Bangla Content Categorization Using Text Based Supervised Learning Methods

Sadek Al Mostakim
Student Id: 011141046
Faiza Ehsan
Student Id: 011141043
Syeda Mahdiea Hasan
Student Id: 011141056

A thesisin the DepartmentofComputer Science and Engineering presented
in partial fulfillment of the requirements for the Degree of
Bachelor of Science in Computer Science and Engineering

United International University
Dhaka, Bangladesh
October, 2018

Declaration

We, [Sadek Al Mostakim, Faiza Ehsan and Syeda Mahdiea Hasan], declare that this thesis titled, Bangla Content Categorization Using Text Based Supervised Learning Methods and the work presented in it are our own. We confirm that:

- This work was done wholly or mainly while in candidature for a [BSc] degree at United International University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at United International University or any other institution, this has been clearly stated.
- Where we have consulted the published work of others, this is always clearly attributed.
- Where we have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- We have acknowledged all main sources of help.
- Where the thesis is based on work done by our self jointly with others, we have made clear exactly what was done by others and what we have contributed our self.

Name: Sadek Al Mostakim, Id: 011141046, Department: CSE

Name: Faiza Ehsan, Id: 011141043, Department: CSE

Name: Syeda Mahdiea Hasan, Id: 011141056, Department: CSE

Certificate

I do hereby declare that the research works embodied in this thesis entitled "**Bangla Content Categorization Using Text Based Supervised Learning Methods**" is the outcome of an original work carried out by Sadek Al Mostakim, Faiza Ehsan and Syeda Mahdiea Hasan under my supervision.

I further certify that the dissertation meets the requirements and the standard for the degree of [BSc] in Computer Science and Engineering.

_____

[Name and designation of Supervisor]

Abstract

The widespread and increasing availability of text documents in electronic form increases the importance of using automatic methods to analyze the content of text documents. Specifically, there is a great development in Bangla content generation due to greater number of users in the recent years in social media. In this paper, we present a supervised learning based Bangla content classification method. We have created a large Bangla content dataset and made it available for use publicly. This dataset was tested using several machine learning algorithms using text based features. Our experiments showed logistic regression worked best compared to other algorithms. We have developed an online tool based on our method and made it available for content categorization at: http://samspark1-001site1.etempurl.com/. We have also made our data extraction tool and the dataset available for use of the other researchers

From: https://github.com/sspaarkk/BanglaNLP.

Acknowledgement

We are very glad to announce that this paper was accepted by International Conference on Bangla Speech and Language Processing (ICBSLP) 218. First of all, we are extremely thankful to our Almighty Allah. All our work was successful only because of the constant support and guidance from our respected supervisor Dr Swakkhar Shatabda. We would also like to thank our families for encouraging us to work diligently.

Table of Content

LIST OF TABLES

LIST OF FIGURES

Chapter1

1 Introduction

1.1 Importance of Bangla Text Categorization

Text categorization is an active research area of text mining where the documents are classified with supervised, unsupervised or semi-supervised knowledge [1]. Traditionally, this task is solved manually, but such manual classification is expensive to scale and also labor intensive. Besides, Sentiment analysis or opinion mining has been quite popular and has led to building of better products, understanding user opinion, executing and managing of business decisions [2].

Bangla is the sixth-most popular language in the world and spoken by a population that now exceeds 250 million. It is the primary language in Bangladesh and second mostly spoken language in India [3], [4]. However, its spread is accelerating with the massive increased usage of social media, where they can share their views and opinions regarding any topic of interests. This results in huge volumes of user-generated information on the micro blogging sites, which are utilized for many applications. This information later come in handy for product review mining where companies analyze the reviews provided by the consumers and decide which product should be improved and take decision regarding product sales. Vice versa the consumer goes through the reviews of previous other consumers and decides what to or not to buy. For completion of this entire procedure, millions of reviews need to be analyzed. This is where text mining makes the work easier.

1.2 Popular Methods

Among various machine learning approaches in document categorization, most popular is supervised learning where underlying input-output relation is learned by small number of training data and then output values for unseen input points are predicted. Various numbers of supervised learning techniques, such as Neural Network [5], K-Nearest

Neighbor [6], Decision Tree [7], Nave Bayes [8], Support Vector Machine [9], and N-grams [10] has been used for text document categorization.

Although text categorization is well studied in other languages for a long time, there are only recent advances in Bangla. Among a few works in Bangla document categorizations are: N-gram techniques [11], Naive Bayes Classifier [12] and Stochastic Gradient Descent [13], etc. However, we have observed that most the work in the literature lack annotated corpora. Most of the methods are not comparable to each other since they used different datasets and do not share them publicly. To add, the size of the datasets were also not large enough. Moreover, there methods are also not available for use later, and comparison becomes quite impossible. Our work is motivated from these observations.

1.3 Our Objectives

In this paper we intend to categorize Bangla documents. We extracted articles from the top news article provider in Bangla for a period of three months and created a large dataset. Several supervised machine learning techniques were used to classify these articles using text based features. Among all the classifiers tested, logistic regression was superior to others. We have also developed a web application based on our method. We have made our data extraction tool and the datasets available for use by the other researchers.

The main contribution of this paper is enumerated in the following:

1) Creating a large Bangla document dataset publicly available.

2) A publicly available tool for extracting Bangla articles from news provider websites.

3) A classification method for classification of Bangla documents.

4) A publicly available Tool for Bangla content categorization.

Rest of the paper is organized as follows: Section II briefly describes the related work in the literature; Section III presents our methods; Section IV presents experimental results and Section V concludes the paper with a summary and a direction for future work.

Chapter 2

2 Background and Literature Review

2.1 Classification Algorithms

Most frequent techniques used for text categorization are mainly K-Nearest Neighbor (KNN), Naïve Bayesian Classifier (NB), Decision Tree (DT), Neural Network (NNet) and Support Vector Machines (SVM). We also have tried some old and new algorithms. We have used 6 algorithms for classification and later compared which classification technique provides the best result. Below a brief description about the 6 algorithms are given.

2.1.1 Algorithm 1: K-Nearest Neighbor (KNN):

KNN algorithm is one of the simplest classification algorithms. It also called lazy learning algorithm. Its objective is to use separate the data points from the database into several classes to predict the class value of a new instance.

It is basically called a lazy learner because KNN does not use the training data sets for any generalization. KNN finds the similar elements or data points. We classify a givenfeature depending on how closely the feature resembles with the training set.

2.1.2 Algorithm 2: Gaussian Naïve Bayes

Because of the assumption of the normal distribution, Gaussian Naive Bayes is best used in cases when all our features are continuous.

Before diving straight to Gaussian NB, we must first take a look at the Naive Bayes probabilistic model.

Mathematically, give a dataset $x = (x1, \dots, xn)$to be classified, NB assigns to an example (dataset feature) a discrete probability,

$\mathbf{P(C_k|X_1 \dots . X_n)}$

For K-classes in the dataset. To learn this multivariate distribution would require a large amount of data. Thus, to simplify the task of learning, we assume that the features are conditionally independent from each other given the class. Consequently leading to the use of Bayes' theorem,

$$\mathbf{P(C_k|x) = \frac{P(C_k)\ P(x|C_k)}{P(x)}}$$

Translating to plain English, the above equation may be understood by

$$\mathbf{posterior = \frac{prior * liklihood}{evidence}}$$

=

By conditional probability, the numerator is just the *joint probability distribution* $\boldsymbol{p(Ck, x)}$, and may be factored through chain rule,

$$\boldsymbol{p(Ck, x) = p(x1|x2, \dots, xn, Ck)p(x2|x3, \dots, xn, Ck)p(xn|Ck)p(Ck)}$$

Now, through the assumption of conditional independence of features, i.e. each feature xi is conditionally independent from every other feature xj for $j \neq i$, we get

$$\boldsymbol{p(xi|xi + 1, \dots, xn, Ck) = p(xi|Ck)}$$

Leading us to the expression of the joint probability model $\boldsymbol{p(Ck, x)}$ as,

$$\boldsymbol{p(x1|Ck)p(x2|Ck) \dots p(xn|Ck) = p(Ck)\prod i = 0np(xi|Ck)}$$

When the data at hand is continuous data, the assumption is that the continuous values for each class are distributed according to a Gaussian distribution. Recall that the probability density function of the normal (Gaussian) distribution is given by

$$\mathbf{f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)}$$

Where $2\sigma^2$ represents the variance of the values in x, while μ represents the mean of the values in x.

So, for Gaussian NB, suppose we have a training data which consists of continuous attribute x, we shall segment the data by class. Then, we compute the mean μ and the variance $2\sigma_k{}^2$ of x per class. We let μk be the mean of the values in x for class Ck, then it follows that we let $\sigma_k{}^2$ be the variance of the values of x for class ck.

Now, assume we have collected some observation values xi. Thus, we have the probability density for xi for class Ck as $p(x = xi|Ck)$. We plug xi to the Gaussian distribution equation with parameters μk and $\sigma_k{}^2$,

$$p(x = xi|Ck) = \frac{1}{\sqrt{2\pi\sigma_k{}^2}} exp\left(\frac{-(x-\mu)^2}{2\sigma_k{}^2}\right)$$

2.1.3 Algorithm 3: Support Vector Machine (SVM)

Support Vector Machines are supervised learning models that evaluate data and distinguish patterns, use for classification and regression analysis. The objective of SVM algorithm is to draw a hyperplane in a multidimensional space that distinctly classifies the data points. If a labeled training dataset is given, the algorithm will output an optimal hyperplane which will categorize the new examples. In two dimensional spaces this hyperplane is a line dividing a plane in two parts where each class will lay in each side by the plane.

We have used two SVM models to train our dataset. These are SVM with linear kernel and SVM with rbf kernel.

SVM with Linear Kernel:

The main focus of support vector machines is to draw a hyperplane. It draws the hyperplane by measuring the distance of two vectors from two different classes. A hyperplane is called the margin of the classes.

In linear SVM there can be many hyperplanes. But the main target is to maximize the distance between the hyperplanes considering the data nearest to the plane from two different classes. From the above observation the hyperplane is chosen for separating the classes. The data that helps to maximize the distance of the hyperplane is called support vectors.

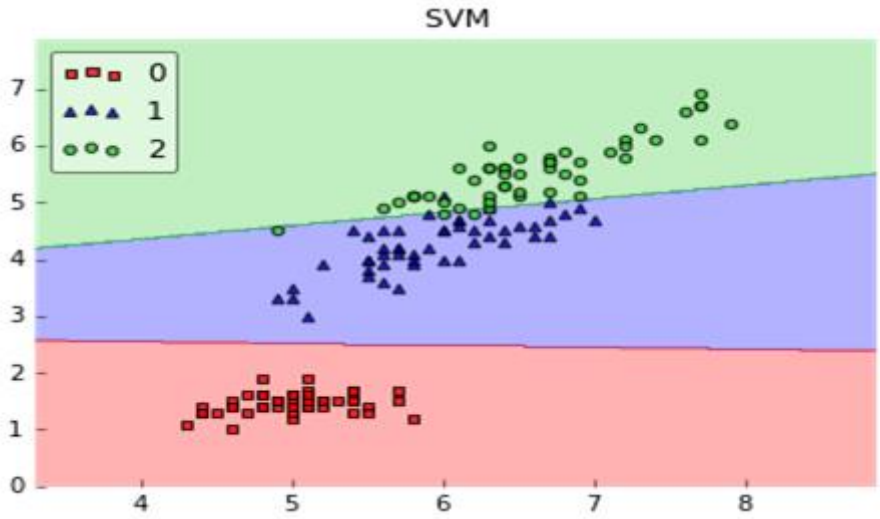

Figure 2-1 : SVM With Linear Kernel

SVM with Radial Basis Function Kernel:

Among all the kernels that is used with SVM the most popular one is Radial Basis Function Kernel. In this kernel the distance between the hyperplane considering the vectors is measured using metric squared Euclidean distance. Here free constant can be selected which is very difficult task to do and can results in overfitting data for a typical amount of constant.

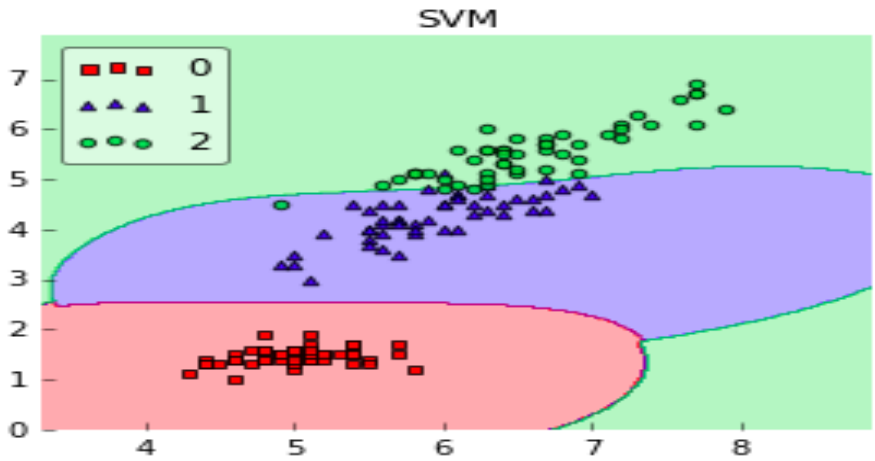

Figure 2-2 : SVM With Rbf Kernel

2.1.4 Algorithm 4: Random Forest

Random forest algorithm is a supervised classification algorithm. The best advantage of this algorithm is that it can be used for both classification and regression problems. It mainly works by creating a forest with some number of trees. The robustness of this algorithm depends on the number of trees in the forest. The higher the number of trees, the higher is the accuracy in this algorithm. Random forest algorithm uses decision tree concept**.** Decision trees are instinctive models that uses top down approach, where the root creates binary splits until stopping criteria is met. The binary splitting of nodes provides a predicted value depending on the internal nodes leading to the terminal nodes. In case of a classification problem, a decision tree outputs a predicted target class for each of the produced terminal nodes. If the training dataset with targets and features are given, the decision tree algorithm will provide with some set of rules**.** Those set of rules can be used to perform the prediction on the test dataset.

In decision tree algorithm calculating nodes and forming the rules are usually done using the information gain and Gini index calculations. But, in random forest algorithm, the process of finding the root node and splitting the feature nodes will happen randomly. The random forest algorithm works maintain two stages. The first stage is creating the random forest. And the second stage is performing prediction for the created random forest classifier.

The pseudo code for creating the random forest is -

1. Randomly select **"k"** features from total **"m"** features.
2. Where **k << m**
3. Among the **"k"** features, calculate the node **"d"** using the best split point.
4. Split the node into **child nodes** using the **best split**.
5. Repeat **1 to 3** steps until "l" number of nodes has been reached.
6. Build forest by repeating steps **1 to 4** for "n" number times to create **"n" number of trees**.

The following pseudo code is used for performing prediction on the created random forest classifier –

1. Takes the test dataset and uses the rules of each randomly created decision tree to predict the outcome and stores the predicted outcome.

2. Calculates the votes for each predicted value.

3. Considers the highly voted predicted target as the final prediction from the random forest algorithm.

2.1.5 Algorithm 5: Logistic Regression

Logistic regression is a classification algorithm used to assign observations to a discrete set of classes. Unlike linear regression which outpu0ts continuous number values, logistic regression transforms its output using the logistic sigmoid function to return a probability value which can then be mapped to two or more discrete classes.

Types of logistic regression

- Binary (Pass/ Fail)
- Multi (Cats/Dogs/ Sheep)
- Ordinary (Low/Medium/ High)

2.2 Literature Review

Many research based works have been done for English language. Literature review shows that many established supervised learning algorithm have been used for text or document categorization. Most frequently used techniques are K-Nearest Neighbor (KNN), Naive Bayes (NB), N-Grams, Decision Tree (DT), Neural Network (NNet), Support Vector Machines (SVM) etc. NB is the most frequently used approaches for text classification and it is easy for implementation and computation. In [8], they have used Nave Bayes technique to categorize the content in a website and they got 80% accuracy for ten categories. A large number of comparative studies are also done in English document categorization. Such as the author in [6] used KNN, NB and Term Gram for this task. They showed a comparative study where the accuracy of KNN is a better choice than NB and Term Gram.

Also in [14], Pratiksha Y. Pawar and S.H. Gawande did a comparative study on DT, KNN, Rocchios Algorithm [15], Back propagation, NB and SVM. Here for 20 new group's dataset they showed SVM performed far better than all the other approaches they have used. Also in [9] authors compare SVM against NB and KNN. Here they proved that SVM is better than KNN and NB. Besides Tam, Santoso and Setiono [16] Showed that KNN performed better than NNet and NB for English Document Categorization. Thorsten Joachims in [17] was the first who proposed the use of linear SVM with TF-IDF. In [18], authors developed a classification model for categorization of cricket sports news. They used SVM which was based on LibSVM and got best performance.

Though most of work has been done for English languages but there are also some other works which have been done for some other languages. Such as for Arabic language, SVM in [19] and NB in [20] have been used for automatic text classification. For Tamil language NNet was used in [21] with vector space model. Another work was done in [22] for Punjabi language.

Few works have been done for categorize Bangla language documents categorization. In [1], authors used NB classifier and Chi Square Distribution feature selection. In [11], authors compared four supervised learning techniques for label web document into 5 categories. Another work which have been done using N-gram [23] procedure to sort Bangla daily paper Corpus. Also in paper [24] authors showed that SVM works better

than other supervised learning for large sample set of classification [25] and derived the algorithm from structural risk minimization theory [26]. In our work, we have used many well-established techniques such as Random Forest, SVC, Linear SVC, Linear Regression, KNN and Gaussian NB. Among all Logistic Regression performed far better than all the other techniques with accuracy 97.3%.

Chapter 3

3 Our Methods

In this section, we provide the details of our methodology. Fig. 3-1 shows the steps of the methodology followed in this paper. First, we used a tool for collecting URLs of news articles from popular news provides in Bangla and stored Bangla articles in the database. Then, we performed pre-processing of the dataset and performed feature extraction on the dataset. After that, we applied feature selection on the dataset to find the optimal set of features. Several classifiers were then tested and the best model was stored. Finally, based on the best model we developed the tool and tested our model that used the same features.

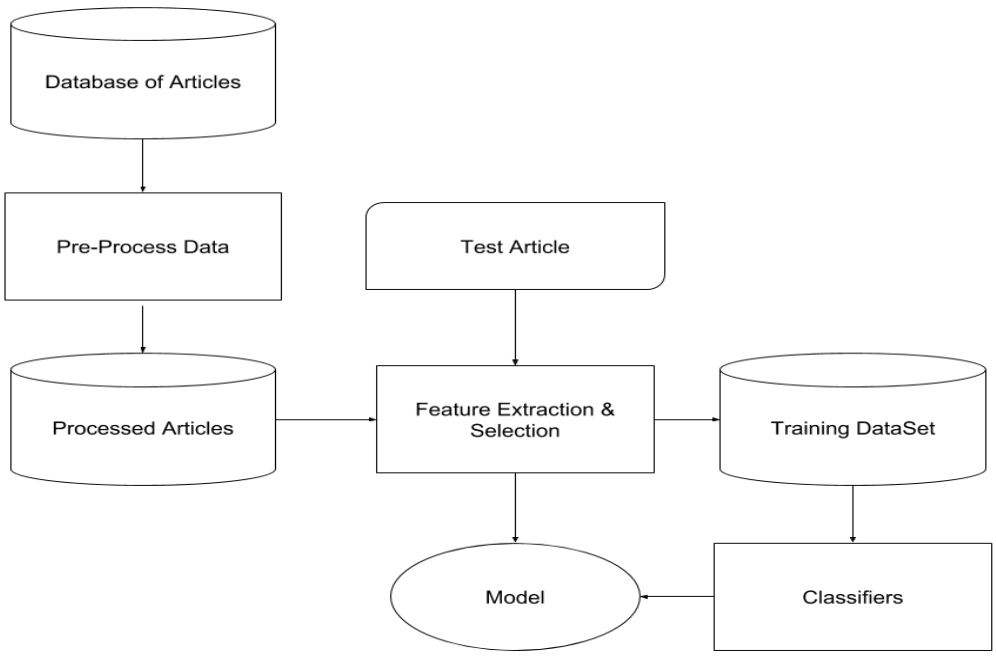

Figure 3-1: System diagram of our methodology

3.1 Preparation of Data

Data Collection is a huge and difficult process. To collect the right amount of data and process it into a right form is a big task. Our data processing had several steps as describe in this section.

1) Collection of Articles: Our first task was to collect articles from online news provider prothom-alo and bdnews24. To do that, we tried to extract pattern of news presentation in that website to collect data in an automated way so that we can use it to build larger dataset easily. At first we built Web API which takes date range as parameter and fetches the URL of all articles within that date range and saves in a SQL SERVER database. Then we fetched URL of articles during July 2017 to September 2017. Using those URLs we fetched all the articles corresponding to those URLs and saved in database. Fig. 3-2 shows a screenshot of the data extraction tool.

2) Punctuation Removal: Firstly meaningless symbols and Punctuations like [. , ” ‘ | ’ { } ( ) !] have been removed to make the dataset noise free. In this task each and every symbol was replaced by a space to split the sentence into words.

3) Stemming: After removing punctuations we extracted the unigrams from articles separated by a white space and stored them. We also kept a list of bivoktis. For each unigrams those we kept we looked for another unigram within each article and if it contained unigram+bivokti then replaced with the token. The algorithm works as below:

For each article

For each bivokti in the bivoktilist

For each unigram

If article contains unigram+bivokti

Then replace with unigram

End for

Endfor

End for

The process is known as finding root words, So that we can reduce the dimensions of final dataset.

4) Eliminating Irrelevant Words: After finding the base words we created a list of unnecessary words which are not important or irrelevant to categorize text document. The list includes numbers, pronouns, conjunctions and some other single letter words.

3.2 Feature Extraction

The n-grams typically are collected from a text or speech corpus. When the items are words, n-grams may also be called shingles. Using Latin numerical prefixes, an n-gram of size 1 is referred to as a ‘unigram’; size 2 is a “bigram”; size 3 is a “trigram”. Here we develop unigram model by splitting each word with a space and make a contiguous list of items from a of text documents. More precisely we split all the sentences into words in a specific text document by using unigram. The main purpose of using n-gram is to convert the documents into series of words where we can calculate the value of TF-IDF easily.

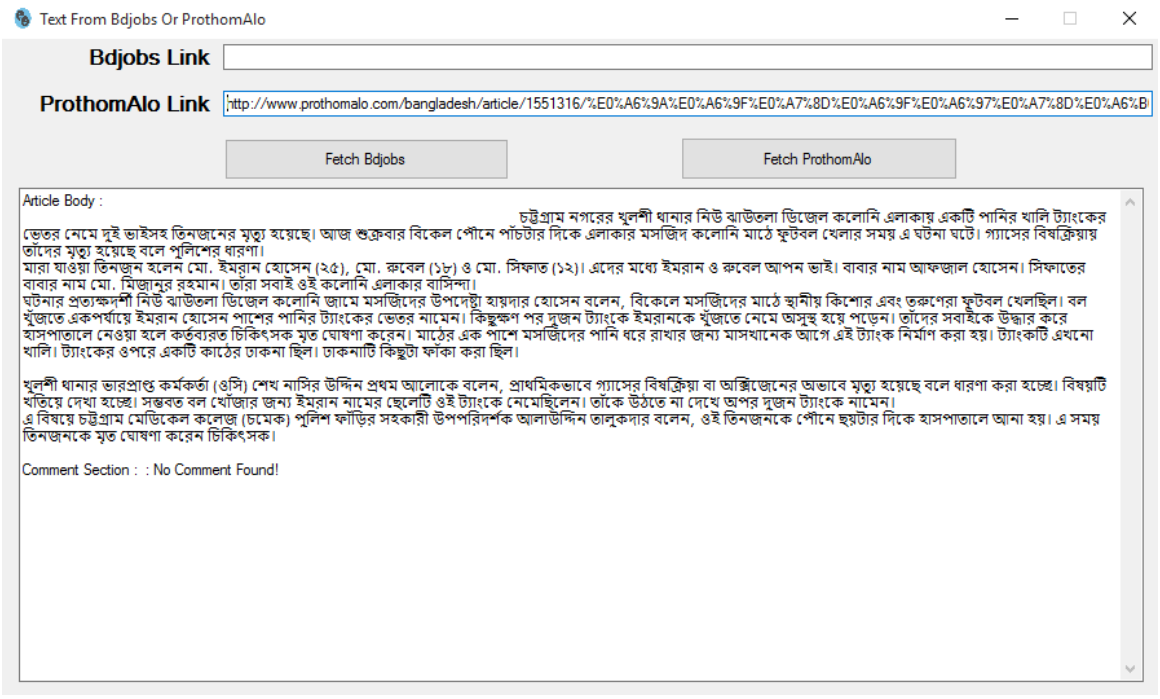

Figure 3-2: Screenshot of the data extraction tool.

1) Calculating TF-IDF: TF-IDF is an information retrieval technique that weights a terms frequency (TF) and its inverse document frequency (IDF). Each word or term has its

respective TF and IDF score. The product of the TF and IDF scores of a term is called the TF-IDF weight of that term. The higher the TF-IDF score, the rarer the term and vice versa. It is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. It is often used as a weighting factor in searches of information retrieval, text mining, and user modeling. The TF-IDF value increases proportionally to the number of times a word appears in the document and is offset by the frequency of the word in the corpus, which helps to adjust for the fact that some words appear more frequently in general. TF-IDF is one of the most popular term-weighting schemes today; 83% of text based recommender systems in digital libraries use TF-IDF. In the case of the term frequency TF(t,d), the simplest choice is to use the raw count of a term in a document, the number of times that term t occurs in document d.

$$\text{TF (t)} = \frac{\text{Number of times term t appears in a document}}{\text{Total number of terms in the document}}$$

Number of times term t appears in a document Total number of terms in the document the inverse document frequency is a measure of how much information the word provides, that is, whether the term is common or rare across all documents. It is the logarithmically scaled inverse fraction of the documents that contain the word, obtained by dividing the total number of documents by the number of documents containing the term, and then taking the logarithm of that quotient.

$$\text{IDF(t)} = log\left(\frac{\text{Total number of documents}}{\text{Number of documents with term t in it}}\right)$$

Total number of documents Number of documents with term t in it. In our dataset we have calculated TF-IDF to find out significant words for each document. Five categories of document are present and the labels are [Technology, Sports, Entertainment, Economy and International]. In our dataset we represent total number of words as columns and total number documents as rows. The total size of the dataset is 5870 rows and 51324 columns. We have made the dataset and the extraction tool available for use at: https://github.com/sspaarkk/BanglaNLP. A summary of the dataset is given in Table I.

We have also created an online API where it is possible to download new articles by date in JSON format with the category as label. The API is available online from:

http: //samspark1-001-site1.etempurl.com/CorpusBuilder/. A screenshot of the web API is given in Fig. 3-3.

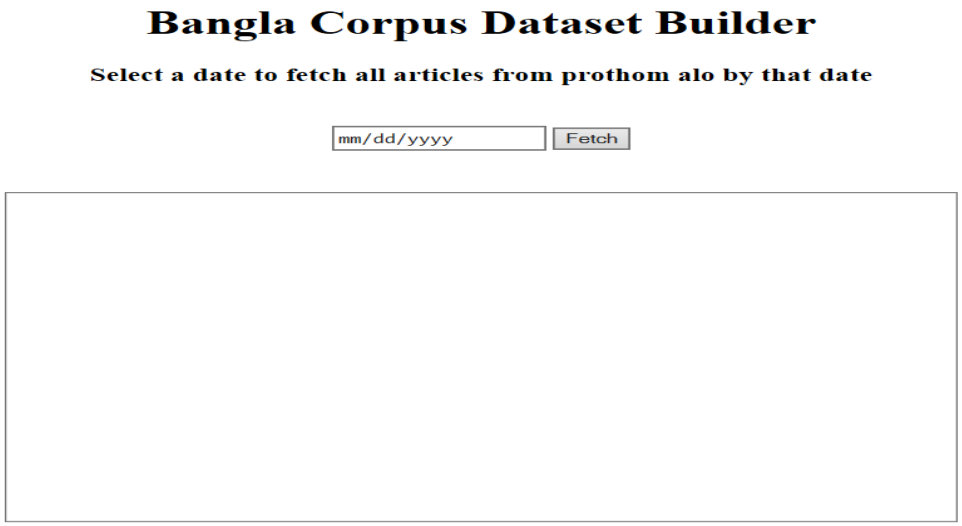

Figure 3-3: An online Web API for fetching data as JSON file

3.3 Feature Selection

Table 1: Summary of Dataset

Category	**Number of Documents**
Technology	459
Economy	864
Entertainment	1292
International	1395
Sports	1960
Total	5870

The total number of features used in our work was 51324. This is a huge number of data and it is the total number of words present in the vocabulary. However, not all words or n-grams are important for document categorization. In the next stage, we have reduced the features to increase the result accuracy. From all the features, we have selected some important features to conduct the prediction where important features indicates those features which have strong relation with the content categories.

We have used chi square statistical test for selecting important features. Chi square test along with an algorithm gives us value for each features. We have selected those features whose value is more than zero and reduced those whose value is less than or equal to zero. By applying this feature selection algorithm in our existing dataset, the number of features we have got is 18644, where the number of instances are still the same.

3.4 Classification Algorithm

Several classification algorithms were used in this paper. They include: Random Forest, Support Vector Machines with radial basis kernel, Support Vector Machines with linear kernel, K-Nearest Neighbor, Gaussian Naive Bayes and Logistic Regression. We have used multi-class classification models of these algorithms. All of the algorithms were implemented using Python 3.6 and Scikit-learn library for machine learning.

Chapter 4

4 Results and Discussion

In this section, we provide details of our experiments. Since the algorithms were stochastic, we have run them 10 times for each of the experiments and reported the average results only.

4.1 Results

Several sampling methods are being used in the literature for comparing between classification algorithms. In this paper, we have used three techniques: training set validation, percentage split validation and cross-fold validation. In training set validation, the whole training set is used for testing and thus have higher chances of over-fitting. In the case of percentage split, the training set is divided into two parts: training and validation. The model is learned using the training set and the validation set is used for testing purpose. In this paper, we have used a 75% training and 25% validation for the percentage split test. In k-fold cross validation, the whole dataset is divided into k equal shaped sub datasets and in each iteration one subset is used as the validation set and the rest as training set. In this paper, we have used a five-fold cross validation. Since, this is a multi-class classification problem, we have used the average of the accuracies achieved in each category. In Table II, we report accuracies achieved by six algorithms used in this paper. Note, that we have also used a feature selection technique in this paper. Table II also shows the results achieved before and after the feature selection. The best values in each criteria is shown in bold faced fonts.

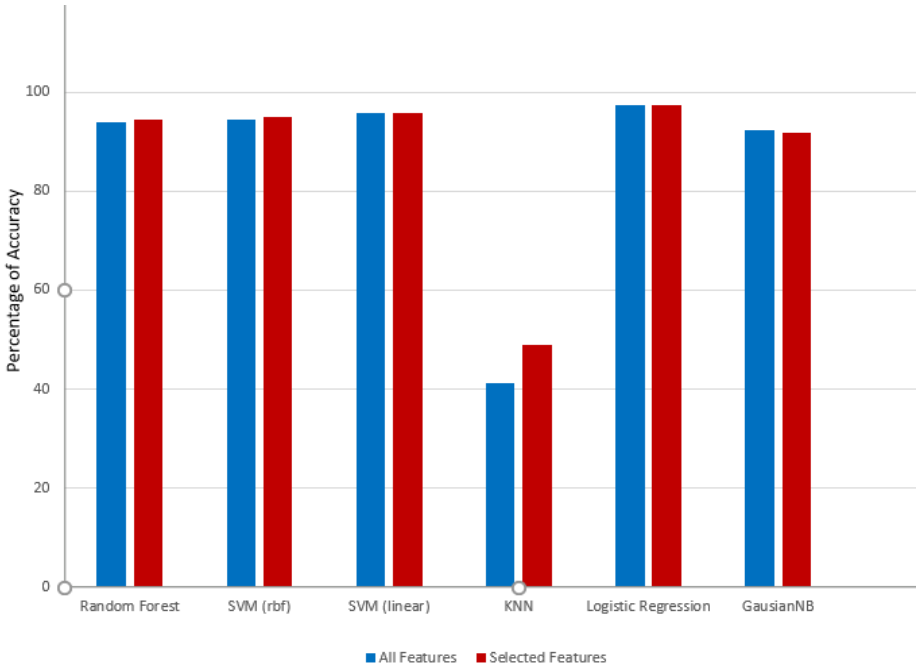

Figure 4-1:Plot of accuracies by different algorithms on the dataset using all features and selected feature using five fold cross validation.

4.2 Discussion

From the values reported in Table II, we can easily note that the best performing algorithm among all the classifiers in the Logistic Regression. Note that, the improvements of Logistic Regression algorithm compared to all the other classifiers. The worst performing classifier among them was the K-Nearest Neighbor algorithm. To show the gain in feature selection technique used in this paper, we plot the accuracies achieved by different algorithms on the dataset using 5-fold cross validations for all features and selected features in a bar chart in Fig. 4-1. The plot shows the improvements achieved by selected features in all the algorithms.

Table 2: Accuracy achieved by different algorithms on different sampling methods for testing and features

Algorithms	Accuracy Measuring Methodology On full dataset			Accuracy Measuring Methodology On selected features		
	Train Set	Percentage Split	Cross-fold Validation	Train Set	Percentage Split	Cross-fold Validation
Random Forest	99.5%	94.7%	93.9%	99.6%	94.7%	94.4%
SVM (rbf kernel)	100%	96%	94.5%	100%	96.3%	95%
SVM (linear kernel)	100%	96.8%	95.8%	100%	96.7%	95.8%
KNN	37.3%	38.8%	41.2%	44.3%	44.8%	48.9%
Logistic Regression	100%	97.6%	97.3%	100%	99.76%	97.3%
Gaussian NB	99.9%	92.9%	92.2%	99.7%	92.2%	91.8%

4.3 Feature Analysis

One of the very popular methods for visualizing document similarity is to use t-distributed stochastic neighbor embedding, t-SNE. By decomposing high-dimensional document vectors into 2 dimensions using probability distributions from both the original dimensionality and the decomposed dimensionality, t-SNE is able to effectively cluster similar documents. By decomposing to 2 or 3 dimensions, the documents can be visualized with a scatter plot. In Fig. 4-2, we have five categories, each category is representing a different color. Here we can clearly visualize that the words with similar meaning are clustered together for each category. This t-SNE represents high dimensional document vectors into 2 dimensions.

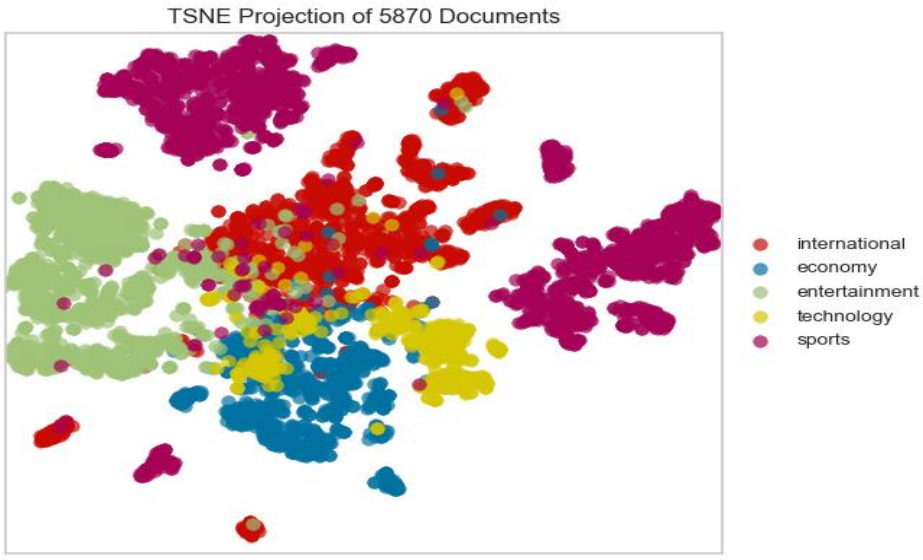

Figure 4-2: Visualizing features with different labels in the data with t-SNE.

We have also used another visualization method that displays how frequently words appear in a given document. We have selected the words and their maximum tf-idf within the dataset and plotted word cloud (See Fig. 4-3). The size of words is proportional to max tf-idf. All the words are then arranged in a cluster or clouds of words.

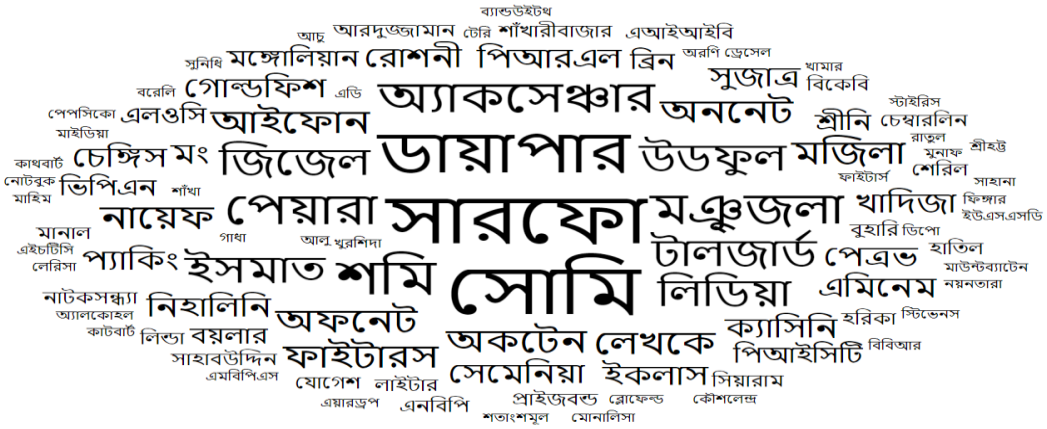

Figure 4-3: Word Cloud based on maximum tf-idf.

We have also selected the word and their average tf-idf within the dataset and plotted word cloud. We have calculated average tf-idf score to find out the words that present in

most of the document but with a high tf-idf value. Here in Fig. 4-4 it shows the words with high value of average tf-idf in large size.

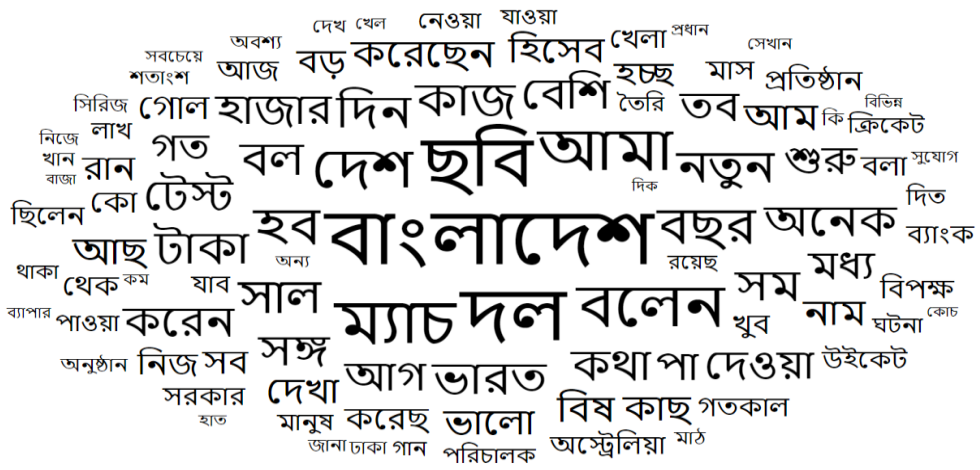

Figure 4-4: Word Cloud based on average tf-idf.

We have also created colored word cloud where each word is given a different color according to category in which it has highest frequency. We have counted the frequency of a word in each category, assign the category and plotted into the graph (See Fig. 4-5).

Figure 4-5: Word Cloud based with Category

4.4 Web Application

For the web tool we have developed a site using asp.net MVC which takes the article as input and removes the bivokties and unnecessary words from the content and generates a new instance just as the dataset. On the other hand, we developed an API using python flask which has a model hosted in it. The API takes an input and gives the predicted output in response. So the web tool sends the prepared processed instance to the API with a POST request and shows the predicted value after getting response from the python API. The whole web tool is prepared using C# and python.

Our web application is freely available to use from http://samspark1-001-site1.etempurl.com/. Fig. 4-6 shows a screenshot of the web application that we developed based on the method proposed in this paper.

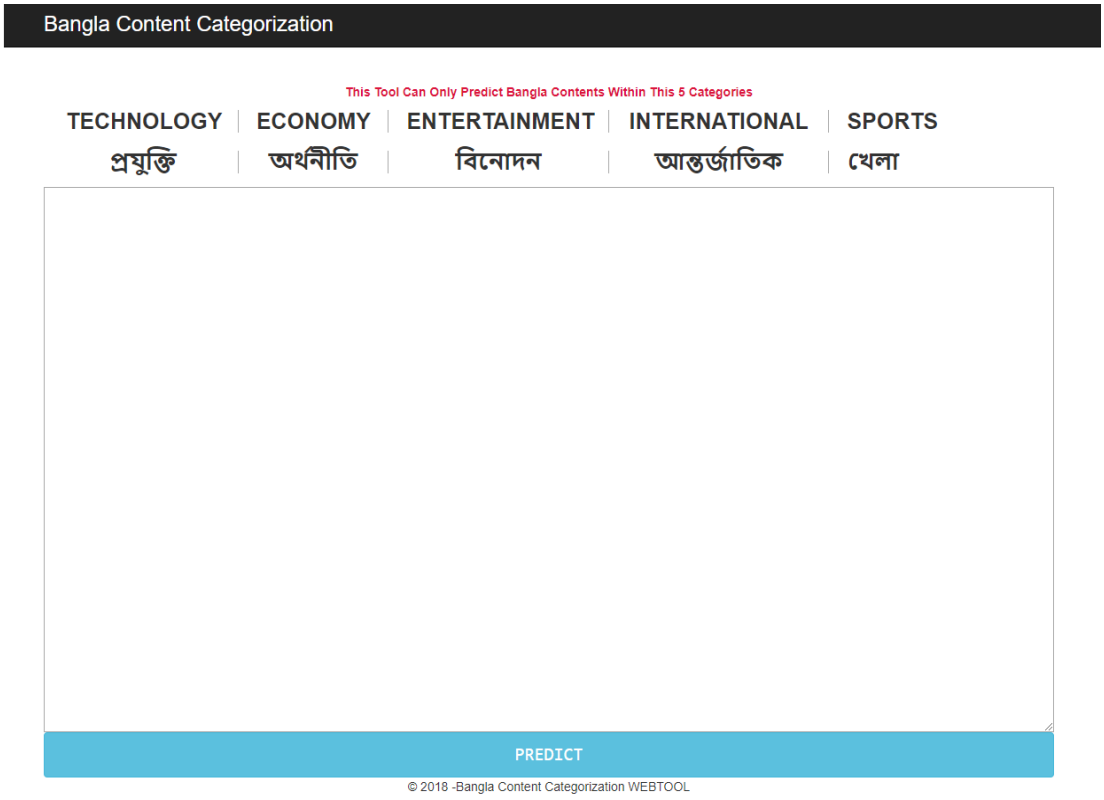

Figure 4-6:Screenshot of the web application developed based on the methods

Chapter 5

5 Conclusion

5.1 Summary

With rapidly increasing technology, the necessity of text categorization has increased due to its widespread use in text indexing, document sorting, text filtering, webpage categorization and many other fields. The rise in user-generated content for Bangla language across various genres news, culture, arts, sports etc. in the web has open the data to be explored and mined effectively, to provide better services and facilities to the consumers. The system described in this paper promotes the idea that using scarce resource, Bangla language can also be perfectly categorized. The result of the method we used is encouraging. Again by this paper we wanted to present a complete pipeline for the classification of Bengali documents. The code and API's are available online. The main contribution of this article is the development of an open source Bengali document dataset

5.2 Limitations

We could have achieved more accuracy. In this paper, we have only worked with 5 categories. Better results could have been achieved if we could manage to work with all the categories. We have found the root words within our dataset using the unigrams. We could have used a dictionary of Bangla words which would have found proper root words.

5.3 Future Work

In future, firstly, we would like to work on our dataset so that we can find root words using a dictionary rather than using the unigrams. By doing this we can find appropriate root words. Secondly, we would like to incorporate with all the categories to build the model completely. We would also like to work on Sentiment Analysis using user

comments made online in Bangla Language, so that it can predict the user views or reactions on a particular topic or news.

6 References

[1] F. Quadery, A. Al Maruf, T. Ahmed, and M. S. Islam, “Semi supervised keyword based bengali document categorization,” in Electrical Engineering and Information Communication Technology (ICEEICT), 2016 3rd International Conference on. IEEE, 2016, pp. 1–5.

[2] K. A. Hasan, M. Rahman et al., “Sentiment detection from bangla text using contextual valency analysis,” in Computer and Information Technology (ICCIT), 2014 17th International Conference on. IEEE, 2014, pp. 292–295.

[3] K. Hasan, A. Mondal, A. Saha et al., “Recognizing bangla grammar using predictive parser,” arXiv preprint arXiv:1201.2010, 2012.

[4] M. A. Islam, K. A. Hasan, and M. M. Rahman, “Basic hpsg structure for bangla grammar,” in Computer and Information Technology (ICCIT), 2012 15th International Conference on. IEEE, 2012, pp. 185–189.

[5] F. Sebastiani, “Machine learning in automated text categorization,” ACM computing surveys (CSUR), vol. 34, no. 1, pp. 1–47, 2002.

[6] V. Bijalwan, V. Kumar, P. Kumari, and J. Pascual, “Knn based machine learning approach for text and document mining,” International Journal of Database Theory and Application, vol. 7, no. 1, pp. 61–70, 2014.

[7] C. Apt´e, F. Damerau, and S. M. Weiss, “Automated learning of decision rules for text categorization,” ACM Transactions on Information Systems (TOIS), vol. 12, no. 3, pp. 233–251, 1994.

[8] A. S. Patil and B. Pawar, “Automated classification of web sites using naive bayesian algorithm,” in Proceedings of the international multiconference of engineers and computer scientists, vol. 1, 2012, pp. 519–523.

[9] Z.Liu,X.Lv,K.Liu,andS.Shi,“Studyonsvmcomparedwiththeother text classification methods,” in Education Technology and Computer Science (ETCS), 2010 Second International Workshop on, vol. 1. IEEE, 2010, pp. 219–222.

[10] J. F¨urnkranz, “A study using n-gram features for text categorization,” Austrian Research Institute for Artifical Intelligence, vol. 3, no. 1998, pp. 1–10, 1998.

[11] A. K. Mandal and R. Sen, “Supervised learning methods for bangla web document categorization,” arXiv preprint arXiv:1410.2045, 2014.

[12] A. N. Chy, M. H. Seddiqui, and S. Das, “Bangla news classification using naive bayes classifier,” in Computer and Information Technology (ICCIT), 2013 16th International Conference on. IEEE, 2014, pp. 366– 371.

[13] F. Kabir, S. Siddique, M. R. A. Kotwal, and M. N. Huda, “Bangla text document categorization using stochastic gradient descent (sgd) classifier,” in Cognitive Computing and Information Processing (CCIP), 2015 International Conference on. IEEE, 2015, pp. 1–4.

[14] P. Y. Pawar and S. Gawande, “A comparative study on different types of approaches to text categorization,” International Journal of Machine Learning and Computing, vol. 2, no. 4, p. 423, 2012.

[15] T. Joachims, “A probabilistic analysis of the rocchio algorithm with tfidf for text categorization.” Carnegie-mellon univ pittsburgh pa dept of computer science, Tech. Rep., 1996.

[16] V. Tam, A. Santoso, and R. Setiono, “A comparative study of centroidbased, neighborhood-based and statistical approaches for effective document categorization,” in Pattern Recognition, 2002. Proceedings. 16th International Conference on, vol. 4. IEEE, 2002, pp. 235–238.

[17] T. Joachims, “Text categorization with support vector machines: Learning with many relevant features,” in European conference on machine learning. Springer, 1998, pp. 137–142.

[18] T.ZakzoukandH.Mathkour,“Textclassifiersforcricketsportsnews,”in Proceedings of International Conference on Computer Communication and Management (ICCCM 2011), 2011.

[19] A. Mohd Mesleh, “Support vector machines based arabic language text classification system: feature selection comparative study,” in Advances in Computer and Information Sciences and Engineering. Springer, 2008, pp. 11–16.

[20] M. El Kourdi, A. Bensaid, and T.-e. Rachidi, “Automatic arabic document categorization based on the na¨ıve bayes algorithm,” in Proceedings of the Workshop on Computational Approaches to Arabic Script-based Languages. Association for Computational Linguistics, 2004, pp. 51– 58.

[21] K. Rajan, V. Ramalingam, M. Ganesan, S. Palanivel, and B. Palaniappan, “Automatic classification of tamil documents using vector space model and artificial neural network,” Expert Systems with Applications, vol. 36, no. 8, pp. 10914–10918, 2009.

[22] V. Gupta and V. Gupta, “Algorithm for punjabi text classification,” International Journal of Computer Applications, vol. 37, no. 11, pp. 30–35, 2012.

[23] M. Mansur, “Analysis of n-gram based text categorization for bangla in a newspaper corpus,” Ph.D. dissertation, BRAC University, 2006.

[24] M. S. Islam, F. E. M. Jubayer, and S. I. Ahmed, “A support vector machine mixed with tf-idf algorithm to categorize bengali document,” in Electrical, Computer and Communication Engineering (ECCE), International Conference on. IEEE, 2017, pp. 191–196.

[25] J. T.-Y. Kwok, “Automated text categorization using support vector machine,” in In Proceedings of the International Conference on Neural Information Processing (ICONIP. Citeseer, 1998.

[26] C. Cortes and V. Vapnik, “Support-vector networks,” Machine learning, vol. 20, no. 3, pp. 273–297, 1995.