Hybrid Text Summarizer for Bangla Document

Mahimul Islam
Department of Computer Science & Engineering
Ahsanullah University of Science & Technology
mahimulislam@gmail.com

Fariha Nuzhat Majumdar
Department of Computer Science & Engineering
Ahsanullah University of Science & Technology
nivamila@gmail.com

Asadullahhil Galib
Department of Computer Science & Engineering
Ahsanullah University of Science & Technology
agalib.aust@gmail.com

Md Moinul Hoque
Department of Computer Science & Engineering
Ahsanullah University of Science & Technology
moinul@aust.edu

***Abstract*— Automatic text summarization is needed to concisely extract a small subset of text portion from a large text where the isolated text may have sentences that are more significant compared to other sentences in the text. Although there have been a lot of approaches to English text summarization, very few works have been done on automatic Bengali text summarization. For the evaluation purpose, a data set was formulated from the scratch with Bengali news documents from two reputed newspapers. The evaluation data set was classified in four different classes with benchmark standard summary text, generated by a group of random people for each of the documents. The current work presents a hybrid approach for dealing with the summarization process of Bengali text documents. The hybrid model is introduced with a goal to improve the overall accuracy of the summary text generation. The proposed model generates a summary text based on keyword scoring, sentiment analysis, and interconnection of sentences. After conducting the evaluation on the existing data set, the proposed system performs with an average of 0.77 Recall Score, 0.57 Precision Score, and 0.64 F-measure Score. Empirical verification with other similar systems shows that the proposed model can be used as an alternative system to address the Text Summarization problem of Bengali documents.**

***Keywords*—Hybrid Bangla Document Summarization; sentence scoring; sentiment analysis; Keyword Ranking; Text Ranking**

I. Introduction

Text summarization requires a short, accurate, and fluent summary of a longer text document. From the summary, important information can be gained which makes the overall procedure more comfortable, and fewer resources are needed. To discover relevant information faster from a huge number of text documents available online, automatic text summarization ideas have been found very significant. A few methods have been explored for the generation of summary from Bengali documents. If the summary contains sentences from the major topics in the document, it has a better chance of giving a better perspective of the document. The summary generation approach of the proposed system is extractive, i.e., they contain sentences as it appears in the document.

According to Kamal Sarkar [1], text summarization involves preprocessing, stemming, sentence ranking, summary generation. The preprocessing step requires the removal of stopwords, stemming, and converting the input into a collection of sentences. Md. Nizam Uddin and Shakil Akter Khan [2] described an extraction-based method for summarizing Bengali documents. Different features such as location, term frequency, and numerical data, etc. were used to rank the sentences. Based on the features, they have designed their Bengali summarizer and concluded that the summary size should be 40 percent of the actual content. Amitava Das and Sivaji Bandyopadhyay [3] have summarized Bengali documents using sentiment information. They have tried to identify the sentiment information in a document and then aggregated that for generating the summary. Mihalcea [4] [5] has focused on text summarization based on graphs. A graph can be constructed considering the sentences as nodes and connecting them with edges. After that, edge weights may be measured by calculating the similarity between two nodes.

Keeping the state-of-the-art in view, a hybrid Bangla Text Summarizer has been proposed in this work, which combines the following methods:

- Sentiment Scoring
- Keyword Ranking and
- Text Ranking

In the proposed method, the top 40 percent of the actual document was considered as the generated summary based on the combined weighted score which we describe in an upcoming section.

The rest of the paper contains: Related works in section II. The detailed proposed model in section III. Evaluation Measures, Experimental Results, and Comparison with other methods in section IV. Finally, the conclusion and future works in section V.

II. Related Works

In this section, an overview of several types of research relevant to automatic text summarization has been discussed. Most of the research work on text summarization is based on English documents. Despite Bengali being the 7^{th} language according to the number of speakers in the world [6], very few researches were conducted on automatic Bengali text summarization.

Kamal Sarkar [1] [7] has discussed text summarization for a single document of Bengali language signifying the impact of thematic term feature and position feature of sentences. In linguistics, thematic feature means to relate to the theme of writing. The work had mainly three phases such as preprocessing, sentence ranking, and summary generation. Sentence ranking has been done with thematic terms and sentence position. The average unigram-based Recall score is 0.4122 and the score for their baseline is 0.3991.

Kamal Sarkar [8] has presented a key phrase-based approach for summarization which focused on extracting a set of key phrases from a document and generating an extractive summary based on that. Key phrases can be single or multi-word. They have used two different datasets, one for English and another for Bengali. He concluded that the results were quite satisfactory in comparison with the previous works.

Porimol Chandro et al. [9] has experimented extraction-based summarization techniques by collaborating individual word and scoring sentences. Experimentation documents were collected from the popular Bengali daily newspapers. They have done sentence ranking based on Term Frequency, Positional Value, Connecting Words, and Sentence length of the document. Combining these parameters, sentences were ranked, and K top-ranked sentences were picked for the summary. The average unigram-based Precision Recall and F-measure scores were 0.80, 0.67, and 0.72 respectively.

Md. Nizam Uddin and Shakil Akter Khan [2] experimented with Bengali text summarization and have put significance on sentence location, cue phrase presence, title word presence, term frequency, and numerical data. They have argued that sentences which appear in the first or last of passage, are of more importance. Moreover, the presence of cue phrases, words from title, words with high frequency, and numerical data also put importance on a sentence. They have achieved an average accuracy of 71.3 percent.

Amitava Das and Sivaji Bandyopadhyay [3] have summarized Bengali documents using sentiment information. They have used a classifier based on a support vector machine. Three kinds of features have been considered which are lexico-syntactic, syntactic, and discourse level. Parts of speech, SentiWordNet, frequency, stemming, chunk label, dependency parsing depth, the title of a document, first paragraph, term distribution, collocation have been used as features in the work. It has been said that the summarization system has achieved a Precision of 72.15 percent, Recall of 67.32 percent, and F-Measure of 69.65 percent.

Efat et. al [10] have discussed Bengali summarization taking into consideration several attributes. They have calculated a sentence's scores based on frequency, sentence position, cue phrases, etc. After calculating scores based on various aspects, the final sentence scores have been calculated as a weighted summation of the scores of individual features. They have presented that 83.57 percent of summary sentences match to human summaries.

Haque, Pervin, and Begum [11] have done text summarization with Bengali documents using sentence ranking and clustering. Sentences have been ranked with term frequency calculation for each sentence and sentence frequency. If the overlap ratio of two sentences has been shown over or equal to sixty percent, then the smaller sentence falls out of consideration, and importance of larger sentence increases. Sentences have been clustered using cosine similarity to group similar sentences. Then the summary has been generated by selecting sentences from clusters based on volumes of clusters. After evaluation, Precision, Recall, and F-score values have been calculated as 0.608, 0.664, and 0.632 respectively.

Haque, Pervin, and Begum in another work [12] have discussed Bangla summarization using key phrases. They have sorted sentences in ascending order based on their scores and sentences with numerical figures have been given importance. After combining the scores, sentences have been ranked. Data set has been made with four hundred newspaper documents that are of wide varieties. Using ROUGE-1 and ROUGE-2, they said that the quality of their summaries has improved.

Akter et al. [13] proposed an extractive summarization using the K-means clustering algorithm. They have used clustering to tackle both single and multiple document summarization issues. Summary sentences have been picked as the top thirty percent of the input document(s).

Paul et al. [14] have discussed summarization process with sentence ranking based on clustering. The term frequency matrix has been used for summarization. Sentences have been given scores based on four different aspects. Based on term frequency, term frequency, and clustering, TFIDF and clustering, the four methods have been implemented. For experiments, the corpus has been made from newspaper articles. Using ROUGE-N, it has been argued that better results have been found with term frequencies.

III. PROPOSED MODEL

In this work, a text summarizer was developed that generates an extractive summary of Bengali documents. Different scores were given to the sentences based on some criteria to select the best ones representing the gist of a given document. The proposed model contains the following steps:

A. *Input Document*

Any Bengali news document can be used as input of the summarizer. After researching different domains, it was found that normally, people are more interested in reading accident, entertainment, economics, and politics related news. So, the top four categories: Accident, Entertainment, Economics, and Politics were selected to conduct the experiments. The corpus was generated by directly extracting news from the Daily Prothom Alo[1] and the Daily Kaler Kantho[2] newspaper without any modification. About 520 news documents from the mentioned categories were collected. The data sets and news documents are available online [15].

B. *Preprocessing*

Preprocessing involves any type of processing performed on raw data. It was done by splitting the input documents into sentences. Then words were tokenized and after that stop-words

[1] *https://www.prothomalo.com/*

[2] *https://www.kalerkantho.com/*

removal technique was used to remove irrelevant words that are frequently used but don't contribute to generating the summary such as অতএব, অথচ, অথবা, etc. For this purpose, a list containing the stop words were used. The list of stop-words is available online [16]. Two different setups were created where one setup used 50 percent of total document for testing purpose and 50 percent of total document for training purpose and the other setup used 70 and 30 percent of the documents respectively. For each setup from each category, a total of eight different lists were created containing the most frequently used words. The score of each word was calculated by using the following formula.

$$Word\ Score = \frac{Word\ frequency}{No.\ of\ words\ present\ in\ all\ documents} \quad (1)$$

Sample scores for a few words are, 'সড়ক', '0.0044033822', 'গত', '0.0028365346', 'দুর্ঘটনায়', '0.0028365346', 'নিহত', '0.0026474323', etc.

C. Document Type Separation

Document type separation involves separating documents in different categories. Before summarizing, the input document has to be categorized as a specific type. Classification is needed to group similar kinds of news documents. A word can have different weights in different categories. But in a specific category, if a word is more frequently used, it would get more score. As mentioned earlier, four categories of news: such as Accident, Economics, Entertainment, and Politics were collected. So, the input document is categorized in any of the stated document types. To do that, a specific list with words and their frequency scores for each category were prepared. For classifying an unknown document, every word of the document was crosschecked with the lists of all four categories. If a match was found, the corresponding score in that category (the same word can have different score in different category) was summed up. The class with the highest score among those four was selected as the class of the document.

D. Individual Sentence Scoring

Three approaches were combined in a hybrid form to determine the score of each sentence of the given unknown document.

1) Sentiment Scoring: For sentiment score calculation existing library of Python language Polyglot's function Word.Polarity() has been used. PolyGlot is a language pipeline with coverage of a wide range of languages and it supports sentiment analysis for 136 Languages. By using the "Word.Polarity()" function unique word's polarity can be calculated (+1 for positive words, -1 for negative words and 0 for neutral words). A sentence can have scoring based on the neutral words (the words with polarity score 0) present in that sentence. So, in a sentence, more neutral words mean more sentiment score. For example, 'বারবার', 0, 'কেন', 0, 'ডুবছে', 0, 'নৌযান', 0, '?', 0, '"', 0 etc.

2) Keyword Ranking: After analyzing the data set it was determined that key phrases don't put extra value on top of keywords. Because there are minimal number of key phrases in the data set. As already mentioned there were four categories of documents and a list for each category with words and respective frequency scores was prepared (in descending order). E.g., 'সড়ক', 'গত', 'দুর্ঘটনায়', 'নিহত', etc. and their respective frequencies are 163, 105, 105, 98, etc. Each sentence can have a scoring based on each word present in it. The words of each sentence are compared with the list and scores were calculated. The summation of these scores determines the score of that individual sentence.

3) Text Ranking: It is a similarity-based ranking model for text processing which can be used in order to find the most relevant sentences in the text. Figure 1 shows a visual representation of the inter connectivity between sentences.

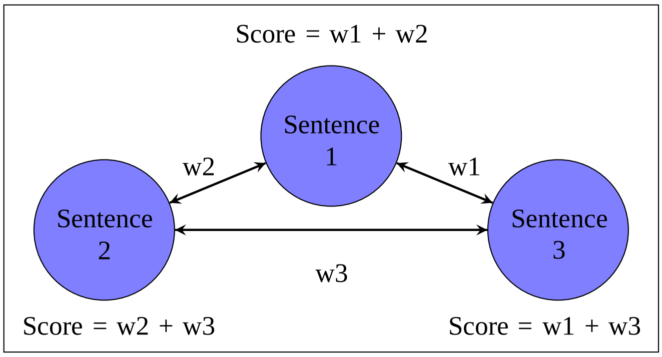

Figure 1. Visual representation of the interconnectivity between sentences.

The method "word2vec" from python's library "gensim.models" was used to find correlated sentences. Text Ranking requires tokenization of each sentence from training data set and converting them into sentence vectors. To achieve this, by using "word2vec" a model was created with the words from each category. Then the model was loaded and using the model each sentence was converted into vector. Then each sentence was compared with all the other sentences present in the text and similarity score with each of them was calculated. The summation of scores define that specific sentence's similarity score. The similarity score between two sentences was calculated using the following formula:

$$Score = \frac{Dot\ Product(Vector1,\ Vector2)}{Normal(Vector1) * Normal(Vector2)} \quad (2)$$

E. Candidate Sentence selection and output generation

The three approaches used, may calculate different scores for a similar sentence because of variation in their scoring model. To select the highest-ranked sentences suggested by three different approaches, weights were multiplied to each of the sentences. The weights for different approach were selected empirically. After the multiplication with weight value, top 40% of the sentences (non-overlapped) were selected for the final generated summary. The summarizer has gone through some repeated validation tests and evaluations by setting different weight values to different methods. Finally, the weights of 0.2, 0.3 and 0.5 were selected for the sentiment scoring method, keyword scoring method and text ranking scoring method respectively.

The scoring of each sentence was done by using the following formula:

$$Sentence\ Score = SS * 0.2 + KR * 0.3 + TR * 0.5 \quad (3)$$

Where in (3),

SS = Total Sentiment based score of that sentence

KR = Total Keyword based score of that sentence
TR = Total Text Rank based score of that sentence

All the sentences of an unknown text document were sorted in descending order, based on their scores. The top 40 percent of total sentences were selected to appear in the summary. Sentences were presented in the same order they appear in the original text to be listed in the summary. The process flow of the proposed model is shown in figure 2.

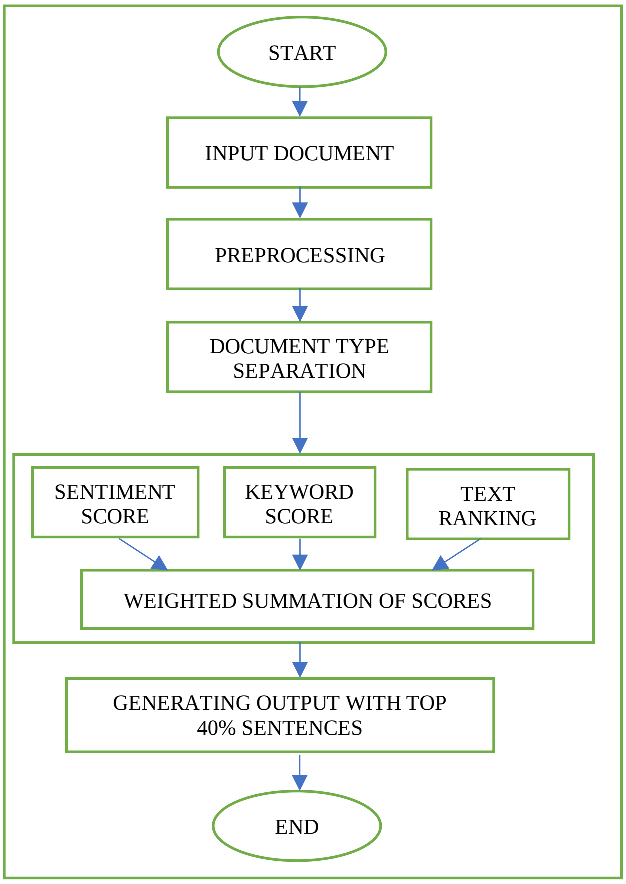

Figure 2. Flow chart of the proposed model for Hybrid summarizer for Bengali Document.

IV. Evaluation Measures & Experimental Verifications

For evaluation purpose, two different data sets were used. In the first phase, a corpus was created from the *Daily Prothom Alo* & the *Daily Kaler Kantho* by extracting 520 online news documents from four different categories of news. Two different setups were used. In the first one, 260 news documents were used for testing and 260 news documents were used for training. In the second one, 364 news documents were used for testing and 156 news documents were used for training. The summaries to be compared with the system generated summary are considered as Benchmark summaries and they were generated by random human contributors. The second data set was collected from the *Bangla Natural Language Processing Community* [17]. This dataset is consisted of two different setups with 100 documents in each. And three model summaries were collected from two groups of scholars to evaluate our proposed system's generated summary. System generated summaries were evaluated with each of the model and the average scores were reported.

A. Evaluation of the first data set

1) Classification Result: Table I and II show the classification results (confusion matrix) of the proposed system for both setups.

TABLE I. Classification Result in 1st Setup

Actual Class	**Predicted Class**			
	Accident	***Economics***	***Entertainment***	***Politics***
Accident	64	1	0	0
Economics	1	60	1	3
Entertainment	5	13	43	4
Politics	6	14	0	45

From table I, it can be seen that for the first setup the overall accuracy of the proposed classifier is (64 + 60 + 43 + 45) / (520 * 0.5) = 0.815 or 81.5 percent.

TABLE II. Classification Result in 2nd Setup

Actual Class	**Predicted Class**			
	Accident	***Economics***	***Entertainment***	***Politics***
Accident	79	9	3	0
Economics	2	86	1	2
Entertainment	17	21	50	3
Politics	8	14	1	68

From table II, it can be seen that for the second setup the overall accuracy of the proposed classifier is (79 + 86 + 50 + 68) / (520 * 0.7) = 0.777 or 77.7 percent.

1) Evaluation of the generated Summary: For the evaluation purpose ROUGE (Recall-Oriented Understudy for Gisting Evaluation) metric was used which is widely used for evaluating the quality of text summarization.

ROUGE-1: It determines the overlap of 1-gram (each word) between the system generated and reference summaries [18].

ROUGE-2: It determines the overlap of bi-grams between the system generated and reference summaries [18]. ROUGE has three main scoring systems, they are Recall, Precision & F-Measure. They can be calculated using the following formulas:

$$Recall = \frac{No.\ of\ overlapping\ words}{No.\ of\ words\ in\ the\ gold\ summary} \quad (4)$$

$$Precision = \frac{No.\ of\ overlapping\ words}{No.\ of\ words\ in\ the\ reference\ summary} \quad (5)$$

$$F-Measure = \frac{2*Precision*Recall}{Precision+Recall} \quad (6)$$

Few empirical experiments were conducted for different setups using our constructed corpus. The evaluation results of first setup are shown in the following tables where about 260 out of 520 documents were tested.

TABLE III. AVERAGE ROUGE-1 SCORES IN 1ST SETUP

Method Type	Scoring Criteria		
	Precision	*Recall*	*F-Measure*
Keyword Ranking	0.6511	0.6469	0.6402
Sentiment Scoring	0.5706	0.7227	0.6266
Text Ranking	0.5743	0.7045	0.6247
Hybrid 1	0.6501	0.7387	0.6842
Hybrid 2	0.6645	0.7357	0.6907
Hybrid 3	0.6636	0.7156	0.6803

TABLE IV. AVERAGE ROUGE-2 SCORES IN 1ST SETUP

Method Type	Scoring Criteria		
	Precision	*Recall*	*F-Measure*
Keyword Ranking	0.5565	0.5592	0.5494
Sentiment Scoring	0.4849	0.6183	0.5327
Text Ranking	0.4706	0.6085	0.5230
Hybrid 1	0.5688	0.6529	0.6005
Hybrid 2	0.5861	0.6584	0.6125
Hybrid 3	0.5802	0.6331	0.5974

The evaluation results of second setup are shown in the following tables where we have used 136 documents (30% of the corpus) out of 520 total documents for the training purpose and a total of 364 (70% of the corpus) out of 520 documents were used for the testing purpose.

TABLE V. AVERAGE ROUGE-1 SCORES IN 2ND SETUP

Method Type	Scoring Criteria		
	Precision	*Recall*	*F-Measure*
Keyword Ranking	0.5203	0.5586	0.5253
Text Ranking	0.4783	0.6321	0.5331
Hybrid 1	0.5124	0.6242	0.5508
Hybrid 2	0.5161	0.6124	0.5480
Hybrid 3	0.5197	0.6005	0.5443

TABLE VI. AVERAGE ROUGE-2 SCORES IN 2ND SETUP

Method Type	Scoring Criteria		
	Precision	*Recall*	*F-Measure*
Keyword Ranking	0.4261	0.4578	0.4308
Text Ranking	0.3808	0.5310	0.4342
Hybrid 1	0.4273	0.5215	0.4600
Hybrid 2	0.4300	0.5120	0.4576
Hybrid 3	0.4320	0.5004	0.4535

In table III, table IV, table V & table VI, hybrid 1 refers to the hybrid system where only keyword ranking (40 percent of the total score) and sentiment score (60 percent of the total score) ranking have been used. Hybrid 2 and Hybrid 3 refer to the hybrid system that combines keyword ranking, sentiment scoring and text ranking methods. In Hybrid 2 system, method weights were set to 30, 20, and 50 percent of the total weights for the keyword ranking, sentiment scoring and text ranking methods respectively. In the Hybrid 3 system, weights were set to 50, 20, 30 percent respectively.

It can be seen that in the 1st Setup, Hybrid 2 system has performed better than all other systems in terms of F-Measure parameter. So, Hybrid 2 system has been used for conducting further comparison experiments.

B. Comparison with existing system models

1) Comparison with existing web-based summarizer: The web-based system of Porimol Chandro et al. [19] has been considered where a total of 260 documents of the manual data set [15] were tested and evaluated with the benchmark summaries.

TABLE VII. COMPARISON OF ROUGE 1 SCORES

Method Type	Scoring Criteria		
	Precision	*Recall*	*F-Measure*
Proposed Hybrid Model 2	**0.6645**	**0.7357**	**0.6907**
Porimol Chandro et al. [19]	0.6440	0.4216	0.4661

TABLE VIII. COMPARISON OF ROUGE 2 SCORES

Method Type	Scoring Criteria		
	Precision	*Recall*	*F-Measure*
Proposed Hybrid Model 2	**0.5861**	**0.6584**	**0.6125**
Porimol Chandro et al. [19]	0.5670	0.3481	0.4019

From table VII & VIII, it can be observed that, the ROUGE-1 and ROUGE-2 scores of our proposed system model are higher than the ROUGE scores of the Porimol Chandro's [19] web-based system.

2) Comparison with other similar systems: In this case, the BNLPC data sets [17] have been considered for the comparison purpose. These data sets have been used previously by Kamal Sarkar [8] and also Haque, Pervin and

Begum [12] in their summarizer. The following tables show the average ROUGE scores of our proposed system and their published systems.

TABLE IX. ROUGE 1 SCORE COMPARISON

Method Type	Scoring Criteria		
	Precision	*Recall*	*F-Measure*
Proposed Hybrid Model 2	**0.5658**	**0.7745**	**0.6487**
Haque, Pervin and Begum 2016 [12]	0.5757	0.6819	0.6166
Kamal Sarkar 2014 [8]	0.5603	0.5515	0.5496

TABLE X. ROUGE 2 SCORE COMPARISON

Method Type	Scoring Criteria		
	Precision	*Recall*	*F-Measure*
Proposed Hybrid Model 2	**0.4958**	**0.7065**	**0.5777**
Haque, Pervin and Begum 2016 [12]	0.5459	0.6433	0.5830
Kamal Sarkar, 2014 [8]	0.5165	0.5075	0.5060

So, from table IX, it can be observed that, in case of ROUGE 1 score, the F-Measure score of the proposed system performs better than all of the previously existed models substantially. From table X, we can see that, our ROUGE-2 Recall score is better than all of the existing systems.

C. Sample Generated Output

An example of a proposed hybrid model 2 generated summary on a news collected from BNLPC data set [17].

আজ রোববার দুপুরে গাংনী উপজেলা পরিষদ চত্বরে ভ্রাম্যমাণ আদালত পরিচালনা করে সহকারী কমিশনার (ভূমি) ও নির্বাহী হাকিম রাহাত মান্নান এ দণ্ড দেন।

আজ এ বিয়ের খবর পেয়ে সহকারী কমিশনার (ভূমি) রাহাত মান্নান বর শরীফুল ইসলাম ও তাঁর শ্বশুর মহিবুল ইসলামকে তাঁর কার্যালয়ে ডেকে পাঠান।

ভ্রাম্যমাণ আদালত সূত্র জানায়, এক মাস আগে গোপনে ব্রজপুর গ্রামের শরিফুলের সঙ্গে মহিবুল ইসলামের দশম শ্রেণি পড়ুয়া মেয়ের বিয়ে হয়।

প্রশাসন গতকাল শনিবার মেহেরপুর জেলা স্টেডিয়ামে গণসমাবেশ করে এই জেলাকে বাল্যবিবাহমুক্ত বলে ঘোষণা দেয়।

সহকারী কমিশনার (ভূমি) রাহাত মান্নান সাংবাদিকদের বলেন, জেলাকে বাল্যবিবাহ মুক্ত রাখার লক্ষ্যে অভিযান অব্যাহত থাকবে।

The subsequent Benchmark Summary collected from BNLPC [17].

মেহেরপুরের গাংনী উপজেলার ব্রজপুর গ্রামে জামাই ও শ্বশুরকে এক মাস করে কারাদণ্ড দিয়েছেন আদালত

ভ্রাম্যমাণ আদালত সূত্র জানায়, এক মাস আগে গোপনে ব্রজপুর গ্রামের শরিফুলের সঙ্গে মহিবুল ইসলামের দশম শ্রেণি পড়ুয়া মেয়ের বিয়ে হয়

সহকারী কমিশনার (ভূমি) রাহাত মান্নান সাংবাদিকদের বলেন, জেলাকে বাল্যবিবাহ মুক্ত রাখার লক্ষ্যে অভিযান অব্যাহত থাকবে

D. Time Analysis for Summary Generation

None of the existing systems performed or presented the time required to generate a summary text. We thought, it will be useful to publish our experimental result to give an idea about time analysis with future directions to improve it.

For our both setups, ten random test documents from each category have been selected for observing the time needed to generate summary for the hybrid summarizer (model 2) and average time per unit time (in seconds) for summary generation has been presented. The configuration used for the testing is as follows: Intel Core i5 Processor, 8 GB Ram and 256 GB SSD.

TABLE XI. TIME NEEDED TO GENERATE SUMMARY

Setup No	Category			
	Accident	***Economics***	***Entertainment***	***Politics***
First	0.603	19. 362	2.416	17.501
Second	22.562	12.158	5.245	22.068

From table XI, it can be seen that, our hybrid system (Model 2) takes a good amount of time for some classes of documents (based on document types and the length of the actual document). So, naturally the process was a bit slow in those cases.

V. CONCLUSION & FUTURE WORKS

Here in this paper, the detailed design and evaluation steps of the proposed model has been discussed. A hybrid method combining three individual ranking systems has been proposed for generating automatic extractive summaries of Bengali documents. A literature review of several related works on automatic Bengali text summarization techniques has been kept in view. Broad explanations of the workflow along with the evaluation results comparing with benchmark summaries for different modules and also comparison with other existing systems have been shown. The time needed to generate summaries for different categories have also been analyzed and listed on tables. After evaluating based on ROUGE-1 and ROUGE-2 evaluation measures the proposed system has shown satisfactory results.

The current work has been presented as an outcome of an ongoing research and in future, our goal is to train the summarizer for many other different categories like literature, international, editorial etc. and provide more importance on sentences that are based on headlines and numerical values. Introducing “Named Entity Recognition” for Bangla text in the system will also be helpful in finding important sentences. Multi-threaded system can be used in order to reduce the summary generation time.

REFERENCES

[1] K. Sarkar, "Bengali text summarization by sentence extraction," in: Proceedings of International Conference on Business and Information Management(ICBIM-2012), 2012, p.p 233-245.

[2] M. N. Uddin, and S. A. Khan, "A study on text summarization techniques and implement few of them for Bangla language," in 2007 10th international conference on computer and information technology, pp. 1--4, IEEE, 2007.

[3] A. Das, and S. Bandyopadhyay, "Topic-based Bengali opinion summarization," in Proceedings of the 23rd International Conference on Computational Linguistics: Posters, pp. 232--240, Association for Computational Linguistics, 2010.

[4] R. Mihalcea and P. Tarau, "TextRank: Bringing Order into Texts," in Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, July 2004, Barcelona, Spain, pp. 404--411.

[5] R. Mihalcea, "Graph-based ranking algorithms for sentence extraction, applied to text summarization," in Proceedings of the ACL Interactive Poster and Demonstration Sessions, pp. 170--173 2004.

[6] "The World Factbook". www.cia.gov. Central Intelligence Agency. Archived from the original on 13 February 2008. Retrieved 21 February 2018.

[7] K. Sarkar, "An approach to summarizing Bengali news documents," in Proceedings of the International Conference on Advances in Computing, Communications and Informatics, pp. 857--862 ACM, 2012.

[8] K. Sarkar, "A keyphrase-based approach to text summarization for English and bengali documents," in International Journal of Technology Diffusion (IJTD), vol. 5, issue 2, pp. 28-38, Apri l 2014.

[9] P. Chandro, M. F. H. Arif, M. M. Rahman, M. S. Siddik, M. S. Rahman, M. A. Rahman, "Automated Bengali Document Summarization by Collaborating Individual Word & Sentence Scoring," in 2018 21st International Conference of Computer and Information Technology (ICCIT), pp. 1--6 , IEEE, 2018.

[10] M. I. A. Efat, M. Ibrahim, and H. Kayesh "Automated Bangla text summarization by sentence scoring and ranking," in 2013 International Conference on Informatics, Electronics and Vision (ICIEV), IEEE, 2013 , pp. 1--5.

[11] M. M. Haque, S. Pervin, and Z. Begum, "Automatic Bengali news documents summarization by introducing sentence frequency and clustering," in 2015 18th International Conference on Computer and Information Technology (ICCIT), pp. 156--160, IEEE, 2015.

[12] M. M. Haque, S. Pervin, and Z. Begum, "Enhancement of keyphrase-based approach of automatic Bangla text summarization," in 2016 IEEE Region 10 Conference (TENCON), pp. 42--46, IEEE, 2016.

[13] S. Akter, A. S. Asa, M. P. Uddin, M. D. Hossain, S. K. Roy, and M. I. Afjal,, "An extractive text summarization technique for Bengali document (s) using K-means clustering algorithm," 2017 IEEE International Conference on Imaging, Vision & Pattern Recognition (icIVPR), pp. 1--6, IEEE, 2017.

[14] A. Paul, M. T. Imtiaz, A. H. Latif, M. Ahmed, F. A. Adnan, R. Khan, I. Kadery, and R. M.Rahman, "Bangla News Summarization," in International Conference on Computational Collective Intelligence, pp. 2884-2889, Springer, 2009.

[15] Mahimul,"Hybrid-Text-Summarizer-For-Bangla-Document," Online available at: https://github.com/mahimulislam/ ResourcesHybridTextSummarization, (last accessed May 2020).

[16] Imran,"Bengali-Sentiment-Analysis," Online available at: https://github. com/Imran-cse/Bengali-Sentiment-Analysis, (last accessed August 2019).

[17] Bangla Natural Language Processing Community, "Dataset for Evaluating Bangla Text Summarization System," Online available at: http: //www.bnlpc.org/research.php, (last accessed August 2019).

[18] C. Y. Lin, E. Hovy, "Automatic evaluation of summaries using n-gram co-occurrence statistics," in Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, pp. 150--157, 2003.

[19] Porimol, "Web Based Bengali Document Summarizer," Online available at: https://bengali-document-summarizer.herokuapp.com, (last accessed May 2020).