

Kamal Sarkar*

Sentiment Polarity Detection in Bengali Tweets Using Deep Convolutional Neural Networks

<https://doi.org/10.1515/jisys-2017-0418>

Received August 14, 2017; previously published online March 13, 2018.

Abstract: Sentiment polarity detection is one of the most popular sentiment analysis tasks. Sentiment polarity detection in tweets is a more difficult task than sentiment polarity detection in review documents, because tweets are relatively short and they contain limited contextual information. Although the amount of blog posts, tweets and comments in Indian languages is rapidly increasing on the web, research on sentiment analysis in Indian languages is at the early stage. In this paper, we present an approach that classifies the sentiment polarity of Bengali tweets using deep neural networks which consist of one convolutional layer, one hidden layer and one output layer, which is a soft-max layer. Our proposed approach has been tested on the Bengali tweet dataset released for Sentiment Analysis in Indian Languages contest 2015. We have compared the performance of our proposed convolutional neural networks (CNN)-based model with a sentiment polarity detection model that uses deep belief networks (DBN). Our experiments reveal that the performance of our proposed CNN-based system is better than our implemented DBN-based system and some existing Bengali sentiment polarity detection systems.

Keywords: Sentiment analysis, opinion mining, deep learning, convolutional networks, deep belief networks, Indian languages.

1 Introduction

Over the past decade, there has been an exponential growth in the use of micro-blogging services such as Twitter and access to mobile phones worldwide. In order to enhance customer satisfaction, online merchants enable their customers to express their opinions about the products or the services they have purchased. A vast amount of blog posts, tweets and comments from the internet users are also available in various domains such as politics, crickets, sports, movies and music. This huge amount of online reviews, tweets and other opinion-related information can be used for various applications such as deriving market intelligence and evaluation of public opinions and views.

“Opinion mining” and “sentiment analysis (SA)” are two terms used interchangeably. Sentiment is defined as a feeling or an opinion, especially one based on emotions [4]. It also refers to an attitude, subjective impression, thought or judgment, which are prompted by feelings. Opinions describe people’s sentiments, appraisals or feelings towards entities and events and their properties. SA is a challenging task because detecting feeling or opinion expressed in textual documents requires deeper natural language processing (NLP) techniques to be combined with data mining tasks. Although there are many issues related to different kinds of SA tasks such as subjectivity detection, opinion extraction and irony detection, the most widely studied task is perhaps sentiment polarity classification which aims to determine

*Corresponding author: Kamal Sarkar, Department of Computer Science and Engineering, Jadavpur University, Kolkata-700032, India, e-mail: jukamal2001@yahoo.com

the overall sentiment orientation (positive, neutral and negative) of the opinions contained within a given document.

Nowadays, as more and more users express their political and religious views on Twitter, tweets become valuable sources of people's opinions. Tweets data can be efficiently used to infer people's opinions for marketing or social studies [12, 38]. So SA of tweets is also an urgent need. SA of short texts such as Twitter messages is challenging because they contain limited contextual information.

In recent years, many studies have been conducted in the particular area of sentiment polarity classification. Two approaches are commonly used for sentiment polarity classification: task-lexicon-based approach and machine-learning-based approach. In lexicon-based approach [24, 29], polarity information about the words in the input text is retrieved from a sentiment lexicon which consists of collection of words where each word is stored along with its polarity information. Using this lexicon information, for a review document or a tweet, the probability of being in a class – either positive class, neutral class or negative class – is computed. In the machine-learning-based approach, machine learning algorithms [5, 7, 16, 27, 30] are trained with hand-crafted features for classifying sentiments of text. The supervised machine learning approach is a relatively popular approach in the area of sentiment polarity classification. The supervised method of sentiment polarity classification uses machine learning algorithms by training a machine learning classifier on a sentiment polarity labeled corpus (where each review or tweet is manually labeled as positive/negative/neutral), and then it is applied to classify new unseen texts. Each textual document is turned into a feature vector. The commonly used features are words, n-grams, surrounding words, punctuation, etc. To apply machine learning algorithms such as support vector machine (SVM), multinomial NB and decision trees to a sentiment polarity classification task, we need to represent each input text as a numeric vector. It is also mandatory to translate each document or tweet into a numeric vector of equal length. This requirement makes the vector size equal to the vocabulary size and hence the vectors become very sparse. This problem becomes worse when we represent short texts like tweets.

Most recently, neural networks have been a most popular topic in the field of NLP, including SA. When more computing power and a large amount of training data are available, the deep-learning-based approach to sentiment polarity classification [11, 17, 35] can give more accurate results than the traditional sentiment classification methods that use SVM, NB, etc. The strength of the deep learning model is that it does not require any human intervention for feature engineering for designing a set of hand-crafted features. Instead of using a set of hand-crafted features, a deep learning model takes an input-dense vector representation of words called word embedding [25], which contains contextual information, and the intermediate layers of the deep neural network learn the features during the training phase itself.

Most of the previous research works on SA involve the analysis of sentiments of text units written in English language [19, 34, 39]. But since India is a multilingual country, many Indians share their opinions or write on the twitter using their mother language or mixture of several languages. So, there is also a need for developing a system that can do SA of texts written in Indian languages. A recent commendable attempt to promote research on SA in Indian language tweets has been made through conducting a shared task on Sentiment Analysis in Indian Languages (SAIL) Tweets, co-located with MIKE 2015 conference held at IIIT Hyderabad, India [28].

Bengali is the most spoken language in Bangladesh, West Bengal, Assam and Tripura. Bengali is the second most spoken language in India. With about 230 million Bengali speakers spread all over the world, the Bangla language is also one of the most spoken languages in the world. To our knowledge, a few number of attempts have been made by the researchers for SA of Bengali tweets [20, 32]. Our research focus is in the area of the sentiment polarity classification of Bengali tweets, and we use deep neural networks for this task. We have also incorporated in our proposed sentiment polarity detection system some lexical knowledge extracted from Bengali SentiWordNet¹ (lexical resource for opinion mining) supplied for SAIL contest [28].

¹ <http://amitavadas.com/sentiwordnet.php>, retrieved in 2015.

2 Related Work

Although most of the research papers on SA present approaches to SA of English texts, work on other languages is still growing. There are some previous research works that have dealt with SA of non-English texts such as German, Chinese, Arabic, Bengali and Hindi.

Zhang et al. [40] used a rule-based approach to Chinese SA which works in two phases: (1) determining each sentence's sentiment based on word dependency, and (2) aggregating sentences in order to predict the document sentiment. Wan (2009) [36] made use of bilingual knowledge including both Chinese resources and English resources for sentiment polarity identification of Chinese product reviews. In this work, Wan (2009) [36] translated a corpus of Chinese documents into English by using Google Translate and Yahoo Babel Fish. In addition, he applied ensemble methods to combine the individual results over Chinese and English datasets. The results for the combination methods improved the performance of individual results.

Ghorbel and Jacot [13] used a corpus with movie reviews in French. They applied a supervised classification combined with SentiWordNet in order to determine the polarity of the reviews.

Rushdi-Saleh et al. [31] presented a corpus of movies reviews in Arabic annotated with polarity and performed several experiments using machine learning techniques. Al-Ayyoub et al. [1] presented Lexicon-based SA of Arabic tweets.

Banea et al. [3] proposed several approaches to cross-lingual subjectivity analysis by directly applying the translations of opinion corpus in English to training an opinion classifier in Romanian and Spanish. This work showed that automatic translation is a viable alternative for the construction of resources and tools for subjectivity analysis in a new target language. A comparative study of NLP techniques applied to SA and topic detection of Spanish tweets has been presented in [2]. Martínez-Cámara et al. [23] applied the supervised approach to SA of Spanish movie reviews using different machine learning algorithms such as SVM, naive Bayes, Bayesian based logistic regression, K-nearest neighbor (KNN) and C4.5.

Das and Bandyopadhyay [9] presented an approach to opinion polarity classification on news texts for Bengali using SVM. Their system identifies semantic orientation of an opinionated phrase as either positive or negative. They also developed a subjectivity classifier for performing sentence level subjectivity classification. Chowdhury and Chowdhury [6] used a semi-supervised bootstrapping approach for the development of the training corpus which avoids the need for labor-intensive manual annotation. For classification, they used SVM and maximum entropy with a combination of various sets of features. SA of Bengali tweets using multinomial NB with n-gram and SentiWordNet features has been presented in [32]. An approach presented in [20] uses a more sophisticated system that uses an unsupervised approach for expanding a (small) Indian sentiment lexicon, leveraging distributional thesauri, sentence level co-occurrence statistics and SVM.

Joshi et al. [15] presented a study on SA for Hindi. They studied three approaches to perform SA in Hindi: the first approach involves training a classifier on this annotated Hindi corpus and using it to classify a new Hindi document, the second approach translates the given document into English and uses a classifier trained on standard English movie reviews to classify the document and third approach uses a lexical resource called Hindi-SentiWordNet and implements a majority-score-based strategy to classify the given document. They proposed a fall-back strategy for SA in Hindi which applies any one of the above-stated three strategies based on the availability of in-language training data or translation system from Hindi to a resource-rich language or SentiWordNet. Sharma et al. [33] presented unsupervised lexicon-based SA of Hindi tweets. Mittal et al. [26] presented SA of Hindi review based on a negation and discourse relation.

A recent commendable attempt to promote research on SA in Indian language tweets has been made through conducting a shared task on SAIL Tweets, co-located with MIKE 2015 conference held at IIIT Hyderabad, India. In this task, the organizers provided tweets for three Indian languages, namely, Bengali, Hindi and Tamil, annotated with positive, negative and neutral polarity. The main objective of this task was to classify the tweets into positive, negative and neutral categories [28]. Most teams participated in the contest used machine-learning-based approaches such as SVM [20], multinomial NB [32] and decision trees. Sarkar and Chakraborty [32] applied multinomial NB for SA for Indian languages, including Bengali and Hindi. Kumar et al. [20] used SVM for SA for Bengali and Hindi tweets.

Most of the previous approaches to SA have used supervised machine learning algorithms such as SVM, multinomial NB, decision tree and KNN for sentiment polarity detection. Recently neural networks have been the most popular topic in the field of NLP, including SA. Deep-learning-based approach to SA [11, 17, 35] has received more attention of the researchers conducting research on various SA tasks.

Kim [17] used a simple convolutional neural networks (CNN) with one layer of convolution on top of word vectors obtained from an unsupervised neural language model. These vectors were trained on 100 billion words of Google News. This work showed that a simple CNN with little hyper-parameter tuning and static vectors achieved excellent results on multiple benchmarks. Socher et al. [35] introduced Sentiment Treebank which includes fine-grained sentiment labels for 215,154 phrases in the parse trees of 11,855 sentences. They used a recursive neural tensor network (RNTN) for predicting fine-grained sentiment for all nodes in the parse trees of sentences. Sentiment label assigned at the root of the parse tree of a sentence is considered as the label of the sentence. They showed that the RNTN performs better than state-of-the-art sentiment classification approaches when it is tested on a test set of positive and negative sentences. Dos Santos and Gatti [11] presented an approach that uses deep convolutional neural networks for the sentiment analysis of short texts. Wang et al. [37] combined a convolutional and recurrent neural network for the sentiment analysis of short texts.

3 CNN-Based Model for Sentiment Polarity Detection

The model architecture that we have used for sentiment polarity detection of Bengali tweets is shown in Figure 1. This architecture is a slight variation of the architecture presented in [8, 17]. As we can see from Figure 1, a tweet is represented as an $n \times k$ matrix where n is the number of words in the tweet and k is the dimension of the word vector. According to this architecture, the input is the concatenated vector of the word vectors corresponding to the words in a tweet (padding zeros is necessary for equalizing the length). The input is submitted to the convolutional layer that performs convolutional operation involving a filter W of size $h \times k$ where h is the size of the window of words and k is the dimension of the word vector. A new feature f_i , which is produced after applying a convolutional operation using a filter W applied to the window of h words, can be expressed as follows:

$$f_i = f(WX_{(i:i+h-1)} + b)$$

where $X_{(i:i+h-1)}$ is the concatenation of vectors of words within the window spanning from the i th word to the $(i+h-1)$ th word in a sentence, b is the bias weight and f is the non-linear function which may be a hyperbolic tangent or rectilinear function.

The filter is applied to each possible window of words obtained by sliding right with stride equal to 1. Thus a feature map f is obtained as follows:

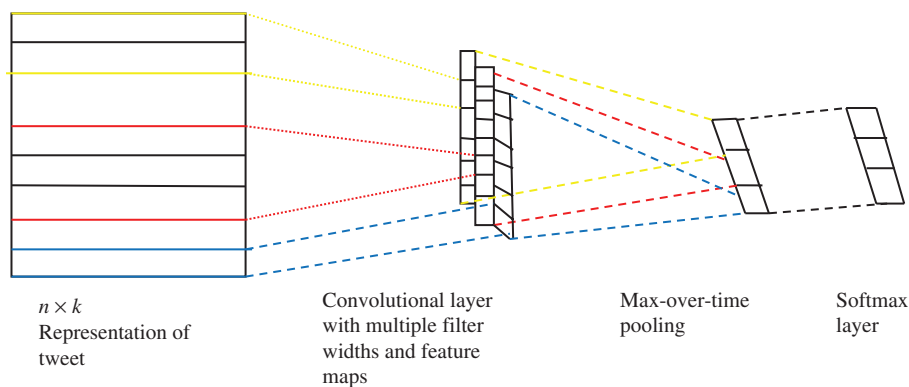


Figure 1: Model Architecture for an Example Tweet.

$$F = [f_1 f_2, \dots, f_{(n-h+1)}],$$

where n is the length of the tweet and h is the window size.

After obtaining the feature maps (number of feature maps depends on the number of filters), we apply max pooling [22] which selects the most important feature from each feature map by finding the maximum value from each feature map f . The pooling can also deal with issues related to the variable sentence length.

If we use m filters (each with varying window size), for each filter, we can get one convoluted feature. Thus, m filters give rise to a feature vector of length m .

After the convolution layer, we add a fully connected layer which accepts the feature vector of length m . The output of this layer is passed to another fully connected layer which is basically the softmax layer whose output is the probability distribution over labels.

4 Our Proposed Sentiment Polarity Detection Method

The various components of our proposed sentiment polarity detection system are shown in Figure 2. Our proposed system has four major components: preprocessing, data representation, training a CNN model and saving the model for future prediction.

Since we deal with the tweet data which are usually noisy data, we need to apply preprocessing on the training data as well as test data. We preprocessed the entire data collection to remove irrelevant characters from the data. For each tweet in our corpus, we remove irrelevant characters such as [, -, _, =, :, +, \$, @, ~, !, ,, ;, /, ^,), (,], {, }, <, >. Words which have occurred only once in the corpus are also removed from the tweets as irrelevant features. In the preprocessing step, we also incorporate some lexical knowledge to enhance the sentiment information contained in the tweets. We use SentiWordNet [10] for finding the sentiment polarity of the individual words contained in the tweets. SentiWordNet that we used here is basically containing three

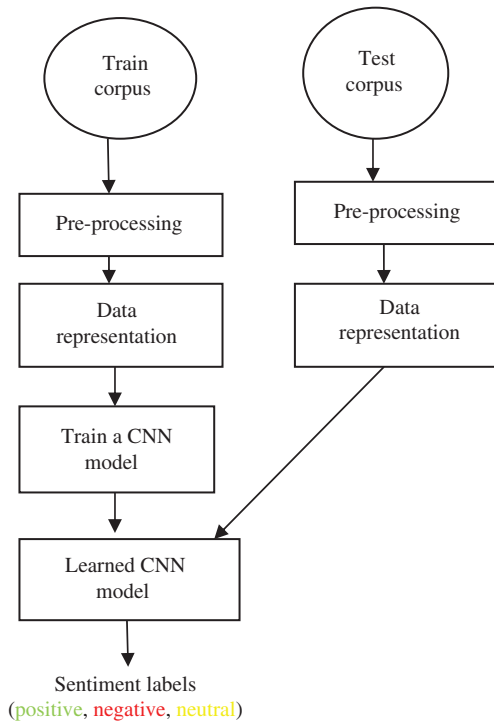


Figure 2: Block Diagram of Our Proposed System.

sets of words: a set of positive words, a set of negative words and a set of neutral words. Each tweet word is searched into SentiWordNet. If the tweet word is found in SentiWordNet, the sentiment tag of the word is retrieved from SentiWordNet and the tweet word is augmented with the retrieved sentiment tag. If the word is not found, the tweet word will be left as it is. For example, the dataset consists of the following two tweets:

Ram is a good boy.
She hates Cockroach.

After tagging with sentiment tags, we have:

Ram is a good (positive) boy.
She hates (negative) cockroach.

Here, “positive” and “negative” are sentiment tags of the words “good” and “hates,” respectively, which are retrieved from SentiWordNet.

For data presentation, we need to create word vectors for all tokens in the vocabulary. We have used our tweet corpus for creating the word vectors. There are 8483 tokens in our vocabulary. A tweet is represented as the vector of size $M \times k$ where M is the maximum size of the tweet in a corpus and k is the dimension of the word vector. The tweet whose size is smaller than the maximum size is padded with zeros. Since sentiment tags obtained from SentiWordNet are assigned to the tweet words at the preprocessing step, they are also considered as the tokens in the vocabulary.

After completion of preprocessing and data representation steps, we train our proposed CNN-based model with our training dataset. After obtaining the trained model, we apply it to our test data for sentiment polarity classification for detecting polarity of sentiment expressed in each test Bengali tweet. Before submission to the trained CNN model, test tweets are also processed using the method similar to that applied to the training data.

4.1 Hyper-Parameters and Training

We have used rectified linear units for all the convolutional and dense layers. For the output layers, we have used the *softmax* unit. For the convolutional layer, we have used the following parameter settings: embedding vector length is set to 32, number of filters is set to 3 and filter window size (h) is set to 3. The dimension of the fully connected layer added after the convolutional layer is set to 30 and the last layer which is basically a *softmax* layer has three output nodes since our sentiment polarity detection problem is a three-class problem. Training of the model is done with a mini-batch size of 10. For training and testing our proposed system, we have used Bengali datasets released for a shared task on SAIL Tweets, co-located with MIKE 2015 conference held at IIIT Hyderabad, India [28]. Since there was no development set released for MIKE contest, we randomly select 10% of the training data as the development set for parameter tuning and we obtain the best results with the above-stated parameter settings. Training is done through the stochastic optimization method called “Adam” [18] with the learning rate set to 0.001. The model is trained up to 100 epochs.

5 Evaluation and Results

We have used Bengali datasets released for a shared task on SAIL Tweets, co-located with MIKE 2015 conference held at IIIT Hyderabad, India [28]. The description of the Bengali sentiment dataset is shown in Table 1.

The training data released for SAIL 2015 contest were distributed in three different files containing negative, neutral and positive tweets. We have conducted the following two experiments:

Experiment 1: In this experiment, we train, tune and test our proposed CNN-based model presented earlier in this paper.

Table 1: Description of the Bengali Sentiment Dataset.

Class	Train data	Test data
Positive	276	215
Negative	356	151
Neutral	368	134
Total	1000	500

Experiment 2: This experiment is conducted to compare our proposed CNN-based sentiment polarity detection model with the traditional deep belief networks (DBN)-based model presented in [14]. Here we have used deep belief neural networks which are basically deep multilayer perceptron neural networks having three hidden layers. It does not include any convolutional layer. To train DBN, we have used a two-step training process similar to the approach proposed in [14]:

- Layer-wise pretraining in unsupervised mode
- Initializing the whole DBN with the weights learnt in unsupervised mode and then fine-tuning weights in the DBN in supervised mode using labeled data.

The training phase of the DBN-based SA system has several steps: (1) preprocessing of data, (2) data representation, (3) layer-wise pretraining of the DBN, (4) assembling all trained layers to create a deep multi-layer perceptron and (5) fine-tuning the DBN in supervised mode using labeled data.

Data preprocessing part is similar to the CNN-based model presented in this paper, but for data representation, both the training data and the test data are represented in the form of a numeric feature vectors using the bag-of-words model which represents each tweet as a vector of size V where V is the size of the vocabulary and the i th component of the vector corresponds to the weight of the i th vocabulary word when the word is present in the tweet. If the word is not present in the tweet, 0 is taken. Weight of a word is calculated using the TF*IDF method.

We have used the DBN with three hidden layers, one input layer and a *softmax* output layer. The size of the input layer is 6225 which is equal to the vocabulary size for our training data. The sizes of three hidden layer sizes are set to 300, 50 and 50, respectively. In the pretraining phase, the greedy algorithm was used to train each layer of weights separately, starting at the bottom. Each layer was pretrained for 30 epochs using the training set with the learning rate set to 0.05. The final network is trained up to 1000 epochs with the learning rate set to 0.1 and batch size set to 10.

5.1 Results

For system performance evaluation, we have used the traditional accuracy measure and reported the results in Table 2. As we can see from Table 2, the performance of the CNN-based SA model is better than the traditional DBN-based model. We have also compared our proposed models with some top SA systems which participated in SAIL 2015 contest. The comparisons of the results are shown in Table 3. Since we have used the training corpus and SentiWordNet for developing our system, we have only compared our obtained results with SAIL 2015 official results released for Bengali constrained runs. In Bengali constrained mode of SAIL 2015 contest, for system development, the contestants were allowed only to use the released training corpus and SentiWordNet.

Table 2: Comparisons of Our Proposed Sentiment Analysis Models.

Models	Accuracy
CNN-based model	46.80
DBN-based model	43

Table 3: Comparisons of Our Proposed Models with Top Three Systems Participated in SAIL 2015 Contest (Constrained Mode).

Systems	Accuracy
Our proposed CNN-based model	46.80
IIT-TUDA [20]	43.20
Our proposed DBN-based sentiment analysis model	43.00
JUTeam_KS [32]	41.20
AmritaCENNLP [21]	31.4

The analysis of the results shown in Table 3 reveals that our proposed CNN-based SA model performs better than the best system (IIT-TUDA) participated in SAIL 2015 contest on SA of Bengali tweets. We have used a deep-learning-based approach for implementing our sentiment polarity detection system, whereas the system named IIT-TUDA participated in SAIL 2015 contest used an unsupervised approach for expanding a (small) Indian sentiment lexicon, leveraging distributional thesauri, sentence level co-occurrence statistics and SVM [20]. The system JUTeam_KS [32] has used multinomial Naïve Bayes with unigram, bigram and SentiWordNet features for SA of Bengali tweets.

6 Conclusions

In this paper, we have presented an approach to sentiment detection of Bengali tweets using deep convolutional neural networks. We have compared the performance of our proposed CNN-based Bengali sentiment polarity detection model with the DBN-based model. One of the critical problems that we faced for our experimentations is scarcity of benchmark datasets for SA in Indian languages. Although we have used the datasets released for SAIL contest 2015 [28], it is insufficient and noisy. In future, we have planned to rectify errors in the training data and create my own data for developing a more accurate sentiment classification model for Bengali tweets.

Another future plan is exploitation of unlabeled data in the training process of DBN- as well as CNN-based models. For improving system performance, some other kinds of deep neural networks such as deep recursive neural networks [35] can also be used.

The model proposed in this paper can be easily extended to other Indian languages such as Hindi and Tamil if training data for the corresponding language are available.

Bibliography

- [1] M. Al-Ayyoub, S. B. Essa and I. Alsmadi, Lexicon-based sentiment analysis of Arabic tweets, *Int. J. Soc. Netw. Min.* **2** (2015), 101–114.
- [2] A. F. Anta, L. N. Chiroque, P. Morere and A. Santos, Sentiment analysis and topic detection of Spanish tweets: a comparative study of NLP techniques, *Procesamiento del Lenguaje Natural* **50** (2013), 45–52.
- [3] C. Banea, R. Mihalcea, J. Wiebe and S. Hassan, Multilingual subjectivity analysis using machine translation, in: *EMNLP*, pp. 127–135, ACL, Honolulu, Hawaii, 2008.
- [4] J. Bowker, *The oxford dictionary of world religions*, Oxford University Press, USA, 1997.
- [5] C. C. Chen and Y. D. Tseng, Quality evaluation of product reviews using an information quality framework, *Decis. Support Syst.* **50** (2011), 755–768.
- [6] S. Chowdhury and W. Chowdhury, Performing sentiment analysis in Bangla microblog posts, in: *International Conference on Informatics, Electronics and Vision (ICIEV)*, pp. 1–6, IEEE, Dhaka, Bangladesh, 2014.
- [7] D. Clarke, P. Lane and P. Hender, Developing robust models for favourability analysis, in: *Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis*, pp. 44–52, Association for Computational Linguistics, Portland, USA, 2011.
- [8] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu and P. Kuksa, Natural language processing (almost) from scratch, *J. Mach. Learn. Res.* **12** (2011), 2493–2537.

- [9] A. Das and S. Bandyopadhyay, Opinion-polarity identification in Bengali, in: *International Conference on Computer Processing of Oriental Languages*, pp. 169–182, Chinese and Oriental Languages Computer Society, California, USA, 2010.
- [10] A. Das and S. Bandyopadhyay, SentiWordNet for Bangla, *Knowledge Sharing Event-4: Task 2* (2010), 1–8.
- [11] C. N. Dos Santos and M. Gatti, Deep convolutional neural networks for sentiment analysis of short texts, in: *COLING*, pp. 69–78, ACL, Dublin, Ireland, 2014.
- [12] G. G. Esparza, A. P. Díaz, J. Canul-Reich, C. A. De Luna and J. Ponce, Proposal of a Sentiment Analysis Model in Tweets for improvement of the teaching-learning process in the classroom using a corpus of subjectivity, *Int J. Combinat. Optimiz. Probl. Inform.* **7** (2016), 22.
- [13] H. Ghorbel and D. Jacot, Sentiment analysis of French movie reviews, in: *Proceedings of the 4th International Workshop on Distributed Agent-Based Retrieval Tools (DART 2010)*, Springer, Geneva, Italy, 2010.
- [14] G. E. Hinton, S. Osindero and Y. W. Teh, A fast learning algorithm for deep belief nets, *Neural Comput.* **18** (2006), 1527–1554.
- [15] A. Joshi, A. R. Balamurali and P. Bhattacharyya, A fall-back strategy for sentiment analysis in Hindi: a case study, in: *Proceedings of the 8th ICON*, Macmillan, India, 2010.
- [16] H. Kang, S. J. Yoo and D. Han, Senti-lexicon and improved Naïve Bayes algorithms for sentiment analysis of restaurant reviews, *Expert Syst. Appl.* **39** (2012), 6000–6010.
- [17] Y. Kim, Convolutional neural networks for sentence classification, in: *Proceedings of the 2014 Conference on EMNLP*, pp. 1746–1751, ACL, Doha, Qatar, 2014.
- [18] D. Kingma and J. Ba, Adam: a method for stochastic optimization, arXiv preprint arXiv:1412.6980, 2014.
- [19] I. Korkontzelos, A. Nikfarjam, M. Shardlow, A. Sarker, S. Ananiadou and G. H. Gonzalez, Analysis of the effect of sentiment analysis on extracting adverse drug reactions from tweets and forum posts, *J. Biomed. Inform.* **62** (2016), 148–158.
- [20] A. Kumar, S. Kohail, A. Ekbal and C. Biemann, IIT-TUDA: system for sentiment analysis in Indian languages using lexical acquisition, in: *International Conference on Mining Intelligence and Knowledge Exploration*, pp. 684–693, Springer, Cham, 2015.
- [21] S. S. Kumar, B. Premjith, M. A. Kumar and K. P. Soman, AMRITA_CEN-NLP@ SAIL2015: sentiment analysis in Indian Language using regularized least square approach with randomized feature learning, in: *International Conference on Mining Intelligence and Knowledge Exploration*, pp. 671–683, Springer, Cham, 2015.
- [22] Y. LeCun, LeNet-5, convolutional neural networks (2015). Available from: <http://yann.lecun.com/exdb/lenet/>. Accessed 17 July, 2016.
- [23] E. Martínez-Cámara, M. T. Martín-Valdivia and L. A. Ureña-López, Opinion classification techniques applied to a Spanish corpus, in: *Proceedings of the 16th International Conference on Natural Language Processing and Information Systems, NLDB'11*, pp. 169–176, Springer-Verlag, Alicante, Spain, 2011.
- [24] P. Melville, W. Gryc and R. D. Lawrence, Sentiment analysis of blogs by combining lexical knowledge with text classification, in: *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1275–1284, ACM, Paris, France, 2009.
- [25] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado and J. Dean, Distributed representations of words and phrases and their compositionality, in: *Advances in Neural Information Processing Systems*, pp. 3111–3119, Curran Associates, Inc., USA, 2013.
- [26] N. Mittal, B. Agarwal, G. Chouhan, N. Bania and P. Pareek, Sentiment analysis of Hindi review based on negation and discourse relation, in: *Proceedings of International Joint Conference on Natural Language Processing*, pp. 45–50, ACL, Nagoya, Japan, 2013.
- [27] B. Pang and L. Lee, A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts, in: *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, p. 271, Association for Computational Linguistics, 2004.
- [28] B. G. Patra, D. Das, A. Das and R. Prasath, Shared task on Sentiment Analysis in Indian Languages (SAIL) tweets-an overview, in: *International Conference on Mining Intelligence and Knowledge Exploration*, pp. 650–655, Springer, Cham, 2015.
- [29] G. Ramakrishnan, A. Jadhav, A. Joshi, S. Chakrabarti and P. Bhattacharyya, Question answering via Bayesian inference on lexical relations, in: *Proceedings of the ACL 2003 Workshop on Multilingual Summarization and Question Answering*, vol. 12, pp. 1–10, Association for Computational Linguistics, Sapporo, Japan, 2003.
- [30] A. Reyes and P. Rosso, Making objective decisions from subjective data: detecting irony in customer reviews, *Decis. Support Syst.* **53** (2012), 754–760.
- [31] M. Rushdi-Saleh, M. T. Martín-Valdivia, L. A. Ureña-López and J. M. Perea-Ortega, OCA: Opinion corpus for Arabic, *J. Am. Soc. Inform. Sci. Technol.* **62** (2011), 2045–2054.
- [32] K. Sarkar and S. Chakraborty, A sentiment analysis system for Indian language tweets, in: *International Conference on Mining Intelligence and Knowledge Exploration*, pp. 694–702, Springer, Cham, 2015.
- [33] Y. Sharma, V. Mangat and M. Kaur, A practical approach to Sentiment Analysis of Hindi tweets, in: *Next Generation Computing Technologies (NGCT), 2015 1st International Conference on*, pp. 677–680, IEEE, Dehradun, India, 2015.
- [34] J. Smailović, M. Grčar, N. Lavrač and M. Žnidaršič, Predictive sentiment analysis of tweets: a stock market application, in: *Human-Computer Interaction and Knowledge Discovery in Complex, Unstructured, Big Data*, pp. 77–88, Springer, Berlin/Heidelberg, 2013.
- [35] R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng and C. Potts, Recursive deep models for semantic compositionality over a sentiment treebank, in: *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, vol. 1631, p. 1642, ACL, Seattle, USA, 2013.

- [36] X. Wan, Co-training for cross-lingual sentiment classification, in: *Proceedings of the ACL*, pp. 235–243, ACL, Singapore, 2009.
- [37] X. Wang, W. Jiang and Z. Luo, Combination of convolutional and recurrent neural network for sentiment analysis of short texts, in: *COLING*, pp. 2428–2437, ACL, Osaka, Japan, 2016.
- [38] U. Yaqub, S. Chun, V. Atluri and J. Vaidya, Sentiment based analysis of tweets during the US Presidential Elections, in: *Proceedings of the 18th Annual International Conference on Digital Government Research*, pp. 1–10, ACM, City University of New York, 2017.
- [39] S. M. Zavattaro, P. E. French and S. D. Mohanty, A sentiment analysis of US local government tweets: the connection between tone and citizen involvement, *Gov. Inform. Q.* **32** (2015), 333–341.
- [40] C. Zhang, D. Zeng, J. Li, F.-Y. Wang and W. Zuo, Sentiment analysis of Chinese documents: from sentence to document level, *J. Am. Soc. Inform. Sci. Technol.* **60** (2009), 2474–2487.