IMPROVING THE QUALITY OF GUJARATI-HINDI MACHINE TRANSLATION THROUGH PART-OF-SPEECH TAGGING AND STEMMER ASSISTED TRANSLITERATION

Juhi Ameta[1], Nisheeth Joshi[2] and Iti Mathur[3]

[1]Department of Computer Engineering, Cummins College of Engineering for Women, Pune, Maharashtra, India
[2,3]Department of Computer Science, Apaji Institute, Banasthali University, Rajasthan, India

[1]juhiameta.trivedi@gmail.com
[2]nisheeth.joshi@rediffmail.com
[3]mathur_iti@rediffmail.com

ABSTRACT

Machine Translation for Indian languages is an emerging research area. Transliteration is one such module that we design while designing a translation system. Transliteration means mapping of source language text into the target language. Simple mapping decreases the efficiency of overall translation system. We propose the use of stemming and part-of-speech tagging for transliteration. The effectiveness of translation can be improved if we use part-of-speech tagging and stemming assisted transliteration. We have shown that much of the content in Gujarati gets transliterated while being processed for translation to Hindi language.

KEYWORDS

Stemming, transliteration, part-of-speech tagging

1. INTRODUCTION

Transliteration is a process that transliterates or rather maps the source content to the target content. While we design a translation model, transliteration proves to be an effective means for those words which are multilingual or which are not present in the training corpus. For a highly inflectional Indian language like Gujarati, naive transliteration i.e. direct transliteration without any rules or constraints, does not prove to be very effective. The main reason behind this is that suffixes get attached to the root words while forming a sentence.

We propose the use of stemming and POS-Tagging (i.e. Part-of-Speech Tagging) for the process of transliteration. Stemming refers to the removal of suffixes from the root word. Root word is actually the basic word to which suffixes get added. For example, in સ્ત્રીઓમાંથી (striiomaaNthii) the root is સ્ત્રી and the suffix is ઓમાંથી.These modules prove to be beneficial in the Natural Language Processing environment for morphologically rich languages.

The rest of the paper is arranged as follows: Section 2 describes the previous history of the related work which is followed by Section3 which describes the proposed work. Evaluation and Results have been focused on in Section 4. Finally we conclude the paper with some enhancements for future work in Section 5.

2. LITERATURE REVIEW

Stemming was actually introduced by Lovins [1] who in 1968 proposed the use of it in Natural Language Processing applications. Two more stemming algorithms were proposed by Hafer and Weiss [2] and Paice [3]. Martin Porter [4] in 1980 suggested a suffix stripping algorithm which is still considered to be a standard stemming algorithm. Another approach to stemming was proposed by Frakes and Baeza- Yates [5] who proposed the use of term indexes and its root word in a table lookup. With the improvement in processing capabilities, there was a paradigm shift from purely rule-based techniques to statistical/ machine learning approaches. Goldsmith [6][7] proposed an unsupervised approach to model morphological variants of European languages. Snover and Brent [8] proposed a Bayesian model for stemming of English and French languages. Freitag [9] proposed an algorithm for clustering of words using co-occurrence information. For Indian languages, Larkey *et al.* [10] used 27 rules to implement a stemmer for Hindi. Ramanathan and Rao [11] used the same approach, but used some more rules for stemming. Dasgupta and Ng [12] proposed an unsupervised morphological stemmer for Bengali. Majumder *et al.* [13] proposed a cluster based approach based on string distance measures which required no linguistic knowledge. Pandey and Siddiqui [14] proposed an unsupervised approach to stemming for Hindi, which was mostly based on the work of Goldsmith.

Considering the research work for part-of-speech tagging, Church [15] proposed n-gram model for tagging, which was then extended as HMM by Cutting *et al.* [16] in 1992. Brill [17] proposed a tagger based on transformation-based learning. Ratnaparkhi [18] proposed Maximum Entropy algorithm. Many researchers have recently proposed taggers with different approaches. Ray *et al.* [19] have proposed a morphology-based disambiguation for Hindi POS tagging. Dalal *et al.* [20] have proposed Feature Rich POS Tagger for Hindi. Patel and Gali [21] have proposed a tagging scheme for Gujarati using Conditional Random Fields. A rule-based Tamil POS-Tagger was developed by Arulmozhi *et al.* [22]. Arulmozhi and Sobha [23] have developed a hybrid POS-Tagger for relatively free word order language. Similarly for Bangla, Chowdhury *et al*. [24] and Sediqqui *et al*. [25] have done significant research in the area of POS-Tagging. Antony and Soman [26] used kernel-based approach for Kannada POS-Tagging. Again a paradigm shift has been observed from purely rule-based schemes to statistical techniques. Taggers for many Indian languages have been proposed but still more work needs to be done as compared to European languages.

Moving towards the work for transliteration, Kirschenbaum and Wintner [27] have proposed a lightly supervised transliteration scheme. Arababi *et al*. [28] used a combination of neural net and expert systems for transliteration. Praneeth *et al*. [29] at LTRC, IIIT-H proposed a language-independent schema using character aligned models. Malik *et al*. [30] followed a hybrid approach for Urdu-Hindi transliteration. Joshi and Mathur [31] proposed the use of phonetic mapping based English-Hindi transliteration system which created a mapping table and a set of rules for transliteration of text. Joshi *et al.* [32] also proposed a predictive approach of for English-Hindi transliteration where the authors provided a suggestive list of possible text that the user entered. They looked at the partial text and tried to provide possible complete list as the suggestive list that the user could accept or provide their own input text. The use of transliteration has been proposed by many researchers for natural language processing and information retrieval applications.

3. PROPOSED WORK

Gujarati is a highly inflectional language as stated earlier. It has a free word-order. There are three genders in Gujarati- Feminine, Masculine and Neuter/Neutral. Suffixes get added to the stems giving the various morphological variants of the same root word.

We propose the use of stemming and POS-Tagging for the purpose of transliteration. Figure 1 shows our system.

```
Let S be the sentence in source language
TRANSLITERATE (S)
I.    Tokenize S
II.   For each s in S
III.        Do matching of s in POS-Tagged Corpus
IV.         If match found,
V.                  then call STEMMER (s), go to step VII
VI.         Else call STEMMER (s), go to step VIII
VII.        Transliterate with Tag rules to resolve ambiguity if present, go to step IX
VIII.       Transliterate without Tag rules, go to step IX      //Direct
                                                                Mappping of
                                                                source word
                                                                into target word
IX.   No more tokens, go to step X
X.    Display the sentence

STEMMER(n)
I.    Match n with suffix list                                  //list of 202 stemming rules
II.   If match found
III.        then remove suffix from n
IV.         t=n, go to step VI
V.    Else  t=n, go to step VI
VI.   Return t
```

Figure1. Transliteration assisted with stemming and part-of-speech tagging

Many ambiguities are observed while we design a translation model from Gujarati-Hindi. One such ambiguity is differentiation of the suffix ે in different cases. Suppose we have the sentence

રામે મને રિપોર્ટ આપી.	→	राम ने मुझे रिपोर्ट दी।
(Raame mane riport aapii.)		(Raam ne mujhe riport dii.)
(Meaning: Ram gave me the report.)		
મારા ઘરે એક બિલાડી છે.	→	मेरे घर पर एक बिल्ली है।
(Maaraa ghare ek bilaadii chhe)		(Mere ghar par ek billi hai)
(Meaning: There is a cat at my home.)		

If these two sentences are carefully observed, the suffix serves different purpose. Hence it is the tag that makes a difference here. રામે is a proper noun and ઘરે is a locative noun**.** Hence to differentiate if a tagged corpus is applied, then during translation if the meanings are not available in the corpus and only the tags are available then the transliterated text will be the actual translation. Similarly, the suffix ીએ poses an ambiguity.

ચાલો ઘેર ચાલીએ.	→	चलो घर चलें।
(Chaalo gher chaaliie.)		(Chalo ghar chaleN.)
(Meaning: Let us go home.)		
રશ્મીએ કિતાબ આપી.	→	रश्मी ने किताब दी।
(Rashmiie kitaab aapii.)		(Rashmii ne kitaab dii.)
(Meaning: Rashmi gave the book.)		

ચાલીએ is a verb whereas, રશ્મીએ is a proper noun.

We created a raw corpus of 5400 POS-tagged sentences and used 202 stemming and tagging rules to assist transliteration. The POS-Tagged corpus is a collection of text files having the sentences in the source language in the form- word_part-of-speech, e.g. પ્રતિબંધ_NN. The strings in the source language are first checked in the tagged corpus so that the word class can be obtained and then stemming is applied which ensures the extraction of the correct root. Transliteration is hence first refined by these modules. So whenever there is an ambiguity in suffixes (i.e. stemming process), corresponding tags resolve the problem of transliteration. These modules can hence help in ambiguity resolution If the corresponding tag is not found in the tagged corpus, naive transliteration is done where direct mapping from the source language into the target one is applied.

4. EVALUATION AND RESULTS

We tested our system on a total of 500 Sentences. The observed results are as follows:

Total number of Sentences tested	500
Total number of words tested	7500
Words for which transliteration and translation are same	4086
Percentage of words for which translation and transliteration are same	54.48
Number of words for which transliteration was wrong	518
Percentage of words for which transliteration was wrong	6.91
Percentage efficiency of transliteration	93.09

Table 1.Table showing evaluated results

Hence for 54.48% of Gujarati words translation and transliteration are same. The efficiency of our transliteration scheme is 93.09% (about 90%).

5. CONCLUSION AND FUTURE WORK

We followed a hybrid approach – a mix of rule-based and corpus-based approach, where we used POS-Tagged corpus and stemming rules to assist the process of transliteration. We achieved 93.09% overall efficiency of the transliteration scheme which makes it a promising approach. It was observed that 54.48% of the Gujarati words have the same translation and transliteration. Such a scheme not only reduces length of the corpus for the translation model

but also it helps in ambiguity resolution. It can be used for other morphologically rich Indian languages as well. As an immediate extension to this work, we plan further to include machine learning approaches and focus on each and every aspect of the scheme so that more accuracy in the transliteration process can be achieved.

REFERENCES

[1] J. B. Lovins, (1968) "Development of Stemming Algorithm", *Mechanical Translation and Computational Linguistics*, Vol. 11, No. 1, pp. 22-31.

[2] M. Hafer and S. Weiss, (1974) "Word segmentation by letter successor varieties", *Information Storage and Retrieval*, Vol. 10, No. 1, pp. 371-385.

[3] C. Paice, (1974) "Another Stemmer", *ACM SIGIR Forum*, Vol. 24, No. 3, pp 56-61.

[4] M. F. Porter, (1980) "An algorithm for suffix stripping", *Program*, Vol. 14, No. 3, pp. 130-137.

[5] W. B. Frakes and R. Baeza-Yates, (1992) "Information Retrieval: Data Structures and Algorithms", Prentice Hall, USA.

[6] J. Goldsmith, (2001) "Unsupervised learning of the morphology of a natural language", *Computational Linguistics*, Vol. 27, No. 2, pp. 153-198.

[7] J. Goldsmith, (2006) "An algorithm for unsupervised learning of morphology", *Natural Language Engineering*, Vol. 12, No. 4, pp. 353-371.

[8] M. G. Snover and M. R. Brent, (2001) "A Bayesian model for morpheme and paradigm identification", in *Proc. of 39th Annual Meeting of the Association of Computational Linguistics*, pp. 482-490.

[9] D. Freitag, (2005) "Morphology induction from term clusters", in *Proc. of 9th Conference on Computational Language Learning*, pp. 128-135.

[10] L. S. Larkey, M. E. Connell and N. Abduljaleel, (2003) "Hindi CLIR in Thirty Days", *ACM Transactions on Asian Language Information Processing*, Vol. 2, No. 2, pp. 130-142.

[11] A. Ramnathan and D. Rao, (2003) "A Lightweight Stemmer for Hindi", in *Proc. of Workshop on Computational Linguistics for South Asian Languages*, *10th Conference of the European Chapter of Association of Computational Linguistics*, pp. 42-48.

[12] S. Dasgupta and V. Ng, (2006) "Unsupervised Morphological Parsing of Bengali", *Language Resources and Evaluation*, Vol. 40, No. 3-4, pp. 311-330.

[13] P. Majumder, M. Mitra, S. K. Parui, G. Kole, P. Mitra and K. Datta, (2007) "YASS: Yet another suffix stripper", *ACM Transactions on Information Systems*, Vol. 25, No. 4, pp. 18-38.

[14] A. K. Pandey and T. J. Siddiqui, (2008) "An unsupervised Hindi stemmer with heuristic improvements", in *Proc. of 2nd Workshop on Analytics for Noisy Unstructured Text Data*, pp. 99-105.

[15] K. W. Church, (1988) "A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text", in *Proc. of Second Conference on Applied Natural Language Processing*, Austin, Texas, February 1988, Association for Computational Linguistics.

[16] D. Cutting, J. Kupiec, J. Pedersen and P. Sibun, (1992) "A Practical Part-of-Speech Tagger", in *Proc. of Third Conference on Applied Natural Language Processing*, ACL, pp. 133-140.

[17] E. Brill, (1995) "Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part of Speech Tagging", *Computational Linguistics*, December 1995, Vol. 21, No. 4, pp. 543-565.

[18] A. Ratnaparkhi, (1996) "A maximum entropy model for part-of-speech tagging", in *Proc. of the Empirical Methods in Natural Language Conference*.

[19] P. R. Ray, V. Harish, A. Basu, S. Sarkar, (2003) "Part of speech tagging and local word grouping techniques for natural language parsing in Hindi", in *Proc. of ICON 2003*.

[20] A. Dalal, K. Nagaraj, U. Swant, S. Shelke and P. Bhattacharyya, (2007) "Building Feature Rich POS Tagger for Morphologically Rich Languages: Experience in Hindi", in *Proc. of ICON 2007*.

[21] C. Patel and K. Gali, (2008) "Part-Of-Speech Tagging for Gujarati Using Conditional Random Fields", in *Proc. of the IJCNLP-08 Workshop on NLP for Less Privileged Languages*, Hyderabad, India, pp. 117–122.

[22] P. Arulmozhi, L. Sobha and K. Shanmugam, (2004) "Parts of Speech Tagger for Tamil", in *Proc. of the Symposium on Indian Morphology, Phonology & Language Engineering*, Indian Institute of Technology, Kharagpur, pp. 55-57.

[23] P. Arulmozhi and L. Sobha, (2006) "A Hybrid POS Tagger for a Relative Free Word Order Language", in *Proc. of the MSPIL-06*, Indian Institute of Technology, Bombay, pp. 79-85.

[24] M. S. A. Chowdhury, N.M. Minhaz Uddin, M. Imran, M.M. Hassan and M. E. Haque, (2004) "Parts of Speech Tagging of Bangla Sentence", in *Proc. of the 7th International Conference on Computer and Information Technology (ICCIT)*, Bangladesh.

[25] M.H. Seddiqui, A. K. M. S. Rana, A. Al Mahmud and T. Sayeed, (2003) "Parts of Speech Tagging Using Morphological Analysis in Bangla", in *Proc. of the 6th International Conference on Computer and Information Technology (ICCIT)*, Bangladesh.

[26] P. J. Antony and K.P. Soman, (2010) "Kernel based part of speech tagger for Kannada", (2010) in *Proc. of Machine Learning and Cybernetics (ICMLC)*, Vol. 4, pp. 2139 –2144.

[27] A. Kirschenbaum and S. Wintner, (2009) "Lightly supervised transliteration for machine translation", in *Proc. of 12th Conference of the European Chapter of the ACL (EACL 2009)*, pp. 433–441.

[28] M. Arbabi, S. M. Fischthal, V. C. Cheng and E. Bart, (1994) "Algorithms for Arabic name transliteration", *IBM Journal of Research And Development*.

[29] P. Shishtla, V. Surya Ganesh, S. Subramaniam and V. Varma, (2009) "A Language-Independent Transliteration Schema Using Character Aligned Models At NEWS 2009".

[30] A. Malik, L. Besacier, C. Boitet and P. Bhattacharyya, (2009) "A Hybrid Model for Urdu Hindi Transliteration", in *Proc. of the 2009 Named Entities Workshop*, ACL-IJCNLP, pp. 177–185.

[31] N. Joshi and I. Mathur, (2010) "Input Scheme for Hindi Using Phonetic Mapping", in *Proc. of the National Conference on ICT: Theory, Practice and Applications*.

[32] N. Joshi, I. Mathur and S. Mathur (2010) "Frequency Based Predictive Input System for Hindi", in *Proc. of the International Conference and Workshop on Emerging Trends in Technology*, ACM, pp 690-693.

AUTHORS

Juhi Ameta has completed her M.tech. in Computer Science from Banasthali Vidyapith, Rajasthan and is a Gold-medalist of her batch. She is currently working as an Assistant Professor at Cummins College of Engineering for Women, Pune, India. Her research interests include Natural Language Processing and Machine Translation. She has worked on EILMT Project funded by DIT, Govt. of India. Her research paper entitled "A Lightweight Stemmer for Gujarati" was published by CSI, Annual National Conference, Ahmedabad Chapter, December 2011.

Nisheeth Joshi is a researcher working in the area of Machine Translation. He has been primarily working in design and development of evaluation Matrices in Indian languages. Besides this he is also actively involved in the development of MT engines for English to Indian Languages. He is one of the expert empanelled with TDIL programme, Department of electronics Information Technology (DEITY), Govt. of India, a premier organization which foresees Language Technology Funding and Research in India. He has several publications in various journals and conferences and also serves on the Programme Committees and Editorial Boards of several conferences and journals.

Iti Mathur is an Assistant Professor at Banasthali University, India. Her research interests are in the area of Ontological engineering, soft computing, machine translation, and information retrieval. She is also a Co-Principal Investigator of English to Indian Language Machine Translation Development System Funded by Govt. of India. The project is a consortium mode project, where 13 institutions are developing machine translators from English to 8 different Indian languages. She has published several papers in natural language processing and information retrieval. She is also Editorial Board Member/Program Committee Member and reviewer of various journals and conferences. She is a Member of IEEE, USA, ACM, USA and CSI, India.