਵਾਈ ਵਾਲੀ ਸਿਟੀ ਨੇ ਕਿਹਾ ਕਿ ਉਹ 5 . 5 ਅਰਬ ਡਾਲਰ ਇਲਾਵਾ ਪੂੰਜੀ ਜੁਟਾਣ ਲਈ ਪਬਲਿਕ ਏਕਸਚੇਂਜ ਆਫਰ ਦਾ ਦਾਇਰਾ ਵਧਾਏਗੀ ।				
9.	ਨਿਊਯਾਰਕ । ਅਮਰੀਕੀ ਸ਼ੇਅਰ ਬਾਜ਼ਾਰ ਵੀਰਵਾਰ ਨੂੰ ਗਿਰਾਵਟ ਦੇ ਨਾਲ ਬੰਦ ਹੋਏ ।				

10.	ਉੱਧਰ , ਵੀਰਵਾਰ ਨੂੰ ਬਾਜ਼ਾਰ ਬੰਦ ਹੋਣ ਦੇ ਬਾਅਦ ਸਰਕਾਰ ਨੇ ਪ੍ਰਮੁੱਖ ਬੈਂਕਾਂ ਦੇ ਸਟਰੇਸ ਟੇਸਟ ਦੇ ਨਤੀਜੇ ਘੋਸ਼ਿਤ ਕੀਤੇ ਜਿਨ੍ਹਾਂ ਦੇ ਮੁਤਾਬਕ ਦੇਸ਼ ਦੇ 10 ਪ੍ਰਮੁੱਖ ਬੈਂਕਾਂ ਨੂੰ ਆਪਣੇ ਬਚਾਵ ਲਈ ਹੋਰ ਨਗਦੀ ਇਕੱਠੀ ਕਰਨੀ ਪਵੇਗੀ ।				
11.	ਵਰੂਣ ਗਾਂਧੀ ਨੇ ਪੀਲੀਭੀਤ ਵਿੱਚ ਮੁਸਲਮਾਨਾਂ ਦੇ ਖਿਲਾਫ ਭੜਕਾਊ ਭਾਸ਼ਣ ਦਿੱਤੇ ਸਨ				
12.	ਇਸ ਮਾਮਲੇ ਦੇ ਪ੍ਰਕਾਸ਼ ਵਿੱਚ ਆਉਣ ਦੇ ਬਾਅਦ ਚੋਣ ਕਮਿਸ਼ਨ ਦੇ ਨਿਰਦੇਸ਼ ਉੱਤੇ ਵਰੁਣ ਗਾਂਧੀ ਦੇ ਖਿਲਾਫ ਏਫ਼ਆਈਆਰ ਦਰਜ ਹੋਈ ਸੀ .				
13.	ਪਹਿਲਾਂ ਹੀ ਵਿੱਤੀ ਸਾਲ ਯਾਨੀ ਸਾਲ 2004 - 05 ਵਿੱਚ ਕੇਂਦਰ ਨੇ ਰਾਜ ਸਰਕਾਰ ਨੂੰ 2831 . 82 ਕਰੋੜ ਰੁਪਏ ਦੀ ਰਾਸ਼ੀ ਉਪਲੱਬਧ ਕਰਾਈ .				
14.	ਸਪਾ ਲਈ ਪਿਛਲੇ ਚੋਣ ਵਿੱਚ ਜਿੱਤੀ 32 ਵਿੱਚੋਂ ਆਪਣੀ 15 ਸੀਟਾਂ ਨੂੰ ਬਚਾਉਣ ਦੀ ਚੁਣੋਤੀ ਹੈ , ਜਿਨ੍ਹਾਂ ਵਿਚੋਂ ਲੱਗਭੱਗ ਅੱਧਾ ਦਰਜਨ ਸੀਟਾਂ ਦੀ ਹਾਰ - ਜਿੱਤ ਤਾਂ ਕਲਿਆਣ ਫੈਕਟਰ ਦੀ ਕਸੌਟੀ ਉੱਤੇ ਹੀ ਕੱਸਿਆ ਜਾਵੇਗਾ ।				
15.	ਉਨ੍ਹਾਂਨੇ ਭਾਜਪਾ ਨੇਤਾਵਾਂ ਨੂੰ ਇਸ ਤਰ੍ਹਾਂ ਦੇ ਬਿਆਨ ਨਹੀਂ ਦੇਣ ਦੀ ਸਲਾਹ ਦਿੱਤੀ । ਇਸਦੇ ਨਾਲ ਹੀ ਸ਼ਰਦ ਯਾਦਵ ਨੇ ਵਿਦੇਸ਼ੀ ਬੈਂਕਾਂ ਵਿੱਚ ਭਾਰਤੀਆਂ ਦੇ ਜਮਾਂ ਕਾਲੇ ਧਨ ਦਾ ਮੁੱਦਾ ਫਿਰ ਚੁੱਕਿਆ ।				
16.	ਰਾਹੁਲ ਨੇ ਕਿਹਾ , ਮਨਮੋਹਨ ਸਿੰਘ ਸਾਡੇ ਪ੍ਰਧਾਨਮੰਤਰੀ ਹਨ , ਉਹ ਯੂਪੀਏ ਦੇ ਵੀ ਪ੍ਰਧਾਨਮੰਤਰੀ ਹੈ .				
17.	ਰਾਹੁਲ ਗਾਂਧੀ ਦੋ ਦਿਨ ਦੇ ਚੋਣ ਪ੍ਰਚਾਰ ਉੱਤੇ ਰਾਜਸਥਾਨ ਵਿੱਚ ਹਨ .				

18.	ਆਰਥਕ ਸੰਕਟ ਦਾ ਦਬਾਅ ਝੇਲਣ ਦੀ ਸਮਰੱਥਾ ਆਂਕਣ ਵਾਲੇ ਇਸ ਟੇਸਟ ਵਿੱਚ ਅਮਰੀਕਾ ਦੇ 10 ਵੱਡੇ ਬੈਂਕ ਬੇਦਮ ਨਿਕਲੇ ਹਨ ।				
19.	ਕੰਪਨੀ ਦੀ ਇਸ ਪਹਿਲ ਤੋਂ ਉਸਨੂੰ ਹੋਰ ਸਰਕਾਰੀ ਸਹਾਇਤਾ ਜਾਂ ਸਰਕਾਰੀ ਪ੍ਰਤੀਭੂਤੀਯੋਂ ਨੂੰ ਇੱਕੋ ਜਿਹੇ ਸ਼ੇਅਰਾਂ ਵਿੱਚ ਪਰਿਵਰਤਿਤ ਕੀਤੇ ਬਿਨਾਂ ਆਪਣਾ ਪੂੰਜੀ ਆਧਾਰ ਵਧਾਉਣ ਵਿੱਚ ਮਦਦ ਮਿਲੇਗੀ ।				
20.	ਉਨ੍ਹਾਂਨੇ ਕਿਹਾ ਕਿ ਰਾਜਗ ਦੁਆਰਾ ਇਸਨੂੰ ਚੁਨਾਵੀ ਮੁੱਦਾ ਬਣਾਏ ਜਾਣ ਦੇ ਬਾਅਦ ਮਜਬੂਰੀ ਵਿੱਚ ਮਨਮੋਹਣ ਸਰਕਾਰ ਹੁਣ ਕਾਰਵਾਈ ਕਰਣ ਦਾ ਦਿਖਾਵਾ ਕਰ ਰਹੀ ਹੈ ।				
21.	ਸੰਪਾਦਕਾਂ ਦੇ ਮੁਤਾਬਕ , ਟਾਇਮ 100 ਸੰਸਕਰਣ ਵਿੱਚ ਅਸੀ ਉਨ੍ਹਾਂ ਲੋਕਾਂ ਦਾ ਨਾਮ ਦਿੰਦੇ ਹਨ ਜੋ ਸਾਡੀ ਦੁਨੀਆ ਨੂੰ ਸਭਤੋਂ ਜ਼ਿਆਦਾ ਪ੍ਰਭਾਵਿਤ ਕਰਦੇ ਹਾਂ .				
22.	ਮੇਗਾਸਟਾਰ ਅਮੀਤਾਭ ਬੱਚਨ ਨੇ ਕੱਲ ਆਪਣੇ ਸੰਵਿਧਾਨਕ ਫਰਜ ਦਾ ਗੁਜਾਰਾ ਕਰਣ ਦੇ ਨਾਲ ਹੀ ਆਪਣੇ ਸਾਮਾਜਕ ਫਰਜ ਦਾ ਵੀ ਬਖੂਬੀ ਗੁਜਾਰਾ ਕੀਤਾ ।				
23.	ਇਸਦੇ ਇਲਾਵਾ ਉਪ ਮੁੱਖਮੰਤਰੀ ਸੁਖਬੀਰ ਬਾਦਲ ਨੇ ਵੀ ਇੰਟਰਨੇਟ ਉੱਤੇ ਕਈ ਪ੍ਰੋਫਾਇਲ ਬਣਾ ਰੱਖੀ ਹੈ ।				
24.	ਹਾਲ ਹੀ ਵਿੱਚ ਲਤਾ ਮੰਗੇਸ਼ਕਰ ਨੇ ਮਧੁਰ ਭੰਡਾਰਕਰ ਦੀ ਫਿਲਮ ‘ਜੇਲ੍ਹ’ ਵਿੱਚ ਇੱਕ ਧਾਰਮਿਕ ਗੀਤ ਰਿਕਾਰਡ ਕੀਤਾ ਹੈ .				
25.	ਹਿਮੇਸ਼ ਜੀ , ਗੱਲ ਤਾਂ ਠੀਕ ਹੈ ਲੇਕਿਨ ਕਰਜ ਦੇ ਹਾਲ ਦੇ ਬਾਅਦ ਤੁਹਾਨੂੰ ਨਹੀਂ ਲੱਗਦਾ ਕਿ ਦਰਸ਼ਕਾਂ ਦਾ ਤੁਹਾਨੂੰ ਹੀਰੋ ਦੇ ਰੁਪ ਵਿੱਚ ਸਵੀਕਾਰ ਕਰਣਾ ਥੋੜ੍ਹਾ ਮੁਸ਼ਕਲ ਹੋਵੇਗਾ .				
26.	ਉਨ੍ਹਾਂਨੇ ਵੀ ਲੋਕਸਭਾ ਚੋਣ ਵਿੱਚ ਜਿੱਤ ਦਾ ਦਾਵਾ ਕੀਤਾ ਹੈ ।				

27.	ਫਿਲਮ ਸਲਮਡਾਗ ਮਿਲਿਅਨੇਇਰ ਦੀ ਬਾਲ ਕਲਾਕਾਰ ਰੁਬੀਨਾ ਅਲੀ ਕਾਫ਼ੀ ਪ੍ਰਸਿੱਧ ਹੋ ਗਈ ਹੈ				
28.	ਮੈਂ ਇਸ ਪਰਵਾਰ ਨੂੰ ਪਿਛਲੇ ਵੀਹ ਸਾਲਾਂ ਤੋਂ ਜਾਣਦਾ ਹਾਂ , ਰਫੀਕ ਬਹੁਤ ਸ਼ਰੀਫ ਆਦਮੀ ਹੈ , ਉਹ ਅਜਿਹੀ ਹਰਕੱਤ ਕਦੇ ਨਹੀਂ ਕਰੇਗਾ				
29.	ਕਰੀਨਾ ਪੂਰੇ ਦਿਨ ਘਾਹ ਨਾਲ ਬਣੇ ਮਚਾਣ ਉੱਤੇ ਤਾਂ ਕਦੇ ਬੈਲਗਾੜੀ ਪਰ ਮਸਤੀ ਕਰਦੀ ਨਜ਼ਰ ਆਈਆਂ ।				
30.	ਸ਼ੁੱਕਰਵਾਰ ਨੂੰ ਕਰੀਨਾ ਦੇ ਆਉਣ ਦੇ ਨਾਲ ਹੀ ਸ਼ਹਿਰ ਵਿੱਚ ਸੈਫ ਦੇ ਵੀ ਆਉਣ ਦੀ ਉੰਮੀਦ ਲਗਾਈ ਜਾ ਰਹੀ ਸੀ ।				
31.	ਹਾਲ ਵਿੱਚ ਮੋਹਮੰਦ ਵਿੱਚ ਹੋਈ ਫੌਜੀ ਕਾਰਵਾਈ ਵਿੱਚ 18 ਚਰਮਪੰਥੀ ਮਾਰੇ ਗਏ ਸਨ				
32.	ਧਿਆਨ ਯੋਗ ਹੈ ਕਿ ਪਾਕਿਸਤਾਨ ਦੇ ਪਸ਼ਚਿਮੋੱਤਰ ਵਿੱਚ ਬੁਨੇਰ ਵਿੱਚ ਪਿਛਲੇ ਕੁੱਝ ਹਫਤੀਆਂ ਵਿੱਚ ਫੌਜ ਅਤੇ ਤਾਲੇਬਾਨ ਚਰਮਪੰਥੀਆਂ ਦੇ ਵਿੱਚ ਭੀਸ਼ਨ ਸੰਘਰਸ਼ ਹੋਇਆ ਹੈ				
33.	ਇਸਲਾਮਾਬਾਦ । ਪਾਕਿਸਤਾਨ ਦੇ ਗ੍ਰਹ ਮੰਤਰੀ ਰਹਿਮਾਨ ਮਲਿਕ ਨੇ ਸੋਮਵਾਰ ਨੂੰ ਕਿਹਾ ਕਿ ਦੇਸ਼ ਦੇ ਬੇਚੈਨ ਪਸ਼ਚਿਮੋੱਤਰ ਖੇਤਰ ਵਿੱਚ ਚੱਲ ਰਹੇ ਇੱਕ ਵੱਡੇ ਫੌਜੀ ਅਭਿਆਨ ਵਿੱਚ ਕਰੀਬ 700 ਤਾਲਿਬਾਨ ਆਤੰਕੀਆਂ ਨੂੰ ਮਾਰ ਗਿਰਾਇਆ ਗਿਆ ਹੈ ਅਤੇ ਸਾਰੇ ਆਤੰਕੀਆਂ ਦਾ ਖਾਤਮਾ ਹੋਣ ਤੱਕ ਉੱਥੇ ਫੌਜੀ ਕਾਰਵਾਈ ਜਾਰੀ ਰਹੇਗੀ ।				
34.	ਗ੍ਰਹ ਮੰਤਰੀ ਨੇ ਕਿਹਾ ਕਿ ਇਹ ਪੂਰੇ ਦੇਸ਼ ਲਈ ਇੱਕ ਪਰੀਖਿਆ ਹੈ ।				
35.	ਪਾਕਿਸਤਾਨ ਦੇ ਸੀਮਾਵਰਤੀ ਇਲਾਕੀਆਂ ਵਿੱਚ ਮਿਸਾਈਲ ਹਮਲੇ ਹੁੰਦੇ ਰਹੇ ਹਨ ਅਤੇ ਇਸਦੇ ਲਈ ਪਾਕਿਸਤਾਨ ਅਮਰੀਕਾ ਤੇ ਇਲਜ਼ਾਮ ਲਗਾਉਂਦਾ ਰਿਹਾ ਹੈ .				

36.	ਪ੍ਰਧਾਨਮੰਤਰੀ ਦੇ ਬਿਆਨ ਦੀ ਭਾਜਪਾ ਨੇ ਵੀ ਆਲੋਚਨਾ ਕੀਤੀ ਹੈ । ਦਿੱਲੀ ਪ੍ਰਦੇਸ਼ ਦੇ ਪ੍ਰਧਾਨ ਮੰਤਰੀ ਆਰਪੀ ਸਿੰਘ ਨੇ ਕਿਹਾ ਕਿ ਪ੍ਰਧਾਨਮੰਤਰੀ ਨੇ ਪਦ ਦੀ ਗਰਿਮਾ ਧੂਮੀਲ ਕੀਤੀ ਹੈ ।				
37.	ਪ੍ਰਧਾਨਮੰਤਰੀ ਦੇ ਸਰਕਾਰੀ ਘਰ ਤੇ ਹੋਈ ਇਹ ਗੱਲਬਾਤ ਇਸ ਮਾਅਨੇ ਵਿੱਚ ਮਹੱਤਵਪੂਰਣ ਹੈ ਕਿ ਮਾਓਵਾਦੀ ਨੇਤਾ ਨੇ ਭਾਰਤ ਤੇ ਨੇਪਾਲ ਦੇ ਅੰਦਰੂਨੀ ਮਾਮਲੇ ਵਿੱਚ ਦਖੱਲੰਦਾਜੀ ਦਾ ਇਲਜ਼ਾਮ ਲਗਾਇਆ ਸੀ ਜਿਨ੍ਹੰ ਬਾਅਦ ਵਿੱਚ ਉਨ੍ਹਾਂਨੇ ਹਲਕਾ ਕਰਣ ਦੀ ਕੋਸ਼ਿਸ਼ ਕੀਤੀ ਸੀ ।				
38.	ਗੁਜਰਾਤ ਵਿੱਚ 2002 ਵਿੱਚ ਹੋਏ ਦੰਗੀਆਂ ਦੇ ਕੁੱਝ ਮਾਮਲੀਆਂ ਵਿੱਚ ਨਿੱਤ ਸੁਣਵਾਈ ਦੇ ਆਧਾਰ ਉੱਤੇ ਫਾਸਟ ਟ੍ਰੈਕ ਅਦਾਲਤਾਂ ਗਠਿਤ ਕਰਣ ਦੇ ਸੁਪਰੀਮ ਕੋਰਟ ਦੇ ਅਜੋਕੇ ਫੈਸਲੇ ਉੱਤੇ ਭਾਜਪਾ ਨੇ ਇਹ ਪ੍ਰਤੀਕਿਰਆ ਦਿੱਤੀ ਹੈ ।				
39.	ਮੌਸਮ ਵਿਭਾਗ ਨੇ ਅੰਡਮਾਨ ਦੇ ਸਾਗਰ ਵਿੱਚ ਮਾਨਸੂਨ ਦੀ ਸਾਲਾਨਾ ਮੀਂਹ ਥੋੜ੍ਹਾ ਦੇਰ ਤੋਂ ਹੋਣ ਦੀ ਸੰਦੇਹ ਜਤਾਈ ਹੈ ।				
40.	ਵਿੰਸ਼ਵ ਦੀ ਤਿੰਨ ਸਭਤੋਂ ਮੋਟੀ ਬਰਫੀਲੀ ਪਰਤਾਂ ਵਿੱਚੋਂ ਇੱਕ ਪੱਛਮ ਵਾਲਾ ਅੰਟਾਰਕਟੀਕਾ ਦੀ ਤਹ ਹੈ ।				
41.	ਫਿਲਹਾਲ ਡਰਨ ਦੀ ਜ਼ਰੂਰਤ ਨਹੀਂ				
42.	ਉਨ੍ਹਾਂ ਦੇ ਅਤੇ ਪਾਇਟ ਦੇ ਵਿਦਿਆਰਥੀਆਂ ਦੇ ਵਿੱਚ ਦਾ ਸੰਵਾਦ ਬੇਹੱਦ ਰੋਚਕ ਰਿਹਾ ।				
43.	ਇੰਦਰਾ ਗਾਂਧੀ ਰਾਸ਼ਟਰੀ ਅਜ਼ਾਦ ਯੂਨੀਵਰਸਿਟੀ [ਇਗਨੂ] ਦੇ ਵਿਦਿਆਰਥੀਆਂ ਨੂੰ ਉਨ੍ਹਾਂ ਦੀ ਮੰਗ ਪਰ ਘਰ ਬੈਠੇ ਪਰੀਖਿਆ ਦੇਣ ਦੀ ਸੁਵਿਧਾ ਮਿਲਣ ਜਾ ਰਹੀ ਹੈ ।				
44.	ਧੋਨੀ ਨੇ ਕਪਤਾਨੀ ਪਾਰੀ ਖੇਡੀ				
45.	ਇਸਦੇ ਬਾਅਦ ਸਚਿਨ ਤੇਂਦੁਲਕਰ ਟੀਮ ਦੀ ਬੇੜੀ ਪਾਰ ਲਗਾਉਣ ਲਈ ਮੈਦਾਨ ਪਰ ਆਏ ਲੇਕਿਨ ਉਹ ਜਿਆਦਾ ਰਣ ਨਹੀਂ ਬਣਾ ਸਕੇ ।				

46.	ਰਾਜਸਥਾਨ ਦੀ ਸ਼ੁਰੁਆਤ ਬੇਹੱਦ ਖ਼ਰਾਬ ਰਹੀ ਅਤੇ ਇੱਕ ਵਾਰ ਦਬਾਅ ਵਿੱਚ ਆਉਣ ਦੇ ਬਾਅਦ ਉਸਦੇ ਸਾਰੇ ਬੱਲੇਬਾਜ ਆਪਣਾ ਵਿਕੇਟ ਸੁੱਟਕੇ ਚਲਦੇ ਬਣੇ ।				
47.	ਸੰਸਾਰ ਬੈਡਮਿੰਟਨ ਚੈਂਪਿਅਨਸ਼ਿਪ ਨੂੰ ਤਿਆਰ ਭਾਰਤ May 08 , 06 : 45 pm				
48.	ਮੋਹਨ ਬਾਗਾਨ ਹੁਣੇ ਤੱਕ ਖਾਤਾ ਨਹੀਂ ਖੋਲ ਪਾਇਆ ਹੈ ਹੋਰ ਗਰੁਪ ਵਿੱਚ ਸਭਤੋਂ ਹੇਠਲੇ ਸਥਾਨ ਪਰ ਹੈ ।				
49.	ਦੋ ਗੋਲ ਨੂੰ ਵਾਧੇ ਬਣਾਉਣ ਦੇ ਬਾਵਜੂਦ ਚੀਨ ਦੇ ਨਾਲ 2 - 2 ਤੋਂ ਡਰਾ ਖੇਡਣ ਦੇ ਬਾਅਦ ਭਾਰਤ ਮੰਗਲਵਾਰ ਨੂੰ ਏਸ਼ਿਆ ਕਪ ਹਾਕੀ ਵਿੱਚ ਸੇਮੀਫਾਇਨਲ ਦੀ ਦੋੜ ਤੋਂ ਬਾਹਰ ਹੋ ਗਿਆ ਜਿਸਦੇ ਨਾਲ ਖਿਤਾਬ ਬਰਕਰਾਰ ਰੱਖਣ ਦਾ ਉਸਦਾ ਸੁੱਪਣਾ ਵੀ ਚੂਰ ਚੂਰ ਹੋ ਗਿਆ ।				
50.	ਸਾਰਵਜਨਿਕ ਖੇਤਰ ਦੇ ਆਈਡੀਬੀਆਈ ਬੈਂਕ ਨੇ ਵੀ ਏਫਡੀ ਉੱਤੇ ਵਿਆਜ ਦਰਾਂ ਅੱਧਾ ਤੋਂ ਇੱਕ ਫੀਸਦੀ ਤੱਕ ਘਟਾ ਦਿੱਤੀਆਂ ਹਨ । ਨਵੀਂ ਦਰਾਂ 21 ਮਈ ਤੋਂ ਲਾਗੂ ਹੋਵੇਗੀ ।				

Intelligibility Test - Literature

S.No.	Sentence	0	1	2	3
1.	ਜੇਕਰ ਤੈਨੂੰ ਉਹ ਚੀਜ ਨਹੀਂ ਮਿਲੇ ਤਾਂ ਖਬਰਦਾਰ ਏਧਰ ਰੁੱਖ ਨਹੀਂ ਕਰਣਾ , ਵਰਨਾ ਸੂਲੀ ਪਰ ਖਿੱਚਵਾ ਦੂੰਗੀ				
2.	ਇਸਦੀ ਸੂਚਨਾ ਨੇ ਅਗਿਆਨ ਬਲਿਕਾ ਨੂੰ ਮੁੰਹ ਢਾਂਪ ਕਰ ਇੱਕ ਕੋਨੇ ਵਿੱਚ ਬਿਠਾ ਰੱਖਿਆ ਹੈ ।				
3.	ਸ਼ਾਮ ਦਾ ਸਮਾਂ ਸੀ , ਨਿਰਮਲਾ ਛੱਤ ਪਰ ਜਾਨਕੇ ਇਕੱਲੀ ਬੈਠੀ ਅਕਾਸ਼ ਕੀਤੀ ਹੋਰ ਤ੍ਰਿਸ਼ਿਤ ਨੇਤਰਾਂ ਤੋਂ ਵੇਖ ਰਹੀ ਸੀ ।				
4.	ਨਿਰਮਲਾ - ਨੇ ਉਦਾਸੀਨ ਭਾਵ ਨੂੰ ਕਿਹਾ - ਤੂੰ ਜਾ , ਮੈਂ ਨਹੀਂ ਜਾਵਾਂਗੀ ।				
5.	ਬਾਗ ਵਿੱਚ ਫੁਲ ਖਿੜੇ ਹੋਏ ਸਨ । ਮਿੱਠੀ - ਮਿੱਠੀ ਸੁਗੰਧ ਆ ਰਹੀ ਸੀ । ਚੇਤ ਦੀ ਸੀਤਲ ਮੰਦ ਸਮੀਰ ਚੱਲ ਰਹੀ ਸੀ ।				
6.	ਇਹ ਕਹਿਕੇ ਕਲਿਆਣੀ ਕਮਰੇ ਦੇ ਬਾਹਰ ਨਿਕਲ ਗਈ ।				
7.	ਮੁੰਸ਼ੀਜੀ ਤਾਂ ਭੋਜਨ ਕਰਣ ਗਏ ਅਤੇ ਨਿਰਮਲਾ ਦਵਾਰ ਦੀ ਚੌਖਟ ਪਰ ਖੜੀ ਸੋਚ ਰਹੀ ਸੀ - ਭਗਵਾਨ ।				
8.	ਸਾਧੁ - ਕਦੇ ਆ ਜਾਵਾਂਗਾ ਬੱਚਾ , ਤੁਹਾਡਾ ਘਰ ਕਿੱਥੇ ਹੈ ?				
9.	ਇੱਕ ਦਿਨ ਨਿਰਮਲਾ ਨੇ ਸਿਆਰਾਮ ਨੂੰ ਘੀ ਲਿਆਉਣ ਲਈ ਬਾਜ਼ਾਰ ਭੇਜਿਆ ।				
10.	ਮਾਤਾ - ਝੂਠ ਨੇ ਬੋਲ ! ਤੂੰ ਪੰਜ ਸੌ ਰੁਪਏ ਦੇ ਨੋਟ ਨਹੀਂ ਭੇਜੇ ਸਨ ?				
11.	ਕੀ ਮੇਰੀ ਹਾਲਤ ਨੂੰ ਹੋਰ ਵੀ ਦਾਰੁਣ ਬਣਾਉਣਾ ਚਾਹੁੰਦੇ ਹੋ ?				
12.	ਉਸ ਦਿਨ ਤੋਂ ਨਿਰਮਲਾ ਦਾ ਰੰਗ - ਢੰਗ ਬਦਲਨ ਲਗਾ ।				
13.	ਕੀ ਇਨ੍ਹਾਂ ਨੂੰ ਸਚਮੁੱਚ ਕੋਈ ਭੀਸ਼ਨ ਰੋਗ ਹੋ ਰਿਹਾ ਹੈ ?				
14.	ਨਿਰਮਲਾ - ਤਾਂ ਮੈਂ ਝੂਠ ਕਹਿੰਦੀ ਹਾਂ ?				
15.	ਬੇਚਾਰੇ ਮੁੰਡੇ ਨੂੰ ਵਾਰ - ਵਾਰ ਦੌੜਾਇਆ ਕਰਦੀ ਹੈ । ਮਤ੍ਰੇਈ ਮਾਂ ਹੈ ਨਹੀਂ ! ਆਪਣੀ ਮਾਂ ਹੋ ਤਾਂ ਕੁੱਝ ਖਿਆਲ ਵੀ ਕਰੇ ।				
16.	ਦਾਨਨਾਥ ਨੂੰ ਅਜਿਹੀ ਉੱਤਮ ਸਪੀਚ ਨੂੰ ਨਹੀਂ ਸੁਣਨ ਨੂੰ ਦਾ ਅਤਿਅੰਤ ਸੋਗ ਹੋਇਆ । ਬੋਲੇ—ਯਾਰ , ਮੈਂ ਜੰਮ ਦਾ ਅਭਾਗਾ ਹਾਂ । ਕੀ ਹੁਣ ਫਿਰ ਕੋਈ ਵਿਖਿਆਨ ਨਹੀਂ ਹੋਵੇਗਾ ?				

17.	ਦਾਨਨਾਥ ਤਾਂ ਇਹ ਗੱਲਬਾਤ ਕਰਕੇ ਆਪਣੇ ਮਕਾਨ ਨੂੰ ਰਵਾਨਾ ਹੋਏ ਅਤੇ ਅਮ੍ਰਿਤ ਰਾਇ ਉਸੀ ਹਨ੍ਹੇਰੇ ਵਿੱਚ , ਵੱਡੀ ਦੇਰ ਤੱਕ ਚੁਪਚਾਪ ਖੜੇ ਰਹੇ ।				
18.	ਅੱਜ ਵੀ , ਜਦੋਂ ਅਮ੍ਰਿਤ ਰਾਇ ਨੇ ਉਸਤੋਂ ਆਪਣੇ ਇਰਾਦੇ ਸਾਫ਼ ਕੀਤੇ ਤੱਦ ਉਸੇਨ ਸੱਚੇ ਦਿਲੋਂ ਉਨ੍ਹਾਂਨੂੰ ਸੱਮਝਾਕੇ ਉੂਂਚ ਨੀਚ ਸੁਝਾਇਆ ।				
19.	ਲਾਲਾ ਬਦਰੀਪ੍ਰਸਾਦ ਅਮ੍ਰਿਤ ਰਾਇ ਦੇ ਬਾਪ ਦੇ ਦੋਸਤਾਂ ਵਿੱਚ ਸਨ ਅਤੇ ਜੇਕਰ ਉਨ੍ਹਾਂ ਨੂੰ ਜਿਆਦਾ ਇੱਜ਼ਤ ਵਾਲਾ ਨਹੀਂ ਸਨ ਤਾਂ ਬਹੁਤ ਹੇਠੇ ਵੀ ਨਹੀਂ ਸਨ । ਦੋਨੋਂ ਵਿੱਚ ਮੁੰਡੇ - ਕੁੜੀ ਦੇ ਵਿਆਹ ਦੀ ਗੱਲਬਾਤ ਪੱਕੀ ਹੋ ਗਈ ਸੀ ।				
20.	ਦਸਮੀਂ ਨੇ ਆਉਂਦੇ ਹੀ ਸਭ ਸਤਰੀਆਂ ਨੂੰ ਉੱਥੇ ਨੂੰ ਹਟਾ ਦਿੱਤਾ , ਪ੍ਰੇਮਾ ਨੂੰ ਇਤਰ ਸੁਘਾਇਆ ਕੇਵਡੇ ਅਤੇ ਗੁਲਾਬ ਦਾ ਛੀਂਟਾ ਮੂੰਹ ਪਰ ਮਾਰਿਆ । ਹੌਲੀ - ਹੌਲੀ ਉਸਦੇ ਤਲਵੇ ਸਹਲਾਏ , ਸਭ ਖਿੜਕੀਆਂ ਖੁੱਲ੍ਹਵਾ ਦਿੱਤੀ ।				
21.	ਮਿਸਟਰ ਸ਼ਰਮਾ— (ਮੂੰਛੇ ਪਰ ਹੱਥ ਫੇਰਕੇ) ਉਹ ਤਾਜ਼ਾ ਖਬਰ ਲਿਆਇਆ ਹਾਂ ਕਿ ਤੁਸੀ ਲੋਕ ਸੁਣਕੇ ਫੜਫੜਾਹਟ ਜਾਇੰਗੇ ।				
22.	ਖੁਲਾਸਾ ਇਹ ਕਿ ਅਮ੍ਰਤਰਾਏ ਨੂੰ ਇੱਥੇ ਤੋਂ ਸੱਤਰਹ ਹਜਾਰ ਰੂਪਆ ਮਿਲਿਆ । ਮੁਨਸ਼ੀ ਬਦਰੀਪ੍ਰਸਾਦ ਨੇ ਇਕੱਲੇ ਬਾਰਾਂ ਹਜਾਰ ਦਿੱਤਾ ਜੋ ਉਨ੍ਹਾਂ ਦੀ ਉਂਮੇਦ ਤੋਂ ਬਹੁਤ ਜ਼ਿਆਦਾ ਸੀ ।				
23.	ਰਾਮ— (ਮੁਸਕਰਾਕੇ) ਚੁਪ । ਅਜਿਹਾ ਵੀ ਕੋਈ ਕਹਿੰਦਾ ਹੈ ।				
24.	ਆਪਣੇ ਦਿਲ ਦਾ ਜਾਣ ਪਹਿਚਾਣ ਉਹਨੂੰ ਇੱਕ ਦਿਨ ਇੰਜ ਮਿਲਿਆ ਕਿ ਬਾਬੂ ਅਮ੍ਰਿਤ ਰਾਇ ਨਿਅਤ ਸਮਾਂ ਪਰ ਨਹੀਂ ਆਏ । ਥੋੜ੍ਹੀ ਦੇਰ ਤੱਕ ਤਾਂ ਉਹ ਉਨ੍ਹਾਂ ਦੀ ਰੱਸਤਾ ਵੇਖਦੀ ਰਹੀ ਮਗਰ ਜਦੋਂ ਉਹ ਹੁਣ ਵੀ ਨਹੀਂ ਆਏ ਤੱਦ ਤਾਂ ਉਸਦਾ ਦਿਲ ਕੁੱਝ ਮਸੋਸਨੇ ਲਗਾ । ਵੱਡੀ ਵਿਆਕੁਲਤਾ ਤੋਂ ਦੌੜੀ ਹੋਈ ਦੀਵਾਜੇ ਪਰ ਆਈ ਅਤੇ ਅੱਧ ਘੰਟੇ ਤੱਕ ਕੰਨ ਲਗਾਏ ਖੜੀ ਰਹੀ , ਫਿਰ ਅੰਦਰ ਆਈ ਅਤੇ ਮਨ ਮਾਰਕੇ ਬੈਠ ਗਈ ।				
25.	ਅਮ੍ਰਤਰਾਏ— (ਦੱਬੀ ਜਬਾਨ ਤੋਂ) ਉਹ ਸਭ ਕਹਾਰ ਮੇਰੇ ਨੌਕਰ ਹੈ ।				

26.	ਅਮ੍ਰਤ०—ਵੇਖੇ ਹੁਣ ਕਦੋਂ ਕਿਸਮਤ ਜਾਗਦਾ ਹੈ । ਮੈਂ ਤਾਂ ਬਹੁਤ ਜਲਦੀ ਮਚਾ ਰਿਹਾ ਹਾਂ ।				
27.	ਮੈ ਤੁਹਾਡੇ ਤੋਂ ਕੋਈ ਅਣ-ਉਚਿਤ ਗੱਲ ਨਹੀਂ ਚਾਹੁੰਦਾ ।				
28.	ਉਨ੍ਹਾਂ ਦੇ ਜਰਾ ਤੋਂ ਇਸ਼ਾਰੇ ਪਰ ਮੈਂ ਆਪਣੇ ਨੂੰ ਨਿਛਾਵਰ ਕਰ ਸਕਦੀ ਹਾਂ ।				
29.	ਬਿੱਲਾਂ—ਦੇ ਕਿਉਂ ਨਹੀਂ ਗਿਆ ।				
30.	ਕੁੱਝ ਦਿਨਾਂ ਤੋਂ ਪੰਡਾਇਨ ਐਰਚੈਬਾਇਨ ਆਦਿ ਨੇ ਵੀ ਦਸਮੀਂ ਦੇ ਰਚਨਾ - ਚੋਣ ਪਰ ਨੱਕ - ਭਰਵੱਟਾ ਚੜ੍ਹਾਨਾ ਛੱਡ ਦਿੱਤਾ ਸੀ ।				
31.	ਉਸਨੇ ਆਉਂਦੇ ਹੀ ਹੁਕਮ ਦਿੱਤਾ ਕਿ ਭੀੜ ਹਟਾ ਦਿੱਤੀ ਜਾਵੇ ।				
32.	ਵਿਆਹ ਦੇ ਚੌਥੇ ਦਿਨ ਬਾਅਦ ਦਸਮੀਂ ਬੈਠੀ ਹੋਈ ਸੀ ਕਿ ਇੱਕ ਔਰਤ ਨੇ ਆਕੇ ਉਸਦੇ ਇੱਕ ਬੰਦ ਲਿਫਾਫਾ ਦਿੱਤਾ ।				
33.	ਚੰਮਨ ਚੌਧਰੀ— ਕਹਿ ਗਏ ਹਨ ਕਿਆਜ ਇਨਕੇਰ ਕੰਮ ਨਹੀਂ ਛੱਡ ਦੇਹਾਂ ਤਾਂ ਟਾਟ ਬਾਹਰ ਕਰ ਦੀਨ ਜੈਹੀ ।				
34.	ਇੱਕ ਦਿਨ ਵ੍ਰਜਰਾਨੀ ਸੁਵਾਮਾ ਦੇ ਸਿਰਹਾਨੇ ਬੈਠੀ ਪੱਖਾ ਝਲ ਰਹੀ ਸੀ ।				
35.	ਪ੍ਰਤਾਪ - ਤਾਂ ਭਈ , ਇੱਕ ਦਿਨ ਮੈਨੂੰ ਵੀ ਨੇਵਤਾ ਦੋ ।				
36.	ਨਵੀਨ ਮਿੱਟੀ ਦੀ ਮਿੱਠੀ - ਮਿੱਠੀ ਸੁਗੰਧ ਆ ਰਹੀ ਹੈ ।				
37.	ਚਿੱਟਾ ਗੁਲਾਬ - ਮੈਂ ਤਾਂ ਬਿਨਾਂ ਗੀਤ ਸੁਣੇ ਅੱਜ ਤੁਹਾਡਾ ਪਿੱਛਾ ਨਹੀਂ ਛੋਡੂੰਗੀ ।				
38.	ਰਾਜਾ ਨੇ ਕਿਹਾ “ਚੰਗੀ ਗੱਲ ਹੈ । ”				
39.	ਤਿੰਨ ਦਿਨ ਗੁਜ਼ਰਨ ਪਰ ਬੁੱਢੀ ਫਿਰ ਉੱਥੇ ਪਹੁੰਚੀ ।				
40.	ਇਸਤਰੀ ਰੋਣ ਲੱਗੀ । ਇੱਕ ਮੁਸਾਫਰ ਉੱਧਰ ਜਾ ਰਿਹਾ ਸੀ ।				
41.	ਸੇਠ ਆਪਣੇ ਜਵਾਈ ਤੋਂ ਮਿਲਕੇ ਵੱਡੇ ਖੁਸ਼ ਹੋਏ ਅਤੇ ਉਨ੍ਹਾਂਨੇ ਉਸਨੂੰ ਵੱਡੀ ਚੰਗੀ ਤਰ੍ਹਾਂ ਤੋਂ ਘਰ ਵਿੱਚ ਰੱਖਿਆ ।				
42.	ਲੱਗਦਾ ਹੈ ਇਹ ਦਰਜ਼ੀ ਲੋਭੀ ਹੈ । ਇਹ ਸਾਨੂੰ ਖਵਾਉਣਾ ਨਹੀਂ ਚਾਹੁੰਦਾ , ਇਸਲਈ ਇਹ ਸਾਰਾ ਡਰਾਮਾ ਕਰ ਰਿਹਾ ਹੈ ।				
43.	ਰਾਜਾ ਮੇਰੇ ਤੋਂ ਡਰ ਗਿਆ ।				

44.	ਪਾਰਬਤੀ ਨੂੰ ਤਰਸ ਆ ਗਈ , ਅਤੇ ਉਨ੍ਹਾਂ ਨੇ ਸ਼ੰਕਰ ਤੋਂ ਪ੍ਰਾਰਥਨਾ ਦੀ ਕਿ ਜਿਵੇਂ ਵੀ ਬਣੇ , ਉਹ ਗਿਲਹਰੀ ਨੂੰ ਫਿਰ ਤੋਂ ਇਸਤਰੀ ਬਣਾ ਦਿਓ ।				
45.	ਰੂਸ ਵਿੱਚ ਇੱਕ ਬਹੁਤ ਵੱਡੇ ਲੇਖਕ ਹੋਏ ਹੈ , ਇਨ੍ਹੇ ਵੱਡੇ ਕਿ ਸਾਰੀ ਦੁਨੀਆ ਉਨ੍ਹਾਂਨੂੰ ਜਾਣਦੀ ਹੈ ।				
46.	ਰਵੀਂਦਰ ਠਾਕੁਰ ਦੀ ਇੱਕ ਵੱਡੀ ਹੀ ਸੀਖ ਦੇਣ ਵਾਲੀ ਰਚਨਾ ਹੈ ।				
47.	ਦੇਵਦੂਤ ਚਲਾ ਗਿਆ ਅਤੇ ਅਗਲੇ ਦਿਨ ਜਦੋਂ ਉਹ ਪਰਤਿਆ ਤਾਂ ਉਸਦੇ ਹੱਥ ਵਿੱਚ ਉਨ੍ਹਾਂ ਬੰਦੀਆਂ ਦੀ ਸੂਚੀ ਸੀ				
48.	ਆਦਮੀ ਨੂੰ ਭੂਮੀ ਤੋਂ ਕਿੰਨਾ ਮੋਹ ਹੁੰਦਾ ਹੈ ।				
49.	ਕਹਿਣ ਦਾ ਮਤਲੱਬ ਇਹ ਕਿ ਹਰ ਆਦਮੀ ਆਪਣੀ ਸਮਰੱਥਾ ਦੇ ਅਨੁਸਾਰ ਕੰਮ ਕਰੇ ਹੋਰ ਜ਼ਰੂਰਤ ਦੇ ਅਨੁਸਾਰ ਪਾਏ				
50.	ਅਸੀ ਆਸ ਕਰਦੇ ਹਾਂ ਕਿ ਪਾਠਕ ਇਸ ਕਿਤਾਬਾਂ ਨੂੰ ਵੱਡੇ ਚਾਵ ਨਾਲ ਪੜਾਂਗੇ , ਦੂਸਰੀਆਂ ਦੀ ਪੜਵਾਏ ਅਤੇ ਇਨ੍ਹਾਂ ਦਾ ਭਰਪੂਰ ਮੁਨਾਫ਼ਾ ਲੈਣਗੇ ।				

Intelligibility Test - Articles

S.No.	Sentence	0	1	2	3
1.	ਏਨੀਮਿਆ (Anemia) ਦੇ ਕਾਰਨ ਔਰਤਾਂ ਵਿੱਚ ਥਕਾਣ , ਉੱਠਣ ਬੈਠਣ ਹੋਰ ਖੜੇ ਹੋਣ ਵਿੱਚ ਚੱਕਰ ਆਣਾ , ਕੰਮ ਕਰਣ ਦਾ ਮਨ ਨਹੀਂ ਕਰਣਾ , ਸਰੀਰ ਵਿੱਚ ਤਾਪਮਾਨ ਦੀ ਕਮੀ , ਤਵਚਾ ਵਿੱਚ ਪਿਲੱਤਣ , ਦਿਲ ਵਿੱਚ ਗ਼ੈਰ - ਮਾਮੂਲੀ ਧੜਕਨ , ਸਾਂਸ ਲੈਣ ਵਿੱਚ ਤਕਲੀਫ , ਸੀਨੇ ਵਿੱਚ ਦਰਦ , ਤਲਵਾਂ ਅਤੇ ਹਥੇਲੀਆਂ ਵਿੱਚ ਠੰਡਾਪਨ ਹੋਰ ਲਗਾਤਾਰ ਰਹਿਣ ਵਾਲਾ ਸਿਰ ਵਿੱਚ ਦਰਦ ਹੁੰਦਾ ਹੈ ।				
2.	ਗਰਭਾਵਸਥਾ (pregnancy) ਦੇ ਦੌਰਾਨ ਢਿੱਡ ਵਿੱਚ ਤੀਵਰ ਦਰਦ ਹੋਰ ਯੋਨੀ ਤੋਂ ਰਕਤ ਸਰਾਵ ਹੋਣ ਲੱਗੇ ਤਾਂ ਇਸਨੂੰ ਗੰਭੀਰਤਾ ਤੋਂ ਲਵੋ ਅਤੇ ਡਾਕਟਰ ਨੂੰ ਤੱਤਕਾਲ ਦੱਸੀਏ ।				
3.	ਸਾਮਗਰੀ : 150 ਗਰਾਮ ਅਰਹਰ ਦਾਲ , 20 ਗਰਾਮ ਮੂੰਗਫਲੀ , 100 ਗਰਾਮ ਗੁਡ , 100 ਮਿ . ਲਈ . ਤੇਲ , 15 ਗਰਾਮ ਦਾਲਚੀਨੀ , 3 ਗਰਾਮ ਹੀਂਗ , 5 ਗਰਾਮ ਈਮਲੀ , 5 ਗਰਾਮ ਅਦਰਕ , 5 ਗਰਾਮ ਲੂਣ , 3 ਗਰਾਮ ਹਲਦੀ , 5 ਗਰਾਮ ਹਰੀਮੀਰਚ , 5 ਗਰਾਮ , 3 ਗਰਾਮ ਕਰੀਪੱਤਾ , 3 ਗਰਾਮ ਨਾਰੀਅਲ (ਕੱਸਿਆ ਹੋਇਆ) , ਧਨਿਆ ਬਰੀਕ ਕਟੀ ਹੁਇ ।				
4.	ਆਇਸਕਰੀਮ ਦੇ ਦਾਗ : - ਜੇਕਰ ਆਇਸਕਰੀਮ ਦੇ ਦਾਗ ਕਪਡੋਂ ਵਿੱਚ ਲੱਗ ਜਾਵੇ ਤਾਂ ਅਮੋਨਿਆ ਦਾ ਘੋਲ ਪਾਓ ।				
5.	ਚਾਵਲ ਦੀ ਖੀਰ ਬਣਾਉਂਦੇ ਸਮਾਂ ਸ਼ਕਰ ਦੇ ਨਾਲ ਥੋਡਾ ਜਿਹਾ ਲੂਣ ਮਿਲਾਉਣ ਤੋਂ ਖੀਰ ਦਾ ਸਵਾਦ ਹੋਰ ਬਢ ਜਾਂਦਾ ਹੈ ।				
6.	ਸੋਨੇ ਦੇ ਜੇਵਰ ਤੇ ਪਿਸੀ ਹਲਦੀ ਲੱਗਾ ਕੇ ਮਸਲਣ ਤੋਂ ਉਹ ਚਮਕਣ ਲੱਗਦੇ ਹੈ				
7.	ਕਦੇ ਵੀ ਇਸਦਾ ਸਪੀਨਰ ਖਾਲੀ ਨਹੀਂ ਚਲਣ ਦਿਓ ।				

8.	ਅੱਜਕੱਲ੍ਹ ਦੀਆਂ ਮਹਿਲਾਵਾਂ ਨੌਕਰੀਪੇਸ਼ਾ ਵਾਲੀਆਂ ਹਨ ਇਸਲਈ ਜਿਆਦਾ ਸਮਾਂ ਘਰ ਤੋਂ ਬਾਹਰ ਗੁਜ਼ਾਰਦੀਆਂ ਹਨ ਸੰਯੁਕਤ ਪਰਵਾਰਾਂ ਵਿੱਚ ਜਾਂ ਜਿਨ੍ਹਾਂ ਦੇ ਮਾਤੇ ਪਿਤਾ ਘਰ ਤੇ ਰਹਤੇ ਹੈ ਉਨ੍ਹਾਂ ਨੂੰ ਬਚਚਾਂ ਦੀ ਦੇਖਬਾਲ ਦੀ ਸਮੱਸਿਆ ਨਹੀ ਹੁੰਦੀ ਹੈ ਪਰ ਏਕਲ ਪਰਵਾਰਾਂ ਵਿੱਚ ਮਾਂ ਦੇ ਦਫਤਰ ਜਾਣ ਦੇ ਬਾਅਦ ਬਚਚੇ ਦੀ ਦੇਖਬਾਲ ਲਈ ਕੋਈ ਨਹੀ ਰਹਿੰਦਾ ਇਸਲਈ ਮਹਿਲਾਵਾਂ ਆਪਣੇ ਬਚਚਾਂ ਦੇ ਲਿਅ ਆਇਆ ਦਾ ਇੰਤਜਾਮ ਕਰਦੀ ਹੈ ।				
9.	ਆਪਣੇ ਜੀਵਨ ਉਦੇਸ਼ਿਅ ਨੂੰ ਜਾਨਣਾ ਅਤੇ ਉਸਨੂੰ ਪ੍ਰਾਪਤ ਕਰਣ ਲਈ ਢ੍ਰਢ ਆਤਮਵੀਸ਼ਵਾਸ ਰੱਖਣਾ , ਇਹੀ ਸਫਲਤਾ ਦੇ ਵੱਲ ਪਹਿਲਾ ਕਦਮ ਹੈ				
10.	ਭਗਤ ਦਾ ਭਗਵਾਨ ਤੋਂ , ਮਨੁੱਖ ਦਾ ਰੱਬ ਤੋਂ , ਸਫਲਤਾ ਦਾ ਸਾਰੇ ਤੋਂ , ਪਿੰਡ ਦਾ ਬਰਹਮੰਡ ਤੋਂ ਮਿਲਣ ਨੂੰ ਹੀ ਯੋਗ ਕਿਹਾ ਗਿਆ ਹੈ				
11.	ਮਿਜੋਰਮ ਦੇ ਇੱਕ 64 ਸਾਲ ਦਾ ਵਿਅਕਤੀ ਜਯੋਨ ਦੀ 50 ਪਤਨੀਆਂ ਅਤੇ 100 ਬੱਚੇ ਹਨ । ਮਿਜੋਰਮ ਤੋਂ ਲੱਗਭੱਗ 80 ਕਿਮੀ ਦੂਰ ਬਕਤਵਾਂਗ ਪਿੰਡ ਦਾ ਨਿਵਾਸੀ ਜਯੋਨ ਆਪਣੇ ਪਰਵਾਰ ਦੇ 180 ਤੋਂ ਜਿਆਦਾ ਸਦਸਯੋਂ ਦੇ ਨਾਲ ਧਰਤੀ ਤੇ ਸਭਤੋਂ ਬਡੇ ਪਰਵਾਰ ਦੇ ਮੁਖੀ ਦੇ ਰੁਪ ਵਿੱਚ ਜਾਣਿਆ ਜਾਂਦਾ ਹੈ ।				
12.	ਇੱਕ ਛੋਟਾ ਬਚਚਾ ਦੂਜੇ ਬਚਚੇ ਤੋਂ , ਜੇਕਰ ਦਿਨ ਨੂੰ ਸੂਰਜ ਨਹੀਂ ਨਿਕਲਿਆ ਤਾਂ ਕੀ ਹੋਵੇਗਾ ,ਦੂਜੇ ਬਚਚੇ ਨੇ ਜਵਾਬ ਦਿੱਤਾ , ਬਿਜਲੀ ਦਾ ਬਿਲ ਬਢ ਜਾਵੇਗਾ ।				
13.	ਬੀਰਬਲ ਨੂੰ ਤੰਬਾਕੂ ਖਾਣ ਦੀ ਆਦਤ ਸੀ ਲੇਕਿਨ ਅਕਬਰ (Akbar) ਨਹੀਂ ਖਾਂਦੇ ਸਨ ਇੱਕ ਦਿਨ ਅਕਬਰ ਨੇ ਤੰਬਾਕੂ ਦੇ ਖੇਤ ਵਿੱਚ ਗਧੇ ਨੂੰ ਘਾਹ ਖਾਂਦੇ ਵੇਖਕੇ ਕਿਹਾ ਬੀਰਬਲ ਇਹ ਵੇਖਿਆ ਤੰਬਾਕੂ ਕਿਵੇਂ ਦੀ ਬੁਰੀ ਚੀਜ ਹੈ , ਗਧੇ ਤੱਕ ਇਸ ਨੂੰ ਨਹੀ ਖਾਂਦੇ ।				

14.	ਇਹ ਸਭ ਕਹਿਣ ਦੀਆਂ ਗੱਲਾਂ ਹਨ ਕਿ ਉਨ੍ਹਾਂ ਨੂੰ ਛੋਡ ਬੈਠੇ ਹਨ ਜਦੋਂ ਅੱਖਾਂ ਚਾਰ ਹੁੰਦੀਆਂ ਹਨ ਮੋਹਬਤ ਆ ਹੀ ਜਾਂਦੀ ਹੈ ।				
15.	ਨਵੇਂ ਅਤੇ ਆਧੁਨਿਕ ਡਿਜ਼ਾਇਨਾਂ ਦੇ ਅਤਪਾਦ ਤੇਜ਼ੀ ਤੋਂ ਬਾਜ਼ਾਰ ਵਿੱਚ ਆ ਰਹੇ ਹੈ । ਇਸ ਲਈ ਜ਼ਰੁਰੀ ਹੈ ਕਿ ਆਪ ਵੀ ਆਪਣੇ ਨੀਤੀਆਂ ਵਿੱਚ ਬਦਲਾਵ ਲਿਆਓ ਅਤੇ ਇਹੀ ਨਹੀਂ ਕਰਦੇ ਰਹੇ , ਅਸੀ ਤਾਂ ਇਸ ਕੰਮ ਨੂੰ ਇਸ ਤਰੀਕੇ ਤੋਂ ਕਰਦੇ ਆ ਰਹੇ ਹੈ ਅਤੇ ਅਜਿਹਾ ਹੀ ਕਰੇਗੇ ।				

Intelligibility Test – Official Language Qoutes

S.No.	Sentence	1	2	3	4
1.	ਸੰਖਿਪਤ ਨੋਟ ਹੇਠਾਂ ਦਿੱਤਾ ਗਿਆ ਹੈ				
2.	ਪ੍ਰਚੱਲਤ ਨਿਯਮਾਂ ਦੇ ਅਨੁਸਾਰ				
3.	ਰਸੀਦ ਪਹਿਲਾਂ ਹੀ ਭੇਜੀ ਜਾ ਚੁੱਕੀ ਹੈ				
4.	ਉੱਤੇ ਕਕੇ ਅਨੁਸਾਰ ਕਾਰਵਾਈ ਕੀਤੀ ਜਾਵੇ				
5.	ਮਾਮਲੇ ਵਿੱਚ ਕਾਰਵਾਈ ਕੀਤੀ ਜਾ ਚੁੱਕੀ ਹੈ				
6.	ਪੱਤਰ ਦੀ ਇੱਕ ਨਕਲ				
7.	ਪਹਿਲਾਂ ਤੋਂ ਪ੍ਰਬੰਧ ਕਰਣਾ ਜਰੂਰੀ ਹੈ				
8.	ਅੱਗੇ ਦੀ ਤਰੱਕੀ ਤੋਂ ਜਾਣੂ ਕਰਾਓ				
9.	ਕਾਰਜ - ਸੂਚੀ ਨਾਲ ਭੇਜੀ ਜਾ ਰਹੀ ਹੈ				
10.	ਅਪੀਲ ਖਾਰਿਜ ਕਰ ਦਿੱਤੀ ਗਈ ਹੈ				
11.	ਜਿੱਥੇ ਤੱਕ ਸੰਭਵ ਹੋ				
12.	ਬਿਲ ਠੀਕ - ਠੀਕ ਬਣਾਇਆ ਗਿਆ ਹੈ				
13.	ਬਜਟ ਵਿੱਚ ਵਿਵਸਥਾ ਹੈ				
14.	ਸਵੀਕਾਰ ਨਹੀਂ ਕੀਤਾ ਜਾ ਸਕਦਾ				
15.	ਮਾਮਲੇ ਦੀ ਜਾਂਚ ਚੱਲ ਰਹੀ ਹੈ				

Intelligibility Test - Blogs

S.No.	Sentence	0	1	2	3
1.	ਮੈਂ ਅਜਿਹਾ ਇਸਲਈ ਲਿਖ ਰਿਹਾ ਹਾਂ ਕਿ ਮੇਰਾ ਇੱਕ ਦੋਸਤ ਜੋ ਮੇਰਾ ਬਲਾਗ ਪੜ੍ਹਦਾ ਹੈ ਉਸਨੇ ਮੈਨੂੰ ਕਿਹਾ ਕਿ ਇਹ ਨਾਰਦ ਦਾ ਏਕਾਧਿਕਾਰ ਖਤਮ ਹੋਵੇਗਾ !				
2.	ਰਾਜੇਸ਼ , ਹਿੰਦੀ ਚਿੱਠੀਆਂ ਦੇ ਜਿਆਦਾਤਰ ਪਾਠਕ ਚਿੱਠੇ ਲਿਖਣ ਵਾਲੇ ਹੀ ਹਨ , ਤੁਹਾਡੇ ਮਿੱਤਰ ਦੀ ਸ਼੍ਰੇਣੀ ਦੇ ਪਾਠਕ ਫਿਲਹਾਲ ਘੱਟ ਹਨ । ਹੁਣੇ ਤੱਕ ਹਿੰਦੀ ਦੇ ਗੈਰ ਚਿੱਠੇਕਾਰ ਪਾਠਕ ਤੁਹਾਡਾ ਅਖਬਾਰ ਹੀ ਪੜ੍ਹਦੇ ਹਨ । ਵੈਸੇ ਪਾਠਕ ਕੋਈ ਵੀ ਹੋਣ , ਕਿੰਨੇ ਹੀ ਕਿਉਂ ਨਹੀਂ ਹੋਣ ਕਿਸੇ ਬਹਿਸ ਅੰਜਾਮ ਤੋਂ ਪਹਿਲਾਂ ਛੱਡਣਾ ਹੋਰ ਮਾਧਿਅਮ ਵੀ ਨਹੀਂ ਚਾਹੁੰਦੇ ।				
3.	ਕੁੱਝ ਲੋਕਾਂ ਨੂੰ ਮਜਾ ਆਉਂਦਾ ਉਸੀ ਵਿਸ਼ਾ ਨੂੰ ਵਾਰ - ਵਾਰ ਫ਼ੇਂਟਨੇ ਵਿੱਚ ਤਾਂ ਤੁਸੀ ਕੀ ਕਰ ਸੱਕਦੇ ਹਨ ! ਪਰਕਾਸ਼ਨ ਦੀ ਅਜਾਦੀ ਹੈ ।				
4.	ਜਿੱਥੇ ਤੱਕ ਮੈਂ ਸੱਮਝਦਾ ਹਾਂ , ਹਿੰਦੀ ਦੇ ਚਿੱਠੇ ਹੁਣੇ ਬਹੁਤ ਹੀ ਸੀਮਿਤ ਮਜ਼ਮੂਨਾਂ ਪਰ ਲਿਖੇ ਜਾ ਰਹੇ ਹੋ । ਅਜਿਹੇ ਵਿਸ਼ਾ , ਜਿਨ੍ਹਾਂ ਵਿੱਚ ਸਾਰਾ ਨੇਟਪ੍ਰਯੋਕਤਾਵਾਂਦੀ ਕੋਈ ਦਿਲਚਸਪੀ ਨਹੀਂ ਹੈ । ਜਦੋਂ ਤੱਕ ਇਹ ਹਾਲ ਰਹੇਗਾ , ਸ਼ਾਇਦ ਹੀ ਹਿੰਦੀ ਚਿੱਠੀਆਂ ਦਾ ਪਾਠਕਵਰਗ ਵਿਕਸਿਤ ਹੋ ਸਕੇ ।				
5.	ਇਹ ਤਾਂ ਸੱਚ ਹੈ , ਹਿੰਦੀ ਦੀ ਕਈ ਪੋਸਟਾਂ ਨੂੰ ਬਿਨਾਂ ਬੈਕਗਰਾਉਂਡ ਜਾਣ ਕੋਈ ਸੱਮਝ ਨਹੀਂ ਸਕਦਾ ।				
6.	ਆਪ ਦੁਆਰਾ ਚੁੱਕਿਆ ਗਿਆ ਪ੍ਰਸ਼ਨ ਬਹੁਤ ਮਹੱਤਵ ਰੱਖਦਾ ਹੈ ।				
7.	ਬਹੁਤ ਠੀਕ ਲਿਖਿਆ ਹੈ ਤੁਸੀਂ , ਸਾਧੁਵਾਦ ! !				
8.	ਹੁਣ ਮੈਂ ਤੁਹਾਨੂੰ ਇੱਕ ਬੇਨਤੀ ਕਰਣਾ ਚਾਹਵਾਂਗਾ ।				
9.	ਜੇਕਰ ਹਿੰਦੀ ਅਤੇ ਹੋਰ ਭਾਰਤੀਭਾਸ਼ਾਵਾਂਲਈ ਇੱਕ ਵਧੀਆ ਟੇਕਸਟ ਏਨਾਲਿਸਿਸ ਦਾ ਔਜਾਰ (ਸਾਫਟਵੇਇਰ) ਬਣਾ ਸਕਣ ਤਾਂ ਭਾਰਤੀਭਾਸ਼ਾਵਾਂ ਦਾ ਬਹੁਤ ਭਲਾ ਹੋ				
10.	ਅਪੀਲ ਖਾਰਿਜ ਕਰ ਦਿੱਤੀ ਗਈ ਹੈ				
11.	ਜਿੱਥੇ ਤੱਕ ਸੰਭਵ ਹੋ				

12.	ਬਿਲ ਠੀਕ - ਠੀਕ ਬਣਾਇਆ ਗਿਆ ਹੈ				
13.	ਕ੍ਰਿਪਾ ਇਸ ਬਾਰੇ ਵਿੱਚ ਗੰਭੀਰਤਾ ਤੋਂ ਵਿਚਾਰ ਕਰੋ ।				
14.	ਤੁਸੀ ਜਿਵੇਂ ਲੋਕੋ ਦਾ ਸਾਡੀ ਭਾਸ਼ਾ ਦੇ ਪ੍ਰਤੀ ਪਿਆਰ ਹਿ ਸਾਰਿਆ ਨੂੰ ਉਤਸ਼ਾਹਿਤ ਰੱਖਦਾ ਹੈ .				
15.	ਬਹੁਤ ਸੁੰਦਰ ਕੋਸ਼ਿਸ਼ ਲਈ ਵਧਾਈ . ਇੱਕ ਬਲਾਗ ਪੋਸਟ ਇਸ ਪਰ ਕੱਲ ਲਿਖਦਾ ਹਾਂ .				

Appendix C

Test Data Set for Accuracy Test

Accuracy Evaluation:

The evaluators are provided with source text along with translated text. A highly intelligible output sentence need not be a correct translation of the source sentence. It is important to check whether the meaning of the source language sentence is preserved in the translation. This property is called accuracy.

Scoring:

The scoring is done based on the degree of intelligibility and comprehensibility. A Four point scale is made in which highest point is assigned to those sentences that look perfectly alike the target language and lowest point is assigned to the sentence which is un-understandable and unacceptable. The scale looks like:

Score 3 : Completely Faithful

Score 2: Fairly faithful: more than 50 % of the original information passes in the translation.

Score 1: Barely faithful: less than 50 % of the original information passes in the translation.

Score 0: Completely Unfaithful. Does not make sense.

Accuracy Test - News

S.No.	Hindi Sentence	Punjabi Sentence	0	1	2	3
1.	मुंबई। रिजर्व बैंक ने शुक्रवार को कहा कि वैश्विक आर्थिक संकट के प्रभाव से देश की अर्थव्यवस्था के उबरने के बाद वह मुद्रास्फीति के अनुमानों और मध्यम काल में इसके परिणामों के प्रबंधन पर ध्यान देगा।	ਮੁੰਬਈ । ਰਿਜਰਵ ਬੈਂਕ ਨੇ ਸ਼ੁੱਕਰਵਾਰ ਨੂੰ ਕਿਹਾ ਕਿ ਸੰਸਾਰਿਕ ਆਰਥਕ ਸੰਕਟ ਦੇ ਪ੍ਰਭਾਵ ਨੂੰ ਦੇਸ਼ ਦੀ ਮਾਲੀ ਹਾਲਤ ਦੇ ਉੱਬਰਣ ਦੇ ਬਾਅਦ ਉਹ ਮੁਦਰਾਸਫੀਤੀ ਦੇ ਅੰਦਾਜੀਆਂ ਅਤੇ ਮੱਧ ਕਾਲ ਵਿੱਚ ਇਸਦੇ ਨਤੀਜੀਆਂ ਦੇ ਪਰਬੰਧਨ ਉੱਤੇ ਧਿਆਨ ਦੇਵੇਗਾ ।				
2.	वाशिंगटन। अमेरिकी बैंकों के स्ट्रेस टेस्ट का नतीजा आखिरकार आ ही गया।	ਵਾਸ਼ਿੰਗਟਨ । ਅਮਰੀਕੀ ਬੈਂਕਾਂ ਦੇ ਸਟਰੇਸ ਟੇਸਟ ਦਾ ਨਤੀਜਾ ਆਖ਼ਿਰਕਾਰ ਆ ਹੀ ਗਿਆ ।				
3.	जिन बैंकों को पूंजी की आवश्यकता है, उन्हें योजना बनाने के लिए 8 जून तक का समय दिया गया है।	ਜਿਨ੍ਹਾਂ ਬੈਂਕਾਂ ਨੂੰ ਪੂੰਜੀ ਦੀ ਲੋੜ ਹੈ , ਉਨ੍ਹਾਂਨੂੰ ਯੋਜਨਾ ਬਣਾਉਣ ਲਈ 8 ਜੂਨ ਤੱਕ ਦਾ ਸਮਾਂ ਦਿੱਤਾ ਗਿਆ ਹੈ ।				
4.	बैंकों को यह योजना अपने नियामकों से मंजूर करानी होगी।	ਬੈਂਕਾਂ ਨੂੰ ਇਹ ਯੋਜਨਾ ਆਪਣੇ ਨਿਆਮਕਾਂ ਤੋਂ ਮਨਜ਼ੂਰ ਕਰਵਾਣੀ ਹੋਵੇਗੀ ।				
5.	अधिकारियों का कहना है कि आर्थिक स्थिति में सुधार के लिए मजबूत बैंकिंग तंत्र जरूरी है।	ਅਧਿਕਾਰੀਆਂ ਦਾ ਕਹਿਣਾ ਹੈ ਕਿ ਆਰਥਕ ਹਾਲਤ ਵਿੱਚ ਸੁਧਾਰ ਲਈ ਮਜਬੂਤ ਬੈਂਕਿੰਗ ਤੰਤਰ ਜਰੂਰੀ ਹੈ ।				

6.	इस टेस्ट से निवेशकों में यह भरोसा लौटेगा कि सारे बैंक कमजोर नहीं हैं।	ਇਸ ਟੇਸਟ ਤੋਂ ਨਿਵੇਸ਼ਕਾਂ ਵਿੱਚ ਇਹ ਭਰੋਸਾ ਪਰਤੇਗਾ ਕਿ ਸਾਰੇ ਬੈਂਕ ਕਮਜੋਰ ਨਹੀਂ ਹਨ ।				
7.	साथ ही कमजोर बैंकों में भी सुधार किया जा सकता है।	ਨਾਲ ਹੀ ਕਮਜੋਰ ਬੈਂਕਾਂ ਵਿੱਚ ਵੀ ਸੁਧਾਰ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ ।				
8.	विक्रम पंडित की अगुवाई वाली सिटी ने कहा कि वह 5.5 अरब डालर अतिरिक्त पूंजी जुटाने के लिए पब्लिक एक्सचेंज आफर का दायरा बढ़ाएगी।	ਵਿਕਰਮ ਪੰਡਤ ਦੀ ਅਗੁਵਾਈ ਵਾਲੀ ਸਿਟੀ ਨੇ ਕਿਹਾ ਕਿ ਉਹ 5 . 5 ਅਰਬ ਡਾਲਰ ਇਲਾਵਾ ਪੂੰਜੀ ਜੁਟਾਣ ਲਈ ਪਬਲਿਕ ਏਕਸਚੇਂਜ ਆਫਰ ਦਾ ਦਾਇਰਾ ਵਧਾਏਗੀ ।				
9.	न्यूयार्क। अमेरिकी शेयर बाजार बृहस्पतिवार को गिरावट के साथ बंद हुए।	ਨਿਊਯਾਰਕ । ਅਮਰੀਕੀ ਸ਼ੇਅਰ ਬਾਜ਼ਾਰ ਵੀਰਵਾਰ ਨੂੰ ਗਿਰਾਵਟ ਦੇ ਨਾਲ ਬੰਦ ਹੋਏ ।				
10.	उधर, बृहस्पतिवार को बाजार बंद होने के बाद सरकार ने प्रमुख बैंकों के स्ट्रेस टेस्ट के नतीजे घोषित किए जिनके मुताबिक देश के 10 प्रमुख बैंकों को अपने बचाव के लिए और नकदी एकत्रित करनी पड़ेगी।	ਉੱਧਰ , ਵੀਰਵਾਰ ਨੂੰ ਬਾਜ਼ਾਰ ਬੰਦ ਹੋਣ ਦੇ ਬਾਅਦ ਸਰਕਾਰ ਨੇ ਪ੍ਰਮੁੱਖ ਬੈਂਕਾਂ ਦੇ ਸਟਰੇਸ ਟੇਸਟ ਦੇ ਨਤੀਜੇ ਘੋਸ਼ਿਤ ਕੀਤੇ ਜਿਨ੍ਹਾਂ ਦੇ ਮੁਤਾਬਕ ਦੇਸ਼ ਦੇ 10 ਪ੍ਰਮੁੱਖ ਬੈਂਕਾਂ ਨੂੰ ਆਪਣੇ ਬਚਾਵ ਲਈ ਹੋਰ ਨਗਦੀ ਇਕੱਠੀ ਕਰਨੀ ਪਵੇਗੀ ।				
11.	वरूण गाँधी ने पीलीभीत में मुसलमानों के ख़िलाफ़ भड़काऊ भाषण दिए थे	ਵਰੁਣ ਗਾਂਧੀ ਨੇ ਪੀਲੀਭੀਤ ਵਿੱਚ ਮੁਸਲਮਾਨਾਂ ਦੇ ਖਿਲਾਫ ਭੜਕਾਊ ਭਾਸ਼ਣ ਦਿੱਤੇ ਸਨ				

12.	इस मामले के प्रकाश में आने के बाद चुनाव आयोग के निर्देश पर वरुण गांधी के ख़िलाफ़ एफ़आईआर दर्ज हुई थी.	ਇਸ ਮਾਮਲੇ ਦੇ ਪ੍ਰਕਾਸ਼ ਵਿੱਚ ਆਉਣ ਦੇ ਬਾਅਦ ਚੋਣ ਕਮਿਸ਼ਨ ਦੇ ਨਿਰਦੇਸ਼ ਉੱਤੇ ਵਰੁਣ ਗਾਂਧੀ ਦੇ ਖਿਲਾਫ ਏਫ਼ਆਈਆਰ ਦਰਜ ਹੋਈ ਸੀ .				
13.	पहले ही वित्तीय वर्ष यानी वर्ष 2004-05 में केंद्र ने राज्य सरकार को 2831.82 करोड़ रूपए की राशि उपलब्ध कराई.	ਪਹਿਲਾਂ ਹੀ ਵਿੱਤੀ ਸਾਲ ਯਾਨੀ ਸਾਲ 2004 - 05 ਵਿੱਚ ਕੇਂਦਰ ਨੇ ਰਾਜ ਸਰਕਾਰ ਨੂੰ 2831 . 82 ਕਰੋੜ ਰੁਪਏ ਦੀ ਰਾਸ਼ੀ ਉਪਲੱਬਧ ਕਰਾਈ .				
14.	सपा के लिए पिछले चुनाव में जीती 32 में से अपनी 15 सीटों को बचाने की चुनौती है, जिनमें से लगभग आधा दर्जन सीटों की हार-जीत तो कल्याण फैक्टर की कसौटी पर ही कसा जाएगा।	ਸਪਾ ਲਈ ਪਿਛਲੇ ਚੋਣ ਵਿੱਚ ਜਿੱਤੀ 32 ਵਿੱਚੋਂ ਆਪਣੀ 15 ਸੀਟਾਂ ਨੂੰ ਬਚਾਉਣ ਦੀ ਚੁਣੋਤੀ ਹੈ , ਜਿਨ੍ਹਾਂ ਵਿਚੋਂ ਲੱਗਭੱਗ ਅੱਧਾ ਦਰਜਨ ਸੀਟਾਂ ਦੀ ਹਾਰ - ਜਿੱਤ ਤਾਂ ਕਲਿਆਣ ਫੈਕਟਰ ਦੀ ਕਸੌਟੀ ਉੱਤੇ ਹੀ ਕੱਸਿਆ ਜਾਵੇਗਾ ।				
15.	उन्होंने भाजपा नेताओं को इस तरह के बयान न देने की सलाह दी। इसके साथ ही शरद यादव ने विदेशी बैंकों में भारतीयों के जमा काले धन का मुद्दा फिर उठाया।	ਉਨ੍ਹਾਂਨੇ ਭਾਜਪਾ ਨੇਤਾਵਾਂ ਨੂੰ ਇਸ ਤਰ੍ਹਾਂ ਦੇ ਬਿਆਨ ਨਹੀਂ ਦੇਣ ਦੀ ਸਲਾਹ ਦਿੱਤੀ । ਇਸਦੇ ਨਾਲ ਹੀ ਸ਼ਰਦ ਯਾਦਵ ਨੇ ਵਿਦੇਸ਼ੀ ਬੈਂਕਾਂ ਵਿੱਚ ਭਾਰਤੀਆਂ ਦੇ ਜਮਾਂ ਕਾਲੇ ਧਨ ਦਾ ਮੁੱਦਾ ਫਿਰ ਚੁੱਕਿਆ ।				

16.	राहुल ने कहा, "मनमोहन सिंह हमारे प्रधानमंत्री हैं, वो यूपीए के भी प्रधानमंत्री हैं."	ਰਾਹੁਲ ਨੇ ਕਿਹਾ , “ਮਨਮੋਹਨ ਸਿੰਘ ਸਾਡੇ ਪ੍ਰਧਾਨਮੰਤਰੀ ਹਨ , ਉਹ ਯੂਪੀਏ ਦੇ ਵੀ ਪ੍ਰਧਾਨਮੰਤਰੀ ਹਨ.”				
17.	राहुल गांधी दो दिन के चुनाव प्रचार पर राजस्थान में हैं.	ਰਾਹੁਲ ਗਾਂਧੀ ਦੋ ਦਿਨ ਦੇ ਚੋਣ ਪ੍ਰਚਾਰ ਉੱਤੇ ਰਾਜਸਥਾਨ ਵਿੱਚ ਹਨ .				
18.	आर्थिक संकट का दबाव झेलने की क्षमता आंकने वाले इस टेस्ट में अमेरिका के 10 बड़े बैंक बेदम निकले हैं।	ਆਰਥਕ ਸੰਕਟ ਦਾ ਦਬਾਅ ਝੇਲਣ ਦੀ ਸਮਰੱਥਾ ਆਂਕਣ ਵਾਲੇ ਇਸ ਟੇਸਟ ਵਿੱਚ ਅਮਰੀਕਾ ਦੇ 10 ਵੱਡੇ ਬੈਂਕ ਬੇਦਮ ਨਿਕਲੇ ਹਨ ।				
19.	कंपनी की इस पहल से उसे अतिरिक्त सरकारी सहायता या सरकारी प्रतिभूतियों को सामान्य शेयरों में परिवर्तित किए बगैर अपना पूंजी आधार बढ़ाने में मदद मिलेगी।	ਕੰਪਨੀ ਦੀ ਇਸ ਪਹਿਲ ਤੋਂ ਉਸਨੂੰ ਹੋਰ ਸਰਕਾਰੀ ਸਹਾਇਤਾ ਜਾਂ ਸਰਕਾਰੀ ਪ੍ਰਤੀਭੂਤੀਯੋਂ ਨੂੰ ਇੱਕੋ ਜਿਹੇ ਸ਼ੇਅਰਾਂ ਵਿੱਚ ਪਰਿਵਰਤਿਤ ਕੀਤੇ ਬਿਨਾਂ ਆਪਣਾ ਪੂੰਜੀ ਆਧਾਰ ਵਧਾਉਣ ਵਿੱਚ ਮਦਦ ਮਿਲੇਗੀ ।				
20.	उन्होंने कहा कि राजग द्वारा इसे चुनावी मुद्दा बनाए जाने के बाद मजबूरी में मनमोहन सरकार अब कार्रवाई करने का दिखावा कर रही है।	ਉਨ੍ਹਾਂਨੇ ਕਿਹਾ ਕਿ ਰਾਜਗ ਦੁਆਰਾ ਇਸਨੂੰ ਚੁਨਾਵੀ ਮੁੱਦਾ ਬਣਾਏ ਜਾਣ ਦੇ ਬਾਅਦ ਮਜਬੂਰੀ ਵਿੱਚ ਮਨਮੋਹਣ ਸਰਕਾਰ ਹੁਣ ਕਾਰਵਾਈ ਕਰਣ ਦਾ ਦਿਖਾਵਾ ਕਰ ਰਹੀ ਹੈ ।				

21.	संपादकों के मुताबिक, "टाइम 100 संस्करण में हम उन लोगों का नाम देते हैं जो हमारी दुनिया को सबसे ज़्यादा प्रभावित करते हैं."	ਸੰਪਾਦਕਾਂ ਦੇ ਮੁਤਾਬਕ , ਟਾਇਮ 100 ਸੰਸਕਰਣ ਵਿੱਚ ਅਸੀ ਉਨ੍ਹਾਂ ਲੋਕਾਂ ਦਾ ਨਾਮ ਦਿੰਦੇ ਹਨ ਜੋ ਸਾਡੀ ਦੁਨੀਆ ਨੂੰ ਸਭਤੋਂ ਜ਼ਿਆਦਾ ਪ੍ਰਭਾਵਿਤ ਕਰਦੇ ਹਾਂ .				
22.	मेगास्टार अमिताभ बच्चन ने कल अपने संवैधानिक दायित्व का निर्वाह करने के साथ ही अपने सामाजिक दायित्व का भी बखूबी निर्वाह किया।	ਮੇਗਾਸਟਾਰ ਅਮੀਤਾਭ ਬੱਚਨ ਨੇ ਕੱਲ ਆਪਣੇ ਸੰਵਿਧਾਨਕ ਫਰਜ ਦਾ ਗੁਜਾਰਾ ਕਰਣ ਦੇ ਨਾਲ ਹੀ ਆਪਣੇ ਸਾਮਾਜਕ ਫਰਜ ਦਾ ਵੀ ਬਖੂਬੀ ਗੁਜਾਰਾ ਕੀਤਾ ।				
23.	इसके अलावा उप मुख्यमंत्री सुखबीर बादल ने भी इंटरनेट पर कई प्रोफाइल बना रखी है।	ਇਸਦੇ ਇਲਾਵਾ ਉਪ ਮੁੱਖਮੰਤਰੀ ਸੁਖਬੀਰ ਬਾਦਲ ਨੇ ਵੀ ਇੰਟਰਨੇਟ ਉੱਤੇ ਕਈ ਪ੍ਰੋਫਾਇਲ ਬਣਾ ਰੱਖੀ ਹੈ ।				
24.	हाल ही में लता मंगेशकर ने मधुर भंडारकर की फ़िल्म ‘जेल’ में एक धार्मिक गीत रिकॉर्ड किया है.	ਹਾਲ ਹੀ ਵਿੱਚ ਲਤਾ ਮੰਗੇਸ਼ਕਰ ਨੇ ਮਧੁਰ ਭੰਡਾਰਕਰ ਦੀ ਫਿਲਮ ‘ਜੇਲ੍ਹ’ ਵਿੱਚ ਇੱਕ ਧਾਰਮਿਕ ਗੀਤ ਰਿਕਾਰਡ ਕੀਤਾ ਹੈ .				
25.	हिमेश जी, बात तो सही है लेकिन कर्ज़ के हश्र के बाद आपको नहीं लगता कि दर्शकों का आपको हीरो के रुप में स्वीकार करना थोड़ा मुश्किल होगा.	ਹਿਮੇਸ਼ ਜੀ , ਗੱਲ ਤਾਂ ਠੀਕ ਹੈ ਲੇਕਿਨ ਕਰਜ ਦੇ ਹਾਲ ਦੇ ਬਾਅਦ ਤੁਹਾਨੂੰ ਨਹੀਂ ਲੱਗਦਾ ਕਿ ਦਰਸ਼ਕਾਂ ਦਾ ਤੁਹਾਨੂੰ ਹੀਰੋ ਦੇ ਰੁਪ ਵਿੱਚ ਸਵੀਕਾਰ ਕਰਣਾ ਥੋੜ੍ਹਾ ਮੁਸ਼ਕਲ ਹੋਵੇਗਾ .				

26.	उन्होंने भी लोकसभा चुनाव में जीत का दावा किया है।	ਉਨ੍ਹਾਂਨੇ ਵੀ ਲੋਕਸਭਾ ਚੋਣ ਵਿੱਚ ਜਿੱਤ ਦਾ ਦਾਵਾ ਕੀਤਾ ਹੈ ।				
27.	फ़िल्म स्लमडॉग मिलियनेयर की बाल कलाकार रुबीना अली काफ़ी प्रसिद्ध हो गई हैं	ਫਿਲਮ ਸਲਮਡਾਗ ਮਿਲਿਅਨੇਇਰ ਦੀ ਬਾਲ ਕਲਾਕਾਰ ਰੁਬੀਨਾ ਅਲੀ ਕਾਫ਼ੀ ਪ੍ਰਸਿੱਧ ਹੋ ਗਈ ਹੈ				
28.	मैं इस परिवार को पिछले बीस सालों से जानता हूँ, रफ़ीक बहुत शरीफ आदमी है, वो ऐसी हरकत कभी नहीं करेगा	ਮੈਂ ਇਸ ਪਰਵਾਰ ਨੂੰ ਪਿਛਲੇ ਵੀਹ ਸਾਲਾਂ ਤੋਂ ਜਾਣਦਾ ਹਾਂ , ਰਫੀਕ ਬਹੁਤ ਸ਼ਰੀਫ ਆਦਮੀ ਹੈ , ਉਹ ਅਜਿਹੀ ਹਰਕੱਤ ਕਦੇ ਨਹੀਂ ਕਰੇਗਾ				
29.	करीना पूरे दिन घास से बने मचान पर तो कभी बैलगाड़ी पर मस्ती करती नजर आईं।	ਕਰੀਨਾ ਪੂਰੇ ਦਿਨ ਘਾਹ ਨਾਲ ਬਣੇ ਮਚਾਣ ਉੱਤੇ ਤਾਂ ਕਦੇ ਬੈਲਗਾੜੀ ਉੱਤੇ ਮਸਤੀ ਕਰਦੀ ਨਜ਼ਰ ਆਈ ।				
30.	शुक्रवार को करीना के आने के साथ ही शहर में सैफ के भी आने की उम्मीद लगाई जा रही थी।	ਸ਼ੁੱਕਰਵਾਰ ਨੂੰ ਕਰੀਨਾ ਦੇ ਆਉਣ ਦੇ ਨਾਲ ਹੀ ਸ਼ਹਿਰ ਵਿੱਚ ਸੈਫ ਦੇ ਵੀ ਆਉਣ ਦੀ ਉੰਮੀਦ ਲਗਾਈ ਜਾ ਰਹੀ ਸੀ ।				
31.	हाल में मोहमंद में हुई सैन्य कार्रवाई में 18 चरमपंथी मारे गए थे	ਹਾਲ ਵਿੱਚ ਮੋਹਮੰਦ ਵਿੱਚ ਹੋਈ ਫੌਜੀ ਕਾਰਵਾਈ ਵਿੱਚ 18 ਚਰਮਪੰਥੀ ਮਾਰੇ ਗਏ ਸਨ				
32.	ग़ौरतलब है कि पाकिस्तान के पश्चिमोत्तर में बुनेर में पिछले कुछ हफ़्तों में सेना और तालेबान चरमपंथियों के बीच भीषण संघर्ष हुआ है.	ਧਿਆਨ ਯੋਗ ਹੈ ਕਿ ਪਾਕਿਸਤਾਨ ਦੇ ਪਸ਼ਚਿਮੋੱਤਰ ਵਿੱਚ ਬੁਨੇਰ ਵਿੱਚ ਪਿਛਲੇ ਕੁੱਝ ਹਫਤੀਆਂ ਵਿੱਚ ਫੌਜ ਅਤੇ ਤਾਲੇਬਾਨ ਚਰਮਪੰਥੀਆਂ ਦੇ ਵਿੱਚ ਭੀਸ਼ਨ ਸੰਘਰਸ਼ ਹੋਇਆ ਹੈ				

33.	इस्लामाबाद,। पाकिस्तान के गृह मंत्री रहमान मलिक ने सोमवार को कहा कि देश के अशांत पश्चिमोत्तर क्षेत्र में चल रहे एक बड़े सैन्य अभियान में करीब 700 तालिबान आतंकियों को मार गिराया गया है और सभी आतंकियों का खात्मा होने तक वहां सैन्य कार्रवाई जारी रहेगी।	ਇਸਲਾਮਾਬਾਦ । ਪਾਕਿਸਤਾਨ ਦੇ ਗ੍ਰਹ ਮੰਤਰੀ ਰਹਿਮਾਨ ਮਲਿਕ ਨੇ ਸੋਮਵਾਰ ਨੂੰ ਕਿਹਾ ਕਿ ਦੇਸ਼ ਦੇ ਬੇਚੈਨ ਪਸ਼ਚਿਮੋੱਤਰ ਖੇਤਰ ਵਿੱਚ ਚੱਲ ਰਹੇ ਇੱਕ ਵੱਡੇ ਫੌਜੀ ਅਭਿਆਨ ਵਿੱਚ ਕਰੀਬ 700 ਤਾਲਿਬਾਨ ਆਤੰਕੀਆਂ ਨੂੰ ਮਾਰ ਗਿਰਾਇਆ ਗਿਆ ਹੈ ਅਤੇ ਸਾਰੇ ਆਤੰਕੀਆਂ ਦਾ ਖਾਤਮਾ ਹੋਣ ਤੱਕ ਉੱਥੇ ਫੌਜੀ ਕਾਰਵਾਈ ਜਾਰੀ ਰਹੇਗੀ ।				
34.	गृह मंत्री ने कहा कि यह पूरे देश के लिए एक परीक्षा है।	ਗ੍ਰਹ ਮੰਤਰੀ ਨੇ ਕਿਹਾ ਕਿ ਇਹ ਪੂਰੇ ਦੇਸ਼ ਲਈ ਇੱਕ ਪਰੀਖਿਆ ਹੈ ।				
35.	पाकिस्तान के सीमावर्ती इलाक़ों में मिसाईल हमले होते रहे हैं और इसके लिए पाकिस्तान अमरीका पर आरोप लगाता रहा है.	ਪਾਕਿਸਤਾਨ ਦੇ ਸੀਮਾਵਰਤੀ ਇਲਾਕੀਆਂ ਵਿੱਚ ਮਿਸਾਈਲ ਹਮਲੇ ਹੁੰਦੇ ਰਹੇ ਹਨ ਅਤੇ ਇਸਦੇ ਲਈ ਪਾਕਿਸਤਾਨ ਅਮਰੀਕਾ ਤੇ ਇਲਜ਼ਾਮ ਲਗਾਉਂਦਾ ਰਿਹਾ ਹੈ .				
36.	प्रधानमंत्री के बयान की भाजपा ने भी आलोचना की है। दिल्ली प्रदेश के महामंत्री आरपी सिंह ने कहा कि प्रधानमंत्री ने पद की गरिमा धूमिल की है।	ਪ੍ਰਧਾਨਮੰਤਰੀ ਦੇ ਬਿਆਨ ਦੀ ਭਾਜਪਾ ਨੇ ਵੀ ਆਲੋਚਨਾ ਕੀਤੀ ਹੈ । ਦਿੱਲੀ ਪ੍ਰਦੇਸ਼ ਦੇ ਪ੍ਰਧਾਨ ਮੰਤਰੀ ਆਰਪੀ ਸਿੰਘ ਨੇ ਕਿਹਾ ਕਿ ਪ੍ਰਧਾਨਮੰਤਰੀ ਨੇ ਪਦ ਦੀ ਗਰਿਮਾ ਧੂਮੀਲ ਕੀਤੀ ਹੈ ।				

37.	प्रधानमंत्री के सरकारी आवास पर हुई यह बातचीत इस मायने में महत्वपूर्ण है कि माओवादी नेता ने भारत पर नेपाल के अंदरूनी मामले में दखलंदाजी का आरोप लगाया था जिसे बाद में उन्होंने हल्का करने की कोशिश की थी।	ਪ੍ਰਧਾਨਮੰਤਰੀ ਦੇ ਸਰਕਾਰੀ ਘਰ ਤੇ ਹੋਈ ਇਹ ਗੱਲਬਾਤ ਇਸ ਮਾਅਨੇ ਵਿੱਚ ਮਹੱਤਵਪੂਰਣ ਹੈ ਕਿ ਮਾਓਵਾਦੀ ਨੇਤਾ ਨੇ ਭਾਰਤ ਤੇ ਨੇਪਾਲ ਦੇ ਅੰਦਰੂਨੀ ਮਾਮਲੇ ਵਿੱਚ ਦਖੱਲੰਦਾਜੀ ਦਾ ਇਲਜ਼ਾਮ ਲਗਾਇਆ ਸੀ ਜਿਨੂੰ ਬਾਅਦ ਵਿੱਚ ਉਨ੍ਹਾਂਨੇ ਹਲਕਾ ਕਰਣ ਦੀ ਕੋਸ਼ਿਸ਼ ਕੀਤੀ ਸੀ ।				
38.	गुजरात में 2002 में हुए दंगों के कुछ मामलों में प्रतिदिन सुनवाई के आधार पर फास्ट ट्रैक अदालतें गठित करने के सुप्रीमकोर्ट के आज के फैसले पर भाजपा ने यह प्रतिक्रिया दी है।	ਗੁਜਰਾਤ ਵਿੱਚ 2002 ਵਿੱਚ ਹੋਏ ਦੰਗੀਆਂ ਦੇ ਕੁੱਝ ਮਾਮਲੀਆਂ ਵਿੱਚ ਨਿੱਤ ਸੁਣਵਾਈ ਦੇ ਆਧਾਰ ਉੱਤੇ ਫਾਸਟ ਟ੍ਰੈਕ ਅਦਾਲਤਾਂ ਗਠਿਤ ਕਰਣ ਦੇ ਸੁਪਰੀਮ ਕੋਰਟ ਦੇ ਅਜੋਕੇ ਫੈਸਲੇ ਉੱਤੇ ਭਾਜਪਾ ਨੇ ਇਹ ਪ੍ਰਤੀਕਿਰਆ ਦਿੱਤੀ ਹੈ ।				
39.	मौसम विभाग ने अंडमान के सागर में मानसून की सालाना बारिश थोड़ा देर से होने की आशंका जताई है।	ਮੌਸਮ ਵਿਭਾਗ ਨੇ ਅੰਡਮਾਨ ਦੇ ਸਾਗਰ ਵਿੱਚ ਮਾਨਸੂਨ ਦੀ ਸਾਲਾਨਾ ਮੀਂਹ ਥੋੜ੍ਹਾ ਦੇਰ ਤੋਂ ਹੋਣ ਦੀ ਸੰਦੇਹ ਜਤਾਈ ਹੈ ।				
40.	विंश्व की तीन सबसे मोटी बर्फीली परतों में से एक पश्चिमी अंटार्कटिका की तह है।	ਵਿੰਸ਼ਵ ਦੀ ਤਿੰਨ ਸਭਤੋਂ ਮੋਟੀ ਬਰਫੀਲੀ ਪਰਤਾਂ ਵਿੱਚੋਂ ਇੱਕ ਪੱਛਮ ਵਾਲਾ ਅੰਟਾਰਕਟੀਕਾ ਦੀ ਤਹ ਹੈ ।				
41.	फिलहाल डरने की जरूरत नहीं	ਫਿਲਹਾਲ ਡਰਨ ਦੀ ਜ਼ਰੂਰਤ ਨਹੀਂ				

42.	उनके और पाइट के विद्यार्थियों के बीच का संवाद बेहद रोचक रहा।	ਉਨ੍ਹਾਂ ਦੇ ਅਤੇ ਪਾਇਟ ਦੇ ਵਿਦਿਆਰਥੀਆਂ ਦੇ ਵਿੱਚ ਦਾ ਸੰਵਾਦ ਬੇਹੱਦ ਰੋਚਕ ਰਿਹਾ ।				
43.	इंदिरा गांधी राष्ट्रीय मुक्त विश्वविद्यालय [इग्नू] के छात्रों को उनकी मांग पर घर बैठे परीक्षा देने की सुविधा मिलने जा रही है।	ਇੰਦਰਾ ਗਾਂਧੀ ਰਾਸ਼ਟਰੀ ਅਜ਼ਾਦ ਯੂਨੀਵਰਸਿਟੀ [ਇਗਨੂ] ਦੇ ਵਿਦਿਆਰਥੀਆਂ ਨੂੰ ਉਨ੍ਹਾਂ ਦੀ ਮੰਗ ਪਰ ਘਰ ਬੈਠੇ ਪਰੀਖਿਆ ਦੇਣ ਦੀ ਸੁਵਿਧਾ ਮਿਲਣ ਜਾ ਰਹੀ ਹੈ ।				
44.	धोनी ने कप्तानी पारी खेली	ਧੋਨੀ ਨੇ ਕਪਤਾਨੀ ਪਾਰੀ ਖੇਡੀ				
45.	इसके बाद सचिन तेंदुलकर टीम की नैया पार लगाने के लिए मैदान पर आए लेकिन वह अधिक रन नहीं बना सके।	ਇਸਦੇ ਬਾਅਦ ਸਚਿਨ ਤੇਂਦੁਲਕਰ ਟੀਮ ਦੀ ਬੇੜੀ ਪਾਰ ਲਗਾਉਣ ਲਈ ਮੈਦਾਨ ਪਰ ਆਏ ਲੇਕਿਨ ਉਹ ਜਿਆਦਾ ਰਣ ਨਹੀਂ ਬਣਾ ਸਕੇ ।				
46.	राजस्थान की शुरुआत बेहद खराब रही और एक बार दबाव में आने के बाद उसके सभी बल्लेबाज अपना विकेट फेंककर चलते बने।	ਰਾਜਸਥਾਨ ਦੀ ਸ਼ੁਰੁਆਤ ਬੇਹੱਦ ਖ਼ਰਾਬ ਰਹੀ ਅਤੇ ਇੱਕ ਵਾਰ ਦਬਾਅ ਵਿੱਚ ਆਉਣ ਦੇ ਬਾਅਦ ਉਸਦੇ ਸਾਰੇ ਬੱਲੇਬਾਜ ਆਪਣਾ ਵਿਕੇਟ ਸੁੱਟਕੇ ਚਲਦੇ ਬਣੇ ।				
47.	विश्व बैडमिंटन चैंपियनशिप को तैयार भारत May 08, 06:45 pm	ਸੰਸਾਰ ਬੈਡਮਿੰਟਨ ਚੈਂਪਿਅਨਸ਼ਿਪ ਨੂੰ ਤਿਆਰ ਭਾਰਤ May 08 , 06 : 45 pm				

48.	मोहन बागान अभी तक खाता नहीं खोल पाया है और ग्रुप में सबसे निचले स्थान पर है।	ਮੋਹਨ ਬਾਗਾਨ ਹੁਣੇ ਤੱਕ ਖਾਤਾ ਨਹੀਂ ਖੋਲ ਪਾਇਆ ਹੈ ਹੋਰ ਗਰੁਪ ਵਿੱਚ ਸਭਤੋਂ ਹੇਠਲੇ ਸਥਾਨ ਪਰ ਹੈ ।				
49.	दो गोल से बढ़त बनाने के बावजूद चीन के साथ 2-2 से ड्रा खेलने के बाद भारत मंगलवार को एशिया कप हाकी में सेमीफाइनल की दौड़ से बाहर हो गया जिससे खिताब बरकरार रखने का उसका ख्वाब भी चूर चूर हो गया।	ਦੋ ਗੋਲ ਨੂੰ ਵਾਧੇ ਬਣਾਉਣ ਦੇ ਬਾਵਜੂਦ ਚੀਨ ਦੇ ਨਾਲ 2 - 2 ਤੋਂ ਡਰਾ ਖੇਡਣ ਦੇ ਬਾਅਦ ਭਾਰਤ ਮੰਗਲਵਾਰ ਨੂੰ ਏਸ਼ਿਆ ਕਪ ਹਾਕੀ ਵਿੱਚ ਸੇਮੀਫਾਇਨਲ ਦੀ ਦੋੜ ਤੋਂ ਬਾਹਰ ਹੋ ਗਿਆ ਜਿਸਦੇ ਨਾਲ ਖਿਤਾਬ ਬਰਕਰਾਰ ਰੱਖਣ ਦਾ ਉਸਦਾ ਸੁੱਪਣਾ ਵੀ ਚੂਰ ਚੂਰ ਹੋ ਗਿਆ ।				
50.	सार्वजनिक क्षेत्र के आईडीबीआई बैंक ने भी एफडी पर ब्याज दरें आधा से एक फीसदी तक घटा दी हैं। नई दरें 21 मई से लागू होंगी।	ਸਾਰਵਜਨਿਕ ਖੇਤਰ ਦੇ ਆਈਡੀਬੀਆਈ ਬੈਂਕ ਨੇ ਵੀ ਏਫਡੀ ਉੱਤੇ ਵਿਆਜ ਦਰਾਂ ਅੱਧਾ ਤੋਂ ਇੱਕ ਫੀਸਦੀ ਤੱਕ ਘਟਾ ਦਿੱਤੀਆਂ ਹਨ । ਨਵੀਂ ਦਰਾਂ 21 ਮਈ ਤੋਂ ਲਾਗੂ ਹੋਵੇਗੀ				

Accuracy Test - Literature

S.No.	Hindi Sentence	Punjabi Sentence	0	1	2	3
1.	अगर तुझे वह चीज न मिले तो खबरदार इधर रूख न करना, वर्ना सूली पर खिंचवा दूँगी	ਜੇਕਰ ਤੈਨੂੰ ਉਹ ਚੀਜ ਨਹੀਂ ਮਿਲੇ ਤਾਂ ਖਬਰਦਾਰ ਏਧਰ ਰੁੱਖ ਨਹੀਂ ਕਰਣਾ , ਵਰਨਾ ਸੂਲੀ ਪਰ ਖਿੱਚਵਾ ਦੂੰਗੀ				
2.	इसकी सूचना ने अज्ञान बलिका को मुंह ढांप कर एक कोने में बिठा रखा है।	ਇਸਦੀ ਸੂਚਨਾ ਨੇ ਅਗਿਆਨ ਬਲਿਕਾ ਨੂੰ ਮੁੰਹ ਢਾਂਪ ਕਰ ਇੱਕ ਕੋਨੇ ਵਿੱਚ ਬਿਠਾ ਰੱਖਿਆ ਹੈ ।				
3.	संध्या का समय था, निर्मला छत पर जानकर अकेली बैठी आकाश की और तृषित नेत्रों से ताक रही थी।	ਸ਼ਾਮ ਦਾ ਸਮਾਂ ਸੀ , ਨਿਰਮਲਾ ਛੱਤ ਪਰ ਜਾਨਕੇ ਇਕੱਲੀ ਬੈਠੀ ਅਕਾਸ਼ ਕੀਤੀ ਹੋਰ ਤ੍ਰਸ਼ਿਤ ਨੇਤਰਾਂ ਤੋਂ ਵੇਖ ਰਹੀ ਸੀ ।				
4.	निर्मला- ने उदासीन भाव से कहा-तू जा, मैं न जाऊंगी।	ਨਿਰਮਲਾ - ਨੇ ਉਦਾਸੀਨ ਭਾਵ ਨੂੰ ਕਿਹਾ - ਤੂੰ ਜਾ , ਮੈਂ ਨਹੀਂ ਜਾਵਾਂਗੀ ।				
5.	बाग में फूल खिले हुए थे। मीठी-मीठी सुगन्ध आ रही थी। चैत की शीतल मन्द समीर चल रही थी।	ਬਾਗ ਵਿੱਚ ਫੁਲ ਖਿੜੇ ਹੋਏ ਸਨ । ਮਿੱਠੀ - ਮਿੱਠੀ ਸੁਗੰਧ ਆ ਰਹੀ ਸੀ । ਚੇਤ ਦੀ ਸੀਤਲ ਮੰਦ ਸਮੀਰ ਚੱਲ ਰਹੀ ਸੀ ।				
6.	यह कहकर कल्याणी कमरे के बाहर निकल गई।	ਇਹ ਕਹਿਕੇ ਕਲਿਆਣੀ ਕਮਰੇ ਦੇ ਬਾਹਰ ਨਿਕਲ ਗਈ ।				
7.	मुंशीजी तो भोजन करने गये और निर्मला द्वार की चौखट पर खड़ी सोच रही थी- भगवान।	ਮੁੰਸ਼ੀਜੀ ਤਾਂ ਭੋਜਨ ਕਰਣ ਗਏ ਅਤੇ ਨਿਰਮਲਾ ਦਵਾਰ ਦੀ ਚੌਖਟ ਪਰ ਖੜੀ ਸੋਚ ਰਹੀ ਸੀ - ਭਗਵਾਨ ।				
8.	साधु- कभी आ जाऊंगा बच्चा, तुम्हारा घर कहां है?	ਸਾਧੁ - ਕਦੇ ਆ ਜਾਵਾਂਗਾ ਬੱਚਾ , ਤੁਹਾਡਾ ਘਰ ਕਿੱਥੇ ਹੈ ?				

9.	एक दिन निर्मला ने सियाराम को घी लाने के लिए बाजार भेजा।	ਇੱਕ ਦਿਨ ਨਿਰਮਲਾ ਨੇ ਸਿਆਰਾਮ ਨੂੰ ਘੀ ਲਿਆਉਣ ਲਈ ਬਾਜ਼ਾਰ ਭੇਜਿਆ ।				
10.	माता-झूठ ने बोल! तूने पांच सौ रुपये के नोट नहीं भेजे थे?	ਮਾਤਾ - ਝੂਠ ਨੇ ਬੋਲ ! ਤੂੰ ਪੰਜ ਸੌ ਰੁਪਏ ਦੇ ਨੋਟ ਨਹੀਂ ਭੇਜੇ ਸਨ ?				
11.	क्या मेरी दशा को और भी दारुण बनाना चाहते हो?	ਕੀ ਮੇਰੀ ਹਾਲਤ ਨੂੰ ਹੋਰ ਵੀ ਦਾਰੁਣ ਬਣਾਉਣਾ ਚਾਹੁੰਦੇ ਹੋ ?				
12.	उस दिन से निर्मला का रंग-ढंग बदलने लगा।	ਉਸ ਦਿਨ ਤੋਂ ਨਿਰਮਲਾ ਦਾ ਰੰਗ - ਢੰਗ ਬਦਲਨ ਲਗਾ ।				
13.	क्या इन्हें सचमुच कोई भीषण रोग हो रहा है?	ਕੀ ਇਨ੍ਹਾਂ ਨੂੰ ਸਚਮੁੱਚ ਕੋਈ ਭੀਸ਼ਨ ਰੋਗ ਹੋ ਰਿਹਾ ਹੈ ?				
14.	निर्मला- तो मैं झूठ कहती हूं?	ਨਿਰਮਲਾ - ਤਾਂ ਮੈਂ ਝੂਠ ਕਹਿੰਦੀ ਹਾਂ ?				
15.	बेचारे लड़के को बार-बार दौड़ाया करती है। सौतेली मां है न! अपनी मां हो तो कुछ ख्याल भी करे।	ਬੇਚਾਰੇ ਮੁੰਡੇ ਨੂੰ ਵਾਰ - ਵਾਰ ਦੌੜਾਇਆ ਕਰਦੀ ਹੈ । ਮਤ੍ਰੇਈ ਮਾਂ ਹੈ ਨਹੀਂ ! ਆਪਣੀ ਮਾਂ ਹੋ ਤਾਂ ਕੁੱਝ ਖਿਆਲ ਵੀ ਕਰੇ ।				
16.	दाननाथ को ऐसी उत्तम स्पीच को न सुनने का अत्यंत शोक हुआ। बोले—यार, मैं जंम का अभागा हूँ। क्या अब फिर कोई व्याख्यान न होगा?	ਦਾਨਨਾਥ ਨੂੰ ਅਜਿਹੀ ਉੱਤਮ ਸਪੀਚ ਨੂੰ ਨਹੀਂ ਸੁਣਨ ਦਾ ਅਤਿਅੰਤ ਸੋਗ ਹੋਇਆ । ਬੋਲੇ—ਯਾਰ , ਮੈਂ ਜੰਮ ਦਾ ਅਭਾਗਾ ਹਾਂ । ਕੀ ਹੁਣ ਫਿਰ ਕੋਈ ਵਿਖਿਆਨ ਨਹੀਂ ਹੋਵੇਗਾ ?				

17.	दाननाथ तो यह बातचीत करके अपने मकान को रवाना हुए और अमृतराय उसी अँधेरे में, बड़ी देर तक चुपचाप खड़े रहे।	ਦਾਨਨਾਥ ਤਾਂ ਇਹ ਗੱਲਬਾਤ ਕਰਕੇ ਆਪਣੇ ਮਕਾਨ ਨੂੰ ਰਵਾਨਾ ਹੋਏ ਅਤੇ ਅਮ੍ਰਿਤ ਰਾਇ ਉਸੀ ਹਨ੍ਹੇਰੇ ਵਿੱਚ , ਵੱਡੀ ਦੇਰ ਤੱਕ ਚੁਪਚਾਪ ਖੜੇ ਰਹੇ ।				
18.	आज भी, जब अमृतराय ने उससे अपने इरादे जाहिर किये तब उसेन सच्चे दिल से उनको समझाकर ऊँच नीच सुझाया।	ਅੱਜ ਵੀ , ਜਦੋਂ ਅਮ੍ਰਿਤ ਰਾਇ ਨੇ ਉਸਤੋਂ ਆਪਣੇ ਇਰਾਦੇ ਸਾਫ਼ ਕੀਤੇ ਤੱਦ ਉਸੇਨ ਸੱਚੇ ਦਿਲੋਂ ਉਨ੍ਹਾਂਨੂੰ ਸੱਮਝਾਕੇ ਉੰਚ ਨੀਚ ਸੁਝਾਇਆ ।				
19.	लाला बदरीप्रसाद अमृतराय के बाप के दोस्तों में थे और अगर उनसे अधिक प्रतिष्ठित न थे तो बहुत हेठे भी न थे। दोनो में लड़के-लड़की के ब्याह की बातचीत पक्की हो गयी थी।	ਲਾਲਾ ਬਦਰੀਪ੍ਰਸਾਦ ਅਮ੍ਰਿਤ ਰਾਇ ਦੇ ਬਾਪ ਦੇ ਦੋਸਤਾਂ ਵਿੱਚ ਸਨ ਅਤੇ ਜੇਕਰ ਉਨ੍ਹਾਂ ਨੂੰ ਜਿਆਦਾ ਇੱਜ਼ਤ ਵਾਲਾ ਨਹੀਂ ਸਨ ਤਾਂ ਬਹੁਤ ਹੇਠੇ ਵੀ ਨਹੀਂ ਸਨ । ਦੋਨੋਂ ਵਿੱਚ ਮੁੰਡੇ - ਕੁੜੀ ਦੇ ਵਿਆਹ ਦੀ ਗੱਲਬਾਤ ਪੱਕੀ ਹੋ ਗਈ ਸੀ ।				
20.	पूर्णा ने आते ही सब स्त्रियों को वहाँ से हटा दिया, प्रेमा को इत्र सुघाया केवडे और गुलाब का छींटा मुख पर मारा। धीरे धीरे उसके तलवे सहलाये, सब खिड़कियाँ खुलवा दीं।	ਦਸਮੀਂ ਨੇ ਆਉਂਦੇ ਹੀ ਸਭ ਸਤਰੀਆਂ ਨੂੰ ਉੱਥੇ ਨੂੰ ਹਟਾ ਦਿੱਤਾ , ਪ੍ਰੇਮਾ ਨੂੰ ਇਤਰ ਸੁਘਾਇਆ ਕੇਵਡੇ ਅਤੇ ਗੁਲਾਬ ਦਾ ਛੀਂਟਾ ਮੂੰਹ ਪਰ ਮਾਰਿਆ । ਹੌਲੀ - ਹੌਲੀ ਉਸਦੇ ਤਲਵੇ ਸਹਲਾਏ , ਸਭ ਖਿੜਕੀਆਂ ਖੁੱਲ੍ਹਵਾ ਦਿੱਤੀ ।				

21.	मिस्टर शर्मा—(मूँछो पर हाथ फेरकर) वह ताजा खबर लाया हूँ कि आप लोग सुनकर फड़क जायँगे।	ਮਿਸਟਰ ਸ਼ਰਮਾ— (ਮੂੰਛੇ ਪਰ ਹੱਥ ਫੇਰਕੇ) ਉਹ ਤਾਜ਼ਾ ਖਬਰ ਲਿਆਇਆ ਹਾਂ ਕਿ ਤੁਸੀ ਲੋਕ ਸੁਣਕੇ ਫੜਫੜਾਹਟ ਜਾਇੰਗੇ ।				
22.	खुलासा यह कि अमृतराय को यहाँ से सत्तरह हजार रूपया मिला। मुंशी बदरीप्रसाद ने अकेले बारह हजार दिया जो उनकी उम्मेद से बहुत ज्यादा था।	ਖੁਲਾਸਾ ਇਹ ਕਿ ਅਮ੍ਰਤਰਾਏ ਨੂੰ ਇੱਥੇ ਤੋਂ ਸੱਤਰਹ ਹਜਾਰ ਰੁਪਆ ਮਿਲਿਆ । ਮੁਨਸ਼ੀ ਬਦਰੀਪ੍ਰਸਾਦ ਨੇ ਇਕੱਲੇ ਬਾਰਾਂ ਹਜਾਰ ਦਿੱਤਾ ਜੋ ਉਨ੍ਹਾਂ ਦੀ ਉਂਮੇਦ ਤੋਂ ਬਹੁਤ ਜ਼ਿਆਦਾ ਸੀ ।				
23.	राम—(मुस्कराकर) चुप। ऐसा भी कोई कहता है।	ਰਾਮ— (ਮੁਸਕਰਾਕੇ) ਚੁਪ । ਅਜਿਹਾ ਵੀ ਕੋਈ ਕਹਿੰਦਾ ਹੈ ।				
24.	अपने दिल का परिचय उसको एक दिन यों मिला कि बाबू अमृतराय नियत समय पर नहीं आये। थोड़ी देर तक तो वह उनकी राह देखती रही मगर जब वह अब भी न आये तब तो उसका दिल कुछ मसोसने लगा। बड़ी व्याकुलता से दौड़ी हुई दीवाजे पर आयी और आध घंटे तक कान लगाये खड़ी रही, फिर भीतर आयी और मन मारकर बैठ गयी।	ਆਪਣੇ ਦਿਲ ਦਾ ਜਾਣ ਪਹਿਚਾਣ ਉਹਨੂੰ ਇੱਕ ਦਿਨ ਇੰਜ ਮਿਲਿਆ ਕਿ ਬਾਬੂ ਅਮ੍ਰਿਤ ਰਾਇ ਨਿਅਤ ਸਮਾਂ ਪਰ ਨਹੀਂ ਆਏ । ਥੋੜ੍ਹੀ ਦੇਰ ਤੱਕ ਤਾਂ ਉਹ ਉਨ੍ਹਾਂ ਦੀ ਰੱਸਤਾ ਵੇਖਦੀ ਰਹੀ ਮਗਰ ਜਦੋਂ ਉਹ ਹੁਣ ਵੀ ਨਹੀਂ ਆਏ ਤੱਦ ਤਾਂ ਉਸਦਾ ਦਿਲ ਕੁੱਝ ਮਸੋਸਨੇ ਲਗਾ । ਵੱਡੀ ਵਿਆਕੁਲਤਾ ਤੋਂ ਦੌੜੀ ਹੋਈ ਦੀਵਾਜੇ ਪਰ ਆਈ ਅਤੇ ਅੱਧ ਘੰਟੇ ਤੱਕ ਕੰਨ ਲਗਾਏ ਖੜੀ ਰਹੀ , ਫਿਰ ਅੰਦਰ ਆਈ ਅਤੇ ਮਨ ਮਾਰਕੇ ਬੈਠ ਗਈ ।				

25.	अमृतराय—(दबी जबान से) वह सब कहार मेरे नौकर हैं।	ਅਮ੍ਰਤਰਾਏ— (ਦੱਬੀ ਜਬਾਨ ਤੋਂ) ਉਹ ਸਭ ਕਹਾਰ ਮੇਰੇ ਨੌਕਰ ਹੈ ।				
26.	अमृत०—देखे अब कब भाग्य जागता है। मैं तो बहुत जल्दी मचा रहा हूं।	ਅਮ੍ਰਤ०—ਵੇਖੇ ਹੁਣ ਕਦੋਂ ਕਿਸਮਤ ਜਾਗਦਾ ਹੈ । ਮੈਂ ਤਾਂ ਬਹੁਤ ਜਲਦੀ ਮਚਾ ਰਿਹਾ ਹਾਂ ।				
27.	मै तुमसे कोई अनुचित बात नहीं चाहता।	ਮੈ ਤੁਹਾਡੇ ਤੋਂ ਕੋਈ ਅਣ-ਉਚਿਤ ਗੱਲ ਨਹੀਂ ਚਾਹੁੰਦਾ ।				
28.	उनके ज़रा से इशारे पर मैं अपने को निछावर कर सकती हूँ।	ਉਨ੍ਹਾਂ ਦੇ ਜਰਾ ਤੋਂ ਇਸ਼ਾਰੇ ਪਰ ਮੈਂ ਆਪਣੇ ਨੂੰ ਨਿਛਾਵਰ ਕਰ ਸਕਦੀ ਹਾਂ ।				
29.	बिल्लो—दे क्यों नहीं गया।	ਬਿੱਲਾਂ—ਦੇ ਕਿਉਂ ਨਹੀਂ ਗਿਆ ।				
30.	कुछ दिनों से पंडाइन औरचौबाइन आदि ने भी पूर्णा के बनाव-चुनाव पर नाक-भौं चढ़ाना छोड़ दिया था।	ਕੁੱਝ ਦਿਨਾਂ ਤੋਂ ਪੰਡਾਇਨ ਔਰਚੌਬਾਇਨ ਆਦਿ ਨੇ ਵੀ ਦਸਮੀਂ ਦੇ ਰਚਨਾ - ਚੋਣ ਪਰ ਨੱਕ - ਭਰਵੱਟਾ ਚੜਾਨਾ ਛੱਡ ਦਿੱਤਾ ਸੀ ।				
31.	उसने आते ही हुक्म दिया कि भीड़ हटा दी जाय।	ਉਸਨੇ ਆਉਂਦੇ ਹੀ ਹੁਕਮ ਦਿੱਤਾ ਕਿ ਭੀੜ ਹਟਾ ਦਿੱਤੀ ਜਾਵੇ ।				
32.	शादी के चौथे दिन बाद पूर्णा बैठी हुई थी कि एक औरत ने आकर उसके एक बंद लिफ़ाफा दिया।	ਵਿਆਹ ਦੇ ਚੌਥੇ ਦਿਨ ਬਾਅਦ ਦਸਮੀਂ ਬੈਠੀ ਹੋਈ ਸੀ ਕਿ ਇੱਕ ਔਰਤ ਨੇ ਆਕੇ ਉਸਦੇ ਇੱਕ ਬੰਦ ਲਿਫਾਫਾ ਦਿੱਤਾ ।				

33.	चम्मन—चौधरी कह गये हैं किआज इनकेर काम न छोड़ देहों तो टाट बाहर कर दीन जैही।	ਚੰਮਨ ਚੌਧਰੀ— ਕਹਿ ਗਏ ਹਨ ਕਿਆਜ ਇਨਕੇਰ ਕੰਮ ਨਹੀਂ ਛੱਡ ਦੇਹਾਂ ਤਾਂ ਟਾਟ ਬਾਹਰ ਕਰ ਦੀਨ ਜੈਹੀ । ਇੱਕ ਦਿਨ ਵ੍ਰਜਰਾਨੀ ਸੁਵਾਮਾ ਦੇ ਸਿਰਹਾਨੇ ਬੈਠੀ ਪੱਖਾ ਝਲ ਰਹੀ ਸੀ ।				
34.	एक दिन वृजरानी सुवामा के सिरहाने बैठी पंखा झल रही थी।	ਇੱਕ ਦਿਨ ਵ੍ਰਜਰਾਨੀ ਸੁਵਾਮਾ ਦੇ ਸਿਰਹਾਨੇ ਬੈਠੀ ਪੱਖਾ ਝਲ ਰਹੀ ਸੀ ।				
35.	प्रताप-तो भई, एक दिन मुझे भी नेवता दो।	ਪ੍ਰਤਾਪ - ਤਾਂ ਭਈ , ਇੱਕ ਦਿਨ ਮੈਨੂੰ ਵੀ ਨੇਵਤਾ ਦੋ ।				
36.	नवीन मिट्टी की मीठी-मीठी सुगन्ध आ रही है।	ਨਵੀਨ ਮਿੱਟੀ ਦੀ ਮਿੱਠੀ - ਮਿੱਠੀ ਸੁਗੰਧ ਆ ਰਹੀ ਹੈ ।				
37.	सेवती-मैं तो बिन गीत सुने आज तुम्हारा पीछा न छोड़ूंगी।	ਚਿੱਟਾ ਗੁਲਾਬ - ਮੈਂ ਤਾਂ ਬਿਨਾਂ ਗੀਤ ਸੁਣੇ ਅੱਜ ਤੁਹਾਡਾ ਪਿੱਛਾ ਨਹੀਂ ਛੋਡੂੰਗੀ ।				
38.	राजा ने कहा “अच्छी बात है।”	ਰਾਜਾ ਨੇ ਕਿਹਾ “ਚੰਗੀ ਗੱਲ ਹੈ । ”				
39.	तीन दिन बीतने पर बुढ़िया फिर वहाँ पहुँची।	ਤਿੰਨ ਦਿਨ ਗੁਜ਼ਰਨ ਪਰ ਬੁੱਢੀ ਫਿਰ ਉੱਥੇ ਪਹੁੰਚੀ ।				
40.	स्त्री रोने लगी। एक मुसाफ़िर उधर जा रहा था।	ਇਸਤਰੀ ਰੋਣ ਲੱਗੀ । ਇੱਕ ਮੁਸਾਫਰ ਉੱਧਰ ਜਾ ਰਿਹਾ ਸੀ ।				
41.	सेठ अपने जमाई से मिलकर बड़े प्रसन्न हुए और उन्होंने उसे बड़ी अच्छी तरह से घर में रखा।	ਸੇਠ ਆਪਣੇ ਜਵਾਈ ਤੋਂ ਮਿਲਕੇ ਵੱਡੇ ਖੁਸ਼ ਹੋਏ ਅਤੇ ਉਨ੍ਹਾਂਨੇ ਉਸਨੂੰ ਵੱਡੀ ਚੰਗੀ ਤਰ੍ਹਾਂ ਤੋਂ ਘਰ ਵਿੱਚ ਰੱਖਿਆ ।				

42.	लगता है यह दरजी लोभी है। यह हमको खिलाना नहीं चाहता, इसलिए यह सारा नाटक कर रहा है।	ਲੱਗਦਾ ਹੈ ਇਹ ਦਰਜ਼ੀ ਲੋਭੀ ਹੈ । ਇਹ ਸਾਨੂੰ ਖਵਾਉਣਾ ਨਹੀਂ ਚਾਹੁੰਦਾ , ਇਸਲਈ ਇਹ ਸਾਰਾ ਡਰਾਮਾ ਕਰ ਰਿਹਾ ਹੈ ।				
43.	राजा मुझसे डर गया।	ਰਾਜਾ ਮੇਰੇ ਤੋਂ ਡਰ ਗਿਆ ।				
44.	पार्वती को दया आ गई, और उन्होने शंकर से विनती की कि जैसे भी बने, वे गिलहरी को फिर से स्त्री बना दें।	ਪਾਰਬਤੀ ਨੂੰ ਤਰਸ ਆ ਗਈ , ਅਤੇ ਉਨ੍ਹਾਂ ਨੇ ਸ਼ੰਕਰ ਤੋਂ ਪ੍ਰਾਰਥਨਾ ਦੀ ਕਿ ਜਿਵੇਂ ਵੀ ਬਣੇ , ਉਹ ਗਿਲਹਰੀ ਨੂੰ ਫਿਰ ਤੋਂ ਇਸਤਰੀ ਬਣਾ ਦਿਓ ।				
45.	रूस में एक बहुत बड़े लेखक हुए हैं, इतने बड़े कि सारी दुनिया उन्हें जानती है।	ਰੂਸ ਵਿੱਚ ਇੱਕ ਬਹੁਤ ਵੱਡੇ ਲੇਖਕ ਹੋਏ ਹੈ , ਇਨ੍ਹੇ ਵੱਡੇ ਕਿ ਸਾਰੀ ਦੁਨੀਆ ਉਨ੍ਹਾਂਨੂੰ ਜਾਣਦੀ ਹੈ ।				
46.	रवीन्द्र ठाकुर की एक बड़ी ही सीख देने वाली रचना है।	ਰਵੀਂਦਰ ਠਾਕੁਰ ਦੀ ਇੱਕ ਵੱਡੀ ਹੀ ਸੀਖ ਦੇਣ ਵਾਲੀ ਰਚਨਾ ਹੈ ।				
47.	देवदूत चला गया और अगले दिन जब वह लौटा तो उसके हाथ में उन आदमियों की सूची थी	ਦੇਵਦੂਤ ਚਲਾ ਗਿਆ ਅਤੇ ਅਗਲੇ ਦਿਨ ਜਦੋਂ ਉਹ ਪਰਤਿਆ ਤਾਂ ਉਸਦੇ ਹੱਥ ਵਿੱਚ ਉਨ੍ਹਾਂ ਬੰਦੀਆਂ ਦੀ ਸੂਚੀ ਸੀ				
48.	आदमी को भूमि से कितना मोह होता है।	ਆਦਮੀ ਨੂੰ ਭੂਮੀ ਤੋਂ ਕਿੰਨਾ ਮੋਹ ਹੁੰਦਾ ਹੈ ।				
49.	कहने का मतलब यह कि हर आदमी अपनी क्षमता के अनुसार काम करे और जरूरत के अनुसार पाये	ਕਹਿਣ ਦਾ ਮਤਲੱਬ ਇਹ ਕਿ ਹਰ ਆਦਮੀ ਆਪਣੀ ਸਮਰੱਥਾ ਦੇ ਅਨੁਸਾਰ ਕੰਮ ਕਰੇ ਹੋਰ ਜ਼ਰੂਰਤ ਦੇ ਅਨੁਸਾਰ ਪਾਏ				

50.	हम आशा करते हैं कि पाठक इन पुस्तकों को बड़े चाव से पढ़ेंगे, दूसरों की पढ़वाये और इनका भरपूर लाभ लेंगे।	ਅਸੀ ਆਸ ਕਰਦੇ ਹਾਂ ਕਿ ਪਾਠਕ ਇਸ ਕਿਤਾਬਾਂ ਨੂੰ ਵੱਡੇ ਚਾਵ ਨਾਲ ਪੜਾਂਗੇ , ਦੂਸਰੀਆਂ ਦੀ ਪੜਵਾਏ ਅਤੇ ਇਨ੍ਹਾਂ ਦਾ ਭਰਪੂਰ ਮੁਨਾਫ਼ਾ ਲੈਣਗੇ ।				

Accuracy Test - Articles

S.No.	Hindi Sentence	Punjabi Sentence	0	1	2	3
1.	एनीमिया (Anemia) के कारण महिलाओं में थकान, उठने बैठ्ने और खडे होने मे चक्र आना, काम करने का मन न करना, शरीर मे तापमान की कमी, तव्चा मे पीलापन, दिल मे असामान्य धड्कन, सांस लेने मे तकलीफ, सीने मे दर्द, तलवो व हथेलियों मे ठंडापन और लगातार रहने वाला सिर मे दर्द होता है।	ਏਨੀਮਿਆ (Anemia) ਦੇ ਕਾਰਨ ਔਰਤਾਂ ਵਿੱਚ ਥਕਾਣ , ਉੱਠਣ ਬੈਠਣ ਹੋਰ ਖੜੇ ਹੋਣ ਵਿੱਚ ਚੱਕਰ ਆਣਾ , ਕੰਮ ਕਰਣ ਦਾ ਮਨ ਨਹੀਂ ਕਰਣਾ , ਸਰੀਰ ਵਿੱਚ ਤਾਪਮਾਨ ਦੀ ਕਮੀ , ਤਵਚਾ ਵਿੱਚ ਪਿਲੱਤਣ , ਦਿਲ ਵਿੱਚ ਗ਼ੈਰ - ਮਾਮੂਲੀ ਧੜਕਨ , ਸਾਂਸ ਲੈਣ ਵਿੱਚ ਤਕਲੀਫ , ਸੀਨੇ ਵਿੱਚ ਦਰਦ , ਤਲਵਾਂ ਅਤੇ ਹਥੇਲੀਆਂ ਵਿੱਚ ਠੰਡਾਪਨ ਹੋਰ ਲਗਾਤਾਰ ਰਹਿਣ ਵਾਲਾ ਸਿਰ ਵਿੱਚ ਦਰਦ ਹੁੰਦਾ ਹੈ ।				
2.	गर्भावस्था (pregnancy) के दौरान पेट मे तीव्र दर्द और योनी से रक्त स्राव होने लगे तो इसे गंभीरता से लें तथा डाक्टर को तत्काल बताएं।	ਗਰਭਾਵਸਥਾ (pregnancy) ਦੇ ਦੌਰਾਨ ਢਿੱਡ ਵਿੱਚ ਤੀਵਰ ਦਰਦ ਅਤੇ ਯੋਨੀ ਤੋਂ ਰਕਤ ਸਰਾਵ ਹੋਣ ਲੱਗੇ ਤਾਂ ਇਸਨੂੰ ਗੰਭੀਰਤਾ ਤੋਂ ਲਵੇਂ ਅਤੇ ਡਾਕਟਰ ਨੂੰ ਤੱਤਕਾਲ ਦੱਸੀਏ ।				

3.	**सामग्री:**150 ग्राम अरहर दाल, 20 ग्राम मूंगफली, 100 ग्राम गुड, 100 मि. ली. तेल, 15 ग्राम दालचिनी, 3 ग्राम हींग, 5 ग्राम ईमली, 5 ग्राम अदरक, 5 ग्राम नमक, 3 ग्राम हलदी, 5 ग्राम हरीमिर्च, 5 ग्राम, 3 ग्राम करीपत्ता, 3 ग्राम नारियल (कसा हुआ), धनिया बारीक कटी हुइ ।	ਸਾਮਗਰੀ : 150 ਗਰਾਮ ਅਰਹਰ ਦਾਲ , 20 ਗਰਾਮ ਮੂੰਗਫਲੀ , 100 ਗਰਾਮ ਗੁਡ , 100 ਮਿ . ਲਈ . ਤੇਲ , 15 ਗਰਾਮ ਦਾਲਚੀਨੀ , 3 ਗਰਾਮ ਹੀਂਗ , 5 ਗਰਾਮ ਈਮਲੀ , 5 ਗਰਾਮ ਅਦਰਕ , 5 ਗਰਾਮ ਲੂਣ , 3 ਗਰਾਮ ਹਲਦੀ , 5 ਗਰਾਮ ਹਰੀਮੀਰਚ , 5 ਗਰਾਮ , 3 ਗਰਾਮ ਕਰੀਪੱਤਾ , 3 ਗਰਾਮ ਨਾਰੀਅਲ (ਕੱਸਿਆ ਹੋਇਆ) , ਧਨਿਆ ਬਰੀਕ ਕਟੀ ਹੁਇ ।				
4.	**आइसक्रीम के दाग:-**अगर आइसक्रीम के दाग कपडों मे लग जाए तो अमोनिया का घोल डालें ।	ਆਇਸਕਰੀਮ ਦੇ ਦਾਗ : - ਜੇਕਰ ਆਇਸਕਰੀਮ ਦੇ ਦਾਗ ਕਪਡੋਂ ਵਿੱਚ ਲੱਗ ਜਾਵੇ ਤਾਂ ਅਮੋਨਿਆ ਦਾ ਘੋਲ ਪਾਓ ।				
5.	चावल की खीर बनाते समय शकर के साथ थोडा सा नमक मिलाने से खीर का स्वाद और बढ जाता है।	ਚਾਵਲ ਦੀ ਖੀਰ ਬਣਾਉਂਦੇ ਸਮਾਂ ਸ਼ਕਰ ਦੇ ਨਾਲ ਥੋਡਾ ਜਿਹਾ ਲੂਣ ਮਿਲਾਉਣ ਤੋਂ ਖੀਰ ਦਾ ਸਵਾਦ ਹੋਰ ਬਢ ਜਾਂਦਾ ਹੈ ।				
6.	सोने दे जेवर पर पिसी हल्दी लगा कर मसलने से वे चमकने लगते हैं।	ਸੋਣ ਦੇ ਜੇਵਰ ਤੇ ਪਿਸੀ ਹਲਦੀ ਲੱਗਾ ਕੇ ਮਸਲਨੇ ਤੋਂ ਉਹ ਚਮਕਣ ਲੱਗਦੇ ਹੈ ।				
7.	कभी भी इसका स्पिनर खाली न चलने दें।	ਕਦੇ ਵੀ ਇਸਦਾ ਸਪੀਨਰ ਖਾਲੀ ਨਹੀਂ ਚਲਣ ਦਿਓ ।				

8.	आजकल की महिलाएं नौकरीपेशा वाली हैं इसलिए ज्यादा समय घर से बाहर बिताती हैं संयुक्त परिवारों मे या जिन के माता पिता घर पर रहते है उन्हे बच्चों की देखबाल की समस्या नही होती है परन्तु एक्ल परिवारों मे मां के दफतर जाने के बाद बच्चे की देखबाल के लिए कोई नही रहता इसलिए महिलाएं अपने बच्चों के लिय आया का इंतजाम करती है।	ਅੱਜਕੱਲ੍ਹ ਦੀਆਂ ਮਹਿਲਾਵਾਂ ਨੌਕਰੀਪੇਸ਼ਾ ਵਾਲੀਆਂ ਹਨ ਇਸਲਈ ਜਿਆਦਾ ਸਮਾਂ ਘਰ ਤੋਂ ਬਾਹਰ ਗੁਜ਼ਾਰਦੀਆਂ ਹਨ ਸੰਯੁਕਤ ਪਰਵਾਰਾਂ ਵਿੱਚ ਜਾਂ ਜਿਨ੍ਹਾਂ ਦੇ ਮਾਤਾ ਪਿਤਾ ਘਰ ਤੇ ਰਹਤੇ ਹੈ ਉਨ੍ਹਾਂ ਨੂੰ ਬਚਚਾਂ ਦੀ ਦੇਖਬਾਲ ਦੀ ਸਮੱਸਿਆ ਨਹੀ ਹੁੰਦੀ ਹੈ ਪਰ ਏਕਲ ਪਰਵਾਰਾਂ ਵਿੱਚ ਮਾਂ ਦੇ ਦਫਤਰ ਜਾਣ ਦੇ ਬਾਅਦ ਬਚਚੇ ਦੀ ਦੇਖਬਾਲ ਲਈ ਕੋਈ ਨਹੀ ਰਹਿੰਦਾ ਇਸਲਈ ਮਹਿਲਾਵਾਂ ਆਪਣੇ ਬਚਚਾਂ ਦੇ ਲਿਅ ਆਇਆ ਦਾ ਇੰਤਜਾਮ ਕਰਦੀ ਹੈ ।				
9.	अपने जीवन उदेश्य को जानना और उसे प्राप्त करने के लिए दृढ आत्मविशवास रखना, यही सफलता की ओर पहला कदम है ।	ਆਪਣੇ ਜੀਵਨ ਉਦੇਸ਼ਿਅ ਨੂੰ ਜਾਨਣਾ ਅਤੇ ਉਸਨੂੰ ਪ੍ਰਾਪਤ ਕਰਣ ਲਈ ਦ੍ਰਢ ਆਤਮਵੀਸ਼ਵਾਸ ਰੱਖਣਾ , ਇਹੀ ਸਫਲਤਾ ਦੇ ਵੱਲ ਪਹਿਲਾ ਕਦਮ ਹੈ ।				
10.	भक्त का भगवान से, मानव का ईश्वर से, व्यष्टि का समष्टि से, पिण्ड का ब्रह्मण्ड से मिलन को ही योग कहा गया है	ਭਗਤ ਦਾ ਭਗਵਾਨ ਤੋਂ , ਮਨੁੱਖ ਦਾ ਰੱਬ ਤੋਂ , ਸਫਲਤਾ ਦਾ ਸਾਰੇ ਤੋਂ , ਪਿੰਡ ਦਾ ਬਰਹਮੰਡ ਤੋਂ ਮਿਲਣ ਨੂੰ ਹੀ ਯੋਗ ਕਿਹਾ ਗਿਆ ਹੈ				

11.	मिजोरम के एक 64 वर्षीय व्यक्ति जियोन की 50 पत्नियां और 100 बच्चे हैं। मिजोरम से लगभग 80 किमी दूर बक्तवांग गांव का निवासी जियोन अपने परिवार के 180 से अधिक सद्स्यों के साथ पृथ्वी पर सबसे बडे परिवार के मुखिया के रुप मे जाना जाता है।	ਮਿਜੋਰਮ ਦੇ ਇੱਕ 64 ਸਾਲ ਦਾ ਵਿਅਕਤੀ ਜਯੋਨ ਦੀ 50 ਪਤਨੀਆਂ ਅਤੇ 100 ਬੱਚੇ ਹਨ । ਮਿਜੋਰਮ ਤੋਂ ਲੱਗਭੱਗ 80 ਕਿਮੀ ਦੂਰ ਬਕਤਵਾਂਗ ਪਿੰਡ ਦਾ ਨਿਵਾਸੀ ਜਯੋਨ ਆਪਣੇ ਪਰਵਾਰ ਦੇ 180 ਤੋਂ ਜਿਆਦਾ ਸਦਸਯੋਂ ਦੇ ਨਾਲ ਧਰਤੀ ਤੇ ਸਭਤੋਂ ਬਡੇ ਪਰਵਾਰ ਦੇ ਮੁਖੀ ਦੇ ਰੁਪ ਵਿੱਚ ਜਾਣਿਆ ਜਾਂਦਾ ਹੈ ।				
12.	एक छोटा बच्चा दुसरे बच्चे से, अगर दिन को सूर्य न निकला तो क्या होगा दुसरे बच्चे ने जवाब दिया, "बिजली का बिल बढ जाएगा।"	ਇੱਕ ਛੋਟਾ ਬਚਚਾ ਦੂਜੇ ਬਚਚੇ ਤੋਂ , ਜੇਕਰ ਦਿਨ ਨੂੰ ਸੂਰਜ ਨਹੀਂ ਨਿਕਲਿਆ ਤਾਂ ਕੀ ਹੋਵੇਗਾ ਦੂਜੇ ਬਚਚੇ ਨੇ ਜਵਾਬ ਦਿੱਤਾ , ਬਿਜਲੀ ਦਾ ਬਿਲ ਬਢ ਜਾਵੇਗਾ ।				
13.	बीरबल को तम्बाकू खाने की आदत थी लेकिन अकबर (Akbar) न खाते थे एक दिन अकबर ने तम्बाकू के खेत मे गधे को घास खाते देखकर कहा बीरबल ये देखा तम्बाकू कैसी बुरी चीज है, गधे तक इस को नही खाते।	ਬੀਰਬਲ ਨੂੰ ਤੰਬਾਕੂ ਖਾਣ ਦੀ ਆਦਤ ਸੀ ਲੇਕਿਨ ਅਕਬਰ (Akbar) ਨਹੀਂ ਖਾਂਦੇ ਸਨ ਇੱਕ ਦਿਨ ਅਕਬਰ ਨੇ ਤੰਬਾਕੂ ਦੇ ਖੇਤ ਵਿੱਚ ਗਧੇ ਨੂੰ ਘਾਹ ਖਾਂਦੇ ਵੇਖਕੇ ਕਿਹਾ ਬੀਰਬਲ ਇਹ ਵੇਖਿਆ ਤੰਬਾਕੂ ਕਿਵੇਂ ਦੀ ਬੁਰੀ ਚੀਜ ਹੈ , ਗਧੇ ਤੱਕ ਇਸ ਨੂੰ ਨਹੀ ਖਾਂਦੇ ।				

14.	**ये** सब कहने की बातें हैं कि उन को छोड बैठे हैं। जब आंखें चार होती हैं मौहबत आ ही जाती है ।।	ਇਹ ਸਭ ਕਹਿਣ ਦੀਆਂ ਗੱਲਾਂ ਹਨ ਕਿ ਉਨ੍ਹਾਂ ਨੂੰ ਛੋਡ ਬੈਠੇ ਹਨ । ਜਦੋਂ ਅੱਖਾਂ ਚਾਰ ਹੁੰਦੀਆਂ ਹਨ ਮੋਹਬਤ ਆ ਹੀ ਜਾਂਦੀ ਹੈ ।				
15.	नए व आधुनिक डिज़ाइनो के अत्पाद तेज़ी से बाज़ार मे आ रहे है। इस के लिए ज़रुरी है कि आप भी अपने नीतियों मे बदलाव लाएँ और यही न करते रहें, "हम तो इस काम को इसी तरीके से करते आ रहे है और ऐसा ही करेगे"।	ਨਵੇਂ ਅਤੇ ਆਧੁਨਿਕ ਡਿਜ਼ਾਇਨਾਂ ਦੇ ਅਤਪਾਦ ਤੇਜ਼ੀ ਤੋਂ ਬਾਜ਼ਾਰ ਵਿੱਚ ਆ ਰਹੇ ਹੈ । ਇਸ ਲਈ ਜ਼ਰੁਰੀ ਹੈ ਕਿ ਆਪ ਵੀ ਆਪਣੇ ਨੀਤੀਆਂ ਵਿੱਚ ਬਦਲਾਵ ਲਿਆਓ ਅਤੇ ਇਹੀ ਨਹੀਂ ਕਰਦੇ ਰਹੇ , ਅਸੀ ਤਾਂ ਇਸ ਕੰਮ ਨੂੰ ਇਸ ਤਰੀਕੇ ਤੋਂ ਕਰਦੇ ਆ ਰਹੇ ਹੈ ਅਤੇ ਅਜਿਹਾ ਹੀ ਕਰੇਗੇ ।				

Accuracy Test – Official Language Quotes

S.No.	Hindi Sentence	Punjabi Sentence	0	1	2	3
1.	संक्षिप्त नोट नीचे दिया गया है	ਸੰਖਿਪਤ ਨੋਟ ਹੇਠਾਂ ਦਿੱਤਾ ਗਿਆ ਹੈ				
2.	प्रचलित नियमों के अनुसार	ਪ੍ਰਚੱਲਤ ਨਿਯਮਾਂ ਦੇ ਅਨੁਸਾਰ				
3.	पावती पहले ही भेजी जा चुकी है	ਰਸੀਦ ਪਹਿਲਾਂ ਹੀ ਭੇਜੀ ਜਾ ਚੁੱਕੀ ਹੈ				
4.	ऊपर कके अनुसार कार्रवाई की जाए	ਉੱਤੇ ਕਕੇ ਅਨੁਸਾਰ ਕਾੱਰਵਾਈ ਕੀਤੀ ਜਾਵੇ				
5.	मामले में कार्रवाई की जा चुकी है	ਮਾਮਲੇ ਵਿੱਚ ਕਾੱਰਵਾਈ ਕੀਤੀ ਜਾ ਚੁੱਕੀ ਹੈ				
6.	पत्र की एक प्रतिलिपि	ਪੱਤਰ ਦੀ ਇੱਕ ਨਕਲ				
7.	पहले से प्रबंध करना जरूरी है	ਪਹਿਲਾਂ ਤੋਂ ਪ੍ਰਬੰਧ ਕਰਣਾ ਜਰੂਰੀ ਹੈ				
8.	आगे की प्रगति से अवगत कराएं	ਅੱਗੇ ਦੀ ਤਰੱਕੀ ਤੋਂ ਜਾਣੂ ਕਰਾਓ				
9.	कार्य-सूची साथ भेजी जा रही है	ਕਾਰਜ - ਸੂਚੀ ਨਾਲ ਭੇਜੀ ਜਾ ਰਹੀ ਹੈ				
10.	अपील खारिज कर दी गई है	ਅਪੀਲ ਖਾਰਿਜ ਕਰ ਦਿੱਤੀ ਗਈ ਹੈ				
11.	जहां तक संभव हो	ਜਿੱਥੇ ਤੱਕ ਸੰਭਵ ਹੋ				
12.	बिल सही-सही बनाया गया है	ਬਿਲ ਠੀਕ - ਠੀਕ ਬਣਾਇਆ ਗਿਆ ਹੈ				
13.	बजट में व्यवस्था है	ਬਜਟ ਵਿੱਚ ਵਿਵਸਥਾ ਹੈ				
14.	स्वीकार नहीं किया जा सकता	ਸਵੀਕਾਰ ਨਹੀਂ ਕੀਤਾ ਜਾ ਸਕਦਾ				
15.	मामले की जांच चल रही है	ਮਾਮਲੇ ਦੀ ਜਾਂਚ ਚੱਲ ਰਹੀ ਹੈ				

Accuracy Test - Blogs

S.No.	Hindi Sentence	Punjabi Sentence	0	1	2	3
1.	मैं ऐसा इसलिए लिख रहा हूं कि मेरा एक दोस्त जो मेरा ब्लाग पढ़ता है उसने मुझे कहा कि यह नारद का एकाधिकार खत्म होगा!	ਮੈਂ ਅਜਿਹਾ ਇਸਲਈ ਲਿਖ ਰਿਹਾ ਹਾਂ ਕਿ ਮੇਰਾ ਇੱਕ ਦੋਸਤ ਜੋ ਮੇਰਾ ਬਲਾਗ ਪੜ੍ਹਦਾ ਹੈ ਉਸਨੇ ਮੈਨੂੰ ਕਿਹਾ ਕਿ ਇਹ ਨਾਰਦ ਦਾ ਏਕਾਧਿਕਾਰ ਖਤਮ ਹੋਵੇਗਾ				
2.	राजेश, हिन्दी चिट्ठों के ज्यादातर पाठक चिट्ठे लिखने वाले ही हैं ,आपके मित्र की श्रेणी के पाठक फिलहाल कम हैं । अभी तक हिन्दी के गैर चिट्ठेकार पाठक आपका अखबार ही पढ़ते हैं। वैसे पाठक कोई भी हों ,कितने ही क्यों न हों किसी बहस अन्जाम से पहले छोड़ना अन्य माध्यम भी नहीं चाहते।	ਰਾਜੇਸ਼ , ਹਿੰਦੀ ਚਿੱਠੀਆਂ ਦੇ ਜਿਆਦਾਤਰ ਪਾਠਕ ਚਿੱਠੇ ਲਿਖਣ ਵਾਲੇ ਹੀ ਹਨ , ਤੁਹਾਡੇ ਮਿੱਤਰ ਦੀ ਸ਼੍ਰੇਣੀ ਦੇ ਪਾਠਕ ਫਿਲਹਾਲ ਘੱਟ ਹਨ । ਹੁਣੇ ਤੱਕ ਹਿੰਦੀ ਦੇ ਗੈਰ ਚਿੱਠੇਕਾਰ ਪਾਠਕ ਤੁਹਾਡਾ ਅਖਬਾਰ ਹੀ ਪੜ੍ਹਦੇ ਹਨ । ਵੈਸੇ ਪਾਠਕ ਕੋਈ ਵੀ ਹੋਣ , ਕਿੰਨੇ ਹੀ ਕਿਉਂ ਨਹੀਂ ਹੋਣ ਕਿਸੇ ਬਹਿਸ ਅੰਜਾਮ ਤੋਂ ਪਹਿਲਾਂ ਛੱਡਣਾ ਹੋਰ ਮਾਧਿਅਮ ਵੀ ਨਹੀਂ ਚਾਹੁੰਦੇ				
3.	कुछ लोगों को मजा आता उसी विषय को बार-बार फ़ेंटने में तो आप क्या कर सकते हैं! अभिव्यक्ति की स्वतंत्रता है।	ਕੁੱਝ ਲੋਕਾਂ ਨੂੰ ਮਜਾ ਆਉਂਦਾ ਉਸੀ ਵਿਸ਼ਾ ਨੂੰ ਵਾਰ - ਵਾਰ ਫ਼ੇਂਟਨੇ ਵਿੱਚ ਤਾਂ ਤੁਸੀ ਕੀ ਕਰ ਸੱਕਦੇ ਹਨ ! ਪਰਕਾਸ਼ਨ ਦੀ ਅਜਾਦੀ ਹੈ ।				

4.	जहाँ तक मैं समझता हूँ, हिन्दी के चिट्ठे अभी बहुत ही सीमित विषयों पर लिखे जा रहे हैं। ऐसे विषय, जिनमें अधिकांश नेट प्रयोक्ताओं की कोई दिलचस्पी नहीं है। जब तक यह हाल रहेगा, शायद ही हिन्दी चिट्ठों का पाठकवर्ग विकसित हो सके।	ਜਿੱਥੇ ਤੱਕ ਮੈਂ ਸੱਮਝਦਾ ਹਾਂ , ਹਿੰਦੀ ਦੇ ਚਿੱਠੇ ਹੁਣੇ ਬਹੁਤ ਹੀ ਸੀਮਿਤ ਮਜ਼ਮੂਨਾਂ ਪਰ ਲਿਖੇ ਜਾ ਰਹੇ ਹੋ । ਅਜਿਹੇ ਵਿਸ਼ਾ , ਜਿਨ੍ਹਾਂ ਵਿੱਚ ਸਾਰਾ ਨੇਟ ਪ੍ਰਯੋਕਤਾਵਾਂਦੀ ਕੋਈ ਦਿਲਚਸਪੀ ਨਹੀਂ ਹੈ । ਜਦੋਂ ਤੱਕ ਇਹ ਹਾਲ ਰਹੇਗਾ , ਸ਼ਾਇਦ ਹੀ ਹਿੰਦੀ ਚਿੱਠੀਆਂ ਦਾ ਪਾਠਕਵਰਗ ਵਿਕਸਿਤ ਹੋ ਸਕੇ ।				
5.	यह तो सच है, हिन्दी की कई पोस्टों को बिना बैकग्राउंड जाने कोई समझ नहीं सकता।	ਇਹ ਤਾਂ ਸੱਚ ਹੈ , ਹਿੰਦੀ ਦੀ ਕਈ ਪੋਸਟਾਂ ਨੂੰ ਬਿਨਾਂ ਬੈਕਗਰਾਉਂਡ ਜਾਣੇ ਕੋਈ ਸੱਮਝ ਨਹੀਂ ਸਕਦਾ ।				
6.	आप द्वारा उठाया गया प्रश्न बहुत महत्व रखता है।	ਆਪ ਦੁਆਰਾ ਚੁੱਕਿਆ ਗਿਆ ਪ੍ਰਸ਼ਨ ਬਹੁਤ ਮਹੱਤਵ ਰੱਖਦਾ ਹੈ ।				
7.	बहुत सही लिखा है आपने, साधुवाद!!	ਬਹੁਤ ਠੀਕ ਲਿਖਿਆ ਹੈ ਤੁਸੀਂ, ਸਾਧੁਵਾਦ !				
8.	अब मैं आपसे एक निवेदन करना चाहूँगा।	ਹੁਣ ਮੈਂ ਤੁਹਾਨੂੰ ਇੱਕ ਬੇਨਤੀ ਕਰਣਾ ਚਾਹਵਾਂਗਾ ।				
9.	अगर हिन्दी तथा अन्य भारतीय भाषाओं के लिये एक बढ़िया टेक्स्ट एनालिसिस का औजार(साफ्ट्वेयर) बना सकें तो भारतीय भाषाओं का बहुत भला हो।	ਜੇਕਰ ਹਿੰਦੀ ਅਤੇ ਹੋਰ ਭਾਰਤੀਭਾਸ਼ਾਵਾਂ ਲਈ ਇੱਕ ਵਧੀਆ ਟੇਕਸਟ ਏਨਾਲਿਸਿਸ ਦਾ ਔਜਾਰ (ਸਾਫਟਵੇਇਰ) ਬਣਾ ਸਕਣ ਤਾਂ ਭਾਰਤੀ ਭਾਸ਼ਾਵਾਂ ਦਾ ਬਹੁਤ ਭਲਾ ਹੋ ।				

10.	अपील खारिज कर दी गई है	ਅਪੀਲ ਖਾਰਿਜ ਕਰ ਦਿੱਤੀ ਗਈ ਹੈ				
11.	जहां तक संभव हो	ਜਿੱਥੇ ਤੱਕ ਸੰਭਵ ਹੋ				
12.	बिल सही-सही बनाया गया है	ਬਿਲ ਠੀਕ - ਠੀਕ ਬਣਾਇਆ ਗਿਆ ਹੈ				
13.	कृपया इस बारे में गम्भीरता से विचार करें।	ਕ੍ਰਿਪਾ ਇਸ ਬਾਰੇ ਵਿੱਚ ਗੰਭੀਰਤਾ ਤੋਂ ਵਿਚਾਰ ਕਰੋ ।				
14.	आप जैसे लोगो का हमारी भाषा के प्रति प्यार हि सबको उत्साहित रखता है.	ਤੁਸੀ ਜਿਵੇਂ ਲੋਕੋ ਦਾ ਸਾਡੀ ਭਾਸ਼ਾ ਦੇ ਪ੍ਰਤੀ ਪਿਆਰ ਹਿ ਸਾਰਿਆ ਨੂੰ ਉਤਸ਼ਾਹਿਤ ਰੱਖਦਾ ਹੈ .				
15.	बहुत सुंदर प्रयास के लिए बधाई. एक ब्लॉग पोस्ट इस पर कल लिखता हूँ.	ਬਹੁਤ ਸੁੰਦਰ ਕੋਸ਼ਿਸ਼ ਲਈ ਵਧਾਈ . ਇੱਕ ਬਲਾਗ ਪੋਸਟ ਇਸ ਪਰ ਕੱਲ ਲਿਖਦਾ ਹਾਂ .				