TEXT NORMALIZATION of CODE MIX and SENTIMENT ANALYSIS

Shashank Sharma
IIIT Bhubaneswar
Odisha, India
a113019@iiit-bh.ac.in

PYKL Srinivas
IIIT Bhubaneswar
Odisha, India
a114011@iiit-bh.ac.in

Rakesh Chandra Balabantaray
IIIT Bhubaneswar
Odisha, India
rakesh@iiit-bh.ac.in

***Abstract* - The field of getting insights from various text forms such as feedback, opinions, blogs and classifying them based on their polarity as positive or negative is known as sentiment analysis. But from last few years we find huge amount of code - mix (mixture of two languages) text available on social media. This text is available in Romanized English format in Indian social media, which is the transliteration of one language into another, which demands normalization to get further insights into the text. In this paper, we have presented various methods to normalize the text and judged the polarity of the statement as positive or negative using various sentiment resources.**

Keywords - code mix, roman script, sentiment, normalization, phonetic

I. INTRODUCTION

Customers give feedback with the delivery of a service or the quality of goods they use. The feedback or opinion of the social customer involves perception, emotion, and unexpected behavior. This feedback is valuable for the company to create new or revise existing services and solutions. This is becoming the fastest and most cost-efficient way to build a significant customer service experience. Creating that direct relationship with the customer can be achieved by reaching out through social media. In today's online business world, a whopping 62 percent of customers have already used social media for customer service issues. These feedbacks are given independent of any formal performance review process. The terminology to understand the feedbacks or gaining insights from social media is commonly referred as sentiment analysis.

The recent challenge in understanding the feedbacks in social media sites is that the text is written in code - mix format. Code mix is the mixture of two languages written in Roman script. Here a word of one language is phonetically represented into another and there are no rules available to get the exact spelling of these words. So we find a lot of variations in the spelling of a particular word.
E.g. the word इज़हार is found written as: ijhar, izhar, ezhar, yezhar, ejhar, eezahar

We have considered 'code - mixing', in this paper, which refers to intra-sentential switching (Bjorn and Amitava, 2014). Gafaranga and Torras (2002) and Bullock et al. (2014) have used the term code switching.

In India, Hindi is the most widely spoken language and is one of the primary languages for central administrative purposes. As a result, we find code - mixing of English and Hindi language in Indian Social Media. The code mix of English and Hindi languages is referred as Hindlish by Sinha and Thakur (2005). We found that this code mix text is written in Roman English in Indian Social Media.

These are a few elements which impact and contribute to code-mix text in Indian Social Media:

a) *Phonetic Typing:*
It is the visual representation of speech sounds. Pronunciation of a word can vary greatly among dialects of a language. These variations result in a single word having multiple spellings when written in Roman script.
E.g. the word भी could be written as bhi, bhe, be, vi

b) *Abbreviations (Short forms):*
It is a shortened form of a word or phrase. It consists of a letter or a group of letters taken from the word.
E.g. the abbreviation of Professor is Prof.

c) *Word play/ Intentional misspelling for verbal effect:*
These are the creative spellings used for expressing feelings by giving verbal effects. Here the spelling of a word is extended by repeating one of the letters multiple times. E.g. yummmmmmy.

d) *Slang words(acronyms):*
Internet and mobile users use the fewest number of characters to convey a comprehensible message. They have ingeniously started using numerals in place of a part of a word, which reduces the number of letters in a word by giving the same phonetic effect. E.g. 2nite, fi9, 4ever, 10q, ttly, bfn

Normalization of these variably spelled Romanized words is prerequisite in order to get the sentiment of the text.

In this paper, we have presented a model that does word-level language identification in the code- mix script of Hindi and English, automatic transliteration of Romanized English language words and judged the sentiment of the script.

II. RELATED WORK

In this bilingual speech community, there is a natural tendency of speakers to mix phrases and sentences during conversation, which has led to substantial code switching in Hindi and English language. Several linguists (Verma, 1976; Annamalai, 1978; Singh, 1985) investigated code switching. Joshi (1982) investigated the phenomenon of code switching and described some important characteristics of intra-sentential code switching. Kanthimathi (2009) described the way in which Tamil speakers combine English and Tamil. Karimi et al. (2011) reviews the key issues to be considered during transliteration and various methodologies introduced in transliteration literature. Hidayat (2012) has clearly defined the terms inter sentential and intra sentential switching and has discussed the reasons for switching the languages by Facebook users. Chandra et at. (2013) analyzed the reasons of language mixing and its characteristics.

A statistical language independent approach for automatic detection of foreign words in mixed language has been introduced by Kundu and Chandra (2012) and achieved an accuracy of 71.82%. Kumaaran, Khapra, Li (2009) did transliteration on Wikipedia documents by various methods and were able to identify the terms that are to be transliterated or translated. Language detection on short text has been done by Gottron, Thomas, Nedim (2010), where Naïve bayes classifier performed well compared to all other approaches. They have also tested the Reuters collection on different models vs. n-grams. Additionally, few worthy works have been done by Kapoor and Gupta (1991), Bhatt (1997) in the field of code - mix.

Goyal et al. (2003) presented a bilingual syntactic parser that operated on Hindi, English and on code mixed languages in Roman English. Sinha and Thankur (2005) have used different morphological analyzer to translate Hinglish to pure standard Hindi and pure English forms. Bhattacharja (2010) showed how grammar could account for Benglish verbs, a particular type of complex predicate, which are constituted of an English word and a Bengali verb. Clark and Araki (2011) normalized the text using various spellchecker. Vyas et al. (2014) showed the accuracy of tagging POS (parts of speech) in Hindi depends on language identification and transliteration. King and Abney (2013) used Conditional random field method to identify the language of the words in mixed language documents. As the users on the social media are commonly using Abbreviation (short form) or SMS language to communicate, just dealing with language identification and transliteration cannot help us in understanding the text in social media sites.

Voluminous work has been done in the field of sentiment analysis in English. The most recent work on lexicon based sentiment analysis has been done by Cho et al. (2013), where the authors have presented a method for improving lexicon-based review classification by merging multiple sentiment dictionaries, and selectively removing and switching the contents of merged dictionaries. Taboada et al. (2011) have incorporated dictionary-based approach to extract the sentiment of the text.

To understand the sentiment of text written in Devanagari script, Joshi et al. (2010) developed Hindi SentiWordNet, a lexical resource based on English format. Rana (2014) used fuzzy logic method for identifying the semantic orientation of opinions for Hindi text. Namita et al. (2013) used Hindi SentiWordNet to classify the Hindi opinions into positive, negative and neutral. Both the authors have experimented on the text, which is written in Devanagari script.

III. OUR MODEL CAPABILITIES

Our model is capable of handling code-mix scripts constituting a variety of elements seen in Indian Social Media. Below we discuss the methodologies to comprehend and discern the sentiments present in the texts:

1) Handling phonetic typing:

The Indian social media mostly uses phonetic typing/ phonetic transcriptions i.e. a Hindi word is transliterated to Roman English, based on the pronunciation of the word.

This word would have different transcriptions as we are expressing the pronunciation of the word which depends on various factors. One of the major factor, is the mother tongue effect. The same Hindi word is pronounced differently across different states in India.

The other factor for phonetic typing is the limitation in the number of English phonetics when compared to any other Indian languages. Because of these factors we see a variety of spellings for the same word.

E.g. A person from Odisha or Bengal pronounces श(sha) as स(sa), व(va) as ब(ba), 'vishwas' as 'biswas'

A person from Tamil pronounces ग(ga) as ह(ha). These variations were seen on social media sites.

Different ways of writing the word 'मुझे' was found as: mujhe, mujhi, muze, muhjhe, mujhse, mujeh, mujhey, muzhe, mujkhe, muhjhi

Because of these slight variations we have designed a model, which could understand the variations in the spelling and transliterate into Devanagari script. Our model checks the following equivalences:

a) Homorganic nasal case
checks equivalence('व्यक्जन','व्यंजन')
checks equivalence('अन्त', 'अंत')
b) Anuswar-chandrabindu exchange
checks equivalence('इन्तेहाँ', 'इन्तेहां')
checks equivalence('हंस', 'हँस')
c) Non-obligatory use of Nukta
checks equivalence('आवाज', 'आवाज़')
d) Homophonic ending
checks equivalence('येतबार', 'एतबार')
checks equivalence('ये', 'ए')

e) Canonical Unicode
checks equivalence('इज़हार', 'इजहार')
f) Popular Usage
checks equivalence('तनहाई', 'तन्हाई')

2) Handling Abbreviations:

Now a days, on social media we find a lot of abbreviations (short forms) being used in English. We have mapped different kinds of shorthand notation spellings into one, for example: u ("you"), y ("why").

3) Handling Wordplay:

We found creative spellings, which includes phonetic spelling and intentional misspelling for verbal effect e.g. that was soooooo big ("that was so big"). In this case, we identified the flaw and corrected it with the right English word.

4) Handling Slang words:

The next most commonly used phenomenon on social media is the usage of slang words or acronyms, for example: 4get ("forget"). We trained our model with 5000 such slang words for correct language identification.

A. Corpus Acquisition

We have used the data from FIRE 2014 and FIRE 2013 (Forum for IR Evaluation) shared task on transliterated search, which has data of English language (mainly) mixed with six different Indian languages. We have just used the corpus of English and Hindi code – mix data for present experiment. These corpora have ambiguous words belonging to two or more languages i.e. word sense disambiguation in multi lingual corpus, explored by Banea and Mihalcea (2011). We created our own corpus of 500 feedbacks, which were manually collected from various social sites such as Facebook.com and Youtube.com. All the linguistic resources used are uploaded and made available at www.linguisticresources.weebly.com.

It was observed that few Hindi words after transliteration had meaning in English words, which pose a challenge in understanding code – mix text.

Two types of ambiguities were found during the course of our research: Firstly, language identification of transliterated text was difficult. E.g. "to" in English is used as preposition, whereas in Hindi it means 'so'. Secondly, we found ambiguity in acronyms: As users have started using short forms, it is quite difficult to identify the language of the word E.g. "sun" in English denotes 'sunday', whereas in Hindi it means 'listen'.

Our model was not able to deal with these ambiguous words so accurately. We look forward to extend our work later in this regard.

B. Our Statistical Approach

We have discussed the workflow of our model, firstly we divide the feedback into tokens. Consider a code – mix statement as:

"Aaj hum fast and furious movie dekhne jayengey. Movie is awesome, uske reviews bhi achche hai." Then these tokens are fed into English language Identifier. The tokens belonging to English language are tagged as /E at the end of each token.

To identify the English tokens we use standard and vast English dictionary along with 5000 slang words. We keep our model updated with the online English Dictionary by using APIs so that we never fall short of words. Tagging English words as: "Aaj hum fast/E and/E furious/E movie/E dekhne jayengey. Movie/E is/E awesome/E, uske reviews/E bhi achche hai."

The remaining tokens are considered to belong to Hindi Language which is written in Roman Script. /H is tagged to these remaining tokens.

"Aaj/H hum/H fast/E and/E furious/E movie/E dekhne/H jayagey/H. Movie/E is/E awesome/E, uske/H reviews/E bhi/H achche/H hai/H "

Now, as we have identified the language of each word in code-mix text which is a mixture of English and Hindi language.

The basic flowchart of our model is shown below:

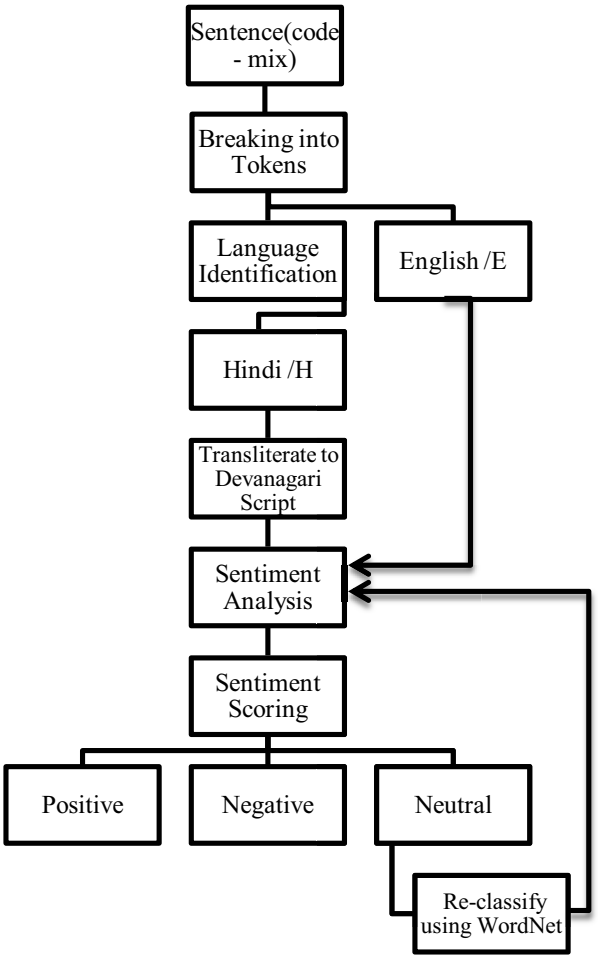

Fig. 1. Flowchart of our model.

We transliterate Romanized Hindi word into Devanagari script by following the transliteration rules which checks the different types of equivalences (as mentioned above).

"आज हम fast and furious movie देखने जायेंगे. Movie is awesome, उसके reviews भी अच्छे है."

Now, we find the sentiment of the statement by using lexicon-based approach, where the sentiments are classified into three categories i.e. Positive, Negative or Neutral. This classification is based on the count of positive and negative lexicons present in the sentence. We have trained our model by using three sentiment resources.

a) Opinion Lexicon: A word set consisting of 6900 positive and negative words.

b) AFINN: It is an affective lexicon consisting of 2477 words, where each word has integer score ranging between -5 (very negative) and +5 (very positive) which is based on the Affective Norms for English Words.

c) Hindi SentiWordNet: It consists of Hindi words in Devanagari script having positive and negative decimal scoring.

We use our sentiment analysis model for Hindi words, which is trained with Hindi SentiWordNet to find the polarity of the sentence based on the Threshold settings.

For example: "आज हम fast and furious movie देखने जायेंगे. Movie is awesome, उसके reviews भी अच्छे है."

The Word 'अच्छे' has a positive score in SentiWordNet. So, we get a positive score of 1.

To find the polarity of English words present in the sentence, we use sentiment analysis model for English words which is trained by Opinion Lexicon and AFINN list.

Word 'awesome' is considered positive by our model.

Here, we get a positive score. The overall sentiment of the statement is 2, giving a positive polarity.

C. Standardization

We had to modify AFINN sentiment resource to meet our requirement. As we were just judging the sentiment into two categories: positive and negative. We have not considered the different degree of negativity and positivity available in AFINN resource.

In the case of Hindi SentiWordNet, we have assumed that all synonyms have the same polarity while all antonyms have the reverse polarity of a word.

D. Threshold Setting & Judgment

The method applied for classifying the feedback as positive or negative depends on the number of words that match the word list to the feedback.

If the numbers of positive words found in a feedback are more, then the feedback is classified as positive, otherwise negative and in this process if the word is not found in any of the sentiment resource then it is considered neutral.

To improve our model accuracy, we have tried to reduce the neutral cases by reclassifying the neutral words which are in English language using WordNet.

The sequence of steps followed in judging the sentiment of a word using WordNet is shown below:

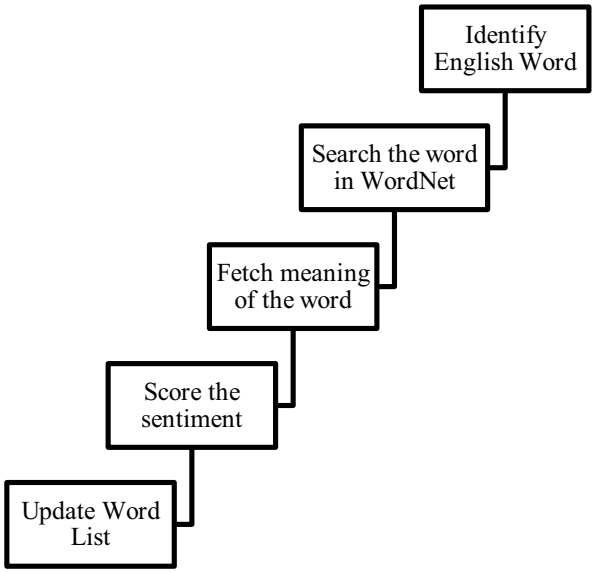

Fig. 2. Using WordNet in judging the sentiment.

Assume that the word "glad" is present in the feedback and this word is not present in the sentiment resource. So this feedback in the first iteration is judged to be neutral. Then these neutral feedbacks are again searched in WordNet. Here the word "glad" is the synonym of "happy" and "happy" is present in Opinion Lexicon, representing its polarity to be positive.

So we update the word "glad" into our positive sentiment resources which reduces the computation cost for future processing. The overall sentiment of the statement which was classified as neutral by static sentiment resources is now classified by using WordNet.

The procedure of searching a word using WordNet is explained in detail using the flowchart below:

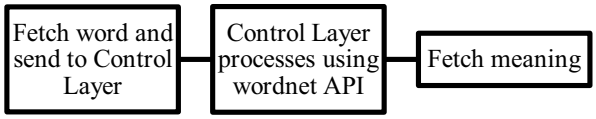

Fig. 3. Word search using WordNet

We had to adopt this approach as WordNet does not contain the polarity of the words. By using this methodology we keep our model updated with the latest WordNet and achieve better accuracy.

E. Result

As there is no standard code – mix (English, Hindi) dataset available to evaluate our model, we took the assistance of 2 faculties and 2 students from IIIT Bhubaneswar to judge the accuracy of our model and we can claim an overall accuracy of more than 85%. As our corpora were collected from FIRE 2013 and FIRE 2014, we were successful in evaluating our accuracy in transliteration. Our model achieved Precision of 0.90.

IV. CONCLUSION

In this paper we have identified the language of code-mix text, which includes Phonetic Typing, Abbreviation (Short forms), Word play, Intentionally misspelt words and Slang words.

We have then transliterated these Romanized English language words which have a variety of spellings and then judged the sentiment of the statement into positive or negative based on lexicon approach. We were able to achieve an accuracy of 85% with our model.

ACKNOWLEDGMENT

We would like to show our gratitude to our parents for sharing their pearls of wisdom with us during the course of this research.

We thank Arjun Roy Choudhury, Sujith George and Sneha Tumkur for their comments and valuable feedback that greatly improved the manuscript.

REFERENCE

[1] Joshi, B. A. R, and P. Bhattacharyya. 2010. A fall-back strategy for sentiment analysis in Hindi: a case study. In International Conference On Natural Language Processing (ICON).

[2] Annamalai, E.1978. The Anglicized Indian Languages: A Case of Code-mixing. International Journal of Dravidian Linguistics, 7, 239-47.

[3] Aravind K. Joshi. 1982. Processing of Sentences with Intra-Sentential Code-Switching. COLING 82, J. Horeck ed., North-Holland Publishing Company.

[4] Barbara E. Bullock, Lars Hinrichs, and Almeida Jacqueline Toribio. 2014. World Englishes, code-switching, and convergence. In Markku Filppula, Juhani Klemola, and Devyani Sharma, editors, The Oxford Handbook of World Englishes. Oxford University Press, Oxford, England. Forthcoming. Online publication: March .

[5] Ben King and Steven Abney. 2013. Labelling the Languages of Words in Mixed-Language Documents using Weakly Supervised Methods. In Proceedings of NAACL-HLT 2013, pages 1110–1119.

[6] Bibekananda Kundu and Subhash Chandra. 2012. Automatic Detection of English Words in Benglish Text: A Statistical Approach. In the 4th International Conference on Intelligent Human Computer Interaction 2012 (IHCI 2012), IEEE, pp.319-322.

[7] Björn Gambäck and Amritava Das. 2014. On measuring the complexity of code mixing. Proceedings of the 1st Workshop on Language Technologies for Indian Social Media (SOCIAL-INDIA), Goa, India, pp 1–8.

[8] Carmen Banea and Rada Mihalcea. 2011. Word Sense Disambiguation with Multilingual Features. In IWCS-11, pp.25-34.

[9] Eleanor Clark and Kenji Araki.2011. Text Normalization in Social Media: Progress, Problems and Applications for a Pre-Processing System of Casual English. In Procedia - Social and Behavioral Sciences, pp. 2-11.

[10] Gottron, Thomas, and Nedim Lipka. 2010. A comparison of language identification approaches on short, query-style texts. Advances in information retrieval. Springer Berlin Heidelberg, 611-614.

[11] Goyal, P. et. al. 2003. Saarthak: A bilingual parser for Hindi, English and Code-switching Structures. EACL Workshop: Computational Linguistics for South Asian Languages: Expanding Synergies with Europe, April 12-17, Budapest.

[12] Heeryon Cho, Jong-Seok Lee, and Songkuk Kim. 2013. Enhancing Lexicon-Based Review Classification by Merging and Revising Sentiment Dictionaries. Proceedings of the 6th International Joint Conference on Natural Language Processing. ACL, pp. 463-470, October 14-18, Nagoya, Japan.

[13] Joseph Gafaranga and Maria-Carme Torras. 2002. Interactional otherness: Towards a redefinition of codeswitching. International Journal of Bilingual- ism, 6(1):1–22.

[14] Kanika Gupta and Monojit Choudhury and Kalika Bali. 2012. Mining Hindi-English Transliteration Pairs from Online Hindi Lyrics, In Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC '12), 23-25 May. Istanbul, Turkey, pages 2459-2465.

[15] K.Kanthimathi. 2009. Tamil-English Mixed Language Used in Tamilnadu. In The International Journal of Language Society.

[16] Kapil Kapoor and Gupta. 1991. English and Indian Languages: Code Mixing. In R. Gupta & K. Ka- poor (Ed.), English in India: Issues and problems, pp. 207-215.

[17] Karimi, K., Scholer, F., and Turpin. 2011. A Machine Transliteration Survey. In ACM Computing Surveys (CSUR), Volume 43 Issue 3, April.

[18] Kumaran, A., Khapra, M., and Li, H. 2010. Report of NEWS 2010 Transliteration Mining Shared Task, in the ACL 2010 Named Entities WorkShop (NEWS- 2010), Uppsala, Sweden, Association for Computational Linguistics, July.

[19] Mittal, Namita, Basant Agarwal, Garvit Chouhan, Nitin Bania, and Prateek Pareek. 2013.Sentiment Analysis of Hindi Review based on Negation and Discourse Relation. In proceedings of International Joint Conference on Natural Language Processing, pp. 45-50.

[20] Rakesh M. Bhatt. 1997. Code-Switching, Constraints, and Optimal Grammar. Lingua 102:223-251.

[21] Ramesh M.K. Sinha and Anil Thakur. 2005. Machine Translation of Bi-lingual Hindi-English (Hinglish) text. Proceeding of the 10th Confer- ence on Machine Translation. Sept. 13-15, MT- Archive, Phuket, Thailand, pp. 149-156.

[22] Rishiraj Saha Roy, Monojit Choudhury, Prasenjit Majumder, and Komal Agarwal. 2013. Overview and datasets of fire 2013 track on transliterated search. In Proceedings of the FIRE 2013 Shared Task on Transliterated Search.

[23] Shishir Bhattacharja. 2010. Benglish Verbs: A Case of Code-Mixing in Bengali. Proceedings of the 24th Pacific Asia Conference on Language, In- formation and Computation, pp.75-84. Institute of Digital Enhancement of Cognitive Processing, Waseda University.

[24] Shweta Rana. 2014. Sentiment Analysis for Hindi Text using Fuzzy Logic Indian Journal of Applied Research, Vol.4, Issue.8 August.

[25] Singh, Rajendra. 1985. Grammatical Constraints on Code mixing: Evidence from Hindi-English. Canadian Journal of Linguistics, 30, 33-45.

[26] Subhash Chandra, Bibekananda Kundu and Sanjay Kumar Choudhury. 2013. Hunting Elusive English in Hinglish and Benglish Text: Unfolding Challenges and Remedies", In Proceedings of 10th International Conference on Natural Language Processing (ICON-2013), India,December 19-20.

[27] Taboada, Maite, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. 2011. Lexicon-based methods for sentiment analysis. Computational linguistics 37, no. 2: 267-307.

[28] Taofik Hidayat. 2012. An analysis of code switching used by facebookers: a case study in a social network site. Student essay for the

study programme Pendidikan Bahasa Inggris (English Education) at STKIP Siliwangi Bandung, Indonesia.

[29] Verma S. K. 1976. Code switching: Hindi-English. Lingua, 38, 153-165.