Tamil Character Recognition, Translation and Transliteration System

M. Prakash, Apoorva Ojha, Priyanshu Raman

Abstract – Optical Character Recognition (OCR) is the machine conversion of handwritten or typed data into machine encoded scripts. English, is the informal link to all the regional languages in India and is used to publish reports, papers, magazines and records. Monolingual systems are incapable of doing so, thus increasing the need of bilingual frameworks. We have created a character recognition system that converts the user's input in the Tamil language to English. Additionally, we can also perform transliteration of Tamil to English and vice versa. The bilingual OCR system, "MOZHI VALLAAN," which has an accuracy of 94%, will be used.

Keywords- Optical Character Recognition, Translation, Transliteration, Longest Common Subsequence, Inception V3.

I. INTRODUCTION

Considerable amounts of document scanners used nowadays come with software that performs character recognition. In our endeavor, we try to recognize the words and alphabets of Tamil. There is no software available currently, which adequately gives us the desired results. A large portion of India still does not understand English. So to converse with the people of that specific area, local languages and dialects have to be used. Even though OCR is a small part of digital image processing, its use in the professional world is huge. It can be used in financial institutes like banks and are also utilized in libraries. There is a far less number of OCRs for Indian languages when compared to the English language. Indian scripts have several symbols, and hence, recognition is a difficult task. After recognizing the characters, we provide an application that translates from Tamil to English and also performs transliteration (phonetic translation) from both Tamil to English and vice versa. Currently, there are no such applications that provide both transliteration and translation together. Our project attempts to perform this task. In this system, we are providing an application that helps to retrieve Tamil characters from images. Our dataset contains images of different Tamil characters in various types of handwriting. There are around 6000 such images. Spelling and semantic checks can be incorporated to correct the error at a stroke, character, and word level. After recognizing these characters, the user may choose to translate them from Tamil to English.

Apart from translation, our application can also perform transliteration, which is basically the phonetic translation of the words. It allows you to transliterate from Tamil to English and vice versa.

Revised Manuscript Received on April, 25 2020.
* Correspondence Author

Priyanshu Raman*, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India.

Apoorva Ojha, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India.

Dr. M. Prakash, Associate Professor, Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India.

II. LITERATURE SURVEY

Ref. No.	Algorithms / Strategies Used	Pros	Cons	Future Scope	Author
[1]	DTW based classification	Handles large data	Handwriting style variations.	Possess more sensitivity to data heterogeneity	N.Joshi, G.Sita, et al
[2]	Shape Feature Database Finite Set Automation	1. Sensitivity 2. Specificity	Two-Dimensional Structure of scripts.	Classification results from independent classifiers	Aparna et al.
[3]	Support Vector Machines	1.Accuracy 2.Sensitivity 3.Precision	Stroke features are not at all reliable as there is a lot of variation in handwriting	Sampling and training efficiently datasets.	Toselli et al.
[4]	HMM	Real-time implementation	Matras or vowel symbols are present at left, right, base, top, or even as different parts around the base consonant.	Matra identification and training datasets accordingly.	Swetha Lakshmi et al.
[5]	SVM with Radical Basic Function	1.Sensitivity 2.Accuracy 3.Specificity	It cannot detect noises.	Precision to be increased.	Suresh Sundaram et al.

2.1 SUMMARY

Recognizing hand-written Indian scripts is a demanding task because of the presence of complex characters and symbols. The framework of the contents and the various types of signs and composing styles cause unnecessary hassles and thus require different procedures for feature representation and recognition. Stroke features are not at all reliable as there is a lot of variation in handwriting. Due to the limitations of the application or the speed of writing, it is possible for a single character to be split into various parts, hence creating uncertainties. Matras or vowel symbols are present at left, right, base, top, or even as different parts around the base consonant. Presence of a large number of characters.

III. SCOPE OF THE SYSTEM

Our application provides three functions. First, it recognizes the character or words entered by the user and then it can both translate and transliterate. This framework can help us in digitizing the archaic scripts and documents which might get damaged if not kept properly. Digitization of the old texts can help us recover these bygone documents. Since everyone cannot understand Tamil, we are also providing a translator that translates these Tamil characters and words into English. Furthermore, we will also be performing transliteration, which can help the people who can speak Tamil but cannot understand it. Also, our structure aims to provide these two functions in the same application, which has not been done before.

4. TAMIL CHARACTER RECOGNITION, TRANSLATION AND TRANSLITERATION

4.1 Description of the Work

Tamil is the official language in 3 countries India, Sri Lanka and Singapore. Tamil was written using a script called vattluttu. The current Tamil script has 12 vowels, 18 consonants and one special character, the aytam.

It has a total of 247 characters. Some of these characters are given below(Figure 1)

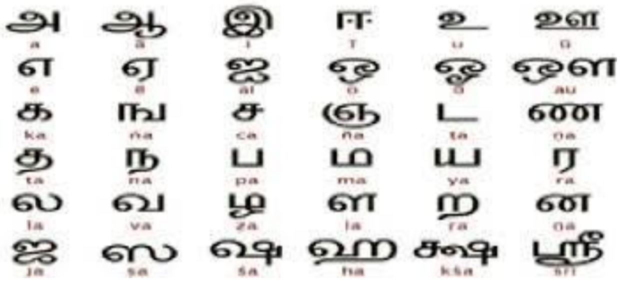


Figure 1. Some Tamil characters

4.2 Transliteration

4.2.1 Preprocessing Phase

The input is given in the form of sentences either in English or Tamil. This process is carried out at word level; words are split into tokens, removing all the punctuators.

4.2.2 Unicode to Language Conversion

The foremost step is to change the text format to the target language. The database is used; all the Unicode characters of Tamil Text are then converted to the English Language.

The sounds of the word for a given Tamil word are mapped to English words based on phonetics, and the same is stored in a table for maximum possible words of English. Before this, when all the terms were considered, it produced highly noisy results. Now, the length is fixed to less than equal to length+5.

4.2.3 Grapheme and Phoneme Based Indexing

In other languages, unlike Tamil for the transliteration process, only the alphabets are involved. In Tamil, both graphemes and phonemes are involved. When the Tamil Language is considered, they require more than one letter as most of the alphabets are adjoint with each other. For example, the alphabet ka in Tamil is the mixture of k(consonant) and a(vowel). Phoneme Based Indexing (Soundex Algorithm) It involves giving a unique code to each grammar based on the sound, and similar words are given almost similar laws to index and compare. It is a hashing system wherein all the words starting with a similar alphabet are given the same code.

4.2.4 Final Phase in Transliteration

The most challenging step of the phase is to identify the correct phonetic for a given the word from the lot. This step is done by Longest Subsequence Score, every word is assigned with a score, the longest subsequence it matches with, the higher is the score. The name which matches the most, i.e., with the highest score is selected. Let w be the letter that goes through the terms T {t1,t2,t3,...tn}. The final term l' is assigned to Ti which has the highest Longest Common Subsequence.

$$l' = \{ti \ \varepsilon \ T|LCS\ (w,ti) >= LCS(w,tj)\ \forall\ tj\ \varepsilon\ T\}$$

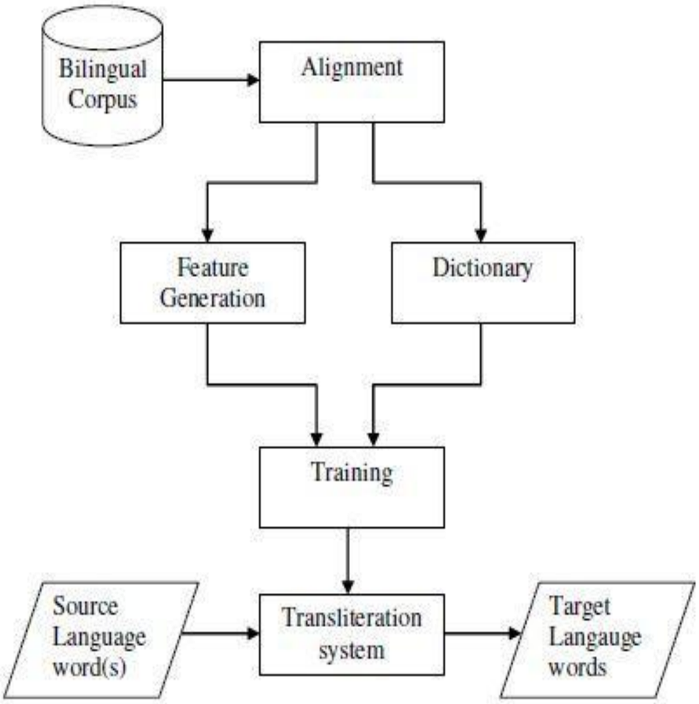


Figure 2. Transliteration system

Figure 2 shows the transliteration system. Given a bilingual corpus, the alignment process assesses and decides which parts of source language and target languages belong together, puths them side by side and forms high quality translation. Using this translation segment, features are extracted from the translation segment. Then the model is trained using the extracted feature set, after the training phase, for a given source language the model yields the transliterated target language.

4.2.5 Difference between the Proposed Model and Previously Available Model

All the other available models either provide the translation of the transliteration process. Not much work has been done in the field of giving both of these on the same platform. We present to you a system wherein both the translation and transliteration process will be done at the same platform within a fraction of seconds, on a single click.

4.3 Approach for Translation

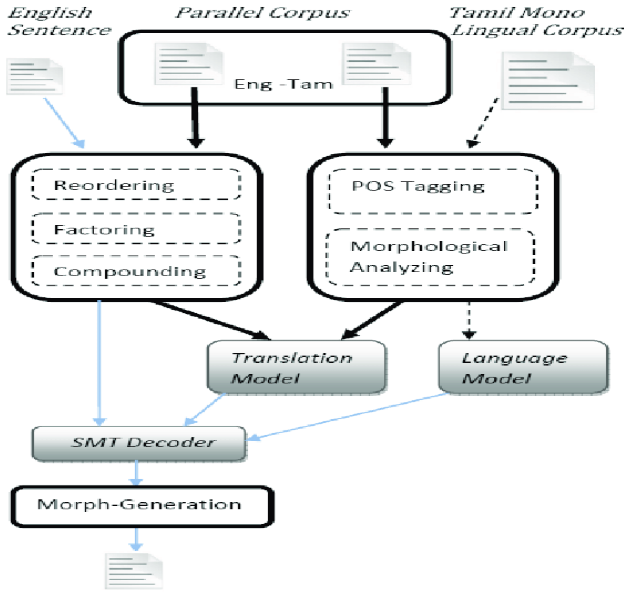


Figure 3. Translation system

Figure 3 shows the architecture of the translation system. The foremost step in the translation process is Reordering. As the grammatical structure in different languages are different, the tokens in the sentences may be reordered for better understanding and comprehension of the reader. Factoring is the process of identifying the right meaning of a particular word using the words before and after that particular word, the more the training data, the better is the accuracy. POS Tagging refers to marking a word in a corpus to a corresponding part of speech tag. Finding the sequence of tags which is most likely to generate a given word sequence. Morphological analysis, a method of exploring all the possible solutions in a multidimensional problem. Herein, it deals with identifying the right meaning of the word and translates it into the target language. Statistical Machine Translation, is a translation paradigm where translations are generated on the basis of trained bilingual corpora, and hence using the decoder the given word/sentence is then translated to target language which is Tamil here.

4.3.1 Converting words into Lexicons

The lexicons are created using the morphological transducer present in the application. This helps in generating the right inflectional forms. Various other markers such as number, gender, and person markers like “aar,” “aarkaL,” “atu,” “ana,” etc. This information helps us in determining the relationship between the subject and the verb of a sentence.

4.3.2 Syntactic structures considered by the application

The list given below identifies the underlying structures which are accounted for by the syntactic parser of the system. These are-

i) A simple sentence with a subject, verb and prepositional phrases

ii) Noun phase with adjectives and a noun

iii) Clause sentences with a Wh operator.

iv) Prepositional phrase consisting of a noun phrase and a proposition.

v) A verb phrase consists of a verb and an and verb

For example, if we convert a word into its syntactic structures-

yoocanaiyum - [“nom”, “noun”, “conj”]
oru - [“adj”, “oru”]

4.3.3 Algorithm used for translation

The algorithm used for conversion is **Inception V3.** The goal of the inception module is to act as a “multi-level feature extractor” by computing 1×1, 3×3, and 5×5 convolutions within the same. The module of the network — the output of these filters are then stacked along the channel dimension and before being fed into the next layer in the system.

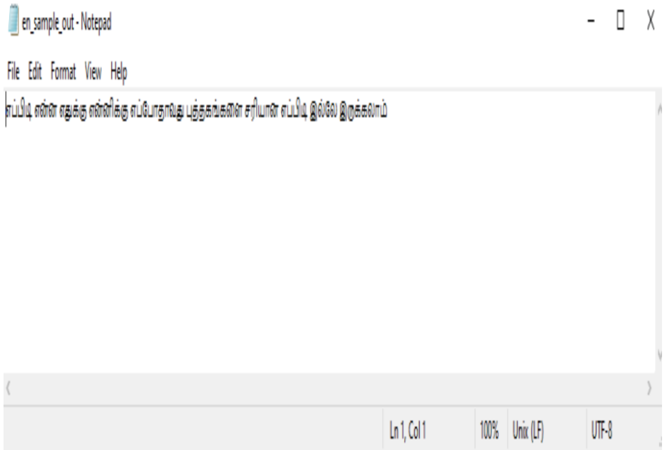
IV. EXPERIMENTAL ANALYSIS AND RESULTS

5.1 Transliteration and Translation

The files containing Tamil characters are prefixed with ta_, and those involving English characters are prefixed with en_.

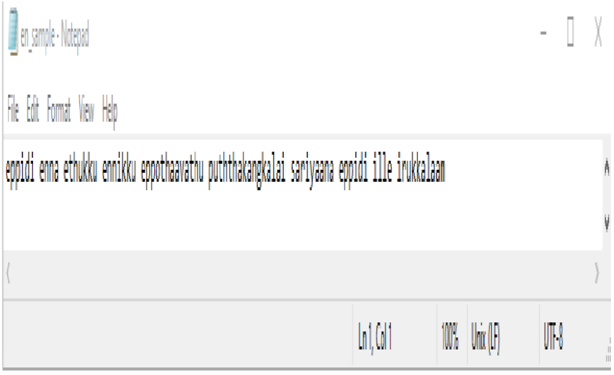
Upon performing the transliteration process, this will produce output in files suffixed _out.txt in the same directory. Upon executing the translation process, it will provide output files suffixed _translate.txt on the same list.

5.1.1 Transliteration



எப்பிடி என்ன எதுக்கு என்னிக்கு எப்போதாவது புத்தகங்களை சரியான எப்பிடி இல்லே இருக்கலாம்

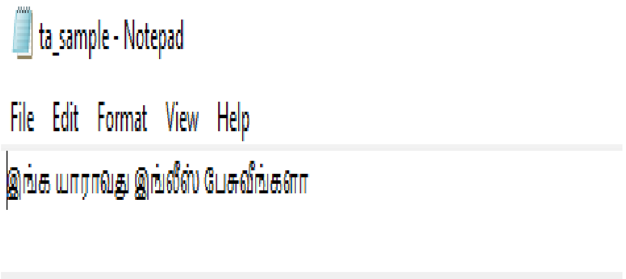
Figure 4. Input for transliteration



eppidi enna ethukku ennikku eppothaavathu puththakangkalai sariyaana eppidi ille irukkalaam

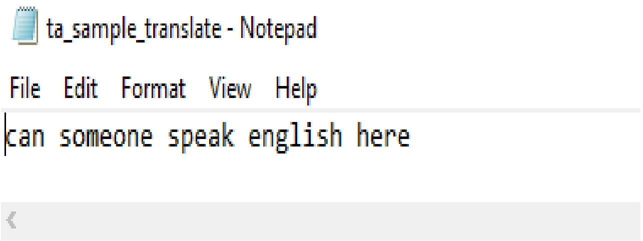
Figure 5. Output for transliteration

5.1.2 Translation



இங்க யாராவது இங்லீஸ் பேசவீங்களா

Figure 6. Input for translation



can someone speak english here

Figure 7. Output for translation

V. CONCLUSION

All the available software or platforms provide the translation of the transliteration process. Not much work has been done in the field of giving both of these on the same platform. We present to you a system wherein both the translation and transliteration process will be done at the same platform within a fraction of seconds, on a single click. The results achieved are very encouraging in terms of measures and the proposed translations and transliterations themselves are well built. Although this work has been done exclusively for Tamil-English languages, this work can be extended to other languages too.

Future Work

i) The work reported on Online HCR for Tamil script may be extended in various directions.

ii) Spelling and semantic checks can be incorporated to correct the error at a stroke, character, and word level.

iii) Incremental learning can be incorporated so that a new stroke or character can be incorporated without the requirement of retraining the entire system

iv) Online Handwritten character recognition for Mobile devices.

REFERENCES

1. Niranjan Joshi, G.Sita, A.G.Ramakrishnan, Sriganesh Madhvanath, “Tamil Handwriting Recognition using Subspace and DTW based Classifiers,” Proceedings of 11th International Conference (ICONIP-2004) published by Springer Berlin Heidelberg, pp. 806- 813, 2004.
2. K.H.Aparna, Vidhya Subramanian, M.Kasirajan, G.Vijay Prakash, V.S.Chakravarthy, “Online handwriting recognition for Tamil,” Proceedings of 9th International Workshop on Frontiers in Handwriting Recognition (IWFHR-9 2004) published by IEEE Computer Society Press, pp.438–443, 2018.
3. Alejandro H.Toselli, Moises Pastor, Enrique Vidal, “On-Line Handwriting Recognition System for Tamil Handwritten Characters,” Iberian Conference on Pattern Recognition and Image Analysis IbPRIA 2007: Pattern, Recognition and Image Analysis, pp 370-377, 2015.
4. Swethalakshmi, “Online Handwritten Character Recognition for Devanagari and Tamil scripts using Support Vector Machines,” Master of Science Thesis, Department of Computer Science and Engineering, Indian Institute of Technology, Madras, 2008.
5. Suresh Sundaram, A G Ramakrishnan, “An Improved Online Tamil Character Recognition Engine using Post-Processing Methods,” Proceedings of 10th International Conference on Document Analysis and Recognition(ICDAR 2009) published by IEEE Computer Society Press, pp.1216-1220, 2016.
6. Mantas, J, “An overview of character Recognition methodologies”, Pattern recognition, vol .19, no. 6, pp. 425-430, 1986.
7. 7)Govindan, V.K.and A.P.Shivaprasad, 1990 Character Recognition- A Review, Pattern Recognition, 23 (7): 671-683.
8. Pal, U, B.B.Chaudhuri, “Indian Script Character Recognition: a Survey”, Pattern Recognition, vol. 37, pp. 1887-1899, 2004.
9. R. Plamondon, S.N.Srihari, “On-line and Offline-hand written recognition: a comprehensive survey,” IEEE Transactions on PAMI, vol.22, no. 1, pp. 63–84, 2000.
10. R. Plamondon, D.Lopresti, L.R.B.Shoemaker, R. Srihari, “On-line Handwriting Recognition,” Encyclopedia of Electrical and Electronics Eng., J.G.Webster, ed., vol. 15, pp.123-146, 1999.
11. R. M.Bozinovic and S.N.Srihari, “Off-line cursive script word recognition,” IEEE Transactions on Pattern Anal. Mach. Intell., vol. 11, no. 1, pp. 68-83, 1989.
12. Hu, M.K.Brown and W.Turin, “HMM-based on-line handwriting recognition,” IEEE Trans. OniPattern Anal.Mach.Intell., vol. 18, no. 10, pp.1039-1045, 1996.
13. D. Deng, K.P.Chan, and Y.Yu, “Handwritten Chinese character recognition using spatial Gabor filters and self-organizing feature maps,” Proc.EEE Inter.Confer.On Image Processing, vol. 3, pp. 940-944, Austin TX, 1994.

14. C-H. Chang, “Simulated annealing clustering of Chinese wordsfor contextual text recognition,” Pattern Recognition Letters, vol. 17, no. 1, pp.57- 66, 1996.

AUTHORS PROFILE

Priyanshu Raman is in the senior year of his undergraduate degree in Computer Science and Engineering at S.R.M. Institute of Science and Technology, Kattankulathur, Chennai. His interests include Data Science, Machine learning and Data Mining. He currently has a CGPA of 7.22. He is also a student member of IET (The Institution of Engineering and Technology).

Apoorva Ojha is in the senior year of her undergraduate degree in Computer Science and Engineering at S.R.M Institute of Technology, Kattankulathur, Chennai. Her interests include Data Science, Artificial Intelligence and Cyber Security. She currently has a CGPA of 9.07. She is also a student member of IET (The Institute Of Engineering and Technology).

Dr. M. Prakash received the Doctor of Philosophy from Anna University, Chennai, India. He received the Master of Technology degree in Information Technology from Anna University, Chennai, India. He is currently working as Associate Professor in the Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, Tamil Nadu, India. He has 13 years of experience in teaching and learning. His research interest includes Big data analytics, Databases and Security. He is a professional member of ISTE, IE(I), ISCA, IAENG, CSTA, IACSIT and UACEE