BARD: Bangla Article Classification Using a New Comprehensive Dataset

Conference Paper · August 2018

DOI: 10.1109/ICBSLP.2018.8554382

2 authors, including:

Md Mofijul Islam
University of Dhaka
21 PUBLICATIONS **132** CITATIONS

SEE PROFILE

BARD: Bangla Article Classification using a New Comprehensive Dataset

Md Tanvir Alam, Md Mofijul Islam
Department of Computer Science and Engineering, University of Dhaka, Dhaka, Bangladesh.
tanvircsedu15@gmail.com, akash.cse.du@gmail.com

***Abstract*—In the literature, automated Bangla article classification has been studied, where several supervised learning models have been proposed by utilizing a large textual data corpus. Despite several comprehensive textual datasets are available for different languages, a few small datasets are curated on Bangla language. As a result, a few works address Bangla document classification problem, and due to the lack of enough training data, these approaches could not able to learn sophisticated supervised learning model. In this work, we curated a large dataset of Bangla articles from different news portals, which contains around 3,76,226 articles. This huge diverse dataset helps us to train several supervised learning models by utilizing a set of sophisticated textual features, such as word embeddings,TF-IDF. In this works, our learning model shows promising performance on our curated dataset, compared to state-of-the-art works in Bangla article classification.**

Furthermore, we deployed our proposed Bangla content classifier as a web application: bard2018.pythonanywhere.com and the video demo of this application is available here: bit.ly/BARD_VIDEO_DEMO. Additionally, we open-sourced the BARD dataset(bit.ly/BARD_DATASET) and source code of this work(bit.ly/BARD_SC).

***Index Terms*—Document Classification, Machine Learning, Bangla Article Dataset**

I. Introduction

The proliferation of unstructured textual data attracts the natural language research community to extract knowledge by mining textual data. The most common and well-studied problem is to categorize the textual documents. Document classification has paramount importance on several applications like searching, filtering, and organizing the textual documents.

During the last decades, several statistical and machine learning approaches have been utilized to extract meaningful features to accurately classify the textual documents. For example, Bag of words, TF-IDF [13], Word embedding (Word2Vec [11]) features have been extracted from the text data in order to train some supervised learning model, for instance, SVM, KNN or Naive Bayes, for classifying the documents. Moreover, various deep learning algorithms, such as Convolutional Neural Network(CNN) and Long Short Term Memory(LSTM) [5], have also been utilized to extract valuable features from the unstructured textual data in order to categorize the documents. Thus, a considerable amount of textual data is required to train these supervised learning models.

Despite several large dataset datasets on different other languages are available, a few datasets are constructed for the Bangla document classification purpose. For this reason, only a couple of works addressed the Bangla text categorization task. However, all of this works utilized small datasets with around a thousand of articles, which is not adequate to train a supervised learning model. Furthermore, most of the state-of-the-art works, which considered the Bangla article classification task, only consider Bag of words or TF-IDF features and thus did not consider the semantic features based techniques like Word2Vec.

In this work, we address the above-mentioned problems and developed a large corpus of Bangla documents, BARD: Bangla Article Dataset. Additionally, we also perform an extensive statistical analysis to extract the textual relationships across the different documents categories. Furthermore, we utilized several textual features, such as TF-IDF and Word2Vec, in order compare the performance of various supervised learning approaches, for instance, Logistic Regression, Neural Network, Naive Bayes, and ensemble approaches.

The major contributions of this work are summarized below:

- To the best of our knowledge, we curated the largest Bangla articles dataset, which consists of around 3,76,226 articles labeled with five document categories.
- We performed an extensive statistical analysis in order to identify the feasibility of the word based features usage in training the supervised learning model.
- We conduct several experiments to compare the performance of different supervised learning models by employing various textual features, specifically TF-IDF and Word2Vec.
- We also developed a web application which helps to classify Bangla text document and provides the statistical analysis of an article.

The remaining parts of the paper are organized as follows: Related Works have been presented in Section II. After that, In Section III, we discuss the details attributes of BARD dataset as well as the textual statistical analysis of BARD dataset articles. Subsequently, the proposed Bangla article classification model and its web application are presented in Section IV. Then, in Section V, we discussed the performance of different supervised learning models and the textual features

978-1-5386-8207-4/18/$31.00

in classifying Bangla articles. Finally, we conclude this work with future directions in Section VI.

II. Related Works

Automated classification of text documents in various languages is a well-studied area. Several supervised learning algorithms have been used to classify text documents, for example, in [6] the author utilizes Support Vector Machine to learn the textual features for document classification. Moreover, Naive Bayes [12] and [2] and KNN [14] have been employed in the state-of-the-art works. In addition, word2vec embedding with SVM is utilized [8] for categorizing English documents. Furthermore, deep learning models, such as CNN [7], have also been introduced in the literature for this purpose.

Compared to other languages, a very few works have been addressed done for Bangla article classification task. In [10], author analyzes the efficiency of n-gram based text categorization for Bangla language with one=year news corpus of Prothom Alo newspaper. Moreover, Naive Bayes based bangla text classification model has been proposed in [3]. In addition, in [9], author compares the performance of four supervised learning Methods: Decision Tree, K-Nearest Neighbor, Naive Bayes, and Support Vector Machine (SVM) for Bangla documents categorization. In this work, TF-IDF have been used to construct feature vector by utilizing 1000 web documents. Similarly, in [4], author uses TF-IDF to train a supervised learning model on 1960 web documents.

To the best of our knowledge, all the state-of-art works, who addressed the Bangla text classification task, utilized a very small corpus of articles which is not very effective to train a supervised learning model. Furthermore, word2vec based semantic textual features have not been studied to build a supervised model for Bangla article categorization task.

III. Dataset and Textual Statistical Analysis

In this section, we present the details of the proposed dataset BARD: Bangla Article Dataset. Subsequently, an extensive textual statistical analysis of the BARD dataset article has been presented.

Category	No. of Documents	No. of Words	Average Sentences per Document	Average words per Sentence
State	242860	57019465	18.50	13.356
Economy	18982	4915141	20.18	13.378
International	32203	7096111	18.47	12.493
Entertainment	31293	6706563	21.70	10.236
Sports	50888	12397415	22.80	11.069

TABLE I. Text Corpus Details

A. BARD: Bangla Article Dataset

We fetched Bangla articles from various Bangla online news portals using Google Search API. We only considered the news articles which belong to five predefined classes: State, International, Economy, Entertainment, Sports. The articles were labeled with its class at the time of fetching. After applying different filtering approaches to remove the duplicate articles as well as irrelevant articles we gathered 3,76,226 Bangla news articles. The distribution of 3,76,226 articles on five categories is shown in Table I, along with the total number of words in each category. Except the State category the distribution of articles on different are quite balance. However, in the statistical analysis, presented in SectionIII-B, we have found out that the State category's articles have the almost unique textual property which can help the supervised learning model to accurately separate this articles from the other category. Additionally, the average number of sentence per document and the average number of words per sentence in each category are quite consistent, which are depicted in Table I.

B. Statistical Analysis

We performed textual statistical analysis on the BARD dataset articles and results are presented in Fig. 1 and 2. The frequency distributions of the top 20 most frequent words for each of the five categories are depicted in Fig. 1. From this analysis, we can easily identify that all the categories have almost similar most frequent words. In other words, these most frequent words do not help to categorize the articles. Hence, we removed around 25 most frequent words from all the articles and performed the statistical analysis again on the filtered dataset, which is presented in Fig. 2. Now, this filtered frequency distribution shows that each categories have some unique distribution of words, which may contribute to categorize the articles.

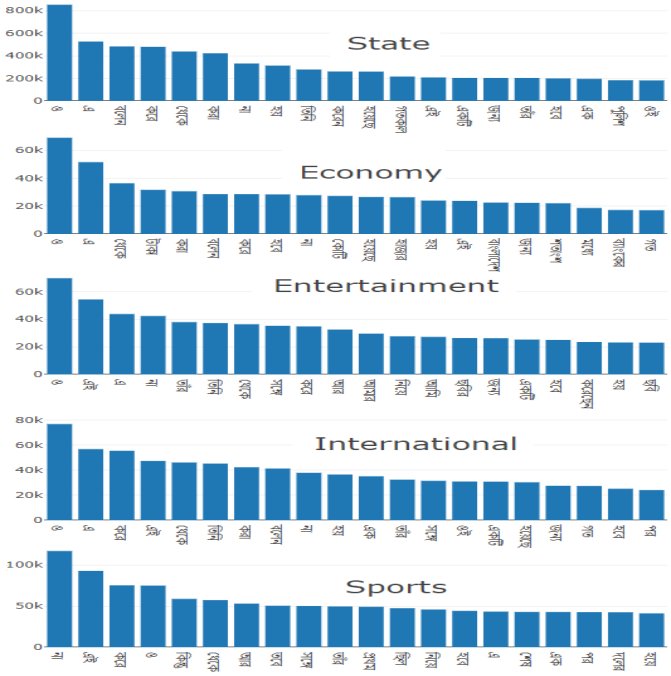

Fig. 1: Most frequent words in each categories.

IV. Proposed Model

We developed a Bangla article classifier utilizing the BARD dataset. Also, TF-IDF [13] and Word2Vec [11] have been utilized to train the supervised learning models for article classification. The Bangla article classification model is depicted in Fig. 3, which has two separate phase: prediction and training phase. Before utilizing the articles in the BARD dataset, our model first conducted some preprocessing on the articles contents, such tokenizing, removing stop words and construction of feature vector for the whole corpus using both TF-IDF [13] and Word2Vec [11] approaches. In the training phase, we first divided the processed dataset into training and test set. The details of preprocessing and feature extraction is described in the following subsections. After building the feature vector, we employed the train dataset to train models with various supervised classification algorithms. Finally we compared the performances of the classifiers on test set and choose the best classifier with best feature to perform prediction task. The different phases of our proposed model is depicted in Fig. 3.

A. Preprocessing

Before tokenizing, we removed all the punctuations and digits(1,2,3...). We tokenized the articles using space as delimiter. At the end of this step, we got a set of words and their frequency in each article.

B. Feature Extraction and Classifier Training

After completing the preprocessing steps, we extracted the feature vector using TF-IDF and Word2Vec approaches.

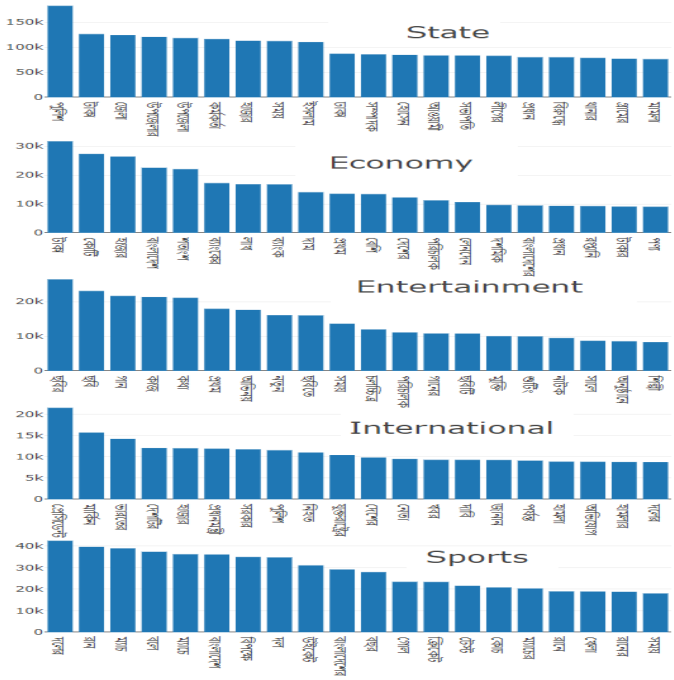

Fig. 2: Most frequent words after removing stop words.

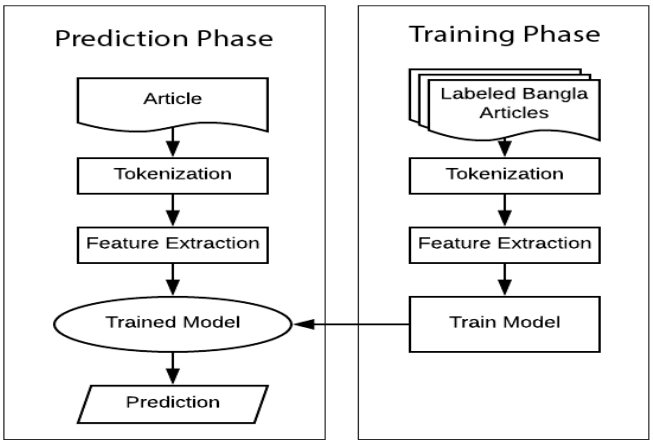

Fig. 3: Overview of the model.

Subsequently, we trained several supervised classifiers, which is presented in details in the subsequent sections.

1) Word2Vec Feature: Word2Vec [11] is a technique used to attain vector representation of words that preserves both semantic and syntactic relationships.

Here for each unique word in a document, we multiplied the corresponding vector with its frequency in the document. Then we summed up the vectors. We used word2vec trained by facebook research [1]. Each vector has a dimension of 300. Let document d contain words W_1 ,W_2 ... W_i ... W_n with multiplicity C_1,C_2 ... C_i ... C_n and Word2Vec representation of W_i be W2V(C_i). Then the feature vector for document d is,

$$V_d = \sum_{i=1}^{n} (W2V(W_i) * C_i)$$

2) TF-IDF Feature: TF-IDF [13] is a widely used technique in Information Retrival. It is a technique for assigning scores to words in a document. TF-IDF score for a word w in a document D in a corpus can be calculated as below:

$$TF = \frac{\text{Frequency of w in D}}{\text{Total number of words D}}$$

$$IDF = \log_e \frac{\text{Total number of documents}}{\text{Number of documents containing w}}$$

$$TF - IDF = TF * IDF$$

In TF-IDF approach, we considered the most frequent words for feature vector. For each document, we calculated the TF-IDF scores for the words under consideration only. From these scores we built the feature vector. We considered both feature vector of size 300 and 3000 words for comparison.

After extracting these features we trained several supervised learning models by utilizing these features. Then we chose the best model with a feature, either TF-IDF or Word2Vec, which performed better than any other models. Using this best model we conducted the prediction in our web application.

C. Bangla Article Classifier Web Application Implementation

We have built a web application for Bangla text classification utilizing the proposed model, which is depicted in Fig. 4. The application named BARD - Bangla Article Classifier can be accessed from http://bard2018.pythonanywhere.com.

1) User Interface: The user interface of the application is very simple, which contains only two pages. In the first page, the user can input the content they want to classify. From this page, they will be redirected to the second page, which presents the result of the classification.

In the result page, the user will be able to see the category of the document and word statistics as shown in Figure 4. Among other statistics, this page shows the probability for different other categories, word frequency table and a bar chart of frequency distribution of the top 20 most frequent words in that text content.

2) Backend: In the experiment evaluation, we found that using word2vec features with neural net is the best prediction model. We used that trained model for the web application. The process of predicting classes is almost identical to the prediction phase of Figure 3.

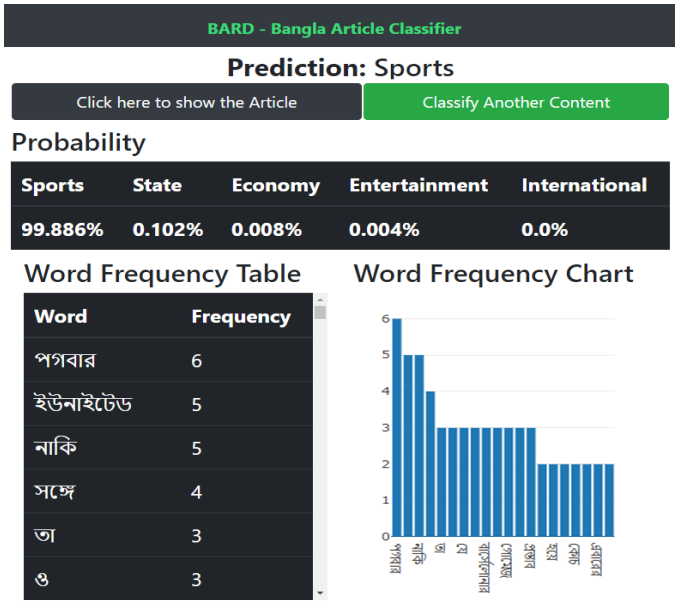

Fig. 4: Web Application of BARD

V. Experimental Result

We have used 10 fold cross validation for our experimental evaluation. We trained five supervised classification models: Logistic Regression, Neural Network, Naive Bayes, Random Forest and Adaboost. Moreover, we also assessed the performance of each supervised learning model using three textual features: word2vec, TF-IDF(3000 word vector), and TF-IDF(300 word vector).

To asses the performance of the prediction model we utilized Precision, Recall and F1 score as evaluation metrics which are defined as

Features	Learning Model	Precision	Recall	F1-score
Word2Vec	Logistic Regression	0.95	0.95	0.95
	Neural Network	0.96	0.96	0.96
	Naive Bayes	0.70	0.71	0.74
	Random Forest	0.91	0.91	0.90
	Adaboost	0.92	0.92	0.92
TF-IDF*	Logistic Regression	0.94	0.94	0.94
	Neural Network	0.96	0.96	0.96
	Naive Bayes	0.87	0.87	0.88
	Random Forest	0.93	0.93	0.92
	Adaboost	0.89	0.89	0.88
TF-IDF**	Logistic Regression	0.85	0.85	0.83
	Neural Network	0.92	0.91	0.91
	Naive Bayes	0.75	0.75	0.78
	Random Forest	0.87	0.87	0.85
	Adaboost	0.82	0.82	0.80

* TF-IDF feature with 3000 word vector size.
** TF-IDF feature with 300 word vector size.

TABLE II. Performance comparison of different supervised learning model and textual features(Word2Vec and TF-IDF)

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1 - score = \frac{2 * (Recall * Precision)}{Recall + Precision}$$

Here, TP = True Positives , FP = False Positives and FN = False Negatives.

The performance measures of the models for different textual features are presented in Table II.

A. Comparison Between Different Learning Algorithms

In the experiment evaluation result, which is depicted in Tale II, we can easily identify that neural network with word2vec is superior to other supervised classification models. This is the indication that we can utilize the deep neural network learning model to improve the article classification performance. Furthermore, Logistic Regression also performed better, specially with the word2vec features.

B. Comparison Between Word2Vec and TF-IDF Features

Among the three approaches, word2vec feature in neural network is providing the best performance having a precision of 0.96. Moreover, TF-IDF with 3000 features gave almost the same performance with the same precision and other metrics. However, if we compare the performance in different folds then neural network with word2vec performed better than the neural network with TF-IDF of 3000 word features. The reason behind is that word2vec can capture the semantic and syntactical features of the words in the text. On the other hand, if we compare the approaches for same size feature vector, we can see that TF-IDF with 300 features(same as Word2vec approach) has a precision of 0.92 only.

Features	Learning Model	Precision	Recall	F1-score
Word2Vec	Logistic Regression	0.95	0.95	0.95
	Neural Network	0.96	0.96	0.96
TF-IDF*	Logistic Regression	0.94	0.94	0.94
	Neural Network	0.96	0.96	0.96
TF-IDF [9]	SVM	0.89	0.89	0.89
TF-IDF [4]	LIBLINEAR	0.93	-	-

* TF-IDF feature with 3000 word vector size.

TABLE III. Performance comparison of different state-of-the-art-works

C. Performance Comparison of State-of-the-art Models

In Table III, we compared the performance of our model with other state-of-the-art works. We choose logistic regression and neural network with both Word2Vec and TF-IDF(3000 features) from our trained models to perform the comparison, as these four models showed better performances in the experiments evaluation. We compare our models with TF-IDF based SVM(Support Vector Machine) Bangla text classification model [9]. This work utilize a dataset of 1000 web documents of five classes. The average performance measures of the model for five classes is shown on III. Moreover, in [4], author used 1960 bangla web documents of five classes. Among the models they introduced, LIBLINEAR performed the best and the average precision is 93%.

From Table III, we can clearly observe that all of our four classification models have outperformed the state-of-the-art Bangla article classification models.

VI. Conclusion

In this work, we curated the largest Bangla article dataset and performed extensive experimentation to compare the performance of different supervised learning model using word2vec and TF-IDF features. In the experimentation, we found out that neural network with word2vec features, having relatively low dimensional feature vector, performed better than the other supervised models and textual features. Additionally, we deployed our prediction model as a web application to classify Bangla articles.

We expect that this dataset will help the Bangla natural language research community to further extend the article classification task. Moreover, this dataset can be utilized in other Bangla NLP(natural language processing) tasks. In future, we have a plan to extend this dataset so that it can be used to solve other Bangla NLP related problems. Furthermore, Deep learning model, such as CNN, LSTM, can also be considered to improve the prediction model.

ACKNOWLEDGEMENTS

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

References

[1] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," CoRR, vol. abs/1607.04606, 2016.

[2] J. Chen, H. Huang, S. Tian, and Y. Qu, "Feature selection for text classification with naïve bayes," Expert Syst. Appl., vol. 36, no. 3, pp. 5432–5435, Apr. 2009.

[3] A. N. Chy, M. H. Seddiqui, and S. Das, "Bangla news classification using naive bayes classifier," in 16th Int'l Conf. Computer and Information Technology, March 2014, pp. 366–371.

[4] A. Dhar, N. S. Dash, and K. Roy, "Application of tf-idf feature for categorizing documents of online bangla web text corpus," in Intelligent Engineering Informatics. Singapore: Springer Singapore, 2018, pp. 51–59.

[5] S. Hochreiter and J. Schmidhuber, "Long short-term memory," Neural Comput., vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

[6] T. Joachims, "Text categorization with support vector machines: Learning with many relevant features," in Proceedings of the 10th European Conference on Machine Learning, ser. ECML'98. Berlin, Heidelberg: Springer-Verlag, 1998, pp. 137–142.

[7] Y. Kim, "Convolutional neural networks for sentence classification," in Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, 2014, pp. 1746–1751.

[8] J. Lilleberg, Y. Zhu, and Y. Zhang, "Support vector machines and word2vec for text classification with semantic features." in ICCI*CC, N. Ge, J. Lu, Y. Wang, N. Howard, P. Chen, X. Tao, B. Zhang, and L. A. Zadeh, Eds. IEEE Computer Society, 2015, pp. 136–140.

[9] A. K. Mandal and R. Sen, "Supervised learning methods for bangla web document categorization," CoRR, vol. abs/1410.2045, 2014.

[10] U. N. K. M. Mansur, M., "Analysis of n-gram based text categorization for bangla in a newspaper corpus." in Proceedings of International Conference on Computer and Information Technology, 2006.

[11] 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 26, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 3111–3119.

[12] I. Rish, "An empirical study of the naive bayes classifier," Tech. Rep., 2001.

[13] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," Inf. Process. Manage., vol. 24, no. 5, pp. 513–523, Aug. 1988.

[14] V. Tam, A. Santoso, and R. Setiono, "A comparative study of centroid-based, neighborhood-based and statistical approaches for effective document categorization," in ICPR, 2002.