Semantic Meaning Based Bengali Web Text Categorization Using Deep Convolutional and Recurrent Neural Networks (DCRNNs)

Md. Rajib Hossain and Mohammed Moshiul Hoque(✉)

Department of Computer Science and Engineering,
Chittagong University of Engineering and Technology,
Chittagong 4349, Bangladesh
moshiul_240@cuet.ac.bd

Abstract. Web text categorization is a procedure of deliberately assigning a web text document into one of the pre-defined classes or categories. It is a very challenging task to manipulate, organize, and categorize an enormous amount of web text data in manually. This paper proposes an automatic text categorization framework to classify Bengali web text data using deep learning. The proposed framework comprises of three key constituents: text to feature extraction, training, and testing. The categorization framework is trained, validated, and tested at $120K$, $12K$, and $36K$ datasets, respectively. The proposed system achieved 99.00% accuracy in the training phase, 96.00% in the validation phase, and 95.83% in the testing phase, respectively.

Keywords: Bengali language processing · Web text categorization · Web crawler · Semantic feature extraction · Deep convolution networks · Recurrent neural network

1 Introduction

In recent years, Bengali text resources are proliferating on the web due to the cyclopean utilization of digital contrivances with Internet availability. Variants of the web pages and digital portals are the most common voluminous reserves of texts in subsistence that are liberatingly available. It covers a vast range of contexts/domains and perpetually updated by the incipient kinds of information which are multi-dimensional. Hence, it is a very arduous task to manipulate, retrieve, or categorize such an abundance of web text data efficiently. To have facilely accessible or sorted out to these enormous amounts of text data, the process of these data should accomplish with some widely accepted text processing approaches. Manual labelling of immense quantities of data into their predefined classes requires substantial time and cost of money, which are inconvenient, fallible, or impracticable. Consequently, the perspicacious text categorization technique is one of the most exceptional desirable solutions to process

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2021
R. Misra et al. (Eds.): ICIoTCT 2020, AISC 1382, pp. 494–505, 2021.
https://doi.org/10.1007/978-3-030-76736-5_45

such an immense amount of text data that truncates human labour, preserves time, and reduce the cost of money.

There are a variety of possible applications of web text categorization system. Academicians, industry, security agencies, data scientists, and some other areas are essential to access the voluminous categorical web text data. With the concern of aims, the problem of web text categorization attracted by many NLP scholars in recent years. There are various challenges to develop a text categorization system from Bengali web texts due to the scarcity of e-resources and the unavailability of the benchmark dataset. The web is a source of the large volume of text data that contains multi-dimensional information, and it is quite impossible to process and retrieve information from these raw text data manually. This work suggests a deep convolution and recurrent neural networks (DCRNNs) based on a predicated framework that automatically categorizes Bengali web texts. The proposed framework extracts the semantic text feature from the web text and engenders a DCRNNs predicated web text classifier model.

2 Previous Work

Although there are a remarkable amount of work has been conducted on classifying web texts in English, Chinese, Arabic, and other European languages, no eminent research activities are conducted till now in the Bengali language perspectives. Web text conventionally found in an unstructured form that should convert into structured textual form before utilizing to text categorization purpose. This conversion is commonly known as feature extraction. Mikolov et al. [11] were developed a prognostication predicated feature extraction algorithm name Word2Vec to extract semantic features from the unstructured texts. A couple of work used Word2Vec models to extract features and statistical machine learning techniques used to classify text documents [9,14]. The Global Vectors for Word Representation (GloVe) is a frequency-predicated text to feature representation technique that captured the semantic and syntactic linguistic features [16]. In recent years, the morphological feature extraction technique has gained more attention by the researchers for Arabic, and English text processing [21,22]. However, it is quite arduous to implement this technique for low resource languages (i.e., Bengali) due to the intricacy in hyper-parameters adjustment and desirable rule cull [5,6].

SVM and statistical machine learning-predicated techniques achieved better results for minuscule dataset than the larger dataset [1,2,9]. Stochastic gradient descent (SGD) predicated system can handle a voluminous amount of data but required colossal time and recollection for classifier model generation [12]. A supervised machine learning-based approach proposed for text categorization, which utilized SVM and C4.5 [10]. This approach is unable to deal with the semantic meaning of texts. A clustering-predicated framework designed for Bengali document categorization, but this work is unable to achieve plausible precision due to high dimensional feature space [14]. Salama et al. [13] developed a semi-supervised machine learning technique for text tag categorization

in industry perspectives. Liu et al. [4] engendered a fuzzy approach for text classification, which abstracted ambiguity is utilizing two-stage learning. It gained excellent precision but required huge training time. Bassine et al. [19] proposed a decision tree based categorization system for Arabic text, which achieved 92.9% precision in 6 different categories. A hierarchical text classification system proposed by Stein et al. [3], which utilized the word embedding technique. They used FastTex to extract semantic features and tree-predicated classification techniques.

Convolution neural networks and semantic word embedding based document categorization systems for English text documents has introduced in [15,18]. A character-CNN based model is proposed to classify short text documents which gained good accuracy but required much time in training and testing phases [17]. Another work based on Gated recurrent neural network (GRNN) and RNN techniques produced higher precision for short texts but demanded more preponderant training time for colossal scale data sets [7]. Hossain et al. [8] developed a DCN based Bengali document categorization system that dealt with 86,199 training, 10,707 test document. This system achieved acquired 94.96% accuracy for 12 categories. In this paper, GloVe algorithm used for extracting semantic and syntactic features [16]. A deep convolution and recurrent neural networks (DCRNNs) based approach propose for Bengali web text categorization purposes.

3 Methodology

The primary purpose of our research is to build up a text categorization framework that can classify or categorize the web texts (written in Bengali) using DCRNNs techniques. The framework comprises of three significant components: text to feature extraction, training, and testing. Figure 1 depicts the overview of the suggested framework.

3.1 Text to Feature Extraction Module

In the initial step, GloVe embedded model trained for feature extraction purposes using our developed corpus, which contained texts of 12 categories. The categories are accident (Acd), art (At), crime (Cm), economics (Em), education (Et), entertainment (Etm), environment (Env), health (Ht), opinion (O), politics (Plt), science & technology (Sct) and sports (S) respectively. Each text data is added zeros padding or truncates the first 2000 words ($W_1, W_2, W_3 ... W_{2000}$). Figure 1 shows a set of 200 extracted features $F_1, F_2, F_3 ... F_{200}$ for each word and add row-wise one after another. GloVe feature extractor provided a 2D (2000×200) feature matrix. The 2D feature matrix is the final feature illustration that forgiven the unstructured labelled data.

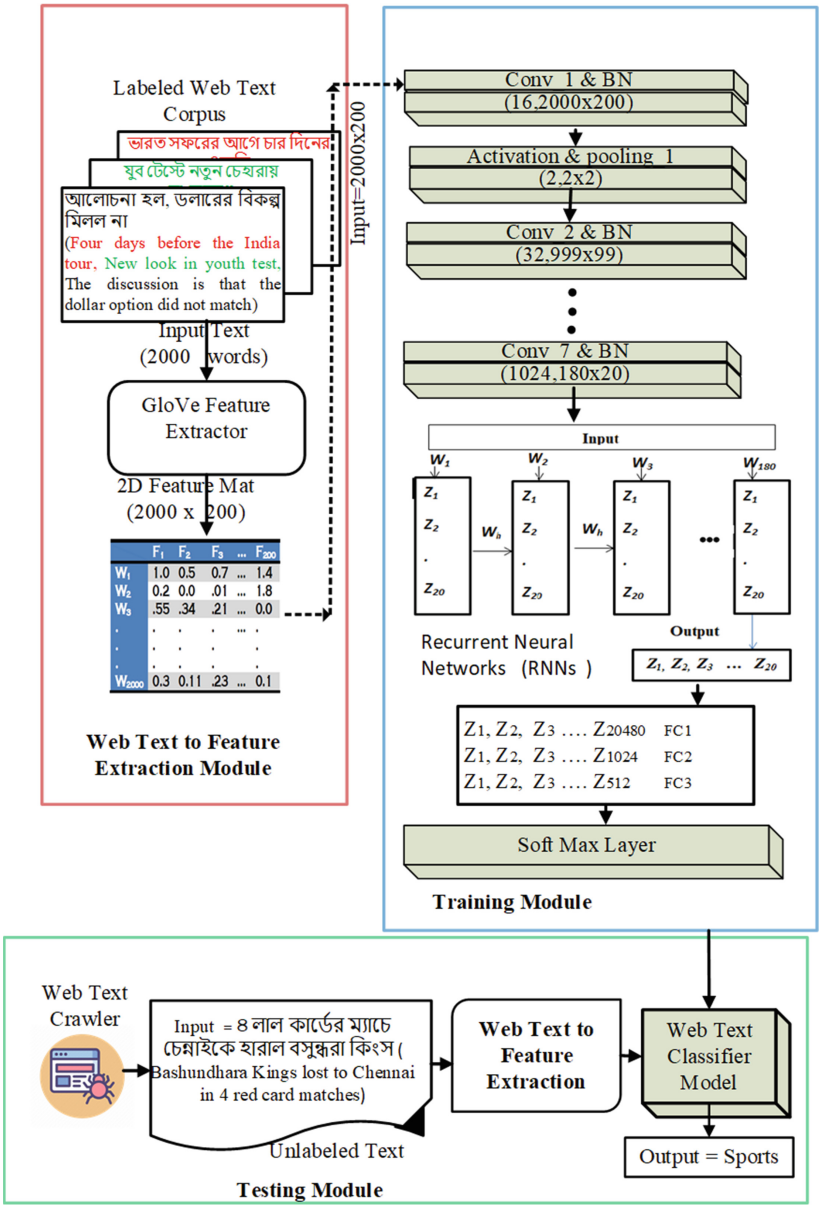

Fig. 1. Proposed framework of Bengali web text categorization.

3.2 Training Module

The propose DCRNNs architecture will train with the developed dataset (D). Suppose, a web text training corpus (WTC), where the WTC =

$D_1, D_2, D_3...D_n$, and D_n refers to the n^{th} web texts and n is the number of text documents in training corpus. Here $D = W_1, W_2, W_3...W_N$ and N is the number of words in each documents. Semantic and syntactic features is extracted for each words (W) and $W_N = F_1, F_2, F_3...F_e$, where F_e is the e^{th} feature of N^{th} word in which $e = 200$ mentioned as the embedding feature length. The input layer is feed with a $X_{2000\times200}$ feature matrix at first convolution (Conv) layer with 16 kernels $(K_{2\times2})$. For each index of feature matrix $X(i, j)$ a convolution operation is applied based on Eq. 1.

$$X(p,q) = \sum_{i=0}^{k_h+p} \sum_{j=0}^{k_w+q} X(p+i, q+j) \times K(i,j) \tag{1}$$

Where, k_h and k_w indicates the kernel height and width. Each of the convolution operations followed by the batch normalization (BN) operation. BN operation is changed the element-wise matrix values. The normalized value improves the training speed and reduces the model over-fitting problem. After BN operation, X is passing through the activation layer. Leaky ReLU activation function is applied according to Eq. 2 [20].

$$X(p,q) = max(X(p,q), (X(p,q) \times 0.01)) \tag{2}$$

Each of the index values changed after the execution of the ReLU activation function. The pooling operation reduces the feature dimension. DCRNNs framework contains seven convolution layers, three pooling layers, and four activation layers. The outputs of the last convolution layer are 1024 kernels(2×2) and $X_{180\times20}$ feature dimension. RNN layers start with 180 hidden units $(W_1, W_2, W_3...W_{180})$ and each of unit feed with 20 features $(Z_1, Z_2, Z_3, ...Z_{20})$. Equations 3–4 are used to generate RNNs output from the input.

$$H_t = W_z \times W_t + W_h \times H_{t-1} + b_t \tag{3}$$

$$Y_t = W_y \times H_t + b_y \tag{4}$$

Where, H_t, W_z, W_h and b_t are means t^{th} time hidden units output, embedding weights, hidden weights and bias value respectively. Y_t and W_y indicates the unit's output and activation weights. For each word, input and output are the same dimensions, and RNNs layer produced a flatted 20480 feature map. This feature map is feed to the first fully connected layers $FC1$ and followed by $FC2$ and $FC3$. The fully connected layer output is a 512 length feature vector that feeds to the softmax layer or output layer. This layer calculates the expected category name using Eq. 5

$$P(category = i|X) = \frac{(e^{X^T W_i})}{\sum_{j=1}^{C} e^{X^T W_j}} \tag{5}$$

Here, X is the i^{th} category feature value, and W_i indicates the softmax layer weights matrix. Total category (C) is equal to 12. The outcome of the softmax layer provides an expected value which used for calculating the deviation.

The deviation value used for backpropagation throws the backward direction. The whole process continues for several epochs and finally produces a web text classifier model.

3.3 Testing Module

Classifier model kernels value is initializing through the architecture. Web crawler crawled web text, and prepared a one-hot vector with size 2000, extracted feature for each word and generate a 2D feature matrix ($X_{2000\times200}$). This feature matrix projected with the classifier model and this model produced a score vector ($V_1, V_2, V_3...V_{12}$) where each value represents an expected category value. The Eq. 5 calculates the score value. The maximum value selected among these values for a particular category and the corresponding index will be the category name.

4 Experiments

The proposed system implemented by Python-based Keras deep learning library and ran in GPU $1070Ti$ based Ubuntu16.04 operating system. The system contains 32 GB physical memory and 8 GB GPU internal memory. The DCRNNs framework is tuned with various hyperparameters: regularization = 0.01, batch size = 64, training epochs = 70, feature extraction = GloVe, coefficient decay = 1.31, dropout = 0.47 and kernel initialized by Xavier function.

4.1 Web Text Corpus

Data acquisition is one of the toughest tasks for digitally resource constraint dialects like Bengali. The web texts are collecting from various online Bengali blogs and newspapers. GloVe embedding algorithm trained with $150K$ labelled and unlabeled data for embedding purposes. Table 1 depicts the summary of the web text embedding dataset.

Table 1. Statistics of the developed data set.

Attributes	Value
Number of web text documents	150000
Number of sentences	425301
Number of words	73407511
Number of unique words	1200000
Text to feature extraction method	GloVe
Number of context size	12
Feature dimension	200

The web crawler is crawled data from the web and annotated by the crowdsourcing technique. A group of 7 undergraduate students of computer science and engineering background who are doing their thesis on Bengali language processing annotated the data by hand in crowdsourcing phase. Another group consisting of ten participants rechecked all labelled data for validation. Three of them have a postgraduate degree in computer science and engineering, and seven have graduated degrees on Bengali literature. In the corpus, $120K$ labelled training data used for classifier model generation. Table 2 shows the category-sapient training and testing data summary. The maximum and the minimum number of sentences in the corpus are 90 and 2, respectively. The maximum length of the sentences is 15 words, and a minimum length of 3 words.

Table 2. Summary of training, testing and validation data sets.

Category name	#Training text	#Validation text	#Testing text
Accident (Acd)	10385	1000	3402
Art (At)	6589	700	2146
Crime (Cm)	10638	1100	3812
Economics (Em)	7842	700	2743
Education (Et)	6475	700	2865
Entertainment (Etm)	10874	1100	3934
Environment (Env)	4239	500	1510
Health (Ht)	4320	500	1280
Opinion (O)	11795	1100	3848
Politics (Plt)	26452	2500	4834
Science & Technology (Sct)	6422	600	1994
Sports (S)	13969	1500	3839
Total	120000	12000	36207

4.2 Evaluation Measures

Training/validation loss and accuracy are measured to evaluate the performance of Bengali web text categorization system. The performance of the unlabeled test dataset measured by precision (P), recall (R), F_1, and accuracy (A). In each training time, losses and accuracy are calculating by Eqs. 6–7.

$$L_i = -W * F_i^t + \sum_{j=1}^{C} e^{W_j * F_i^t} \tag{6}$$

$$A_i = \frac{P_i}{H_i} \tag{7}$$

Here, L_i and F_i refers to i^{th} time loss and feature vector respectively. The symbol W indicates the soft max layer weight matrix. A_i, P_i and H_i represents by i^{th} time accuracy, number of web text predicted correctly and total number of web text process at that time (t) respectively.

In testing phase precision, recall, accuracy and F_1-measure are calculated by Eqs. 8–11.

$$\textbf{Precision} = \frac{T_{pos}}{T_{pos} + F_{pos}} \tag{8}$$

$$\textbf{Recall} = \frac{T_{pos}}{T_{pos} + F_{neg}} \tag{9}$$

$$\textbf{Accuracy} = \frac{T_{pos} + T_{neg}}{T_{pos} + F_{pos} + T_{neg} + F_{neg}} \tag{10}$$

$$\textbf{F}_\textbf{1} - \textbf{measure} = \frac{2 * P * R}{R + P} \tag{11}$$

5 Results and Analysis

Our developed dataset evaluates the proposed web text categorization system. Figure 2 and Fig. 3 shows the training/validation loss and accuracy in respect

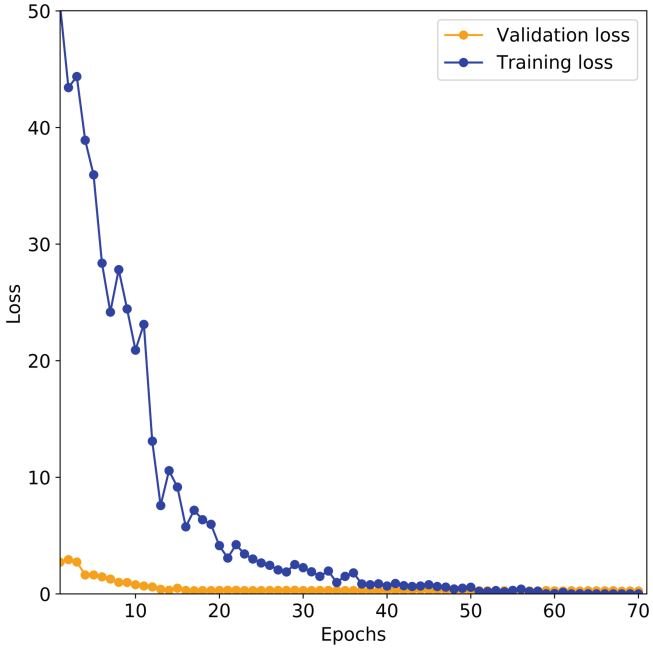

Fig. 2. Training and validation epochs vs loss

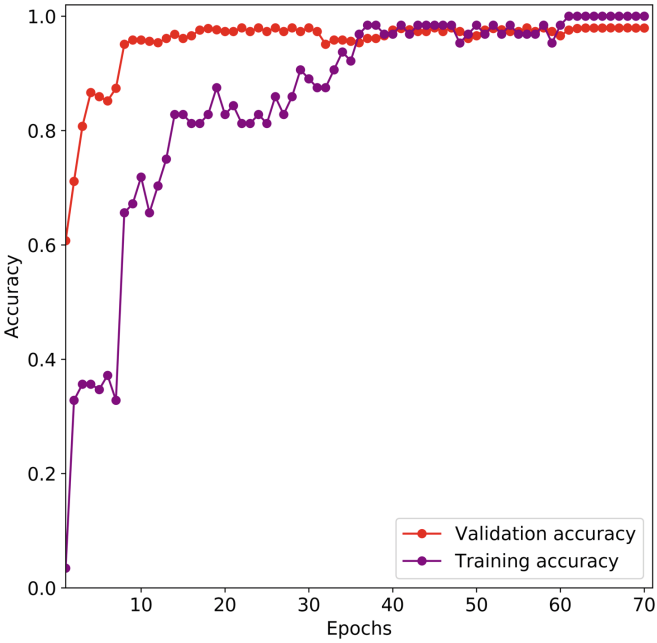

Fig. 3. Training and validation epochs vs accuracy

of a number of epochs. The logarithmic training loss commences with 50.0, and validation loss commences with 2.8 at the first epoch. The loss values gradually minimized from 2 to 30. Validation and training loss is stable after epoch 30 and maintains steady to 70 epochs. Training accuracy is amending up-to 40 epoch and after the session has no astronomically immense vicissitude in accuracy according to epoch number. Ergo, both the precision and loss are converged at epoch number 70.

Table 3 exhibits the precision, recall and F_1-measure of the web text categorization system. The result indicates that the utmost accuracy (99.00%) is attained for entertainment (Etm) category, while minimum accuracy (94.00%) found for economics (Em). The overall accuracy of the system degraded due to the intra-class overlapping during the data distribution.

5.1 Comparison with Previous Techniques

In order to evaluate the effectiveness of the proposed system, we compared it with the available techniques. Table 4 illustrates the number of training and testing data with categories. The proposed system is working with more train, and testing data compare to previous techniques and also has a more significant number of classes.

Table 5 shows the outcomes of a comparison between the suggested technique and the available techniques concerning accuracy. The comparison result

Table 3. Summary of statistical analysis of web text categorization

Category name	Precision	Recall	F_1-score	support
Health (Ht)	0.99	0.97	0.97	1280
Accident (Acd)	0.96	0.92	0.94	3402
Art (At)	0.97	0.93	0.94	2146
Crime (Cm)	0.99	0.96	0.97	3812
Economics (Em)	0.93	0.94	0.94	2743
Education (Et)	0.98	0.93	0.95	2865
Entertainment (Etm)	0.99	0.99	0.99	3934
Environment (Env)	0.95	0.94	0.94	1510
Opinion (O)	0.96	0.96	0.96	3848
Politics (Plt)	0.96	0.98	0.97	4834
Science & Technology (Sct)	0.94	0.98	0.96	1994
Sports (S)	0.96	0.98	0.97	3839
Avg./total	0.956	0.957	0.958	36207

Table 4. Comparison regarding the number of classes and size of the dataset.

Techniques	No. of train data	No. of test data	No. of category
SVM and TF-IDF [10]	1000	118	5
Word2Vec and K-NN [14]	19750	4713	7
DCNNs and Word2Vec [8]	86199	10707	12
SGD and Word2Vec [9]	10000	4651	9
Proposed (DCRNNs)	**120000**	**36207**	**12**

indicates that the proposed web text categorization system outperformed the existing techniques in terms of higher accuracy and a more significant number of document categories.

The accuracy of Mandal et al. [10] and Ahmed et al. [14] techniques are lower due to the limitations of their feature extraction techniques. Hossain et al. [8] developed a Word2Vec and DCNNs based system with 94.96% accuracy, which is lower than the proposed system (95.89%). Dhar et al. [2] developed a

Table 5. Performance comparison.

Techniques	Accuracy (%)
SVM and TF-IDF [10]	89.14
Word2Vec and K-NN [14]	91.02
DCNNs and Word2Vec [8]	94.96
SGD and Word2Vec [9]	93.00
Proposed (DCRNNs)	**95.89**

web text categorization system with 97.09% accuracy, which is a bit higher than the proposed technique but they considered a fewer number of categories (8). Moreover, this technique is not capable of handling wide-scale datasets.

6 Conclusion

Web text categorization is a popular research agenda for highly resource languages. It is comparatively a challenging task to develop an automatic text categorization system for resource-constrained languages like Bengali. This paper presents a DCRNNs based text categorization system to classify Bengali web text contents. A corpus consisting of 150000 web text documents in 12 categories developed to evaluate the system which is larger than the existing techniques. The proposed system achieved 95.89% accuracy on the developed corpus. The accuracy of the current implementation can be improved by including more labelled data and categories. RNNs may be superseded by long short term recollection (LSTM) for ameliorating the overall accuracy of the system. These are kept for future research issues.

Acknowledgement. This work was supported by the Establishment of CUET IT Business Incubator Project, BHTPA, ICT Division, Bangladesh under the research project, "Automatic Bengali Document Categorization based on Summarization Techniques".

References

1. Dhar, A., Dash, N.S., Roy, K.: Classification of Bangla text documents based on inverse class frequency. In: 3rd International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), Bhimtal, India, 05 November, pp. 1–6 (2018)
2. Dhar, A., Dash, N.S., Roy, K.: Categorization of Bangla web text documents based on TF-IDF-ICF text analysis. In: Social Transformation - Digital Way CSI, vol. 836. Springer, Singapore (2018)
3. Stein, R.A., Jaques, P.A., Valiati, J.F.: An analysis of hierarchical text classification using word embeddings. Journal of CoRR, vol. abs/1809.01771 (2018)
4. Liu, H., Burnap, P., Alorainy, W., Williams, M.L.: A fuzzy approach to text classification with two-stage training for ambiguous instances. IEEE Trans. Comput. Soc. Syst. **6**, 227–240 (2019)
5. Cotterell, R., Schütze, H.: Morphological Word Embeddings. American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp.1287–1292 (2019)
6. Salama, R.A., Youssef, A., Fahmy, A.: Morphological word embedding for Arabic. In: The 4th International Conference on Arabic Computational Linguistics (ACLing), Dubai, UAE, vol. 142, pp. 83–93 (2018)
7. Wang, D., Gong, J., Song, Y.: W-RNN: news text classification based on a weighted RNN. Journal of CoRR, vol. abs/1909.13077, 28 September 2019

8. Hossain, M.R., Hoque, M.M.: Automatic Bengali document categorization based on deep convolution nets. In: Emerging Research in Computing, Information, Communication and Applications. Advances in Intelligent Systems and Computing, vol. 882. Springer, Singapore (2019)
9. Hossain, M.R., Hoque, M.M.: Automatic Bengali document categorization based on word embedding and statistical learning approaches. In: International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2) (2018)
10. Mandal, A.K., Sen, R.: Supervised learning methods for Bangla web document categorization. Int. J. Artif. Intell. Appl. (IJAIA) **5**(5), 93–105 (2014)
11. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. Journal of CoRR (2013)
12. Kabir, F., Siddique, S., Kotwal, M., Huda, M.: Bangla text document categorization using stochastic gradient descent (SGD) classifier. In: Proceedings of the International Conference on Cognitive Computing and Information Processing, pp. 1–4 (2015)
13. Liebeskind, C., Kotlerman, L., Dagan, I.: Text categorization from category name in an industry motivated scenario. J. Lang. Resour. Eval. **49**(2), 227–261 (2015)
14. Ahmad, A., Amin, M.R.: Bengali word embeddings and its application in solving document classification problem. In: 19th International Conference on Computer and Information Technology, pp. 425–430 (2016)
15. Johnson, R., Zhang, T.: Deep pyramid convolutional neural networks for text categorization. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL), vol. 01, pp. 562–570 (2017)
16. Pennington, J., Socher, R., Manning, C.D.: GloVe: global vectors for word representation. In: Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (2014)
17. Zhang, X., Zhao, J., LeCun, Y.: Character-level convolution networks for text classification. In: 28th International Conference on Neural Information Processing Systems, NIPS 2015, vol. 01, pp. 649–657 (2015)
18. Xu, K., Feng, Y., Huang, S., Zhao, D.: Semantic relation classification via convolutional neural networks with simple negative sampling. In: Empirical Methods in Natural Language Processing, Lisbon, Portugal, pp. 536–540 (2015)
19. Bahassine, S., Madani, A., Kissi, M.: Arabic text classification using new stemmer for feature selection and decision trees. J. Eng. Sci. Technol. **12**, 1475–1487 (2017)
20. Maas, A.L., Hannun, A.Y., Ng, A.Y.: Rectifier nonlinearities improve neural network acoustic models. In: Proceedings of the ICML (2013)
21. Al-Taani, A.T., Al-Sayadi, S.H.: Classification of Arabic text using singular value decomposition and fuzzy c-means algorithms. In: Johri, P., Verma, J., Paul, S. (eds.) Applications of Machine Learning. Algorithms for Intelligent Systems, 05 May, pp. 111–123. Springer, Singapore (2020)
22. Bhagat, C., Mane, D.: Text categorization using sentiment analysis. In: Bhalla, S., Kwan, P., Bedekar, M., Phalnikar, R., Sirsikar, S. (eds.) Proceeding of International Conference on Computational Science and Applications. Algorithms for Intelligent Systems, 05 January, pp. 361–368S. Springer, Singapore (2020)