“English to Bangla Machine Translator”

Submitted By:

Shanta Islam
ID: 121-35-276

Habibullah Balaly
ID: 121-35-277

Supervisor Name:
MD. Fahad Bin Zamal
Lecturer

Department of Software Engineering,

Daffodil International University

DECEMBER, 2015

APPROVAL

This Report titled "English to Bangla Machine Translator", submitted by Habibullah Balaly, ID No:121-35-277 and Shanta Islam, IDNo:121-35-276 to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Software Engineering and approved as to its style and contents.

BOARD OF EXAMINERS

Dr. Touhid Bhuiyan — **Head**
Associate Professor
Department of Software Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Md. Asraf Ali — **Internal Examiner 1**
Associate Professor
Department of Software Engineering
Faculty of Science & Information Technology
Daffodil International University

Rubaida Easmin — **Internal Examiner 2**
Lecturer
Department of Software Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Nasim Akhtar — **Examiner**
Professor, Head of the Department
Department of Computer Science & Engineering
Faculty of Electrical and Electronic Engineering
Dhaka University of Engineering & Technology, Gazipur

DECLARATION

We hereby declare that we have taken this thesis under the supervision of **MD. Fahad Bin Zamal, Lecturer, Department of Software Engineering, Daffodil International University.** We also declare that neither this thesis nor any part of this has been submitted elsewhere for award of any degree.

Shanta
20.04.16

Shanta Islam
ID: 121-35-276
Batch : 7th
Department of Software Engineering
Faculty of Science & Information Technology
Daffodil International University

Habibullah
20.04.16

Habibullah Balaly
ID: 121-35-277
Batch : 7th
Department of Software Engineering
Faculty of Science & Information Technology
Daffodil International University

Certified by:

20.4.16

MD. Fahad Bin Zamal
Lecturer
Department of Software Engineering
Faculty of Science & Information Technology
Daffodil International University

ACKNOWLEDGEMENT

Firstly, we are thankful to Allah. Thousands of giving thanks to Allah enabled us to finish the job.

We take this opportunity to express our profound gratitude and deep regards to our guide Mr. Md. Fahad Bin Zamal for his exemplary guidance, monitoring and constant encouragement throughout the course of this project . The blessing, help and guidance given by him time to time shall carry us a long way in the journey of life on which we are about to embark. Here are some special people, we are thankful to them for their co-operation of these work, It made this work so special.

Lastly, we are thankful to our parents, brothers, sisters and friends for their constant encouragement.

ABSTRACT

There are many software has been created for translating English language to Bangla language using various parsing techniques. But there is no significant achievement is found in this field yet. A new project has been developed for implementing such an improved translator. In this project, we just describe rule based machine translation system. Though RBMT needs a large dictionary, they can never hope to have a complete coverage of unrestricted text. Particular problem arise for highly productive word classes like proper nouns. We have added a transliteration module to the translation system to transliterate out-of-vocabulary words. Prepositional systems across languages vary to a considerable degree, and this cross-linguistic diversity increases as we move from core, physical senses of preposition into the metaphoric extensions of prepositional meaning. Where English uses prepositions, Bangla typically uses post positions and in some cases attaches inflections to the head nouns. A preposition-handling module is added to the translation system to handle English prepositions during translation. As the inflection of verb plays a very important role in English to Bangla translation, special attentions have been given in decomposing the verb and then extracting the information from the inflection. The forms of a verb in English are only 4 or 5 but in Bangla it is a huge number. We have implemented a module to reform Bangla verb according to tense, person, and voice.

TABLE OF CONTENTS

CONTENT **PAGE**

Chapter 02: MACHINE TRANSLATION

Chapter 03: PROPOSED ARCHITECTURE

Chapter 04: IMPLEMENTATION OF PROPOSED SYSTEM

Chapter 05: SYSTEM DESIGN

Chapter 06: SYSTEM LIFE CYCLE

Chapter 07: TESTING

Chapter 08: FUTURE WORK AND CONCLUSION

List of Tables

List of Figures

CHAPTER 01

INTRODUCTION

1.1 Introduction

Various researches have been done for translating English language into Bangla Language using various parsing techniques. But there is no achievement is found in this field yet. A new project has been developed for implementing such an improved translator. In this work, we describe a Rule-Based Machine Translator (RBMT) system that translates English sentences to Bangla sentences. . We have added a transliteration module to the translation system to transliterate out-of-vocabulary words. Prepositional systems across languages vary to a considerable degree, and this cross-linguistic diversity increases as we move from core, physical senses of preposition into the metaphoric extensions of prepositional meaning. We have implemented a module to reform Bangla verb according to tense, person, and voice.

1.2 History of Machine Translation

The idea of machine translation may be traced back to the 17th century. In 1629, René Descartes proposed a universal language, with equivalent ideas in different tongues sharing one symbol. The first researcher in the field, Yehosha Bar-Hillel, began his research at MIT (1951).

MT on the web started with SYSTRAN Offering free translation of small texts (1996), followed by AltaVista Babel fish, which racked up 500,000 requests a day (1997). More innovations during this time included MOSES, the open-source statistical MT engine (2007), a text/SMS translation service for mobiles in Japan (2008), and a mobile phone with built-in speech-to-speech translation functionality for English, Japanese and Chinese (2009). Recently, Google announced that Google Translate translates roughly enough text to fill 1 million books in one day (2012).

1.3 What is Machine Translation (MT)?

MT is the abbreviation of machine translation. MT is a sub-field of computational linguistics that investigates the use of software to translate text or speech from one natural language to another. MT performs simple substitution of words in one natural language for words in another, but that alone usually cannot produce a good translation of a text because recognition of whole phrases and their closest counterparts in the target language is needed. Solving this problem with corpus

and statistical techniques is a rapidly growing field that is leading to better translations, handling differences in linguistic typology, translation of idioms, and the isolation of anomalies. Current machine translation software often allows for customization by domain or profession (such as weather reports), improving output by limiting the scope of allowable substitutions. This technique is particularly effective in domains where formal or formulaic language is used. It follows that machine translation of government and legal documents more readily produces usable output than conversation or less standardized text. Improved output quality can also be achieved by human intervention: for example, some systems are able to translate more accurately if the user has unambiguously identified which words in the text are names. With the assistance of these techniques, MT has proven useful as a tool to assist human translators and, in a very limited number of cases, can even produce output that can be used as is (e.g., weather reports).The progress and potential of machine translation have been debated much through its history. Since the 1950s, a number of scholars have questioned the possibility of achieving fully automatic machine translation of high quality. Some critics claim that there are in-principle obstacles to automatizing the translation process.

1.4 Based Machine Translation Technology (RBMT)

The rule-based machine translation paradigm includes transfer-based machine translation, Interlingua machine translation and dictionary-based machine translation paradigms. This type of translation is used mostly in the creation of dictionaries and grammar programs. Unlike other methods, RBMT involves more information about the linguistics of the source and target languages, using the morphological and syntactic rules and semantic analysis of both languages. The basic approach involves linking the structure of the input sentence with the structure of the output sentence using a parser and an analyzer for the source language, a generator for the target language, and a transfer lexicon for the actual translation. Adapting to new domains in itself is not that hard, as the core grammar is the same across domains, and the domain-specific adjustment is limited to lexical selection adjustment.

1.4.1 Types of RBMT

There are three different types of rule-based machine translation systems:

- **Direct Systems:** (Dictionary Based Machine Translation) map input to output with basic rules.
- **Transfer RBMT Systems:** (Transfer Based Machine Translation) employ morphological and syntactical analysis.
- **Interlingua RBMT Systems:** (Interlingua) use an abstract meaning.

1.4.2 Basic Principles

The main approach of RBMT systems is based on linking the structure of the given input sentence with the structure of the demanded output sentence, necessarily preserving their unique meaning. The following example can illustrate the general frame of RBMT:

A girl eats an apple. Source Language = English; Demanded Target Language = German

Minimally, to get a German translation of this English sentence one needs:

1. A dictionary that will map each English word to an appropriate German word.
2. Rules representing regular English sentence structure.
3. Rules representing regular German sentence structure.

And finally, we need rules according to which one can relate these two structures together.

Accordingly we can state the following **stages of translation**:

1st: getting basic part-of-speech information of each source word:
a = indef.article; girl = noun; eats = verb; an = indef.article; apple = noun
2nd: getting syntactic information about the verb "to eat":
NP-eat-NP; here: eat – Present Simple, 3rd Person Singular, Active Voice
3rd: parsing the source sentence:
(NP einenApfel) = the object of eat

Often only partial parsing is sufficient to get to the syntactic structure of the source sentence and to map it onto the structure of the target sentence.

4th: translate English words into German
a (category = indef.article) =>ein (category = indef.article)
girl (category = noun) =>Mädchen (category = noun)
eat (category = verb) =>essen (category = verb)
an (category = indef. article) =>ein (category = indef.article)
apple (category = noun) =>Apfel (category = noun)
5th: Mapping dictionary entries into appropriate inflected forms (final **generation**):
A girl eats an apple. =>EinMädchenissteinenApfel.

1.4.3 Components

The RBMT system contains:

- a **SL morphological analyzer** - analyses a source language word and provides the morphological information;
- a **SL parser** - is a syntax analyzer which analyses source language sentences;
- a **translator** - used to translate a source language word into the target language;
- a **TL morphological generator** - works as a generator of appropriate target language words for the given grammatical information;
- a **TL parser** - works as a composer of suitable target language sentences;
- **Several dictionaries** - more specifically a minimum of three dictionaries:

  A *SL dictionary* - needed by the source language morphological analyzer for morphological analysis,
  A *bilingual dictionary* - used by the translator to translate source language words into target language words,
  A *TL dictionary* - needed the target language morphological generator to generate target language words.

The RBMT system makes use of the following:

- a **Source Grammar** for the input language which builds syntactic constructions from input sentences;
- a **Source Lexicon** which captures all of the allowable vocabulary in the domain;
- **Source Mapping Rules** which indicate how syntactic heads and grammatical functions in the source language are mapped onto domain concepts and semantic roles in the interlingua;
- a **Domain Model**/**Ontology** which defines the classes of domain concepts and restricts the fillers of semantic roles for each class;
- **Target Mapping Rules** which indicate how domain concepts and semantic roles in the interlingua are mapped onto syntactic heads and grammatical functions in the target language;
- A **Target Lexicon** which contains appropriate target lexemes for each domain concept;
- A **Target Grammar** for the target language which realizes target syntactic constructions as linearized output sentences.

1.4.4 Advantages

- No bilingual texts are required. This makes it possible to create translation systems for languages that have no texts in common, or even no digitized data whatsoever.
- **Domain independent**: Rules are usually written in a domain independent manner, so the vast majority of rules will "just work" in every domain,
- **No quality ceiling:** Every error can be corrected with a targeted rule, even if the trigger case is extremely rare. This is in contrast to statistical systems where infrequent forms will be washed away by default.
- **Total control:** Because all rules are hand-written, you can easily debug a rule based system to see exactly where a given error enters the system, and why.
- **Reusability**: Because RBMT systems are generally built from a strong source language analysis that is fed to a transfer step and target language generator, the source language analysis and target language generation parts can be shared between multiple translation systems, requiring only the transfer step to be specialized.

1.4.5 Limitations

- Insufficient amount of really good dictionaries. Building new dictionaries is expensive.
- Some linguistic information still needs to be set manually.
- It is hard to deal with rule interactions in big systems, ambiguity, and idiomatic expressions.

Failure adapt to new domains. Although RBMT systems usually provide a mechanism to create new rules and extend and adapt the lexicon, changes are usually very costly and the results, frequently, do not pay off.

1.5 Statistical Machine Translation Technology

Statistical machine translation tries to generate translations using statistical methods based on bilingual text corpora, such as the Canadian Hansard corpus, the English-French record of the Canadian parliament and EUROPARL, the record of the European Parliament. Where such corpora are available, good results can be achieved translating similar texts, but such corpora are still rare for many language pairs. The first statistical machine translation software was CANDIDE from IBM. Google used SYSTRAN for several years, but switched to a statistical translation method in October 2007. In 2005, Google improved its internal translation capabilities by using approximately 200 billion words from United Nations materials to train their system; translation accuracy improved. Google Translate and similar statistical translation programs work by detecting patterns in hundreds of millions of documents that have previously

been translated by humans and making intelligent guesses based on the findings. Generally, the more human-translated documents available in a given language, the more likely it is that the translation will be of good quality. With further development, this may allow statistical machine translation to operate off of a monolingual text corpus.

SMT's biggest downfall includes it being dependent upon huge amounts of parallel texts, its problems with morphology-rich languages (especially with translating *into* such languages), and its inability to correct singleton errors.

1.5.1 Benefits

The most frequently cited benefits of statistical machine translation over rule-based approach are:

- **Better use of resources**
  - There is a great deal of natural language in machine-readable format.
  - Generally, SMT systems are not tailored to any specific pair of languages.
  - Rule-based translation systems require the manual development of linguistic rules, which can be costly, and which often do not generalize to other languages.

- **More natural translations**

Rule-based translation systems are likely to result in Literal translation. While it appears that SMT should avoid this problem and result in natural translations, this is counterbalanced by the fact that using statistical matching to translate rather than a dictionary/grammar rules approach can often result in text that include apparently nonsensical and obvious errors.

1.5.2 Limitations

- Corpus creation can be costly for users with limited resources.
- The results are unexpected. Superficial fluency can be deceiving.
- Statistical machine translation does not work well between languages that have significantly different word orders (e.g. Japanese and European languages).

The benefits are overemphasized for European languages.

1.6 Rule-based Machine Translation Vs Statistical Machine Translation

Rule-based machine translation (RBMT) is generated on the basis of morphological, syntactic, and semantic analysis of both the source and the target languages. Corpus-based machine translation (CBMT) is generated on the analysis of bilingual text corpora. The former belongs to the domain of rationalism and the latter empiricism. Given large-scale and fine-grained linguistic rules, RBMT systems are capable of producing translations with reasonable quality, but constructing the system is very time-consuming and labor-intensive because such linguistic resources need to be hand-crafted, frequently referred to as knowledge acquisition problem. Moreover, it is of great difficulty to correct the input or add new rules to the system to generate a translation. By contrast, however, adding more examples to a CBMT system can improve the system since it is based on the data, though the accumulation and management of the huge bilingual data corpus can also be costly

CHAPTER 02

MACHINE TRANSLATION

2.1 Machine Translation (MT)

Machine Translation is the process of translating text units of source language into a target language by using computers. The term Machine Translation can be defined as "translation from one natural language (source language (SL)) to another language (target language (TL)) using computerized systems, with or without human assistance" **Machine translation**, sometimes referred to by the abbreviation **MT**(not to be confused with **computer-aided translation**, **machine-aided human translation (MAHT)** or **interactive translation**) is a sub-field of computational linguistics that investigates the use of software to translate text or speech from one natural language to another.

On a basic level, MT performs simple substitution of words in one natural language for words in another, but that alone usually cannot produce a good translation of a text because recognition of whole phrases and their closest counterparts in the target language is needed. Solving this problem with corpus and statistical techniques is a rapidly growing field that is leading to better translations, handling differences in linguistic typology, translation of idioms, and the isolation of anomalies.

Machine translation (EBMT) approach was proposed by Makoto Nagao in 1984. Machine translation is based on the idea of analogy. In this approach, the corpus that is used is one that contains texts that have already been translated. Given a sentence that is to be translated, sentences from this corpus are selected that contain similar sub-sentential components. A machine translation system initially would not be able to differentiate between the meanings because syntax does not change. With a large enough ontology as a source of knowledge however, the possible interpretations of ambiguous words in a specific context can be reduced. Other areas of usage for ontologies within NLP include information retrieval, information extraction and text summarization.

2.1.1 Challenges in MT

Machine Translation is a complex and challenging research area because language translation in itself is difficult. While human cognitive processes language interpretation or understanding, and translation on many levels, a machine processes data, linguistic form and structure, not meaning and sense.

Lexical Differences

To fully understand the complexity of designing a MT system, it is necessary to analyze the functions of the human brain during a translation process. When human beings translate they usually start by attempting to decipher the source text on three levels:

- Semantic level: understanding words out of context, as in a dictionary.
- Syntactic level: understanding words in a sentence.
- Pragmatic level: understanding words in situations and context.

Syntactic Differences

Structural differences between languages also present challenges for machine translation. Languages often differ in the basic word order of Subject-Verb-Object. English, Bahasa Malaysia and Chinese are SVO (Subject-Verb-Object) languages, meaning that the verb usually comes in between the subject and object. In contrast, Japanese and Hindi are SOV (Subject-Object-Verb) languages while Arabic is a VSO (Verb-Object-Subject) language. Therefore, developing a system for language pairs with different word orders such as English to Japanese will be more difficult as compared to development for language pairs with the same word orders such as English to Bahasa Malaysia.

Additionally, tenses that exist in one language may not exist in another language. English, for example, has explicit present progressive and present structure; whereas Arabic has only one tense that encompasses both of these English structures.

2.1.2 Translation Process

Machine translation is the process of translating from source language text into the target language. The following diagram shows all the phases involved.

i) Text Input

This is the first phase in the **machine translation process** and is the first module in any MT system. The sentence categories can be classified based on the degree of difficulty of translation. Sentences that have relations, expectations, assumptions, and conditions make the MT system understand very difficult. Speaker's intentions and mental status expressed in the sentences require discourse analysis for interpretation. This is due to the inter-relationship among adjacent sentences. World knowledge and commonsense knowledge could be required for interpreting some sentences.

ii) De-formatting and Re-formatting

This is to make the machine translation process easier and qualitative. The source language text may contain figures, flowcharts, etc. that do not require any translation. Soonly translation portions should be identified. Once the text is translated the target text is to be reformatted after post-editing. Reformatting is to see that the target text also contains the non-translation portion.

iii) Pre-editing and Post editing

The level of pre-editing and post-editing depend on the efficiency of the particular MT system. For some systems segmenting the long sentences into short sentences may be required. Fixing up punctuation marks and blocking material that does not require translation are also done during pre-editing. Post editing is done to make sure that the quality of the translation is up to the mark. Post-editing is unavoidable especially for translation of crucial information such as one for health. Post-editing should continue till the MT systems reach the human-like.

iv) Analysis, Transfer and Generation

Morphological analysis determines the word form such as inflections, tense, number, part of speech, etc. Syntactic analysis determines whether the word is subject or object. Semantic and contextual analysis determines a proper interpretation of a sentence from the results produced by the syntactic analysis. Syntactic and semantic analysis is often executed simultaneously and produces syntactic tree structure and semantic network respectively. This results in internal structure of a sentence. The sentence generation phase is just reverse of the process of analysis.

v) Morphological analysis and generation

Computational morphology deals with recognition, analysis and generation of words. Some of the morphological process is inflection, derivation, affixes and combining forms. Inflection is the most regular and productive morphological process across languages. Inflection alters the form of the word in number, gender, mood, tense, aspect, person, and case. Morphological analyzer gives information concerning morphological properties of the words it analyses.

vi)Syntactic analysis and generation

As words are the foundation of speech and language processing, syntax can considered as the skeleton. Syntactic analysis concerns with how words are grouped into classes called parts-of-speech, how they group their neighbors into phrases, and the way in which words depends on other words in a sentence.

vii) Grammar formalism

Grammar formalism is a framework to explain the basic structure of a language. Researchers propose the following grammar formalisms:

Phrase Structure Grammar (PSG), Dependency Grammar Case Grammar, Systematic Grammar, Montague Grammar, The variants of PSG are, Context Free PSG, Context Sensitive PSG, Augmented Transition Network Grammar (ATN),Definite Clause (DC) Grammar ,Categorical Grammar Lexical Functional Grammar (LFG), Generalized PSG , Head Driven PSG , Tree Adjoining (TAG).

Not all the grammars suit a particular language. PSG, for example, does suit Japanese while dependency grammar does suite. Case grammar is popular as sentence in different languages that express the same contents may have the same case frames.

viii) Parsing and Tagging

Tagging means the identification of linguistic properties of the individual words and parsing is the assessment of the functions of the words in relation to each other.

ix) Semantic and Contextual analysis and Generation

Semantic analysis composes the meaning representations and assigns them the linguistic inputs. The semantic analyzers use lexicon and grammar to create context independent meanings. The source of knowledge consists of meaning of words, meanings associated with grammatical structures, knowledge about the discourse context and commonsense knowledge.

2.1.3 MT Approaches

The aim of machine translation (MT) is to produce high-quality translation automatically. Though this aim is not yet achieved MT research has achieved significant developments. This article intends to explain the design strategies for machine translation. Bilingual MT systems are designed specifically for particular two languages, Tamil and English, for example. Multilingual MT systems support translation of more than two languages. There are three basic approaches as listed below.

1. Direct Translation
2. Interlingua Approach
3. Transfer Approach

Direct translation approach belongs to the first generation of machine translation systems. The Interlingua and transfer approaches are characteristic of the second generation systems.

DIRECT TRANSLATION APPROACH

Direct translation is also called binary translation approach as it is designed for a particular pair of languages. Translation is direct from the source text to the target text. Very little syntactic and semantic analysis is done. Source language analysis is oriented specifically to the production of equivalent target language text. MT systems that use this approach use a bilingual dictionary.

INTERLINGUA APPROACH

An **Interlingua** is knowledge representation formalism. It is language neutral as it is independent of the way particular languages express meaning. Interlingua machine translation techniques produce the Interlingua in such a way that translating from source language to more than one target language is possible. Interlingua approach has to face the challenge of designing an efficient and comprehensive knowledge representation. Moreover complete resolution of all ambiguities in the source language text is required to make this approach possible.

TRANSFER APPROACH

Transfer approach has three stages. The first stage converts source language text into abstract source language oriented representations. The second stage deals with lexical differences between languages and converts the representations resulted from the first stage into corresponding target language representations. Finally the third stage generates the target language text. Transfer systems require dictionaries for source and target languages. These dictionaries should contain morphological, grammatical, and semantic information. A bilingual transfer dictionary is required to relate base source language forms and base target language forms. Both Interlingua and transfer approaches are now known generically as rule-based systems.

2.1.4 MT Classification

Machine Translation is the process of translating text units of source language into a target language by using computers. The term Machine Translation can be defined as "translation from one natural language (source language (SL)) to another language (target language (TL)) uses computerized systems, with or without human assistance" Machine translation systems can be divided in two generations direct translation and indirect systems. First generation translation is known as direct translation. In such systems, translation is done word by word or phrase by phrase. In such systems very minimal linguistic analysis of input text is conducted. Direct translation is also called binary translation.

Second generation machine translation systems are called indirect systems. In such systems the source language structure is analyzed and text is transformed into a logical form. The target

language translation is then generated from the logical form of the text. The transition from direct systems to indirect systems is illustrated in Figure 1.

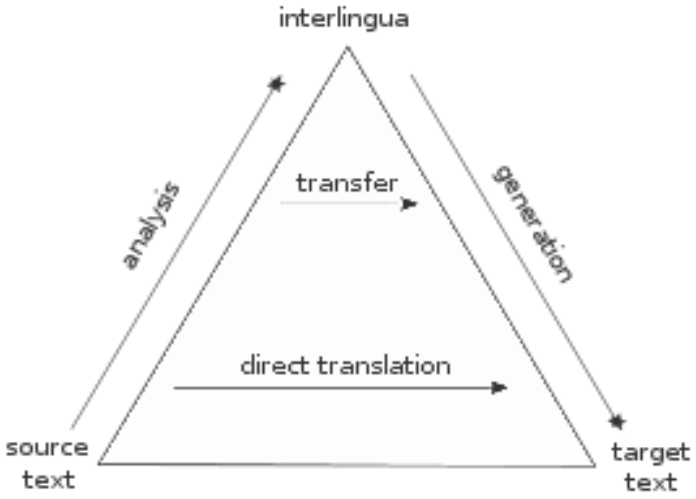


Figure 2.1: Transfer and Interlingua "pyramid" diagram.

2.2 Rule-based Machine Translation (RBMT)

Normally there are 3 stages for Machine Translation: (1) Parsing (2) Transfer and (3) Generation. But here we will propose a transfer architecture which will be used in the syntactic transfer of English to Bengali and which will have less time complexity i.e. most optimal. The proposed transfer architecture has five stages: (1) Tagging (2) Parsing (3) Change CNF parse tree to normal parse tree (4) Transfer of English parse tree to Bengali parse tree. (5) Generation with morphological analysis. In parsing stage we used Cockey-Younger- Kasami (CYK) algorithm which has minimized parsing steps from exponential order (in TopDown or BottomUp parsing) to polynomial order. This algorithm requires that grammar is in Chomsky Normal Form (CNF). But problem with defining grammar in this way is that transfer of English parse tree to Bengali parse tree is not easy. Because elements of the English parse tree that are geographically distant may need to be in close proximity in the Bengali parse tree. So we used an approach where the English parse tree, which is generated via CYK parsing algorithm, is changed into another form of English parse tree, which in turn can be easily transferred into Bengali parse tree.

Machine Translation from English to Bengali has become one of the most important tasks as far as Natural Language Processing of Bengali language is concerned. Our aim in this paper is to

present a transfer architecture, which is not only successful in translating English sentences (Simple) to corresponding Bengali sentences, but also does it in most optimal way. Normally there are 3 stages for Machine Translation: (1) Parsing (2) Transfer and (3) Generation. But here we will propose a transfer architecture which has 5 stages. We say this,

Transfer architecture will take less time because among the different stages parsing is the most critical stage and we have done the parsing in optimal way with a dynamic Parsing algorithm which has minimized parsing steps from exponential order (in TopDown or BottomUp parsing) to polynomial order.This algorithm requires that grammar is in Chomsky Normal Form. But there are some problems associated with defining grammar that way.

NORMAL TRANSFER ARCHITECTURE:

There are three stages in the normal transfer architecture. Here we discuss in short about them

Parsing Stage:

Parsing is the process to form a source language dependent representation (Parse Tree) from a source language sentence by using grammar defined previously. [5]For example if the grammar is defined following way:

1. S=NP+PRIN+NP // He gave a pen
2. S=NP+AP+NP // He has given a pen
3. S=NP+AP+OBJ1+OBJ2// He has given me a pen
4. S=NP+PRIN+OBJ1+OBJ2// He gave me a pen
5. NP=DET+NOUN // a pen
6. NP=PRON // He
7. AP=AUX+PRIN // has given
8. OBJ2=DET+NOUN // a pen
9. PRON=‘He’ | ‘She’
10. DET=‘a’ | ‘the’
11. NOUN=‘pen’ | ‘boy’
12. PRIN= ‘give’ | ‘gave’ | ‘make’
13. AUX=‘has’ | ‘is’
14. OBJ1=‘him’ | ‘her’

Then the parse tree for the sentence “He gave me a pen” is as bellows:

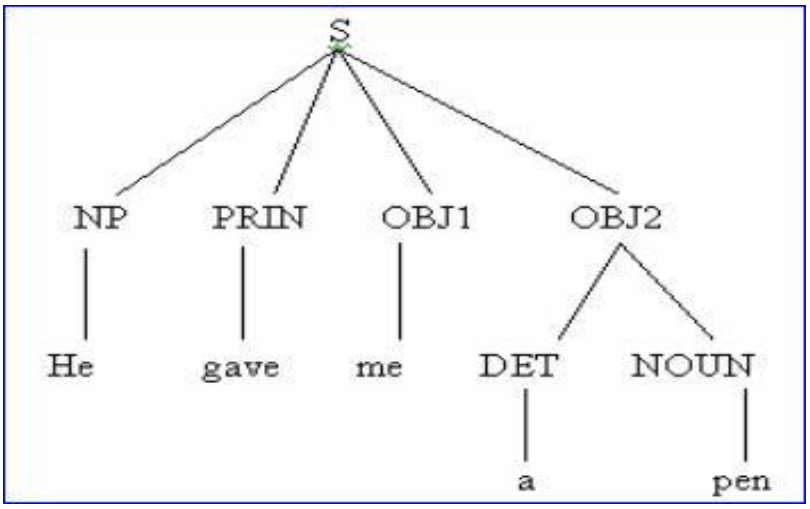


Figure 2.2: English Parse Tree

Parsing can be done in many ways: for example Top Down, Bottom Up, or Dynamic Approach (like chart Parsing or CYK algorithm) .

Transfer Stage: In this stage the English parse tree is changed into their corresponding Bengali parse tree. It uses bilingual dictionary to convert English word to Bengali word.

For example the following is the generated Bengali parse tree from English parse tree shown in figure 3.

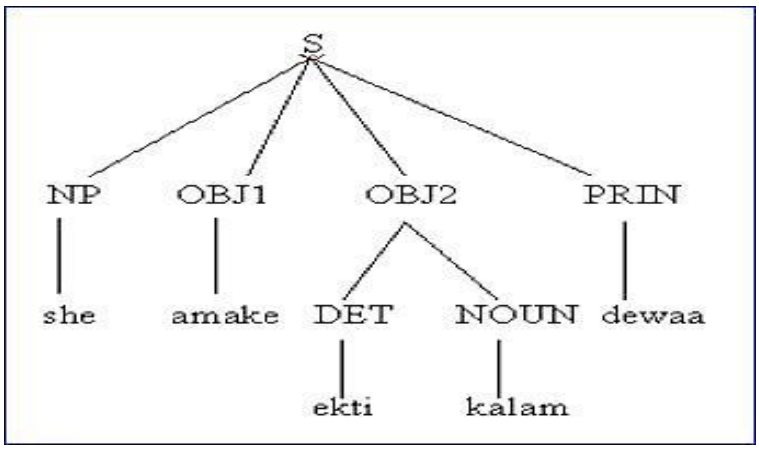


Figure 2.3: Bengali Parse Tree

Syntactic transfer systems rely on mappings between the surface structures of sentences:

A collection of tree-to-tree transformations is applied recursively to the analysis tree of the SL sentence in order to construct a TL analysis tree. The tree-to-tree transformation algorithm is a recursive, non-deterministic, top down process in which one side of the tree-to-tree transfer rules are matched against the input structure, resulting in the structure on the right-hand side.

Problems with CYK parsing:

There are some restrictions on every grammar. For example in Regular Grammar productions are restricted toA1 = a O r A1 = aA2, Where 'A1' and 'A2' are member of non-Terminals and 'a' is a member of Terminals defined in grammar. Chomsky Normal Form where each production is either, A = BC and A = a, Where 'A', 'B', 'C' are members of non-Terminals and 'a' is a member of Terminals defined in grammar. For example here is an English grammar that is defined in Chomsky Normal Form (CNF) form:

S=NP+VP
NP=DET+NOUN
NP= book | money
NP= He
DET= a | the
NOUN= pen | home
VP=VP+PP
VP=AP+NP
VP=PRIN+NP
VP=AP+DOBJ
VP=PRIN+DOBJ
AP=AUX+PRIN
PRIN=give | tell
AUX=has
DOBJ=OBJ1+OBJ2
OBJ1=me | him
OBJ2=DET+NOUN
PP= prepositional phrase

But problem is that grammar defined that way is not suitable for transfer rules. We cannot define a grammar such that transfer structure in SL and TL are markedly different.

Elements of the SL structure that is geographically distant (as they are embedded indifferent branches of the structure) may need to be in close proximity in the TL. We will now show some example in support of this. In the following example, DOBJ and PRIN, these non-terminals can be swapped easily using proper transfer technique. So there should be no problem in transfer at all.

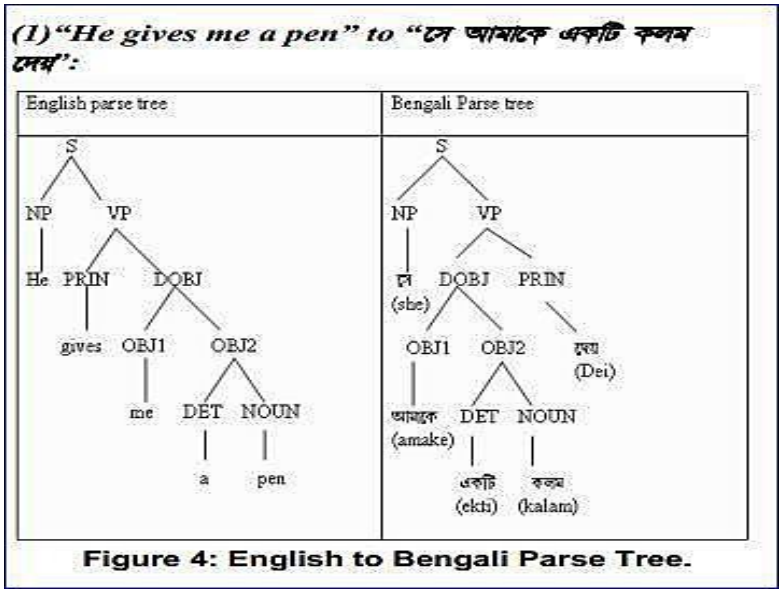


Figure 2.4: English to Bengali parse tree

In the above example, PP and PRN these two non-terminals has to be merged although they are quite distant. The distance factor can be even more evident in the next example.

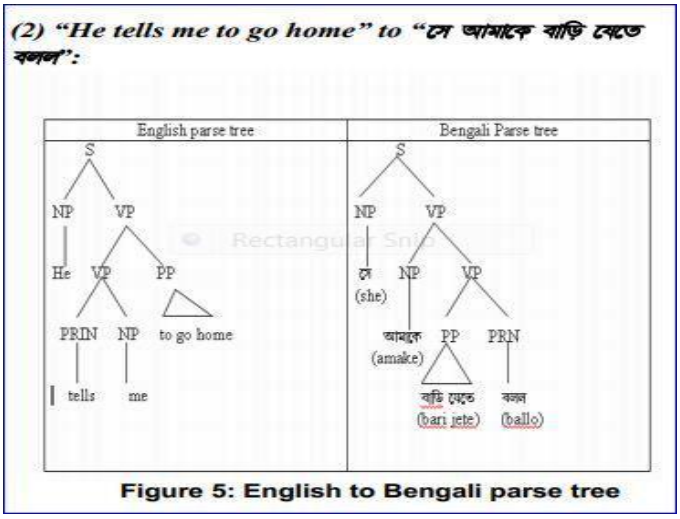


Figure 2.5:English to Bengali parse tree

In the above example, PP and PRN these two non-terminals has to be merged although they are quite distant. The distance factor can be even more evident in the next example.

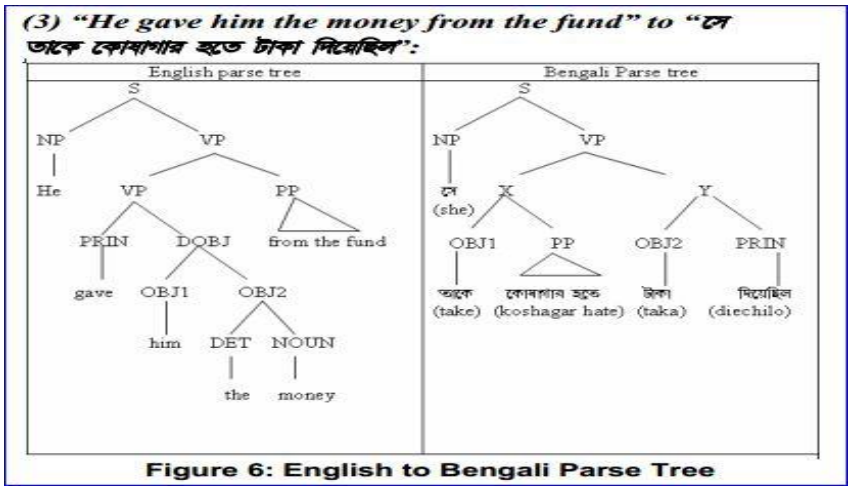


Figure 6: English to Bengali Parse Tree

Figure 2.6:English To Bengali parse tree

In the above example, OBJ1 and PP as well as PRIN and OBJ2 these two sets of non-terminals has to be merged although they are quite distant.

2.2.1 Grammar Description for RBMT

Formal description of the structure of a following language, let there is a rule like this in a certain grammar in English S = NP + VP + NP (for example "He eats rice") its corresponding Bengali rule will be like this S = NP + NP + VP (for example "সেভাতখায়") Algorithm for this rule can be:

For the sentence: "He gave me a pen" if we transfer it as the parse tree shown in Figure 3 we will get the sentence "সেএকটিকলমদেয়া". But here morphological analysis is not taken into consideration. We can easily see that here Subject: 3rd person, singular number Tense: Past Indefinite so,"দেয়া -- →দিয়েছিল". So the final Bengali sentence is "সেএকটিকলমদিয়েছিল"।

Generating Parse Tree (Parsing):

It is one of the complex steps and consists of chunking & parsing. The chunkier returns phrase name based on POS tagging. Producing a full parse tree is a task that builds on the NLP algorithms, which goes in grouping the chunked phrases into a tree diagram that illustrates the structure of the sentence. The full parsing algorithm is implemented by the OpenNLP library. Based on chunk string parse tree is generated. Open NLP can detect following phrases and based on these parse tree is generated.

ADJP	Adjective Phrase
PP	Prepositional Phrase
ADVP	Adverb Phrase
PRT	Particle
CONJP	Conjunction Phrase
SBAR	Clause introduced by a subordinating conjunction
INTJ	Interjection
UCP	Unlike Coordinated Phrase
LST	List marker
VP	Verb Phrase
NP	Noun Phrase

Figure 2.7: Phrase Tags Used By the Open NLP.

Consider following example. English Sentence: He gave me a book. After Chunking: [NP He/PRP] [VP gave/VBD] [NP me/PRP] [NP a/DT book/NN]. After Parsing: (TOP (S (NP(PRP He)) (VP(VBD gave) , (NP(PRP me)) (NP(DT a) (NN book))), (. .))) Fig. 8 shows the parse tree for "He gave me a book."

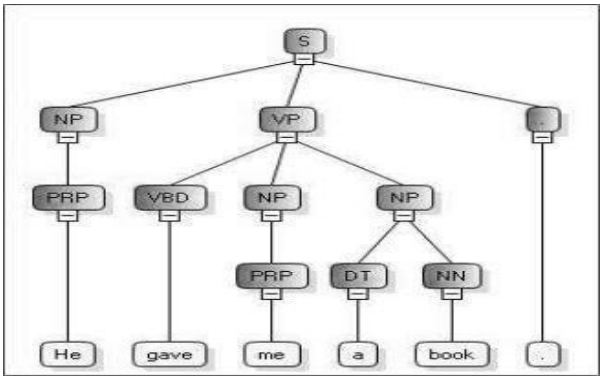


Figure 2.8: English parse tree for "He gave me a book."

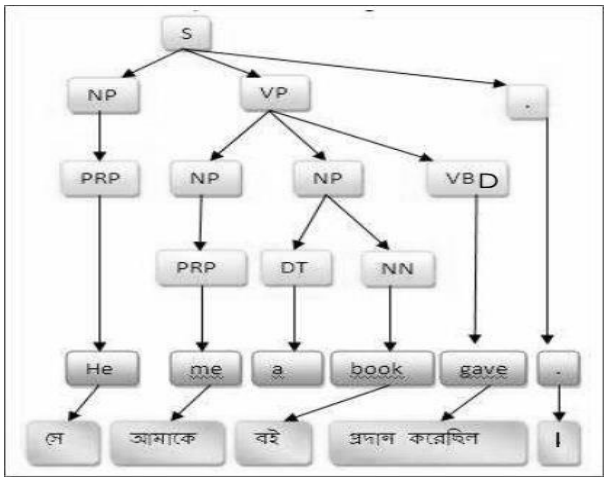


Figure 2.9: Bangla parse tree for "*সে আমাকে বই প্রদান করেছিল।*"

Transfer English to Bangla Parse Tree:

Most of the Machine Translation uses Chomsky Normal Form (CNF) for defining grammar. Here we have used a sorting method for defining Bangla grammar. Let's see the structure of English sentence and Bangla sentence according to CNF. After that we will describe our sorting method that means English to Bangla parse tree transfer algorithm. English grammar in CNF:

1. S = NP + VP + PP + NP
2. S = NP + AP+ NP+PP

And corresponding Bangla grammar in CNF

1. S = NP + NP + PP + VP
2. S = NP + PP + NP + AP

Where,

NP = DET + NOUN + PRONOUN

DET = "a"|"an"|"The"

VP = Auxiliary + Principle

From the above example, for each rule in English there must be a Bangla rule for generating Bangla parse tree. In Context Free Grammar (CFG) we have to define grammar for different sentences. But here we have used a sorting method for generating Bangla parse tree. After generating English parse tree we found whole sentence with phrase name and also tag name. We have defined index value for each phrase/chunk and also defined index value for each tag. Table 1 and 2 shows index value for phrases and tags:

PhraseName	PhraseIndex
NP	0
ADVP	1
ADJP	2
VP	3
..	..

Table 2.1: Phrase Indexing

TagSorting		
TagName	**TagIndex**	**AlterTagIndex**
NNP	0	
PRP$	1	
PRP	1	
DT	1.5	
NN	2	2
WP	2	
JJ	3	3

Table 2.2: Tag Indexing

These index values are used for sorting. Now consider English parse tree corresponding Bangla parse tree which are generated from following chunk string. Chunk string for English Sentence: NP (PRP)+ VP(VBD) + NP(PRP) +NP(DT,NN)……..(1) Chunk string for Bangla sentence:

NP(PRP) + NP(PRP) + NP(DT, NN) + VB……..(2). Here we need to generate equation (2) from equation (1). If we can generate equation (2), then easily we can generate Bangla sentence. Now we will use our sorting method to get Bangla parse tree that means equation (2).Consider following example.

Step 1: Englishsentence→S = He + gave + me + a pen.

Step 2: Tagging and Parsing, S =NP (PRP) + VP (VBD) + NP (PRP) + NP (DT, NN).

Step 3: After Indexing, S = 0(1) + 3 + 0(1) +0(1.5, 2)D.
After getting index value we will perform sorting based on index.

Step 4: After Sorting

BS = 0(1) + 0(1) + 0(1.5, 2) + 3

BS= NP (PRP) + NP (PRP) + NP (DT, NN)+VP (VBD).
BS= He + me + a pen + gave.

Step 5: After dictionary search

BS= "সে + আমাকে+ কলম+প্রদান+করেছিল।"

Consider another example,

Step 1: English Sentence→S = what + is + your +name?

Step 2: Tagging and Parsing
S=NP (WP) + VP (VBZ) + NP (PRP$, NN)?

Step 3: After Indexing, S= 0(2) + 3 + 0(1.2, 2)? After getting index value we will perform sorting based on index. After Sorting we will get.

Step 4: After Sorting, BS = 0(1.2, 2) + 0(2) + 3?
BS = 0(1.2, 2, 2) + 3?
BS=NP (PRP$, NN, WP) + VP (VBZ)?
BS= your, name, what + is?

Step 5: After dictionary search
BS=আপনারনামকি?

By considering above steps we may transfer English sentence to Bangla sentence.
So if we define a grammar this way,

S=NP+PRIN+NP+PP
S=NP+AP+NP+PP
S=NP+AP+OBJ1+OBJ2+PP
S=NP+PRIN+OBJ1+OBJ2+PP

-then there would have been no problems stated above and we can implement the tree to tree transfer with recursive algorithm as we shown in . Here is a table which shows some English rules and their corresponding Bengali rules.

English Rule	**Bangla Rule**
1.S= NP+PRIN+NP +PP 2. S= NP+AP+NP+PP 3.S= NP+AP+OBJ1+OBJ2+PP 4. S=NP+PRIN+OBJ1+OBJ2+PP	1. S=NP+PP+NP+PRIN 2. S =NP+PP+NP+AP 3. S=NP+OBJ1+PP+OBJ2+AP 4. S=NP+OBJ1+PP+OBJ2+PRIN

Table 2.3: English to Bengali Transfer Rules

Here we will discuss how to transfer CNF parse tree to Normal Parse tree. The following figure is the parse tree transfer for the sentence “He gives me a pen”

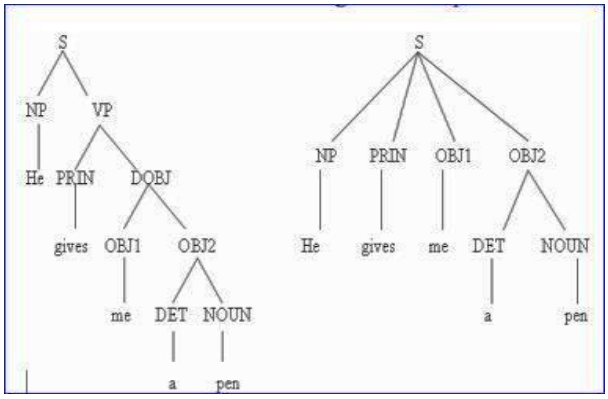


Figure 2.10: Normal Parse Tree

Currently our system is working for English to Bengali Translation. Our current system can translate simple sentences which are given in the knowledge base. We also defined a way to translate a complex sentence using sub-sentential EBMT. So while separating the chunk in the machine translation process we can use our knowledge base for translating the separated small chunks of the sentence. As this system can add more rules in the knowledge base, eventually it can be used for general purpose English to Bengali machine translation.

Outputs:

Our system can now do following types of English to Bengali translation: From this example we can say that EBMT is better as we can improve our system by adding more rules.

English	**Previous**	**Current**
I am a man	আমিএকটিমানুষ	আমিএকজনমানুষ
I have a cat	আমারআছেএকটিবিড়াল	আমারএকটিবিড়ালআছে
I have to love	আমারভালবাসতেআছে	আমাকেভালবাসতেহবে

Now we can see that, I have a cat = PRP + VBP + NP; I have to love = PRP + VBP + S If we add a new rule for “PRP + VBP + S” in our knowledge base then we can also translate “I have to love” properly. In this way we can update the system.

2.3 Statistical Machine Translation

Statistical machine translation (**SMT**) is a translation paradigm where translations are generated on the basis of statistical models whose parameters are derived from the analysis of bilingual text corpora. The statistical approach contrasts with the rule-based approaches to machine translation as well as with translation. The first ideas of statistical machine translation were introduced by Warren Weaver in 1949. Statistical machine translation was re-introduced in 1993 by researchers at IBM's Center and has contributed to the significant resurgence in interest in machine translation in recent years. Nowadays it is by far the most widely studied machine translation method.

2.4 Hybrid Machine Translation

Hybrid machine translation (HMT) leverages the strengths of statistical and rule-based translation methodologies. Several MT organizations (such as Asia Online, LinguaSystran, and Polytechnic University of Valencia) claim a hybrid approach that uses both rules and statistics. The approaches differ in a number of ways:

Rules post-processed by statistics: Translations are performed using a rules based engine. Statistics are then used in an attempt to adjust/correct the output from the rules engine.

Statistics guided by rules: Rules are used to pre-process data in an attempt to better guide the statistical engine. Rules are also used to post-process the statistical output to perform functions such as normalization. This approach has a lot more power, flexibility and control when translating.

CHAPTER 03

PROPOSED ARCHITECTURE

Proposed Architecture of E2BMT

At first, the system divides paragraph into sentences and a sentence into words. The punctuation of the end of every sentence is placed in the stack and the place of punctuation is replaced by a space. After that the system finds out the subject of the sentence. In this process if the sentence starts with verb it takes the subject as 2nd person. But if the subject is not found, the system checks whether gerund is used as a subject. Here, corresponding parts of speech of the every word of the sentence is found out from the database named Ovidhan. For every word, the system checks whether a word is used as a subject or object or both manner or none. After that, the system checks that given sentence belongs to what kinds of sentence or tense. At last, Bangla meaning of every word of the sentence is found out from the Ovidhan and placed according to tense, type of sentence, number of verb, person of the subject. In case of setting up the Bangla meaning of the verb, person of the subject is checked and the form of verb is changed according to the person. Besides, in this process, it is checked whether the verb is in passive form. After all, according to voice, tense and object, the correct form of verb is placed. The architectural block diagram of the proposed machine translation system is depicted in the following figure

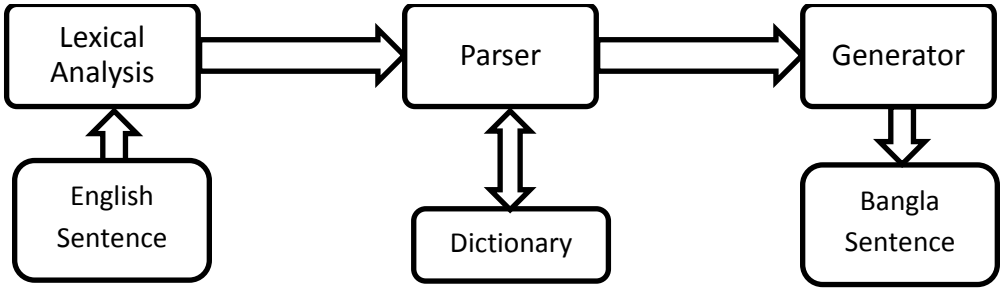


Figure 3.11: Block diagram of the proposed machine translation system.

The system works in three steps- lexical analyzing, parsing and Bangla sentence generation.

3.1 Lexical Analysis

The lexical analyzer is the first processing phase of this translator. It reads the English sentence, separates the words and populates it with lexical information. The following strategies are used for word morphology:

 Read the whole word and search it in the Ovidhan, if it is found proceed to the next word.

 If the word is not in the Ovidhan, check it for the proper noun as described earlier, then if it is a proper noun proceeds for the next word.

 Discard the last letter from the word and apply above two steps to the remaining words, if any one succeeds check whether the discarded letters is a valid suffix for the recognized word. If it succeeds it moves to the next word.

 Repeat steps (iii) each time discarding the one letter from the end of the word until the word is recognized or the length of the word becomes zero, declare the word is not in the dictionary or invalid.

For the sentence "Joy is drinking milk.", after lexical analysis phase the result will be as follows-

Joy	Noun
is	Aux
Drinking	Verb
Milk	Noun

3.2 Parsing

The next phase is parsing. The parser uses a rule based top down parser to parse the input sentence for syntactic correctness. For each word, the system finds the corresponding information of the word from the dictionary. It is expected that the parser report any syntax error in an intelligible fashion. And it should also recover from commonly occurs error so that it can continue processing the remainder of its input.

The top-down parse for the sentence "Joy is drinking milk", would be given by

S→NP VP
S→N VP
S→Joy VP
S→Joy AUX V OBJ
S→Joy is V OBJ

S→Joy is drinking OBJ
S→Joy is drinking NP
S→Joy is drinking N
S→Joy is drinking milk.

The top-down parse tree for the above derivation is shown in Figure 12

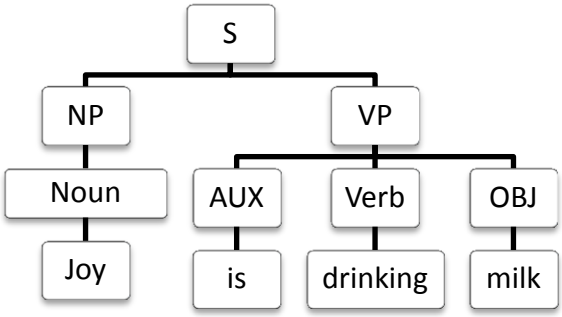


Figure 3.12: Parse tree for "Joy is drinking milk"

3.3 Translation

The next step is to perform translation. The generator produces the Bangla sentence from the parser output. In this step, the system fetches the Bangla meaning for each token from the dictionary .The Bangla meaning for each noun or pronoun or adjective or adverb is directly replaced. But for verb, preposition and article, artificial intelligence is applied for the appropriate Bangla meaning. In this system, we use the following artificial intelligence.

3.3.1 Preposition

Table explaining the relation between the Bangla "Bivokti" and English preposition is given below:

বিভক্তি	**Preposition**
কর্তায়১মা	
কর্মে২য়া কে,রে,প্রতি	To
করণে৩য়া দ্বারা,দিয়ে,কর্তৃক	by (a doer), with (an instrument)
নিমিত্তে৪র্থী জন্য	For

অপাদানে৫মী হইতে,থেকে,চেয়ে	from, than
সম্বন্ধে৬ষ্ঠী র,এর	Of
অধিকরণে৭মী এ,য়,তে	in, at, to

Table 3.4: Preposition Mapping

3.3.2 Article

An article is a word placed before a noun and pre-modifies the noun. The meaning of the noun modifies as follows:

1. If the article is “a” or “an” then place “একজন/একটি” before the meaning of the noun.

Example: I am a doctor.
Meaning: আমিএকজনডাক্তার

2. If the article is “the” then place “টি” after the meaning of the noun.

Example: The boy is drawing.
Meaning: বালকটিআঁকছে

3.3.3 Verb

The table showing the modification that was applied by the translator for different verb is present below:

Forms of Verbs	Tense	For 1st Person	For 2nd Person	For 3rd Person
End With "ওয়া"		"ওয়া" is changed to		
	Present Indefinite	" ই"	"ও"	"ন"
	Present Continuous	"চ্ছি "	"চ্ছ"	চ্ছে"
	Present Perfect	"এছি"	"এছ"	"এছে"
	Past	"লাম"	"লে"	"ল"
	Future	"ব"	"বে"	"বে"
		"া" Is changed to		
End With "া"	Present Indefinite	"া"	Subtract "া"	Subtract "া"
End With "া"		Add "ি"		Add "ে"
	Present Continuous	"ছি"	"ছ"	"ছে"
	Present Perfect	"এছি"	"এছ"	"এছে"
	Past	"লাম"	"লে"	"ল"
	Future	"ব"	"বে"	"বে"
End with নো		" নো" Is changed to		
	Present Indefinite	"ই"	"ও"	"ন"
	Present Continuous	"চ্ছি"	"চ্ছ"	"চ্ছে"
	Present Perfect	"য়েছি"	"য়েছ"	"য়েছে"
	Past	"য়েছিলাম"	"য়েছিলে"	"য়েছিল"
	Future	"ব"	"বে"	"বে"

Table 3.5: Verb Mapping

3.3.4 Algorithm

The main steps of our proposed system is given in the following Algorithm

E2B_TRANSLATION ( )

{

1. Take an input English sentence from user.

2. Split the sentence into words.

3. for each word do

4. {

5. From dictionary find the appropriate meaning, parts of speech, type (whether subject, object or none), person (if subject or object)

6. }

7. From above find the subject, the verb and the object of the sentence.

8. Determine the structure of the sentence by the placement of subject, object, verb, person of subject etc.

9. Put the word meanings as an order by which a corresponding Bangla structure of the English sentence formed.

10. Show the Bangla sentence at the user interface in any Bangla font.

}

3.3.5 Code Map ofthe Proposed System

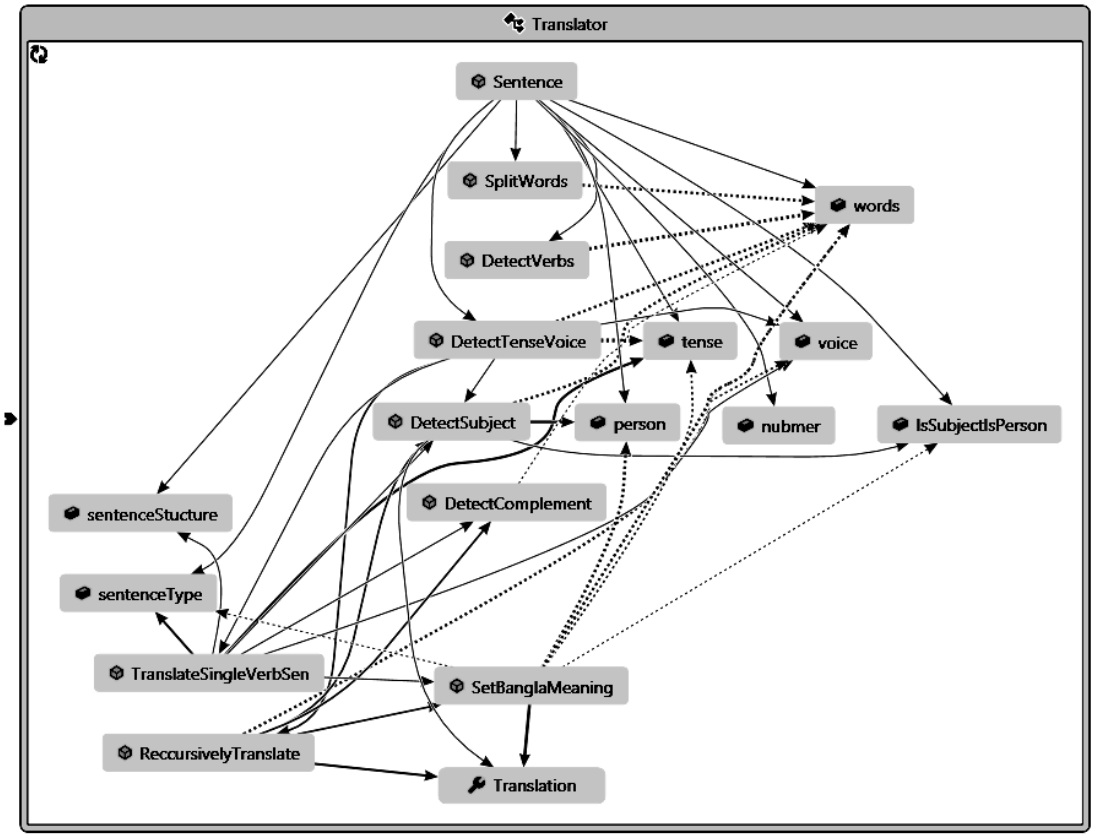


Figure 3.13: Code map Of the proposed system

CHAPTER 04

IMPLEMENTATION OF PROPOSED SYSTEM

4.1 Proposed system

- ✓ A Rule-Based Machine Translator (RBMT) system that translates English sentences to Bangla sentences.
- ✓ We have added a transliteration module to the translation system to transliterate out-of-vocabulary words
- ✓ Prepositional systems across languages vary to a considerable degree.
- ✓ Physical senses of preposition into the metaphoric extensions of prepositional meaning. Where English uses prepositions.

We have implemented a module to reform Bangla verb according to tense, person, and voice.

4.2 Feasibility Study

When we try to find out the Bangla meaning of English word, we faced many problems. We don't get proper or suitable meaning. We think that's a big problem for us. Even the most famous google also failed in this work. They can't give us suitable translation for any word, sentences and paragraph. Then we take a decision to make a system for this. After selecting the system we have collected requirements and research many in this area. We studied and analyzed the requirements. We noticed that, if peoples get the most successful result, then it will be helpful for them and us. After all of these we think this system will be a great system for all of us.

4.3 Review Works

We have reviewed all of our works properly which we have covered in our system. We have checked out all of the information that we took from other source to make our system good for all. We now think, we have done our best to make our system work properly.

There are some merits and demerits of our system have and they are the following:

Merits	Demerits
1. **Easy to new coder** 2. **Take not much time to execute** 3. **System can run in every platform** 4. **Easy to use a user**	1. **As system deals with a huge amount of data it must have a proper backup.**

4.4 Tools

- HTML
- CSS
- C# Java Script
- Asp.net
- Entity framework
- Sql server
- Linq

4.5 Objectives of this project

- ✓ To append new words in the dictionary.
- ✓ System can detect syntactic errors.
- ✓ To translate both active voice and passive voice sentence with some exception.
- ✓ System can translate all kinds of sentences.
- ✓ User can enrich the stock of words.

System follows all grammatical rules and use Unicode for translation.

4.6 Raw Requirement

Functional Requirements

1. This will be a web based system.
2. User can write anything in English like paragraph, sentence, and word.
3. System will translate from what user write in English to Bangla.
4. System will also translate non English word like (name, location, etc.) to Bangla using Unicode.
5. System will also show result in sentence by sentence.
6. System will also show that the inputted sentence are valid or not
7. System will also show the sentence type
8. System will also show the tense name of the inputted sentence
9. System will also identify that the inputted sentence is negative or positive
10. System will also find the subject verb and complement.
11. System will also show result in word by word.
12. System will also show the parts of speech of the word
13. System will also show Bangla meaning of the word.
14. User should be register
15. User can login as admin
16. Admin can teach the system by inputting new logic of grammar
17. Admin also set the exception of grammar.
18. Admin can add new word and Bangla meaning of word.

Non Functional Requirement

1. As the system deals with a huge amount of data it must have a proper backup.
2. The most important feature of the System is the search option so this should work efficiently.
3. The interface should not be complicated and it should be easy to be followed by the user.
4. System will run in every web platform.

4.7 Software Requirement Specification

Functional Requirement

Requirement ID	Requirement Title	Priority
FR-01	This will be a web based system.	High
FR-02	User can write anything in English like paragraph, sentence, and word.	High
FR-03	System will translate from what user write in English to Bangla.	High
FR-04	System will also translate non English word like (name, location, etc.) to Bangla using Unicode	High
FR-05	System will also show result in sentence by sentence.	High
FR-06	System will also show that the inputted sentence are valid or not	High
FR-07	System will also show the sentence type	Medium
FR-08	System will also show the tense name of the inputted sentence	Medium
FR-09	System will also identify that the inputted sentence is negative or positive	Medium
FR-10	System will also find the subject verb and complement	High
FR-11	System will also show result in word by word	Medium
FR-12	System will also show the parts of speech of the word	Medium
FR-13	System will also show Bangla meaning of the word	High
FR-14	User should be register	High
FR-15	User can login as admin	High
FR-16	Admin can teach the system by inputting new logic of grammar	High
FR-17	Admin also set the exception of grammar	High
FR-18	Admin can add new word and Bangla meaning of word	High

Non Functional Requirement

Requirement ID	Requirement Title	Priority
NFR-01	As the system deals with a huge amount of data it must have a proper backup	High
NFR-02	The most important feature of the System is the search option so this should work efficiently	High
NFR-03	The interface should not be complicated and it should be easy to be followed by the user	Medium
NFR-04	System will run in every web platform	Medium

4.8 Step by Step Execution Procedure

Phase 1: Source Text is split into words.

Phase 2: Retrieve information about each word from the dictionary.

Phase 3: Separate words according to punctuation mark and group words between two punctuation marks.

Phase 4: Detect verbs. Is it Auxiliary, principal, infinitives, gerund etc.?

Phase 5: Determine whether the groups of words make a valid sentence or not.

Phase 6: If the sentence is a valid sentence then apply the rules to the words to get translation of that sentence. Otherwise it is translated as a phrase.

Phase 7: Repeat Phase 4, 5 and 6 for every group of words.

4.9 Examples

Example1: Suppose, the source text is-

"We have come to university."

Phase 1: Source Text is split into words.

0	1	2	3	4	5
We	Have	Come	To	university	.

Phase 2: Retrieve information about each word from the dictionary.

Index	English	POS	Type	Bangla	RoleInSentence
0	we	Pronoun	p,1,p,s	আমরা	
1	have	Verb	axlr,t,f,,,2,		
2	come	Verb	prp	আসা	
3	To	preposition	?	থেকে	
4	university	Noun		বিশ্ববিদ্যালয়	
5	.				

Phase 3: Separate words according to punctuation mark and group words between two punctuation marks.

Index	English	POS	Type	Bangla	RoleInSentence
0	we	pronoun	p,1,p,s	আমরা	
1	have	Verb	axlr,t,f,,,2,		
2	come	Verb	prp	আসা	
3	To	preposition	?	থেকে	
4	university	Noun		বিশ্ববিদ্যালয়	
5	.				EOS

Phase 4: Detect verbs

Index	English	POS	Type	Bangla	RoleInSentence
0	we	pronoun	p,1,p,s	আমরা	
1	have	Verb	axlr,t,f,,,2,		AxlrVerb
2	come	Verb	prp	আসা	MainVerb
3	To	preposition	?	থেকে	
4	university	Noun		বিশ্ববিদ্যালয়	
5	.				EOS

Phase 5: Since the sentence has appropriate forms of verb so it is a valid English sentence.

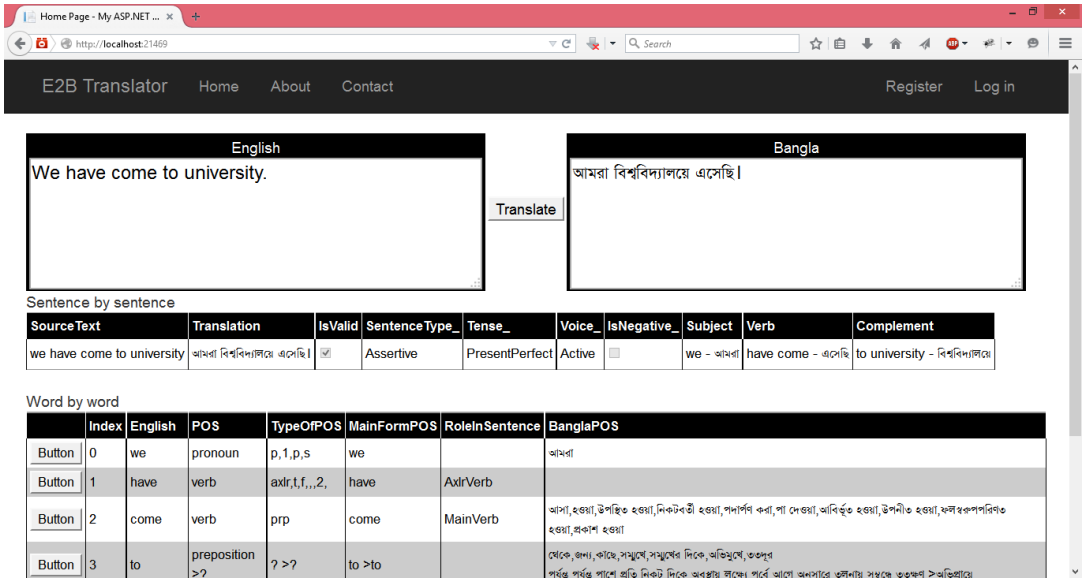
Phase 6: Detect necessary information to translate.

Type	Tense	Voice	IsNegative	Subject	Verb	Complement
Assertive	PresentPerfect	Active	☐	we - আমরা	have come – এসেছি	to university – বিশ্ববিদ্যালয়ে

Result:

SourceText	Translation
we have come to university	আমরাবিশ্ববিদ্যালয়েএসেছি।

Screenshot



Example 2: “He said that he was going to station for buy a ticket.”

Phase 1: Source Text is split into words.

0	1	2	3	4	5	6	7	8	9	10	11	12
he	Said	That	he	Was	going	to	station	for	buy	a	ticket	.

Phase 2: Retrieve information about each word from the dictionary.

Index	English	POS	TypeOfPOS	BanglaPOS	RoleInSentence
0	He	Pronoun	p,3,s,s	সে	
1	said	verb >adjective	pp >?	বলা>কথিত,ব্যক্ত	
2	that	conjunction >pronoun >adjective >adverb	? >? >? >? >?	যে>এটা, >সেই, >তা, >তত,ততদূর	
3	He	Pronoun	p,3,s,s	সে,	
4	was	Verb	axlr,t,f,,s,5,4		
5	going	verb >adjective >noun	ing >? >? >?	যাওয়া,চলা>চালু,সক্রিয়,চলন্ত, >চলন,গতি,যাত্রা,গমন	
6	To	preposition	?	থেকে	
7	station	noun >verb	?,?,? >b	স্টেশন,স্থান>স্থাপনকরা	
8	For	preposition	?	জন্য,নিমিত্তে	
9	buy	verb >noun	b >?,?,?	কেনা,ক্রয়করা>কেনা	
10	A	Article	?	একটি	
11	ticket	noun >?	?	টিকিট	
12	.				

Phase 3: Separate words according to punctuation mark and group words between two punctuation marks.

Index	English	POS	TypeOfPOS	BanglaPOS	RoleInSentence
0	He	pronoun	p,3,s,s	সে	
1	said	verb >adjective	pp >?	বলা>কথিত,ব্যক্ত	
2	that	conjunction >pronoun >adjective >adverb	? >? >? >? >?	যে>এটা, >সেই, >তা, >তত,ততদূর	
3	He	pronoun	p,3,s,s	সে,	
4	was	Verb	axlr,t,f,,s,5,4		
5	going	verb >adjective >noun	ing >? >? >?	যাওয়া,চলা>চালু,সক্রিয়,চলন্ত, >চলন,গতি,যাত্রা,গমন	
6	To	preposition	?	থেকে	
7	station	noun >verb	?,?,? >b	স্টেশন,স্থান>স্থাপনকরা	
8	for	preposition	?	জন্য,নিমিত্তে	
9	buy	verb >noun	b >?,?,?	কেনা,ক্রয়করা>কেনা	
10	A	Article	?	একটি	
11	ticket	noun >?	?	টিকিট	
12	.				EOS

Phase 4: Detect verbs

Index	English	POS	TypeOfPOS	BanglaPOS	RoleInSentence
0	He	pronoun	p,3,s,s	সে	
1	said	verb >adjective	pp >?	বলা>কথিত,ব্যক্ত	MainVerb
2	that	conjunction >pronoun >adjective >adverb	? >? >? >? >?	যে>এটা, >সেই, >তা, >তত,ততদূর	Joiner
3	He	pronoun	p,3,s,s	সে,	
4	was	Verb	axlr,t,f,,s,5,4		AxlrVerb
5	going	verb >adjective >noun	ing >? >? >?	যাওয়া,চলা>চালু,সক্রিয়,চলন্ত, >চলন,গতি,যাত্রা,গমন	MainVerb
6	To	preposition	?	থেকে	
7	station	noun >verb	?,?,? >b	স্টেশন,স্থান>স্থাপনকরা	
8	for	preposition	?	জন্য,নিমিত্তে	
9	buy	verb >noun	b >?,?,?	কেনা,ক্রয়করা>কেনা	
10	A	Article	?	একটি	
11	ticket	noun >?	?	টিকিট	
12	.				EOS

Phase 5: Since the sentence has appropriate forms of verb so it is a valid English sentence.

Phase 6: Detect necessary information to translate. The sentence has two parts

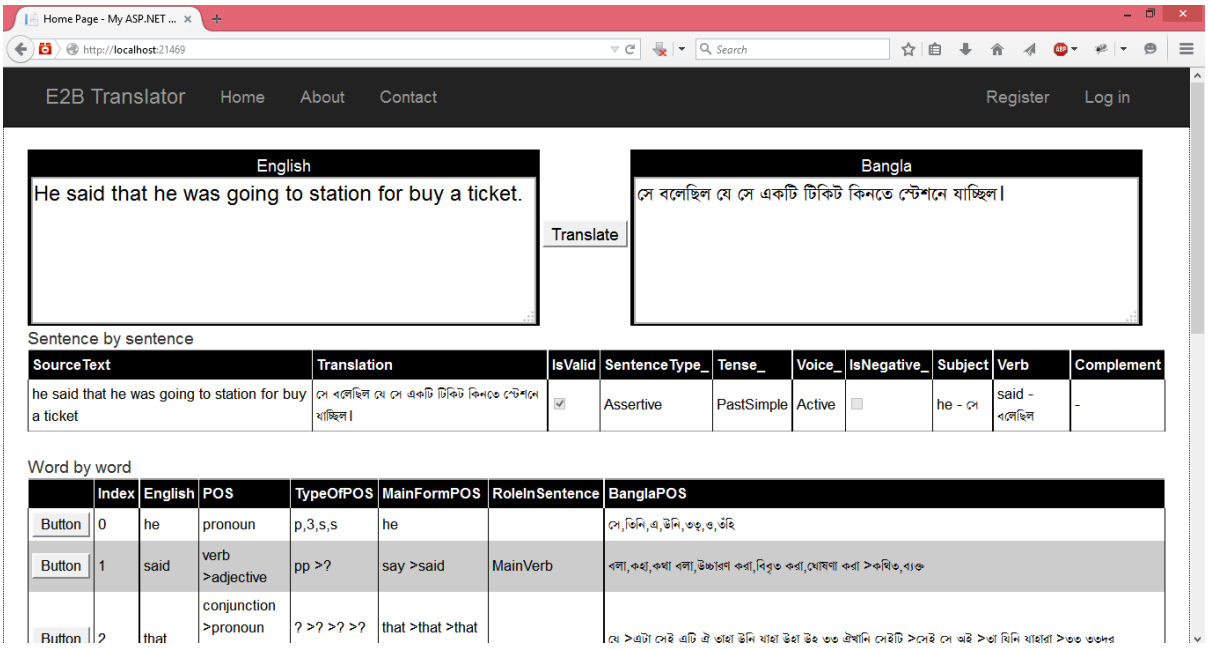
Part 1:

Type	Tense	Voice	IsNegative	Subject	Verb	Complement
Assertive	Past Continuous	Active	☐	he - সে	was going - যাচ্ছিল	to station for buy a ticket- একটিটিকিটকিনতেস্টেশনে

Result:

SourceText	Translation
He said that he was going to station for buy a ticket.	সেবলেছিলযেসেএকটিটিকিটকিনতেস্টেশনেযাচ্ছিল।

Screenshot:



CHAPTER 05

SYSTEM DESIGN

5.1 Use Case of Proposed System

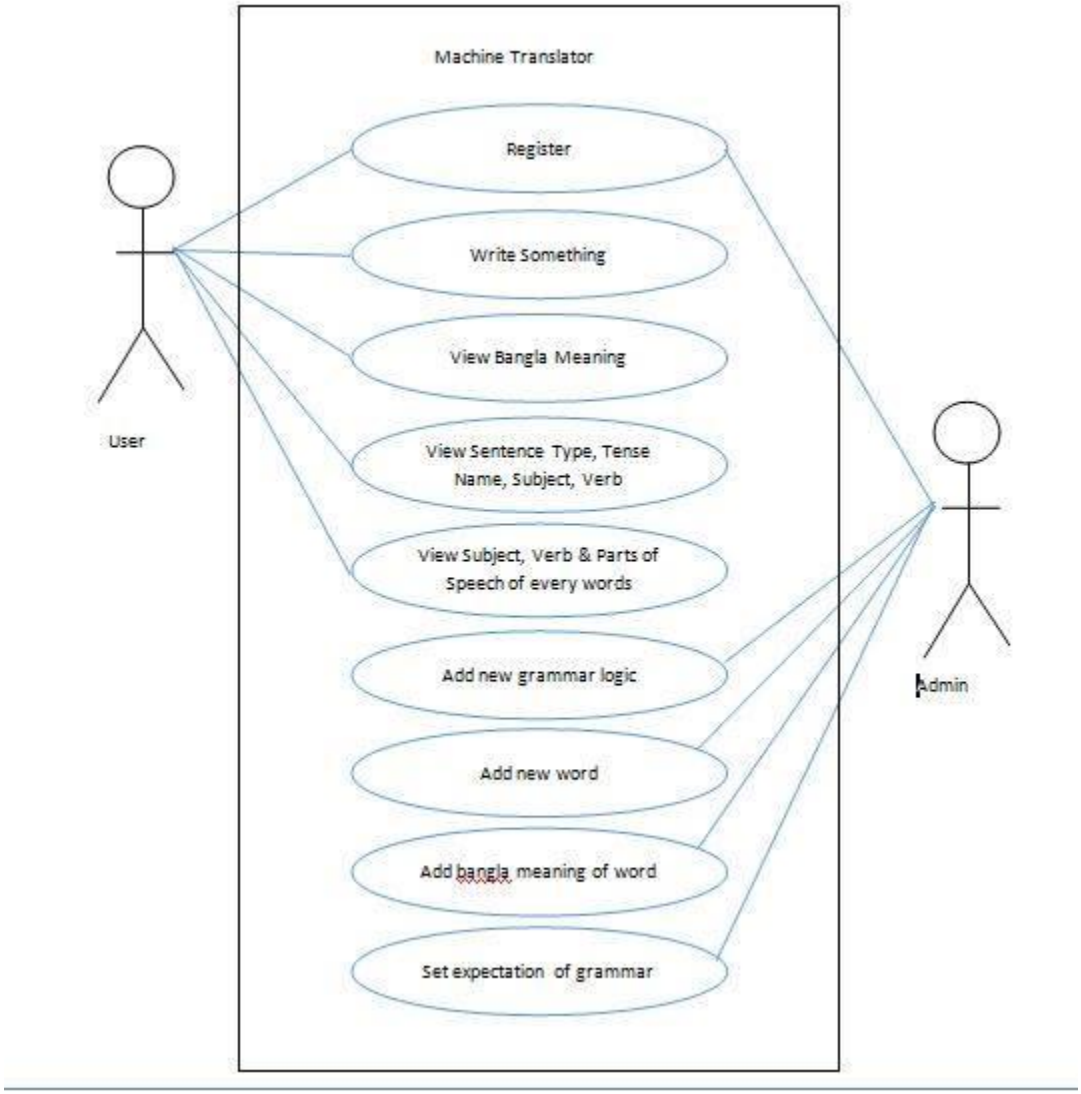


Figure 5.14: Use Case

5.2 Data Flow Diagram

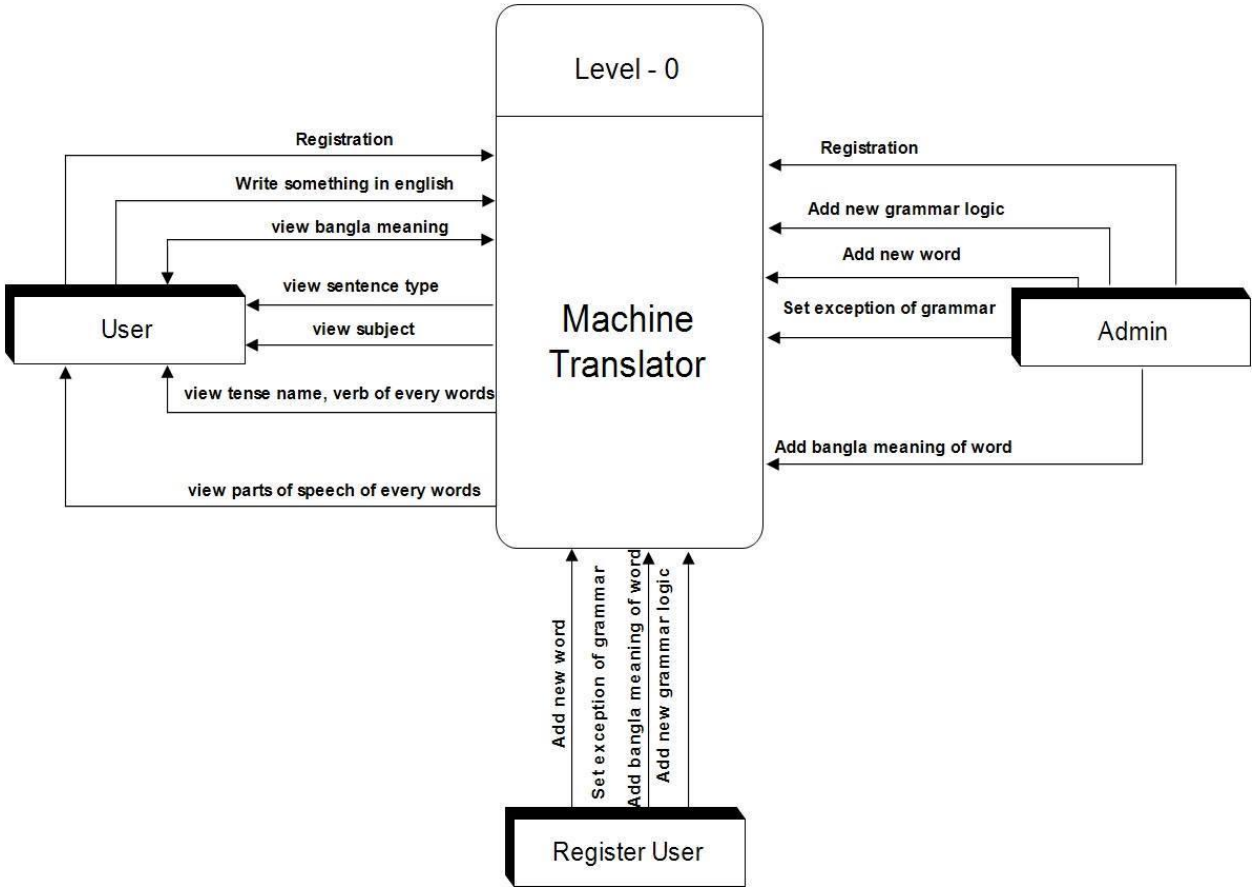


Figure 5.15: Data Flow Diagram

5.3 Entity Relation Diagram

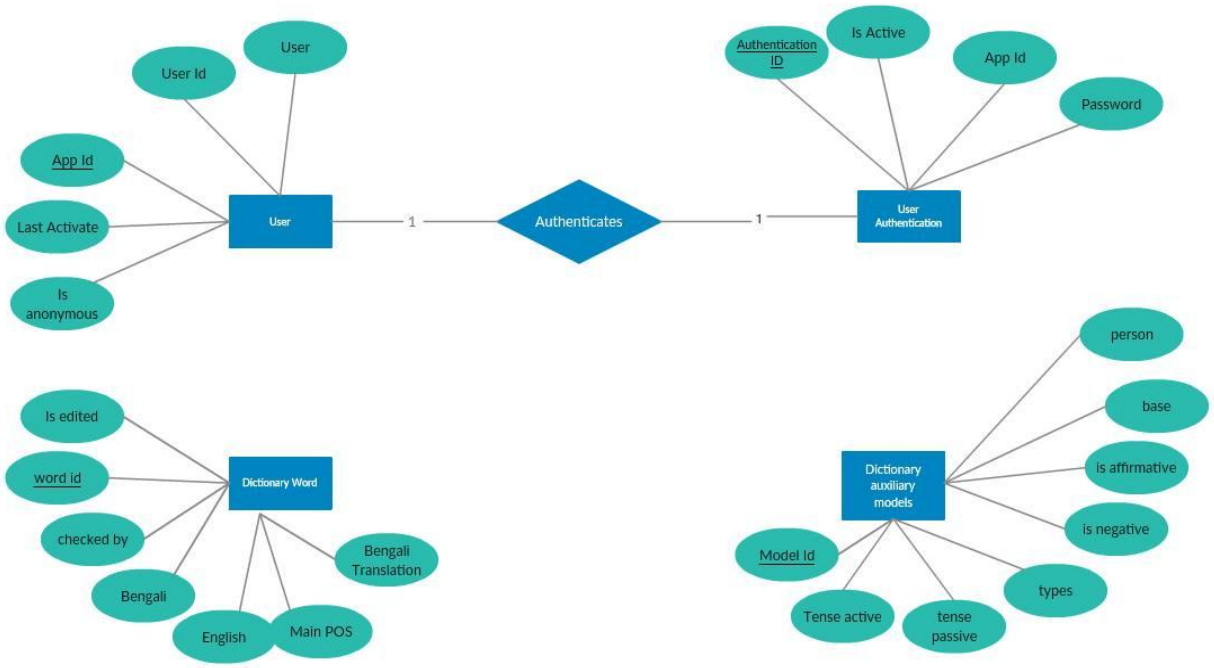


Figure 5.16: ERD Diagram

5.4 Class Diagram

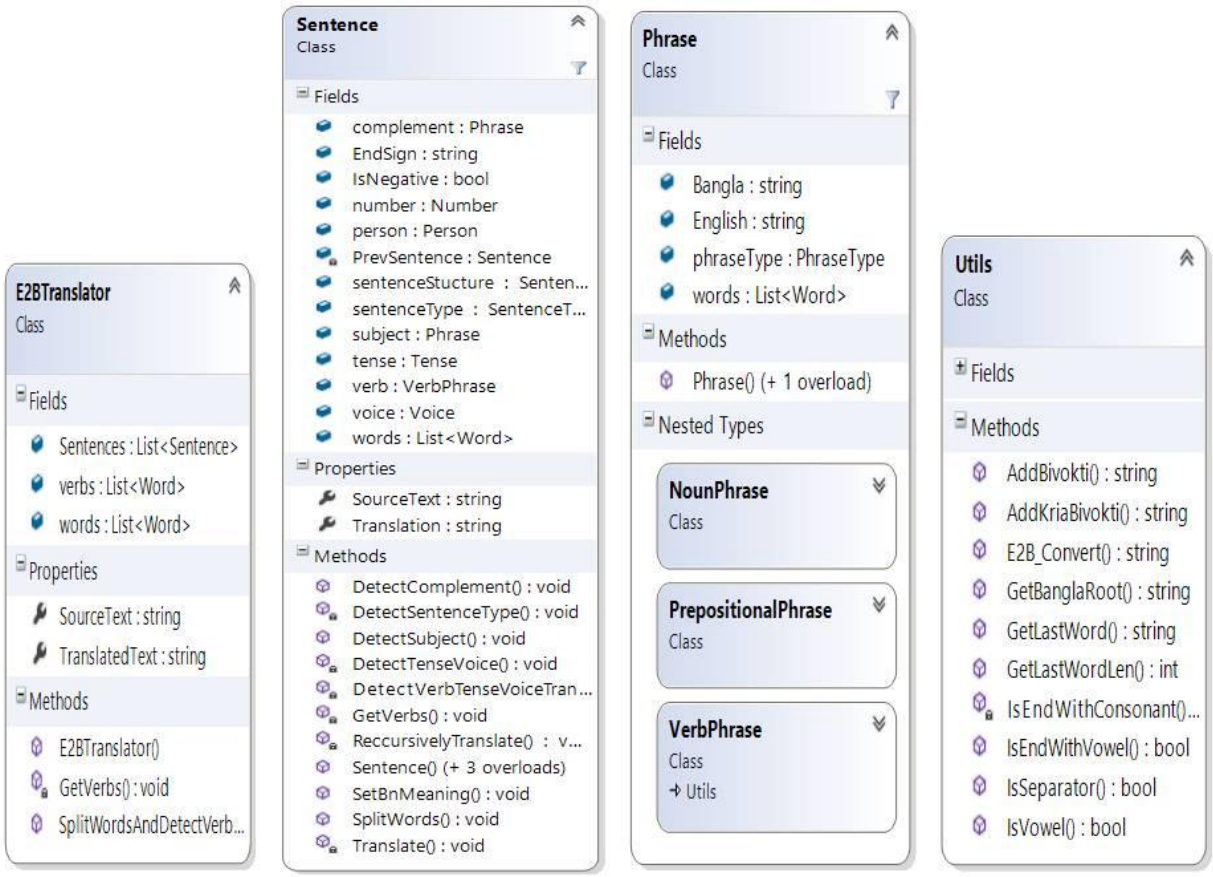


Figure 5.17: Class diagram

CHAPTER 06

SYSTEM LIFE CYCLE

6.1 System Development Life Cycle (SDLC)

Agile development model is also a type of incremental model. Software is developed in incremental, rapid cycles. This result in small incremental releases with each release building on previous functionality. Each release is thoroughly tested to ensure software quality is maintained. It is used for time critical applications. Extreme programming (XP) is currently one of the most well-known agile development life cycle models.

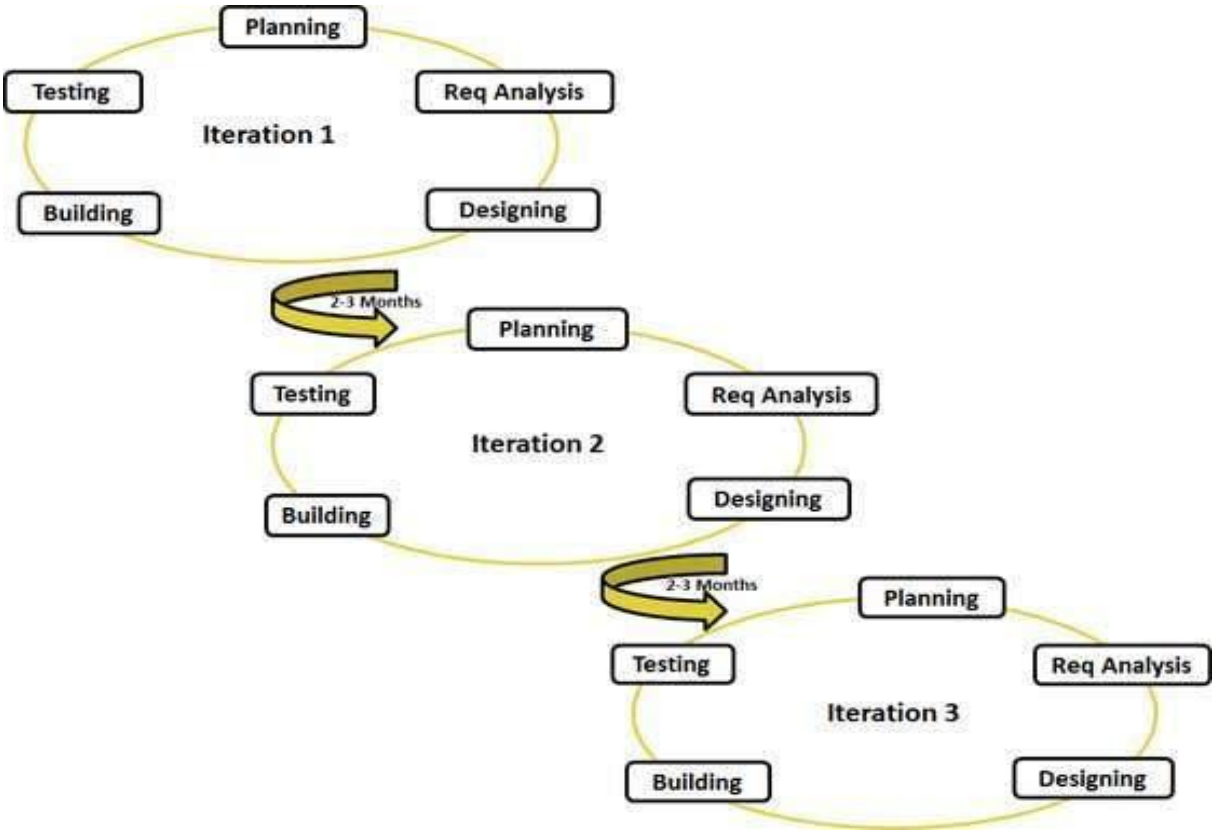


Figure 6.18: Agile model

CHAPTER 07

TESTING

7.1 Testing

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test.Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include the process of executing a program or application with the intent of finding software bugs (errors or other defects).

Software testing involves the execution of a software component or system component to evaluate one or more properties of interest. In general, these properties indicate the extent to which the component or system under test:

- meets the requirements that guided its design and development,
- responds correctly to all kinds of inputs,
- performs its functions within an acceptable time,
- is sufficiently usable,
- can be installed and run in its intended environments, and
- Achieves the general result its stakeholder's desire.

7.2 Testing Principles

Before applying methods to design effective test cases, a software engineer must understand the basic principles and guide software testing. Software testing is an extremely creative and intellectual challenging task. When testing follows the principles given below, the creative element of test design and execution rivals any of the preceding software development steps.

- Testing must be done by an independent party. Testing should not be performed by the person or team that developed the software since they tend to defend the correctness of the system.
- Test of invalid and unexpected input conditions as well as valid conditions. The program should generate correct messages when an invalid test is encountered and should generate correct results when the test is valid.
- Testing is the process of executing software with the intent of finding errors.
- Keep software static during test. The system must not be modified during the implementation of the set of the designed test cases.
- Document test cases and test results.

- Provide expected test result if possible. A necessary part of test documentation is the specification of expected results, even if providing such results is impractical.

7.3 Testing Technique

Because of the feasibility of its human designers and its own abstract, complex nature, software development must be accompanied by quality assurance activities. It is not unusual developers to spend 40% of the total project time on testing. For life critical software, testing can cost 3 to 4 times as much as all other activities combined. The destructive nature of testing requires that the developers discard preconceived notions of the correctness of developed software.

7.4 Test Plan

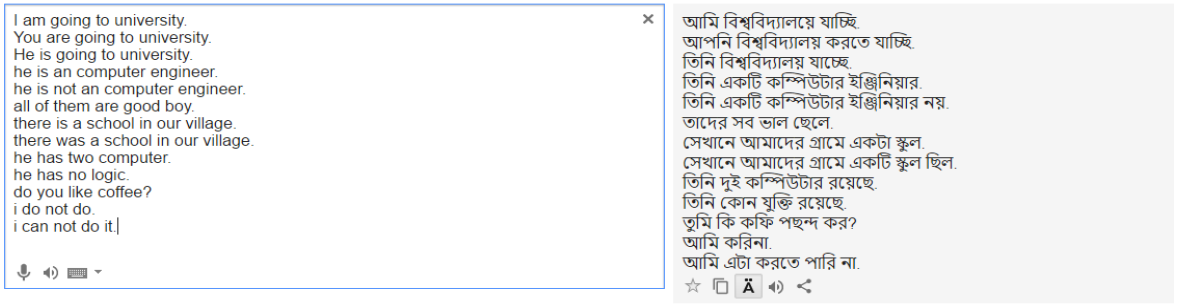
- ✓ Select the modules to be tested.
- ✓ Select the test data.
- ✓ Determined the expected results.
- ✓ Test the module using results.
- ✓ Determine the actual results.
- ✓ Compare expected and actual results.
- ✓ Test module selection and planning sequence.

7.5 Test Case Report

A set of test inputs, execution condition and expected results developed for a particular objective such as to exercise a particular system path or to verify compliance with specific requirement.

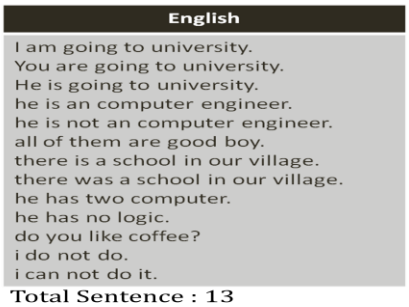
Test 01(Testing simple sentence):

Google Translator

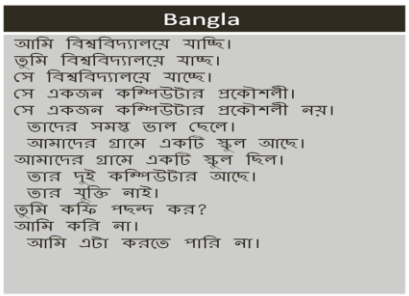


Total Sentence : 13
Pass: 4
Fail: 9

E2B Translator

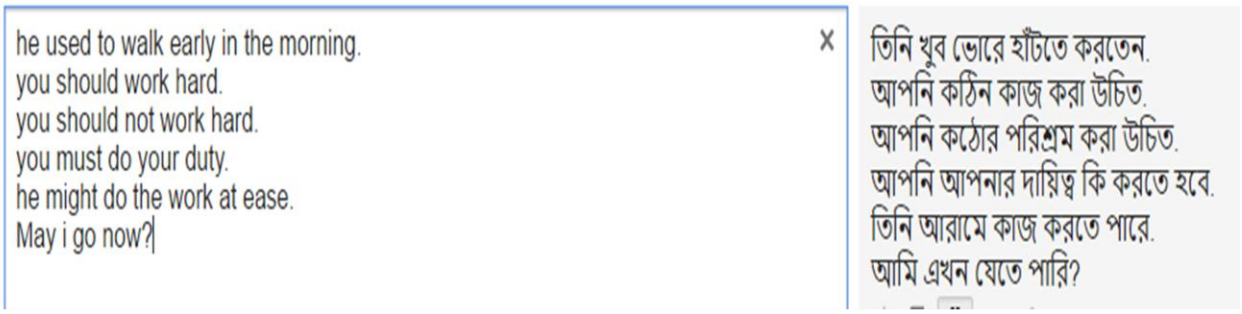


Total Sentence : 13
Pass: 12
Fail: 1



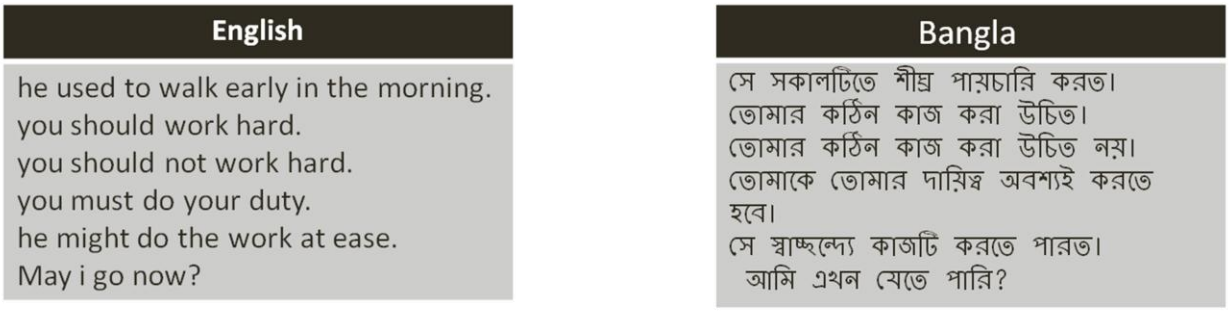
Test 02 (Testing complex sentence):

Google translator



Total Sentence : 6
Pass: 1
Fail: 5

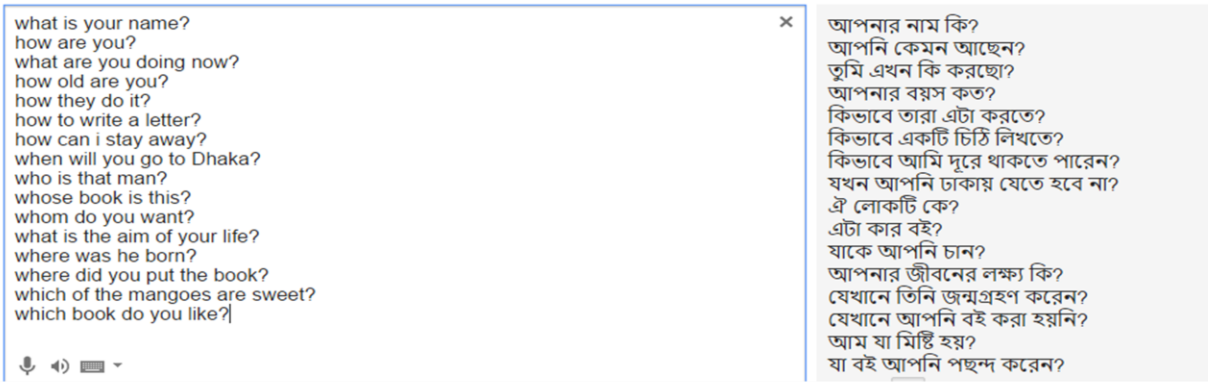
E2B Translator



Total Sentence : 6
Pass: 5
Fail: 1

Test 03 (Testing Sentence):

Google translator



what is your name?	আপনার নাম কি?
how are you?	আপনি কেমন আছেন?
what are you doing now?	তুমি এখন কি করছো?
how old are you?	আপনার বয়স কত?
how they do it?	কিভাবে তারা এটা করতে?
how to write a letter?	কিভাবে একটি চিঠি লিখতে?
how can i stay away?	কিভাবে আমি দূরে থাকতে পারেন?
when will you go to Dhaka?	যখন আপনি ঢাকায় যেতে হবে না?
who is that man?	ঐ লোকটি কে?
whose book is this?	এটা কার বই?
whom do you want?	যাকে আপনি চান?
what is the aim of your life?	আপনার জীবনের লক্ষ্য কি?
where was he born?	যেখানে তিনি জন্মগ্রহণ করেন?
where did you put the book?	যেখানে আপনি বই করা হয়নি?
which of the mangoes are sweet?	আম যা মিষ্টি হয়?
which book do you like?	যা বই আপনি পছন্দ করেন?

Total Sentence : 16
Pass: 7
Fail: 9

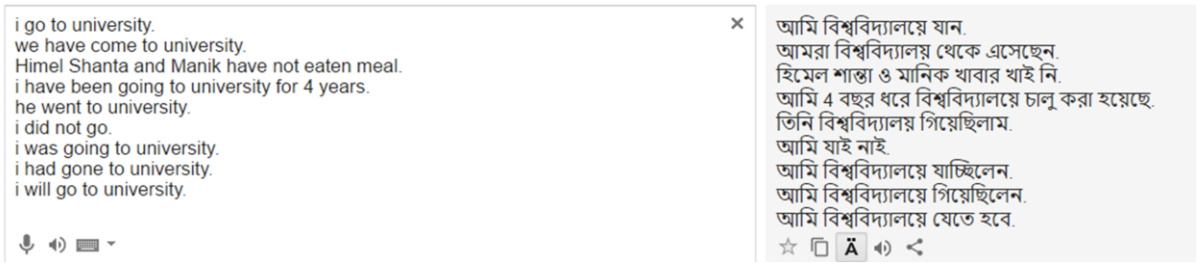
E2B translator

English	Bangla
what is your name?	তোমার নাম কি?
how are you?	তুমি কেমন আছ ?
what are you doing now?	তুমি এখন কি করছ?
how old are you?	তোমার বয়স কত?
how they do it?	তারা কিভাবে এটা করে?
how to write a letter?	কিভাবে একটি পত্র লেখে?
how can i stay away?	আমি কিভাবে দূরে থাকতে পারি?
when will you go to Dhaka?	তুমি কখন ঢাকা(dhaka)তে যাবে?
who is that man?	ঐ মানুষটি কে?
whose book is this?	এইটাটা কার বই?
whom do you want?	তুমি কাকে চাও?
what is the aim of your life?	তোমার জীবনের লক্ষ্যটি কি?
where was he born?	জাত সে কোথায় ছিল?
where did you put the book?	তুমি বইটি কোথায় রেখেছিলে?
which of the mangoes are sweet?	কোন আমগুলি মিষ্টি?
which book do you like?	কোন বই তুমি পছন্দ কর?

Total Sentence : 16
Pass: 15 (2 warning)
Fail: 1

Test 04 (Testing tense):

Google translator



Total Sentence : 9
Pass: 1
Fail: 9

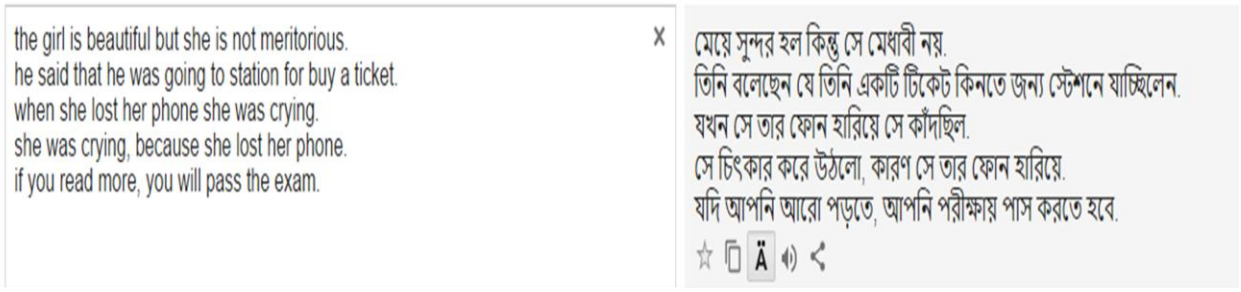
E2B translator

English	Bangla
i go to university. we have come to university. Himel Shanta and Manik have not eaten meal. i have been going to university for 4 years. he went to university. i did not go. i was going to university. i had gone to university. i will go to university.	আমি বিশ্ববিদ্যালয়ে যাই। আমরা বিশ্ববিদ্যালয়ে এসেছি। হিমেল (himel) শান্টা (shanta) এবং মানিক (manik) খাবার খায়নি। আমি 4 বছরের জন্য বিশ্ববিদ্যালয়ে যাচ্ছি। সে বিশ্ববিদ্যালয়ে গিয়েছিল। আমি গিয়েছিলাম না। আমি বিশ্ববিদ্যালয়ে যাচ্ছিলাম। আমি বিশ্ববিদ্যালয়ে গিয়েছিলাম। আমি বিশ্ববিদ্যালয়ে যাব।

Total Sentence : 9
Pass: 15 (1 warning)
Fail: 0

Test 05 (Testing complex):

Google translator



Total Sentence : 5
Pass: 0
Fail: 5

E2B translator

Total Sentence : 5
Pass: 3
Fail: 2

CHAPTER 08

FUTURE WORK AND CONCLUSION

8.1 Future Work

We need to find a way to update new sentence rules for English to Bangla EBMT. We are trying to Voice to Voice Machine translator, and then improve its efficiency for general purpose use. Make a machine learning system so that user can train it.

8.2 Conclusion

In English, many words can have multiple meanings but while developing the system, we were not able deal with all possible meanings; we worked with only one meaning of these types of words, which still remains a problem for the generalization of the translator. Our main target was to develop any kind of software, which can really help people in some problems. The system provides the user the facility to append new words in the dictionary. Though the number of the given words is a subset of the English language, but the user can enrich the stock of words with the help of an expert who has sufficient knowledge in both English and Bangla language by inserting new words directly to database or by program interface, The system can detect syntactic errors, if any, in the input sentence as part of error detection. But it is not universal. It has been implemented in such a way to make it possible for extending in the future successfully and efficiently. This system can translate all kinds of sentences. It also has been designed to translate both active voiced and passive voiced sentences with some exception.

SCHEDULE AND PLANNING

GANTT CHART

SL	Task name	Start	End	Duration	2015 Aug- Dec	2016 Jan- March
01	Requirement collection	1-08-2015	03-09-2015	34 days	34	
02	System analysis and design	4-09-2015	30-09-2915	29 days	29	
03	Coding	1-10-2015	10-01-2016	70 days	70	
04	Testing	11-01-2016	11-02-2016	30 days	30	
05	Maintenanceand evaluation	12-02-2016	07-03-2016	25 days	25	

REFERENCES

[1] Albat, Thomas Fritz. "Systems and Methods for Automatically Estimating a Translation Time." US Patent 0185235, 19 July 2012.http://en.wikipedia.org/wiki/History_of_machine_translation

[2] First and most notably Bar-Hillel, Yeheshua. "A demonstration of the nonfeasibility of fully automatic high quality machine translation." in Language and Information: Selected essays on their theory and application (Jerusalem Academic Press, 1964), Pp. 174–179.http://en.wikipedia.org/wiki/Rule-based_machine_translation

[3] "Madsen, Mathias: The Limits of Machine Translation (2010)". Docs.google.com. Retrieved 2012-06-12.http://en.wikipedia.org/wiki/Dictionary-based_machine_translation

[4] "Koehn, Philipp (2010). *Statistical Machine Translation"*. Cambridge: Cambridge University Press. p. 15. http://en.wikipedia.org/wiki/Statistical_machine_translation.htm/Statistical_machine _translation

[5] Nirenburg, Sergei (1989). "Knowledge-Based Machine Translation". *Machine Trandation 4 (1989), 5 - 24*. Kluwer Academic Publishers. R

[6] Lonsdale, Deryle; Mitamura, Teruko; Nyberg, Eric (1995). "Acquisition of Large Lexicons for Practical Knowledge-Based MT". *Machine Translation 9: 251-283*. Kluwer Academic Publishers. Retrieved 20 June 2012.http://en.wikipedia.org/wiki/Statistical_machine_translation

[7] Retrieved 20 June 2012.http://en.wikipedia.org/wiki/Rule-based_machine_translation

[8] Hettige, B.; Karunananda, A.S. (2011). "Computational Model of Grammar for English to Sinhala Machine Translation". The International Conference on Advances in ICT for Emerging Regions - ICTer20 11 : 026-031. Retrieved 20 June 2012.jvdm@translationautomation.com

[9] Lagarda, A.-L.; Alabau, V.; Casacuberta, F.; Silva, R.; Díaz-de-Liaño, E. (2009). "Statistical Post-Editing of a Rule-Based Machine Translation System". *Proceedings of NAACL HLT 2009: Short Papers, pages 217–220, Boulder, Colorado*. Association for Computational Linguistics. Retrieved 20 June 2012.http://www.en.wikipedia.org/wiki/wikipedia.citation needed.com

[10] Chitu, Alex (22 October 2007). "Google Switches to Its Own Translation System". Googlesystem.blogspot.com. Retrieved 2012-08-13.

[11] "Google Translator: The Universal Language". Blog.outer-court.com. 25 January 2007. Retrieved 2012-06-12.

[12] https://translate.google.com/about/intl/en_ALL/

[13] http://www.mt-archive.info/10/HyTra-2013-Tambouratzis.pdf

[14] “S. Vogel, H. Ney and C. Tillmann”. 1996. HMM-based Word Alignment in StatisticalTranslation. In COLING ’96: The 16th International Conference on Computational Linguistics, pp. 836-841, Copenhagen, Denmark.
[15] F. Och and H. Ney. (2003). A Systematic Comparison of Various Statistical Alignment Models. Computational Linguistics, 29(1):19-51
[16] D. Chiang (2005). A Hierarchical Phrase-Based Model for Statistical Machine Translation. In Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05).
[17] Albat, Thomas Fritz. "Systems and Methods for Automatically Estimating a Translation Time." US Patent 0185235, 19 July 2012.
[18] First and most notably Bar-Hillel, Yeheshua: "A demonstration of the nonfeasibility of fully automatic high quality machine translation," in *Language and Information: Selected essays on their theory and application* (Jerusalem Academic Press, 1964), pp. 174–179.
[19] Nagao, M. 1981. A Framework of a Mechanical Translation between Japanese and English by Analogy Principle, in Artificial and Human Intelligence, A. Elithorn and R. Banerji (eds.) North- Holland, pp. 173–180, 1984.
[20] "The Association for Computational Linguistics – 2003 ACL Lifetime Achievement Award". Association for Computational Linguistics. Retrieved 2014-03-14.
[21] http://kitt.cl.uzh.ch/clab/satzaehnlichkeit/tutorial/Unterlagen/Somers1999.df
[22] http://www.princeton.edu/~achaney/tmve/wiki100k/docs/Example-based_machine_translation.html.
[23] Vossen, Piek,*Ontologies*. In: Mitkov, Ruslan (ed.) (2003),“Handbook of Computational Linguistics”, Chapter 25. Oxford: Oxford University Press.
[24] J.M. Cohen observes (p.14): "Scientific translation is the aim of an age that would reduce all activities to techniques. It is impossible however to imagine a literary-translation machine less complex than the human brain itself, with all its knowledge, reading, and discrimination."
[25] Claude Piron, *Le défi des langues* (The Language Challenge), Paris, L'Harmattan, 1994.
[26] See the annually performed NIST tests since 2001 and Bilingual Evaluation Understudy.
[27] Muegge (2006), "Fully Automatic High Quality Machine Translation of Restricted Text: A Case Study," in *Translating and the computer 28. Proceedings of the twenty-eighth international conference on translating and the computer, 16–17 November 2006, London*, London: Aslib. ISBN 978-0-85142-483-5.
[28] Way, Andy; Nano Gough (20 September 2005). "Comparing Example-Based and Statistical Machine Translation". *Natural Language Engineering***11** (3): 295–309. doi:10.1017/S1351324905003888. Retrieved 2014-03-23.

[29] "Machine Translation: From Real Users to Research," Proceedings of the 6th Conference of the Association for Machine Translation in the Americas, AMTA 2004.
[30] Berwick, R., A. Weignberg, "Parsing Efficiency, Computational Complexity and the Evaluation of Grammatical Theories" Linguistic Inquiry, 1982.
[31] Documentation available at lingo.stanford.edu/courses/03/pg.
[32] Shah Asaduzzaman "A comprehensive study on MT towards development of Bangla- Translation system". Thesis at CSE, Bangladesh University of Engineering and Technology, September 1999.
[33] Bonnie Jean Dorr "UNITRAN: A Principle-Based Approach to Machine Translation", MIT, Artificial Intelligence Laboratory, Cambridge, Mass. (December, 1987).
[34] ftp://publications.ai.mit.edu/ai-publications/pdf/AITR-1000.pdf.
[35] Arturo Trujillo, "Translation Engines: Techniques for Machine Translation", Springer-Verlag London Limited, 1999.
[36] Documentation available at lingo.stanford.edu/courses/03/pg.
[37] Mohammed Mohisul Hoque and Muhammad Masroor Ali, "A parsing Methodology for Bangla Natural Sentences", ICCIT. 2003.
[38] Arturo Trujillo, "Translation Engines: Techniques for Machine Translation", Springer-Verlag London Limited, 1999.
[39] Berwick, R., A. Weignberg, "Parsing Efficiency, ComputationalComplexity and the Evaluation of Grammatical Theories" Linguistic Inquiry, 1982.
[40] Documentation available at lingo.stanford.edu/courses/03/pg.Ali, "Transfer Machine Translation-An Experience with Bangla English Machine Translation System"-ICCIT, 2003.
[41] Open NLPClass Library Help. [Online]. StatisticAvailable at: http://opennlp.sourceforge.net/[accessed 15 February 2014]
[42] Cal parsing ofEnglish sentences. [Online]. Available at:http://www.codeproject.com/KB/recipes/english
[43] W. Weaver (1955). Translation (1949). In: *Machine Translation of Languages*, MIT Press, Cambridge, MA.
[44] P. Brown, S. Della Pietra, V. Della Pietra, and R. Mercer (1993). The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, **19(2)**, 263-311.

APPENDIX

Project Scope

Primary Phase				
SL	**Job Description**	**Start Date**	**End Date**	**Total Days**
1	Idea Searching.	14-Aug-2015	17-Aug-2015	3
2	Domain Knowledge Gathering.	18-Aug-2015	21-Aug-2015	3
3	Resource Availability Checking.	22-Aug-2015	25-Aug-2015	3
4	Brain Storming.	26-Aug-2015	06-Sep-2015	11
5	Feasibility Study.	07-Aug-2015	17-Sep-2015	10
			Total Days	30

Idea Proposal

Advanced Phase				
SL	**Job Description**	**Start Date**	**End Date**	**Total Days**
1	Idea Sharing with Supervisor.	18-Sep-2015	21- Sep-2015	3
2	Advanced Domain Knowledge Gathering.	22- Sep-2015	25- Sep-2015	3
3	Advanced Resource Gathering from Supervisor and from online open resources.	2- Sep-2015	12- Sep-2015	10
4	Advanced Brain Storming with Supervisor.	13- Sep-2015	18- Sep-2015	5
5	Advanced Feasibility Checking with Supervisor.	19- Sep-2015	25- Sep-2015	6
6	SDLC Model Selection for System Development.	26- Sep-2015	29- Sep-2015	3
			Total Days	30

Requirement Collection

SL	Job Description	Start Date	End Date	Total Days
1	Develop System Scenario.	30- Sep-2015	5-Oct-2015	5
2	Requirement Identify from System Scenario.	6-Oct-2015	11- Oct-2015	5
3	Row Requirement Collection from Identified Requirement.	12-Oct-2015	22- Oct-2015	10
4	Functional Requirement Collection.	23-Oct-2015	2- Nov-2015	10
5	Software Requirement Specification (SRS).	3-Nov-2015	23- Nov-2015	20
6	Manage All Requirement and Arrange all Information.	24- Nov-2015	29- Nov-2015	5
7	Non-Functional Requirement Specification.	30- Nov-2015	4-Dec-2015	5
			Total Days	60

System Design

Physical System Design				
SL	**Job Description**	**Start Date**	**End Date**	**Total Days**
1	Graphical User Interface (GUI) UX Modeling.	5-Dec-2015	10- Dec-2015	5
2	System Prototype Designing.	11- Dec-2015	21- Dec-2015	10
			Total Days	15

Logical System Design

1	System Environment Designing.	22- Dec-2015	27- Dec-2015	5
2	Use Case Diagram Designing.	28- Dec-25	1-Jan-2016	3
3	Detail Flow Diagram DFD-Level: 0 Designing.	2-Jan-2016	5-Jan-2016	3
4	Detail Flow Diagram DFD-Level: 1 Designing.	6-jan-2016	10-jan-2016	4
			Total Days	15
			Total Days	30

System Implementation

Technology Used				
SL	**Job Description**	**Start Date**	**End Date**	**Total Days**
1	Language: Object Oriented Programming C#	11-jan-2016	21-jan-2016	10
2	ASP.Net Frame Work MVC-5	22-jan-2016	1-feb-2016	10
3	Bootstrap Web Designing Frame Work	2-feb-2016	12-Nov-2016	10
4	HTML – 5, CSS – 3	13-feb-2016	23-feb-2016	10
5	Database MS-Sql Server -2014	24-feb-2016	4-march-2016	10
6	Proto Type Design Tool https://pidoco.com (Pidoco)			
			Total Days	50

System Testing

Black Box Testing				
SL	**Job Description**	**Start Date**	**End Date**	**Total Days**
1	Boundary Value Testing.	5-march-2015	9-march-2015	5
2	System Functionality Testing.	10-march-2015	15-march-205	5
			Total Days	10
			Total Days	210