American Journal of Computer Science and Engineering
2014; 1(4): 25-29
Published online October 30, 2014 (http://www.openscienceonline.com/journal/ajcse)

An algorithmic approach to develop auto translation system - Bengali to English

Md. Ahsan Arif[1], Chandni Bhattacharya[2], Abul Barkat Mohammad Abdus Salam[3], Suriya Islam[4]

[1]Department of CSE, Asian University of Bangladesh, Dhaka, Bangladesh
[2]Department of CSE, Dhaka University of Engineering and Technology, Dhaka, Bangladesh
[3]Software Engineer, Solution9 Ltd., Dhaka, Bangladesh
[4]Graduated from CSE, Asian University of Bangladesh, Dhaka, Bangladesh

Email address

mdahsanarif@yahoo.com (M. A. Arif), chandniaub@yahoo.com (C. Bhattacharya), asalam345@gmail.com (A. B. M. A. Salam), tamanna.aub039@gmail.com (S. Islam)

To cite this article

Md. Ahsan Arif, Chandni Bhattacharya, Abul Barkat Mohammad Abdus Salam, Suriya Islam. An Algorithmic Approach to Develop Auto Translation System - Bengali to English. *American Journal of Computer Science and Engineering.* Vol. 1, No. 4, 2014, pp. 25-29.

Abstract

In today's globalized world, keeping aware and staying ahead on all sorts of issues from health, environment and energy to political affairs, security and the financial crisis has become a mammoth task. Breaking down language barriers means first-hand access to relevant information that helps to keep the citizen, the policy maker, governmental authorities, the private sector, industry, etc. better informed, especially on fast changing and impacting issues. In the last decade, the working method in Computational Linguistics has evolved to achieve an ultimate goal gradually. Fifteen years back, most of the researches focused on selected example of sentences. Now a day, the information access and utilization of large text is at a common stage. The final output of this research would be like below: Definition of the Bengali's word/information need; Selection of the Bengali's natural information sources to be used; Translation of the Bengali user's query expressed in natural language into the targeting language(English/German/French) of the information source, if necessary; Translation of the Bengali Expression from the Targeting(Foreign) Language to the query language of each information system; Implementation of Bengali Expressions obtained from the query language; Assessment of results by the user and the redefinition of the query expressions if the results are not relevant; and Selecting and obtaining the documents that respond to the user's needs.

Keywords

Natural Language Processing, Machine Translation, Fuzzy Matching, Computational Linguistics

1. Introduction

Since the beginning of Natural Language Processing, computers have played a role in the translation of text between languages. The two main branches of this field have been computer-assisted Human Translation, where the computer facilitates a human translator, and human-assisted Machine Translation (MT), where the computer translates text, and the human editor corrects the translation.

One important resource for computer-assisted human translation is *Translation Memory* (TM). First proposed in the 1970s, the idea is that the translator can consult a database of previous translations, looking for anything similar enough to the current segment to be translated, and can then use the retrieved example as a model. The key to the process is efficient storage of the segments in the TM, and, most importantly, an efficient matching scheme [4][5].

TM is limited in that it can only translate text that has previously been translated. However, where it does find a match, the translation produced is of high quality (it was in fact written by a human). Automatic machine translation, on the other hand, can translate previously unseen text, however the quality of the

output is rarely good, and requires human editing.

Both TM and MT are promising areas of investigation, but we will restrict our investigation to the area of TM, as the scopes for improvement here in a short project are far higher.

Traditionally, TM involved matching of a complete sentence against those stored in the TM. If an identical sentence was found, then its stored translation was offered for the new sentence. However, the possibility of exactly repeated sentences is small, except in the context of re-translating a modified document.

More recently, commercial TM systems have introduced 'fuzzy matching', allowing matching based on character-string similarity. A new sentence can match one in the TM if it is substantially similar to one stored, and the translation of the original sentence is offered as the starting point for the human translator to modify.

However, even fuzzy matching is not enough to make TM as useful as it could be for text which is basically new. The goal of this work will be to extend the flexibility of TM by reducing the size of units that TM functions on, and thus increase the probability that the word sequence will be observed again.

In observance of the past researches on different languages; like English, European, Asian and American, for the translation system, most of them has asserted on English to Native transitions. Some of the researches have also been done on Native Language (NL) to English translations. We have considered here all the past researches and afterward we decided that TM along with MT will highly be incorporated with knowledge based linguistic dictionary [1]. We hope that these entrepreneurs will up-heave the translation system a step forward, which will be a success. It is our proposal that this knowledge based dictionary will be accessed by an artificial intelligence based algorithm, which will also be within the integrated compiling system. The advantages of this process do not hamper the functions of email, webpage and text translations etc. and this will be represented through an application program interface (API)[2][3].

2. Problem Definitions

Bengali is the 4th widely spoken language with more than 250 million speakers, most of them live in Bangladesh and the Indian state of West Bengal. The alphabet consists of forty nine consonants and eleven vowels. Most of the words are built from consonantal roots in which inflections and derivations are generated by vowel changes, insertions and deletions. We know Bengali is the combination of Sanskrit, Hindi, Arabic and other languages. So the grammar of this language is too much prosperous. Therefore, it is not an easy task to convert Bengali into English. Some research works have been done on this topic. Among the researches, Google Translate is one of them through which we can get an output. But the output is not satisfactory because if we write **গোলাপী গোলাপী রং এর জামা পরে খেলছে।** then it will translate like the pink color of pink clothes plays. But this is not the actual translation of the sentence. So, there is some lacking in the widely used conversion technique at present.

3. State of the Art

According to our Analysis, we have planned to develop such an algorithm which can convert any naturally expressed Bengali sentences without any human effort. I will consider the assertive and interrogative sentences. For example, the sentence **"বাংলাদেশ ক্রিকেট দল বিশ্বকাপ জয়ের জন্য প্রতিদিন সকালে স্বতঃস্ফুর্তভাবে নতুন কোচের সাথে ঢাকা স্টেডিয়ামে ক্রিকেট অনুশীলন করে।"**

At first, we will check the punctuation for identifying the type of sentence and the part of sentence. For this sentence, the punctuation is "।" so it is an assertive sentence.

Secondly, we will convert each and every word in English from Knowledge Base (KB) dictionary. For the above example sentence, at first the programming parser (user defined function) will find English word for "বাংলাদেশ". So it finds an English word "Bangladesh" from KB dictionary. This means that dictionary construction and data entry is essential for this kind of implementation. We also have implemented an algorithm for the noun or name such as "রহিম". It will write as "Rahim" using the phonetics technique. One of the popular applications for this kind of algorithm is AVRO phonetic (English to Bengali). After that the next word is "ক্রিকেট". It will find an English word for this word is "criket", then for the word "দল" it will find "team", for the word "বিশ্বকাপ" it will find "the worldcup". After considering the word "জয়ের", it will find an English word "Joy" and the same time the duplication column (Entity/Field of the Record in the Database Table) will indicate that same type of Bengali word is available in KB Dictionary. Because two Bengali words sometimes creates a meaningful word and this idea is essential for the South Asian region. Otherwise Human support become essential with computer generated translation. After that programming parser immediately considers the next word "জন্য" and then it will again search the word "জয়ের জন্য". Finally it will get output "to win". Following the above process explained, all the words will be translated to English with real meanings. Here the main goal is matching multiples word at a time with the KB Dictionary. This is a new approach with this proposal according to published research article. This process is not only the searching related word from dictionary. It is constructing a meaningful sentence immediately after the completion of Bengali word translation.

We have planned to add manner or attribute for every word as like the Bengali words "জন্য, কারণে". If this kind of word found after any word then the attribute or manner will be treated as Reason. There are some Bengali words like "ভাবে, করে, করতে-করতে". If the program found this type of word after any word then the attribute will be treated as "Manner1". In the same way if there are some words like "শেষে, মিশে, সাথে, দিয়ে" after any word then it will be treated as "Manner2". There are some common words like "সাধারণভাবে, প্রাকৃতিকভাবে". The attribute or manner for these kind of words will be treated as Adverb of Frequency. If there are some words like "হতে, থেকে, চেয়ে" then before these words will be treated as Destination1 (starting point) and after these words is Destination2. If the ending part of any word consists "এ, য়, তে" then the attribute

will be treated as “place”. If any word found like “সকালে, বিকালে, ১০টায়” then these words will be indicated as “time” attribute. If there are any “কে” found in the ending part of any word then it will be as “Object1” attribute. Any word, which does not belongs to any other attribute and if the placement of that word is before the verb and if the word is not subject then it will be treated as “object2” attribute. The rest of the words will be treated as subject.

Finally, I have a sentence examining plan to translate properly is:

Subject + Adverb of Frequency + Verb + Object1 + Object2 + Object[n] +Destination1 + Destination2 + Destination[n] + Manner1 + Manner2 + Manner[n] + Place + Time + Reason.

Here n represents the number of inputs.

So, according to attribute we can convert any Bengali Sentence into English. The English conversion for above example according to the structure is as follows:

Bangladesh cricket team (subject) practices (Verb) cricket (object1) spontaneously (manner1) with new (adjective) coach (manner2) in stadium (place) in the morning (time) to win world cup (reason).

4. Goals

A large bilingual corpus of aligned sentence patterns will allow a far higher range of previously unseen text to be totally translated, or partially translated as a starting point for a human editor.

5. Methodology

In general terms, the approach will be as follows:

- Algorithm Construction with an integration of Artificial Intelligence.
- Knowledge Base Database Design and Data Management.

Previously many researchers considered the Sentence Alignment, Phrase Alignment, Sentence Pattern, Pattern Matching Technique and NP & Circumstances Matching etc[6][7]. Above all the Fuzzy Matching technique is also popularly used. But my proposal will work in slightly different way[8][9].

Finally, the above two steps of methodology is represented using the following steps and the logical flowchart also.

At first, code will get input from users.(Recommended that all the sentences must have punctuation. At present this is a limitation)

After getting input, the program will separate all sentences according to the punctuations.

Next all words will be separated inside the memory according to the space. This operation may perform using pointer-array programming.

Hereafter, code will make meaningful combination of words by the help of KB dictionary. Such as "অনুশীলন" is a word and "করে" is also a word but individually inside the sentence they cannot create any meaningful theme. "অনুশীলন করে" is a meaningful word and it has also an English word inside the proposed KB dictionary. This type of combined word is not available traditionally inside the market dictionary.

After that, program will arrange the word according to attributes(such as subject type, word manner(transitive verb, finite verb, non finite verb, adverb of frequency, manner1, manner2, destination1, destination2, place, time, reason, phrase, verbal noun, adjective, noun)) by the help of KB Dictionary proposed.

If the word does not exist in dictionary then code will try to find out suffix / বিভক্তি as article "টি, টা, খানা, খানি, টির" , "র, এর, কে" etc. in the end of that word.

If the code finds an article/বিভক্তি at the end of that word then code will again search that word without article in to the KB dictionary.

If that word is found then code will place “the” for the article and for the "র, এর”. We have inserted “s” in the last position of that word. Code will do nothing for the "কে” and it return to the step 3.

If that word is not in the KB dictionary then the code will decide that the "টি, টা, খানা, খানি, টির" etc is not article. It is a part of word. Example: সুইটি

Following this processes code will take all decision for the words.

Finally code will create clauses such as subject, adverb of frequency, verb (transitive, finite, nonfinite), object1, object2, manner1, manner2, destination1, destination2, place, time, reason, phrase, verbal noun.

In overall code will find out which word at the end because the sentence format depends on it.

If the code finds "TV" then it is a “transitive verb”. If the code finds "Phrase" then it is another type.

After that code will follow a structure for translation and code will place the translated word in that structure according to the manner and the structure such as (Subject + Adverb of Frequency + Verb + Object1 + Object2 + Destination1 + Destination2 + Manner1 + Manner2 + Place + Time + Reason).

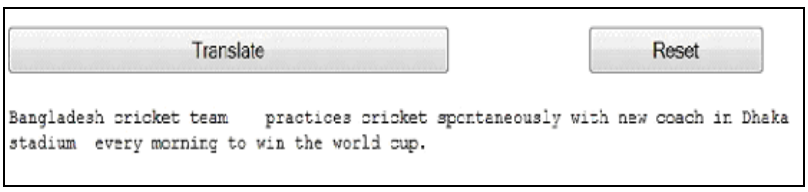

Figure 1. *Proposed Translator's Output*

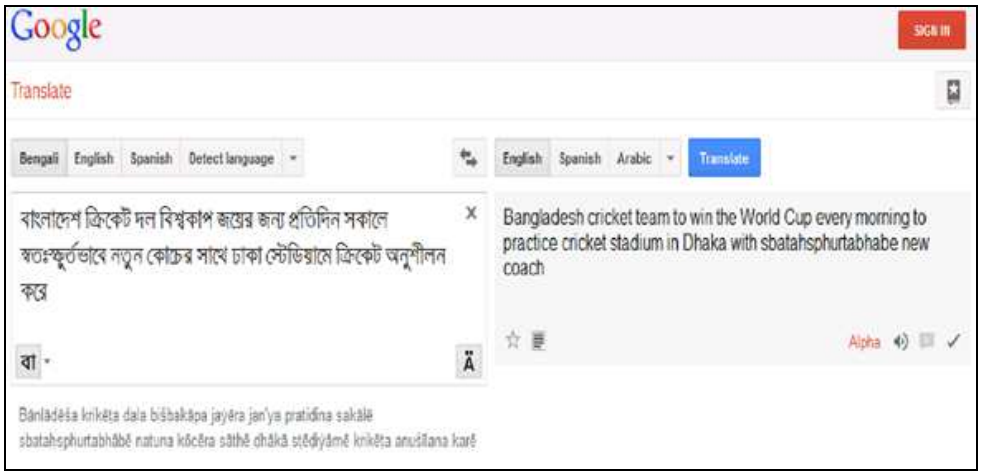

Figure 2. *One of the popular translators from Internet*

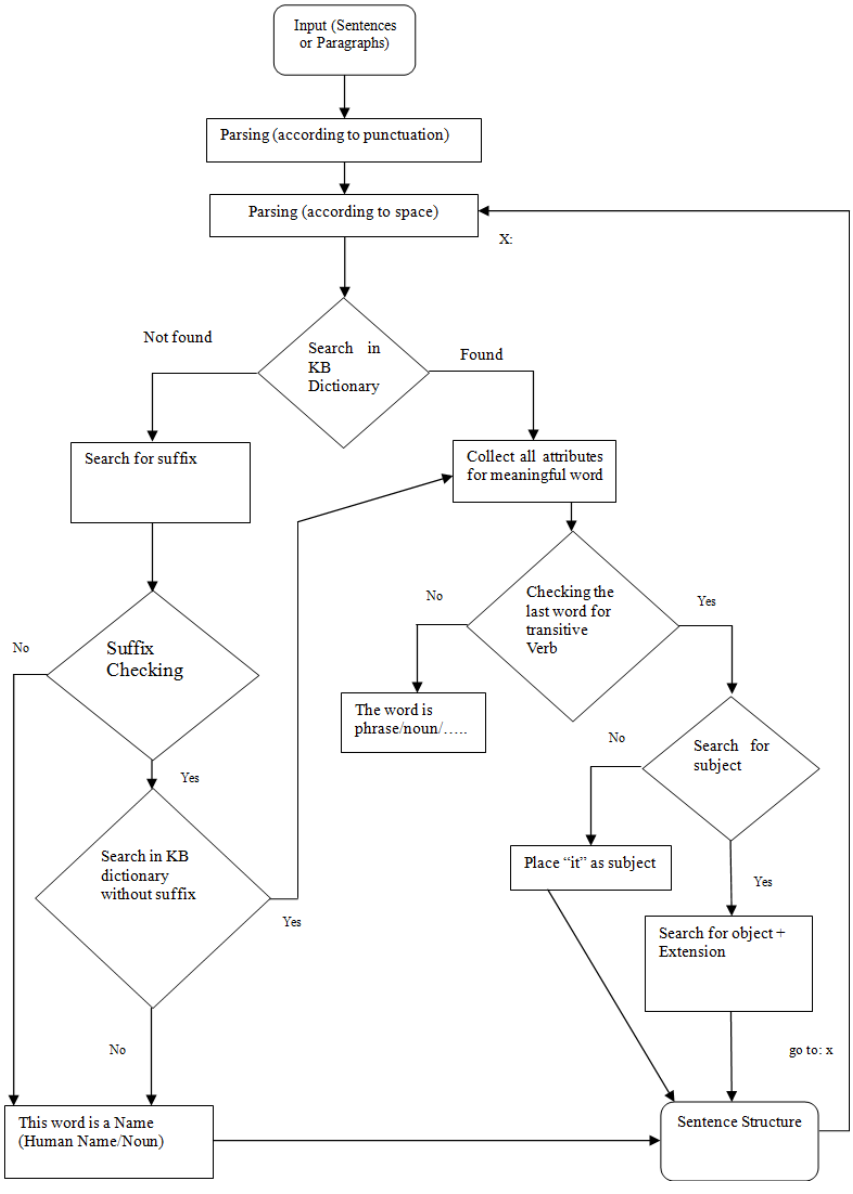

Figure 3. *Logical flow diagram of proposed Algorithm*

The discussed algorithm has partially developed using Asp.Net/C# with MS Access database. The output is shown in Fig. 1 and the recent popular internet base translator's output in Fig. 2.

References

[1] Birch, A. C. Callison-Burch, M. Osborne and P. Koehn. 2006. Constraining the Phrase-Based, Joint Probability Statistical Translation Model. In *Proceedings of the Workshop on Statistical Machine Translation, ACL*, pages 154–157, New York City.

[2] Carl, M. and S. Hansen. 1999. Linking Translation Memories with Example-Based Machine Translation. *Machine Translation Summit VII*, Singapore, pp. 617–624.

[3] European Association for Machine Translation (EAMT): http://www.eamt.org/iamt.php

[4] Hewavitharana, S. S. Vogel, A. Waibel. 2005. Augmenting a Statistical Translation System with a Translation Memory. In *Proceedings of EAMT 2005 Conference*.

[5] Hutchins, W.L. 2006. Future prospects in machine translation usage and research. Presentation in February 2006 at the University of Leeds, UK. Unpublished tutorial available at http://ourworld.compuserve.com/homepages/wjhutchins/Leeds-2006.pdf - last accessed on 28/09/2006

[6] Koehn, P., F.J. Och and D. Marcu. 2003. Statistical Phrase-Based Translation.

[7] Langlais, P. and M. Simard. 2002. Merging Example-Based and Statistical Machine Translation: An Experiment. S.D. Richardson (Ed.): AMTA 2002, LNAI 2499, pp. 104–113, 2002. Springer-Verlag Berlin Heidelberg 2002.

[8] Planas, E. 2000. Extending translation memories. Report, NTT Cyber Solutions Laboratories, Japan.

[9] Vogel, S. and H. Ney. 2000. Construction of a Hierarchical Translation Memory. In Proc. of COLING, pages 1131–1135.

Biography

Md. Ahsan Arif is an Assistant Professor of the Department of CSE, Asian University of Bangladesh, Bangladesh. He completed his graduation and post graduation from Computer Science and Engineering Discipline. He published 15 research articles in various reputed journals. Please find all the details at: http://md-ahsan-arif.blogspot.com/

Chandni Bhattacharya received her B.Sc. Engineering in Computer Science and Engineering from the Department of CSE, Asian University of Bangladesh. She is a student of the M.Sc. (Engg) in CSE program at the Dhaka University of Engineering and Technology, Dhaka, Bangladesh. Simultaneously, she is working as a Software Developer at Systech Digital Ltd., Uttara, Dhaka, Bangladesh.

A.B.M. Abdus Salam completed his B.Sc. Engineering in Computer Science and Engineering from the Department of CSE, Asian University of Bangladesh. He developed lots of Desktop and Web based application during his job career. Currently, he is working as a Software Engineer at Solution9 Ltd, Uttara, Dhaka, Bangladesh.

Suriya Islam completed her B.Sc. Engineering in Computer Science and Engineering from the Department of CSE, Asian University of Bangladesh.