Bangla News Headline Categorization

CSE-837: Machine Learning

Group: VIII

Submitted by
Amran Hossain (BSSE 0917)
Niraj Chaudhary (BSSE 0942)
Zahid Hasan Rifad (BSSE 0944)

Submitted To

Dr. B M Mainul Hossain
Associate Professor
Institute of Information Technology
University of Dhaka

Institute of Information Technology

University of Dhaka

Submission Date:26-04-2021

Contents

List of Figures

Abstract

This project categorizes news headlines from various Bengali newspapers. In addition, it also lets one classify newspapers based on the category they may correspond to national, international, sports etc. There are a lot of works for another linguistic newspapers but we have observed that there is no work for Bengali newspapers. So, we wanted to do this work for Bengali newspapers. We worked for eight categories from the newspapers and completed the classification of news headlines.

Bangla News Headline Categorization

Introduction

Text categorization or classification, is a way of assigning documents to one or more predefined categories. This helps the users to look for information faster by searching only in the categories they want to, rather than searching the entire information space. The importance of classifying text becomes even more apparent, when the information is too big in terms of volume. There is categorization system of news headlines for another languages. But there is no work for Bengali newspaper. So, we built a system for news categorization for Bengali newspapers. In our project, to make this system automatic, classification methods based on machine learning have been introduced. In these techniques, classifiers are built (or trained) with a set of training documents. The trained classifiers are then used to assign documents to their suitable categories. Amongst the vast information available on the web, we chose the domain of news because we observed that the current news websites do not provide efficient search functionality based on specific categories and do not support any kind of visualization to analyze or interpret statistics and trends. The fact that news data is published and referenced on a frequent basis makes the problem even more relevant. This motivated us to build a system keeping two types of users in mind, the first user is the news reader who is interested in browsing news articles based on category and the other is the stakeholder or analyst who is interested in analyzing the statistics to identify past and present patterns in news data. Also, Various news Company want to categorize the news based on published news on newspaper.

Data

Data Collection:

The data was collected from various bangla newspaper with scraping. There is more than one lac data in our dataset. We have collected data various newspapers like Bangladesh pratidin, dainik juganttor, daily inquilab, kalerkantho and so on. We used Chrome Web Scrapper and python languages for scrapping data from websites. There are three columns in our dataset. These are Headlines, category and newspaper_name.

	headline	category	newspaper name
0	মাদারীপুরে নদীতে গোসল করতে গিয়ে শিশুর মৃত্যু	bangladesh	BDProtidin
1	আখাউড়া ইমিগ্রেশন দিয়ে ভারত থেকে আগত ১৪ যাত্রী ...	bangladesh	BDProtidin
2	অবৈধ বিক্রি বন্ধে বালুর স্তূপ নিলামে তুললেন ইউএনও	bangladesh	BDProtidin
3	গরু চুরির অভিযোগে যুবককে নির্যাতন, ইউপি সদস্য ...	bangladesh	BDProtidin
4	নন্দীগ্রামে খালেদা জিয়ার রোগমুক্তি কামনায় মসজি...	bangladesh	BDProtidin
...	...	...	...
103868	ব্লকচেইন নিয়ে সেমিনার করল বিসিএস	it	jugantor
103869	উবারের দক্ষিণ এশিয়ার প্রেসিডেন্ট হলেন প্রভজিৎ সিং	it	jugantor
103870	এই ঈদে বিরাট হাট বসছে 'দেশীগরু বিডিডটকমে'	it	jugantor
103871	প্রবাসীরাও পশু অর্ডার করতে পারবেন অনলাইন হাটে	it	jugantor
103872	বিনামূল্যে প্রশিক্ষণ পাবেন পাঁচ হাজার জন	it	jugantor

103873 rows × 3 columns

Figure 1:Dataset description

The headlines distribution of each categories represents in the following figure. This dataset is an imbalanced dataset.

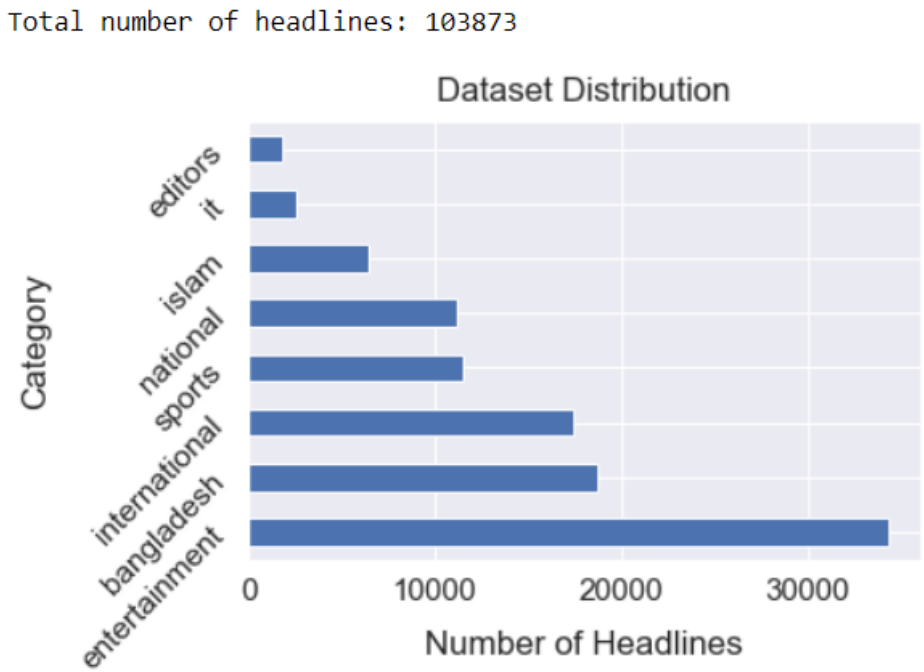

Figure 2: Data distribution

Data Clean:

As the headlines are small in length it is not mandatory to remove the stopwords from the headlines. We use regular expression for remove unnecessary data from our dataset. After cleaning the sample data would look like this.

```
Original:  পদ্মায় ১৪ কোজির রুই, দাম ৩৫ হাজার ৭৫০!
Cleaned: পদ্মায় ১৪ কেজির রুই  দাম ৩৫ হাজার ৭৫০
 Category:-->  bangladesh

Original:  বাংলাদেশের পাখি
Cleaned: বাংলাদেশের পাখি
 Category:-->  editors

Original:  মানবপাচার রোধে বাংলাদেশের জিরো টলারেন্স নীতি গ্রহণ
Cleaned: মানবপাচার রোধে বাংলাদেশের জিরো টলারেন্স নীতি গ্রহণ
 Category:-->  international

Original:  বগুড়ার সংঘর্ষে সাংবাদিকসহ আহত ৭
Cleaned: বগুড়ার সংঘর্ষে সাংবাদিকসহ আহত ৭
 Category:-->  bangladesh

Original:  স্ত্রী-সন্তানসহ করোনায় আক্রান্ত আজিজুল হাকিম
Cleaned: স্ত্রী সন্তানসহ করোনায় আক্রান্ত আজিজুল হাকিম
 Category:-->  entertainment
```

Figure 3: Remove unnncessary symbols

Data summary:

Data summary includes the information about number of documents, words and unique words have in each category class. Also, include the length distribution of the headlines in the dataset.

Class Name : entertainment
Number of Documents:33717
Number of Words:213598
Number of Unique Words:19121
Most Frequent Words:

নতুন	2588
নিয়ে	2288
ও	1802
গান	1381
খান	1074
না	1015
র	1012
নাটক	890
মুক্তি	844
চলচ্চিত্র	838

Figure 4:Data summery for entertainment category

Class Name : bangladesh
Number of Documents:18665
Number of Words:131368
Number of Unique Words:14722
Most Frequent Words:

মৃত্যু	1126
নিহত	1041
গ্রেফতার	976
আটক	891
ও	830
২	686
লাশ	671
উদ্ধার	659
থেকে	622
আহত	615

Figure 5:Data summery for bangladesh category

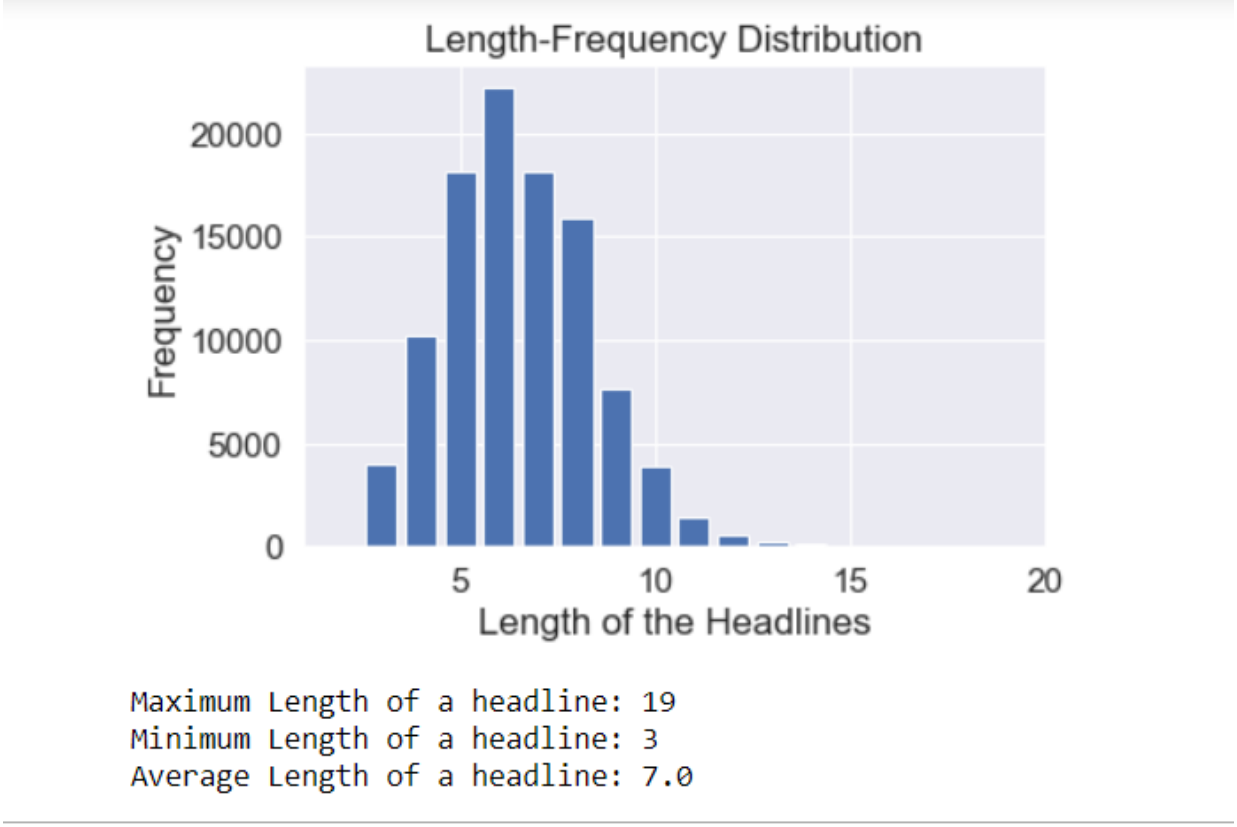

Figure 6:length frequency distribution

From this graphical information we can select the suitable length of headlines we have to use for making every headline into a same length.

From this below graph, we can understand document wise word frequency distribution for each headline. It is a data statistics graphical representation.

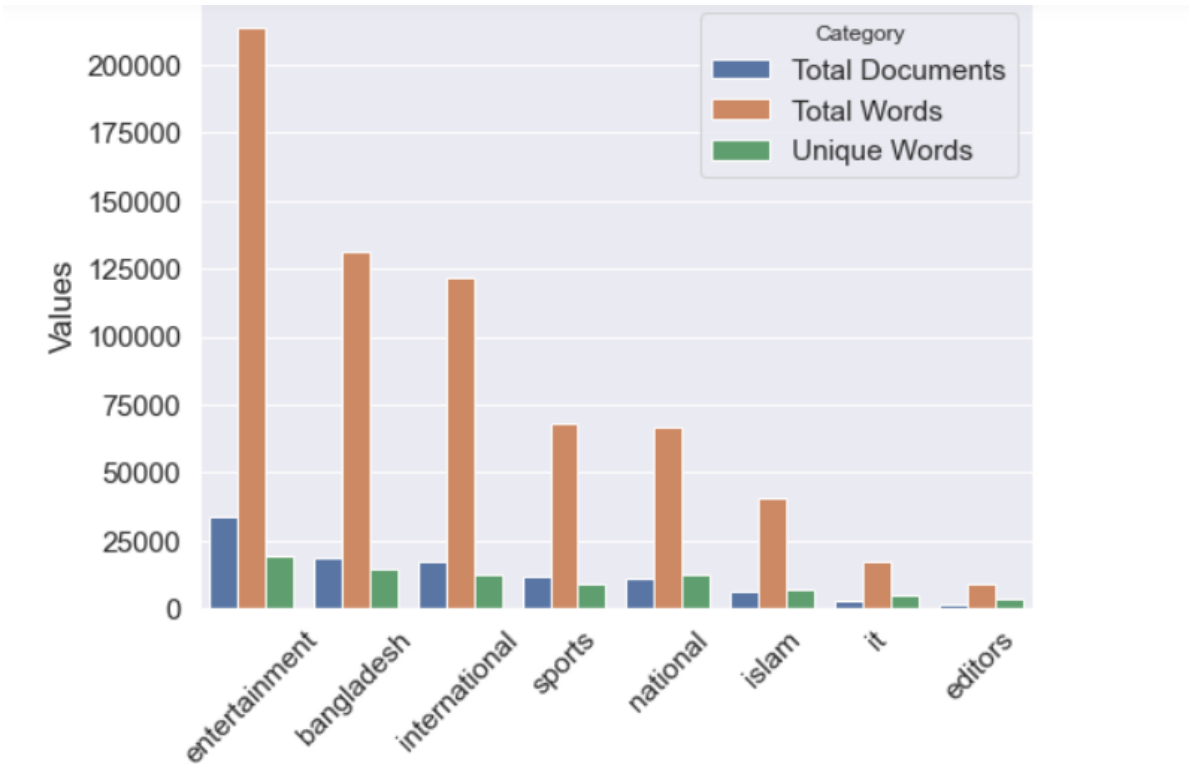

Figure 7: Data statistics summery

Data preparation for model building:

The text data are represented by a encoded sequence where the sequences are the vector of index number of the contains words in each headlines. The categories are also encoded into numeric values. After preparing the headlines.

```
====== Encoded Sequences ======
শেরপুরে কর্মহীন মানুষের মাঝে খাবার বিতরণ
 [43, 36, 110, 108, 17, 16, 7, 37]

====== Paded Sequences ======
শেরপুরে কর্মহীন মানুষের মাঝে খাবার বিতরণ
[ 43  36 110 108  17  16   7  37   0   0   0   0   0   0   0   0   0   0
  0   0   0]
```

Figure 8: Data encoding

And label encoding looks likes belows:

```
                    ===== Label Encoding =====
Class Names:--> ['bangladesh' 'editors' 'entertainment' 'international' 'islam' 'it'
 'national' 'sports']
bangladesh   0

entertainment   2

international   3

bangladesh   0

entertainment   2

entertainment   2

entertainment   2

entertainment   2

entertainment   2
```

Figure 9:label encoding

For model evaluation encoded headlines are splitted into train test validation set. The distribution has

Dataset Distribution:

Set Name	Size
Full	102626
Training	73890
Test	10263
Validation	18473

Figure 10:train test dataset

Methodology

For our project we used two deep learning algorithms. Long short-term memory and Gated recurrent unit. After this result will be compared between these algorithms.

LSTM

Long Short-Term Memory networks – usually just called “LSTMs” – are a special kind of RNN, capable of learning long-term dependencies. They work tremendously well on a large variety of problems, and are now widely used.

LSTMs are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior, not something they struggle to learn!

All recurrent neural networks have the form of a chain of repeating modules of neural network. In standard RNNs, this repeating module will have a very simple structure, such as a single tanh layer.

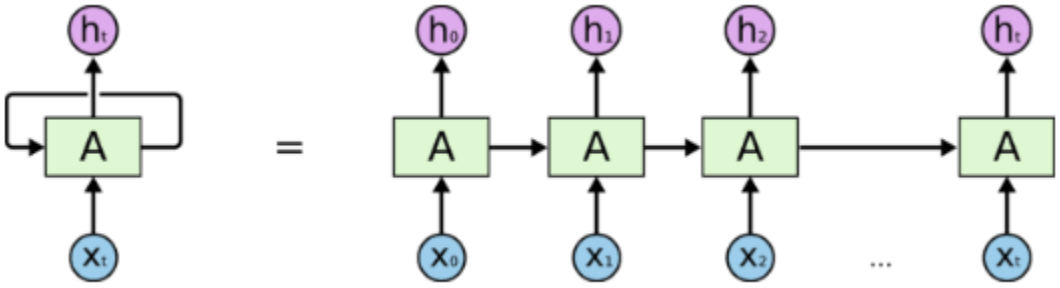

Figure 11:LSTM architecture

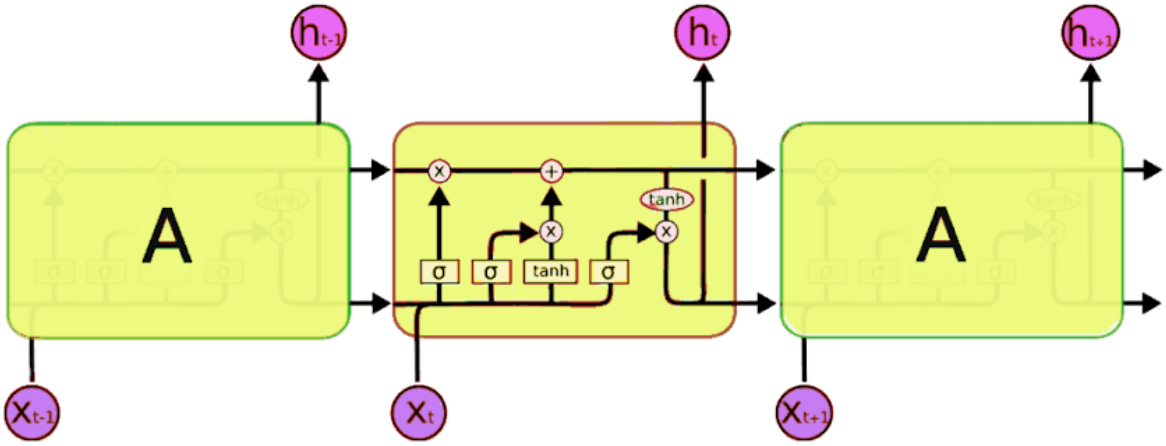

Figure 12: LSTM architecture

The used model architecture consists of a embedding layer (input length =24, embedding dim = 64), LSTM layer (n_unites = 64), two dense layer(n_unites = 24,6) , a dropout and a softmax layer:

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding (Embedding)        (None, 21, 64)            3648000
_________________________________________________________________
bidirectional (Bidirectional (None, 21, 256)           197632
_________________________________________________________________
bidirectional_1 (Bidirection (None, 256)               394240
_________________________________________________________________
dense (Dense)                (None, 24)                6168
_________________________________________________________________
flatten (Flatten)            (None, 24)                0
_________________________________________________________________
dense_1 (Dense)              (None, 6)                 150
=================================================================
Total params: 4,246,190
Trainable params: 4,246,190
Non-trainable params: 0
```

Figure 13:LSTM Model Architecture

GRU

GRU can also be considered as a variation on the LSTM because both are designed similarly and, in some cases, produce equally excellent results.

To solve the vanishing gradient problem of a standard RNN, GRU uses, so-called, **update gate and reset gate**. Basically, these are two vectors which decide what information should be passed to the output. The special thing about them is that they can be trained to keep information from long ago, without washing it through time or remove information which is irrelevant to the prediction.

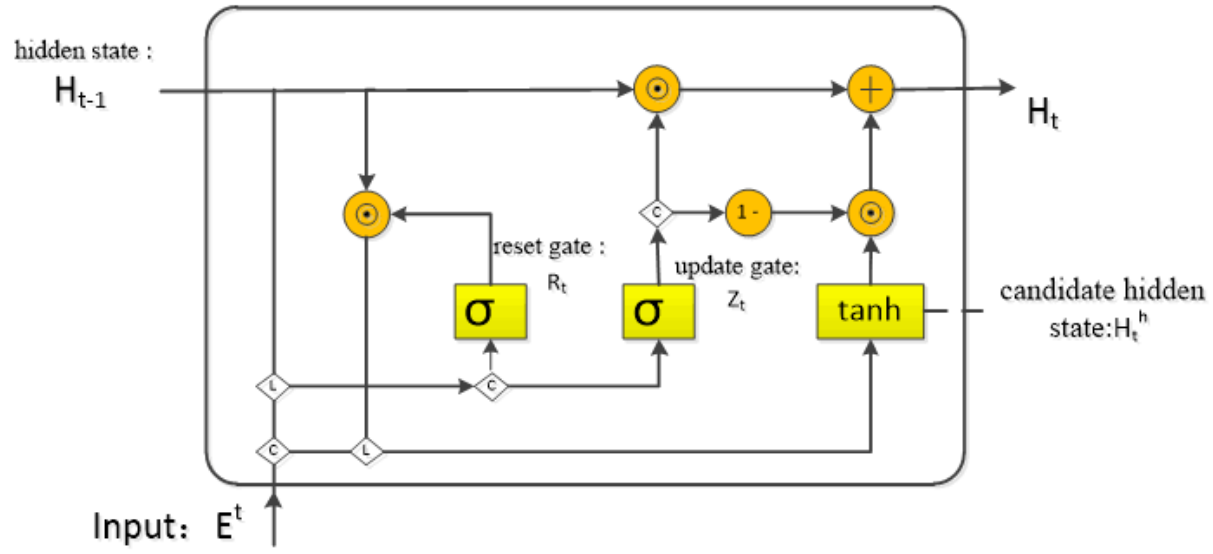

Figure 14:GRU architecture

Update gate:

We start with calculating the **update gate z_t for time step t** using the formula:

$$z_t = \sigma(W^{(z)}x_t + U^{(z)}h_{t-1})$$

Figure 15:update gate

When *x_t* is plugged into the network unit, it is multiplied by its own weight *W(z)*. The same goes for *h_(t-1)* which holds the information for the previous *t-1* units and is multiplied by its own weight *U(z)*. Both results are added together and a sigmoid activation function is applied to squash the result between 0 and 1. The update gate helps the model to determine how much of the past information (from previous time steps) needs to be passed along to the future.

Reset gate:

Essentially, this gate is used from the model to decide how much of the past information to forget**.** To calculate it, we use:

$$r_t = \sigma(W^{(r)}x_t + U^{(r)}h_{t-1})$$

Figure 16:reset gate

This formula is the same as the one for the update gate. The difference comes in the weights and the gate's usage, which will see in a bit.

Current Memory content:

Let's see how exactly the gates will affect the final output. First, we start with the usage of the reset gate. We introduce a new memory content which will use the reset gate to store the relevant information from the past. It is calculated as follows:

$$h_t^{'} = \tanh(Wx_t + r_t \odot Uh_{t-1})$$

Figure 17: current memory content

1. Multiply the input *x_t* with a weight *W* and *h_(t-1)* with a weight *U*.
2. Calculate the Hadamard (element-wise) product between the reset gate *r_t* and *Uh_(t-1)*. That will determine what to remove from the previous time steps. Let's say we have a sentiment analysis problem for determining one's opinion about a book from a review he wrote. The text starts with "This is a fantasy book which illustrates…" and after a couple paragraphs ends with "I didn't quite enjoy the book because I think it captures too many details." To determine the overall level of satisfaction from the book we only need the last part of the review. In that case as the neural network approaches to the end of the text it will learn to assign *r_t* vector close to 0, washing out the past and focusing only on the last sentences.
3. Sum up the results of step 1 and 2.
4. Apply the nonlinear activation function *tanh*.

Final Memory at Current timestamp

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot h_t^{'}$$

Figure 18:final memory at current timestamp

The procedure should be followed by:

1. Apply element-wise multiplication to the update gate *z_t* and *h_(t-1).*
2. Apply element-wise multiplication to *(1-z_t)* and *h'_t.*
3. Sum the results from step 1 and 2.

The used model architecture consists of a embedding layer (input length =24, embedding dim = 64), GRU layer (n_unites = 64), two dense layer (n_unites = 24,6), a dropout and a softmax layer:

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding (Embedding)        (None, 21, 64)            3648000
_________________________________________________________________
bidirectional (Bidirectional (None, 128)               49920
_________________________________________________________________
dense (Dense)                (None, 24)                3096
_________________________________________________________________
flatten (Flatten)            (None, 24)                0
_________________________________________________________________
dense_1 (Dense)              (None, 6)                 150
=================================================================
Total params: 3,701,166
Trainable params: 3,701,166
Non-trainable params: 0
```

Figure 19: architectural model for GRU

Result analysis

LSTM Model

In this simple model we have got 82.74% validation accuracy for such a multiclass imbalanced dataset. Besides Confusion Matrix and other evaluation measures have been taken to determine the effectiveness of the developed model. From the confusion matrix it is observed that the maximum number of misclassified headlines are fall in the category of national, international and editors and it makes senses because these categories headlines are kind of similar in words. The accuracy, precision, recall and f1-score result also demonstrate this issue.

Figure 20: LSTM accuracy

	precision	recall	f1-score	support
Editors	86.92	86.42	86.67	1892.000000
IT	50.00	8.78	14.94	148.000000
International	93.11	96.69	94.87	3328.000000
National	82.23	89.61	85.76	1751.000000
Islam	80.87	64.28	71.62	697.000000
Sports	53.10	63.90	58.00	241.000000
Bangladesh	53.64	61.62	57.35	1089.000000
Entertainment	87.40	70.19	77.86	1117.000000
accuracy	82.74	82.74	82.74	0.827438
macro avg	73.41	67.69	68.38	10263.000000
weighted avg	82.91	82.74	82.37	10263.000000

Figure 21:LSTM precision, recall and f1-score

GRU Model

In this model the accuracy is about 87.48% which is better than LSTM Model. The accuracy, precision, recall and f1-score result also demonstrate this issue.

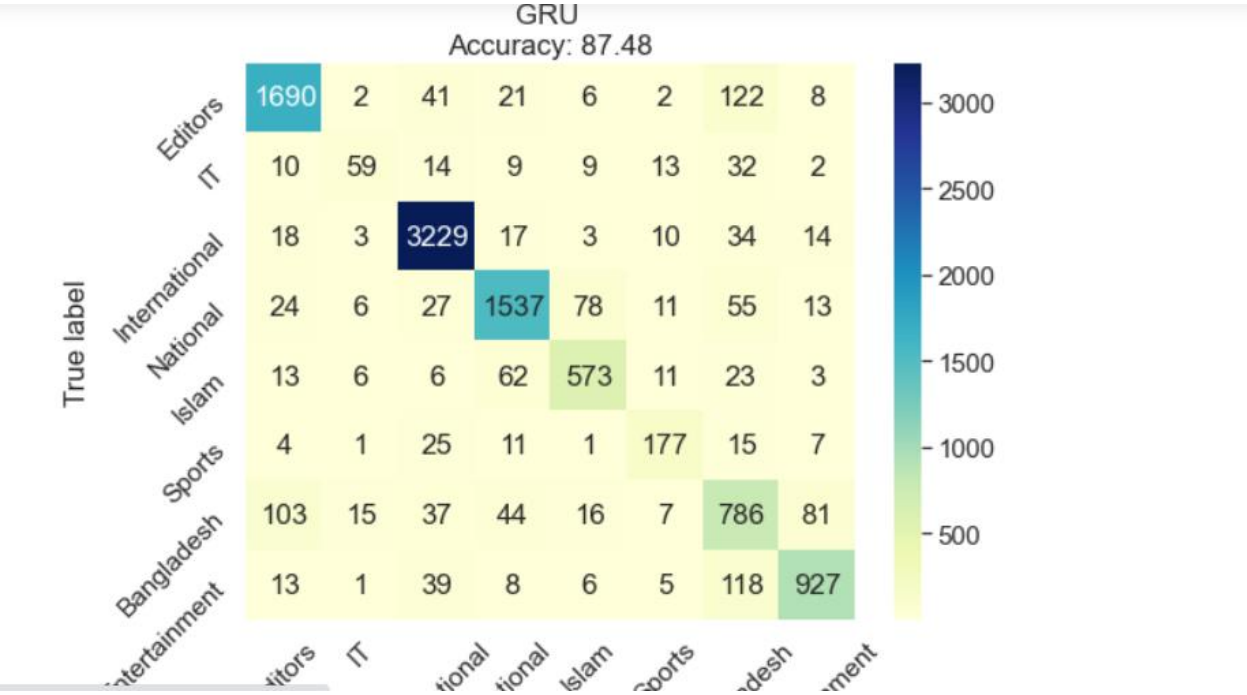

Figure 22: GRU accuracy

	precision	recall	f1-score	support
Editors	90.13	89.32	89.73	1892.000000
IT	63.44	39.86	48.96	148.000000
International	94.47	97.03	95.73	3328.000000
National	89.94	87.78	88.84	1751.000000
Islam	82.80	82.21	82.51	697.000000
Sports	75.00	73.44	74.21	241.000000
Bangladesh	66.33	72.18	69.13	1089.000000
Entertainment	87.87	82.99	85.36	1117.000000
accuracy	87.48	87.48	87.48	0.874793
macro avg	81.25	78.10	79.31	10263.000000
weighted avg	87.50	87.48	87.42	10263.000000

Figure 23: precision, recall, f1-score for GRU

Conclusion

This paper has derived a machine learning based model for News headlines Categorization for Bengali newspaper. Most of the studies in the literature consider for another linguistic newspaper. The paper differs it for Bengali newspaper. GRU is the strongest algorithm for finding good model for this categorization procedure. The findings from the categorizations are mostly consistent with the literature. As we used two algorithms for this classification, we can differ the result from one model to another model. We have taken eight categories for news categorization. The results do not depend on categories. More data, balanced and dissimilar data give a more accurate result for this procedure. Various news Company want to categorize the news based on published news on newspaper. So, they may get their results as they want.

References

1. https://towardsdatascience.com/understanding-gru-networks-2ef37df6c9be
2. https://colah.github.io/posts/2015-08-Understanding-LSTMs/
3. https://realpython.com/beautiful-soup-web-scraper-python/
4. https://schoolofdata.org/handbook/recipes/scraper-extension-for-chrome/
5. https://www.bd-pratidin.com/
6. https://www.kalerkantho.com/
7. https://www.dailyinqilab.com/
8. https://www.jugantor.com/
9. https://www.researchgate.net/publication/221274229_Categorization_of_News_Articles_A_Model_Based_on_Discriminative_Term_Extraction_Method
10. http://www.iosrjournals.org/iosr-jce/papers/Vol18-issue1/Version-3/D018132226.pdf
11. https://www.sciencedirect.com/science/article/pii/S157106610800529X/pdf?md5=faa230893b6acf8688c2228dfb68cdca&pid=1-s2.0-S157106610800529X-main.pdf
12. https://stackoverflow.com/questions/4521426/delete-blank-rows-from-csv