Media Trend Observation by Analyzing Bangla News Articles using LDA Method

Md.Tanvir Hossain Hemel
(Student ID: 160238)
S.M.Muhaiminul Islam
(Student ID: 160240)

Computer Science and Engineering Discipline
Khulna University
Khulna-9206, Bangladesh
March,2020.

Contents

List of figure

List of tables

Chapter 1

Introduction

1.1 Introduction

This is an era of infinite number of digital data around us. With the rapid globalization of data, it has become very important to extract desired knowledge from data. We can use this tons of data to observe trending matters of media. For this purpose we can apply topic modeling approach on those data. Topic modeling has been introduced as a technique for researching on natural language processing. It is a technique for finding a collection of topics that represents the main information of the document from a group of documents which is consists of thousands of different kind of data corpus. It can also be thought of as a form of text mining a way to obtain recurring patterns of words in textual documents [1].
Topic modeling has been used frequently in languages like English, French etc. But in Bangla the use of topic modeling is almost rare. In recent years Bangla has become one of the richest languages over the world. Bangla has become one of the most popular languages in the world after the announcement to observe February 21st as International Mother Language Day annually by UNESCO on November 17th, 1999 [2].As Bengali is used everyday by more than 250 million people of the world, primary language in Bangladesh and secondary language in India [3, 4, 5], it has huge potential for the business as well as governments. So Bangla will join the race of working field of natural language processing in near future. Bangla newspaper, Bangla Wikipedia, Bangla literature, Bangla news portals, blogs, eBooks, web pages, search engines even the lyrics of Bangla songs have become very available and informative in recent time. Among them various topics become trendy. Everyday a new topic can be popular as that topic is discussed by most of the people. From tons of news around us, everybody wants to know what is the most important, frequently discussed topic around us. But in Bangla this opportunity is not familiar to us because of the lack of research on NLP using this language. Working with Bangla is quite difficult because of it's complex grammatical structure. Besides the scarcity of datasets as well as the tools also discourage researchers to walk in this field. However, for knowing the valuable informations from huge amount of data, we have to extract the hidden topical patterns of the words from various kind of documents. But it is really a big challenge to extract this huge amount of Bangla documents and find relation between them and get meaningful information. Topic modeling can solve this problem.Topic modeling is one of the effective methods for

finding useful hidden structure in the collection of documents [6]. And applying topic modeling on newspaper is much efficient for knowing current trend because newspapers always discuss the current and most happening incident. By evaluating the recent news, we can have an idea about media trend. But topic modeling methods only gives us the topic which is mostly repeated, but it doesn't say us the label of those topic. So we cannot easily understand which topic stands for which matter. To solve this problem we have created our own method to label the topics. By which we will be able to have a study about recently happening facts.
Since there are very few works on Bangla topic modeling, we have tried to focus on this field and extract knowledge from Bangla news articles to observe the trend of the media as well as develop the topic modeling method by giving the topic specific labels.

1.2 Problem Statement

Keeping track of political, national or economical events are important for different application now a days. This kind of data is useful for business organization for setting up there policy to political or governing body of a country to measure the impact of their decision or to get a clue about what they should do. Digital data from sources like Online Newpapers, Blogs, Social Media can help a lot to get a clue on currently trending matters. For this purpose, Topic Modeling is a well known technique. Topic Modeling technique helps us to analyze discussed topic throughout the given text corpus and convenient way to visualize them for more insights. However conventional topic modeling techniques often offers modeling the text corpus with some non-labeled topic number. But for a proper understanding on trends, labeling those topics (Preferably with user defined Label Set) is much important. It makes the model more easy to understand for a human being.

Again working with languages other than English (Like Bangla) often get hindered by issues like lack of dataset and tools to preprocess them. There are scarcity of well-structured corpus that is a must to perform any kind of Natural Language Processing experiment.

So an effort has been made to find out a convenient method to label generated Topics by topic modeling algorithms in this research. We have also examined the method on a Bangla language corpus and attempted to get some insights on occurring trend from output of our method.

1.3 Objectives

As an effort of solving problems mentioned earlier some objectives has been set.Those objectives have guided us through the research to reach at our goal.

(a) Collecting text document from different sources.Also we will structure them in a proper way for the model training.
(b) Preprocessing the texts is the next thing to do. Preprocessing includes punctua-

tion removal, Stemming word token, removing stop words, filtering out unnecessary word tokens etc.
(c) Determine the optimum number which will be used as the number of topics while extracting topics using topic modeling algorithm on our dataset. In this purpose we will use topic coherence measurement.
(d) Topic modeling algorithm with optimum number of topics on preprocessed text corpus from our Data Set will be applied then. We will follow Latent Dirichlet Allocation method with proper parameter for building up the model.
(e) Assign label to each topics generated by LDA will be done. We will try to label the topics from a predefined Label Set which matches with the topic most.
(f) Finally we will get some text documents as test data and calculate topic distribution of the model. Then we will try to get some insights on various kinds of trending matters with the help of label of each topic we assigned before.

1.4 Thesis Organization

1.4.1 Chapter 1: Introduction

This chapter includes the introductory concepts, problem statements, our goal and the objectives.

1.4.2 Chapter 2: Literature Survey

In this chapter the background topics have been discussed. Here we have shown different types of topic modeling method. Also we have discussed some related works which are also done in the field of topic modeling or Bangla language.

1.4.3 Chapter 3: Building Bangla News Article Dataset

This chapter contains all the works which are related to dataset. Extracting different types of features from our dataset has been discussed here.

1.4.4 Chapter 4: Design of Media Trend Observing Model

This chapter has been discussed with the methodology we have used in our research. All the working steps have been given broadly there.

1.4.5 Chapter 5: Result Discussion

In this chapter we have given some example and tested our proposed method. Results have been briefly discussed here.

1.4.6 Chapter 6: Conclusion and Future Works

Here we have concluded our research as well as proposed some future works that can be applied to develop this method further.

Chapter 2

Literature Survey

To find the solution of keyword searching, finding the relevant topic from a huge number of documents, many techniques of topic modelling has been introduced. Here some of them are discussed briefly:

2.1 Background Knowledge

2.1.1 VSM-Vector Space Model

It is an algebraic model for representing text documents as vectors. It is used for keyword search. It represents both the document and the query as a vector. D = (w1 , w2 , w3 , ..., wn) Q = (q1 , q2 , q3 , ..., qn) This method has been used in a large part of information retrieval research. Any documents is consist of a large number of words from where we just need smaller set of words that are the key words of that document. So, that we apply stop word removal and stemming technique on it[7]. Doing that we consider the unique words as the collection of documents which is named as dictionary. Then vectors from the collection of the documents can be collected as a matrix which is called as term document matrix[8]. Where the matrix represents how many times the terms are being repeated in that document. After that translates a document or keyword query into a vector in vector space[9]. Typically terms are single words, keywords, or longer phrases.If words are chosen to be the terms, the dimensionality of the vector is the number of words in the dictionary. Now cosine of the document and query vector is used to see the numeric similarities between those vectors. By comparing the deviation of the angle between the document vector and query vector comparative rankings of documents for that query can be calculated.

$$cos\theta = \frac{(q.d)}{((|q|.|d|))} \tag{2.1}$$

The less the value of the angle between document and query the much similarities exist.

Advantages:

1. Simple model based on linear algebra
2. Allows computing a continuous degree of similarity between queries and documents
3. Allows ranking documents according to their possible similarities
4. Allows partial matching

Disadvantages:

1. Long documents are poorly represented because they have poor similarity
2. Search keywords must precisely match document terms, substrings may cause problem
3. Semantic sensitivity; documents with similar context but different term vocabulary won't be associated
4. The order in which the terms appear in the document is lost in the vector space representation.

2.1.2 LSI-Latent Semantic Indexing

LSI uses bag of word model, which results in a term-document matrix(occurrence of terms in a document). Rows represent terms and columns represent documents. LSI learns latent topics by performing a matrix decomposition in the document-term matrix using Singular value decomposition(SVD). It is also known as Latent Semantic Analysis-LSA. LSA assumes that words that are close in meaning will occur in similar pieces of text. First of all, it converts documents into term document matrix[10]. The matrix containing word counts per paragraph (rows represent unique words and columns represent each paragraph) is constructed from a large piece of text. The main idea behind LSI is to utilize term co-occurrence to derive a set of latent concepts, words which frequently occur together are assumed to be more semantically associated[11]. SVD is used to reduce the number of rows while preserving the similarity structure among the column. Document term matrix(X), decomposes into U and V, which are orthogonal matrices and S is a diagonal matrix. This is called a singular value decomposition.

$$X = USV^t \tag{2.2}$$

Paragraphs are then compared by taking the cosine of the angle between the two vectors formed by any two columns. Values close to 1 represent very similar paragraphs while values close to 0 represent very dissimilar paragraphs [12].

Advantages:

1. Text does not need to be in sentence form for LSI to be effective. It can work with lists, free-form notes, email, Web-based content, etc.
2. LSI automatically adapts to new and changing terminology, and has been shown to be very tolerant of noise

3. LSI overcomes two of the most problematic constraints of Boolean keywords query: multiple words that have similar meanings (Synonims) and words that has more than one meaning

Disadvantages:

1. Compare the documents better in only the low-dimensional space
2. LSA cannot capture polysemy (i.e., multiple meanings of a word) because each occurrence of a word is treated as having the same meaning due to the word being represented as a single point in space
3. Limitations of Bag of Words (BOW) model, where a text is represented as an unordered collection of words

2.1.3 PLSA

To overcome the difficulties of LSA, PLSA which is known as Probabilistic Latent Semantic Analysis was introduced by Thomas Hoffman in 1999. It is also known as Probabilistic Latent Semantic Indexing. This method has been evolved from the LSA method. In LSA it reduces the term-document matrix using singular value decomposition. As well as in PLSA this decomposition is obtained from a mixture decomposition obtained from latent class model. In this method, let d denotes the label of a document, z is a topic, w represents a word. Therefore, P (z—d) denotes the probability of topic z in document d, and P (w—z) means the probability of word w in topic z [9]. To search a word in the document a procedure is followed. First select a document d which have the probability P(d). Then randomly choose a topic z from the topic distribution P(z—d) . Finally randomly choose a word w from the corresponding distribution over the topic P(w—z)[13]. PLSI focuses on the co-occurrence of the words and documents.

$$P(w,d) = P(d)\sum P(z|d)P(w|z) \tag{2.3}$$

Advantages:

1. It reduces term-document matrix with respect to LSA.
2. Singular value decomposition in PLSA is obtained from a mixture decomposition obtained from latent class model.

Disadvantages:

1. As there is no parameters to this model, we don't know how to assign probabilities to new documents
2. The number of parameters for pLSA grows linearly with the number of documents. So it can create overfitting.

2.1.4 LDA-Latent Dirichlet Allocation

It is one of the most popular methods for topic modelling. In our research we have used LDA to extract topic from news articles. This topic modelling method has been described briefly below:

To find the topics from a document, LDA does the following for each document m:

1. Let there are k topics across the documents
2. Distribute these k topics across document m this distribution is known as α and can be symmetric or asymmetric
3. For each word w in document m, assume its topic is wrong but every other word is assigned the correct topic
4. Probabilistically assign word w a topic based on two things:

   -what topics are in document m

   -how many times word w has been assigned a particular topic across all of the documents this distribution is called β
5. Repeatedly done this process until all the documents are being checked

This work can be explained with a plate notation

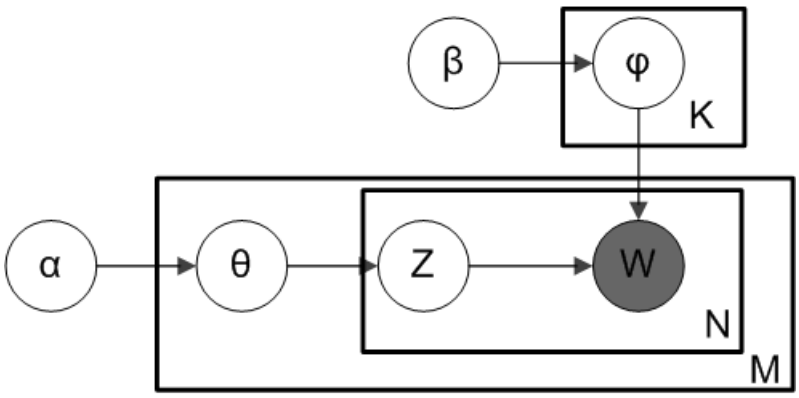

Figure 2.1: Plate notation for LDA with Dirichlet-distributed topic-word distributions from [14]

Here,
α is the per-document topic distributions,
β is the per-topic word distribution,
θ is the topic distribution for document m,
ϕ is the word distribution for topic k,
z is the topic for the n-th word in document m and
w is the specific word.

2.2 Related Works

2.2.1 Statistical Topic Modeling

In [15] a research has been done using LDA on some news article of The Times of India. They applied LDA to identify topic from those news articles. Then they have applied Labeled LDA to label the topics. They have also implemented Partially LDA to name the sub topic. They have used a dataset of 400 news articles. Finally they have measured the performance of their model using CVB0 and GS approaches.

2.2.2 Optimize Bangla Topics and News Classification Using LDA

In [16] topic modeling has been introduced on Bangla. In this research LDA has been implemented on Bangla news corpus. Also they have extracted topic using Doc2Vec approach. Then they have compared between this two methods. According to their research, LDA gave much better result than Doc2Vec.

2.2.3 Bengali Text Summarization

In [17] they have shown an approach to summarize information of news portal, blogs, books etc. by using extractive method. By summarization they have tried to extract the basic ideas of the topic and also showed whether the topic is relevant or not. They have extracted the topic of the document by word and sentence frequency.

2.2.4 A Deep Learning Approach Combined with the LDA For Topic Modeling in Bangla

In [18] they have researched on Bangla topic modeling and sentiment analysis. They have dealt with finding the topics from news corpus. Also they have classified the news with similarity measurement. They extracted the topics from news corpus using LDA topic modeling technique. As their dataset they have used the Bangla comments from Facebook. Then they compared the models by experimenting document similarity. They have also implemented LDA on Bnagla news corpus collected from *Prothom Alo* newspaper.

2.2.5 Labeled LDA

In [19] researchers have distinguished between SVM method of text classification and the Labeled LDA. They have used two LLDA based classifiers named LLDA-C and SLLDA-C to classify the news article. They have proved that both LLDA-C and SLLDA-C are better classifier than SVM. They have assigned the topics one or multiple labels also using LLDA.

2.3 Conclusion

In this chapter different methods for topic modeling are discussed. Related works of topic modeling and Natural Language Processing on Bangla Language have also discussed.

Chapter 3

Building Bangla News Article Dataset

3.1 Introduction

In this chapter we have briefly discussed about our dataset. We have also extracted some knowledge or features from our dataset which could be use as resource. As well as an overall idea of the dataset can have by these. As the resource of Bangla datasets are not so familiar so we have tried to contribute in this field for future work on Bangla.

3.2 Creating Datasets

- **Training Dataset :** We have created this dataset with 70,000 Bengali news articles among them 40,000 was collected from *kaggle*[1]. The dataset was not balanced for all categories. For example, number of news of *National* category was 12,239 whereas there were only 3,354 news articles of the category *Sports*. So we have collected another 30,000 news articles of different category from different Bangla Daily Newspapers which was required to balance the dataset. In Training Dataset there are news of different category like national, international, sports, technology, economy etc.

- **Labeling Dataset :** For labeling the topics we have used another datasets consist of Bangla news article. In which the category of all the news were known. We have used six category of news each of them are consist of ten news articles.

- **Media Trend Testing Dataset :** For observing the media trend we have used another news corpus collected from recently published Bangla daily newspaper. We have collected news article of three weeks. For each week we have collected fourty recent news.

  Trend Testing Dataset no.01 was consist of news from 10th July to 16th July,2019.

[1] `https://www.kaggle.com/zshujon/40k-bangla-newspaper-article`.

Trend Testing Dataset no.02 contains forty news of the date between 2nd October to 8th October,2019.

And Trend Testing Dataset no.03 was created by the top news of the date from 5th December to 11th December,2019

3.3 Data Preprocessing

Before using the news articles we preprocessed the data to get better result. As well as we preprocessed the words to make it more efficient to apply our methods. Bangla is grammatically very complex language. So it was a big challenge to preprocess the documents without deprecating the hidden meaningful topics.

- **Tokenization and punctuation removal :**

  In this step all the text splits into sentences, as well as all the sentences splits into words. So after this step the full corpus becomes only a set of words. Here we have also removed punctuation marks from the text corpus.

	Before Tokenizing and Removing Punctuation Marks	After Tokenizing and Removing Punctuation Marks
Step 1 : Paragraph to sentence	ব্যাবসায়ী ও আড়তদারেরা বলছেন, গুজবের কারনে হঠাত পেঁয়াজের দাম আবারও বেড়েছে।	[ব্যাবসায়ী ও আড়তদারেরা বলছেন গুজবের কারনে হঠাত পেঁয়াজের দাম আবারও বেড়েছে]
Step 2 : Sentence to words	[ব্যাবসায়ী ও আড়তদারেরা বলছেন গুজবের কারনে হঠাত পেঁয়াজের দাম আবারও বেড়েছে]	['ব্যাবসায়ী', 'ও', 'আড়তদারেরা', 'বলছেন', 'গুজবের', 'কারনে', 'হঠাত', 'পেঁয়াজের', 'দাম', 'আবারও', 'বেড়েছে']

Figure 3.1: Tokenization and Punctuation Removal

- **Stop words removal :** Stop words are those words which are less significant to represent meaning of a document. These words are less significant in a document but occur frequently e.g. conjunctions. So we need to remove those words to have only significant words to our corpus. We have created a dictionary having 415 stop words. Applying that dictionary we have removed unnecessary words from our dataset.

Before Removing Stop Words	After Removing Stop Words
['ব্যাবসায়ী', 'ও', 'আড়তদারেরা', 'বলছেন', 'গুজবের', 'কারনে', 'হঠাত', 'পেঁয়াজের', 'দাম', 'আবারও', 'বেড়েছে']	['ব্যাবসায়ী', 'আড়তদারেরা', 'গুজবের', 'পেঁয়াজের', 'দাম', 'বেড়েছে']

Figure 3.2: Stopwords Removal

- **Stemming :**

  All the individual words are reduced to their root form in this step by stemming process. We have used an open source Bengali Stemmer [2] to stem the words after removing stop words from the corpus. It splits out Suffix out of Bangla word token as shown in Fig 3.3

Before Stemming	After Stemming
['ব্যাবসায়ী', 'আড়তদারেরা', 'গুজবের', 'পেঁয়াজের', 'দাম', 'বেড়েছে']	['ব্যাবসায়ী', 'আড়তদার', 'গুজব', 'পেঁয়াজ', 'দাম', 'বেড়']

Figure 3.3: Stemming

3.4 Creating Bag of Words

In this step, a dictionary is created from those preprocessed data containing the number of times a word appears in the total corpus.
We filtered out words occurring less than 0.1% or more 0.8% throughout the document.Among those we used 100000 words for building up the BoW(Bag of Words). Example is shown in Fig 3.4

Preprocessed Data	Bag of Words
['ব্যাবসায়ী', 'আড়তদার', 'গুজব', 'পেঁয়াজ', 'দাম', 'বেড়']	{'ব্যাবসায়ী' : 1, 'আড়তদার' : 1, 'গুজব' : 1, 'পেঁয়াজ' : 1, 'দাম' : 1, 'বেড়' : 1}

Figure 3.4: Creating Bag of Words

As we can see in this set of words, none of the words are repeated once, so all the words will have the bag of words value 1.

3.5 Other Extracted Knowledge

3.5.1 Creating TF-IDF

tf–idf or TFIDF, short for *term frequency–inverse document frequency*, 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 number of documents in the corpus that contain the word, which helps to adjust for the

[2] `https://github.com/banglakit/bengali-stemmer/tree/dev/bengali_stemmer`.

fact that some words appear more frequently in general. Here we have extracted this tf-idf values upon the words of our datasets to see their appearance among the documents.
Here is a least example of tf-idf scores of each words of the corpus which can be named as token.

token	score
অকুস্থল	0.05665898103050247
অঙ্গাঙ্গিভাব	0.05603938662721833
অতলান্তিক	0.06879822550063801
অনুশীলন	0.028417754443459866
অপরাধ	0.022060548479296904
অবস্থান	0.032462545537314244
অভিধান	0.04729025240858795
অভিনয়	0.1659858945005715
অভিনয়শিল্পী	0.18121562298952534
অভিনেত	0.03665756470969712
অভিনেতা	0.07198095000413145
অর্থ	0.015391081304169517
অর্থাৎ	0.018885321058800135
অসত্য	0.32394310393530074
অসহায়	0.04792565335234179
অসৎ	0.04007102904219596

Figure 3.5: TF-IDF score of individual tokens

3.5.2 Creating Unigram, Bigram and Trigram

An n-gram is a contiguous sequence of n items from a given sample of text or speech. Using Latin numerical prefixes, an n-gram of size 1 is referred to as a "*unigram*"; size 2 is a "*bigram*" (or, less commonly, a "digram"); size 3 is a "*trigram*. We have built different dictionaries having this unique list of unigram, bigram and trigram list for both stemmed and non-stemmed words.

A short example is given here applying unigram, bigram and trigram on text both stemmed in figure 3.6 and non-stemmed in figure 3.7.

Non-Stemmed Text	দ্রব্যমূল্য নিয়ন্ত্রণে অভিযান শুরুর আগে বাজারের নাম ফাঁস হয়ে যাচ্ছে
Unigram	দ্রব্যমূল্য, নিয়ন্ত্রণে, অভিযান, শুরুর, আগে, বাজারের, নাম, ফাঁস, হয়ে, যাচ্ছে
Bigram	দ্রব্যমূল্য_নিয়ন্ত্রণে, নিয়ন্ত্রণে_অভিযান, অভিযান_শুরুর, শুরুর_আগে, আগে_বাজারের, বাজারের_নাম, নাম_ফাঁস, ফাঁস_হয়ে, হয়ে_যাচ্ছে
Trigram	দ্রব্যমূল্য_নিয়ন্ত্রণে_অভিযান, নিয়ন্ত্রণে_অভিযান_শুরুর, অভিযান_শুরুর_আগে, শুরুর_আগে_বাজারের, আগে_বাজারের_নাম, বাজারের_নাম_ফাঁস, নাম_ফাঁস_হয়ে, ফাঁস_হয়ে_যাচ্ছে

Figure 3.6: n-Gram of Non-stemmed text

Stemmed Text	দ্রব্যমূল্য নিয়ন্ত্রণ অভিযান শুরু আগ বাজারে নাম ফাঁস
Unigram	দ্রব্যমূল্য, নিয়ন্ত্রণ, অভিযান, শুরু, আগ, বাজারে, নাম, ফাঁস
Bigram	দ্রব্যমূল্য_নিয়ন্ত্রণ, নিয়ন্ত্রণ_অভিযান, অভিযান_শুরু, শুরু_আগ, আগ_বাজারে, বাজারে_নাম, নাম_ফাঁস
Trigram	দ্রব্যমূল্য নিয়ন্ত্রণ অভিযান, নিয়ন্ত্রণ অভিযান_শুরু, অভিযান_শুরু_আগ, শুরু_আগ_বাজারে, আগ_বাজারে_নাম, বাজারে_নাম_ফাঁস

Figure 3.7: n-Gram of Stemmed text

3.5.3 Creating Cluster

Clustering is the grouping of particular sets of data based on their characteristics, according to their similarities. K-means clustering is one of the most popular clustering algorithms in machine learning. We have applied this clustering method to see how individual words get clustered and keep themselves together. Kmeans formula keeps the words together which appears frequently among the same documents. Thus the text clustering using kMeans algorithm works.
Here is an example of some words classified in 6 clusters by applying KMeans clustering upon the corpus.

Cluster 0	অভিনয়শিল্পী,নাটক,আত্মীয়,করছি,সেপা,কবর,বলছি,নিহত,পরদিন,সাবেক,বানাইছি,জয়তু,গীতা,সালাম,আবুল
Cluster 1	তিনি,অভিনেতা,বিখ্যাত,যোগ,কদিন,আরেকজন,প্রেসিডেন্ট,হলিউড,অভিনেত,প্র্যাকট,অ্যান্ড,থিওরি,পিপল,লেভ,অনুশীলন
Cluster 2	বেসাতি,একবচন,বহুবচন,অভিনয়,মিথ্যাশ্রয়ী,কল্পনাশ্রয়ী,সত্যরূপ,বিশ্বাসযোগ্যভাব,যাহা,তাহা,অভিধান,সন্নিকট,সাতেক,জন্মগতভাব
Cluster 3	মিথ্য,কথা,শুন,খারাপ,যাক,বোঝা,আসল,সত্যি,ব্যাপার,বিষয়,হয়নি,ঘটেনি,ঘটব,প্রকাশ,কর
Cluster 4	উপস্থাপন,বিস্তারিত,নির্ধারণ,সম্মেলন,মেলা,নিশ্চিত,স্বভাবচিত্র,চালু,ভ্রমণ,সংবাদ,দুর্ঘটন,বর্তমান,ব্যবস,তিন,অর্থ
Cluster 5	লাগ,সোজা,করানো,সাদা,ক্যানসার,আক্রান্ত,ডাক্ত,চলাফ,মাইক্রোবাস,গ্রাম,দুর্ঘটনায়,মা,দুর্ঘটনা,সাম্ব্রি,গাড়ির

Figure 3.8: K-Means Clustering

3.6 Conclusion

In this chapter we have briefly discussed about our datasets, their preprocessing methods. Some other knowledge which has been extracted from our dataset is also described here.

Chapter 4

Design of Media Trend Observing Model

4.1 Introduction

To extract topic from news articles we have used LDA topic modeling method. LDA is stands for Latent Dirichlet Allocation. It is one of the most popular methods for topic modeling. In natural language processing, latent Dirichlet allocation (LDA) is a generative statistical model that allows sets of observations to be explained by unobserved groups that explain why some parts of the data are similar [14]. Topic modeling is used to find the main topic and hidden structure of those topics from a bunch of documents. We are going to use this method of topic modeling over Bangla language. Each document is a collection of many topics and each topic is a collection of various words, where each document is considered to have a set of topics. This is relatable with probabilistic latent semantic analysis(pLSA). LDA assumed to have a sparse Dirichlet prior which have the intuition that the document is cover only few topics and that topic have only few words which can be called key words of that topic. That is the only difference between LDA and pLSA topic modeling. LDA is a generalization of the pLSA model, which is equivalent to LDA under a uniform Dirichlet prior distribution [20]. The working procedure of LDA has described next. After extracting topics from news article we will give them label. As LDA doesn't assign any label for the extracted topic, we have applied a method to create "labeled LDA" and used it for further observation of media trend.

4.2 Methodology

We have used Bangla news article as the corpus. Here the steps are described which have done to apply LDA on the documents. Then we gave the topics proper label by training them with some true topic. Then we have used our labeled LDA model to see the change in the recent media trend. For doing these the steps we have completed are:

1. Creating Dataset
2. Data Preprocessing
3. Creating Bag of Words

4. Get optimum number for extracting topics
5. Applying LDA using the Bag of Words and Dictionary and extract topics
6. Label the topics using label set and create labeled LDA
7. Using the labeled LDA to observed the recent media trend

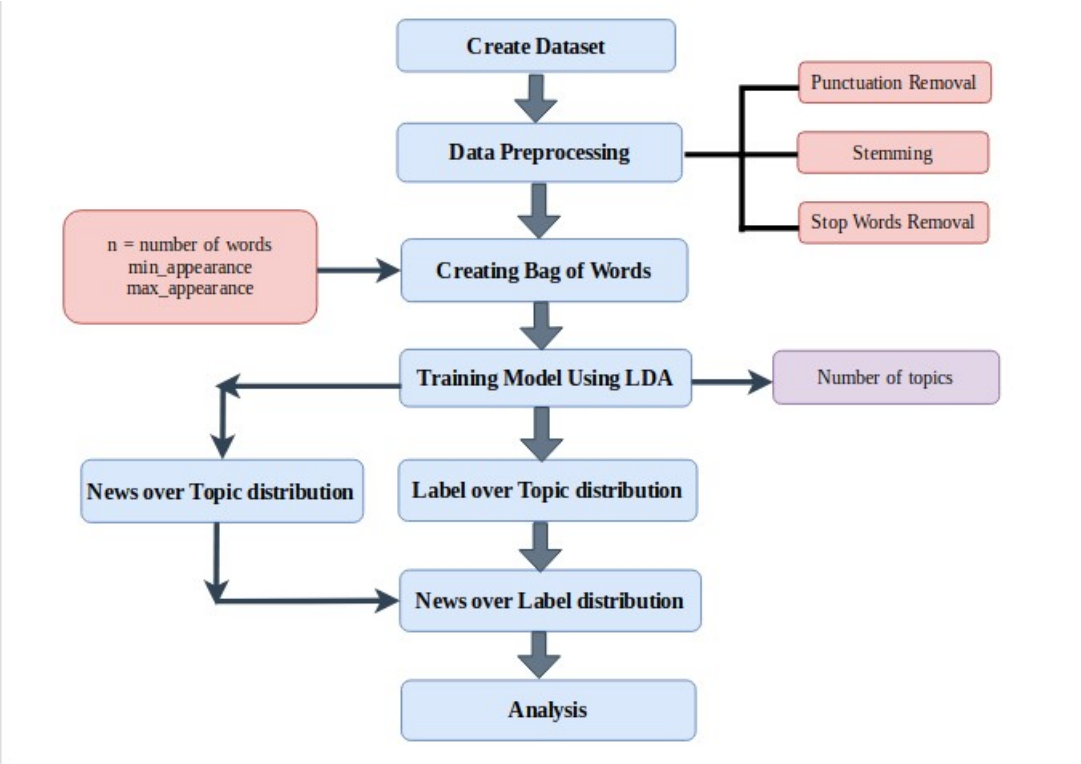

Figure 4.1: Block diagram of the proposed method

4.2.1 Determine optimum number of topics

Topic modeling methods gives us as many number of topics as we want. But to get most accurate topics from the datasets, optimum number of topics should be extracted. To get the optimum number of topic, topic coherence is a popular method. Topic Coherence measures score a single topic by measuring the degree of semantic similarity between high scoring words in the topic. These measurements help distinguish between topics that are semantically interpretable topics and topics that are artifacts of statistical inference. A set of statements or facts is said to be coherent, if they support each other.
We can get different coherence score for different number of topics. In Table 4.1 we can see the difference between coherence scores for different number of topics.

From the highest coherence score we can have the number of topics which will be optimum for our dataset. Here the coherence score for *6* is highest. So 6 topics will be the optimum number of topics to be extracted. So we have extracted 6 topics from our dataset using LDA.

Table 4.1: Coherence Scores for Different Number of Topics

Number of Topics	Coherence Score
4	0.477380780466019
5	0.479065339611616
6	0.483202558454852
7	0.479796847937369
8	0.472343695209541

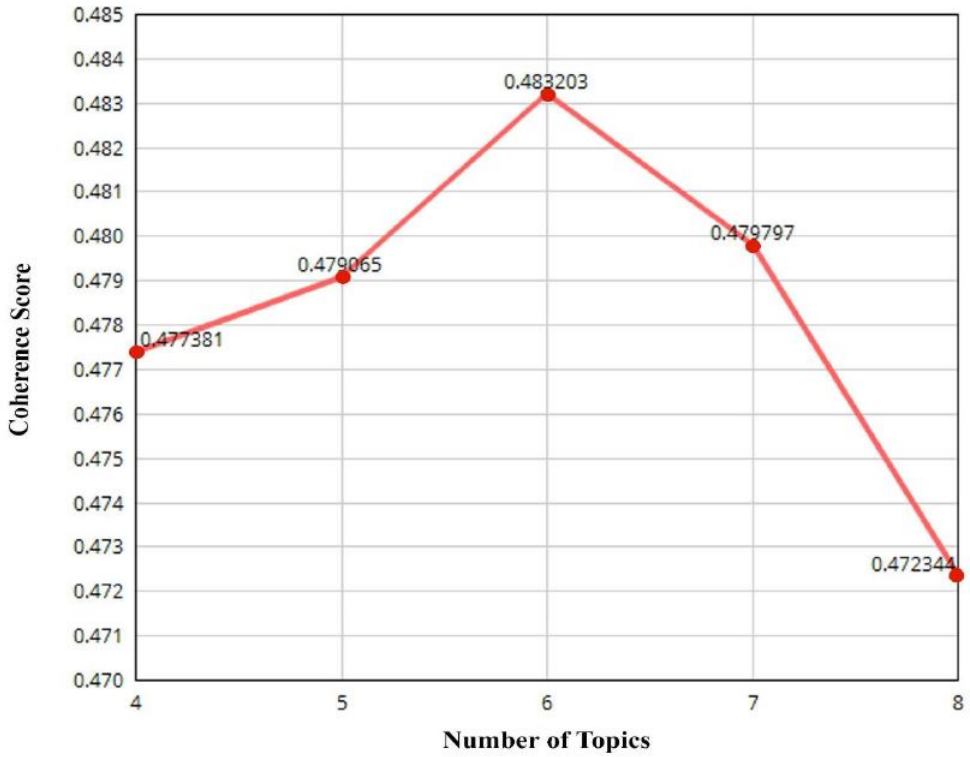

Figure 4.2: Label-Graphical Representation of Coherence Scores

4.2.2 Training model using LDA

We trained our topic model M using LDA on Train Dataset. We have set the number of topics 6 which has appeared to be the optimum number for this dataset during coherence measurements. We have also built the dictionary for getting meaningful insights out of the Model. Also it will help us in unseen document comparison with existing model. We trained out topic model M using LDA on *Training Dataset* for 6 topics as $T = \{t_0, t_1, t_2, t_3, t_4, t_5\}$

4.2.3 Assigning Label to Each Topic

For each $l_x \in L$ we had a set of documents $D_x = \{d_{x1}, d_{x2}...d_{xn}\}$ which are the representative of l_x. We computed relevance of each document in D_x with model and store as a matrix M_x

In our dataset, we had a label set L with 6 labels as $L = \{$*National*, *Sports*, *International*, *Technology*, *Economics* and *Others*$\}$

$$M_x = (T * D_x) = p(t_i|d_{xj}) \tag{4.1}$$

where i and j is index of elements of T and D For our dataset the Topic-Document matrix M_{sports} for topic *'sports'* can be seen in Table 4.2

Table 4.2: Correspondence of documents from label *Sports*

	Doc0	Doc1	Doc2	Doc3	Doc4	Doc5	Doc6	Doc7	Doc8	Doc9
Topic 0	0.032	0.01	0.01	0.01	0.01	0.01	0.01	0.107	0.01	0.01
Topic 1	0.814	0.994	0.995	0.995	0.960	0.837	0.977	0.473	0.995	0.846
Topic 2	0.149	0.01	0.01	0.01	0.037	0.01	0.018	0.01	0.01	0.01
Topic 3	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.225	0.01	0.150
Topic 4	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Topic 5	0.01	0.01	0.01	0.01	0.01	0.150	0.01	0.192	0.01	0.01

Topic-Document matrix M for other topics are also given below.

Table 4.3: Correspondence of documents from label *National*

	Doc0	Doc1	Doc2	Doc3	Doc4	Doc5	Doc6	Doc7	Doc8	Doc9
Topic 0	0.01	0.01	0.01	0.140	0.231	0.01	0.01	0.048	0.795	0.01
Topic 1	0.01	0.01	0.01	0.01	0.018	0.01	0.01	0.01	0.01	0.01
Topic 2	0.01	0.01	0.030	0.01	0.272	0.01	0.01	0.01	0.01	0.01
Topic 3	0.937	0.995	0.965	0.855	0.01	0.992	0.995	0.01	0.201	0.995
Topic 4	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.215	0.01	0.01
Topic 5	0.055	0.01	0.01	0.01	0.478	0.01	0.01	0.730	0.01	0.01

Table 4.4: Correspondence of documents from label *International*

	Doc0	Doc1	Doc2	Doc3	Doc4	Doc5	Doc6	Doc7	Doc8	Doc9
Topic 0	0.027	0.01	0.01	0.994	0.01	0.028	0.01	0.284	0.01	0.084
Topic 1	0.01	0.01	0.01	0.01	0.022	0.01	0.01	0.138	0.01	0.040
Topic 2	0.01	0.01	0.012	0.01	0.445	0.440	0.463	0.271	0.01	0.01
Topic 3	0.969	0.995	0.111	0.01	0.127	0.488	0.530	0.306	0.364	0.874
Topic 4	0.01	0.01	0.873	0.01	0.396	0.01	0.01	0.01	0.149	0.01
Topic 5	0.01	0.01	0.01	0.01	0.01	0.042	0.01	0.01	0.482	0.01

Table 4.5: Correspondence of documents from label *Economy*

	Doc0	Doc1	Doc2	Doc3	Doc4	Doc5	Doc6	Doc7	Doc8	Doc9
Topic 0	0.01	0.01	0.01	0.01	0.01	0.01	0.086	0.01	0.143	0.01
Topic 1	0.01	0.01	0.01	0.01	001	0.01	0.01	0.01	0.01	0.01
Topic 2	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.011	0.01
Topic 3	0.435	0.202	0.01	0.01	0.441	0.107	0.01	0.01	0.209	0.01
Topic 4	0.563	0.795	0.441	0.995	0.492	0.249	0.912	0.938	0.635	0.955
Topic 5	0.01	0.01	0.554	0.01	0.065	0.640	0.01	0.056	0.01	0.01

Table 4.6: Correspondence of documents from label *Technology*

	Doc0	Doc1	Doc2	Doc3	Doc4	Doc5	Doc6	Doc7	Doc8	Doc9
Topic 0	0.01	0.01	0.270	0.01	0.01	0.01	0.01	0.01	0.072	0.295
Topic 1	0.018	0.01	0.059	0.01	0.023	0.01	0.01	0.01	0.01	0.01
Topic 2	0.01	0.697	0.548	0.405	0.01	0.01	0.01	0.01	0.01	0.051
Topic 3	0.01	0.01	0.01	0.01	0.01	0.248	0.01	0.01	0.098	0.01
Topic 4	0.914	0.01	0.020	0.439	0.969	0.301	0.124	0.01	0.297	0.333
Topic 5	0.066	0.294	0.102	0.152	0.01	0.443	0.868	0.994	0.530	0.316

Table 4.7: Correspondence of documents from label *Others*

	Doc0	Doc1	Doc2	Doc3	Doc4	Doc5	Doc6	Doc7	Doc8	Doc9
Topic 0	0.018	0.012	0.569	0.370	0.679	0.339	0.582	0.368	0.01	0.01
Topic 1	0.201	0.012	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Topic 2	0.723	0.368	0.341	0.285	0.296	0.559	0.387	0.487	0.154	0.467
Topic 3	0.018	0.378	0.075	0.269	0.01	0.01	0.01	0.046	0.315	0.506
Topic 4	0.018	0.012	0.01	0.067	0.01	0.01	0.01	0.040	0.375	0.01
Topic 5	0.018	0.217	0.01	0.01	0.01	0.096	0.01	0.055	0.138	0.01

Now for each $t \in T$ and each $l_x \in L$ we calculated matrix N_{T*L} as

$$P(t|l_x) = \frac{\sum_{n=0}^{n=j} p(t|d_{xn})}{j} \tag{4.2}$$

Where j is the number of documents in D_x, T is topic set and L is label set. According to Eqn (4.2) we calculated probability distribution of each label among the topics. The matrix N_{T*L} for our experiment is shown in Table 4.8(Converted into percentage)
This process has shown in algorithm 1.

Algorithm 1 Algorithm for Label-Topic Correspondence

```
D ←set of documents of size m
M ←model trained by number of topic n
Mat ←Empty matrix of dimension n*m
for all doc in D do
  for all topic in M do
    Mat[topic][doc] := correspondence(topic, doc)
  end for
end for
```

Table 4.8: Label-Topic Correspondence

	National (জাতীয়) %	Sports (খেলা) %	International (আন্তর্জাতিক) %	Technology (তথ্যপ্রযুক্তি) %	Economy (অর্থনীতি) %	Others (অন্যান্য) %
Topic 0	12.17	1.39	14.22	5.59	2.34	18.79
Topic 1	0.19	88.88	2.07	1.02	0.10	9.08
Topic 2	3.00	2.04	16.24	16.15	0.34	19.45
Topic 3	69.43	3.76	47.65	3.50	13.98	13.51
Topic 4	2.60	0.51	14.15	33.05	70.02	16.24
Topic 5	12.61	3.42	5.67	40.69	13.22	22.93

We have per label topic distribution as N_{T*L} as shown in Table 4.8. We calculated another Matrix X_{T*L} to get per topic label distribution in percentage.
For any r, r_{t_r*L} from N_{T*L} represents any *row*. We calculated coefficient to be multiplied with elements of row for each row k_r as Eqn 4.3

$$k_r = \frac{100}{\sum_{n=0}^{size(T)} N_{T_r*L_n}} \tag{4.3}$$

For Example, From Table 4.8 k_1 can be calculated as Eqn 4.4

$$k_1 = \frac{100}{0.19 + 88.88 + 2.07 + 1.02 + 0.10 + 9.08} = 0.99 \tag{4.4}$$

Then we multiplied k_r with each value from the row k is derived. Thus we get Matrix X_{T*L} as presented in 4.9 for our experimental data.
In algorithm 2 this process is shown.

Algorithm 2 Algorithm for Row Normalized Label-Topic Correspondence

```
for all topic in M do
  for all i := 1 to m step 1 do
    total+ = Mat[topic][i]
  end for
  average = total/m
  Mat2[topic][label] := average * 100
end for
```

From the table we observe that every topic is appeared to be a mixture of different label at different ratio. However there are some topic like *Topic 1* biased enough to a single label to be labeled uniquely. On the other hand some topics like *Topic 0* or *Topic 2* are much more dispersed within more than one label to be labeled in unique way.

From the normalized value of Table 4.9 we have assigned each topic some category. As one topic can be form of different types of categories, we have tried to find which category is strongly appearing and which one is partially appearing in the topics by evaluating the Normalized Label-Topic Correspondence value. For example, in Topic 3 we can see that Label-Topic Correspondence value for National category is very high and after that International category has a medium Label-Topic Correspondence value.

From Figure 4.3 we can also see that in Topic 3, National news are appearing strongly as well as International news remains as partial category.

If a new news article tends to any of the topic which is extracted by LDA, according to our label we can make the decision which thing has been described strongly on that news as well as which matter(s) has been described partially on that article.

Table 4.9: Row Normalized Label-Topic Correspondence

	National (জাতীয়)	Sports (খেলা)	International (আন্তর্জাতিক)	Technology (তথ্যপ্রযুক্তি)	Economy (অর্থনীতি)	Others (অন্যান্য)
Topic 0 %	22.27	2.54	26.02	10.23	4.28	34.39
Topic 1 %	0.19	87.99	2.05	1.01	0.09	8.99
Topic 2 %	5.25	3.57	28.42	28.26	0.60	34.03
Topic 3 %	45.82	2.48	31.45	2.31	9.23	8.92
Topic 4 %	1.90	0.37	10.33	24.12	51.11	11.86
Topic 5 %	12.74	3.45	5.72	41.09	13.35	23.16

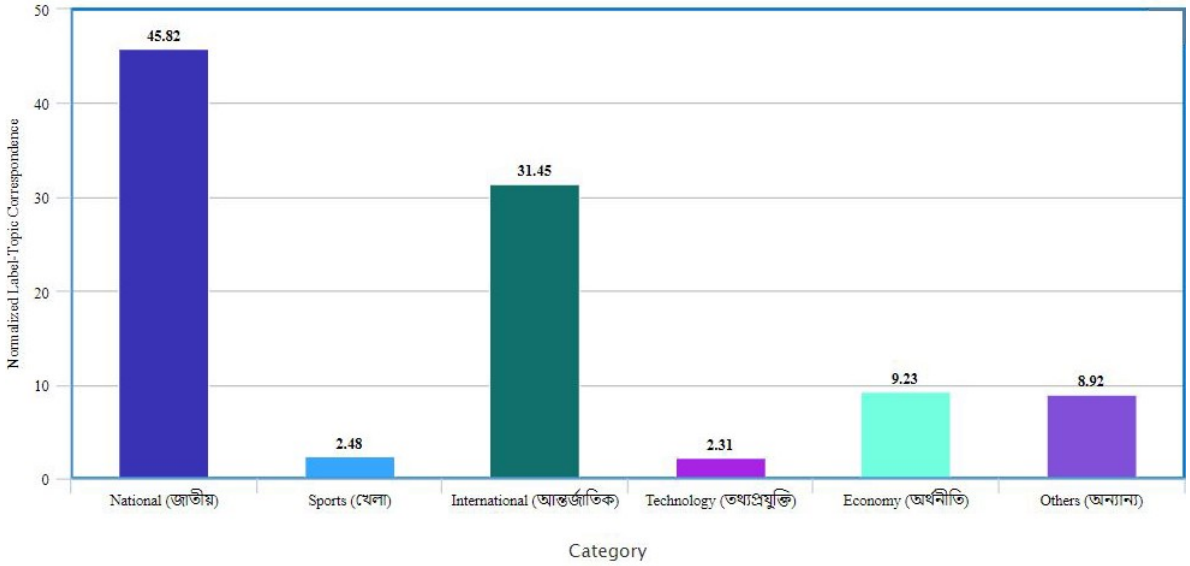

Figure 4.3: Label-Topic Correspondence Value Distribution

4.2.4 Guided LDA based Semi-supervised Model

Topic modeling is generally an unsupervised learning algorithm. We know that National and International are categories of their own. But if we don't get many articles about them or if they get mentioned together, they might get classified into one topic. Again if the dataset is being biased to an individual category, most of the extracted topic can be formed by that single category. To solve this problem and turn this unsupervised LDA to a semi-supervised model, guided lda is being introduced.

For this semi-supervised model of LDA we have given six boost words list each of them having 1000 of words. This words were distributed from the built cluster described in *section 3.5.3*. Using guided LDA we have also extracted six topics and observed how the topics kept similarities and dissimilarities among them.

Table 4.10: Give Label to the Topics

	Strongly Appearing Category	**Partially Appearing Category**
Topic 0	None	National(জাতীয়) International(আন্তর্জাতিক) Others(অন্যান্য)
Topic 1	Sports(খেলা)	None
Topic 2	None	International(আন্তর্জাতিক) Technology(তথ্যপ্রযুক্তি) Others(অন্যান্য)
Topic 3	National(জাতীয়)	International(আন্তর্জাতিক)
Topic 4	Economy(অর্থনীতি)	Technology(তথ্যপ্রযুক্তি)
Topic 5	Technology(তথ্যপ্রযুক্তি)	Others(অন্যান্য)

As 4.2 we have again calculated the correspondence of documents from label sports. We have tested the model with respect to ten news from the category *Sports.*
For each $l_x \in L$ we had a set of documents $D_x = \{d_{x1}, d_{x2}...d_{xn}\}$ which are the representative of l_x. We computed relevance of each document in D_x with model and store as a matrix M_x

In our dataset, we had a label set L with 6 labels as $L = \{National, Sports, International, Technology, Economics$ and $Others\}$

$$M_x = (T * D_x) = p(t_i|d_{xj}) \tag{4.5}$$

where i and j is index of elements of T and D For our dataset the Topic-Document matrix M_{sports} for topic *'sports'* can be seen in Table 4.11

Table 4.11: Correspondence of documents from label *Sports* for guided model

	Doc0	Doc1	Doc2	Doc3	Doc4	Doc5	Doc6	Doc7	Doc8	Doc9
Topic 0	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.531	0.01	0.01
Topic 1	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.125
Topic 2	0.01	0.010	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
Topic 3	0.891	0.988	0.995	0.214	0.996	0.984	0.408	0.465	0.995	0.657
Topic 4	0.01	0.01	0.01	0.782	0.01	0.01	0.589	0.01	0.01	0.215
Topic 5	0.102	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01

Comparing between table 4.2 and table 4.11 we can see that guided LDA have also extracted the topics more likely to the normal LDA. In normal LDA *Topic 1* had high intention to have the news related to the sports. The other topics may have a little appearance of sports. Here in guided model, *Topic 3* is the representative of the Topic 1 of normal LDA model. This topic has shown high relevance for the same news of label sports.
Both normal LDA and guided LDA models can extract the topic very efficiently. Guided LDA is preferable for short and biased datasets. As our model has been built on a huge non-biased dataset, we have used normal method of LDA to give the topic some label for observing the media trend. Guided LDA has been seen as a baseline of normal model.

4.2.5 Calculating Topic Distribution of Test News

We have got our trend testing dataset from different weekly news as a separate collection. For every collection, we calculated percentage of news classified in different topic.
For Test Set $W = \{w_0, w_1, w_2...w_n\}$ we calculated a table of $T * W$ dimension using Eqn 4.5 and Eqn 4.2. This represents ratio of news in each week that are classified in different topic.
For our Test News set of $\{w_0, w_1, w_2\}$ and each had forty news, we calculated the matrix as Y. It is shown as a table in Table 4.12

Table 4.12: Week News Distribution over Topic

	w1	w2	w3
Topic 0	17.05	20.15	19.13
Topic 1	33.97	9.38	17.32
Topic 2	13.95	6.10	16.25
Topic 3	7.90	24.24	24.18
Topic 4	24.02	28.59	18.93
Topic 5	3.11	11.54	4.19

4.2.6 Observing Media Trend

We had our *Trend Testing Dataset* of particular three weeks which had forty news of each week.This can be described as $W = \{w_1, w_2, w_3\}$. Each $w \in W$ can be written as $w_n = \{d_{n0}, d_{n1}, d_{n2}...d_{n39}\}$

Using Eqn 4.2 and Eqn 4.5 we again calculated distribution of news of each week against the six topics of already trained model. In Fig 4.4 we see the percentage of participation of different topic in each week. e.g. in week 1, 33.97% news has discussed the matter(s) which are in Topic 1. In week 2 that number down to 9.37%.

Already we had two tables to get accurate amount of involvement of individual label in each weekly news. Those are

- **Label Topic Correspondence Table** as X. It is actually a $T * L$ Matrix where T and L are Topic set and L is label set. Instance of this table for out experiment is Table 4.9
- **Weekly News Distribution Table** as Y. It represents Weekly News distribution over topics in T . For out Data Set, we have Table 4.12

For our Dataset we have arranged data from Table 4.12 as Stacked Bar Chart in Fig 4.4. We have topic distribution of weekly data but still those topic are mixture of labels. To get weekly data distributed over labels rather than topic we constructed a new Matrix Z representing the distribution of weekly news over different label. Each entry $z \in Z$ is calculated as Eqn 4.6

$$z = Z_{t,l} = \sum_{t=0}^{size(T)} Y_{tl} * X_{tl}\% \tag{4.6}$$

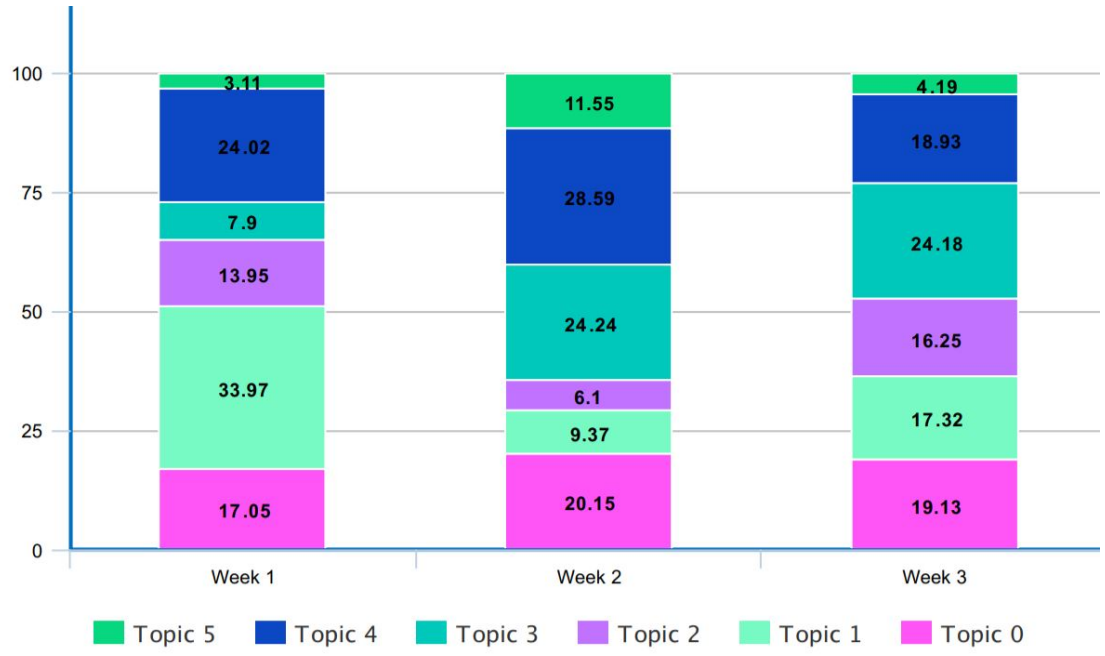

Figure 4.4: Label-Topic Correspondence Value Distribution

4.3 Conclusion

We have discussed and developed methods to achieve the objective that have been discussed earlier. That included Topic Model Development, Label assignment to Topic and testing with real news article as Test Data

Chapter 5

Result Discussion

5.1 Label Assignment

From Table 4.9 we had label distribution over the topics. We saw that each of the labels were distributed among the topics. There some label had significant dominance on a certain topic. But in case of some topics, no labels could show significant participation.For example, from figure 4.3 we saw that *Topic 3* is a combination of

- 45.82% of *National*,
- 2.48% of *Sports*,
- 31.45% of *International*,
- 2.31% of *Technology*,
- 9.23% of *Economy* and
- 8.92% of *Others*

For our context of application we considered more than 20% but less than 40% of participation as *Partial Appearance*.
Also more than 40% of participation was counted as *Strong Appearance* on that topic.
Thus the result could be seen in Fig 4.10

5.2 Insights on Media Trend

We had topic distribution per week's news in Fig 4.4. However every segment represents a single *topic* yet multiple label from *Label Set* because every topic is a composition of different labels with some percentage.
For example we have 33.97% news at *week 1* from *Topic 1*. Again *Topic 1* is composed of 87.99% of sports. Also *Sports* have similar involvement at different weight in other topics too. So to get exactly how much *Sports* was being discussed over the whole week, we calculated by similar approach. The pictorial presentation can be seen in Fig 5.1.

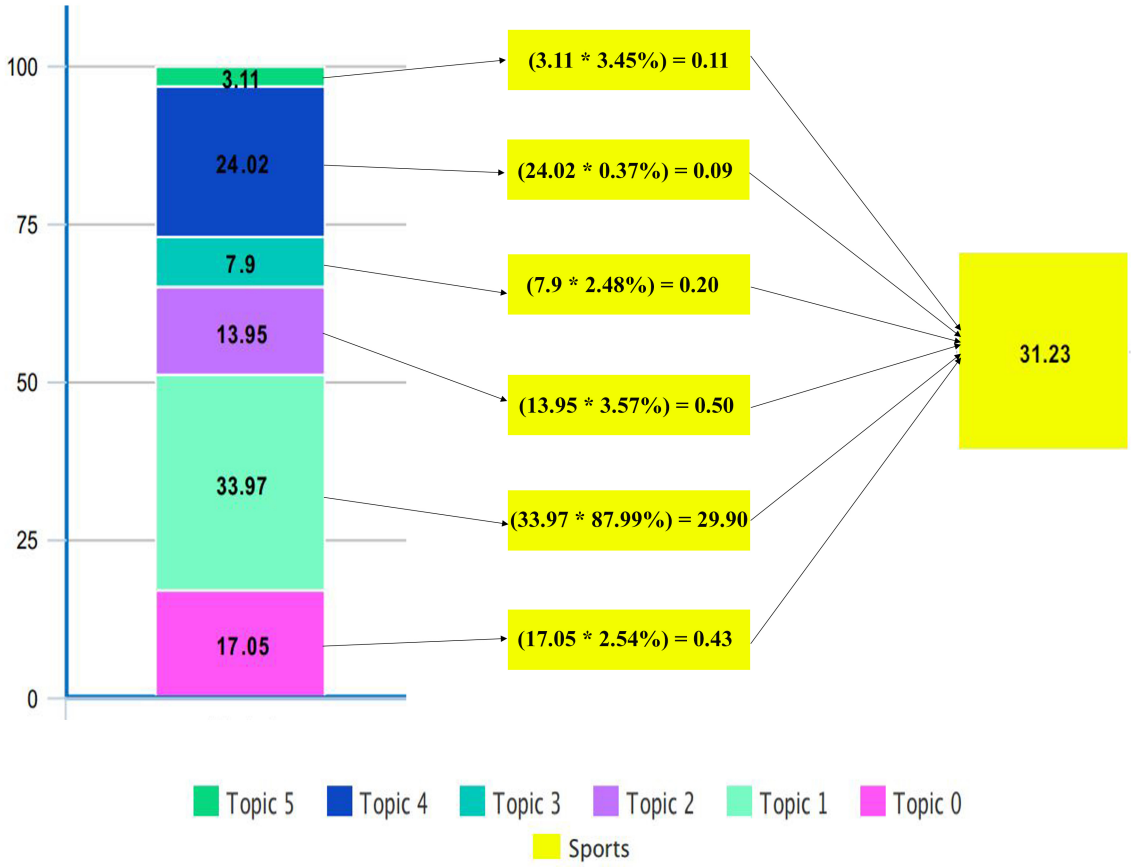

Figure 5.1: Label-Topic Correspondence Value Distribution for *Sports*

Similarly we have calculated amount of participation as per *Label* for each *label* shown in Fig 5.2

Then we plotted new distribution on the label and have true participation of label in weekly news. From 5.2 we can see the new ratio of the happening media trend among that week. This is the real portion of discussion which had been done on that particular time.

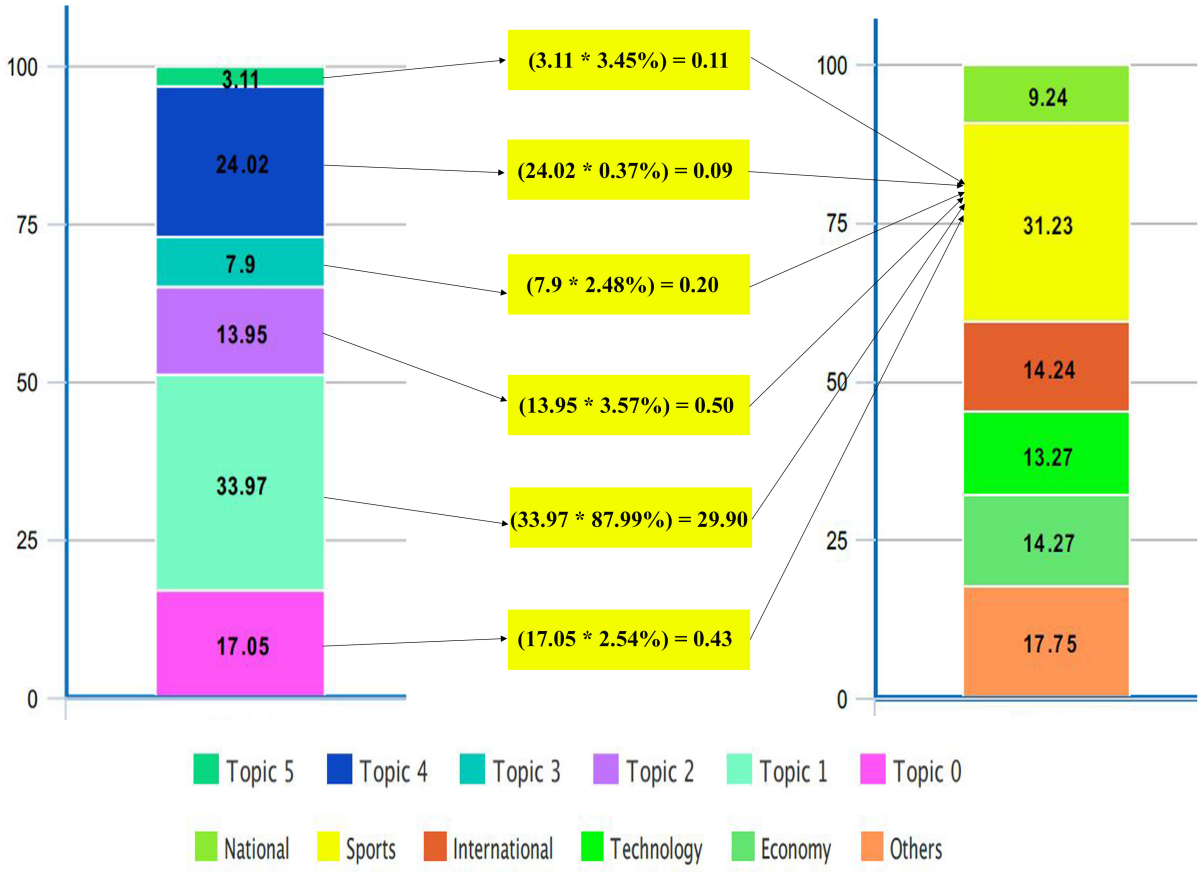

Figure 5.2: Label-Topic Correspondence Value Distribution

In 5.2 we have tested the trend by our Trend Testing Dataset no.01 where we have collected forty news from Bengali Daily newspapers. According to the date of that time, *ICC World Cup 2019* was happening. So most of newspapers cover the sports news on front page.

From 5.3 we can also see that the portion of Sports are very significant on that time duration. 34.06% of the news discussed about sports. Other topic like National, International, Technology, Economy or Others was average discussed topic then.

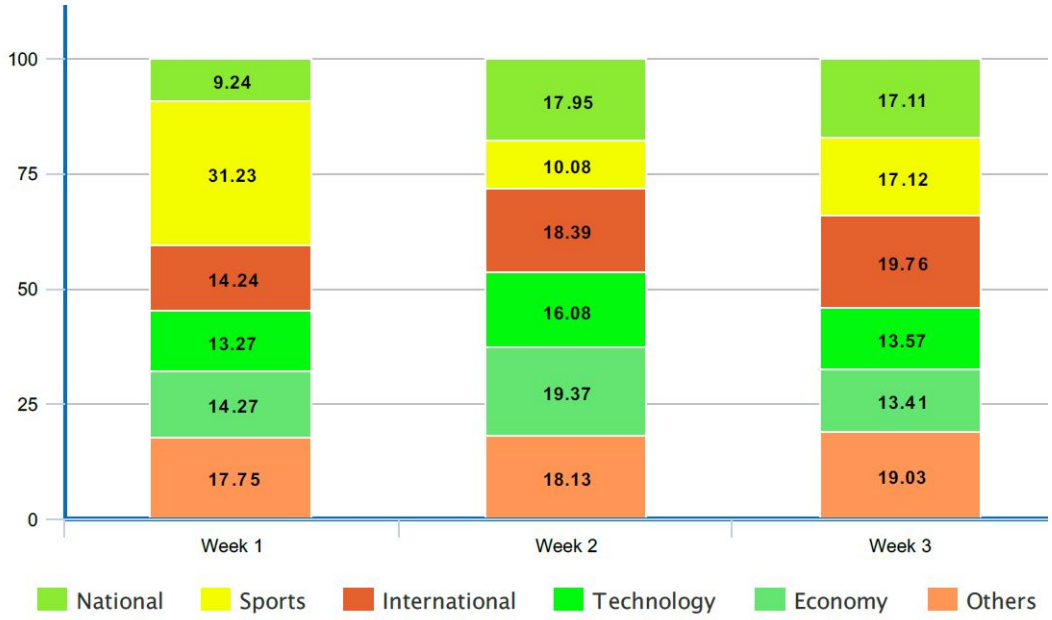

Figure 5.3: Label-Topic Correspondence Value Distribution

From 5.3 we also have the information on other two datasets collected on other two weeks.At week 2 we see strong presence of National and Economy labeled news than previous Week. That may be because of price hike in the market of Onion during the time news was collected. This issue took much attention so it appeared many national newspaper as top news. Price hike of any commodity usually has National and Economical impact. So that may be the reason for National and Economical having impact on the stacked bar chart of Week 2 as well.
Week 3 chart from Fig 5.3 is less biased to any particular label.

5.3 Conclusion

Here we have discussed about the result we got from experiment on Test Data with our proposed method. We have discussed this in a more intuitive way here.

Chapter 6

Conclusion and Future Works

In our research we have tried to develop a method to evaluate the Model generated by Topic Modeling Algorithms-LDA. Also we have tried to observe Other Unseen Text Corpus with a more human readable form by assigning user defined Labels. As a proof of it's application we have performed some experiment on certain News Article to trace down media trend with the flow of time. We have given related theories and test results of our experiment.

In our work we did not applied *Lemmatization* at the time of preprocessing the data. In our work if this step can also be done, data will more preprocessed and topic will be extracted more precisely.

Lexical Prior can be added to our work. Some seed words will be given to train the model. By controlling those seed words, we will be able to control the labeling more specifically and accurately.

We can implement this similar method on *Social Media Data* to observe the discussion over the social media. By doing this, prediction of social occurrence can be made.

Bibliography

[1] Rubayyi Alghamdi and Khalid Alfalqi. "A survey of topic modeling in text mining". In: *Int. J. Adv. Comput. Sci. Appl.(IJACSA)* 6.1 (2015).

[2] Wikipedia contributors. *International Mother Language Day — Wikipedia, The Free Encyclopedia*. [Online; accessed 14-December-2019]. 2019. URL: https://en.wikipedia.org/w/index.php?title=International_Mother_Language_Day&oldid=928939734.

[3] KM Hasan, Amit Mondal, Amit Saha, et al. "Recognizing Bangla grammar using predictive parser". In: *arXiv preprint arXiv:1201.2010* (2012).

[4] Md Asfaqul Islam, KM Azharul Hasan, and Md Mizanur Rahman. "Basic hpsg structure for bangla grammar". In: *2012 15th International Conference on Computer and Information Technology (ICCIT)*. IEEE. 2012, pp. 185–189.

[5] KM Azharul Hasan, Amit Mondal, and Amit Saha. "A context free grammar and its predictive parser for bangla grammar recognition". In: *2010 13th International Conference on Computer and Information Technology (ICCIT)*. IEEE. 2010, pp. 87–91.

[6] Jiaqi Zhu et al. "Mining user-aware rare sequential topic patterns in document streams". In: *IEEE Transactions on Knowledge and Data Engineering* 28.7 (2016), pp. 1790–1804.

[7] EE Ogheneovo and RB Japheth. "Application of vector space model to query ranking and information retrieval". In: *International Journal of Advanced Research in Computer Science and Software Engineering* 6.5 (2016).

[8] Bhagyashree Vyankatrao Barde and Anant Madhavrao Bainwad. "An overview of topic modeling methods and tools". In: *2017 International Conference on Intelligent Computing and Control Systems (ICICCS)*. IEEE. 2017, pp. 745–750.

[9] Lin Liu et al. "An overview of topic modeling and its current applications in bioinformatics". In: *SpringerPlus* 5.1 (2016), p. 1608.

[10] S Deepu, Pethuru Raj, and S Rajaraajeswari. "A Framework for Text Analytics using the Bag of Words (BoW) Model for Prediction". In: *Proceedings of the 1st International Conference on Innovations in Computing & Networking (ICICN16), Bangalore, India*. 2016, pp. 12–13.

[11] Dharmendra Sharma and Suresh Jain. "Evaluation of stemming and stop word techniques on text classification problem". In: *International Journal of Scientific Research in Computer Science and Engineering* 3.2 (2015), pp. 1–4.

[12] Susan T Dumais. "Latent semantic analysis". In: *Annual review of information science and technology* 38.1 (2004), pp. 188–230.

[13] Barbara Rosario. "Latent semantic indexing: An overview". In: *Techn. rep. INFOSYS* 240 (2000), pp. 1–16.

[14] Wikipedia contributors. *Latent Dirichlet allocation — Wikipedia, The Free Encyclopedia*. `https://en.wikipedia.org/w/index.php?title=Latent_Dirichlet_allocation&oldid=927159179`. [Online; accessed 14-December-2019]. 2019.

[15] Suganya C and Vijaya S. "Statistical Topic Modeling for News Articles". In: *International Journal of Engineering Trends and Technology* 31 (Jan. 2016), pp. 232–239. DOI: 10.14445/22315381/IJETT-V31P242.

[16] Mustakim Al Helal and Malek Mouhoub. "Topic Modelling in Bangla Language: An LDA Approach to Optimize Topics and News Classification". In: *Computer and Information Science* 11.4 (2018).

[17] Sheikh Abujar et al. "A heuristic approach of text summarization for Bengali documentation". In: *2017 8th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*. IEEE. 2017, pp. 1–8.

[18] Mustakim Al Helal. "Topic Modelling and Sentiment Analysis with the Bangla Language: A Deep Learning Approach Combined with the Latent Dirichlet Allocation". PhD thesis. Faculty of Graduate Studies and Research, University of Regina, 2018.

[19] Yiqi Bai and Jie Wang. "News classifications with labeled LDA". In: *2015 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K)*. Vol. 1. IEEE. 2015, pp. 75–83.

[20] Mark Girolami and Ata Kabán. "On an equivalence between PLSI and LDA". In: *SIGIR*. Vol. 3. 2003, pp. 433–434.