An Efficient Automated English to Bengali Script Conversion Mechanism

Enakshi Mukhopadhyay[1], Priyanka Mazumder[2], Saberi Goswami[3], Romit S Beed[4]

Abstract

The authors aim at developing an efficient, unequivocal and automated method of generating Bengali language using English alphabets and simple English punctuation notes. Such art of writing Bengali language using English scripts shall be of immense help for those Bengali-speaking persons who cannot write in Bengali, yet can speak well and would require written communication in Bengali for official and personal conversation. Currently, Bengali keyboards are not available in the market, and accordingly, users desirous of writing in Bengali shall be at liberty of using conventional computer keyboards and automatically generate Bengali script using this system.

Keywords

Bengali language, jukhtakkhors, folas, kars, matras, Unicode, mapping, Vrinda font.

1. Introduction

Bengali language is the 6th most widely used writing system in the contemporary world which uses Bengali alphabet called *Bangla hôrôf* or *Bangla lipi* and ranks 7th among the most popular spoken languages of the world [1]. In terms of number, it has been estimated that there are at present about 250 million Bengali-speaking individuals spanning mainly in two countries: several provinces of India such as West Bengal, Tripura and parts of Assam, and Bangladesh in entirety including several million non-resident citizens of India and Bangladesh. There are currently two standard styles in Bengali language: the *Sadhubhasa* (Textual Speech) and the *Chalitbhasa* (Colloquial Speech). The Sadhubhasa was formulated by the language of early Bengali poetical works. With the passage of time, Sadhubhasa form of Bengali language underwent major transformation and emerged as the official literary and business language in early 19th century. Chalitbhasa, on the contrary, is based on the dialects of Kolkata (Calcutta) and its neighbouring areas on the bank of River Bhagirathi. *Chalitbhasa* started dominating since the early 20th century, and by the early 21st century it had become the dominant literary language as well as the standard colloquial form of speech among all.

It is to be noted that in Bangladesh, Bengali has been declared as the state language of the Republic in Article 3 of the Constitution of Bangladesh [2]. It is also clear that the Bangla Bhasha Procholon Ain (Bengali Language Implementation Act), introduced in 1987, specifies that Bengali should be made compulsory in courts and offices of Bangladesh since delivery of judgments in the form of court order or verdict written in English language often proved to be inconvenient for people not conversant with English language. Under such compulsion, the courts of law have started delivering judgments in Bengali- the language of the common people. With the widespread penetration of Computers in all Government offices and Courts these days, the necessity of some user friendly system of automatically generating Bengali Script has mounted putting immense pressure for new computer-literates to develop software that will cater to the demands under reference.

Although Bengali is spoken by 10% of people all over the world, there is a great percentage of people who can speak but cannot write in Bengali - mainly Bengali NRIs and the non-Bengali Community who have shifted to Bengal mainly for purpose of expanding their business. The latter never had any formal Bengali education. Therefore a suitable writing system becomes very necessary for these classes of people.

A suitable system of Bengali scripts therefore is a necessity for those who would wish to make use of written communication in Bengali formally or informally. As per a survey conducted recently in Kolkata on the residents whose native language is not

Manuscript received June 4, 2014.

Enakshi Mukhopadhyay, Computer Science, St. Xavier's College (Autonomous), Kolkata, India.
Priyanka Mazumder, Computer Science, St. Xavier's College (Autonomous), Kolkata, India.
Saberi Goswami, Computer Science, St. Xavier's College (Autonomous), Kolkata, India.
Romit S Beed, Computer Science, St. Xavier's College (Autonomous), Kolkata, India.

Bengali, it has been observed that majority of them cannot either read or write Bengali well but most of them (about 60%) strongly believe that being able to speak Bengali is an advantage in Kolkata [3]. Furthermore, 39% of the sample surveyed strongly disagree that Bengali is unavoidable in Kolkata while 49% strongly endorse that communications with native Bengali speakers are mandatorily conducted in Bengali.

Thus it is felt that an automated tool should be avaliable which should provide a solution to the problems of writing in Bengali using computers. Automated tool signifies the easy and user friendly approach of the concerned software. The non-availabality of ready made Bengali keyboards makes it quite difficult and sometimes impossible to type in Bengali. English keyboards are readily available and hence, writing, or more specifically, typing in English is not at all problematic. On the contrary typing in Bengali is not so smooth, especially when it comes to the representation of particular Bengali cases like Juktakkhors, Matras, Pholas etc. The concerned software also includes a help keymap for the easy utilization of the application by the users. A survey on the existing applications have revealed that many of them do not have proper keymaps to guide the users. With this application, users can take a look at the help keymap as and when needed, thereby avoiding errors and confusions. The mapping technique has been simplified as far as permissible. Different applications follow different mapping strategies.Some of these are very complex which might create a problem for the user. In this software, an attempt has been made to minimize these complexities so that the user remembers the easy rules of typing.

A very important feature that has been added in the said application is worth mentioning. Certain human errors have been removed by inclusion of an auto-correct feature. With this the user can avoid checking the correct spelling of the word and instead, type freely. The correct spelling will be automatically generated. It should be noted that eliminating all spelling problems is a tedious job. This application has taken a step towards spelling correction with the help of auto-correct feature. Hence it can be stated that this automated tool for writing in Bengali can be of immense use to the group of people who wish to or need to converse in Bengali through computers.

The corresponding Bengali script would automatically be generated as the user keeps on typing. The aim of this system shall be to release a full set of Bangla Script that supports all the major Bangla juktakhars (conjuncts) that are in use these days. The results of present work could be freely shared by all in need. Free Software is about empowering users, and about granting them rights over the software they use. Along with the conversion mechanisms and various other functionalities, a clear interface with simple yet attractive graphics will be part of the system under discussion.

2. Literature Review

In today’s world, with the sharp increase in globalisation, there has been an increase in the need for different translation tools from English to other languages as most documents/writing tools are in English [4]. Bengali, being the fourth most spoken language in the world[5], has been a matter of research since long. The microsoft specific guidelines [6] has been provided for the users dealing with Microsoft specific products on Bengali and this project has been developed following all conventions. The Jatiyo National standard keyboard has a lot of complexity and far from being similar to what the pronounciation is [7] [8]. Various software avaliable in the market today have used the keyboard combination and sequences for implementing the task efficiently and for this the key codes have been of special significance [9] . Many existing software are there which have been studied extensively. Certain bugs, inappropriate mapping and anamolies have been encountered. In this application various possibilities of typing a particular word has been taken care of for the easy utilisation by the users.

In a Bengali Text Generation Software, Ankur,[10] we have noticed that it is not compatible with all the browsers. In Quillpad software [11] it is seen that there is lack of provision for the users to know how to type specially the complex words having Juktakkhors. It is difficult to use it with ease because of the lack of assistance. Google translator [12] also is not free of bugs and it hardly converts properly and most of the letters are not recognised during conversion. Review of the existing softwares further revelead that the software provided by Tamilcube.com also exhibit anamoly which makes it tough to handle. Star21 [13] is another sofware we studied which though gives many option of convertion of one language to another but is not foolproof when actually put to use.

Few software which need to be downloaded are of such large size that they tend to make the systems slow and give error when used for conversion. This project is developed to overcomes, if not all, but most of the errors including special functionalities and other attributes which will make it efficient and effective for users.

Along with the work on translation from English to Bengali there has been considerable work on traslation from English to many other languages considering the high pace of globalisation. English to Hindi transaltion software also exists [14] along with English to Hindi dictionary [15] and vice versa [16]. English to Tamil dictionary also exists and these help in finding the corresponding Bengali or Tamil word corresponding to an English word.

3. Script Conversion Mechanism

System Model

Our work is to generate Bengali Script as the user types in English, keeping the Bengali pronunciations in mind. This is done by mapping the English characters to the unicodes corresponding to the Bengali characters. The basic system works as follows- the user types in English on the user interface page using a standard English keyboard. The application will perform the necessary mapping operations and generate the appropriate Bengali characters.

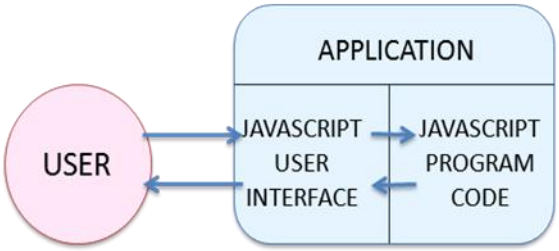


Figure 1 : Basic System Model

Complexities of Bengali Script

Unlike English language, Bengali consists of various chararacteristics. The JUKTAKKHORS, MATRAS, PHOLAS, KARS etc. are unique in case of Bengali script and English has no provision for them.

JUKTAKKHORS are special cases where two or more consonants are combined to represent a single letter. The representations of these characters are very complex. A MATRA is a horizontal line present at the top of certain Bengali characters. Several JUKTAKKHORS also have MATRAS. Single consonants and single vowels also have MATRAS. The use of vowels in Bengali differs from that of English. Most of the time a consonant and a vowel combine to form a single letter. The combined vowel looks different from the letter vowel and this form of a vowel is called a KAR. Like KARS, another typical characteristic of Bengali language is the use of PHOLAS. This is a situation where two consonants combine to form a single character.[17].

The way in which Bengali words are written sometimes differ from the way they are spelt. Matras are an integral part of bengali while writing each letter because their absence and presence and change the meaning of the letter all together. This logic is also very imprtant. These are indeed special cases and are not available in English Language. These fonts of JUKTAKKHORS are also very difficult to implement and while typing in english often they get jumbled up [18]. So this application has overcome this problem of displaying these complex JHUKTAKKHORS in a proper format. Even the Bengali punctuation differs. The fullstop notation in English is represented as a '|' or DARI.[19]

Unlike in western scripts where the letter-forms stand on an invisible baseline, the Bengali letter-forms instead hang from a visible horizontal left-to-right headstroke called মাত্রা matra. The presence and absence of this matra can be important. For example, the letter ত tô and the numeral ৩ "3" are distinguishable only by the presence or absence of the matra, as is the case between the consonant cluster ত্র trô and the independent vowel এ e. The Bengali script has ten Numerical digits (0 to 9). Bengali numerals have no horizontal headstroke or মাত্রা "matra". Bengali punctuation marks, apart from the downstroke daṛi (|), the Bengali equivalent of a full stop, have been adopted from western scripts and their usage is similar. Commas, semicolons, colons, quotation marks, etc. are the same as in English. The concept of using capital letters is absent in the Bengali script, hence proper names are unmarked.

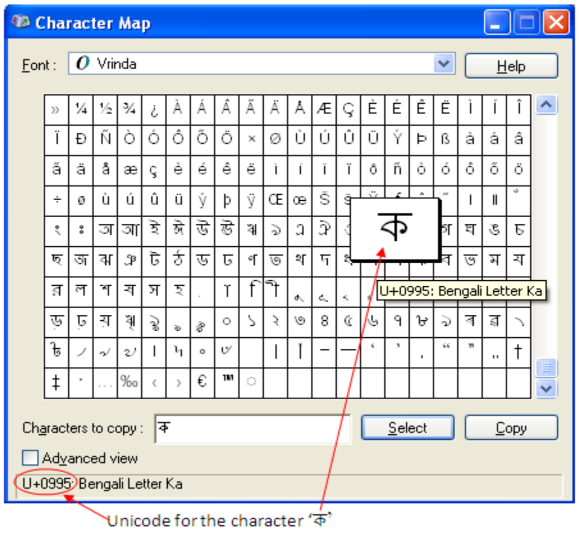
The following inconsistencies are inherent in the Bengali script and orthography. They often put additional burden on the person learning the script. The inconsistencies manifest themselves in various ways. Sometimes there are multiple different letters

or symbols for the same sound (over-production). Sometimes a letter loses its original sound value.
Like : ত and ৎ , শ, ষ and স , জ and য , ন and ণ

Bengali Characters Set:
Consonants: ক খ গ ঘ ঙ চ ছ জ ঝ ঞ ট ঠ ড ড় ঢ ঢ ণ ত থ দ ধ ন প ফ ব ভ ম য য় র ল শ ষ স হ ৎ ্
Independent vowels: অ আ ই ঈ উ ঊ ঋ এ ঐ ও ঔ
Vowel signs: া ি ী ু ূ ৃ ে ৈ ো ৌ
Combining marks: ঁ ং ঃ ় ্
Symbols & punctuation: ৳ । ॥
Numbers: ০ ১ ২ ৩ ৪ ৫ ৬ ৭ ৮ ৯
Other symbols in the Bengali block: ৲ ৴ ৵ ৶ ৷ ৸ ৹ ৺ ঽ ৰ ৱ ৗ ৠ ঌ ৡ ৢ ৣ ড় ঢ় য়

Character Mapping

The Bengali Unicodes have been used here for displaying Bengali characters. The Vrinda font contains the Bengali characters along with their respective Unicodes. The Vrinda font is available in all versions of Windows and can be accessed from the Character Map. [20,21] One can see for the vrinda font, the Unicode of ক ('\u0995 in javascript)

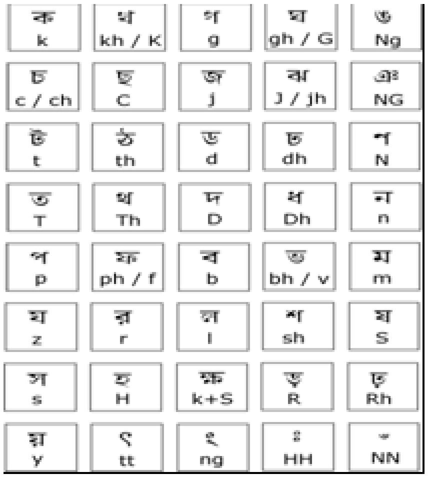


Character Map
Font : Vrinda
Help
ক
U+0995: Bengali Letter Ka
Characters to copy : ক
Select
Copy
□ Advanced view
U+0995: Bengali Letter Ka
Unicode for the character 'ক'

Figure 2 : Unicodes in the Character Map

and this is used to map the English 'k' onto the Bangla ক in the application. Hence when the user types the English alphabet 'k' from the keyboard then ক appears on the screen of the application. The mapping implementation of the Bengali characters can be summarized as follows:

Table 1 : Mapping Bengali to English



ক k	খ kh / K	গ g	ঘ gh / G	ঙ Ng
চ c / ch	ছ C	জ j	ঝ J / jh	ঞ NG
ট t	ঠ th	ড d	ঢ dh	ণ N
ত T	থ Th	দ D	ধ Dh	ন n
প p	ফ ph / f	ব b	ভ bh / v	ম m
য z	র r	ল l	শ sh	ষ S
স s	হ H	ক্ষ k+S	ড় R	ঢ় Rh
য় y	ৎ tt	ং ng	ঃ HH	ঁ NN

Generating Bengali involves typing the English characters in the same sequence as they are spelt. For example the user just needs to press them in a proper sequence and the corresponding Bangla word will be shown as output. The user should write "aa-kars" or "ee-kars" after the characters. Here are some examples:
ami = আমি mukhosh = মুখোশ machh = মাছ
Aj prik+sa sheS = আজ পরিক্ষা শেষ
Representation of words such as 'Trishna' and 'Ratri' are actually different in Bengali, although they sound or spell similar in English i.e., the "Tri" part. One uses a Bengali alphabet 'rofola' and the other 'rhi'. Therefore the Bengali spellings should be kept in mind and mechanisms should be improvised for representing the words using their appropriate Bengali spellings. Introduction of a 'khondetyo' as a Bengali character is necessary because our research concludes that no current Bengali application contains this Bengali character presently.
A portion of the English character input and its corresponding mapping to Bengali is given below:
Mapping Numbers:

```
phonetic['0']='\u09e6';//'shunno';
phonetic['1']='\u09e7';//'ek';
phonetic['2']='\u09e8';//'dui';
```

Mapping Vowels:

```
phonetic['II']='\u0988'; // dirgho
phonetic['e']='\u09C7'; // e kar
phonetic['E'] = '\u098F'; // E
phonetic['U'] = '\u0989'; // hrossho u
```

Mapping Similar Sounding Characters

```
phonetic['dh']='\u09A2'; // ddho
phonetic['b']='\u09AC'; // bo
```

```
phonetic['bh']='\u09AD'; // bho
phonetic['v']='\u09AD'; // bho
```

Keycodes

The main task involves processing the inputs typed through the keyboard. As an example if the SHIFT button is pressed along with key 'T', then 'ট' is generated. When 'T' is pressed without the SHIFT key the 'ত' is generated. Presented beneath is the KEYCODES which is used in the program to indentify the corresponding key strokes and map to the desired output character accordingly.

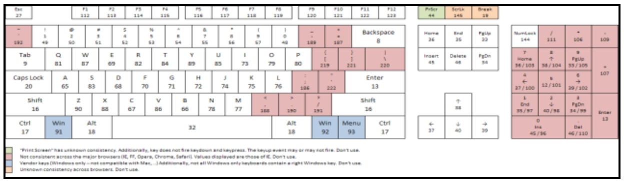


Figure 3: Key codes

Automization Rules

There exist some peculiarities of the Bengali characters. Certain special characters exist as a result of merging two consecutive characters. Similarly certain characters are never found to exist together. These are summarized below which are incorporated in the software.

- No Bengali word contains 'অ' between two consonants. Thus 'অ' is represented by an entirely different code say 'Ao' and separated from the other common vowel codes. However 'অ' can be used to begin a Bengali word.
- After র usually ন does not appear, instead it is usually followed by ণ, so we have designed our application such that even if the user types "n" corresponding to ন then too ণ appears.
- We have also noticed that if we type "o" in the beginning then we get ও but ও does not appear in the middle of a word usually so for correct the users even if they type "o" in the middle of a word, the provision of automatically redirecting it to o–kar.

4. Algorithm and Implementation

The broad steps are as follows:

- Accept input from the user.
- Pass input as parameter to a function.
- Map the input to its corresponding Unicode.
- For JUKTAKKHORS, MATRAS, KARS & FOLAS perform necessary operations.
- Generate the Bengali script as the output.

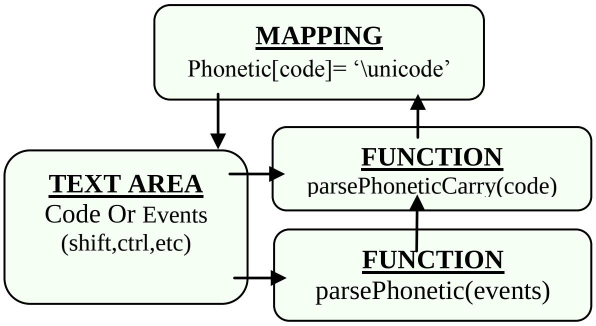


Figure 4: Block Diagram

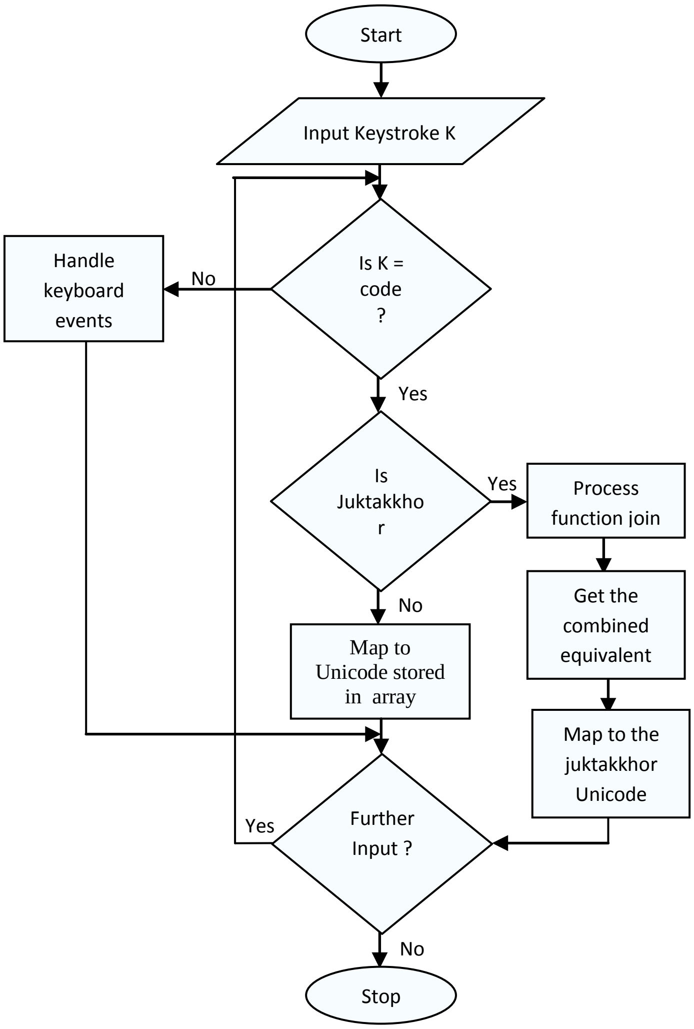


Figure 5: Flowchart

User Interface Functions

In the user interface part we have tried to implement the common functionalities like CUT, COPY and PASTE. NEW and OPEN has also been included to provide the user with a friendly environment.. The functions ‘Copy’ is given below:

```
function Copy()

{
       var field = document.getElementById("bangla");
       var startPos = field.selectionStart;
      var endPos = field.selectionEnd;
      var field_value = field.value;
      var field_value2 = field.value;
      var selectedText = field_value. substring(startPos,endPos);
      window.clipboardData.setData('Text',selectedText);
}
```

Variables Initialization

```
var carry = ''; //This variable stores each keystrokes
var old_len =0; //This stores length parsed bangla
charcter
var ctrlPressed=false;
var len_to_process_oi_kar=0;
var first_letter = false;
var carry2="";
isIE=document.all? 1:0;
var switched=false;
```

Functions for Checking Key Events

```
function checkKeyDown(ev)

{
 //just track the control key
 var e = (window.event) ? event.keyCode : ev.which;
if (e=='17')
{
 ctrlPressed = true;
}
else if(e==16)
 shift=true;
}

function checkKeyUp(ev)
{
//just track the control key
var e = (window.event) ? event.keyCode : ev.which;
if (e=='17')
  {
            ctrlPressed = false;
            //alert(ctrlPressed);
  }
}
```

Function For Parsing the Key Code

```
function parsePhonetic(evnt)
{
 // main phonetic parser
 var t = document.getElementById(activeta); // the
active text area
var e = (window.event) ? event.keyCode: evnt.which;
// get the keycode
if (shift)
{
var char_e = String.fromCharCode(e).toUpperCase();
// get the character equivalent to this keycode
            shift=false;
}
else
var char_e = String.fromCharCode(e); // get the
character equivalent to this keycode

lastcarry = carry;
carry += "" + char_e;        //append the current
character pressed to the carry

if ((phonetic['vowels'].indexOf(lastcarry)!=-1 &&
phonetic['vowels'].indexOf(char_e)!=-1) ||
(lastcarry==''  '' &&
phonetic['vowels'].indexOf(char_e)!=-1) )
{
      //let's check for dhirgho i kar and dhirgho u kar :
if(carry=='ii' || carry=='uu' || carry=='ee' || carry=='oo')
       {carry = lastcarry+char_e;}
else
            {
            char_e = char_e.toUpperCase();
            carry = lastcarry+char_e;
            }
}
```

Coding For Few Autocorrect Features

```
if(lastcarry=='S' && char_e=='n' )        // after
murdhonyo sho always murdhonyo no comes
                char_e=char_e.toUpperCase();

 if(lastcarry=='r' && char_e=='n' )        // after ro
always murdhonyo no comes
                char_e=char_e.toUpperCase();

if(char_e=='o' && lastcarry=='' ||
lastcarry==phonetic['vowels']) // if o comes frst den
letter o
     char_e=char_e.toUpperCase();
```

Code for Implementing The Juktakkhor

```
bangla = parsePhoneticCarry(carry); // get the combined equivalent
tempBangla = parsePhoneticCarry(char_e); // get the single equivalent

if (char_e=="+" || char_e=="="||char_e=="`")
{
  if(carry=="++" || carry=="=="||carry=="``")
   {
              insertConjunction(char_e,old_len);
              old_len=1;
              return false;
   }
//otherwise this is a simple joiner
insertAtCursor("\u09CD");
old_len = 1;
carry2=carry;
carry=char_e;
return false;
}
```

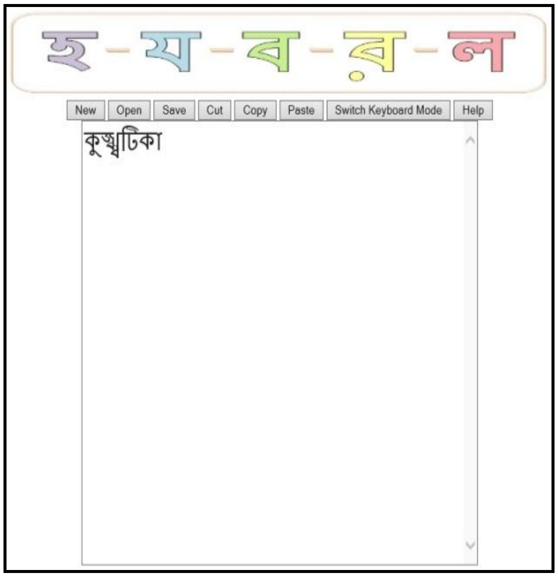
Clubbing

```
else if(carry=="Ao")
{
 // its a shore o
insertConjunction(parsePhoneticCarry("ao"),old_len)
;
old_len=1;
return false;
}
else if (carry == "ii")
{
// process dirgho i kar
              insertConjunction(phonetic['ii'],1);
              old_len = 1;
              return false;
}

else if (carry == "oI")
{
//oi kar
insertConjunction('\u09C8',old_len); //same treatment like ou kar (By manchu)
              old_len = 1;
              return false;
}

 else if (carry == "oo")
{
//dirghou kar
insertConjunction('\u09C2',old_len); //
              old_len = 1;
              return false;
}

       else if (carry == "oU")
{
// ou kar
insertConjunction("\u09CC",old_len);
              old_len = 1;
              return false;
}
```

Function for Passing the User Input to Generate the Bengali script

```
 function parsePhoneticCarry(code)
{
if (!phonetic[code]) //no bangla equivalent for this keystroke
    {return ''; //return a null value    }
else
  {  return  (  phonetic[code]);    //  found  bangla equivalent   }
 }
```

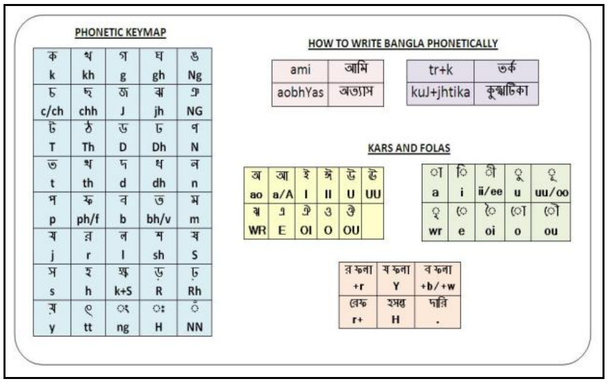
Graphical user interface



কুজ্ঝটিকা

Figure 6: User Interface

The following figure is a demonstration of the Help Screen. This is useful for naïve users who are not conversant with the existing mapping technique. This will help him to choose the corresponding characters through table observation.



PHONETIC KEYMAP

ক k	খ kh	গ g	ঘ gh	ঙ Ng
চ c/ch	ছ chh	জ J	ঝ jh	ঞ NG
ট T	ঠ Th	ড D	ঢ Dh	ণ N
ত t	থ th	দ d	ধ dh	ন n
প p	ফ ph/f	ব b	ভ bh/v	ম m
য j	র r	ল l	শ sh	ষ S
স s	হ h	ক্ষ k+S	ড় R	ঢ় Rh
য় y	ৎ tt	ং ng	ঃ H	ঁ NN

HOW TO WRITE BANGLA PHONETICALLY

ami	আমি
aobhYas	অভ্যাস

tr+k	তর্ক
kuJ+jhtika	কুজ্ঝটিকা

KARS AND FOLAS

অ ao	আ a/A	ই I	ঈ II	উ U	ঊ UU
ঋ WR	এ E	ঐ OI	ও O	ঔ OU	

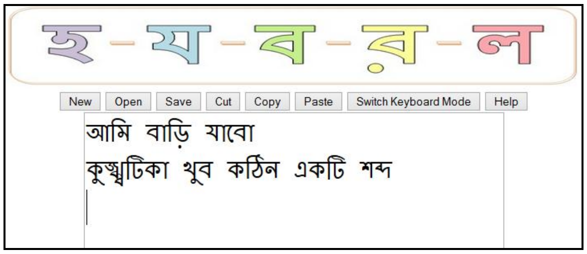
া a	ি i	ী ii/ee	ু u	ূ uu/oo
ৃ wr	ে e	ৈ oi	ো o	ৌ ou

র ফলা +r	য ফলা Y	ব ফলা +b/+w
রেফ r+	হসন্ত H	দাড়ি .

Figure 7: Help Screen

5. Results

First we take a simple example : "ami baRi jabo"
The output should be: "আমি বাড়ি যাবো "
We also take a complicated example which involves juktakkhors. In order to generate a word using juktakkhor we need to use joiners like '+' , '`' or '='. So in order to generate the word "কুজ্ঝটিকা" , we need to type kuJ+jh+bTika. The middle portion shows how that complex part has to be executed. Now using this convention we type the sentence: কুজ্ঝটিকা খুব কঠিন একটি শব্দ



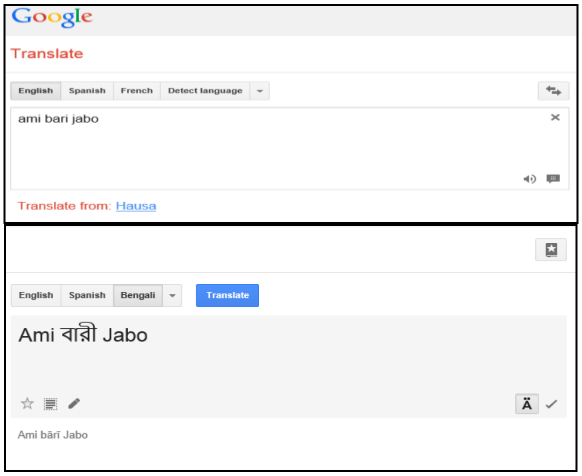
আমি বাড়ি যাবো
কুজ্ঝটিকা খুব কঠিন একটি শব্দ

Figure 8: Result using our Application

6. Comparison

Here we present few comparisons using some other Translation Software. We tried to generate the sentence আমি বাড়ি যাবো using those software but the results were anomalous. These are shown below:

Google Translate



ami bari jabo

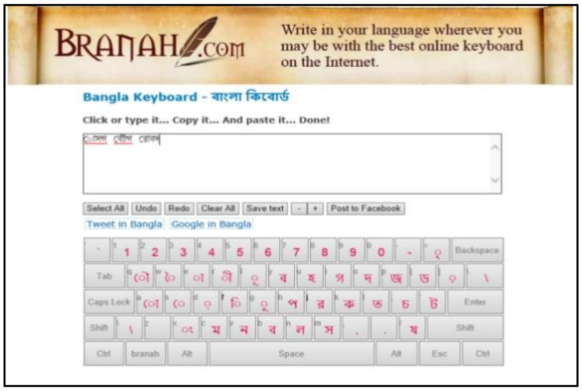
Translate from: Hausa

Ami বারী Jabo

Ami bārī Jabo

Figure 9: Result using Google Translate

www.branah.com



Write in your language wherever you may be with the best online keyboard on the Internet.

Bangla Keyboard - বাংলা কিবোর্ড

Click or type it... Copy it... And paste it... Done!

Figure 10: Result using our other applications

7. Conclusion and Future Scope

The aspects mentioned have been used for developing a system for generating Bangla Script in which an attempt has been made to overcome the complexities of implementing JUKTAKKHORS and MATRAS and develop an application with the ease of typing the bengali words as close to there pronunciation as posible In future we can add more features like editing text styles. We can even improve the autocorrect feature for more complex words. It can also be extended to create an English to Bengali dictionary or a simple **বাংলা অভিধান** that would provide users an additional benefit of finding the synonym of a Bengali word along with the ease of typing. We can also implement a speech recognition feature which would be able to generate the bengali script phonetically as the user speaks.

Acknowledgment

We would like to thank the Principal of St. Xavier's College (Autonomous) Rev. Dr. Fr. J. Felix Raj, S.J. for his continuous support and encouragement and the members of the Department of Computer Science for their guidance and help.

References

[1] Asiatic Society of Bangladesh(2003). Banglapedia, the national encyclopedia of Bangladesh. Asiatic Society of Bangladesh, Dhaka.

[2] Nahid Ferdouci, Bengali language situation in the judicial system in Bangladesh, T*he Dhaka University Journal of Linguistics*: Vol.2 No.3 February, 2009.

[3] *Aditi Ghosh, Language in Urban Society:Kolkata and Bengali,University of Kolkata, SOUTH ASIAN LANGUAGE REVIEWVOL. XV. No. 1. Jan 2005.*

[4] *Murphy Coy, Translating foreign language in SAS® with Google Translate, School of Information System, SMU, Singapore,* Paper 096-2012.

[5] Anshuman Pandey, *Language Support, tugboat, Volume 20, No. 2, 1999.*

[6] *Microsoft, Bengali (India) Style Guide.*

[7] Sneha Tripathi and Juran Krishna Sarkhel,Approaches to Machine Translation, Annals of library and Information Studies Vol-57, pp. 388-393, December 2010.

[8] Judith Francisca, Md. Mamun Mia, Dr. S. M. Monzurur Rahman, Adapting rule based machine translation from english to bangla, Indian Journal of Computer Science and Engineering (IJCSE), Vol. 2 No.3 Jun-Jul 2011.

[9] Charles Bigelow and Kris Holmes, *the design of a unicode font, electronic publishing, vol. 6(3), 289-305, September 1993.*

[10] www.modular-infotech.com/html/index.html.

[11] www.quillpad.in/index/html#.U3nFZKiSzko.

[12] https://translate.google.co.in/#auto/bn/.

[13] www.star21.com/translator/english/bengali.

[14] Rashmi Gupta, Nisheeth Joshi and Iti Mathur, Analysing quality of english-hindi machine translation engine outputs using Bayesian classification, International Journal of Artificial Intelligence & Applications (IJAIA), Vol. 4, No. 4, July 2013.

[15] Hindi to English Glossary, http://www.columbia.edu/itc/mealac/pritchett/00urduhindilinks/shacklesnell/325hindienglish.pdf.

[16] The student's practical dictionary, http://www.nptidurgapur.com/pdf%20files/English-hindi_Dictionary.pdf.

[17] Md. Ahsan Arif,Md. Mobarak Hossain,Arif Tanvi, Algorithm for Natural Language Processing: A Bengali Language Perspective, ARPN Journal of Systems and Software, VOL. 3, NO. 6, October 2013.

[18] William Radice, Teach Yourself Bengali, Hodder & Shoughton, ISBN 0-340-86029-4.

[19] Prof. B.B. Chaudhury, Resource Centre for Indian Language Technology Solutions – Bangla, Indian Statistical Institute, Kolkata.

[20] Addison-Wesley, The Unicode Standard 4.0, the Unicode Consortium , 2003.

[21] Davis, Mark. 2001. Unicode standard annex #19: UTF-32. Version 3.1.0. Cupertino, CA: The Unicode Consortium.

Enakshi Mukhopadhyay received her B.Sc(Hons) degree with 1st class in Computer Science from Asutosh College under Calcutta University and also completed her M.Sc(Hons) with 1st class in Computer Science from St. Xavier's College (Autonomous) under Calcutta University, Kolkata, India in 2014. Her areas of interest are data structure and programming.

Priyanka Mazumder received her B.Sc(Hons) degree with 1st class in Computer Science from Asutosh College under Calcutta University and also completed her M.Sc(Hons) with 1st class in Computer Science from St. Xavier's College (Autonomous) under Calcutta University, Kolkata, India in 2014. Her areas of interest are programming and networking.

Saberi Goswami received her B.Sc(Hons) degree with 1st class in Computer Science from Bethune College under Calcutta University and also completed her M.Sc(Hons) with 1st class in Computer Science from St. Xavier's College (Autonomous) under Calcutta University, Kolkata, India in 2014. Her areas of interest are programming and network security.

Romit S Beed completed his M.Tech in Computer Sc and Engg from the University of Calcutta in 2005 after doing his M.Sc in Computer Sc from the same University. He is an Assistant Professor in the Department of Computer Sc., St. Xavier's College, Kolkata from 2005. Presently he is the Coordinator of the Post Graduate Department of Computer Science. His research areas are DBMS, Software Engineering and Network Security.