Developing a Corpus of Abstractive Text Summarization

Software Requirements Specification

Team Members

Aniket Sangwan - 180001005

Sarthak Jain - 180001047

Miten Shah - 180001049

Sundesh Gupta - 180001057

Prepared for

CS 258 Software Engineering

Spring 2020

Revision History

Date	Description	Comments
13/01/2020	Initial Version	

Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date

Table of Contents

I. Introduction

In this section, a review of the entire document is provided. The reader would get familiarized with the content before the further details are described.

A. Purpose

Now-a-days, research in the field of Artificial Intelligence & Machine Learning is growing at pace. For this research, we need datasets on which our model would be trained. Our Project is to create a dataset for Abstractive Text Summarization and also to create a model for the same.
The dataset can be used for further research and our model associated with User Interface, will be a end-product for Text Summarization.

B. Scope

1. Our Dataset will provide the corpus of article-summary pair. Various articles from different websites will be collected.
2. Approximate 5000 Number of Article in the dataset
3. It will primarily contain the following content
    a) Headline
    b) Body
    c) Date
    d) Source
4. Our portal will provide an interface for the user to upload / type the article, and the portal will return the prescribed headline for the article.
5. Our Project is mainly intended to be used for further research in Natural Language Processing.

C. Definitions, Acronyms and Abbreviations

Server : A program that awaits and fulfils requests from client programs in the same or other computers.
Deep Learning : A machine learning method that simulates the neural network in the human brain.
Classifier : It is a systematic approach to building classification models from an input data set.
Word2Vec : A matrix representation of words and their meaning similarities.
NLP: Natural Language Processing
Tokenizer : A component that breaks up a sequence of string into pieces such as words, phrases, symbols and other elements called tokens.
Stopword : These are words which are filtered out before processing natural language data.

D. References

1. IEEE Recommended Practice for Software Requirements Specifications: http://www.cse.msu.edu/~cse870/IEEEXplore-SRS-template.pdf
2. DUC: https://www.aclweb.org/anthology/W04-1003.pdf
3. Gigaword: https://dl.acm.org/doi/pdf/10.5555/2391200.2391218?download=true
4. New York Times: https://catalog.ldc.upenn.edu/docs/LDC2008T19/new_york_times_annotated_corpus.pdf
5. CNN/Daily News: http://papers.nips.cc/paper/5945-teaching-machines-to-read-and-comprehend.pdf
6. NEWSROOM:: https://arxiv.org/pdf/1804.11283.pdf
7. Text Summarization Model:

E. Overview

We are going to focus on describing the system concerning product perspective, product functions, user description, constraints & assumptions and dependencies in the following document. Additionally, user requirements for performance and ease of navigation will be enclosed.

II. General Description

A. Product Perspective

There are several existing datasets on text summarization. We briefly describe the properties of these datasets:

DUC: The Document Understanding Conference dataset contains 500 news articles and their summaries capped at 75 bytes. The summaries are written by human authors and there exist more than one summary per article which is its major advantage over other existing datasets. However, DUC datasets are small, which makes it difficult to use them as training data

Gigaword: The Gigaword Corpus contains nearly 10 million documents from seven newswire sources, including the Associated Press, New York Times Newswire Service, and Washington Post Newswire Service. Compared to other existing datasets used for summarization, the Gigaword corpus is the largest and most diverse in its sources. While Gigaword does not contain summaries, prior work uses Gigaword headlines as simulated summaries.

New York Times Corpus: The New York Times dataset is a large collection of dataset containing several hundred thousand articles written between 1987–2007 that have paired summaries. The summaries were written for the corpus by library scientists, rather than at the time of publication.

CNN/Daily Mail: The dataset includes CNN and Daily Mail articles (39 sentences on average), each associated with several bullet point descriptions.

NEWSROOM: This corpus consists of 1.3 million diverse summaries combining abstractive and extractive strategies. It is one of the most recent large-scale dataset introduced for text summarization.

Almost all these existing summarization datasets DUC, Gigaword, New York Times, CNN/Daily Mail and NEWSROOM consist of news articles. The news articles have their own specific styles and therefore the systems trained on only news may not be generalized well. On the other hand, the existing datasets may not be large enough (DUC) to train a sequence-to-sequence model, the summaries may be limited to only headlines (Gigaword), they may be more useful as an extractive summarization dataset (New York Times) and their abstraction level might be limited (CNN/Daily mail). To overcome the issues of the existing datasets, we present a new large-scale dataset. We also use an existing model for abstractive text summarization that evaluates the performance of existing and proposed dataset.

From the user point of view, he/she will be able to access the dataset in view mode only, and they will be able to add article to the model and would receive a text summary for the article.

B. Product Functions

Our Project can be classified into 4 product functions.
Creating a Dataset: We will create a dataset that will contain headline, body, date and source from different websites.
Model: We will Implement some existing model suitable for our dataset.
Web Portal: The web portal will provide the end-user a interface to upload the article. We can also check the effectiveness of our model using this portal.
Server: The server will basically process the input article and will return the prescribed headline.

C. User Characteristics

The Text Summarization Dataset can be used by researchers to overcome the most challenging NLP tasks. The user-interface for text summarization is intended for users with curiosity and appreciation of the abstractive text summarization.

D. Constraints

1. Some web pages crawled during dataset generation might not contain a summarizable data.
2. Articles provided by users to generate text summary may give inaccurate results due to limitation of how neural networks work.

E. Assumptions and dependencies

1. Development would be done on a machine that supports Linux Kernel.
2. Stable Network Connection is required.
3. Text passed to the model is in English language.
4. Cloud Hosting Platform/ Server to host the Web Portal.
5. Powerful Processor/ GPU to train the model.

III. Specific Requirements

A. External Interface Requirements

1. User Interfaces

a) The user is required to paste the text, upload a file or enter a link for the document to be summarized.

b) User will input the desired length of the summary.

c) The entered input would be parsed and converted into the required format which will be further processed for summarization.

d) The summarized output will be sent to the user via another link.

2. Hardware Interfaces

Not Applicable.

3. Software Interfaces

a) The software includes a web interface to communicate with the user.

4. Communication Interfaces

a) The input and output will be served to the user via Http protocol in html format (via a browser).

B. Functional Requirements

1. Text Summarizer Requirements

a) The system should provide text parser functions which can take the whole text and separate into sentences, paragraphs and words.

b) The system should provide text-to-feature function which can take the necessary part and obtain a feature vector.

c) The system needs a classifier which is well-trained to select summary sentences.

d) The system should provide a sentence modifier to beautify and polish output text while changing some words with their synonyms etc.

2. **Summary Setting Requirements**
    a) The system should take parameters such as summary length from the user before summarizing.

3. **Train System Requirements**
    a) The system should provide taking new data to train classifiers to improve reliability.
    b) The input file for the training model must be in a specific format as defined.

4. **Web Scraper Requirements**
    a) The Scraper requires the URL(s) with which it would start crawling.
    b) The Scraper requires the number of webpages to crawl further(depth of crawling).

C. Non-Functional Requirements

1. **Usability**
   The system should be easy to use. The summarized text should be instantly available to the user to save time.
2. **Reliability**
   This software will be developed with machine learning and deep learning techniques. Training set will be used to compare results and measure reliability.
3. **Maintainability**
   The users will be able to access the program at any time, so maintenance won't be a big issue. The administrators should have easy access to models for fast updates.
4. **Performance**
   Calculation time and response time should be as little as possible, because one of the software's features is timesaving. The whole cycle of summarizing a document should not be more than 10 sec for a single page file.
   The capacity of servers should be high to host many sessions at the same time.
5. **Security**
   The software needs to be frequently updated so that it is free of common vulnerabilities. The administrator will do a timely inspection of the database to ensure safety.

D. Logical Database Requirement:

Our proposed dataset will contain the following:

- Article Text
- Summary
- Metadata:

- Date
- Source