



GLOBAL JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY: G INTERDISCIPLINARY

Volume 20 Issue 3 Version 1.0 Year 2020

Type: Double Blind Peer Reviewed International Research Journal

Publisher: Global Journals

Online ISSN: 0975-4172 & Print ISSN: 0975-4350

A Systematic Approach to English to Bangla Sentence Translator

By Joyassree Sen, Bappa Sarkar, Md. Shamim Hossain & Md. Nazrul Islam

Islamic University

Abstract- This paper deals with the design and development of an expert sentence translation system. In this translator, the source language is English, and the target language is Bangla. The implemented translation system determines the relationship among different forms of English and Bengali sentences and makes appropriate correspondence between English and Bengali grammar. Here, we have been developing a top-down parsing program. The system incorporates itself with the dictionary and gives the corresponding Bengali meaning. The system performs translation procedure in three steps. The lexical analyzer reads the English sentence, tokenizes into words, and stores information into a stack. The lexical analyzer uses the English to Bangla dictionary and word morphology for finding lexical information. The parser parses the input sentence and identifies the types of it and finds tense, phrase, clauses, etc. The generator generates a Bangla sentence, which is equivalent to the given input English sentence. It uses the output of the lexical analyzer and the parser to make Bengali sentence. This system can translate all kinds of sentences. But the limitation is that it cannot handle semantic and contextual problems.

Keywords: MT (machine translator), parser, lexical analysis, stack, syntax, dictionary.

GJCST-G Classification: J.5



Strictly as per the compliance and regulations of:



© 2020. Joyassree Sen, Bappa Sarkar, Md. Shamim Hossain & Md. Nazrul Islam. This is a research/review paper, distributed under the terms of the Creative Commons Attribution-Noncommercial 3.0 Unported License (<http://creativecommons.org/licenses/by-nc/3.0/>), permitting all non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

A Systematic Approach to English to Bangla Sentence Translator

Joyassree Sen^α, Bappa Sarkar^σ, Md. Shamim Hossain^ρ & Md. Nazrul Islam^ω

Abstract- This paper deals with the design and development of an expert sentence translation system. In this translator, the source language is English, and the target language is Bangla. The implemented translation system determines the relationship among different forms of English and Bengali sentences and makes appropriate correspondence between English and Bengali grammar. Here, we have been developing a top-down parsing program. The system incorporates itself with the dictionary and gives the corresponding Bengali meaning. The system performs translation procedure in three steps. The lexical analyzer reads the English sentence, tokenizes into words, and stores information into a stack. The lexical analyzer uses the English to Bangla dictionary and word morphology for finding lexical information. The parser parses the input sentence and identifies the types of it and finds tense, phrase, clauses, etc. The generator generates a Bangla sentence, which is equivalent to the given input English sentence. It uses the output of the lexical analyzer and the parser to make Bengali sentence. This system can translate all kinds of sentences. But the limitation is that it cannot handle semantic and contextual problems.

Keywords: MT (machine translator), parser, lexical analysis, stack, syntax, dictionary.

I. INTRODUCTION

A translator is a program, which reads the source language as input and translates it into target language as output. The goal is to design and build software that will take English contexts as inputs and then analyze, understand them and finally generate Bengali languages, so that eventually we will be able to address our computer as though we were directing another person. By far, the utmost human linguistic communication occurs as speech. Written language is a recent invention and still plays a less central role than articulate sound in most activities. But processing of written is more facile than speech language [1]. For example, the pronunciation of a word differs with person

to person, but the structure or component of a word doesn't vary with persons. So the translation of the written form of language can be efficiently programmed. Parser plays a vital role in this translation process. A parser for a grammar (G) is a program that takes a string (W) as an input and produces a parse tree as an output [8]. There are two basic types of parsers, top-down parser, and bottom-up parser. We use top-down parsing. The general parsing process is being illustrated in the following figure.

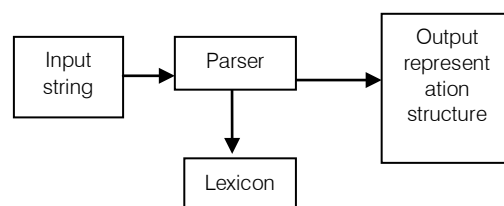


Fig. 1: Parsing an input to create an output structure

II. THE FUNDAMENTAL ARCHITECTURE OF THE PROPOSED MACHINE TRANSLATION (MT) SYSTEM

The basic architectural block diagram of the proposed MT system is depicted in the following figure. The system works in three steps: lexical analyzing, parsing, and Bengali sentence generation. The Lexical analyzer reads the English sentence, separates the words, and populates it with lexical information. After lexical analysis, all the words of a sentence and the resulting facts are stored into a stack for the parser. The parser uses a rule-based top-down parser to parse the input sentence for syntactic correctness. Finally, the generator produces Bangla Sentence from the parser outputs and the dictionary.

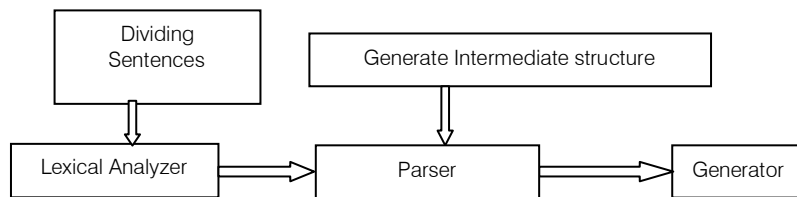


Fig. 2: Block diagram of the proposed translator

Author ^α ^σ ^ρ ^ω: Dept. of Computer Science and Engineering, Islamic University, Kushtia, Bangladesh. e-mails: joyassreesen@yahoo.com, bappacse07@gmail.com, shamim@cse.iu.ac.bd, shilunazrul@yahoo.com

a) *Lexical analysis of English contexts*

The Lexical analysis phase of MT is the first step, which is implemented by the Lexical analyzer. It can scan the whole document at a time or one by one. The later strategy is better for parsing, although it is a slow process. Thus sentence scanning and parsing technique have been used in the proposed system. It is looking for the sentence delimiter. It gathers all the information about the words. It tokenizes and sends the information to the parser and subsequently to the generator. During the execution of the Lexical analysis phase, the Lexical analyzer reads the input sentences from the keyboard or a text file given by the user and separates the words. To find the word in the Dictionary, the Lexical analyzer uses the word morphology techniques. For example, in any human language, a word is used in different form, and the dictionary contains only one form. In the lexicon, only the singular form is presented. It can generate its plural by using morphological techniques. The word morphology uses the following strategies:

1. Read the whole word and search it in the dictionary, proceed to the next word if it is found.
2. If dictionary is failed to match the word then check it for the proper noun. If it is proper noun then proceeds for the next word.
3. Discard the last letter from the word and apply the above two steps to the remaining words, if succeeds, check whether the discarded letters are a valid suffix for the recognized word. If it succeeded, it moves to the next word.
4. 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.

The above procedure considers only the suffix morphology: it can also incorporate prefix morphology for slandering prefixes. For using morphology, it requires to envisage what will be the meaning of the processed word and its part of speech.

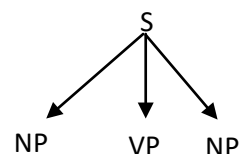
b) *Parsing English Sentence (Syntactic Analysis)*

This translator model uses the lexical analyzer. The parser obtains a string of token and verifies the source language which can be generated by the grammar. And one can use any efficient algorithm for parsing. We use top-down parsing technique for this purpose [4]. This parser reports any syntax error in an intelligible fashion. And it should also recover from commonly occurs errors so that it can continue processing of its input. There are two strategies of error recovery, and they are called panic mode and phrase-level recovery. Now the parser knows the format of the English sentence. While parsing, the system determines and keeps some significant information such as number and person of the subject, type of verb, the format of

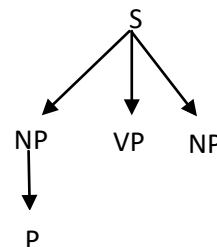
preposition etc. That will be useful during the formation of the Bengali sentence.

i. *Top-Down Parsing*

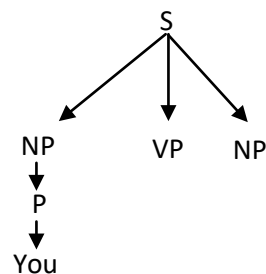
A top-down parser starts by hypothesizing a sentence and gradually predicting lower level element until individual terminal symbols have been written. In other words, top-down parser attempting to find a leftmost derivation for an input string. It trying to build a parse tree from the root of the input and creating the nodes of the parse tree in order. For example, consider the grammar $S \rightarrow NP VP NP$, $NP \rightarrow N | P$, $VP \rightarrow V$, $N \rightarrow \text{Babu} | \text{cow} | \text{mango}$, $P \rightarrow \text{You} | \text{I} | \text{He}$, $V \rightarrow \text{eat} | \text{drink} | \text{walk}$ and the input $w = \text{I eat mango}$. A top-down parser is used to construct a parse tree of this sentence, initially creates a tree consisting of a single node labeled S. An input pointer points I, the first symbol of w. Then use the first production of S to expand the tree and obtain



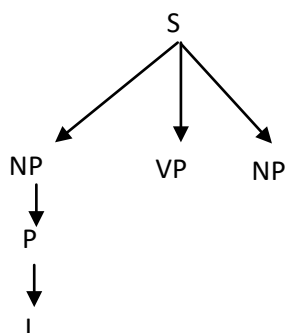
The leftmost symbol of the tree is a nonterminal, so expand it with the production rule for NP and obtain



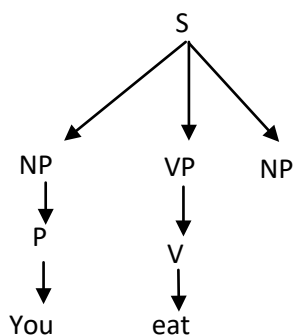
Again, the leftmost symbol is a nonterminal, so expand it with the production rule for P and obtain the following tree.



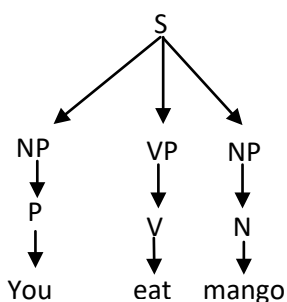
Now the leftmost symbol of the tree is a terminal; compare it with the word pointed by the pointer, which is I and does not match. Then, go back to P and see whether there is another alternate for P, which might produce a match.



Now, using the second alternative of P, replace the terminal you with terminal I, and find a match as shown. Then, forward the pointer to the next symbol of w, eat and go to the next leftmost nonterminal symbol of the tree, which is VP and expand it with its production rule. The last symbol inserted is not a terminal, which is a nonterminal V and expand it using its production rule and obtain following tree representation.



Then, forward the pointer to the next symbol of w, mango and go to the next leftmost nonterminal symbol of the tree, which is NP and expand it with production rule. The last symbol inserted is a nonterminal, so expand this using its production rule and obtain following tree representation.



Now, the last inserted word is a terminal symbol, the pointer reaches at the end of the input sentence w, and complete parsing process.

c) *Intermediate Representation*

After parsing, the system knows the structure of the given English sentence. For each formation of grammars, the system also has the structure in the database that compares the Bengali pattern of the

corresponding English structure. With the structure the system now translates the input English sentence into a converted form. It is the intermediate representation of the sentence.

d) *Translating into Bangla*

The next step is to perform the translation. In this phase, the system fetches the Bengali meaning for each token from the dictionary. The Bengali meaning for each noun or pronoun or adjective or adverb is replaced directly. But in the case of preposition and article, artificial intelligence is applied for the appropriate Bangla meaning.

i. *Modification of meaning*

We had to modify the meaning of the words according to different kinds of criterions such as verbs, articles, and prepositions [10]. In this section, we present different modifications to the meaning of the words.

a. *Verb*

The need for the verb table is due to the differences between the form of the verbs in English and Bengali languages. In English, there are four kinds of forms of each verb, but in Bengali, there are more than twenty-seven kinds of forms for each verb. So it becomes too difficult to find out the appropriate form of Bengali meaning. For this reason, we have to use a reliable method, which can create a direct link between the English and Bengali form of verb. For the present form of verb there are three categories in the meaning of the verb. Those are:

Verb that ends with a meaning “ওয়া”(ooa). Example: Give=“দেওয়া”

Verb that ends with a meaning “া”(akar). Example: Do=“করা”

Verb that ends with a meaning “নো”(no). Example: Spin=“ঘোরানো”

Following table represent different formation of verb:

Rule	English word	Bangla meaning	1 st person	2 nd person	3 rd person
”ওয়া”(ooa) changed to “ই” for 1 st person, ”ও” for 2 nd person and “য়” for the third person.	Give	দেওয়া	দেই	দেও	দেয়
”া”(akar) changed to “ই” for 1 st person, “ও” for 2 nd person and “য়” for the third person.	Do	করা	করি	করো	করে
”নো”(no) changed to “ই” for 1 st person, ”ও” for 2 nd person and “য়” for the third person.	Spin	ঘোরানো	ঘোরাই	ঘোরাও	ঘোরায়

b. Preposition

A preposition is a word placed before a pronoun or noun-equivalent to show its relation to any other word of the sentence. The noun or pronoun or the noun equivalent is called its object.

III. DESIGNING THE DATABASE

Our first job for any language conversion is to design and build a database to work as a dictionary. For that, we first determine the properties of any word which are required to understand the use of it in any kind of sentence. We needed a primary key column that contains the characteristics of any word which will be unique. And that required property is the word itself. And now we need a few columns to store various meanings of the word; the maximum meaning can occur to only verbs. Verbs have various meanings for the different person of the subject. They may also have special meanings that are required after modals or as gerunds. So we created four columns to hold those meanings and took two columns for containing their parts of speech and their type. We also create a column named person to store the person of word if it is used as a subject. We also created two columns for holding the antonyms and synonyms of the word. And then, we inserted the words and their properties in the table. We use MySql Query Browser to perform those actions.

a) Returning a required property of a word

We create a function named ‘find’ to return particular properties of certain words. While calling that function, we passed a string and two integers as parameters. The string contains the word to match with the dictionary. First integer specifies from which column we search the word to match. The other integer specifies contexts of which column will return to function. A different column holds different properties like meanings, parts of speech etc, of the word.

b) User input to the Database

While translating, we may find a word that doesn't exist in our dictionary. In that case, we ask the user to give the meaning and other properties of the function. We simply call another window page to take the input and have passed the inputs into the ‘insert’ function. The insert function takes the properties as parameters, builds a query statement, and then calls ‘execute update’ function passing that statement as a parameter. The function executes that query and inserts the properties.

IV. ALGORITHM

Here we will see the general procedure of our translation task.

Algo_Translation()

Step-1: Take and input English sentence from the user.

Step-2: Split the sentence into word.

Step-3: For each word do step 4

Step-4: From dictionary find the appropriate meaning, parts of speech, type(whether the subject, object or none), person (if subject or object)

Step-5: From above, find the subject, the verb, and the object of the sentence.

Step-6: Determine the structure of the sentence by the placement of subject, object, verb, person of subject etc.

Step-7: Put the word meanings as an order by which a corresponding Bangla Structure of the English Sentence Formed.

Step-8: Show the Bangla sentence at the user interface in any Bangla font}

V. TRANSLATING THE SENTENCE TO BANGLA

Dividing Our work into following parts

1. Pre-processing the sentence.
 - i. Dividing the sentence into words
 - ii. Checking if any word doesn't exist in the Dictionary
 - iii. Setting and saving parts of speech and types of words.
2. Finding the primary subject of the sentence.
3. Determining the type of sentence.
4. Putting the appropriate meaning in order.

a) *Pre-processing of the sentence before actual translating*

We divide the sentence using the split function and store the words in a string array. Then calling the 'find' function for each word we first see whether it exist in the dictionary or they are some processed form of a word that exists in the dictionary. We pass the word to a function named 'match' to check some modified form of an existing word (such as eats is a modified form of eat). By using this function we re-modify the word and match it with an existing word. If it exists then we get and save their parts of speech in 'POS' array of strings again with the help of 'find' function and save their types (whether it can be used as subject or object or both or none) in 'TYPES' array of string.

i. *Finding the Subject*

The most vital job to translate is to find the subject of the sentence. The form of verb varies with the person of subject, and positioning of words depends on it. So we create and use the function 'find Sub' for finding the subject. In that function, we check the previously construct 'Types' array. In a sentence, the subject is the first encountered word, which is either a noun or a pronoun or a gerund. So we search the Types array for the first word whose type is either 'subject' or 'both' until the occurrence of a verb or the end of sentence and place the meaning of the subject in a sentence. If we find an adjective before the occurrence of the subject then we place that before the subject. If we do not trace a subject then we assume the sentence as an imperative one, and the person of the subject is assigned. After finding the subject, we return from the function. Though we will return from the function if we find a verb assuming the sentence as imperative, however, we will continue even if we find auxiliary verb or a modal verb, but we will stop if we find 'let'. We also set the Boolean variable to have modal as true if we find a modal. We also set the Boolean used[i] as true where i+1 is the position of the subject or Auxiliary verb.

ii. *Setting the type of sentence*

Now one of our important jobs is to determine the type of sentence. We have divided it according to its meaning. We determine whether this is assertive,

interrogative, imperative, optative, or exclamatory. We initially assume the sentence type as an assertive one. However, if the first word is a modal one but not 'let' or is an auxiliary verb or 'wh' pronoun, then sets the type as an interrogative sentence and calls the created function 'SetAs Question'. If it finds a verb as the first word, then it sets type as imperative. If the first word is 'may', then it is an optative one. Otherwise, we stick to our initial guess as assertive. If the first word is a modal, then the type is also modal, and then we set Boolean has modal as true. The function determines the questioning word, when the function 'setAsQuestion' is called. If the first word is 'wh' type, then the questioning word is the corresponding meaning. Otherwise the questioning word is the meaning of 'what' ("ki"). After determining the questioning word, we place that word after the subject.

iii. *Putting the Bangla meaning together*

Now we have done all the requirements of understanding the forms and meaning of a sentence. Now we will put the appropriate meanings in the sentence and thus will be building our translated sentence. For this, first, we use a for-loop to consider all the words in the given sentence. If the considering word is a preposition, then we will retrieve the corresponding meaning from the second column of the table dictionary from the database. Then we hold the meaning in a stack as it will use later after the object. If we find an "Adjective", then that will also be held in a stack. If we trace a word of type "negation", then we check if the previous word was an auxiliary verb or modal or do, does, did. If so, then we understand that the sentence is a negative one. When the type is negative, then we place the meaning of that word in the string "AfterVerb". If we find a verb, then we call function 'placeverb' and 'placeobject'. If we find a word, it can be used as an object. And if we find an auxiliary verb, we neglect that and move to the next word. And in any other case, we put the word straight into the sentence.

b) *Process of getting output*

Let the input sentence is, "I am eating mango and he is drinking milk." At first, the sentence is broken into two sentences, and those are:

1. "I am eating mango."
2. "He is drinking water."

Then the system takes the first sentence for processing. The sentence is read from left to right and grouped into words that are separated by space. Function split() is used to do this. The output of this function is a sequence of words constituting the sentence:

1. I
2. Am
3. Eating
4. Mango

Then we search each word in the dictionary and obtain its parts of speech and its meaning. We rearrange words of the sentence by using the rules of part of speech. Here we have the following English structure:

Pronoun + Aux + Verb + Noun (I am eating mango)

We get following sentence structure after re-arranging it.

Pronoun + Noun + Verb (I mango eating)

After this step, the words are arranged as

1. I
2. Mango
3. Eating

Now an intermediate representation is constructed by direct word-to-word interpretation. After this step, we have intermediate Bangla representation:

আমি আম খাচ্ছি

Now the meaning of “eating”, i.e. “খাচ্ছি” is modified according to the person of the subject (1st person in this case) and tense (continuous in this case).

After all other necessary modifications, the meaning of the sentence is “আমি আম খাচ্ছি”. It is the output of the translator. Other part of the sentence is “he is drinking water”, which follow the same process. After this, the conjunction meaning picked from the database and places it between two parts of the sentence. Finally, we will see the following output.

আমি আম খাচ্ছি এবং সে পানি খাচ্ছে

VI. SOME EXPERIMENTAL RESULT

Input sentence (in English)	Translated Sentence (in Bangla)
It is his luck	এটা তার ভাগ্য।
It is a wrong word.	এটা একটি ভুল শব্দ।
The market is noisy	বাজারটি কোলাহলময়
Let me go.	আমাকে যেতে দাও।
Let him go.	তাকে যেতে দাও।
See the sky.	আকাশটি দেখ।
Read the book.	বইটি পড়।
Am I eating?	আমি কি খাচ্ছি?
Is he eating?	সে কি খাচ্ছে?
Do I eat rice?	আমি কি ভাত খাই?
Does he go to school?	সে কি স্কুলে যায়?
How are you?	তুমি কেমন আছো?
May I go?	আমি কি যেতে পারি?
When will you go?	তুমি কখন যাবে?
Which book do you like?	তুমি কোন বই পছন্দ কর?
We are enjoying song.	আমরা গান উপভোগ করছি।
This is my country.	এই আমার দেশ।
Shall I go?	আমি কি যাব?
You will drink milk.	তুমি দুধ পান করবে।
Walking is good.	হাটা ভালো।
I have asked him.	আমি তাকে জিজ্ঞাসা করেছিলাম।
He has added it.	সে এটা যোগ করেছিলো।
The sky is blue.	আকাশ নীল।
Reading book is good.	বই পড়া ভালো।
I run fast.	আমি দৌড়াই।
I drink milk and you see fish	আমি দুধ পান করি এবং তুমি মাছ দেখ।
I am weak but he is strong.	আমি দুর্বল কিন্তু সে শক্তিশালী।

VII. DISCUSSION

The user can be able to append the required word and its necessary information in the dictionary. Suppose, the input sentence of our system is “Bangladesh is a beautiful country.” And our dictionary does not contain the word “beautiful”. So, for this input, the system responds with the message as follows:

“দুঃখিত, আপনার দেওয়া কোন একটি শব্দ অভিধানে নাই”

The user can eliminate this problem by adding the bangla meaning of the word “beautiful” to the dictionary, and then we get the following output
“বাংলাদেশ একটি সুন্দর দেশ”

In the case of a syntactically correct sentence, the system translates nicely. However, when a sentence

is syntactically incorrect, or the delimiter of the sentence does not match the sentence type, the system tries to find the nearest match and translate and finally gives the closest Bangla output. This system can translate an assertive, interrogative, optative, imperative, exclamatory, simple, compound, and complex sentences. There are more sophisticated compound sentences, which may contain more than one clause. The system cannot provide any output for this type of sentences. We can solve this problem by adding rules for parsing multi-casual sentence. This system can translate both active voiced and passive voiced sentences. Besides this, the morphological techniques of English and Bangla are implemented in the scanning and generation phases, respectively. But Bangla sentence has several construction rules for personal pronouns. So, this system does not support all of them. This system supports one form only. For example, consider the translation of the sentence, "You are a

good girl.". The translation may be "তুমি একজন ভাল মেয়ে।" or "তুই একজন ভাল মেয়ে।" etc. However, the system gives only "তুমি একজন ভাল মেয়ে।" as output. The word "You" has several meanings, and it is difficult to translate it in the correct form in the context of Bangla grammar. Similarly, the meaning of "He" or "She" may be সে or তিনি. This type of problem is common in Bangla grammar. So, it is difficult to construct a Machine Translator for Bangla to handle all possible meaning of a word. In English language, the same word can be used as different parts of speech in different sentences. As a result, identify the correct form of a particular word in a sentence is a difficult task.

In most cases, the preposition does not maintain specific rules in English sentences. Therefore, it is difficult for a rule-based parser to correctly identify prepositions for a particular meaning. For example, "to" is a preposition and it can be used as-

1. "I run quickly to the large house". Here "to" is used to represent Bangla suffix "দিকে"
2. "I have no money to buy a car". Here "to" is used to represent Bangla suffix "মতো"
3. "He goes to school". Here "to" is used to represent Bangla suffix "দিকে"

There is no precise grammar in English for determining cases of nouns and pronouns. But it is an essential tool in Bangla to express something clearly. Therefore, cases of different nouns and pronouns in an English sentence should be identified properly before translating a sentence into Bangla. Here, detecting the relative position of the noun and pronoun with the verb and other words in English.

On the other hand, In Bangla, they are identified by considering the suffixes with nouns and pronouns.

Bangla is a relatively free word order language than English. So, sentence construction in Bangla has a less specific rules. For example, let us consider the sentence "You have given him pen". This sentence can be translated as

1. "তুমি তাকে কলম দিয়েছ"
2. "তাকে তুমি কলম দিয়েছ"
3. "তুমি কলম দিয়েছ তাকে"
4. "তাকে কলম দিয়েছ তুমি"

Therefore, for Bangla sentences, parser design is very difficult. More complicated grammar should be developed to avoid the problem.

VIII. CONCLUSION

The system provides the user with the facility to append new words in the dictionary. Though the number of the given words is a subset of the English language. The user can enrich the stock of words with the help of an expert who has sufficient knowledge in both english and bangla language. Although the developed system is

successful in many aspects, Still have some limitations those are:

1. The knowledge base in this system is not self-learning. It cannot interfere the existing decision in the knowledge base.
2. The system cannot handle the contextual and semantic problems.

REFERENCE AND BIBLIOGRAPHY

1. M. Haque and M. Hasan, "English to Bengali Machine Translation: An Analysis of Semantically Appropriate Verbs," 2018 International Conference on Innovations in Science, Engineering and Technology (ICISSET), Chittagong, Bangladesh, 2018, pp. 217-221.
2. Nur Hossain Khan, Md. Farukuzzaman Khan, Md. Mojahidul Islam, Md. Habibur Rahman, Bappa Sarker, Verification of Bangla Sentence Structure using N-Gram, Global Journal of Computer Science and Technology: A Hardware & Computation Volume 14 Issue 1 Version 1.0 Year 2014
3. M. K. Rhaman and N. Tarannum, "A Rule Based Approach for Implementation of Bangla to English Translation," 2012 International Conference on Advanced Computer Science Applications and Technologies (ACSAT), Kuala Lumpur, 2012, pp. 13-18.
4. Md. Musfique Anwar, Mohammad Zabed Anwar and Md. Al-Amin Bhuiyan, Syntax Analysis and Machine Translation of Bangla Sentences, IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.8, August 2009.

5. A. P. Mukta, A. Mamun, C. Basak, S. Nahar and M. F. H. Arif, "A Phrase-Based Machine Translation from English to Bangla Using Rule-Based Approach," 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE), Cox's Bazar, Bangladesh, 2019, pp. 1-5.
6. Md. Shahidur Rahman," Bangali Sorting Algorithm :A Linguistic Approach ", NCCPB ,2004.
7. Md. Abdul Muttalib," A Review on Computer Based Bangali Processing ", NCCPB ,2004
8. P .C. Wren & Martin," English Grammar & Composition", & Company LTD, 2000 .
9. Lelin Mehedy, S. M. Niaz. Arifin, M .Kaykobad," Bangali Syntax Analysis: A Comprehensive Approach ", ICCIT, 2003.
10. Masud, M .A .N Jorder, M. M. Tariq-UI-Azam," A Case Study on Syntactic Approach to Natural Language Processing", Undergraduate Thesis Paper, Computer Science And Engineering Discipline, Khulna University, Bangladesh November 2001 .
11. Aho, A.V. Shethi, R, Ulman, J.D. "Compiler Principle Techniques and Tools", Pearson Education Sixth Indian Reprint, 2001.
12. Chowdhury, Z.A., "Modern Applied Grammar Translation Composition", Globe Library (PVT.) Limited Edition 1997.
13. Ahmed E. "English Grammar and Translation", Globe Library, 1989.
14. M.D., Rahman, S.H., Vchchotor Bangali Grammar", Globe Library, 1989.