Solving Data Sparsity for Aspect based Sentiment Analysis using Cross-linguality and Multi-linguality

Md Shad Akhtar*, Palaash Sawant+, Sukanta Sen*, Asif Ekbal* and Pushpak Bhattacharyya*

* *Department of Computer Science & Engineering*
Indian Institute of Technology Patna, India
{shad.pcs15,sukanta.pcs15,asif,pb}@iitp.ac.in
+ *Goa University*, palaash77@gmail.com

Abstract

Efficient word representations play an important role in solving various problems related to Natural Language Processing (NLP), data mining, text mining etc. The issue of data sparsity poses a great challenge in creating efficient word representation model for solving the underlying problem. The problem is more intensified in resource-poor scenario due to the absence of sufficient amount of corpus. In this work, we propose to minimize the effect of data sparsity by leveraging bilingual word embeddings learned through a parallel corpus. We train and evaluate Long Short Term Memory (LSTM) based architecture for aspect level sentiment classification. The neural network architecture is further assisted by the hand-crafted features for the prediction. We show the efficacy of the proposed model against state-of-the-art methods in two experimental setups i.e. multi-lingual and cross-lingual.

1 Introduction

Sentiment analysis (Pang and Lee, 2005) tries to automatically extract the subjective information from a user written textual content and classifies it into one of the predefined set of classes, e.g. *positive*, *negative*, *neutral* or *conflict*. Sentiment analysis performed on coarser level (i.e. document or sentence level) does not provide enough information for a user who is critical of finer details such as *battery life* of a laptop or *service* of a restaurant etc. Aspect level sentiment analysis (ABSA) (Pontiki et al., 2014) serves such a purpose, which first identifies the features (or aspects) mentioned in the text and then classifies it into one of the target classes. For example, the following review is for a restaurant where the writer shares her/his experience. Though s/he likes the *food* but certainly not happy with the *service*.

*One of the best **food** we had in a while but the **service** was very disappointing.*

Analyzing such reviews on sentence level will reflect only an overall sentiment (i.e. *conflict*) of the sentence ignoring critical information such as *food* and *service* qualities. However, ABSA will first identify all the aspects in the text (i.e. *food* and *service*) and then associate *positive* with *food* and *negative* with *service*. Identification of aspect terms is also known as aspect term extraction or opinion target extraction. In this work, we focus on the second problem i.e. aspect level sentiment classification.

Literature survey suggests a wide range of research on sentiment analysis (at the document or sentence level) is being carried out in recent years (Turney, 2002; Kim and Hovy, 2004; Jagtap and Pawar, 2013; Poria et al., 2016; Kaljahi and Foster, 2016; Gupta et al., 2015). However, most of these researches are focused on resource-rich language like English. Like many other Natural Language Processing (NLP) problems, research on sentiment analysis involving Indian languages (e.g. Hindi, Bengali etc.) are very limited (Joshi et al., 2010; Bakliwal et al., 2012; Kumar et al., 2015; Balamurali et al., 2012; Singhal and Bhattacharyya, 2016). Due to the scarcity of various qualitative resources and/or tools in such languages, the problems have become more challenging and non-trivial to solve. The research on ABSA involving Indian languages has started only very recently, for e.g. (Akhtar et al., 2016a,b).

2 Motivation and Problem Definition

Indian languages are resource-constrained in nature as there is a lack of ready availability of different qualitative lexical resources and tools. In a supervised machine learning framework, good amount of training data always have a great impact on the overall system performance. Low-resource languages (such as the Indian [Hindi etc.]) usu-

Proceedings of NAACL-HLT 2018, pages 572–582
New Orleans, Louisiana, June 1 - 6, 2018.

ally suffer due to the non-availability of sufficient training data instances. In order to solve the data and resource scarcity problem in one language, researchers often utilize cross-lingual setup to leverage the resource-richness of other languages by projecting the task into a common problem space (Zhou et al., 2016; Balamurali et al., 2012; Singhal and Bhattacharyya, 2016; Barnes et al., 2016). The projection is often performed with the help of machine translation or bilingual dictionaries.

In recent times, deep learning (DL) techniques have shown success in solving several NLP problems. A good word representation is the essence of any deep learning approach. In the absence of qualitative word embeddings, it turns out to be a non-trivial task for any DL framework to effectively learn hidden features (e.g. lexical, syntactic, semantics etc.), which may effect the performance. The quality of word embeddings can be preserved by employing state-of-the-art distributed word representation models such as Word2Vec (Mikolov et al., 2013) or GloVe (Pennington et al., 2014) provided a huge corpus to train on. Due to this limitation, quality of word embeddings in Indian languages usually are not at par with that of resource-rich languages like English.

Data sparsity in word representation (i.e. absence of representation of any word) is another problem that often has to be dealt with. In order to solve any NLP task, out-of-vocabulary (OOV) words in a word embedding model pose a serious challenge to the underlying learning algorithm. For a missing word representation, the literature suggests two possible solutions: a) zero vector (Bahdanau et al., 2017) or b) random vector (Dhingra et al., 2017). However, in both the cases the resultant vector could be completely out of context and often does not fit well with others. Further, word embedding of a word in a source language has absolutely no correlation with the word embedding of the same word (translated) in the target language, hence, it cannot be directly used for training and/or testing in a cross-lingual setup. The prime motivation of the work is to minimize the effect of *data sparsity* and thereby, enabling any deep learning framework to effectively learn its hidden features.

In this paper, we propose to solve the *data sparsity* problem in a resource-scarce language scenario (here, primarily Hindi and also French embeddings)[1]. We hypothesize that addressing data sparsity in an intelligent manner would yield increased performance. We utilize bi-lingual word embedding (Luong et al., 2015) trained on English-Hindi and English-French parallel corpus to bridge the language divergence in the vector space. The proposed method is based on a deep learning (DL) architecture named Long Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997). We try to establish our hypothesis through experiments on *aspect based sentiment classification* task in both the setups i.e. multi-lingual and cross-lingual for *English-Hindi* and *English-French* language pairs. Aspect based sentiment classification deals with assigning the sentiment polarity (i.e. *positive*, *negative*, *neutral* or *conflict*) to the aspect terms. For evaluation, we use the datasets provided in (Akhtar et al., 2016a) for Hindi, SemEval-2014 shared task on ABSA (Pontiki et al., 2014) for English and SemEval-2016 shared task on ABSA (Pontiki et al., 2016) dataset for French.

2.1 Contributions

Major contributions of our current work are as follows: **a)** we train and use bilingual embeddings on Amazon product review corpus consisting of parallel sentences of English-Hindi and English-French, which serve as a bridge between the two languages; **b)** we propose to solve the problem of *data sparsity* in low-resource language word embedding by utilizing the word embedding created on resource-rich language; and **c)** to further improve the system's prediction we extract and use various English side semantic features of the machine translated words.

As we already mentioned, the research on ABSA involving Indian languages are limited. Some of the recent works include the one reported in (Akhtar et al., 2016a,b). The authors in (Barnes et al., 2016) employed bilingual word embeddings for sentiment classification in a cross-lingual setup. To the best of our knowledge, our current attempt is the very first of its kind to employ bilingual word embeddings for a multi-lingual scenario. Our proposed system differs with the existing systems in the following ways.

1. **Setup**: System (Barnes et al., 2016) defines a cross-lingual setup while (Singhal and Bhat-

[1] We use French to show how generic our proposed approach is. Compared to English, French does not have enough sentiment annotated data

tacharyya, 2016) is multi-lingual in nature. In contrast, our proposed system is applied to both multi-lingual and cross-lingual setups.

2. **Approach**: System (Akhtar et al., 2016a) defines classical feature driven approach while the system (Barnes et al., 2016) utilized bilingual word embeddings as feature values to train a Support Vector Machine (SVM) classifier. Rest of the systems (Akhtar et al., 2016b; Singhal and Bhattacharyya, 2016) (including the proposed one) are based on deep neural network architecture. However, the techniques employed are very much different. Akhtar et al. (2016b) is a CNN-SVM based system with the assistance of multi-objective optimized features, while Singhal and Bhattacharyya (2016) is a CNN based system that translate the source language texts into target language text (English) for training and evaluation. In comparison, our proposed method employ LSTM to solve the data sparsity problem in both multi-lingual as well as cross-lingual setups.

3. **Problem addressed**: Authors in (Singhal and Bhattacharyya, 2016) focused on sentence level sentiment classification while our present work focuses on fine-grained sentiment classification at the aspect level.

4. **Word Embeddings:** The proposed system employs shared vector-space bilingual word embeddings for training and testing while (Singhal and Bhattacharyya, 2016) projected the source language train & test data into target language using machine translation and utilizes target side pre-computed word vectors for training the system. Whereas, the system reported in (Akhtar et al., 2016b) employed mono-lingual word embeddings for training and evaluation.

5. **Data Sparsity:** The system of (Akhtar et al., 2016b) does not address the problem of data sparsity, while our proposed system tries to minimize the effect of data sparsity. Our proposed system tackles the data sparsity problem by replacing the OOV word with its translated form which usually happens to be its closest neighbor in the shared vector space, hence, the semantic closeness is preserved to an extent. Whereas, system (Singhal and Bhattacharyya, 2016) addressed the data sparsity by translating every word of the source language into target language which may introduce loss of sentiment in the target language as a side-effect (Mohammad et al., 2016).

6. **Hand-crafted Features:** The proposed system employs much richer set of lexicon based features than that of (Singhal and Bhattacharyya, 2016). Also, we do not augment polar words in the training instances as done in (Singhal and Bhattacharyya, 2016), rather we use sentiment scores of these lexicons as features themselves in the training and testing instances. Whereas, the authors in (Akhtar et al., 2016b) obtained an optimized feature vector through the application of multi-objective genetic algorithm.

3 Proposed Methodology

We propose to use a Long Short Term Memory (LSTM) architecture on top of bilingual word embeddings for the prediction. LSTM is a special kind of recurrent neural network (RNN) which efficiently captures long term dependencies. Bidirectional LSTM is an extended version of LSTM which takes both forward and backward sequences into account. Our model consists of two bidirectional LSTM layers followed by two fully-connected layers and an output layer.

3.1 Bilingual Word Embedding

We employ bilingual word embeddings (Luong et al., 2015) trained on a parallel *English-Hindi* (and *English-French*) corpus. We generate a parallel corpus for Amazon product review datasets[2] (consisting of approx. 7.2M sentences) using an in-house product review domain based *English→Hindi* (*English→French*) Statistical Machine Translation (SMT) system (*English→Hindi*: 39.5 BLEU score and *English→French*: 37.9 BLEU score). We employ widely used and standard machine translation tool *Moses* (Koehn et al., 2007) to train the phrase-based SMT system. The alignment information are obtained from the *mosesdecoder* (Koehn et al., 2007) during translation of the reviews.

The parallel corpus along with the alignment information are used to train two (English and Hindi)

[2]http://snap.stanford.edu/data/other.html

Skip-Gram word2vec (Mikolov et al., 2013) models which share the common vector space. If a word W_S is aligned to word W_T, then the context information C_T of target word W_T is also used as context of the source word W_S along with its own context C_S for computing word vectors. By utilizing the context information of both source and target side, resultant word embeddings of W_S and W_T are semantically closer to each other in the vector space.

Bilingual skip-gram model creates two separate word embeddings, i.e. one each for source (Hindi) and target language (English). First, we extract word representations for all the words in a sentence from the Hindi bilingual word embeddings. Subsequently at the second step we translate all the OOV words (words whose representations are missing in Hindi bilingual embeddings) into English and then perform another lookup in English embeddings. For instance, if embedding of a word 'अच्छाlachcha' is unknown we translate it in English as '*good*', and use its word embeddings in place of the source word 'अच्छाlachcha'. Thus the missing representation of OOV word is replaced by its translated target side representation. Since, both English and Hindi word embeddings share a common vector space, this replacement strategy proves to be an effective technique. In our case, we observe a reduction of approximately 65% and 37% OOV words, respectively for Hindi and French by our proposed replacement strategy. Consequently, an increase in accuracy value is observed during evaluation.

Hindi is a morphologically rich language. Many inflected words in Hindi share a common translated word in English. For example, based on the gender of the subject Hindi has two forms for word 'goes': 'जाता है | jAtA hai' (male) or 'जाती है | jAtI hai' (female). Therefore, if representation of one word (जाता है | jAtA hai) is missing in Hindi embedding we can still find its representation in English through its translation i.e. 'goes'. Bilingual embedding also helps in addressing the spelling variation cases. For e.g. two differently spelled words in Hindi such as 'कम्बिनेशन | kambineshana' and 'कंबीनेशन | kaMbIneshana' translate to an English word 'combination'.

We repeat the above process for *English-French* language pair to obtain two (English and French) word2vec models. We also released computed bilingual word embeddings for the research community[3].

3.2 Features

We employ various hand-crafted features to assist the network. We try to leverage the effectiveness of English side resources by translating a word into English and then extracting its feature representation. We use following set of features in our task. It should be noted that we do not include any lexical or syntactic features during training as distributed word embedding models are good at capturing such features. So, during the training phase, network adapts its weights to learn the relevant set of these features from the word embeddings itself.

1. **Bing Liu** (Ding et al., 2008) **& MPQA** (Wiebe and Mihalcea, 2006) **lexicons**: We define a feature that marks the positivity/negativity scores of the words in a sentence. We assign a score of +1 & -1, respectively to each positive and negative word in the sentence. For unseen words, we use score as 0. We extract one such feature from each lexicon.

2. **SentiWordNet** (Baccianella et al., 2010): Three features are extracted for every word denoting its positivity (posScore), negativity (negScore) and objectivity (1 - [posScore + negScore]) scores, respectively.

3. **Semantic Orientation (SO)** (Hatzivassiloglou and McKeown, 1997): Semantic orientation defines the association of a word *w.r.t.* its positivity and negativity. Semantic orientation (SO) of a word is the difference of point-wise mutual information of a word w in positive and negative reviews. We calculate the SO score of each word in the context window of size ± 5 and take the cumulative SO score as the feature value.

3.3 Cross-lingual and Multi-lingual Setups

We evaluate our proposed approach for two setups i.e. multi-lingual and cross-lingual setups. In multi-lingual setup, the proposed model is trained and evaluated on datasets of the same language i.e. Hindi or French. We pre-process our datasets to reduce the effect of data sparsity by utilizing the resource-rich language i.e. English. In contrast, the

[3]Bi-lingual word embeddings available at http://www.iitp.ac.in/~ai-nlp-ml/resources.html

cross-lingual setup employs dataset of resource-rich language (i.e. English) for training and during evaluation Hindi or French dataset is used. Similar to the multi-lingual setup, we pre-process the test dataset to reduce the effect of data sparsity in cross-lingual setup as well.

An overall schema of the proposed methodology is depicted in Figure 1 for both multi-lingual and cross-lingual setups. Figures 1a and 1b show the training architectures for the cross-lingual and multi-lingual scenarios, respectively. Since our test datasets for both the variants are in Hindi (or French), testing scenario for cross-lingual and multi-lingual setups are also the same as represented in Figure 1c.

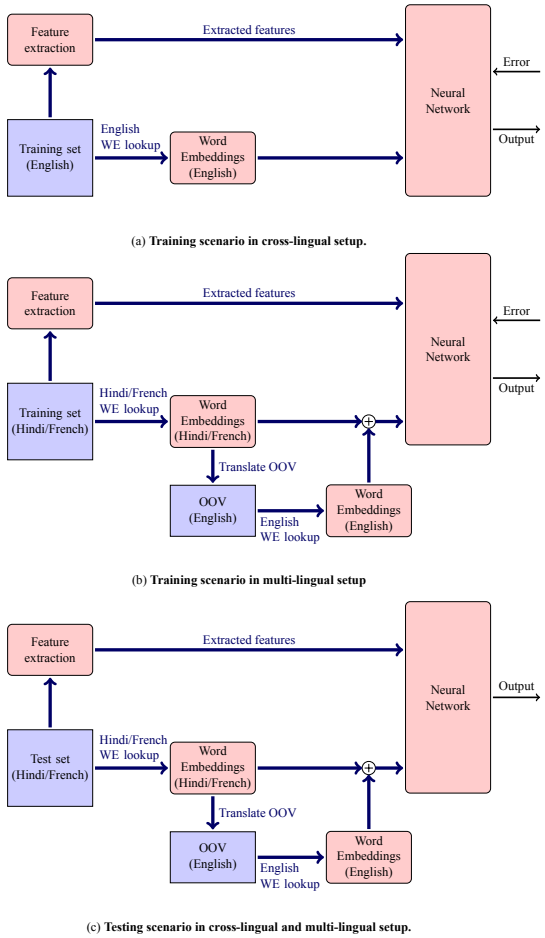

Figure 1: Proposed schema for *English-Hindi* and *English-French* language pairs.

3.4 Neural Network Architecture

For the successful marriage of word embeddings and extracted features, we try three different architectures as depicted in Figure 2. In the first architecture ($A1$, Figure 2a), we concatenate extracted features of each word of an instance with the corresponding word representations and pass it through a LSTM network followed by dense and output layers. In the second architecture ($A2$, Figure 2b), we do not combine features and word representations together. Rather, we learn sentence embeddings through a LSTM network and then concatenate it with the extracted features before feeding to the dense layer. Finally, in the third architecture ($A3$, Figure 2c), we train separate LSTMs for the extracted features and word embeddings. Subsequently, we merge their representations at the dense layer. The choice of separate LSTMs for the hand-crafted features in architecture $A3$ is driven by the fact that the dimension of a word embedding is usually very high as compared to its corresponding hand-crafted features. If trained together, as in architecture $A1$, extracted features of low dimension usually get overshadowed by the high-dimensional word embeddings. Thus making it non-trivial for the network to learn from the extracted features. Further, to exploit the sequence information of words in a sentence we pass hand-crafted features of each word through a separate LSTM layer. For example, in the following review sentence, there are two positive words ('*liking*' and *'recommending'*) and only one negative word ('*far*'). In a model that takes into account only the simple polar word score, the sentence would have high relevance towards the positive sentiment. However, the sequence information of the phrase "*far from liking and recommending*" dictates the negative sentiment of the sentence.

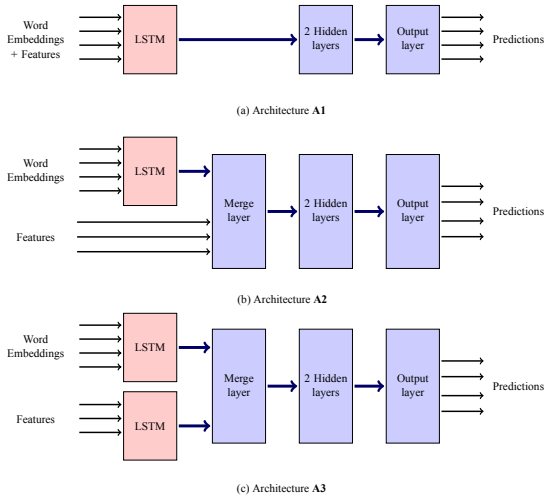

Figure 2: Neural Network Architectures.

"*I'm far from liking and recommending this phone to anyone.*"

In contrast to $A3$, architecture $A2$ does not rely on the sequence information of the extracted features and let the network to learn on its own.

We use 300 dimension word embeddings for the experiments. Each LSTM layer contains 100 neurons while two dense layers contain 100 and 50 neurons respectively.

4 Experimental Results

In this section, we describe the datasets, experimental setup, results and provide necessary analysis.

4.1 Datasets

We use Hindi ABSA dataset released by (Akhtar et al., 2016a) for our evaluation purpose. A total of 5,417 review sentences are present along with 4,509 aspect terms. Each aspect term belongs to one of the four sentiment classes: '*positive*', '*negative*', '*neutral*' and '*conflict*'. We split the dataset into 70%, 10% and 20% as training, development and test, respectively for the experiment. For French case, we use the SemEval-2016 shared task on ABSA (Pontiki et al., 2016) restaurant dataset. It consists of 2,429 review sentences and 3,482 aspect terms. In cross-lingual setup, we utilize English dataset of SemEval-2014 shared task on ABSA (Pontiki et al., 2014) for training and Hindi ABSA dataset for testing. The English dataset comprises of product reviews in two domains i.e. restaurant and laptop. However, we only employ laptop domain dataset as most of the reviews in Hindi ABSA datasets belong to the electronics domain. For training in cross-lingual setup, we combine the training and gold test dataset together. In total, there are 3,845 review sentences comprising of 3,012 aspect terms. For English-French case, we use English restaurant dataset of SemEval-2016 shared task on ABSA (Pontiki et al., 2016) for the training and French ABSA dataset (Pontiki et al., 2016) for evaluation. The SemEval-2016 English restaurant dataset contains 3,365 aspect terms across 2,676 review sentences.

4.2 Experiments

We use Python based neural network library, Keras[4] for implementation. For English-Hindi, all the four classes (namely *positive*, *negative*, *neutral* and *conflict*) were considered, whereas for English-French three classes (all except *conflict* class) were used for classification. Since there is no false class, we use accuracy value as metric to measure the performance of the system. Also, we utilize accuracy value for the direct comparison with the existing state-of-the-art systems. LSTM network is trained with early stopping criteria on (i.e. preserving best learned parameter at each epoch). We set the number of epochs and patience value as 100 & 20 respectively. In other words, we run the experiments for maximum 100 epochs and if validation loss does not reduce for consecutive 20 epochs training stops and reports the best epoch attained so far. As activation function, we utilize *'tanh'* at the intermediate layers, while for classification, we use *'softmax'* at the output layer. To prevent the network from over-fitting, we incorporate an efficient regularization technique called *'Dropout'* (Srivastava et al., 2014). At each layer of training, dropout skips few hidden neurons randomly. We fix dropout rate to be 45% during training while for optimization we use *'adam'* optimizer (Kingma and Ba, 2014).

Experimental results for aspect sentiment classification in multi-lingual and cross-lingual setups are reported in Figure 3 for both the language pairs. In total, we evaluate our model for four cases i.e. **a.** *En-Hi multi-lingual*, **b.** *En-Hi cross-lingual*, **c.** *En-Fr multi-lingual* and **d.** *En-Fr cross-lingual* scenarios. The non-root four-boxed nodes report performance of the respective methods for the four cases. The left subtree represents LSTM based baseline system that utilizes monolingual word embedding (WE) (i.e. word2vec model trained only on 7.2M Hindi and French sentences respectively). Whereas the right subtree represents usage of bilingual word embeddings in all the cases. Comparison between monolingual WE and bilingual WE shows competing results. Monolingual WE (a_M: 63.64%) in multi-lingual scenario performs better than the bilingual WE (a_B: 62.51%) for *English-Hindi* case, while bilingual WE (c_B: 70.89%) reports better performance as compared with monolingual WE (a_M: 66.29%) for *English-French* case. We observe a performance loss of approx. 1 point with bilingual embeddings for English-Hindi case. However, after addressing the problem of data sparsity (i.e. when OOV words are translated and corresponding English word embeddings are computed) the same LSTM network reports an improved accuracy value of 64.83% (a_{BO}) for English-Hindi case, thus observ-

[4] http://keras.io

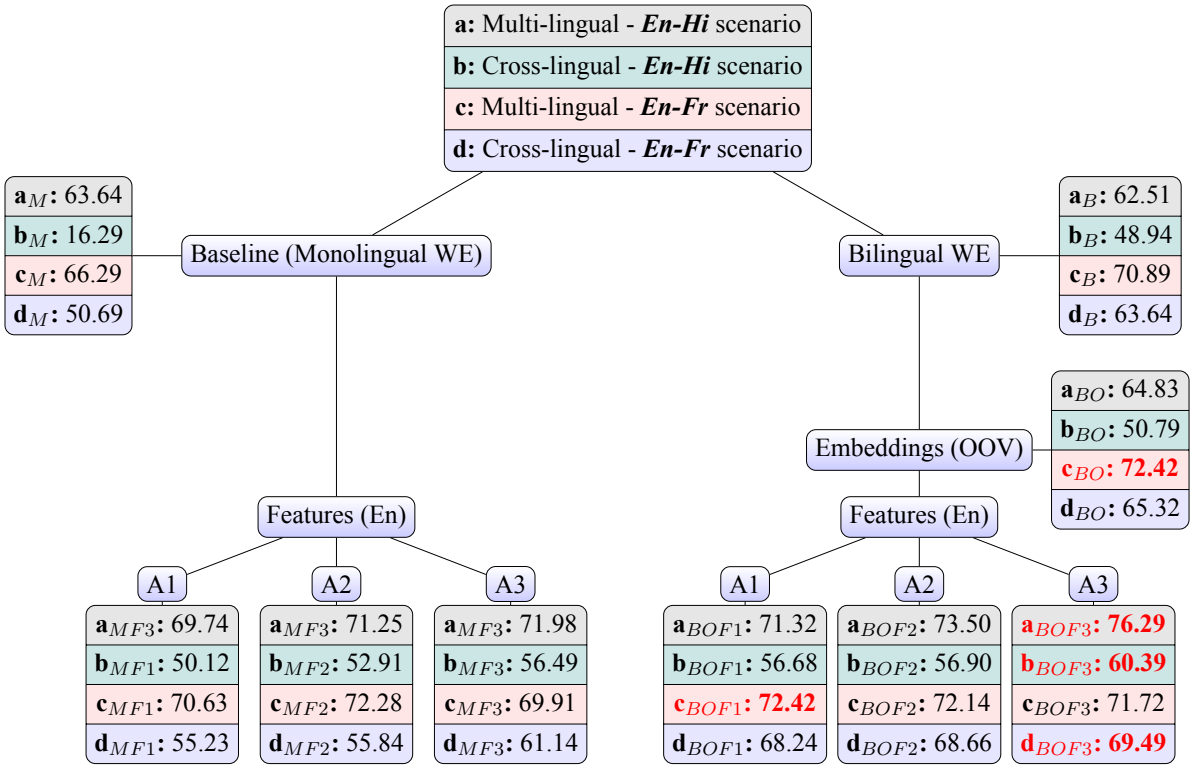

Figure 3: **Aspect classification in Multi-lingual and Cross-lingual setups for English-Hindi and English-French scenarios:** *Left subtree* represents various baselines and their corresponding results. *Right subtree* represents the proposed approach at different levels. Four-box rectangles at non-root levels show accuracy values for **a.** *multi-lingual (En-Hi)*, **b.** *cross-lingual (En-Hi)*, **c.** *multi-lingual (En-Fr)* & **d.** *cross-lingual (En-Fr)* scenarios respectively. **OOV:** Out-of-vocabulary words. **A1:** Word embeddings and extracted features are combined and fed into single LSTM network. **A2:** Extracted features are directly merged with LSTM output of word embedding. **A3:** One LSTM network each for word embeddings and extracted features. Subscripts **M:** *monolingual WE*; **B:** *bilingual WE*; **O:** *Embeddings(OOV)*; **F:** *Features*; **1,2,3:** *Architecture A1, A2 & A3* respectively.

ing a performance increase of more than 2 points. For English-French case, we also observe the improvement with embeddings of OOVs. This suggests that the richness of target language (English) word embeddings helps the system to efficiently solve the problem encountered in resource-poor source language. Since the resources are limited for resource-poor language we try to leverage the high-quality lexicon features of English in our system. Consequently, we introduce the extracted features of Section 3.2 to the network. For English-Hindi multi-lingual scenario, the performance increments from $A1$ to $A2$ to $A3$ indicate that the resource-richness of English language plays a crucial role in classification. While we incorporate English side lexicon features for English-French multi-lingual scenario, we observe no performance improvement like the others. For this case, our system reports an accuracy of 72.42% with (c_{BOF1}) and without (c_{BO}) the use of extra features.

Results of cross-lingual setup for English-Hindi case, where we train the network utilizing English dataset and evaluate the model on Hindi dataset, are reported in row 2 of the four-boxed nodes in Figure 3. The baseline model for cross-lingual setups (left subtree of Figure 3) employs monolingual word embeddings of English and Hindi for training and testing respectively. Since the vector spaces of two different languages are completely unrelated, it is no surprise that the baseline system achieves merely 16.29% (b_M) accuracy. Using only the bilingual word embeddings the system achieves 48.94% (b_B) accuracy. By increasing the coverage of input word embeddings using machine translation the proposed system obtains an increased accuracy of 50.79% (b_{BO}). This improvement in accuracy, again, justifies the use of translated words for obtaining the word embeddings. Further, with the inclusion of target-side lexicon based features our proposed approach reports a significant performance improvement of approximately 6-10 points for all the three archi-

tectures (b_{BOF1}, b_{BOF2} & b_{BOF3}).

Results of English-French cross-lingual scenario are reported in row 4 of the four-boxed nodes in Figure 3. We observe similar phenomenon in cross-lingual setup with the English-French case as well. The baseline system, where we utilize separate monolingual WE for training and testing in English and French respectively, reports an accuracy of 50.69% (d_M), while employing bilingual embeddings the system obtains a sharp jump of approx. 13 points with an accuracy value of 63.64% (d_B). Further, with the inclusion of OOV words and lexicon features performance of the system improves to 65.32% (d_{BO}) and 69.49% (d_{BOF3}), respectively.

We observe four phenomena from these results: i) use of lexicon-based features is the driving force in predicting the sentiment; ii) qualitative lexicons of the resource-rich language can assist in solving the problems of resource-poor languages; iii) embeddings of the OOV words improves the performance of the system with or without assistance of extra features; and iv) use of separate LSTMs (one for word embeddings and the other for features) helps the network to efficiently extract relevant features for prediction without interfering each other (except for the multi-lingual English-French scenario).

4.3 Comparative Analysis

Comparative results reported in Figure 4 show that our proposed system clearly outperforms the baseline model in both the setups and for both the language pairs. In multi-lingual setup, we compare the proposed model against three state-of-the-art systems (Akhtar et al., 2016a; Singhal and Bhattacharyya, 2016; Akhtar et al., 2016b) for English-Hindi case. An accuracy of 65.96% was reported by the system (Akhtar et al., 2016b), while the system (Singhal and Bhattacharyya, 2016) obtained an accuracy of 68.31%. However, our proposed system reports an accuracy of 76.29%, which is approx. 10% & 8% higher compared to the systems of (Akhtar et al., 2016b) and (Singhal and Bhattacharyya, 2016) respectively. In English-French case, our proposed system reports an improvement of approx. 6 points over the baseline. For cross-lingual setup in English-Hindi case, we compare our proposed method with the state-of-the-art system proposed in (Barnes et al., 2016; Singhal and Bhattacharyya, 2016). On the same dataset their systems reported to have achieved an accuracies

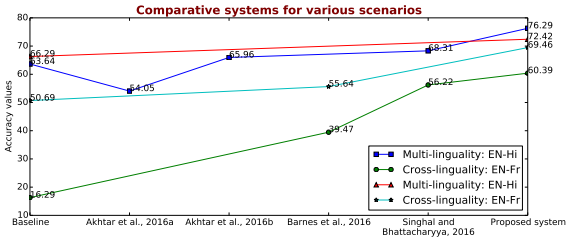

Figure 4: Comparison with the baseline and state-of-the-art methods.

of 39.47% & 56.22% as compared to 60.39% of our proposed system. In English-French case, the system proposed in (Barnes et al., 2016) obtains accuracy value of 55.64% against 69.49% in our proposed architecture. Statistical significance tests (*t*-test) confirm that performance increments in the proposed model are significant w.r.t. state-of-the-art methods with *p*-value=0.03 and *p*-value=0.01 respectively in multi-lingual and cross-lingual setups.

4.4 Discussions

The prime motivation of our current work is to minimize the effect of *data sparsity* while learning through deep neural network architecture. For this, we propose to use bilingual embeddings computed from a parallel corpus, which is created utilizing a MT system. Similarly, absence of a large aligned corpus in resource-poor language can be addressed through the application of a MT system. Since, the MT system is not fully accurate, there must be some errors introduced while translating. This, in turn, affects the bilingual word embedding. Another limitation of our work is that 7.2M sentences is not a big number in terms of word embedding computation. However, the underlying method performs considerably better compared to the state-of-the-art systems, even with all these constraints.

To show the effectiveness of bilingual embeddings in minimizing data sparsity, we also experiment with a mono-lingual Hindi embeddings computed on 53M sentences. Following the proposed approach (except computing embeddings for OOV words), we obtain an accuracy of 77.74% in aspect classification task. Table 1 shows comparison with mono-lingual and multi-lingual approach for classification. Despite all those limitations discussed above (i.e. SMT error & corpus size), the proposed method with bilingual embeddings (76.29%) performs considerably at par against the monolin-

gual embeddings created from a very large corpus of 53M (77.74%). However, the monolingual WE computed using the same amount of corpus (i.e. 7.2M sentences) produces an accuracy of only 63.64%. Further with the help of lexicon based features accuracy of this system increases to 70.86% (compared to 76.29% of our proposed model). It is also to be observed that performance of the system is improved by just including representations of the OOV words. Performance of the proposed system would have been much better if we would not have above mentioned limitations.

Models	Bilingual	Models	Monolingual	
	Size = 7.2M		Size = 7.2M	Size = 53M
Bilingual	62.51	Monolingual	63.64	68.74
Bilingual + Embedding (OOV)	64.83			
Bilingual + Embedding (OOV) + Feature (Eng)	76.29	Monolingual + Feature (Eng)	70.86	77.74

Table 1: Comparative analysis of monolingual embeddings and bilingual embeddings in multi-lingual setup.

4.5 Error Analysis

We perform error analysis on the obtained results. Quantitatively, '*neutral*' is the most problematic class in both multi-lingual and cross-lingual setups. It mainly confuses with '*positive*' class. Approximately, 20% & 40% of '*neutral*' instances are tagged as '*positive*' in multi-lingual and cross-lingual setups, respectively. Our system does not predict '*conflict*' class at all, possibly due to the insufficient number of instances for training. Qualitatively, following are the few cases where our system performs below par.

- **Lack of polar information inside context:** Our system finds it challenging to classify sentiment of the aspect terms whose polar information lie outside the context window. In the following sentence aspect term is 'वज़न |weight' and the actual sentiment towards it is positive. The polar information 'तुलना में लगभग आधा |about half as compared' and 'हल्का |lighter' are far from the aspect term, hence, not captured within the context window.
  Devanagari: इसका वज़न नए आईपैड की तुलना में लगभग आधा है और यह अन्य उपलब्ध 7-इंच टेबलेट्स से भी हल्का है।
  Transliteration: isakA vaZana nae AIpaiDa kI tulanA meM lagabhaga AdhA hai aura yaha anya upalabdha 7-iMcha TebaleTsa se bhI halkA hai.
  Translation: Its weight is about half as compared to the new iPad and it is lighter than other available 7-inch tablets.

- **Implicit sentiment:** Presence of implicit sentiment is not correctly classified by the proposed system. Following review contains 'बनावट |built' as an aspect term and its negative sentiment is derived from the phrase 'प्लास्टिक फील |plastic feel'.
  Devanagari: इस टेबलेट की बनावट काफी प्लास्टिक फील देता है।
  Transliteration: isa TebaleTa kI banAvaTa kAphI plAsTika phIla detA hai.
  Translation: The built of this tablet gives a fairly plastic feel.

5 Conclusion

In this paper, we present a deep learning based LSTM architecture built on top of bilingual word embeddings for aspect level sentiment classification. Bilingual word embeddings try to bridge the language barrier between a resource-rich and resource-poor languages in a shared vector space. We propose to reduce the effect of data sparsity in a resource-poor language word embeddings by projecting OOV words into target side and utilize the target side word embeddings. In addition, we also exploit various resources of English for assisting the proposed model. We show the effectiveness of the proposed method in two different setups, i.e. multi-lingual and cross-lingual. Experimental results show that the proposed system outperforms various state-of-the-art systems in both the setups. In future, we would like to explore the application of proposed method in another aspect level sentiment analysis task known as aspect term extraction or opinion target extraction.

6 Acknowldgements

Asif Ekbal acknowledges Young Faculty Research Fellowship (YFRF), supported by Visvesvaraya PhD scheme for Electronics and IT, Ministry of Electronics and Information Technology (MeitY), Government of India, being implemented by Digital India Corporation (formerly Media Lab Asia).

References

Md Shad Akhtar, Asif Ekbal, and Pushpak Bhattacharyya. 2016a. Aspect based Sentiment Analysis in Hindi: Resource Creation and Evaluation.

In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016), May 23-28, 2016*. European Language Resources Association (ELRA), Portorož, Slovenia, pages 2703–2709.

Md Shad Akhtar, Ayush Kumar, Asif Ekbal, and Pushpak Bhattacharyya. 2016b. A Hybrid Deep Learning Architecture for Sentiment Analysis. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016): Technical Papers, December 11-16, 2016*. Osaka, Japan, pages 482–493.

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC 2010), May 17-23, 2010*. European Language Resources Association (ELRA), Valletta, Malta, pages 2200–2204.

Dzmitry Bahdanau, Tom Bosc, Stanislaw Jastrzebski, Edward Grefenstette, Pascal Vincent, and Yoshua Bengio. 2017. Learning to Compute Word Embeddings On the Fly. *CoRR* abs/1706.00286. http://arxiv.org/abs/1706.00286.

Akshat Bakliwal, Piyush Arora, and Vasudeva Varma. 2012. Hindi Subjective Lexicon: A Lexical Resource For Hindi Polarity Classification. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC 2012), May 21-27, 2012*. Istanbul, Turkey, pages 1189–1196.

A. R. Balamurali, Aditya Joshi, and Pushpak Bhattacharyya. 2012. Cross-Lingual Sentiment Analysis for Indian Languages using Linked WordNets. In *Proceedings of the 24th International Conference on Computational Linguistics (COLING): Posters, 8-15 December 2012*. Mumbai, India, pages 73–82.

Jeremy Barnes, Patrik Lambert, and Toni Badia. 2016. Exploring Distributional Representations and Machine Translation for Aspect-based Cross-lingual Sentiment Classification. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016): Technical Papers, December 11-16, 2016*. Osaka, Japan, pages 1613–1623.

Bhuwan Dhingra, Hanxiao Liu, Ruslan Salakhutdinov, and William W. Cohen. 2017. A Comparative Study of Word Embeddings for Reading Comprehension. *CoRR* abs/1703.00993. http://arxiv.org/abs/1703.00993.

Xiaowen Ding, Bing Liu, and Philip S. Yu. 2008. A Holistic Lexicon-based Approach to Opinion Mining. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*. ACM, New York, NY, USA, WSDM '08, pages 231–240.

Deepak Kumar Gupta, Kandula Srikanth Reddy, Asif Ekbal, et al. 2015. PSO-ASent: Feature Selection Using Particle Swarm Optimization for Aspect Based Sentiment Analysis. In *Natural Language Processing and Information Systems (NLDB 2015), June 17-19 2015*. Springer, Passau, Germany, pages 220–233.

Vasileios Hatzivassiloglou and Kathleen R. McKeown. 1997. Predicting the Semantic Orientation of Adjectives. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Madrid, Spain, pages 174–181. https://doi.org/10.3115/976909.979640.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9(8):1735–1780.

VS Jagtap and Karishma Pawar. 2013. Analysis of Different Approaches to Sentence-level Sentiment Classification. *International Journal of Scientific Engineering and Technology (ISSN: 2277-1581) Volume* 2:164–170.

Aditya Joshi, AR Balamurali, and Pushpak Bhattacharyya. 2010. A Fall-back Strategy for Sentiment Analysis in Hindi: a Case Study. In *Proceedings of the 8th International Conference on Natural Language Processing (ICON 2010)*. Kharagpur, India.

Rasoul Kaljahi and Jennifer Foster. 2016. Detecting Opinion Polarities using Kernel Methods. In *Proceedings of the Workshop on Computational Modelling of People's Opinions, Personality, and Emotions in Social Media*. Osaka, Japan, pages 60–69.

Soo-Min Kim and Eduard Hovy. 2004. Determining the sentiment of opinions. In *Proceedings of the 20th international conference on Computational Linguistics*. Association for Computational Linguistics, page 1367.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *CoRR* abs/1412.6980. http://arxiv.org/abs/1412.6980.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*. Association for Computational Linguistics, pages 177–180.

Ayush Kumar, Sarah Kohail, Asif Ekbal, and Chris Biemann. 2015. IIT-TUDA: System for Sentiment Analysis in Indian Languages Using Lexical Acquisition. In *Mining Intelligence and Knowledge Exploration*, Springer, pages 684–693.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Bilingual Word Representations with Monolingual Quality in Mind. In *NAACL Workshop on Vector Space Modeling for NLP*. Denver, United States, pages 151–159.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781* .

Saif M. Mohammad, Mohammad Salameh, and Svetlana Kiritchenko. 2016. How Translation Alters Sentiment. *Journal of Artificial Intelligence Research* 55(1):95–130. http://dl.acm.org/citation.cfm?id=3013558.3013562.

Bo Pang and Lillian Lee. 2005. Seeing Stars: Exploiting Class Relationships for Sentiment Categorization with Respect to Rating Scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, Stroudsburg, PA, USA, ACL '05, pages 115–124. https://doi.org/10.3115/1219840.1219855.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar, pages 1532–1543. http://www.aclweb.org/anthology/D14-1162.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphee De Clercq, Veronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Núria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. SemEval-2016 Task 5: Aspect Based Sentiment Analysis. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*. Association for Computational Linguistics, San Diego, California, pages 19–30. http://www.aclweb.org/anthology/S16-1002.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*. Dublin, Ireland, pages 27–35.

Soujanya Poria, Erik Cambria, and Alexander Gelbukh. 2016. Aspect Extraction for Opinion Mining with a Deep Convolutional Neural Network. *Knowledge-Based Systems* 108:42–49.

Prerana Singhal and Pushpak Bhattacharyya. 2016. Borrow a Little from your Rich Cousin: Using Embeddings and Polarities of English Words for Multilingual Sentiment Classification. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016): Technical Papers, December 11-16, 2016*. Osaka, Japan, pages 3053–3062.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research* 15:1929–1958. http://jmlr.org/papers/v15/srivastava14a.html.

P. D. Turney. 2002. Thumbs up or thumbs down?: Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Association for Computational Linguistics (ACL)*. Philadelphia, USA, pages 417–424.

Janyce Wiebe and Rada Mihalcea. 2006. Word Sense and Subjectivity. In *Proceedings of the 21st International Conference on Computational Linguistics (COLING) and the 44th Annual Meeting of the Association for Computational Linguistics (ACL)*. Association for Computational Linguistics, Stroudsburg, PA, USA, ACL-44, pages 1065–1072. https://doi.org/10.3115/1220175.1220309.

Xinjie Zhou, Xiaojun Wan, and Jianguo Xiao. 2016. Cross-Lingual Sentiment Classification with Bilingual Document Representation Learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)*. Berlin, Germany, pages 1403–1412.