Title	Query-Focused Extractive Text Summarization for Multi-Topic Document
Author(s)	水野, 慎一朗
Citation	
Issue Date	2022-09
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/18052
Rights	
Description	Supervisor:Nguyen Minh Le, 先端科学技術研究科, 修士(情報科学)

Master's Thesis

Query-Focused Extractive Text Summarization for Multi-Topic Document

Shinichiro Mizuno

Supervisor Prof. Nguyen Minh Le

Graduate School of Advanced Science and Technology
Japan Advanced Institute of Science and Technology
(Information Science)

August, 2022

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, we make a brief introduction of the background of this study, the problem that we would like to deal with, and the objectives of the research.

1.1 Background

With the continuous increase in the amount of information flowing online, it is important to provide a system for quickly finding the information you need. Document summarization is an effective tool for quickly going through huge amount of information and understanding its essence by providing a general overview of the content without having to read through the entire text. However, as people have different interests for each individual, if the summary is generated based on a different perspective from the one you expect, you could not find the information that you look for. For example, suppose there is a book on business strategy and the book contains the statements of marketing, finance, and other several topics. A person who is interested in marketing needs a summary about marketing from the book, and a person interested in finance needs a summary about finance from the book. If you give a summary based on a finance perspective to those interested in marketing, it is not helpful. Perspective is important in creating summary from documents with multiple topics (multi-topic documents).

In addition, obtaining information across the documents in consistent perspectives increases the comparability of information and makes the information useful. For example, assume that there are multiple books regarding business strategy that include statements on marketing. We could compare the differences in the statements of marketing among the books, if we consistently obtain the statements of marketing across the books. Consistency is the key in obtaining information across the documents.

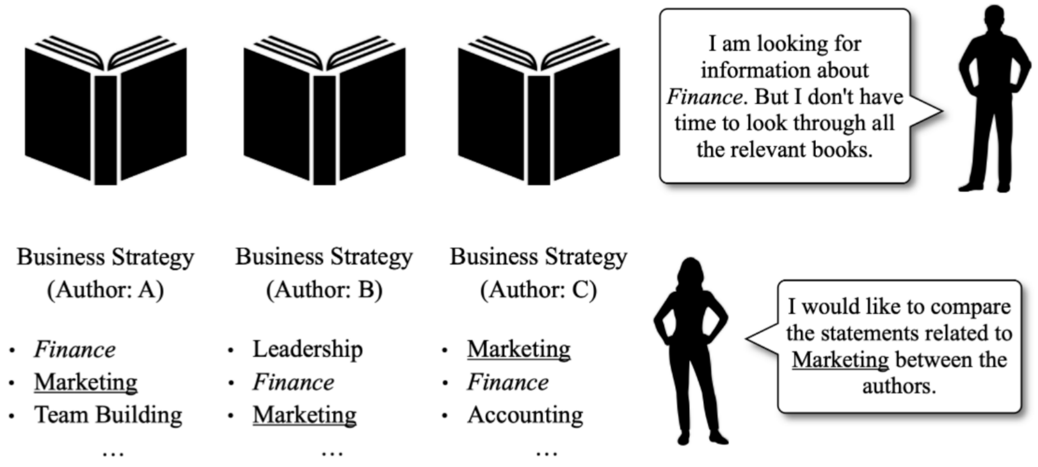

I am looking for information about *Finance*. But I don't have time to look through all the relevant books.
Business Strategy (Author: A)
- *Finance*
- Marketing
- Team Building
...
Business Strategy (Author: B)
- Leadership
- *Finance*
- Marketing
...
Business Strategy (Author: C)
- Marketing
- *Finance*
- Accounting
...
I would like to compare the statements related to Marketing between the authors.

Figure 1.1: Multi-Topic Documents and Summary Perspectives

1.2 Problem Statement

How do we solve the problem of obtaining information from multi-topic documents in consistent perspectives? One of the approach is to consider the topic as a query for the document and to extract the information as a summary of the document. With the approach, we can solve the problem as a query-focused extractive text summarization task. Based on the assumption, we attempted to find an existing dataset for the task of query-focused text summarization. However, there is no existing dataset that satisfies the requirement of our problem. DUC 2005, 2006, 2007 (DUC 2005-2007) is one of the well-known query-focused text summarization datasets, although datasets for query-focused text summarization are very limited. However, DUC 2005-2007 provides only one query and corresponding summaries for a single document, although we are looking for a dataset with multiple queries and corresponding summaries for the single document. DUC 2005-2007 does not fit for the purpose of extracting summaries through multiple queries from a multi-topic document.

Another approach would be to solve this problem as a Question Answering (QA) problem, where documents and questions are given and answer spans corresponding to the questions are obtained from the documents. However, existing QA datasets do not satisfy the requirement of our problem either. If you look at QA datasets such as SQuAD and TriviaQA, you would find these datasets are similar to query-focused summarization datasets in that they consist of a triple of ”document”, ”query”, and ”output”. However, existing QA datasets have a single answer span to be selected from a document, although a multi-topic document require multiple answer spans to be selected as the target topic statement appears several times. Thus, existing QA datasets do not fit for the purpose of extracting multiple summaries through multiple queries from a multi-topic document.

1.3 Objectives

As discussed above, existing datasets do not fit for the purpose of our problem regarding multi-topic documents. We create a novel dataset to deal with the problem that we want to solve. The requirement for the dataset is that documents consist of multiple topics and extractive summary be provided for each of the topics. We introduce what data we leverage and how we construct our dataset.

In addition, since our dataset is a new dataset, no reasonable method has been established for extracting topic-by-topic text from multi-topic documents. We propose a novel query-focused summarization method for the created dataset. For comparison purpose, we devise a solution to solve this problem as a Question Answering (QA) task. We compare the performance of query-focused summarization methods, QA task methods, and several existing baseline methods through experiments.

In summary, the objectives of our study are as follows; one is to build a novel dataset of multi-topic documents. The second is to establish a method for extracting topic-by-topic text from multi-topic documents.

1.4 Thesis Outline

The content outline of this thesis is as follows;

- Chapter2: Related Work: We introduce an overview of existing dataset for query-focused summarization tasks and QA tasks, and some SOTA methods of extractive text summarization.

- Chapter3: Dataset: We introduce the background of creating new dataset and how we build the new dataset, followed by some analysis on the dataset.

- Chapter4: Baseline Models: We introduce how we leverage baseline models and apply them to the query-focused summary extractor to process our data set with the detail implementation.

- Chapter5: Proposed Models: We introduce the high level concept of our solution approach, followed by the detail architecture and the implementation of our proposed methods.
- Chapter6: Experiments: We conduct experiments to explore the optimal parameters for each model through the validation data, followed by the performance evaluation on the models with optimal parameters with some analysis.
- Chapter7: Conclusions: We conclude the research.

Chapter 2

Related Work

In this chapter, we introduce an overview of existing dataset for query-focused summarization tasks and QA tasks, and some SOTA methods of extractive text summarization.

2.1 Datasets

2.1.1 Query-Focused Summarization Tasks

DUC 2005-2007 is widely known as a dataset for query-focused text summarization tasks. DUC 2005 was prepared as a dataset for solving user-oriented system tasks [1]. In this task, the system was expected to generate concise and well-organized summaries for specific queries from 25-50 document clusters of news articles. The supervised summary data for the queries was manually annotated. For DUC 2006 and 2007, additional datasets were prepared in a similar manner to DUC 2005. DUC 2005-2007 provides only one query and corresponding summaries for a single document, although we are looking for a dataset with multiple queries and corresponding summaries for the single document. DUC 2005-2007 does not fit for the purpose of extracting summaries through multiple queries from a multi-topic document.

QMSum [2] is one of the query-focused text summarization datasets, which consists of meeting minutes of three different domains: product meetings, academic meetings, and committee meetings. In QMSum, there are multiple queries and their summaries for each meeting minute. QMSum is a multi-topic text summarization dataset, which is close to the dataset we look for. However, the queries of QMSum are not set consistently throughout the dataset. As mentioned in section 1.1, consistency in perspective across the dataset is the key for our problem.

2.1.2 Question Answering Tasks

The Stanford Question Answering Dataset (SQuAD) is a manually constructed Question Answering (QA) dataset consisting of approximately 100k triples of question-answer-Wikipedia articles (Rajpurkar et al., 2016). Answers to questions are represented by corresponding text spans on the articles, and some questions have no answers. SQuAD's task is to extract the text span that would be the answer in a passage given a question and a Wikipedia passage. TriviaQA is also a reading comprehension dataset containing over 650K triples of questions, answers, and evidence. TriviaQA contains 95K question/answer pairs written by trivia enthusiasts and an average of six independently collected evidence documents per question to answer the questions. The task of TriviaQA is to extract the text span that would be the answer from the evidence document given a question. Similar to query-focused text summarization datasets such as DUC2005-2007, these QA datasets consist of triples of "document", "query", and "output". While "output" in DUC2005-2007 is sentences, the "output" in these QA datasets is one or a sequence of words.

2.2 Methods

2.2.1 Query-Focused Extractive Summarization

Various approaches have been proposed for query-specific extractive text summarization; Goldstein et al. [5] proposed Maximum Marginal Relevance (MMR) that is an unsupervised learning approach that greedily selects sentences and considers the trade-off between relevance to the query and redundancy with the extracted summary text. Ouyang et al. [6] proposed a supervised method that extracts query-dependent and query-independent features and learns feature weights using Support Vector Regression. Cao et al. [7] proposed a neural attention summarization system (AttSum) that simultaneously handles query importance ranking and sentence saliency ranking. This model automatically learns distributed representations of sentences as well as documents. Ren et al. [8] proposed a contextual relation-based neural summarization system (CRsum) that uses contextual relations among sentences to improve sentence scoring. Zhu et al. [9] proposed a query-focused extractive summarization model using BERT [10], a pretrained language model. This model uses the concatenation of queries and sentences as input and generates binary scores whether or not each sentence should be included in the summary extraction. This method achieved SOTA in query-focused extractive text summarization for the DUC 2005-2007 dataset.

2.2.2 Generic Extractive Summarization

Pretrained language models have been successful with generic extractive summarization; HIBERT [11] pretrains a hierarchical encoder for the extractive model on unlabeled data, then classifies sentences with a model initialized from the pretrained encoder. BERTSum [12] recorded SOTA at the time with a model that combined BERT [10] with Transformer [13] in summarization layer. MatchSum [14] formulated the extractive summarization task as a semantic text matching problem. It matches source sentences and summary candidates in semantic space, rather than extracting sentences individually and modeling sentence relationships.

Chapter 3

Dataset

As discussed in section 1.3, we build a new dataset of multi-topic documents. In this chapter, we introduce a data source of the new dataset and dataset building pipeline that incorporates data collection process and dataset creation process. Then, we make some analysis on the created dataset.

3.1 Data Source

The requirement for the dataset is that documents consist of multiple topics and extractive summary be provided for each of the topics. As a data source of the dataset that satisfies this requirement, we take advantage of "integrated reports". Integrated reports are issued by listed companies mainly on an annual basis for investors that integrates financial information, such as business overview and financial status, with non-financial information, such as environmental and social initiatives. Integrated reports of some companies show the relevance of their environmental and social initiatives to the 17 goals of the SDGs (Sustainable Development Goals).

The SDGs are a collection of 17 interlinked global goals designed to be a "blueprint to achieve a better and more sustainable future for all". See Figure 3.1 for the goals. The SDGs were announced at the 2015 UN Summit and are aimed at achieving a sustainable world by 2030. Each company is expected to disclose their initiatives on the SDGs to communicate the sustainability of its business. In order to demonstrate the relevance of their environmental and social initiatives to the SDGs, an increasing number of companies are labeling each of their initiatives with the 17 goals of SDGs in their integrated reports.

These integrated reports are not only appropriate for this study as multi-topic documents, but they can also be seen as corpora that have already been

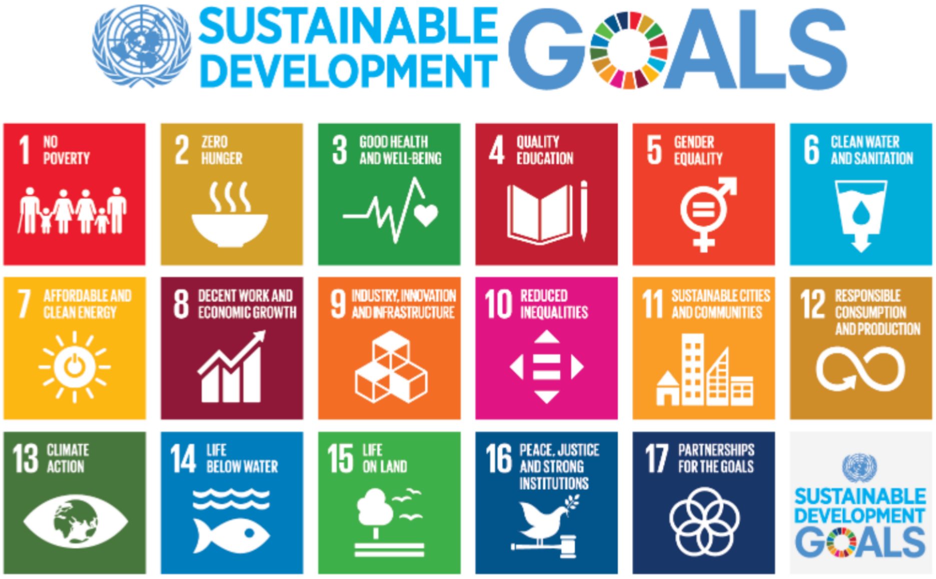
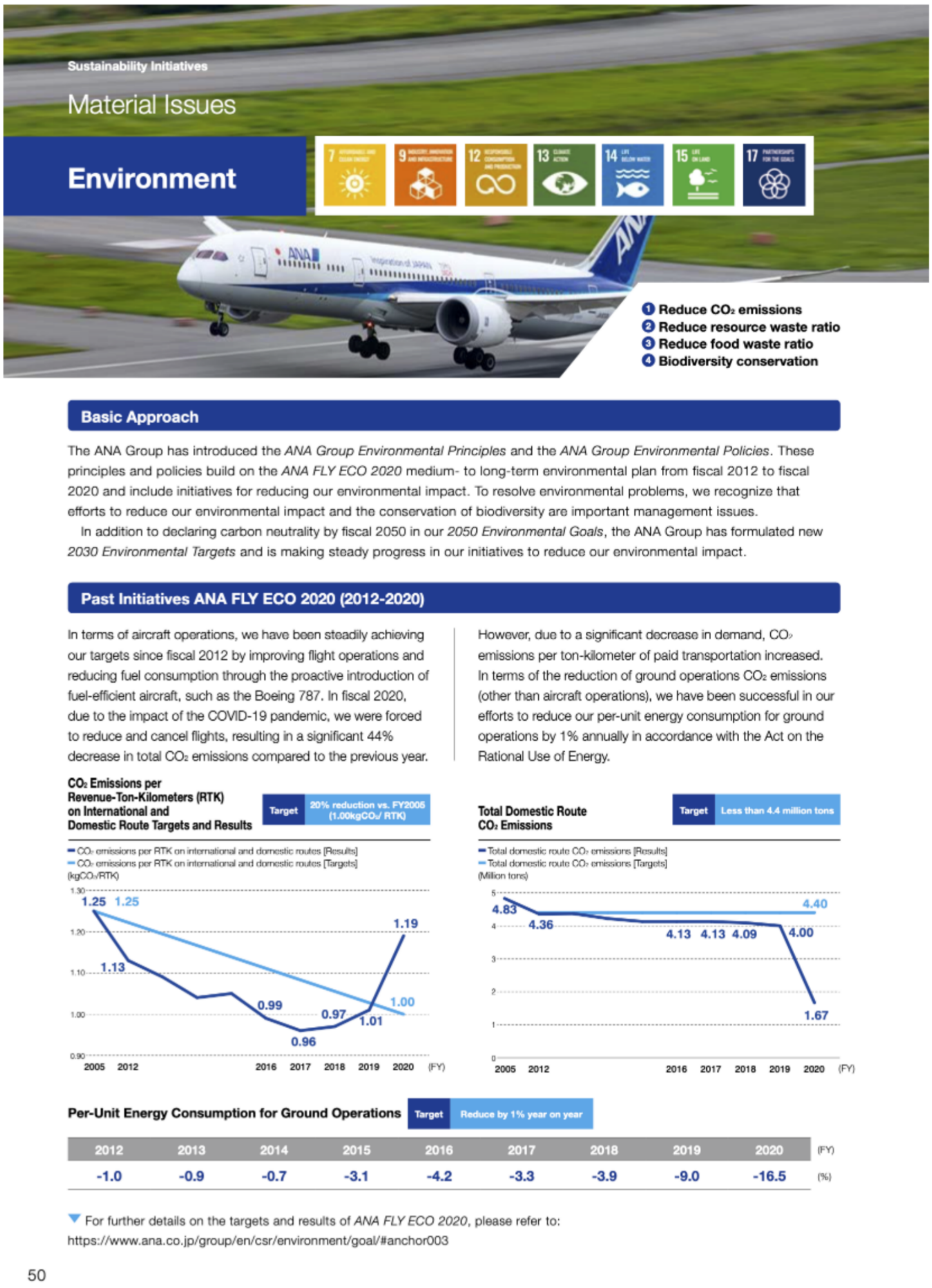

SUSTAINABLE DEVELOPMENT GOALS
1 NO POVERTY
2 ZERO HUNGER
3 GOOD HEALTH AND WELL-BEING
4 QUALITY EDUCATION
5 GENDER EQUALITY
6 CLEAN WATER AND SANITATION
7 AFFORDABLE AND CLEAN ENERGY
8 DECENT WORK AND ECONOMIC GROWTH
9 INDUSTRY, INNOVATION AND INFRASTRUCTURE
10 REDUCED INEQUALITIES
11 SUSTAINABLE CITIES AND COMMUNITIES
12 RESPONSIBLE CONSUMPTION AND PRODUCTION
13 CLIMATE ACTION
14 LIFE BELOW WATER
15 LIFE ON LAND
16 PEACE, JUSTICE AND STRONG INSTITUTIONS
17 PARTNERSHIPS FOR THE GOALS
SUSTAINABLE DEVELOPMENT GOALS

Figure 3.1: Sustainable Development Goals

annotated by corporate IRs with lables that indicate the relevance of their environmental and social initiatives to the 17 goals of the SDGs. By making use of these labels, we can drastically reduce the work that would normally be required to manually read and assign labels to each document one by one.

3.2 Data Collection Pipeline

The data collection pipeline incorporates the following tasks: identifying the companies that publish integrated reports, downloading the PDF files, and selecting the PDF files to be used in our dataset. See Figure 3.3 for the pipeline. First, we identify the companies that publish integrated reports. We leverage the list reported by Disclosure IR Research Institute Ltd., in which the names of the companies that publish integrated reports are listed out. After identifying the companies, we download PDF files for the past five years from 2017 to 2021 from the websites of 251 companies that publish integrated reports in English (754 files in total). Note that not all companies have issued integrated reports for all five years, as some companies have started issuing integrated reports more recently. We check the contents of the integrated reports and selected only those integrated reports that have labels indicating the relevance of the company's environmental and social initiatives to the 17



Sustainability Initiatives

Material Issues

Environment

1 Reduce CO₂ emissions
2 Reduce resource waste ratio
3 Reduce food waste ratio
4 Biodiversity conservation

Basic Approach

The ANA Group has introduced the *ANA Group Environmental Principles* and the *ANA Group Environmental Policies*. These principles and policies build on the *ANA FLY ECO 2020* medium- to long-term environmental plan from fiscal 2012 to fiscal 2020 and include initiatives for reducing our environmental impact. To resolve environmental problems, we recognize that efforts to reduce our environmental impact and the conservation of biodiversity are important management issues.

In addition to declaring carbon neutrality by fiscal 2050 in our *2050 Environmental Goals*, the ANA Group has formulated new *2030 Environmental Targets* and is making steady progress in our initiatives to reduce our environmental impact.

Past Initiatives ANA FLY ECO 2020 (2012-2020)

In terms of aircraft operations, we have been steadily achieving our targets since fiscal 2012 by improving flight operations and reducing fuel consumption through the proactive introduction of fuel-efficient aircraft, such as the Boeing 787. In fiscal 2020, due to the impact of the COVID-19 pandemic, we were forced to reduce and cancel flights, resulting in a significant 44% decrease in total CO₂ emissions compared to the previous year. However, due to a significant decrease in demand, CO₂ emissions per ton-kilometer of paid transportation increased. In terms of the reduction of ground operations CO₂ emissions (other than aircraft operations), we have been successful in our efforts to reduce our per-unit energy consumption for ground operations by 1% annually in accordance with the Act on the Rational Use of Energy.

CO₂ Emissions per Revenue-Ton-Kilometers (RTK) on International and Domestic Route Targets and Results

Target 20% reduction vs. FY2006 (1.00kgCO₂/ RTK)

CO₂ emissions per RTK on international and domestic routes [Results]
CO₂ emissions per RTK on international and domestic routes [Targets]
(kgCO₂/RTK)
1.30
1.20
1.10
1.00
0.90
1.25
1.25
1.13
0.99
0.96
0.97
1.01
1.19
1.00
2005
2012
2016
2017
2018
2019
2020
(FY)

Total Domestic Route CO₂ Emissions

Target Less than 4.4 million tons

Total domestic route CO₂ emissions [Results]
Total domestic route CO₂ emissions [Targets]
(Million tons)
5
4
3
2
1
0
4.83
4.36
4.13
4.13
4.09
4.00
1.67
4.40
2005
2012
2016
2017
2018
2019
2020
(FY)

Per-Unit Energy Consumption for Ground Operations Target Reduce by 1% year on year

2012	2013	2014	2015	2016	2017	2018	2019	2020	(FY)
-1.0	-0.9	-0.7	-3.1	-4.2	-3.3	-3.9	-9.0	-16.5	(%)

▼ For further details on the targets and results of *ANA FLY ECO 2020*, please refer to:
https://www.ana.co.jp/group/en/csr/environment/goal/#anchor003

50

Figure 3.2: Sample page from an integrated report (ANA HOLDINGS INC., 2021). As seen at the top of this page, the company's initiatives are labeled with the SDGs goals. In addition to qualitative statements, the report incorporates graphs, tables, and photographs.

goals of SDGs. After the selection, 250 integrated reports remained.

We perform the pipeline with one person and it takes approximately 15 minutes per company, or approximately 60 hours for the entire company.

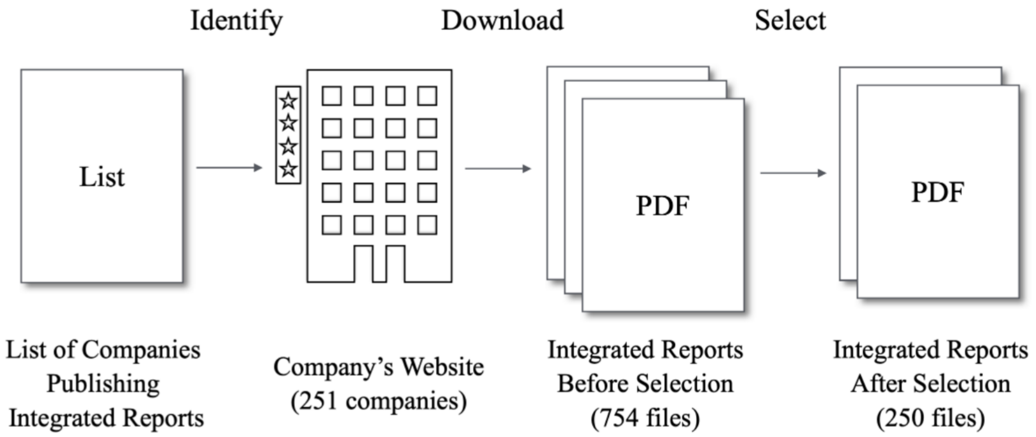

Identify
Download
Select
List
PDF
PDF
List of Companies Publishing Integrated Reports
Company's Website (251 companies)
Integrated Reports Before Selection (754 files)
Integrated Reports After Selection (250 files)

Figure 3.3: Data Collection Pipeline

3.3 Dataset Creation Pipeline

In order to create a dataset from the collected data, we generally do some annotation work. However, as already mentioned, the integrated report in PDF format already has a Goal No. label assigned to the text related to the SDGs. See Figure 3.2 for a sample page of integrated report with SDGs Goal No. label. What we need to do is to extract the text from the PDF files, labels the summary text with Goal No. label and align the summary text with source text. See Figure 3.4 for the pipeline.

In extracting text data from PDF files, we compared automated methods and a manual extraction method. As for the automated methods, we compare Apache Tika and PDFMiner. The manual extraction method is simply to select and copy the text from the PDF pages. Both automated tools have advantages in terms of effort compared to the manual method, but they cannot extract text as expected especially when the text representation on the PDF has a complex structure. For example, if the text is presented in a four-column structure on an A4 spread PDF page, and there are also figures and tables inserted, text extraction would not proceed in the expected direction. See Figure 3.5. In fact, manual extraction is more accurate than automated methods. Manual work is also necessary anyway, since the SDGs

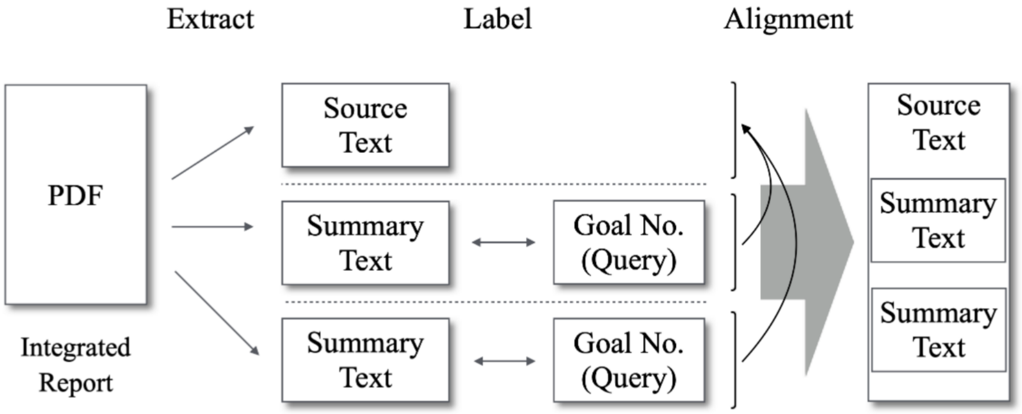


Figure 3.4: Dataset Creation Pipeline

labels are represented by pictures and the labeled text scope needed to be visually verified. After comparison of the methods, we chose manual extraction method.

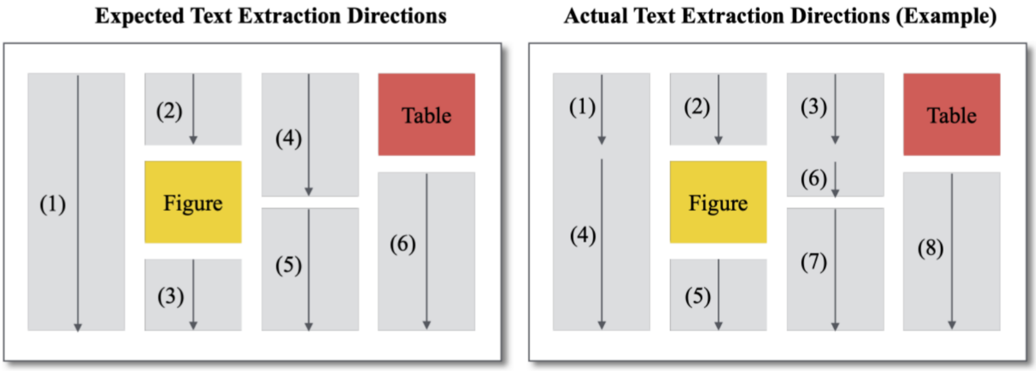
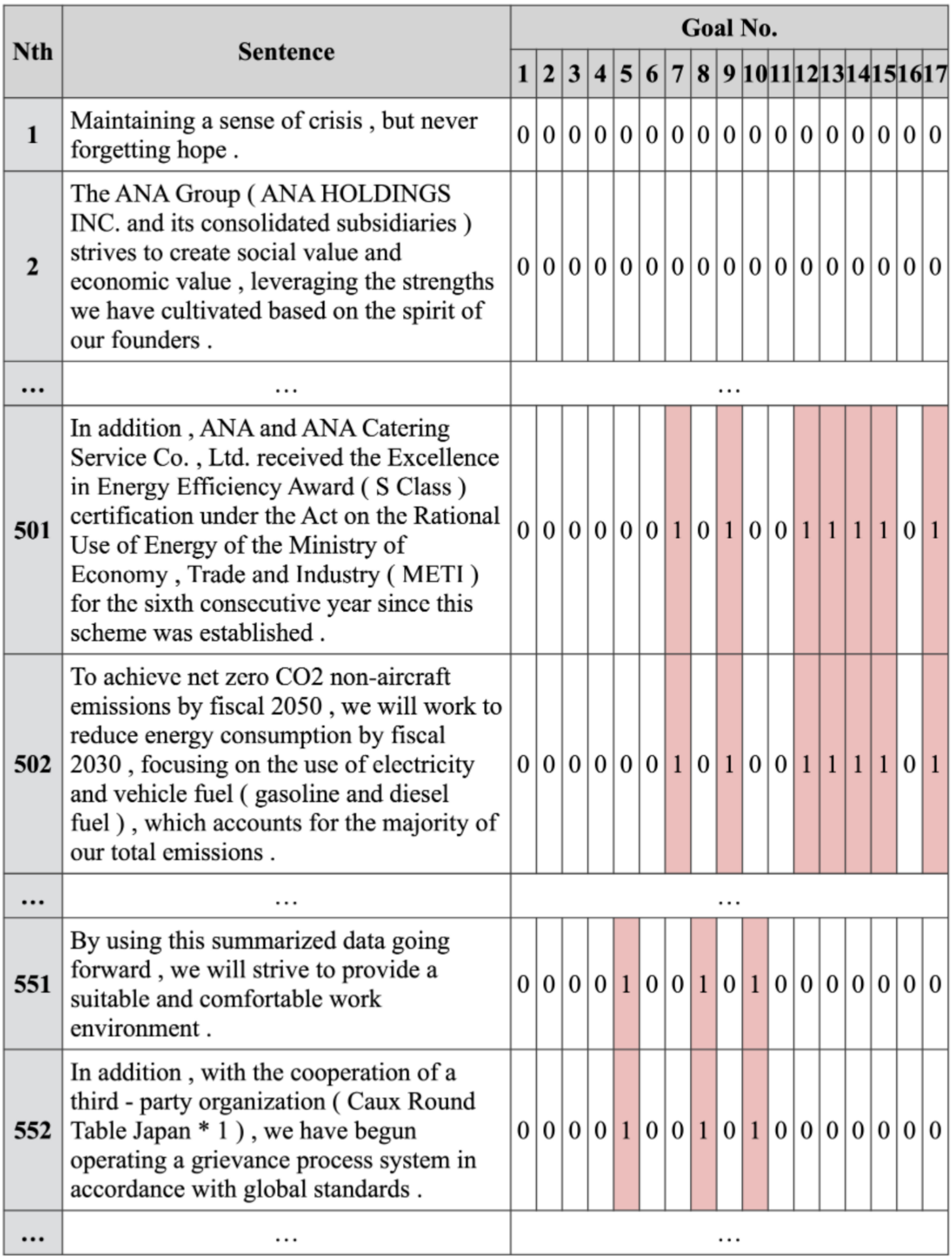

Expected Text Extraction Directions
(1)
(2)
Figure
(3)
(4)
(5)
Table
(6)
Actual Text Extraction Directions (Example)
(1)
(4)
(2)
Figure
(5)
(3)
(6)
(7)
Table
(8)

Figure 3.5: Text Extraction Directions in PDF

In the extraction process, we extract source text and summary text from PDF files. Source text denotes all the text from the PDF files and summary text denotes the text from the labeled area with SDGs Goal No. in the PDF pages. After extracting text from PDF files, we label summary text with Goal No. by adding Goal No. in the text file name to maintain the annotation.

After the labelling process, we align the summary text with source text to indicate which part of source text is the summary text for each Goal No. In the alignment process, each sentence of the source text and summary text are matched and the target value ”1” or ”0” is assigned to the position of



Nth	Sentence	Goal No.																
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1	Maintaining a sense of crisis , but never forgetting hope .	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	The ANA Group (ANA HOLDINGS INC. and its consolidated subsidiaries) strives to create social value and economic value , leveraging the strengths we have cultivated based on the spirit of our founders .	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
…	…	…																
501	In addition , ANA and ANA Catering Service Co. , Ltd. received the Excellence in Energy Efficiency Award (S Class) certification under the Act on the Rational Use of Energy of the Ministry of Economy , Trade and Industry (METI) for the sixth consecutive year since this scheme was established .	0	0	0	0	0	0	1	0	1	0	0	1	1	1	1	0	1
502	To achieve net zero CO2 non-aircraft emissions by fiscal 2050 , we will work to reduce energy consumption by fiscal 2030 , focusing on the use of electricity and vehicle fuel (gasoline and diesel fuel) , which accounts for the majority of our total emissions .	0	0	0	0	0	0	1	0	1	0	0	1	1	1	1	0	1
…	…	…																
551	By using this summarized data going forward , we will strive to provide a suitable and comfortable work environment .	0	0	0	0	1	0	0	1	0	1	0	0	0	0	0	0	0
552	In addition , with the cooperation of a third - party organization (Caux Round Table Japan * 1) , we have begun operating a grievance process system in accordance with global standards .	0	0	0	0	1	0	0	1	0	1	0	0	0	0	0	0	0
…	…	…																

Figure 3.6: Example of Our Dataset

each sentence on the source document; “1” if it matched the sentence in the summary text, else ”0”. See Figure 3.6.

We perform the whole process of dataset creation with one person and it takes approximately 30 minutes per report, with all reports taking approximately 120 hours.

3.4 Statistics

	Query	No. of Sentences	Ratio
Document	-	173,664	1.00
Summary	1	1,493	0.01
	2	1,338	0.01
	3	8,891	0.05
	4	3,932	0.02
	5	6,201	0.04
	6	2,849	0.02
	7	6,938	0.04
	8	10,217	0.06
	9	8,102	0.05
	10	4,522	0.03
	11	6,078	0.03
	12	9,676	0.06
	13	8,761	0.05
	14	2,985	0.02
	15	4,482	0.03
	16	3,815	0.02
	17	6,630	0.04
Average	-	5,701	0.03

Table 3.1: Statistics of Our Dataset

The statistics of our dataset is shown in Table 1. The total number of sentences in all document is 173,664, and the average number of summary sentences per query is 5,701, which represents 3% of all document. This fact indicates that the dataset is imbalanced. The number of summary sentences also varies by query. The highest number is 10,217 in No. 8, and the lowest is 1,338 in No. 2.

Compared to DUC 2005-2007, the number of sentences per document is 26 in DUC 2005-2007, while in our dataset it is 695, indicating that the documents have a large number of sentences per document. See Table 3.2.

Number of	DUC 2005-2007	Our Dataset
(A) Documents	3,968	250
(B) Sentences in Total	102,820	173,664
(C) Sentences per Document (=(B)/(A))	26	695
(D) Query per Document	1	17
(E) Sentences in Summary Text	1,961	96,910
(F) Summary Sentences per Query (=(E)/(D))	1,961	5,701

Table 3.2: Comparison between DUC 2005-2007 and Our Dataset

In addition, the number of queries per document is one in DUC 2005-2007, whereas in our dataset there are 17 queries per document.

3.5 Characteristics

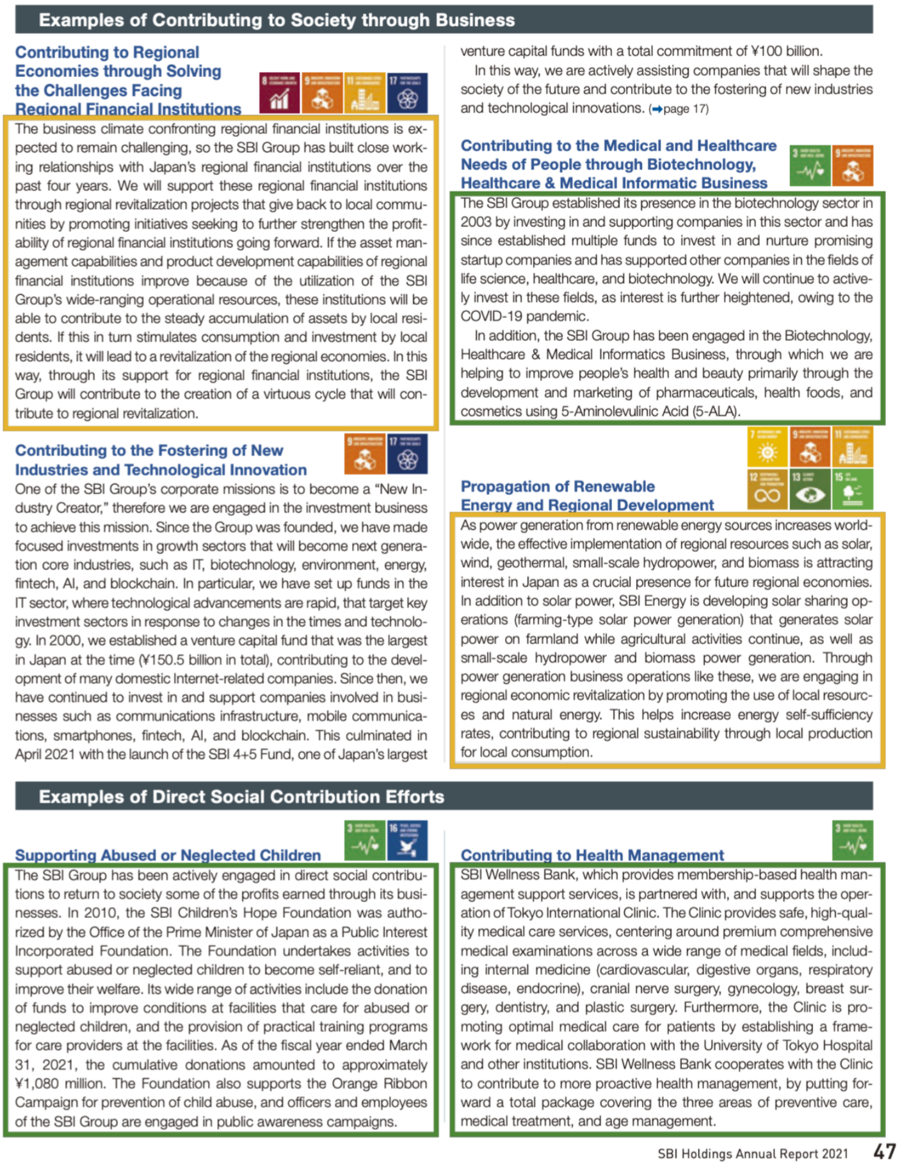
Unlike news articles dataset, the summary text in this dataset does not tend to appear at the beginning of the source document and it tends to appear anywhere in the source document. In addition, there are also no constraints on the length of the summary. The summary text may cover several pages in the original PDF, or it may be only a few sentences. It is also possible that the summary text of one query may indicate the same text as that of another query.

Another characteristic of our dataset is that the summary text is a set of sentence sequences rather than a set of scattered sentences. This is because labels are assigned to certain areas in the documents. This area may be seen as an article, but the document does not consist of articles because there is no boundary of articles within the document. Therefore, we may not take a topic model approach to determine whether or not each article in a document is eligible for summary. See Figure 3.7 for the sample page representing the characteristic.

3.6 Consistency

As mentioned above, the labeling is done by the issuing companies and there is no claer standards on the labelling shared among them. Therefore, labeling is not always consistent. For some companies, not all the initiatives are labeled in their integrated reports. For example, in GS Yuasa 2019 and Sumitomo Forestry 2020, the SDGs initiatives and ESG initiatives are respectively presented. Even though the SDGs initiatives and ESG initiatives have much in common, the ESG initiatives are not labeled with the goals of SDGs, while SDGs initiatives are labeled.

In addition, there is a room for broad interpretation of the definition of goals, leading to a lack of consistency. For example, Goal No. 8 ”Promote sustained, inclusive and sustainable economic growth, full and productive employment and decent work for all” includes a goal for ”economic growth” as well as a goal for ”decent work”. Since any companies aim for economic growth through their business, Goal No. 8 could be labeled broadly. In addition, Goal No. 8 is labeled for the statement regarding ”decent work”, which makes it difficult to keep consistency.



Examples of Contributing to Society through Business

Contributing to Regional Economies through Solving the Challenges Facing Regional Financial Institutions

The business climate confronting regional financial institutions is expected to remain challenging, so the SBI Group has built close working relationships with Japan's regional financial institutions over the past four years. We will support these regional financial institutions through regional revitalization projects that give back to local communities by promoting initiatives seeking to further strengthen the profitability of regional financial institutions going forward. If the asset management capabilities and product development capabilities of regional financial institutions improve because of the utilization of the SBI Group's wide-ranging operational resources, these institutions will be able to contribute to the steady accumulation of assets by local residents. If this in turn stimulates consumption and investment by local residents, it will lead to a revitalization of the regional economies. In this way, through its support for regional financial institutions, the SBI Group will contribute to the creation of a virtuous cycle that will contribute to regional revitalization.

Contributing to the Fostering of New Industries and Technological Innovation

One of the SBI Group's corporate missions is to become a "New Industry Creator," therefore we are engaged in the investment business to achieve this mission. Since the Group was founded, we have made focused investments in growth sectors that will become next generation core industries, such as IT, biotechnology, environment, energy, fintech, AI, and blockchain. In particular, we have set up funds in the IT sector, where technological advancements are rapid, that target key investment sectors in response to changes in the times and technology. In 2000, we established a venture capital fund that was the largest in Japan at the time (¥150.5 billion in total), contributing to the development of many domestic Internet-related companies. Since then, we have continued to invest in and support companies involved in businesses such as communications infrastructure, mobile communications, smartphones, fintech, AI, and blockchain. This culminated in April 2021 with the launch of the SBI 4+5 Fund, one of Japan's largest venture capital funds with a total commitment of ¥100 billion.

In this way, we are actively assisting companies that will shape the society of the future and contribute to the fostering of new industries and technological innovations. (➡page 17)

Contributing to the Medical and Healthcare Needs of People through Biotechnology, Healthcare & Medical Informatic Business

The SBI Group established its presence in the biotechnology sector in 2003 by investing in and supporting companies in this sector and has since established multiple funds to invest in and nurture promising startup companies and has supported other companies in the fields of life science, healthcare, and biotechnology. We will continue to actively invest in these fields, as interest is further heightened, owing to the COVID-19 pandemic.

In addition, the SBI Group has been engaged in the Biotechnology, Healthcare & Medical Informatics Business, through which we are helping to improve people's health and beauty primarily through the development and marketing of pharmaceuticals, health foods, and cosmetics using 5-Aminolevulinic Acid (5-ALA).

Propagation of Renewable Energy and Regional Development

As power generation from renewable energy sources increases worldwide, the effective implementation of regional resources such as solar, wind, geothermal, small-scale hydropower, and biomass is attracting interest in Japan as a crucial presence for future regional economies. In addition to solar power, SBI Energy is developing solar sharing operations (farming-type solar power generation) that generates solar power on farmland while agricultural activities continue, as well as small-scale hydropower and biomass power generation. Through power generation business operations like these, we are engaging in regional economic revitalization by promoting the use of local resources and natural energy. This helps increase energy self-sufficiency rates, contributing to regional sustainability through local production for local consumption.

Examples of Direct Social Contribution Efforts

Supporting Abused or Neglected Children

The SBI Group has been actively engaged in direct social contributions to return to society some of the profits earned through its businesses. In 2010, the SBI Children's Hope Foundation was authorized by the Office of the Prime Minister of Japan as a Public Interest Incorporated Foundation. The Foundation undertakes activities to support abused or neglected children to become self-reliant, and to improve their welfare. Its wide range of activities include the donation of funds to improve conditions at facilities that care for abused or neglected children, and the provision of practical training programs for care providers at the facilities. As of the fiscal year ended March 31, 2021, the cumulative donations amounted to approximately ¥1,080 million. The Foundation also supports the Orange Ribbon Campaign for prevention of child abuse, and officers and employees of the SBI Group are engaged in public awareness campaigns.

Contributing to Health Management

SBI Wellness Bank, which provides membership-based health management support services, is partnered with, and supports the operation of Tokyo International Clinic. The Clinic provides safe, high-quality medical care services, centering around premium comprehensive medical examinations across a wide range of medical fields, including internal medicine (cardiovascular, digestive organs, respiratory disease, endocrine), cranial nerve surgery, gynecology, breast surgery, dentistry, and plastic surgery. Furthermore, the Clinic is promoting optimal medical care for patients by establishing a framework for medical collaboration with the University of Tokyo Hospital and other institutions. SBI Wellness Bank cooperates with the Clinic to contribute to more proactive health management, by putting forward a total package covering the three areas of preventive care, medical treatment, and age management.

SBI Holdings Annual Report 2021 47

Figure 3.7: Sample page from an integrated report (SBI Holdings, Inc., 2021). We have outlined the text area to which Goal No.3 and No.11 is assigned with green border lines and orange border lines to indicate the extent of the summary text. The summary text is not a set of scattered sentences, rather a set of sentence sequences.

Chapter 4

Proposed Methods

In this chapter, we introduce the high level concept of our solution strategy, followed by the detail architecture and the implementation of our proposed methods.

4.1 Solution Strategy

In our dataset, unlike DUC 2005-2007 [1] and QMSum [2], queries are fixed throughout the dataset. Given the characteristic our our dataset, the problem we solve in this study can be viewed as a multiclass problem, where for each sentence we classify which of 17 goals it falls under, or none of them. our solution strategy is to simplify the problem by viewing the multiclass problem as multiple two-class problems. This strategy is called One-vs-Rest strategy, and we employ this strategy to apply a generic summary extractor to a query-focused summary extractor. We explain the detail architecture and implementation of the query-focused summary extractor in section 4.2. For comparison with the query-focused summary extractors, we apply the QA task method to our problem. We explain the detail architecture and implementation of the QA task application in section 4.3.

4.2 Multi-BERTSum

As discussed in the previous section, our solution strategy is to apply a generic summary extractor to a query-focused summary extractor. We employ BERTSum [12] for the generic summary extractor. BERTSum is a generic summary extractor that returns binary classification results whether each sentence is a summary or not and the binary classification fits for our

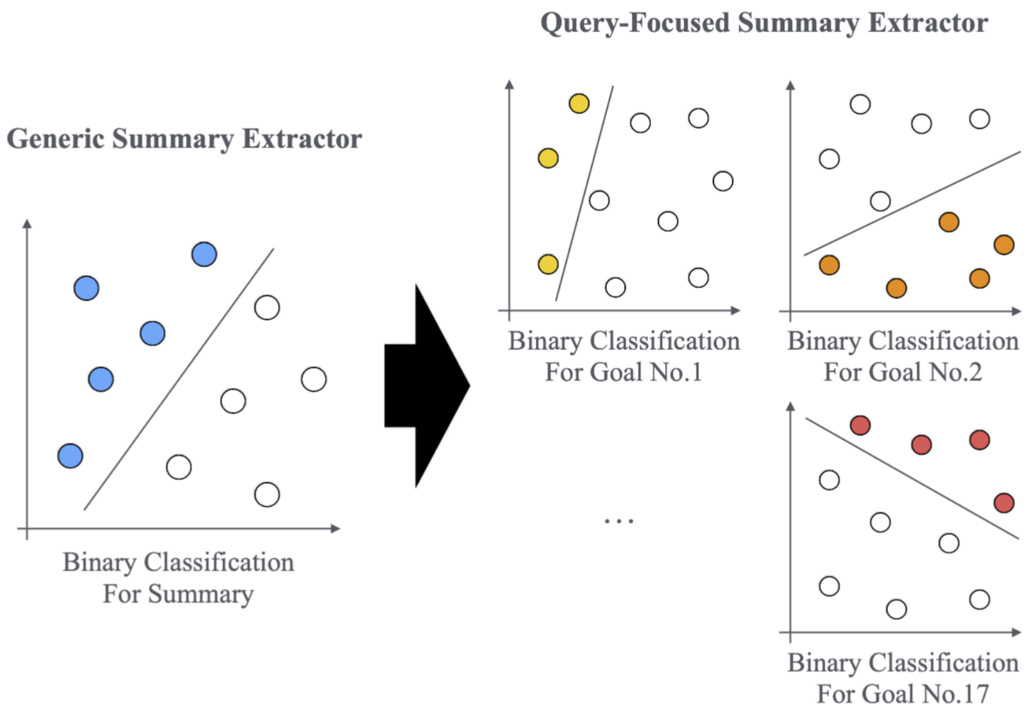

Query-Focused Summary Extractor
Generic Summary Extractor
Binary Classification For Summary
Binary Classification For Goal No.1
Binary Classification For Goal No.2
...
Binary Classification For Goal No.17

Figure 4.1: One-vs-Rest Strategy for Query-Focused Summary Extractor

strategy. The architecture of our proposed method consists of multiple encoding layers incorporating a pretrained BERT model and multiple classification layers incorporating Transformer's encoding layer, in accordance with original BERTSum architecture. Each layer is multiplexed by each query. See Figure 4.2 for the architecture.

4.2.1 Input Implementation

The input representation of each token is constructed by summing the corresponding token, segmentation, and positional embeddings. Following the implementation of Yang et al. [12], the token embedding inserts a [CLS] token at the beginning and a [SEP] token at the end to delineate the boundaries of each sentence. In segment embeddings, "1" and "0" are given every other sentence. In positional embeddings, the absolute position of each token in the input sequence is given. This architecture needs to take into account the input sequence length constraint and we adopt the Sliding Window approach as explained above.

Since BERT limits the maximum number of tokens to 512 in a single

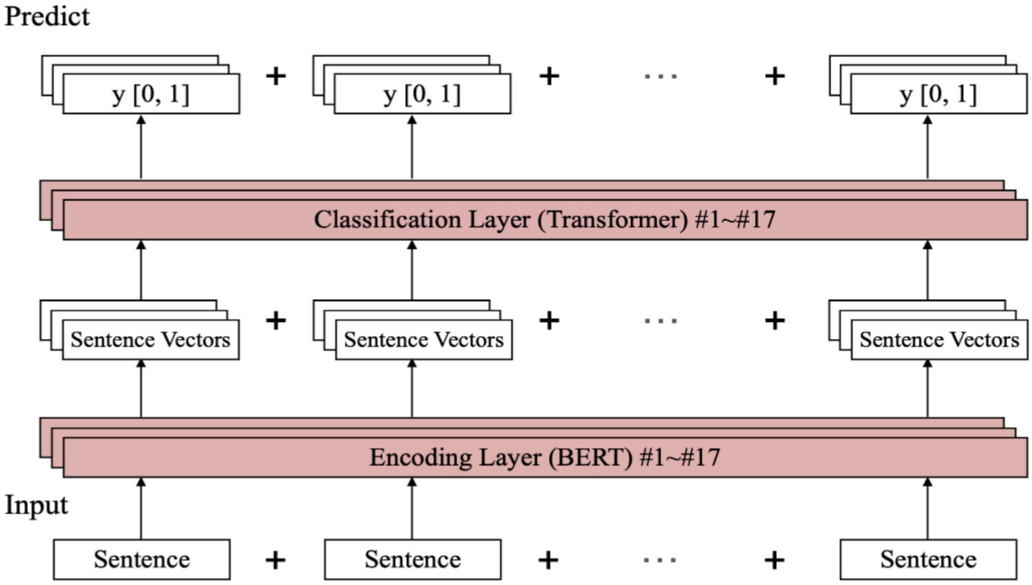
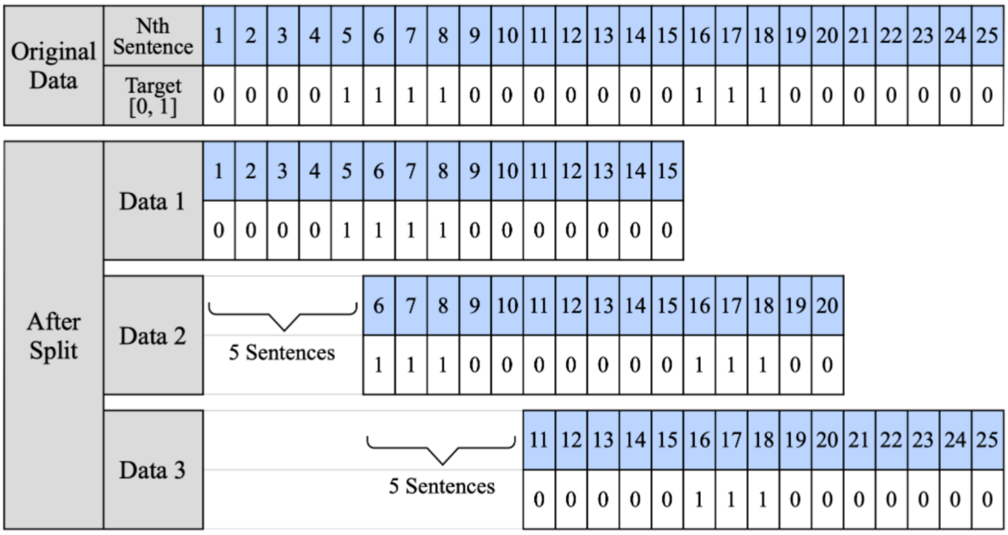


Figure 4.2: Architecture of Multi-BERTSum with Transformer Classifier

input, if the input document includes more than 512 tokens, you need to adjust the way it is given. Since the documents of our dataset are huge in volume and every document is significantly longer than 512 tokens in our dataset, we need to consider the solution. A commonly used approach to deal with the constraint is to split the documents into several individual chunks and predict responses from each chunk separately. However, simply splitting a document into chunks may result in the loss of context near the boundaries. Therefore, we adopt ”Sliding Window” approach proposed by Wang et al. [19], where the input document is split by shifting a certain length. Note that we set 5 sentences to the sliding length in splitting the documents in our implementation. See Figure 4.3 for the high level concept of ”Sliding Window”.

4.2.2 Model Implementation

In encoding layer, we incorporate uncased version of pretrained BERT for finetuning. Once the input representation is passed through the encoding layer, the output sentence vectors are given to the classification layer with Transformer encoding layer to obtain the probability that the sentence is a summary sentence for each query. The probabilities at the classification layer are formulated as follows. We use Adam [18] as the optimizer with learning



Original Data	Nth Sentence	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
	Target [0, 1]	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0
After Split	Data 1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15										
		0	0	0	0	1	1	1	1	0	0	0	0	0	0	0										
	Data 2						6	7	8	9	10	11	12	13	14	15	16	17	18	19	20					
				5 Sentences			1	1	1	0	0	0	0	0	0	0	1	1	1	0	0					
	Data 3											11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
									5 Sentences			0	0	0	0	0	1	1	1	0	0	0	0	0	0	0

Figure 4.3: High Level Concept of Sliding Window (Ours)

rate of 0.002, and use binary cross entropy loss for the loss function.

$$P(s_i|q_j, D) = sigmoid(W_j\ Transformer(h_i^L) + b_j) \tag{4.1}$$

4.2.3 Predictor Implementation

Since the Sliding Window method is adopted for this model, sentence overlap occurs when integrating the split sentences at the document level. See Figure 4.3 for the overlapped sentences in 6th sentence to 20th sentence. Thus, it is necessary to select single ones from the overlapped ones. To solve the problem, we apply a scoring method where a sentence with highest socre are selected from the overlapped ones. We devise 3 types of scoring method; one is to score the degree of centrality in sentence sequence, the second is to score the closeness to the top of sentence sequence, and the third is to score the closeness to the bottom of sentence sequence. We call it the scoring methods as ”alignment” (”center”, ”top”, and ”bottom” for each) For example, in the instance of Figure 4.3, 6th to 20th sentences are overlapped among Data 1, Data 2, Data 3. In case of ”center” alignment, 6th to 10th sentences of Data 1, 11th to 15th sentences of Data 2, and 16th to 20th sentences of Data 3 are selected. We explore the optimal alignment for each model by query through the experiments (section 6.1).

4.2.4 Threshold

In text summarization, there is typically a restriction on the length of sentences extracted as a summary. However, there is no restriction on the length of sentences in our dataset. Therefore, a boundary is required to determine whether a sentence should be considered as a summary or a non-summary. We define the boundary as threshold T; if a returned value from a model for a sentence exceeds threshold T, the sentence are selected as summary. The threshold T and the predicted value y are formulated as follows. Here, s denotes sentence and q denotes query. We explore the optimal T for each model by query through the experiments (section 6.1).

$$y = \begin{cases} 1 & (P(s_i|q_j, D) \geqq T) \\ 0 & (P(s_i|q_j, D) < T) \end{cases} \tag{4.2}$$

4.2.5 Simple Classifier

To compare the performance in classification layer with Transformer encoder, we build another model implementing simplified classifier with linear layer and sigmoid functions instead of Transformer.

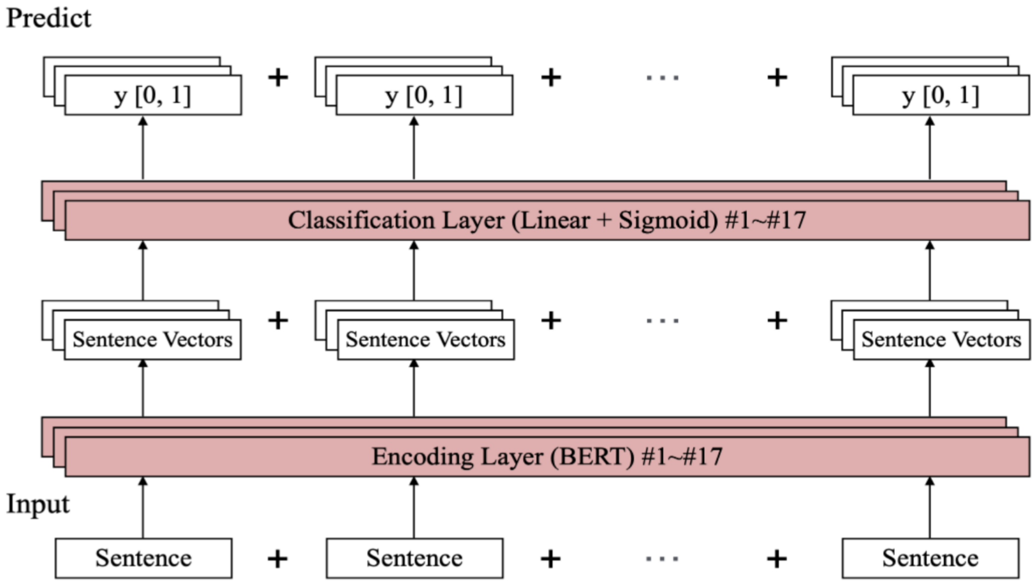


Figure 4.4: Architecture of Multi-BERTSum with Simple Classifier

4.3 Multi-Span-Selector

In this section, we consider a solution to solve our problem as a Question Answering (QA) problem. In QA problem such as SQuAD [3] and TriviaQA [4], documents and questions are given to a model and the model returns answer spans corresponding to the questions from the documents. Our dataset is characterized by the fact that summary targets are clustered together in a certain area of given document. Although QA problems normally assumes a few words, we apply the approach of span selection to sentence span selection in our dataset.

The architecture of the sentence span selector consists of multiple encoding layers incorporating a pretrained BERT model and multiple classification layers incorporating linear span selection layer. Span selection layer returns the predicted start position and end position of summary sentences. Each layer is multiplexed by each query, in accordance with Muti-BERTSum models.

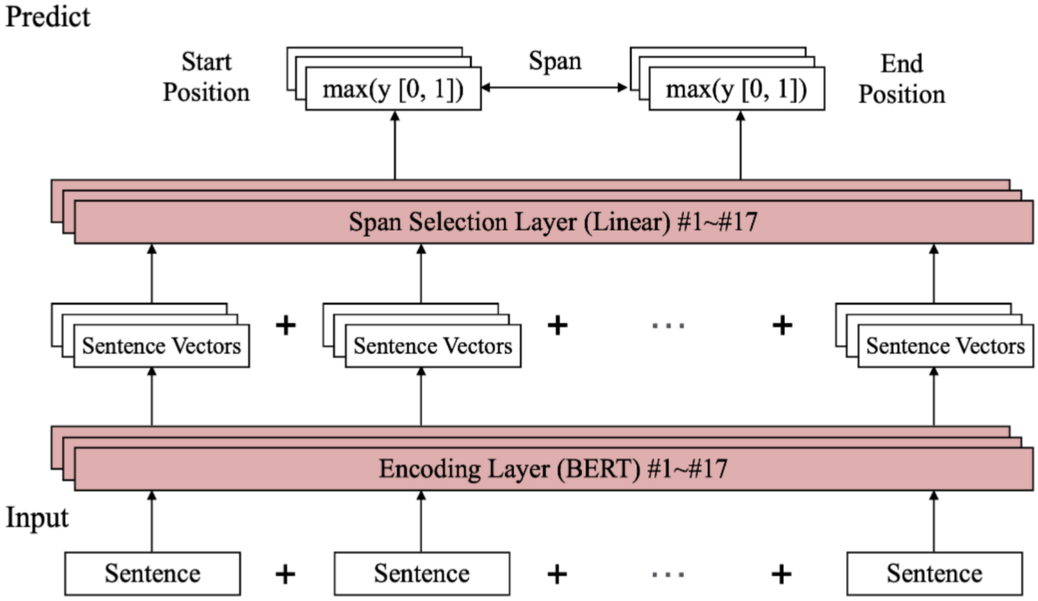


Figure 4.5: Architecture of Multi-Span-Selector

4.3.1 Input Implementation

Since encoding layer of the span selector is the same architecture of BERTSum, the input representation is also the same as BERTSum. See section 4.2.1.

4.3.2 Model Implementation

Once the input representation is passed through the encoding layer, the sentence vectors are derived from the layer. The sentence vectors are passed to the span selection layer, where each sentence vector is converted to two probabilities; one is the probability of being the start position of a summary sentences span and the other is the probability of being end position of a summary sentences span.

In this method, in case there is no summary target for the input, it is still necessary to express the absence of the summary target as a sentences span. To solve this problem, we set the the last sentence of the input as the start position of a sentences span and the first sentence of the input as the end position of a sentences span.

4.3.3 Predictor Implementation

Once the list of two probabilities are derived from the span selection layer, the sentence of highest probability for the start position and the end position is selected respectively to indicate the summary sentences span from the input. To identify the absence of the summary sentences span, we compare the absolute position of the start position and the end position. If the end position is smaller or equal to the start position we identify the summary sentences span is absent for the input.

Chapter 5

Baselines

For experimental purpose, we compare our proposed methods with several baselines. In this chapter, we introduce 2 unsupervised methods and 2 supervised methods and how we apply those baseline methods to our dataset with the detail implementation.

5.1 LEAD

As one of the baselines for text summarization, we apply "LEAD" method, a widely known extractive summary baseline, in which leading sentences of a document are selected as predictions within the range of summary length. For news articles summary datasets such as CNN/DailyMail news highlights dataset [15], the LEAD method is widely used as a baseline as summary text tends to appear at the beginning of the document.

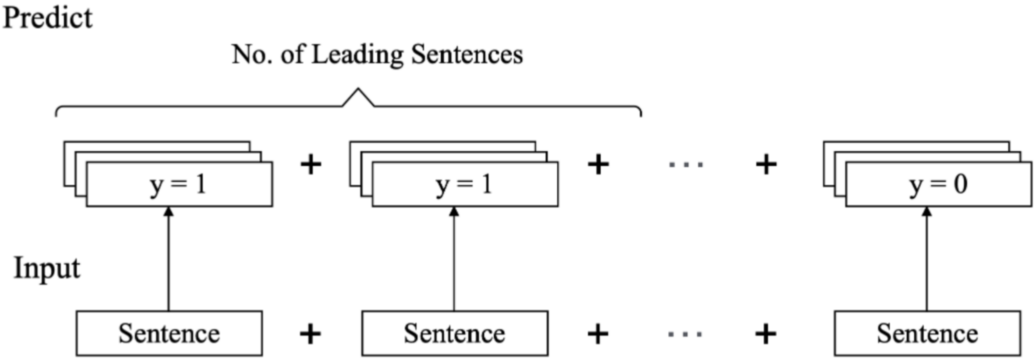


Figure 5.1: Architecture of LEAD

Since there are no limitations on the summary length in our dataset, we explore the length of the leading sentences through experiments. Based on the experimental results, the length of the leading sentence with the highest F1 score is passed to the model.

5.2 MMR

The other baseline we apply is maximum marginal relevance (MMR) [5], a model that seeks to reduce redundancy while maintaining query relevance through ranking documents and selecting appropriate sentences for text summarization. The MMR is formulated as follows;

$$MMR \overset{\text{def}}{=} \underset{D_i \in R \backslash S}{\arg\max} \left[\lambda Sim_1(D_i, Q) - (1-\lambda) \max_{D_j \in S} Sim_2(D_i, D_j) \right] \tag{5.1}$$

D is the document collection, Q is the query, and R is the list of sentences already ranked by the IR system. S is the set of sentences already selected as summary in R, R§is the set difference, i.e., the set of documents not yet selected as S in R, and Sim is the similarity metric used in document retrieval and relevance ranking between documents and queries. From the above definition, MMR computes the standard relevance rank when the parameter $\lambda = 1$ and the maximum diversity rank among documents in R when $\lambda = 0$. Here, we apply TFIDF to represent D and Q in vectors and apply cosine similarity to implement Sim1 and Sim2 [16].

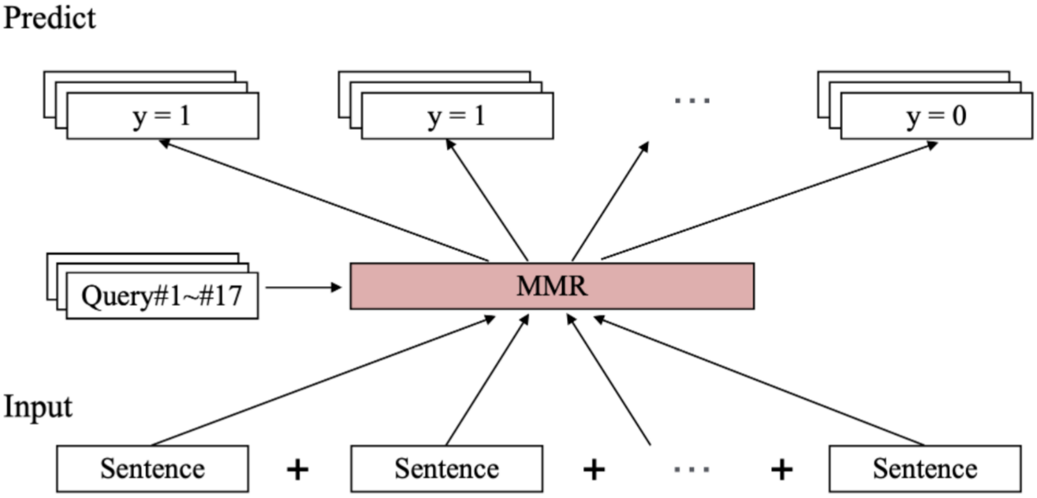


Figure 5.2: Architecture of MMR

In MMR, the length of a summary sentence is a given one, and the highest scoring sentences are selected as summary sentences from the top to the given length. Since there are no constraints on sentence length in this dataset, we explore the optimal sentence length through the validation data.

5.3 Sentence BERT

Sentence BERT [17] is a model that applies a pretrained BERT [10]network that derives semantically meaningful sentence embeddings using a Siamese network structure. This architecture significantly reduces the effort required to find most-similar pairs while maintaining the accuracy of BERT. We apply Sentence BERT to our query-focused summarization task.

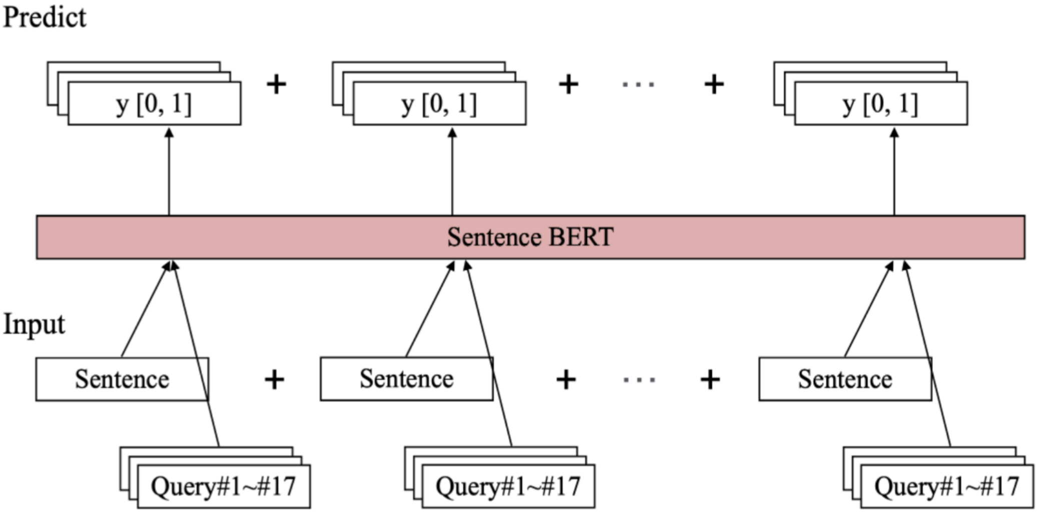


Figure 5.3: Architecture of Sentence BERT

5.3.1 Input Implementation

For the use case of Sentence BERT inference, sentence-sentence pairs are given to the model and the cosine similarity of each pair is derived. In our study, we create sentence-query pairs from our dataset for all the combination of sentences and 17 queries and pass the pairs to Sentence BERT model.

5.3.2 Model Implementation

For our method, we finetune the pretrained model of Sentence BERT using the training data of our dataset. We use all-mpnet-base-v2 for the pretrained model of BERT. The inference process is trained by giving a value of [0, 1] as a target label. If the sentence is a summary for a query, we give the similarity label as ”1” for the sentence-query pair, else we give ”0”.

5.3.3 Predictor Implementation

We obtain the cosine similarities for all the combination of sentences in a document and 17 queries. The cosine similarities of sentence-query pair are then used to determine whether the sentence are summary or non-summary. For this model, threshold T needs to be defined to determine whether the derived probabilities indicate summary or non-summary. See section for the concept of threshold. We explore the optimal T by query through the experiments (see section 6.1).

5.4 BERT-Base

BERT-Base is a model based on the architecture built by Zhu et al. [9] who proposed a model using BERT for query-focused summarization. We apply this architecture to our task. This architecture consists of a pretrained BERT-based encoding layer and a linear classification layer.

5.4.1 Input Implementation

Following standard BERT practice, the input representation of each token is constructed by summing the corresponding token, segmentation, and position embeddings. Following the implementation of Zhu et al. [9], the query token sequence is concatenated with the sentence token sequence and a [L1] token is inserted before the query token sequence. In each sentence, a [CLS] token is inserted at the beginning and a [SEP] token at the end to draw a clear sentence boundary. In segment embeddings, [1] and [0] are given to indicate the query token sequence and sentence token sequence respectively. In positional embeddings, the absolute position of each token in the input sequence is given.

For BERT-Base, we also need to consider the limitation of token length as discussed in section 4.2. Since BERT limits the maximum number of tokens to 512 in a single input, we use ”Sliding Window” approach to adjust

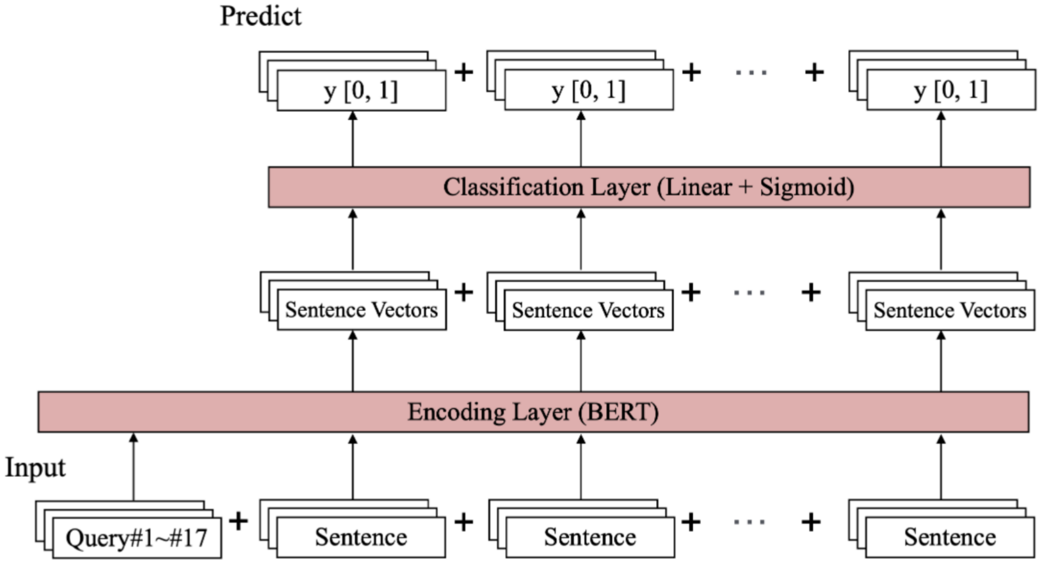


Figure 5.4: Architecture of BERT-Base

the input sequence to be within 512 length. For BERT-Base, query token sequence needs to be included in the 512 length of the input sequence.

5.4.2 Model Implementation

The input representation is passed to the pretrained BERT encoding layer and sentence vectors are derived as output. The sentence vectors are given to a classification layer that calculates the probability that the sentence is a summary sentence corresponding to the query. The classification layer consists of a linear layer and a sigmoid function. The probability is given to the [CLS] token representing each sentence in the document and not to the [L1] token representing the query. The probabilities at the classification layer are formulated as follows.

$$P(s_i|q_j, D) = sigmoid(Wh_i^L + b) \tag{5.2}$$

We use Adam [18] as the optimizer with learning rate of 0.002, and use binary cross entropy loss for the loss function, which are the same values as set with Multi-BERTSum with Transformer Classifier.

5.4.3 Predictor Implementation

Since the Sliding Window method is adopted for BERT-Base, sentence overlap occurs when integrating output the split sentences at the document level. Thus, corresponding output also overlaps when integrating it at the document level. Thus, it is necessary to select single output from the overlapped ones. In our proposed model, we introduced the concept of alignment, where we score the overlapped sentence positions and determine the single output based on the type of alignment methods. For BERT-Base, we use ”center” alignment method, following the implementation of Zhu et al. [9]. With ”center” alignment method, we score the degree of centrality of overlapped sentence position in the original sentences.

Since BERT-Base derives probabilities of being a summary sentence from the model, we need to define the threshold T for BERT-Base to determine whether the derived probabilities indicate summary or non-summary. See section for the concept of threshold. We explore the optimal T by query through the experiments (section 6.1).

Chapter 6

Experiments

In this chapter, we conduct experiments to explore the optimal parameters for each model through the validation data, followed by the performance evaluation on the models with optimal parameters. We introduce our analysis on the results of the performance evaluation.

6.1 Hyper-Parameters Exploration

The models explained in chapter 4 and 5 require hyper-parameters specific to our datasets. Therefore, we conduct experiments on the validation data and explore optimal hyper-parameters. We use F1-score to measure the performance in the exploration. Note that prior to the experiments, we split the 250 documents into training, validation, and test data at a ratio of 7:1.5:1.5.

6.1.1 Baselines

Hyper-parameters of baselines (LEAD, MMR, Sentence-BERT, and BERT-Base) are explored as follows;

LEAD

The summary length L is the hyper-parameter for LEAD. The summary length indicates the number of leading sentences. In experiment, we explore the length in the range of [1, 1,000] in increments of 1 for each query.

MMR

MMR requires two hyper-parameters: λ and summary length L. λ is a parameter that represents the weighting for the relevance rank and diversity rank, and sentence length is a parameter that represents the length of the summary sentences. In experiment, we explore the optimal λ in the range of [0, 1] and summary length in the range of [5, 300] in increments of 5 for each query.

Sentence BERT

In our Sentence BERT, the threshold T is the hyper-parameter. The threshold T is a boundary that determines whether the cosine similarity of sentence-query pair is summary or non-summary. In experiment, we explore the optimal T in the range of [0, 0.7] in increments of 0.01 for each query.

BERT-Base

In BERT-Base, we explore the optimal threshold T from the range of [0, 0.7] in increments of 0.01 for each query.

6.1.2 Proposed Methods

Hyper-parameters of our proposed methods (Multi-BERTSum with Transformer Classifier, Multi-BERTSum with Simple Classifier, and Multi-Span-Selector) are explored as follows;

Multi-BERTSum with Transformer Classifier

Multi-BERTSum with Transformer Classifier requires two hyper-parameters; threshold and alignment. Alignment indicates the degree of the position at which a particular sentence is selected from the overlapped sentences generated through Sliding Window method. There are three types of alignment; top, center, and bottom. we explore the optimal threshold T in the range of [0, 0.7] in increments of 0.01 and optimal alignment among the three options for each query.

Multi-BERTSum with Simple Classifier

Multi-BERTSum with Simple Classifier requires the same hyper-parameters as Multi-BERTSum with Transformer Classifier, although the optimal hyper-parameters are separately explored from Multi-BERTSum with Transformer Classifier.

Query	Unsupervised			Supervised	
	LEAD	MMR		Sentence BERT	BERT-Base
	L	λ	L	T	T
1	258	0.9	50	0.32	0.02
2	258	0.9	20	0.17	0.07
3	270	0.9	120	0.24	0.07
4	278	0.9	115	0.25	0.10
5	660	0.9	30	0.15	0.14
6	260	0.9	10	0.36	0.10
7	493	0.9	80	0.19	0.19
8	461	0.9	185	0.11	0.12
9	278	0.9	295	0.15	0.17
10	459	0.9	110	0.28	0.15
11	270	0.9	160	0.27	0.22
12	260	0.9	185	0.14	0.21
13	493	0.9	200	0.39	0.32
14	496	0.9	20	0.23	0.15
15	258	0.9	50	0.23	0.14
16	479	0.9	75	0.32	0.35
17	471	0.9	295	0.16	0.01

Table 6.1: Hyper-Parameters Explored for Baselines

Multi-Span-Selector

As Multi-Span-Selector does not have threshold T, hyper-parameter for the model to search is alignment. We explore the optimal alignment among the three options for each query.

6.2 Results of Performance Evaluation

Given the optimal hyper-parameters explored in the previous section, we experiment with each model through the test data and compare their performance. For fair performance evaluation, we compare the performance only among supervised methods. We evaluate the unsupervised methods for referential purpose.

We use F1-score to measure the performance on our dataset because it is basically a binary classification task that returns if the sentence is summary or non-summary for a query. In addition, F1-score is more appropriate than precision, recall, and accuracy in case the dataset is imbalanced. We show

Query	Multi-BERTSum				Multi-Span
	Transformer		Simple		-Selector
	T	A	T	A	T
1	0.01	bottom	0.05	center	bottom
2	0.12	top	0.07	top	bottom
3	0.02	center	0.01	center	bottom
4	0.18	center	0.10	center	center
5	0.07	center	0.02	center	center
6	0.07	center	0.13	bottom	center
7	0.22	center	0.06	center	center
8	0.08	center	0.09	center	center
9	0.01	center	0.02	center	center
10	0.03	center	0.15	bottom	center
11	0.01	center	0.19	center	top
12	0.10	center	0.26	center	top
13	0.05	center	0.01	center	bottom
14	0.02	center	0.07	center	center
15	0.05	center	0.60	center	bottom
16	0.17	center	0.27	bottom	top
17	0.02	center	0.02	center	bottom

Table 6.2: Hyper-Parameters Explored for Proposed Methods

the accuracy of the experimental results just for referential purpose.

F1-scores from the baselines are shown in Table 6.3 and the ones from the proposed methods are shown in Table 6.4. The accuracy scores from the baselines are shown in Table 6.5 and the ones from the proposed methods are shown in Table 6.6.

Among the unsupervised methods, F1-score of LEAD is 0.067 and the one with MMR is 0.093. On the other hand, among the Supervised methods, F1-score of Sentence BERT is 0.298 and the one with BERT-Base is 0.302. Among the proposed methods, F1-score of Multi-BERTSum with Transformer Classifier is 0.379 and the one with Multi-BERTSum with Simple Classifier is 0.389, and the one with Multi-Span-Sellector is 0.350.

As a result, Multi-BERTSum with Simple Classifier achieves best performance with F1-score of 0.389. This result indicates that the proposed method outperforms BERT-Base, which is the best baseline, by 30%. The other proposed methods such as Multi-BERTSum with Transformer Classifier and Multi-Span-Selector also significantly outperforms the baseline. This results indicates the effectiveness of One-vs-Rest strategy on our dataset. Multi-BERTSum with Simple Classifier outperforms Multi-Span-Selector by

Query	Unsupervised		Supervised	
	LEAD	MMR	Sentence BERT	BERT-Base
1	0.015	0.017	0.042	0.078
2	0.008	0.054	0.175	0.289
3	0.058	0.087	0.279	0.237
4	0.036	0.075	0.269	0.286
5	0.065	0.115	0.309	0.360
6	0.012	0.125	0.424	0.375
7	0.094	0.113	0.328	0.375
8	0.120	0.131	0.287	0.329
9	0.079	0.097	0.317	0.365
10	0.050	0.041	0.253	0.256
11	0.044	0.072	0.253	0.277
12	0.087	0.125	0.330	0.360
13	0.148	0.116	0.402	0.427
14	0.053	0.079	0.336	0.338
15	0.045	0.095	0.349	0.375
16	0.037	0.045	0.178	0.048
17	0.082	0.084	0.173	0.182
Total	0.067	0.093	0.298	0.302

Table 6.3: F1-Scores of Baselines on Our Dataset

11%, which indicates query-focused text summarization approach outperforms QA task approach.

6.3 Analysis

6.3.1 Non-standardized Annotation

Looking at the experimental results, we identify that there is a variation in performance across queries in any method. For example, the results of Multi-BERTSum with Simple Classifier show F1 score of 0.177 for query 1 and 0.611 for query 6. This performance gap is considered to be due to the characteristics of this dataset. One of the characteristics is that the annotation was not made in a standardized approach. As mentioned in section 3.1, corporate IRs in each company individually made the annotation, which results in inconsistent labelling among the publishing companies.

Query	Multi-BERTSum		Multi-Span
	Transformer	Simple	-Selector
1	0.156	0.177	0.142
2	0.240	0.253	0.075
3	0.299	0.364	0.271
4	0.403	0.393	0.434
5	0.362	0.421	0.370
6	0.588	0.611	0.519
7	0.467	0.455	0.433
8	0.361	0.345	0.349
9	0.409	0.367	0.309
10	0.209	0.202	0.235
11	0.317	0.300	0.244
12	0.454	0.436	0.410
13	0.523	0.542	0.470
14	0.428	0.439	0.379
15	0.467	0.499	0.395
16	0.305	0.360	0.316
17	0.205	0.197	0.164
Total	0.379	0.389	0.350

Table 6.4: F1-Scores of Proposed Methods on Our Dataset

6.3.2 Complexity in Understanding Query Relevance

Some of the SDGs goals (No.3, 8, 9, 10, 16, and 17) have room for broad interpretation, which may lead to also inconsistent labeling. In fact, the performance for those SDGs goals are relatively low in performance. In case of Multi-BERTSum with Simple Classifier, for instance, the results show No.3 with 0.364, No.8 with 0.345, No.9 with 0.368, No.10 with 0.202, and No.17 with 0.197. The other characteristics is that as mentioned in Table the dataset is imbalanced and there are relatively fewer labeled summary for goal No.1 and No.2, and as a result, models are considered not fully trained, and the performance does not improve.

6.3.3 Architectural Impact on Performance

Architectural difference also contributes to making an difference in performance. The architectural structures of BERT-Base and Multi-BERTSum with Simple Classifier are almost the same, except for whether One-vs-Rest strategy is applied or not. Therefore, the performance difference between

Query	Unsupervised		Supervised	
	LEAD	MMR	Sentence BERT	BERT-Base
1	0.618	0.920	0.989	0.971
2	0.615	0.965	0.984	0.987
3	0.588	0.800	0.926	0.873
4	0.589	0.825	0.966	0.969
5	0.209	0.936	0.939	0.957
6	0.605	0.972	0.981	0.962
7	0.356	0.861	0.932	0.916
8	0.389	0.714	0.885	0.879
9	0.577	0.563	0.916	0.897
10	0.376	0.825	0.963	0.954
11	0.591	0.751	0.945	0.929
12	0.603	0.715	0.905	0.885
13	0.375	0.684	0.933	0.922
14	0.341	0.954	0.973	0.963
15	0.616	0.913	0.965	0.949
16	0.347	0.875	0.973	0.978
17	0.366	0.567	0.925	0.736
Total	0.480	0.814	0.947	0.925

Table 6.5: (Reference)Accuracy of Baselines on Our Dataset

BERT-Base and Multi-BERTSum with Simple Classifier can be attributed to One-vs-Rest strategy. The effectiveness of One-vs-Rest strategy is considered to be due to the efficiency of the models as each model is dedicated to the feature of the query.

6.3.4 Document Splitting Approach

The performance of Multi-Span-Selector is not as effective as Multi-BERTSum (Transformer Classifier and Simple Classifier). Our implementation of Multi-Span-Selector only identifies single span for the input sentences, although multiple spans are supposed to be selected for an input in some cases. Current implementation of document splitting considers only token length. However, we need to consider another document splitting approach to improve the performance for Multi-Span-Selector.

Query	Multi-BERTSum		Multi-Span
	Transformer	Simple	-Selector
1	0.983	0.984	0.989
2	0.988	0.989	0.990
3	0.901	0.919	0.892
4	0.976	0.975	0.979
5	0.955	0.951	0.950
6	0.988	0.987	0.986
7	0.950	0.945	0.939
8	0.891	0.896	0.909
9	0.915	0.917	0.918
10	0.967	0.963	0.961
11	0.940	0.927	0.958
12	0.932	0.930	0.943
13	0.941	0.931	0.921
14	0.968	0.968	0.960
15	0.974	0.977	0.965
16	0.971	0.974	0.975
17	0.930	0.934	0.949
Total	0.951	0.951	0.952

Table 6.6: (Reference)Accuracy of Proposed Methods on Our Dataset

6.3.5 Difficulties in Understanding Long Document

Multi-BERTSum with Transformer Classifier fell slightly short of that with Simple Classifier. It implies Transformer architecture in classifier layer does not contribute to performance improvement, although Transformer Classifier outperforms Simple Classifier in the original experiments of BERTSum [12]. Because the documents in our dataset are long (Table 3.2), we split one document into many sub-documents and pass them to the model, given the input length constraints of BERT. Due to the data imbalanceness, summary text are rarely found in input sentences. Even in case it includes summary text, they often exceed 512 tokens. The dataset used in the original BERTSum is a complete document within a single input, allowing Transformer to understand the entire document. Unlike such a dataset, our Transformer does not cover the entire context for a single input and it could be a reason that our Transformer Classifier does not outperform Simple Classifier.

6.3.6 Dataset Structual Characteristics

In our dataset, the summary text appears neither in the upfront section nor in the last section of the document, but rather in the middle section of the document in many cases. The result of "LEAD" method indicates that summary does not exist in the upfront area of the dataset. See table 6.3 and 6.5. Company's business introduction is stated in the upfront section and financial information is stated in the last section of the document. However, since the middle section has a very large amount of text, it is still a difficult problem to identify the summary text by its structure alone.

Chapter 7

Conclusion and Future Work

In this study, we propose a novel dataset of query-focused summarization with multi-topic document and propose a novel summary extractor that generates topic-by-topic summary for the dataset. The proposed method achieved 30% improvement over the existing baseline methods applied to the dataset. For future work, we would like to address the following points;

7.1 Dataset

The current dataset consists of 250 documents, which is a limited number for deep learning methods. In particular, our dataset is imbalanced data with only about 3% of the source sentences being target summary sentences, and the summary sentences for certain queries are very small. Adding more documents to the dataset could improve the performance of the model. In addition, as analyzed in section 6.3.1, labeling by corporate IR is not strictly standardized across the integrated reports. By incorporating certain standardization methods in the annotation work we expect to improve the performance of each methods due to the consistency of dataset.

7.2 Understanding Long Document Structures

The current methods simply divides long documents into sub-documents by a certain number of tokens. In this approach, the sub-documents themselves are not organized into a certain semantic cohesion, preventing the model from understanding the sub-documents. To solve this problem, we would like to verify whether performance can be improved by dividing the sentence structure into paragraph units and then passing them to the model as input sub-documents.

Bibliography

[1] Hoa Trang Dang. Overview of duc 2005. Proceedings of the document understanding conference, volume 2005, pp. 1–12 (2005)

[2] Ming Zhong, Da Yin, Tao Yu, Ahmad Zaidi, Mutethia Mutuma, Rahul Jha, Ahmed Hassan Awadallah, Asli Celikyilmaz, Yang Liu, Xipeng Qiu, and Dragomir Radev.: QMSum: A New Benchmark for Query-based Multi-domain Meeting Summarization. Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 5905–5921 (2021)

[3] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ Questions for Machine Comprehension of Text. Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 2383–2392 (2016)

[4] Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 1601–1611 (2017)

[5] Jaime Carbonell and Jade Goldstein. The use of mmr, diversity-based reranking for reordering documents and producing summaries. Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR ’98, pp. 335–336 (1998)

[6] You Ouyang, Wenjie Li, Sujian Li, and Qin Lu. The use of user modelling to guide inference and learning. Information Processing & Management, 47(2), pp. 227–237 (2011)

[7] Ziqiang Cao, Wenjie Li, Sujian Li, Furu Wei, and Yanran Li. AttSum: Joint learning of focusing and summarization with neural attention.

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pp. 547–556 (2016)

[8] Pengjie Ren, Zhumin Chen, Zhaochun Ren, Furu Wei, Jun Ma, and Maarten de Rijke. Leveraging contextual sentence relations for extractive summarization using a neural attention model. IProceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '17, pp. 95–104 (2017)

[9] H. Zhu, L. Dong, F. Wei, B. Qin, T. Liu. Transforming Wikipedia into Augmented Data for Query-Focused Summarization. arXiv preprint arXiv:1911.03324 (2019)

[10] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pp. 4171–4186 (2019)

[11] Xingxing Zhang, Furu Wei and Ming Zhou. HIBERT: Document Level Pre-training of Hierarchical Bidirectional Transformers for Document Summarization. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 5059–5069 (2019)

[12] Yang Liu and Mirella Lapata. Text summarization with pretrained encoders. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 3730–3740. Association for Computational Linguistics, Hong Kong, China (2019)

[13] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17, pp. 6000–6010 (2017)

[14] Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, Xuanjing Huang. Extractive Summarization as Text Matching. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 6197–6208 (2020)

[15] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. Advances in Neural Information Processing Systems, pp. 1693—1701 (2015)

[16] Xiaojun Wan and Jianguo Xiao. Graph-based multi-modality learning for topic-focused multi- document summarization. Proceedings of the 21st International Joint Conference on Artificial Intelligence, PP. 1586–1591 (2009)

[17] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 3982–3992 (2019)

[18] Diederick P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. International Conference on Learning Representations (ICLR) (2015)

[19] Zhiguo Wang, Patrick Ng, Xiaofei Ma, Ramesh Nallapati, and Bing Xiang. Multi-passage BERT: A Globally Normalized BERT Model for Open-domain Question Answering. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 5878–5882 (2019)