A Semi-Automated Approach to Generate Bangla Dataset for Question-Answering and Query-Based Text Summarization

Authors

Mueeze Al Mushabbir, 170041013

Refaat Mohammad Alamgir, 170041031

Ahmed Azaz Humdoon, 170041032

Supervisor

Dr. Md Kamrul Hasan, PhD

Professor, Systems and Software Lab (SSL)

Department of Computer Science and Engineering

A thesis submitted to the Department of Computer Science and Engineering (CSE) in partial fulfillment of the requirements for the degree of B.Sc.

Islamic University of Technology (IUT)
Department of Computer Science and Engineering (CSE))
Academic Year : 2020-2021
April, 2022

Declaration of Authorship

This is to certify that the work presented in this thesis is the outcome of the analysis and experiments carried out by Mueeze Al Mushabbir, Refaat Mohammad Alamgir, and Ahmed Azaz Humdoon under the supervision of Professor Dr. Md Kamrul Hasan, PhD, Professor, Department of Computer Science and Engineering (CSE), Islamic University of Technology (IUT). It is also declared that neither of this thesis nor any part of this thesis has been submitted anywhere else for any degree or diploma. Information derived from the published and unpublished work of others has been acknowledged in the text and a list of references is given.

Mueeze

Mueeze Al Mushabbir
170041013

Refaat

Refaat Mohammad Alamgir
170041031

Ahmed Azaz Humdoon
170041032

Md. Kamrul Hasan

Supervisor
Dr. Md Kamrul Hasan, PhD
Professor, Systems and Software Lab (SSL)
Department of Computer Science and Engineering

Acknowledgements

We would like to express our grateful appreciation for **Dr. Md Kamrul Hasan, PhD**, Professor, Department of Computer Science & Engineering, IUT and **Dr. Hasan Mahmud, PhD**, Assistant Professor, Department of Computer Science & Engineering, IUT for being our mentors. Their motivation, suggestions and insights for this research have been invaluable. Without their support and proper guidance this research would never have been possible. Their valuable opinion, time and input provided throughout the thesis work, from first phase of thesis topics introduction, subject selection, proposing algorithm, modification till the project implementation and finalization which helped us to do our thesis work in proper way. We are really grateful to them.

Abstract

With the vast amount of information available on the Internet, finding answers to questions is as important as ever in today's day and age. In Natural Language Processing Research, Question Answering (QA) and Query-based Text Summarization (QBSUM) are there to tackle this challenge. However, most of the work being done neglects low resource languages such as Bangla, resulting in the small number of quality datasets available in the literature. Therefore to address this research gap, in this work, we propose a semi-automated methodology for generating a Bangla dataset with Natural Questions for three tasks - Question Answering (QA), Query-based Single Document Text Summarization (SD-QBSUM) and Query-based Multi-Document Text Summarization (MD-QBSUM). We then provide baselines for this dataset on those tasks and also compare our dataset with existing ones on various metrics.

Keywords

Question Answering, Query Based Single Document Summarization, Query Based Multi-Document Summarization, Semi-Automatic Approach, Semi-Supervised Method, Natural Question, mT5, mBERT, SBERT.

Contents

Figures

Tables

Chapter 1

Introduction

In the advent and progress in the domain Natural Language Processing (NLP) and Deep Learning (DL) around the world, tasks like Question-Answering (QA) and Query-based Summarization (QBSUM) have gained a lot of traction and popularity in recent years. While they are indeed two of the most popular and important tasks of current era, a closer look in the domain shows that majority of the work has mostly been done only in English language and not in low-resource languages like Bangla, which is the very reason that inspired us to take up on the research work in the domain of QA and QBSUM in Bangla.

In this chapter we will take a brief walk through the contents of the thesis and the report. This will include contents from Motivation to Problem Statement to Thesis goals and objectives and Thesis contribution.

1.1 Motivation

The research gap that we see in Bangla NLP research domain, i.e. scarcity of Bangla corpus, motivates us to create quality and standard Bangla datasets in these domains. The available datasets are lacking in quality. The reason for this is that due to being Machine-translated, semantic relation and coherency in the context (document) is lost. We hope to facilitate various NLP tasks in Bangla, including Multilingual QA, Summarization, Chatbot and Reading Comprehension.

1.2 Problem in the Domain

In the NLP Research Domain, there is a significant lack of available datasets in Bangla for the tasks of Question Answering and Query Based Summarization. The small amount of datasets that do exist for QA, again lack proper quality to be useful. Whereas, there does not exist even a single dataset for the task Query based Single Document Summarization and Query based Multi Document Summarization in Bangla.

Thus, Our problem statement becomes as follows:

”Building a semi-automatic pipeline for creating 3 datasets on account of Natural Queries in order to improve and facilitate tasks of question-answering, query-based single and multi-document summarization.”

1.3 Research Scope and Our Goal

From the background study, we find that the tasks of QA and QBSUM are indeed two of the most popular tasks in the research community. However, this is true only for English language where there are abundance of documents, corpus, and datasets available. Compared to English, our native language Bangla lacks resources tremendously, so much so that there are only limited number of datasets available in the domain. In this era of prominent research work of NLP, this absence of proper dataset produces a great disadvantage and can be considered as a significant scope for research work.

As a result, we form our research goal to fulfill the gap and accelerate the research work in Bangla QA and QBSUM by creating 3 datasets of good quality for each of the 3 sub-tasks of the domain, namely - Question-Answering (QA), Single Document Query-based Summarization (SD-QBSUM), and Multi-Document Query-based Summarization (MD-QBSUM).

1.4 Objective

In order to achieve our research goal, we formulate the following objectives:

1. Improving the quality of Question-Answer Dataset in Bangla, because the existing ones lack quality in terms of either corpus size, annotation quality, or semantic relationship in the context.
2. Working on the Natural queries from real users of Google search engine [1], because these are the questions that general people usually have.
3. Proposing a semi-automated pipeline for creating a dataset on QA, SD-QBSUM, or MD-QBSUM.
4. Introducing Single Document Query-based Summarization (QBSUM) in Bangla, because to the best of our knowledge, there exists no such dataset in Bangla yet.
5. Extending Single Document to Multi-Document QBSUM Dataset in Bangla, because Multi-document QBSUM dataset is also absent in Bangla, to the best of our knowledge.

1.5 Research Challenges

After doing an extensive literature review and looking at the related work, we have identified some key research challenges that we are likely to encounter. Firstly, in contrast to many other languages, Bangla has relatively smaller amount of source documents on the Internet, which makes it difficult to gather quality and resourceful documents in Bangla.

Furthermore, to the best of our knowledge, no open source natural questions, which is an integral part of the QA and QBSUM datasets, exists for Bangla. Collecting and cleaning the raw data also pose a great challenge in the work. We need to bear in mind that since it will be a semi-automatic process of dataset creation, we have to be aware of and account for any coherence shortcomings of machine translation (e.g. when translating the questions of Google NQ Dataset)

There will be also be some hardware challenges as the data extraction tasks (for curating the final datasets) and the deep learning models (for benchmarking the final datasets) will require large storage space and sufficient GPU power. Integrating the development of three datasets for three tasks into one single process is a challenging process as well. Finally, the tradeoff between Human annotation (which will yield higher quality dataset) and synthetic data (which will speed up the process) must be taken care of.

1.6 Thesis Contribution

In order to create the 3 datasets in Bangla, we first tackle the issue of collecting Natural Questions in Bangla. Since there are no existing opensource datasets in Bangla that provides this type of questions, we take help of Google Natural Questions Dataset [1] and translate the questions from English to Bangla with Human Verification Process (shown in Figure 3.1), also known as Human-in-the-Loop (HITL). Then, to build a Bangla document corpus, we collect Bangla document (or context) from Wikipedia corresponding to the natural questions extracted before. Finally, as annotation process, we again introduce HITL to manually annotate the data and ensure quality.

The main contributions of this thesis work are as follows:

- We propose a pipeline that combines the creation of 3 datasets
- We provide 3 datasets which tackles three tasks in QA and QBSUM domain
- We create a Multi-Document Corpus set that is similar to a Wikipedia corpus set
- we incorporate a semi-supervised approach to annotate data in a semi-automatic manner
- we introduce HITL (Human-in-the-loop) to ensure high quality of the dataset
- We make Natural Questions available in Bangla

1.7 Thesis Outline

Moving forward, chapter 2 contains our background study and literature review. Chpater 3 describes our proposed methodology, including justification of our methodology. Chapter 4 notes our steps to create the dataset. Chapter 5 details out our experiments and experimental setup. In Chapter 6 we show detailed analysis of our dataset and discussions. Lastly, chapter 7 includes our plans for future work.

Chapter 2

Background Study and Literature Review

To understand the research scope in the domain of QA and QBSUM, we must first understand the concepts that lie in the background of the thesis. We here describe concepts of Question Answering, Query Based Summarization to anything that is required to know to understand this thesis work.

Additionally, we also show our literature review and any work that is related. This includes: Bengali SQuAD, TyDi QA, AQUAMUSE for our purpose.

2.1 Background Study

Question-Answering

The task of Question-Answering (QA) is also known as open-domain question-answering. A major source of document, which is also known as Contenxt in the literature, in this case is Wikipedia [1]. In the task of QA, a context (document, article, or a passage) is given along with a question, and the machine in turn tries to find an answer to that question from the provided context. In this case, the answers are usually short, precise, and contains only a few words maximum. A diagrammatic example can be given in Figure 2.1:

[1] www.wikipedia.org

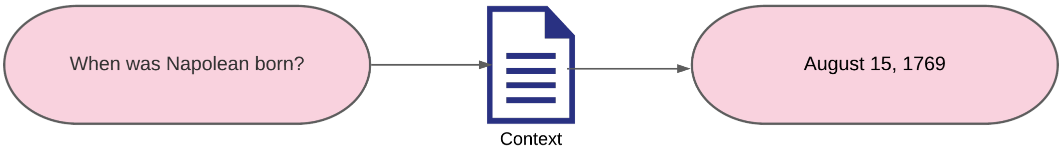


Figure 2.1: Given a question and a context (document) QA finds a very short answer

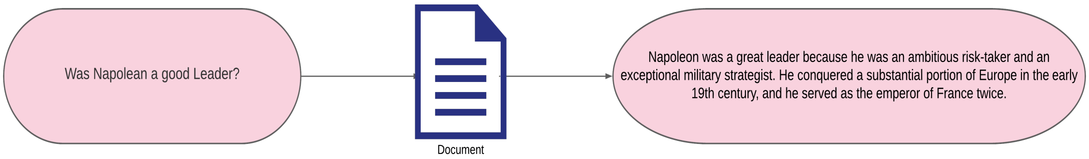

Was Napolean a good Leader?
Document
Napoleon was a great leader because he was an ambitious risk-taker and an exceptional military strategist. He conquered a substantial portion of Europe in the early 19th century, and he served as the emperor of France twice.

Figure 2.2: Given a question and a single document as context SD-QBSUM finds a summarized answer

Query-Based Summarization

In contrast to QA, Query-Based Summarization (QBSUM) tries to figure out a summarized version of the answer to satisfy the query (or question). In addition, based on the number of documents in the context, QBSUM can be of 2 category:

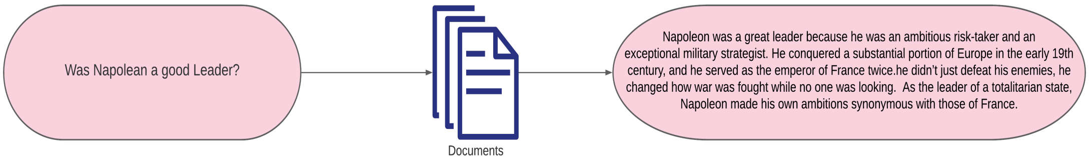
1. **Single document QBSUM (SD-QBSUM):** provided context includes a single document, shown in Figure 2.2

2. **Multi-document QBSUM (MD-QBSUM):** provided context includes multiple documents, shown in Figure 2.3

Here, as well, a context (single document or multi-document) and a query (or question) is given as input, and the machine tries to provide a summary focusing the information related to the query. In this case, the queries tend to be more open-ended and the query-based summarized answers are usually longer and have to be derived from parts of the context.

2.1.1 Dataset Characteristics

In the research domain of NLP, the datasets for Question Answering and Query Based Summarization yield a specific trend of characteristics. The usual charac-



Napoleon was a great leader because he was an ambitious risk-taker and an exceptional military strategist. He conquered a substantial portion of Europe in the early 19th century, and he served as the emperor of France twice.he didn't just defeat his enemies, he changed how war was fought while no one was looking. As the leader of a totalitarian state, Napoleon made his own ambitions synonymous with those of France.

Figure 2.3: Given a question and multiple documents as context MD-QBSUM finds a summarized answer

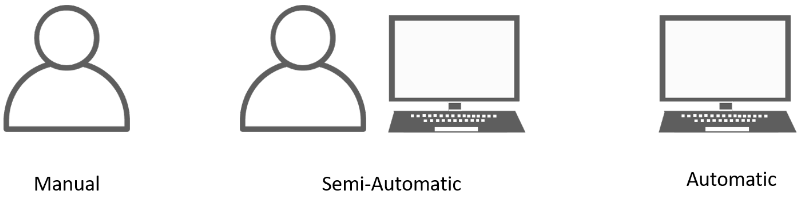


Figure 2.4: Approaches for Creating a Dataset

teristics of the datasets contain, context - a passage or article, question(s) - one or more relevant or non-relevant questions to the context, answer text - a textual answer or null if there is no answer, Tokens or Bytes - denoting answer span.

Approaches of Creating Datasets

Related to the work of QA and QBSUM in Bangla, a major factor that needs contribution is Datasets, since there are only a few of them that exist. In order to contribute in the work of creating dataset, the well-accepted approaches can be:

- **Automatic:** an automated process run only by computer program(s) without human intervention at any point.

- **Semi-Automatic:** A computerized automated process that includes **Human-in-the-loop (HITL)** approach at a point. Part of the process is done automatically and other part is done by manual operations.

- **Manual:** The whole process is done manually by humans without the use of any automated computer process.

Types of Questions

In the domain of QA and QBSUM, the available questions can be categorized in several ways. But depending on how the questions are generated, it can be divided into 2 major categories, that is:

- **Synthetic Questions:** Questions that are generated by machines or by some people who are creating a dataset. This does not reflect questions that the general public may have.
- **Natural Questions:** Questions that real people have (usually retrieved from stored search engine queries) [1]. Any sort of search engines are usually a good source for collecting this type questions, because general people from all aspects surf the internet all throughout the day and search various queries on various topics. For our work, we take such kind of queries from Google's research dataset Google Natural Questions.

2.1.2 Semi-Supervised Annotation

Semi-Supervised approach combines supervised and unsupervised approach. At first, a portion of the data is manually/automatically annotated with proper labeling without noise. With this labeled data, a model is first trained and then inferred on the unlabeled data.

Then from this inference, a random sample is taken and added to the labeled dataset. This new portion of labeled dataset is then used to train the model again and infer and add the new labeld data. Thus this loop goes on and labelling is performed.

2.2 Related works and Literature Review

2.2.1 Bengali SQuAD

The key contribution in this paper is creating a QA dataset in Bangla and this was done by machine-translating the most well known QA dataset that exists in the literature, i.e. SQuAD 2.0 [2] The authors of the Bengali SQuAD paper provide

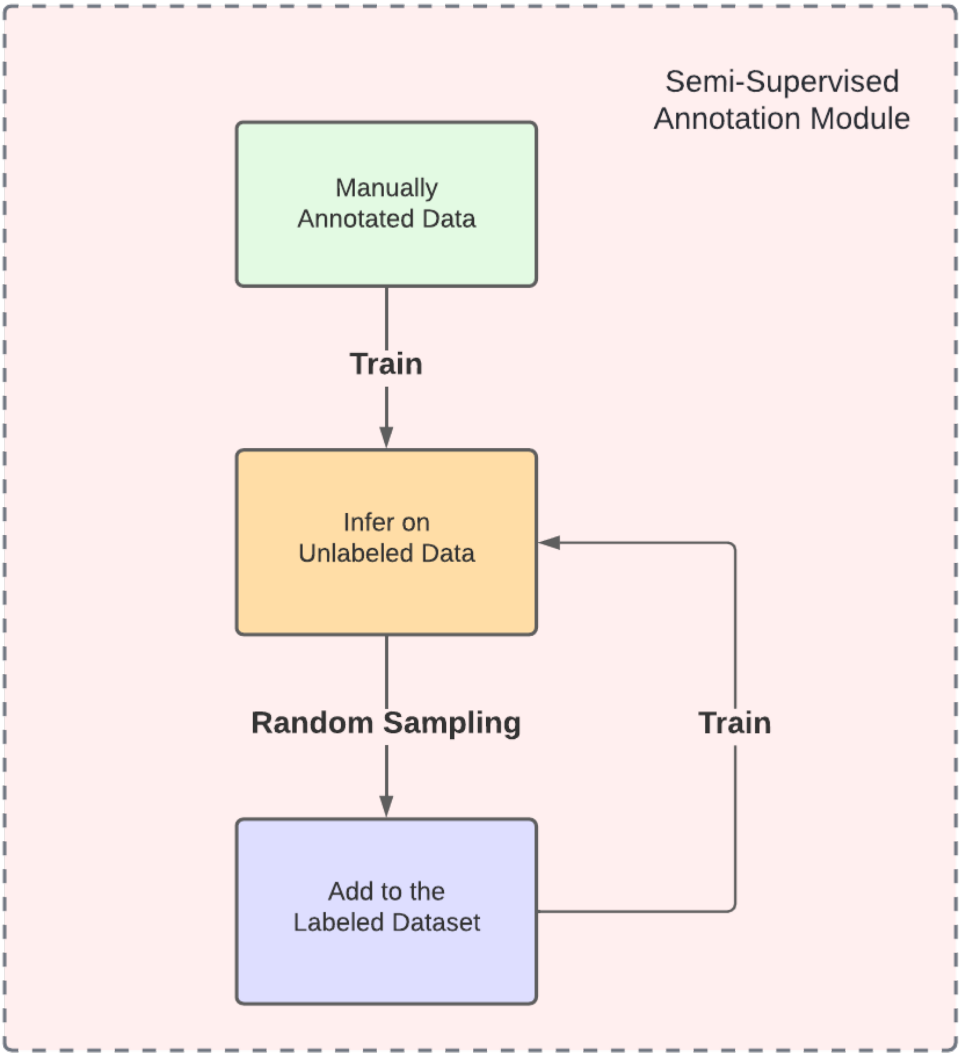
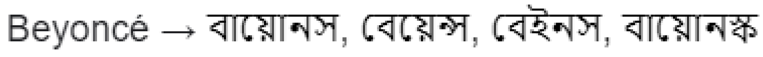
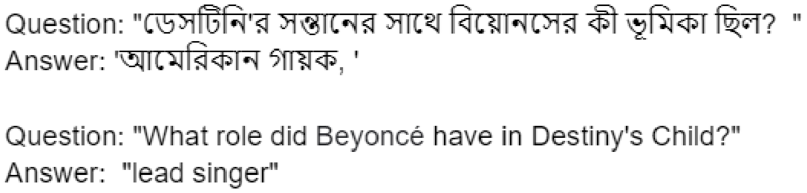


Figure 2.5: Semi-Supervised Annotation Module



Beyoncé → বায়োনস, বেয়েন্স, বেইনস, বায়োনস্ক

Figure 2.6: Inconsistent spellings



Question: "ডেসটিনি'র সন্তানের সাথে বিয়োনসের কী ভূমিকা ছিল? "
Answer: 'আমেরিকান গায়ক, '

Question: "What role did Beyoncé have in Destiny's Child?"
Answer: "lead singer"

Figure 2.7: Wrong answer

benchmarking scores by applying some pre-trained deep learning models on their dataset and finally proceed to compare these models' performances with that of a survey conducted on children. [3] Despite the valiant effort that went into creating this, we have identified some issues in their dataset which we hope to address in our work.

Most of the problems were related to their translation. We have observed that the answers sometimes fail to recognize named entities and even when they do, spellings are not always kept consistent across the whole dataset as shown in Figure 2.6 Unnatural and semantically incoherent translations such as the translated question in Figure 2.7 were also seen. In the same figure, we can also notice that the target label was wrongly annotated. And lastly, in other cases, some answers were completely missed and therefore those were empty while their English counterparts were not.

Most importantly, they have not made their full dataset public which made it difficult to do an extensive analysis as only some samples were publicly available.

2.2.2 AQUAMUSE

The paper AQUAMUSE proposes a pipeline for automatically generating datasets for both abstractive and extractive multi-document QBSUM. [4]

As shown in the Figure 2.8, long answers are taken from Google NQ dataset

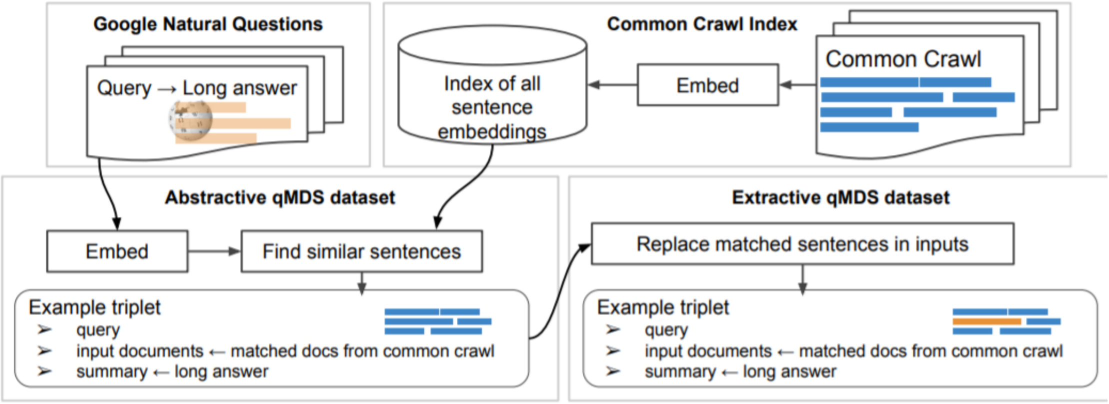


Figure 2.8: AQUAMUSE pipeline for generating conjugate abstractive and extractive query based multi-document summarization datasets

and are considered to be the summaries. These summaries are then matched with embedded document sentences collected from a cleaned version of Common Crawl known as the Colossal Clean Crawled Corpus. [5] The matching is done slightly differently for the two types of query-based text summarization. For abstractive summaries, semantic relevance of the summaries with the document sentences are found and exact matches are avoided through the use of special parameters. On the other hand, for extractive summaries, in-place substitution of the summaries with the document sentences is done, which results in the best matched sentence in the document to be replaced with a sentence from the summary.

Even though this pipeline is structured and well laid out, one will run into a brick wall if they are to apply the same technique for Bangla. One of the key reasons for this is that Google NQ does not have Bangla dataset as of yet and the corresponding Wikipedia articles for the NQ dataset may not be similar in length, content or even exist at all. Secondly, annotation is done in a discriminative fashion (which means that summaries already exist as long answers and documents are checked with these summaries to determine the semantic relevance) rather than in a generative mode (where summaries are written either by experts or by machines). Furthermore, as already stated, the long answers from Wikipedia are kept as the overall summarized answer, which means that the answer only reflects Wikipedia content and not other multiple relevant documents.

2.2.3 TyDi QA

TyDi QA is a dataset for question answering task. It includes data in 11 languages that are, according to them, ”typologically” different. [6]

They propose their dataset for a couple of tasks - Primary and Secondary. [6] In primary task, the core task is to identify a passage or a minimum span of words that includes the answer, or return no answer if there are not any. Conversely, the secondary task, also known as Gold Passage task, guarantees that there are answers and the task is to find them. [2]

In 11 languages, there are around 204K data sample in total. But for Bangla, there are only around 10k for primary task and around 2k for secondary task.

2.3 Models

There are several machine and deep learning models that we used in our work. They are majorly Transformer- or BERT-based. Descriptions about these models and the reason as to why they were chosen are given in the later chapter of Experiments, in the section of Baseline Models.

[2] https://github.com/google-research-datasets/tydiqa

Chapter 3

Proposed Methodology

Our idea is a single process to generate datasets for three different tasks - QA, QBSUM-SD, and QBSUM-MD. The whole process can further be thought of as two sub-process as explained in the following sections.

3.1 Single Document summarization

As illustrated in Figure 3.1 , we first extract some elements from the Google NQ Dataset - Questions (in English), Answers, Wikipedia Articles and their URLs. Among these elements, only the questions and URLs are needed in the next step so we ignore the rest. The questions will be translated to Bangla and this will be a HITL(Human-in-the-loop) process, i.e. the translations will be done by machine and supervised by a human. On the other side, we retrieve the corresponding Bangla Wikipedia articles if they exist. Finally, we will use these Wikipedia articles and provide them to the Semi-Supervised Module where there will be labelling performed automatically and the first set of true labelling will be annotated by human annotators who will mark the short answers (which will be necessary for Question Answering dataset) and long answers (which will be used to create the Query-based Text Summarization Single Document dataset).

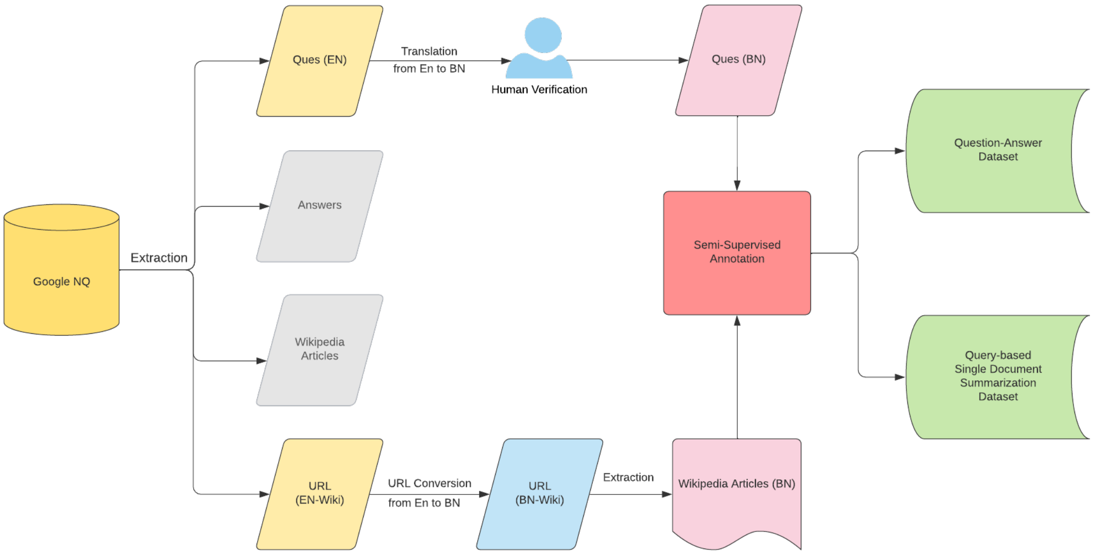


Figure 3.1: Proposed Methodology - Single Document QA + QBSUM

3.2 Multi-Document summarization

The method for generating dataset for QBSUM for Multi-Document is quite similar to that for QA and QBSUM-SD with a few key differences as shown in Figure 3.2. In the previous sub-process where the corresponding Bangla Wikipedia articles are retrieved, those documents are matched with different documents from Common Crawl and Banglapedia, using cosine similarity. Thus, we will now have multiple documents which will be the context for QBSUM-MD. On the other side, we will also have the questions translated with HITL as before. Finally, these questions and multiple documents will be provided to the Semi-Supervised Module for annotation. In this module, there will be annotation done automatically based on the first set of true labelling that is done by human annotators, who will find the answers to these questions, thereby completing the creation of the dataset.

3.3 Semi-Supervised Annotation Module

As mentioned in the previous chapter 2, we adopt a semi-supervised approach to annotate the data. this way there is portion of the data that is manually annotated

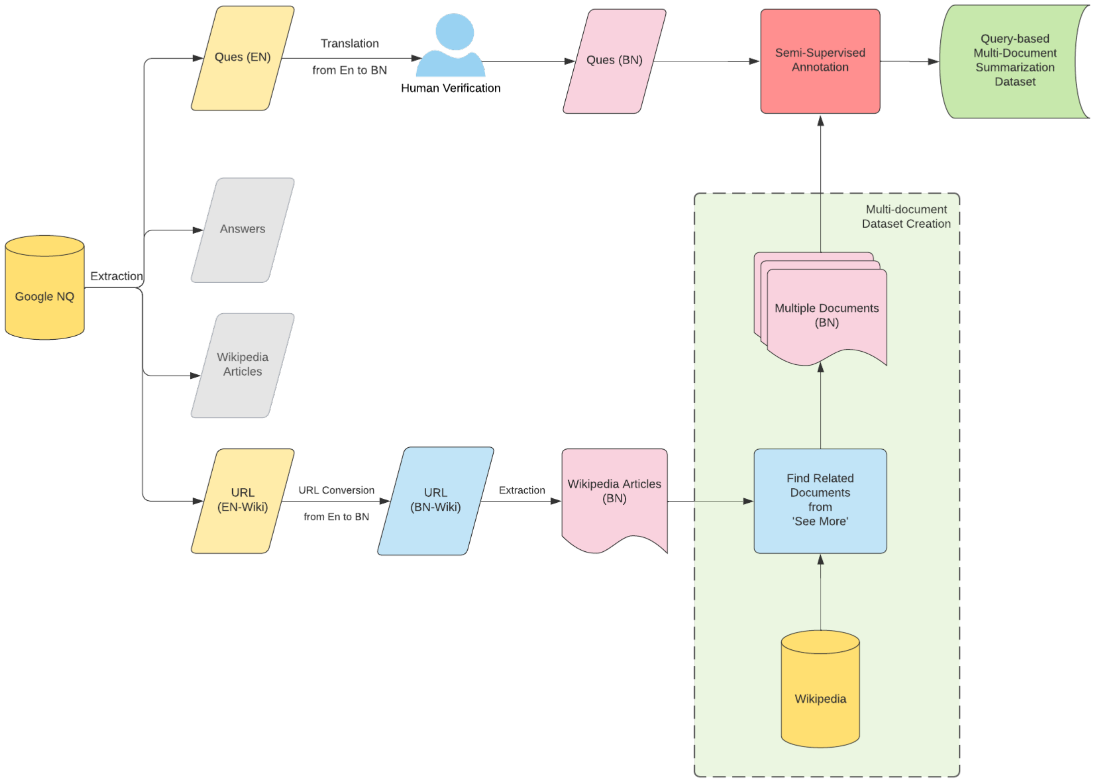


Figure 3.2: Proposed Methodology - Multi-Document QBSUM

by human annotators. And the rest of the annotation comes from training the model and inferring on the unlabeled data.

Chapter 4

Dataset Creation

Our Dataset creation is followed as shown in the previous chapter diagrams 3.1 and 3.2.

4.1 Data Extraction from Google NQ

In Google NQ, it consists of English data. The attributes of each data are: URL, Question, Context, Answer, Answer Tokens/Bytes. There are 2 types of dataset provided by Google NQ - All data, where there are html document and token and Simplified, where there are no html document or tokens, simply text of the corpus is present. We collect data from the non-simplified version which is of size around 42GB.

4.2 Bangla Wikipedia Page Scrape and Cleaning

From the data samples collected from Google NQ, we go through each of them, and check if there exists any URL for Bangla Wikipedia page of the same topic from the corresponding URL of the English Wikipedia page from Google NQ. If the URL is valid, we collect the contents using BeautifulSoup.

With BeautifulSoup, we clean the collected content. There might be speacial characters and equations, however those documents were collected using their article id and wikipedia python library in an automated way. There is another challenge, that is cleaning table. Since table data are very complex and parsing is also difficult, we used Python Pandas library. It reads the table html and generates a DataFrame. From there, we can gather the texts and add to the scraped and cleaned data.

4.3 English Question Translation

The data we collected from Google NQ contains questions in English language. We need to translate them to Bangla. So we use the Google Translation API, and translate all the questions to Bangla. Afterwards, to verify the data, we apply a manual human verification, who checks whether the questions translated are valid or not, and also fixes if it is not. This is one point where we adopted a Human In the Loop (HITL) approach.

4.4 Annotation

From the wikipedia bangla pages scraped and the questions translated to bangla, we now apply Semi-Supervised Annotation Module here. In this module, there will be a human annotator who will first annotate manually some data. These are considered as gold samples because the human annotator annotation process is considered rarely noisy.

After human annotators annotated manually, the labeled data are used to train the model and infer on unlabeled data. Then from those predicted labels, a portion of sample is taken randomly and added to the labeled data. Later, this labled dataset with newly predicted labels are fed to the model and trained again, and it goes in such fashion.

And Thus We obtain our dataset.

4.5 Multi-Document Corpus Collection

Our initial approach was to find similar documents based on similarity measures like shown in AQUAMUSE. However, we find that the encoder used in the AQUA-

MUSE - Universal Sentence Encoder, does not perform well in Bangla Language. Also, the number of documents contained in the C4 dataset is over 7 million. Finding similar documents in this vast pool of documents is not resourcefully feasible. So, we adopt to a different approach.

In the newer approach, as shown in the figure 3.2, we extract URLs of the articles related to the given article from the bottom of the page, where there is a section 'related articles'. In this section, the article itself notes to other articles that are related to this one. So, we collect them and attach to the given document, and thus create a data sample with set of related documents .

Chapter 5

Experiments

5.1 Exerimental Setup

In this chapter, we talk about our experiments and the experimental setups. It includes setup from hardware, software to Running Experiments.

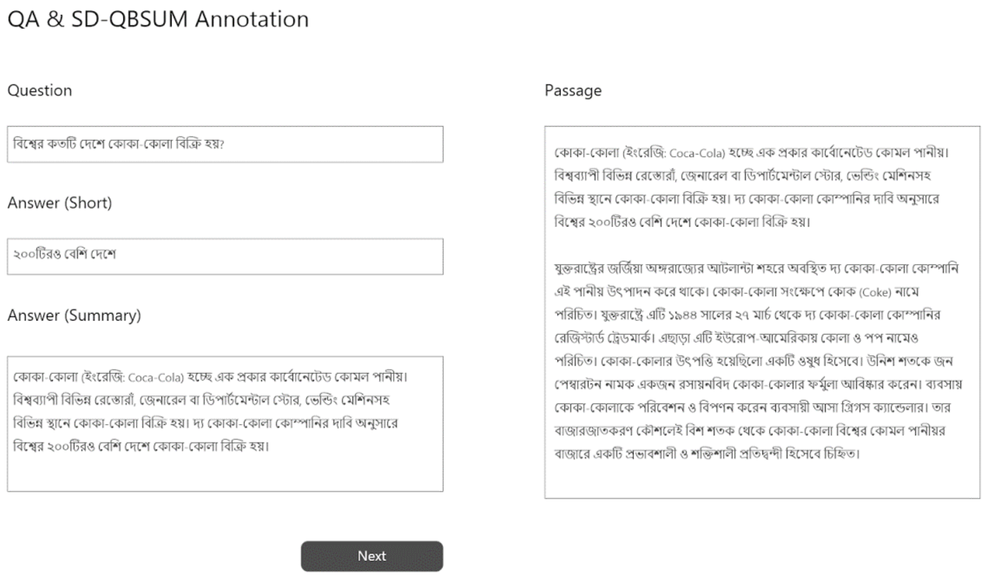
5.1.1 Hardware

The hardwares we used to conduct our experiment are given below:
To train our models we used GPU from google colab with specifications of

- GPU: K80 or T4 (12GB VRAM).
- RAM: 16GB.
- STORAGE: 80GB.

Specification of local machines used to store the dataset used for experimentation

- CPU: Intel Core i7 10th Generation.
- RAM: 16GB.
- STORAGE: 1TB.



QA & SD-QBSUM Annotation

Question

বিশ্বের কতটি দেশে কোকা-কোলা বিক্রি হয়?

Answer (Short)

২০০টিরও বেশি দেশে

Answer (Summary)

কোকা-কোলা (ইংরেজি: Coca-Cola) হচ্ছে এক প্রকার কার্বোনেটেড কোমল পানীয়। বিশ্বব্যাপী বিভিন্ন রেস্তোরাঁ, জেনারেল বা ডিপার্টমেন্টাল স্টোর, ভেন্ডিং মেশিনসহ বিভিন্ন স্থানে কোকা-কোলা বিক্রি হয়। দ্য কোকা-কোলা কোম্পানির দাবি অনুসারে বিশ্বের ২০০টিরও বেশি দেশে কোকা-কোলা বিক্রি হয়।

Passage

কোকা-কোলা (ইংরেজি: Coca-Cola) হচ্ছে এক প্রকার কার্বোনেটেড কোমল পানীয়। বিশ্বব্যাপী বিভিন্ন রেস্তোরাঁ, জেনারেল বা ডিপার্টমেন্টাল স্টোর, ভেন্ডিং মেশিনসহ বিভিন্ন স্থানে কোকা-কোলা বিক্রি হয়। দ্য কোকা-কোলা কোম্পানির দাবি অনুসারে বিশ্বের ২০০টিরও বেশি দেশে কোকা-কোলা বিক্রি হয়।

যুক্তরাষ্ট্রের জর্জিয়া অঙ্গরাজ্যের আটলান্টা শহরে অবস্থিত দ্য কোকা-কোলা কোম্পানি এই পানীয় উৎপাদন করে থাকে। কোকা-কোলা সংক্ষেপে কোক (Coke) নামে পরিচিত। যুক্তরাষ্ট্রে এটি ১৯৪৪ সালের ২৭ মার্চ থেকে দ্য কোকা-কোলা কোম্পানির রেজিস্টার্ড ট্রেডমার্ক। এছাড়া এটি ইউরোপ-আমেরিকায় কোলা ও পপ নামেও পরিচিত। কোকা-কোলার উৎপত্তি হয়েছিলো একটি ওষুধ হিসেবে। উনিশ শতকে জন পেম্বারটন নামক একজন রসায়নবিদ কোকা-কোলার ফর্মুলা আবিষ্কার করেন। ব্যবসায় কোকা-কোলাকে পরিবেশন ও বিপণন করেন ব্যবসায়ী আসা গ্রিগস ক্যান্ডেলার। তার বাজারজাতকরণ কৌশলেই বিশ শতক থেকে কোকা-কোলা বিশ্বের কোমল পানীয়র বাজারে একটি প্রভাবশালী ও শক্তিশালী প্রতিদ্বন্দী হিসেবে চিহ্নিত।

Figure 5.1: Annotation Software

5.1.2 Annotation Software

For annotating data manually, we developed a software with simple python windows GUI library Tkinter. A snippet of the annotation software is given below:

5.1.3 Baseline Models

After the completion of the creating our datasets, we choose the following state-of-the-art multi-lingual deep learning models to experiment on our datasets. These models were carefully picked owing to their good performance in different multi-lingual tasks and some other reasons which are specific to each architecture that are discussed below.

- mT5 [7] is a multilingual version of ”Text-to-text Transfer Transformer” (T5). The model is pretrained on a large scale dataset called common crawl dataset. The dataset consists of 101 languages which makes the model very useful to apply in multilingual tasks. The basic concept that is applied here is to treat every text processing problem such as summarization, question answering, machine translation, text classification as ”text-to-text” approach.

Meaning input is must be textual data and the output generated must also be textual. It generally performs better when both the input and the output of the task consists of text strings.

- mBERT is the multilingual variant of the BERT [8] model. We have chosen it as it is pre-trained on multiple languages including Bengali, thus it can generalize well on our dataset during experimentation. In addition, it can perform downstream tasks better by utilizing its pre-trained capability. Moreover, pre-trained models have some other advantages such as (1) it doesn't require much labeled data (2) it requires less effort on building the models architecture (3) it allows transfer learning therefore saving time and resources.

- XLM-ROBERTa [9] is the multilingual version of the robustly optimized BERT [8] model. The main idea behind this paper is that, if we pretrain multilingual language models using large scale data, then we can gain strong performance for various types of cross-lingual tasks. Moreover, this approach can also achieve better performance on low-resource languages than under-trained multilingual language models.

- Sentence-BERT [10] is a variation of the BERT [8] model. This model can be applied to compute state-of-the-art sentence, text and image embeddings for more than 100 languages. It targets to reduce the huge computational overhead BERT needs as both sentences of the sentence-pair task are required to be fed into the network. Sentence-bert utilizes the benefits of Siamese and Triplet network architectures. This procedure helps to extract and attain sentence embeddings that are relevant and semantically meaningful. Similarity matching techniques such as cosine similarity can be used to compare these embeddings which can help to search for related and comparable pairs. This customizations makes executing other similar tasks such as question answering much faster and simpler. While BERT takes almost 65 hours to find similar item pairs Sentence-BERT minimizes the effort drastically to mere 5 seconds all the while retaining the performance of BERT. Some of the used pretrained Sentence Transformer models are described below.

- xlm-r-100langs-bert-base-nli-stsb-mean-tokens and xlm-r-100langs-bert-base-nli-mean-tokens are both pretrained sentence transformers models which are used for executing tasks such as clustering or semantic search. It portrays sentences and paragraphs to a 768 dimensional vector space.

- Paraphrase-xlm-r-multilingual-v1 is the multilingual variant of teacher model paraphrase-distilroberta-base-v1. It is trained on large-scale parallel data consisting of paraphrase sentences on more than 50+ languages and in this case XLM-R is used as the student model.

- Paraphrase-multilingual-mpnet-base-v2 is the multilingual version of teacher model paraphrase-mpnet-base-v2. It is trained on parallel data of large scale paraphrase sentences covering 50+ languages where xlm-roberta-base is used as the student model.

- LABSE [11] (Language Agnostic Bert Sentence Embeddings) is an improvement of the multilingual BERT model that derives language agnostic cross lingual sentence embeddings. LABSE proposes a sophisticated method to learn monolingual sentence embeddings by utilizing combination of methods such as masked language modeling, translation language modeling, dual encoder translation ranking and additive softmax margin. The reason we have selected to use this model is that pretrained models like BERT are an effective procedure for learning only monolingual sentence embeddings. On the oher hand, LABSE uses a combination of aforementioned methods above to learn multilingual sentence embeddings. Furthermore, usage of the multilingual pretrained language model significantly reduces the amount of parallel data needed to achieve excellent performance by 80%.

5.2 Experiments

Our experiments consists of running the above mentioned models on our dataset to train, validate and test.

We take mBERT, mT5, and XLMRoBERTa models for Question answering task, and the rest of the model for Query Based Single- and Multi-Document

Summarization. These models are usually run for 10-100 epochs depending on the stage of the experiments.

5.2.1 Choosing Baseline Models

To carry out our experiments we have chosen multilingual models for the most part. Majority of the datasets used in the domain of Natural Language Processing are mostly in English. This makes it hard for monolingual models to generalize on low-resource languages like Bengali. Multilingual models trained on datasets that consists of multiple languages generalizes and performs well on cross-lingual transfer tasks of low-resource languages. Monolingual models often overlook the different complex aspects of different languages. Differences in grammar, syntax style and word order also causes problems for typical neural models. For this reason we have chosen multilingual models to conduct our experiment.

Chapter 6

Result Analysis and Discussion

In this chapter, we detail out and discuss our results and analysis them along with our Dataset. We show Statistical analysis, comparisons among other datasets and baseline scores.

6.1 Statistical Analysis of the Dataset

To see the statistical information about our dataset, let us refer to the following table:

Table 6.1: Quantitative Analysis on our Dataset

	Our Dataset		
Metric	**Question Answering (around)**	**Single Document QBSUM (around)**	**Multi-Document QBSUM (around)**
No.of Data	38k		2k (each with 3 different documents)
No. of Unique Context	30k		5k
Avg. Context Length (Words/Char)	486 / 3k		
Avg. Question Length (Words/Char)	7 / 45		
Avg.Answer Length (Words/Char)	3 / 18	58 / 343	
Avg. Lexical Overlapping (Words) [Question and Ans Sentence]	1.25	1.33	

In the above table, we see the statistical information about our dataset, including: number of samples, unique context, Average length of question, context, and answers etc.

About how much answers and what type of answer do we have, can be described in the following figure:

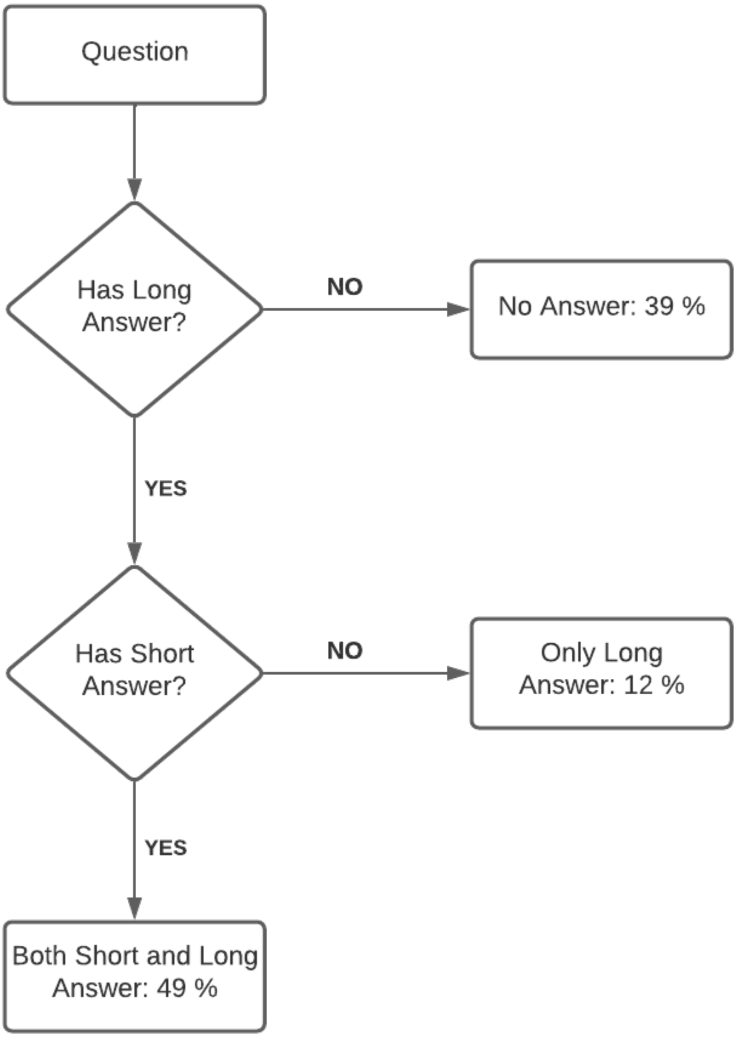


Figure 6.1: Dataset Types of Answers Count

6.2 Comparison Among QA Datasets

A in-detail comparison among the three datasets of the QA is given below, and also stated in the later sections:

Table 6.2: Quantitative Analysis on QA Datasets

Metric	Bengali SQuAD (around)	TyDiQA (around)	Our Dataset (around)
No.of Data	**90k**	10k	38k
No. of Unique Context	14k	1.8k	**30k**
Avg. Context Length (Words/Char)	99.5/705	104/641	**486/3k**
Avg. Question Length (Words/Char)	**8/56**	9/48	7/45
Avg.Answer Length (Words/Char)	2/13	2/14	**3/18**
Avg. Lexical Overlapping (Words) [Question and Ans Sentence]	1.96	1.8	**1.25**

6.2.1 Our Dataset vs Bengali SQuAD

Since we will be creating a QA dataset in Bangla, it makes sense to compare our plan with the existing dataset - Bengali SQuAD. For any QA dataset, we need questions, context, and answers.

The main difference between our work and that of Bengali SQuAD will be in how the answers will be documented. In contrast to their machine translation approach, we plan to annotate our answers through humans. The source documents that we will be using as context are Wikipedia articles. Even though the corpus size of the Wikipedia‘ articles (around 30 thousand) is smaller in comparison to Bengali SQuAD (around 90 thousand), our dataset will still contain more words on average per document (600 words on average) compared to Bengali SQuAD (which contains 100 words on average). These comparisons are visualized with bar graphs below.

And lastly, the questions to be used in our dataset will be natural and taken from Google NQ, and translated to Bangla, whereas Bengali SQuAD translated the questions (which were synthetic) from the SQuAD 2.0 dataset.

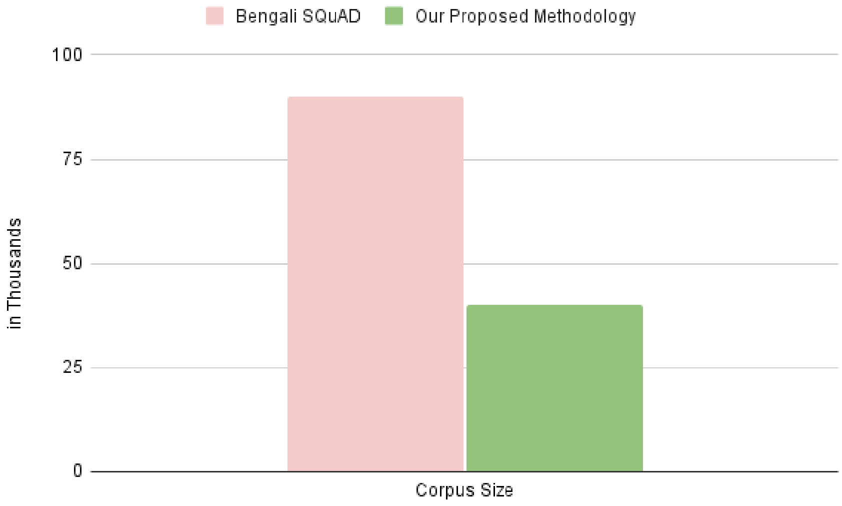


Figure 6.2: Corpus Size - Our Proposed Methodology vs BSQuAD

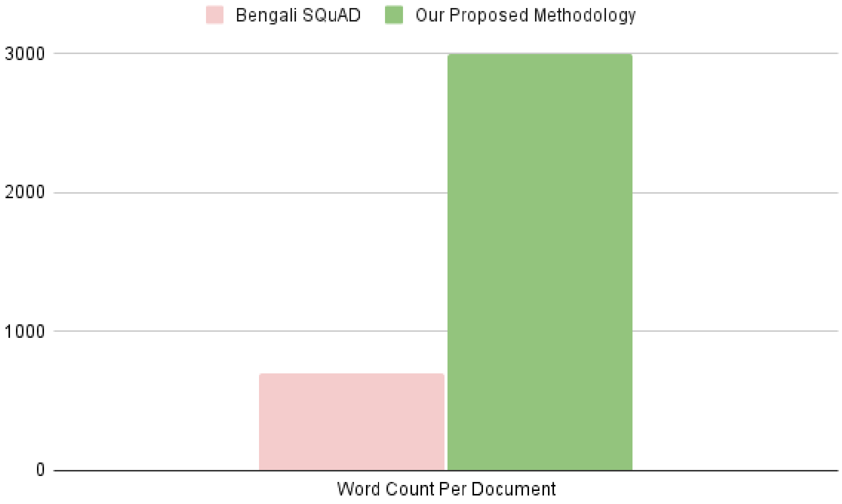


Figure 6.3: Word Count Per Document - Our Proposed Methodology vs BSQuAD

6.2.2 Our Dataset vs AQUAMUSE

For our semi-automatic approach for dataset generation, we are taking inspiration from AQUAMUSE. Consequently, it becomes mandatory to compare the two approaches so that the key differences are highlighted. The first way in which our proposed methodology differs is in the annotation process. AQUAMUSE followed a predominantly discriminative approach, where they already had the summaries (from long answers in Google NQ), and their task was simply to identify the documents in Common Crawl that best match these summaries, using semantic similarity. However, we will follow a combination of discriminative and generative approach. We will first match the documents (i.e. Wikipedia articles) with those from the Common Crawl, and once we do that we will generate a summary which will be done by humans.

The important thing to note here is that AQUAMUSE is using a discriminative approach for matching sentence with documents, however we will match documents from Wikipedia with documents from Common Crawl.

6.3 Evaluation Metric

All our measurements are done on Exact Matching. Exact Matching refers to complete full string matching between the Gold Answer (actual answer) and the Predicted answer. A prediction in counted as positive if the prediction matches with the Gold Answer. And as such, we count the number of samples that match exactly, and divide it by the number of total samples, thus getting the score of Exact Matching.

6.4 Baseline

This section of the chapter refers to the baseline scores that we have gotten by training and testing various models on our dataset. The models are mentioned in the previous chapter in the section of Baseline Models. Several of them are models for Question Answering task, and several others are for QBSUM tasks.

The baseline scores of various aspects are stated in the next sections:

6.4.1 Baseline Comparing Among Question Answering Datasets

For comparing the baselines among the different question answering datasets, we have first separated 3 training sets and 3 test sets that come from Bengali SQuAD, TyDiQA and our dataset. These test sets were used to calculate the exact matching scores. Then the trained datasets were evaluated on each of the 3 test sets.

For example, in the third column of table 6.3, we report the scores after evaluating the performance of the mT5 model trained on our dataset and tested on each of the 3 test sets.

Table 6.3: Baseline - Question Answering (using pre-trained mT5 model on all the datasets)

		Trained On		
		Bengali SQuAD	TyDiQA	Our Dataset
Tested On	Bengali SQuAD	4.0	6.0	2.0
	TyDiQA	6.0	11.0	8.0
	Our Dataset	4.0	14.0	42.0

The mT5 model trained on Bengali SQuAD performs poorly on all the test sets. This is consistent with our belief that Bengali SQuAD contains a lot of noise and inconsistencies in the dataset so consequently it does not produce satisfactory results. Furthermore, the models trained on other datasets also have bad performance on the Bengali SQuAD and this suggests that Bengali SQuAD is unsuitable for any sort of training or testing.

On the other hand, the model trained on TyDiQA had unsatisfactory performance on our test set and similarly the model trained on our dataset had similar results on the TyDiQA test set. We can attribute this fact to the different nature of the two datasets. We have used the TyDiQA gold passage dataset for training, and in this dataset every example is guaranteed to have an answer. However, in our dataset, the examples may or may not have an answer. This is because we want to teach the model to be able to answer a question but we also want it to refrain from answering if the answer does not exist in the context.

6.4.2 Baseline of Question Answering on Our Dataset

In this section, we discuss the baselines scores of the different Question Answering models on our dataset. For this set of experiments, we have chosen 3 multilingual models - mT5, mBERT, xlm-ROBERTa.

All of the scores were calculated are exact match scores which are are expressed as percentage. We use our separate test set for carrying out the evaluations.

In the first part of the experiment, we apply a zero-shot setting, where the model has no idea about the training dataset, and it was simply used to perform inference on the test set and the output scores were reported. The low score of mT5 in the zero-shot setting can be attributed to the fact that it was not aware of the task of question answering. The mT5 model was simply pre-trained in an unsupervised manner on the multilingual common crawl corpus.

After training the models, we observe a significant rise in performance of all the models. This shows that our dataset is suitable for training a model for the task of question answering. We believe that with a larger dataset, and training our model for more epochs, the performance will rise much more.

Table 6.4: Baseline - Question Answering (Exact Match expressed as a percentage (%))

Model No.	Model Name	Zero-Shot Setting (Not Trained on Our Dataset)	Trained on Our Dataset
1	mT5	6.0	42.0
2	mBERT	26.0	34.0
3	xlm-ROBERTa	28.0	40.0

6.4.3 Baseline of Single Document and Multi-Document Query Based Summarization on Our Dataset

Table 6.5: Baseline - Single-Doc and Multi-Doc QBSUM (Exact Match expressed as a percentage (%))

Model No.	Model Name	Zero-Shot Setting (Not Trained on Our Dataset)	Trained on Our Dataset
1	xlm-r-100langs-bert-base-nli-stsb-mean-tokens	18.0	76.0
2	paraphrase-multilingual-mpnet-base-v2	16.0	46.0
3	paraphrase-xlm-r-multilingual-v1	22.0	57.9

For both single document and multi-document query based summarization, different multilingual SBERT models were used. SBERT models were suitable here as summarization involves outputting the most salient information and consists of sentences. SBERT develops a sentence level understanding and hence we have chosen these models to carry out our experiments on.

We used a zero-shot setting here as well and we observed a similar rise in performance on our test set. The first model in the above table proved to be the best among the three models on which the experiments were carried out.

Chapter 7

Conclusion and Future Work

In this thesis work, we show pipeline and approach to create datasets for 3 different tasks of NLP - QA, SD-QBSUM and MD-QBSUM, and continue to build them. We did face lot of challenges, and there are shortcomings in the current work. We plan to work and address them later.

Specially, completion of the dataset with manual annotation, creating multi-document dataset with proper similarity measurements and outside of wikipedia are some. Additionally, we can extend the work and continue to the task of topic modeling, cross-lingual question answering, and overall summarization.

We hope that our work, now and later when fully completed, will help accelerate QA, QBSUM research in Bangla NLP, and will inspire similar endeavours in other low resource languages as well.

Bibliography

[1] T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, J. Devlin, K. Lee, *et al.*, "Natural questions: a benchmark for question answering research," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 453–466, 2019.

[2] P. Rajpurkar, R. Jia, and P. Liang, "Know what you don't know: Unanswerable questions for squad," *arXiv preprint arXiv:1806.03822*, 2018.

[3] T. Tahsin Mayeesha, A. Md Sarwar, and R. M. Rahman, "Deep learning based question answering system in bengali," *Journal of Information and Telecommunication*, vol. 5, no. 2, pp. 145–178, 2021.

[4] S. Kulkarni, S. Chammas, W. Zhu, F. Sha, and E. Ie, "Aquamuse: Automatically generating datasets for query-based multi-document summarization," *arXiv preprint arXiv:2010.12694*, 2020.

[5] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *arXiv preprint arXiv:1910.10683*, 2019.

[6] J. H. Clark, E. Choi, M. Collins, D. Garrette, T. Kwiatkowski, V. Nikolaev, and J. Palomaki, "Tydi qa: A benchmark for information-seeking question answering in typologically diverse languages," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 454–470, 2020.

[7] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, and C. Raffel, "mt5: A massively multilingual pre-trained text-to-text transformer," *arXiv preprint arXiv:2010.11934*, 2020.

[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[9] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," *arXiv preprint arXiv:1911.02116*, 2019.

[10] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," *arXiv preprint arXiv:1908.10084*, 2019.

[11] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang, "Language-agnostic bert sentence embedding," *arXiv preprint arXiv:2007.01852*, 2020.