ISSN: 2395-7639

International Journal of Multidisciplinary Research in Science, Engineering, Technology & Management (IJMRSETM)

(A Monthly, Peer Reviewed Online Journal)

Visit: www.ijmrsetm.com

Volume 6, Issue 4, April 2019

Machine Translation of English to Hindi : A Literature Survey

Chinmay Bodhe, Harshita Singh

Dept. of CSE, D.Y.Patil College of Engineering, Akurdi, Pune, India

ABSTRACT: Transliteration becomes quite challenging when it comes to convert texts such as names and technical terms from one script to another using approximate phonetic or spelling equivalent of the other language. In this paper we are addressing development of a system to recognise the lyrics of songs and displaying in either English or Hindi after transliterating into the user preferred language. The proposed system uses Audio Recognition as a part of Natural Language Processing for the input and further extraction of the sound inventories of the lyrics and their conversion in required form using generative model in several stages. As a detail description to the music media running, brief information is displayed such as album genre, artist, composer and other attributes to the playing song. We describe and evaluate the process of transliteration of songs from English to Hindi or vice versa for complete understanding of pronunciation of the words in preferred language.

KEYWORDS: Transliteration, Natural Language Processing, Information Extraction, Machine Translation

I. Introduction

In the diverse cultural difference, transliteration proves to be a vital element in preserving the phonological structure of the words. As with time, music has managed to build a bridge between several cultures removing the scriptural barrier up to certain extent, transliteration provides further tools to retrieve the originality of language and singing. Majority of the national population takes interest in English songs but often crooked pronunciation reduces the confidence to sing for the regional Hindi speakers. To eliminate such barriers, we are introducing a system which not only provides preferable linguistic options but also helps to enunciate the proper phonemes of both English and Hindi songs. In this paper we aim to solve music recognition and lyrics display as NLP task using segments like text classification, information extraction techniques and other attributes n machine transliteration. The proposed exercise of transliteration is executed by mapping word by word from one system to another manually. Input to the system is initialised using voice recognition and lyrics of the song are taken in account to get equivalent spelling to its pronunciation using the unique codes of the language. Furthermore the mapping deduces the transliterated song dynamically while being played in the system. Since the phonology of the language is protected syllable by syllable, new words are formed based on each input which may or may not be presented in the registered dictionary, hence such kind of mapping results in a successful accuracy of high rate in the entire transliteration process.

There exists many research works in the field of machine transliteration in the name entity recognition or translation. Several techniques are used in conversion of text scripts from one language to another in order such as SMT (Statistical Based Translation) approach, FST (Finite State Transducers) approach, Hybrid approach, etc.

II. HYBRID APPROACH

Hybrid Approach was used to develop a system for English-Hindi name entity transliteration by [1] using Stanford NER tool. Their proposed work used hybrid method to provide transliteration of English to Hindi language by breaking words for accurate phonemes and then processing the merge of produced word to get output. In order to implement a transliteration system for English Hindi, first analyzed the spellings of different words and found that for Indian languages, most people use different spellings for the same words and all these spellings are taken to be correct. For example, let us consider the word □□□□ . For this Hindi word, different people would use the following spellings: Bharat, Bharath, Bhaaratha, and Bhaarat. This is a non- exhaustive list and there can be many more variations to this word. Since this is a general phenomenon and cannot be captured by a rule based approach to transliteration. They tried to apply a hybrid approach to this. First we defined rules to capture phonemes of the English words.It was identified that a word can be divided in seven different phonemes which are a group of vowels (V) and consonants (C). Once this was done, we collected the text from the web. We generally used news sites for this. We collected 10,000 sentences from these sites. In order to identify name entities we used Stanford's NER tool for name entity extraction. In all we extracted 42,371 name entities. Table II shows thestatistics of these name entities. After extraction of name entities, we applied our phonetic algorithm onto them and extracted different phonemes. Then each English phonemes was transliterated into Hindi, thus this created a knowledgebase of English and Hindi phonemes. Here, Prob(Hindi) was the probability of the Hindi(phoneme, while Count(English, Hindi) was the count of the number of times a combination of a English and Hindi phonemes were seen in the knowledge base and Count(English) was the total count of the occurrence of a particular English phoneme in the knowledgebase. Thisprocess has led to attain the accuracy up to 83.04% in the transliteration of language.

III.ERROR DRIVEN LEARNING

Another approach on transliteration based on errors- driven Learning is used by [2] Ying using Direct Orthographical Mapping (DOM). This method of machine transliteration is applied by segmenting a word according to syllables and then mapping them directly into the target language without considering its pronunciation. They studies the performance of two- stage machine transliteration based on Conditional Random Fields. The idea of error-driven learning is something like Boosting . First a learner C1 was built on part of training data G1. And then the first classifier is used on a new dataset. Wrong prediction will occur without a doubt. The first classifier should learn more from the error samples if we want to improve the ability of the classifier. Therefore we put these samples not being classified correctly into G1 to form a new training set named G2. Then G2 is used to train a new learner C2. If C2 is stronger than C1 when tested on another data open, it proves helpful when error prediction data of C1 is added into the training set. Then test C2 on new data and repeat such procedure until the performance of classifier does not increase or there is no more data to be added. CRF-based binary classifier can do the chunking with over 90% accuracy. Therefore the hard problem focuses on the stage of mapping the chunks into target strings. To cut down the complexity of computation meanwhile outputting the global prediction, train CRFs classifiers step by step, on each new step focusing on the error prediction of the current classifier until the performance reduces or the limitation of the hardware of PC. Transliteration based on error-driven learning reduces the complexity of computation in CRFs model training, but still outputs the global prediction.

IV.APPLYING FINITE RULES TO DATA

English- Hindi transliteration by applying finite rules to databefore training , transliterationd result is increased.The hypothesis carried forward with initial parallel corpus where Hindi data is in UTF notation and is converted to wx format to achieve transliteration. The rules are applied to bilingual corpus of Indian Name Entities with source in English and target in Hindi script. The phrase based statistical machine

translation is used to train the English- Hindi corpus for improved results over UTF format.A series of experiments, to show that, after applying finite rules to the data before training, transliteration result get increased. The format of our data during the experiment is in baseline format. The idea behind using the baseline format is 'character-to- character' alignment. We perform training, tuning and testing using Moses and Stanford Phrasal. Initially we start our work with a parallel corpus of English-Hindi, where Hindi data is in UTF-8 notation. After performing experiments in UTF format, converted the target side data from UTF to wx format using the UTF8_wx converter to achieve transliteration. Then moved to the final experiment of applying finite rules to the data. The target data is in wx- notation, we know that, if a word in source data ends with 'a' will definitely ends with 'A' in when transliterated in Hindi (wx-notation). So, the idea is applying one more 'a' to the end of these words, so that the machine will automatically understand when 'aa' encounters it have to transliterate to 'A' in target notation. For automatic evaluation BLEU for the entire test set of 400 words was used. BLEU measures the precision of n-grams with respect to the reference

translations, with a brevity penalty. A higher BLEU score indicates better translation. After performing experiments it was observed that the scores got improved in wx-notaion over UTF-8 and more when we apply finite rules to data before training. In Moses, the scores get improved by 33.5% on wx- notation over UTF-8 and 37.4% on finite rule experiment over UTF-8. In Stanford Phrasal, the scores get improved by 27.1% on wx-notation over UTF-8 and 29.1% on finite rule experiment over UTF-8. The future scope involves the work on corpus consist of Indian as well as foreign names and be able to transliterate from English to Hindi script (UTF or wx-notation) with the same phoneme. For example: Aby

According to Indian style phoneme: □□□

According to foreign style phoneme: □□□

V. BUCKWALTER'S TRANSLITEARTION SYSTEM

Syntax directed translator for English to Hindi language was proposed by [4] Pankaj Kumar, Sheetal Srivastava and Monica Joshi reflecting on existing MT projects such as ANGLABHARTI, MATRA and UCSG MAT which are nationally used for translation. Their work focuses on the automatic translation of text using compilation technique through syntax generator translator. This method is trained by using specific words from encyclopaedia and dictionary maintained exclusively for meaningful Hindi words. This helps in having an intelligent translation process which makes sense in both the languages equally. the Java port of the homonym product developed in Perl by Tim Buckwalter, it works with a transliteration of the Arabic word. This transliteration uses Buckwalter's transliteration system. It includes Java classes for the morphological analysis of Arabic text files, whatever their encoding.

VI. CONTEXT INFORMED PHRASE-BASED

The transliteration system was modeled by translating characters rather than words as in character-level translatio systems. They used a memory-based classification framework that enables efficient estimation of these features while avoiding data sparseness problems. The experiments were both at character and transliteration unit (TU) level and reported that position - dependent source context features produce significant improvements in terms of all evaluation metrics. In this way the problem of machine transliterationwas successfully implemented by adding source context modeling into state-of-the-art log-linear phrase-based statistical machine translation (PB-SMT). In their experiment, they also showed that by taking source context into account, improve the system performance substantially.

VII. CONCLUSION

In this paper work, we have presented a survey on developments of different machine transliteration systems for Indian languages especially English to Hindi. Additionally we tried to give a brief idea about the existing approaches that have been used to develop machine transliteration tools. From the survey we found out that almost all existing Indian language machine transliteration systems are based on statistical and phoneme basedapproach. The main effort and challenge behind each and every development is to design the system by considering the agglutinative and morphological rich features of language to make the system more efficient forransliteration.

REFERENCES

1. Hybrid Approach to English-Hindi Name Entity Transliteration Shruti Mathur and Varun Prakash Saxena IEEE, 2014
2. Machine Transliteration Based on Error-driven Learning YingQin ICALP, 2012
3. English- Hindi Transliteration by applying finite rules to data before training using StatisticaMachine-Translation Mitali Haldar,Anant DevTyagi ICCCE, 2012 (IEEE, 2013)
4. Amitava Das, Asif Ekbal, Tapabrata Mandal and Sivaji Bandyopadhyay (2009), 'English to Hindi Machine Transliteration System at NEWS', Proceedings of the 2009 Named Entities Workshop, ACL-IJCNLP 2009, page 80-83, Suntec, Singapore.
5. Semantic Similarity/Relatedness for Cross Language Plagiarism Detection Hanane Ezzikouri, Mohammed Oukessou, Mohammed Erritali IEEE, 2016
6. MusiXmatch lyrics API is a robust service that permits to search and retrieve lyrics in the simplest possible way-Https://developer.musix.com/documentation
7. Shazam Application is a magical mobile app that recognises music, TV and media. It's the best way to discover, explore and share music and TV
8. TVShachi Mall, Innovative Algorithms for Parts of Speech Tagging in Hindi-English Machine TranslationLanguage,International Conference on Green Computing andInternet of Things, IEEE, 2015