Consonant-Vowel Sequences as Subword Units for Code-Mixed Languages

Upendra Kumar, Vishal Singh, Chris Andrew,
Santhoshini Reddy, Amitava Das
Indian Institute of Information Technology, Sri City, AP, India, 517588
{upendra.k14, vishal.s14, chris.g14, santhoshini.g14, amitava.das}@iiits.in

Abstract

In this research work, we develop a state-of-art model for identifying sentiment in Hindi-English code-mixed language. We introduce new phonemic sub-word units for Hindi-English code-mixed text along with a hierarchical deep learning model which uses these sub-word units for predicting sentiment. The results indicate that the model yields a significant increase in accuracy as compared to other models.

Introduction

The evolution of social media texts such as blogs, micro-blogs (e.g., Twitter), WhatsApp, and informal chats have created many new opportunities for information access and language technologies, but have also presented many new challenges. This makes it one of the primary research areas of the present era. In social media, non-English speakers [according to statistics half of messages on Twitter arent in English (Schroeder, Minocha, and Schneider 2010)] do not always use English to express their thoughts. Users can be frequently seen using their native language to convey meaning, inserting English words in between for easier comprehension. In order to study the phenomenon of mixing multiple languages, researchers have introduced two key language generation processes known as code-mixing and code-borrowing. In this paper, however, we do not distinguish between the two.

Our work focuses on addressing the specific challenges of using out-of-vocabulary words required for developing an efficient model to analyze sentiment from Hindi-English code-mixed language. In order to do so efficiently, we introduce a heuristic for segmenting words in phonemic sub-word units instead of using word or character level features. We use CNNs and Bi-LSTMs to learn representations from these sub-word units to evaluate the sentiment of a sentence. We observe that the phonemic sub-words are able to obtain better representation when compared with word, character or character *n*-gram based representations. For the purpose of testing our models, we have created a new dataset of Hindi-English code-mixed sentences consisting of 18k sentences collected from Twitter.

Related Works

In recent times, a significant amount of research has been done to develop computational models for identifying sentiment from code-mixed texts. (Sharma, Srinivas, and Balabantaray 2015) use Hindi SentiWordNet and normalization techniques to detect sentiment in Hi-En code-mixed tweets. (Rudra et al. 2016) use lexicon based features (swear words, exclamation marks, sentiment words and negation words) for developing a classifier for sentiment analysis. The first attempt for a deep learning based method for Hindi-English code-mixed text was proposed by (Joshi et al. 2016). They used character n-grams as sub-word units that were obtained as convolutions over characters and passed to a LSTM layer followed by softmax layer. For Hi-En code-mixed text (Joshi et al. 2016) address the problem of rare or out-of-vocabulary words without any text normalization. In this paper, we propose a novel approach, without any need of explicit text normalization, for creating sub-word units and a new hierarchical model that efficiently learns sentence representations from these units.

Dataset

Due to paucity of available code-mixed text, research on code-mixed texts has been limited to handcrafted lexicon based models. In order to address this problem, we have created a dataset of 18K code-mixed tweets using the Twitter API[1] by querying tweets from Twitter accounts that frequently use Hindi-English code-mixed style for tweeting. Based on the sentiment of the text, data was manually annotated into three classes: positive, neutral and negative(-1, 0, 1). The collected dataset was further labelled for language tags (Hi, En, Un) on the word-level. The Code-mixing index, which measures the degree of mixing or inter-usage of the two languages with each other, is also evaluated for each sentence in the corpus. Out of 18K tweets, around 4K tweets were discarded to make the sentiment distribution equal across the three classes. 80% of the corpus data was used for training and 20% was used for testing.

Methodology

In this paper we focus on romanized texts, where both Hindi and English words are expressed using Roman script. How-

[1] https://developer.twitter.com/en/docs

ever, the use of Roman script for a Hindi word may produce spelling variations, as the Roman script is not an *Abugida* script (Daniels 1990). Abugida scripts have a one-to-one mapping between spelling and pronunciation, where the same units will have the same pronunciations regardless of their context. This is not so in English, where it is common to see different pronunciations for the same combinations of characters.

The variations in spelling produce a number of rare and out-of-vocabulary words that introduce errors in the classification process. We propose to segment words in phonemic sub-word units consisting of consonant-vowel sequences inspired from fundamental principles of an *Abugida* language. These units eliminate the problems associated with rare words, as they are replaced by more commonly occurring phonemic units. Words are segmented into sub-word sequences of C^+V^+ that acts as an approximate syllable consisting of onset and rime (without coda) parts. For example, Hindi word *kitab* will be segmented into *ki*, *ta* and *b* and English word *book* will be segmented into *boo* and *k*. From other examples like *chandrama* (where the C^+V^+ rule fails to break it into correct syllables), exceptional cases can obviously be inferred. In code[2] a few of these frequently occurring complications were addressed revolving around characters like *r*, *m*, *n*, *s* and *l*. These characters are expected to be at end of the syllable if they are preceded by a vowel. But, other complications were ignored for now. We expect that these sub-word units having rich distribution as compared to words, will perform better with deep learning models.

Each sentence s_i is first tokenized into word tokens $w_{i,j}$. For each word token $w_{i,j}$, a list of approximate syllables/sub-word units $u_{i,j,k}$ is obtained. In order to obtain the predicted sentiment y_i for sentence s_i, we propose a hierarchical BiLSTM network using these units as highlighted in Figure 1. The first layer corresponds to the embedding layer which encodes the sequences of sub-words to their corresponding vector representations. The second layer functions as a word encoder which learns to compose representations of constituent sub-word units $u_{i,j,k}$ into representation of a word $w_{i,j}$. The third layer obtains representation for a sentence s_i. Finally, a fully connected layer is added to use these representations in order to infer and predict the sentiment associated with the sentence. In next section we compare this approach with past works done in sentiment analysis for code-mixing.

Experiments and Results

In order to validate our model, multiple experiments were done using different word based model architectures using LSTM and CNN. The best performance of word-based classifier was 50.43% using LSTM with pre-trained GloVE embeddings. Further, the sub-word LSTM model proposed in (Joshi et al. 2016) was used for the purpose of comparison. It performed better than other word-based models yielding 57.88% accuracy. In next experiment, we used the proposed sub-word units with a hierarchical model as illustrated in Figure 1 yielding an accuracy of 74.62%. It is clear that the model proposed in this paper significantly outperforms the latter sub-word model by a large margin of 16.74%. This is an ongoing research work where our objective is to circumvent the problem of rare words and unavailability of large data-sets. Along with learning sub-word embedding which are rich in morphological information we are further looking into joint methods to harness both semantic information as well as morphological information.

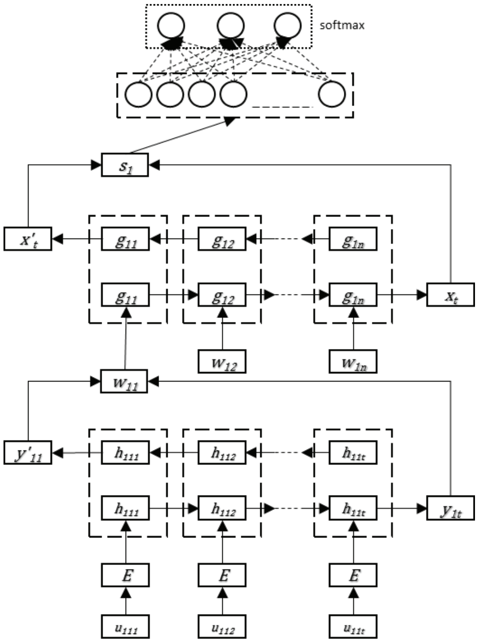

Figure 1: Hierarchical model for sentiment analysis using phonemic sub-word units. The figure illustrates the graph for sentence s_1.

References

Daniels, P. T. 1990. Fundamentals of grammatology. *Journal of the American Oriental Society* 727–731.

Joshi, A.; Prabhu, A.; Shrivastava, M.; and Varma, V. 2016. Towards sub-word level compositions for sentiment analysis of hindi-english code mixed text. In *COLING*, 2482–2491.

Rudra, K.; Rijhwani, S.; Begum, R.; Bali, K.; Choudhury, M.; and Ganguly, N. 2016. Understanding language preference for expression of opinion and sentiment: What do hindi-english speakers do on twitter? In *EMNLP*, 1131–1141.

Schroeder, A.; Minocha, S.; and Schneider, C. 2010. The strengths, weaknesses, opportunities and threats of using social software in higher and further education teaching and learning. *Journal of Computer Assisted Learning* 26(3):159–174.

Sharma, S.; Srinivas, P.; and Balabantaray, R. C. 2015. Text normalization of code mix and sentiment analysis. In *Advances in Computing, Communications and Informatics (ICACCI), 2015 International Conference on*, 1468–1473. IEEE.

[2] Refer to ortho.py in complementary zip file.