RSET
RAJAGIRI SCHOOL OF
ENGINEERING & TECHNOLOGY
(AUTONOMOUS)

Mini Project Report On

Simplify

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

in

Computer Science & Engineering

By

Adithyakrishnan (U2103015)

Ali Thalhathe (U2103027)

Anjana Raju (U2103036)

Anna Laju (U2103038)

Under the guidance of

Ms Asna P K

Department of Computer Science & Engineering
Rajagiri School of Engineering & Technology (Autonomous)
(Affiliated to APJ Abdul Kalam Technological University)
Rajagiri Valley, Kakkanad, Kochi, 682039
May 2024

CERTIFICATE

This is to certify that the mini project report entitled ***"Simplify"*** *is a bonafide record of the work done by* ***Adithyakrishnan A(U2103015)***, ***Ali Thalhathe (U2103027)***, ***Anjana Raju (U2103036)***, ***Anna Laju (U2103038)***, *submitted to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B. Tech.) in Computer Science and Engineering during the academic year 2023-2024.*

Ms Asna P K
Asst Professor
Dept. of CSE
RSET

Mr Harikrishnan M
Asst Professor
Dept. of CSE
RSET

Dr Preetha K.G
HOD
Dept. of CSE
RSET

ACKNOWLEDGEMENTS

I wish to express my sincere gratitude towards Dr P. S. Sreejith, Principal of RSET, and Dr. Preetha K.G., Head of the Department of Computer Science and Engineering for providing me with the opportunity to undertake my mini project, "Simplify".

I am highly indebted to my project coordinators, **Mr Harikrishnan M**,Asst Professor, Department of Computer Science and Engineering for their valuable support.

It is indeed my pleasure and a moment of satisfaction for me to express my sincere gratitude to my project guide **Ms Asna P K** for her patience and all the priceless advice and wisdom she has shared with me.

Last but not the least, I would like to express my sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

Adithyakrishnan A

Ali Thalhathe

Anjana Raju

Anna Laju

Abstract

Text simplification is crucial for enhancing the accessibility of information to diverse audiences, including individuals with cognitive disabilities, language learners, and those with limited literacy skills. Our project Simplify explores the application of Natural Language Processing (NLP) techniques, specifically leveraging the T5 (Text-to-Text model Transformers) model, for text simplification tasks. We present an approach where the T5 model is trained on a specialized dataset curated for text simplification, termed as "ASSET" (A Simplified and Standardized English Text corpus). ASSET comprises pairs of complex sentences and their simplified counterparts, providing rich training data for the T5 model to learn the mappings between complex linguistic structures and its simplified equivalents. Through fine-tuning T5 on the ASSET dataset, we plan to demonstrate the effectiveness of the model in generating simplified versions of text while preserving the original semantics.T5 works with a general-purpose text-to-text format. This means it can be applied to various NLP tasks, including text simplification, by simply treating the simplification process as converting complex text to a simpler version thereby enabling it to capture intricate language patterns, resulting in more accurate and contextually appropriate simplifications. Our findings highlight the potential of leveraging state- of-the-art NLP models like T5 for automating the text simplification process, thereby improving the accessibility and readability of textual content across various domains.

Contents

List of Figures

Chapter 1

Introduction

1.1 Background

Accessing information is important, but it can pose challenges for individuals who find it difficult to read or understand complex language. It may include people with learning disabilities, people who do not have English as their first language, and those with limited reading skills. Text simplification is developed with an intention to help those struggling with this. It takes difficult language and makes it easier to understand without changing the meaning. Our project Simplify, a website designed to simplify text, uses Natural Language Processing (NLP).The main model involved is a Transformer model which consists of T5 as the encoder and BART as the decoder.T5 is pre-trained on a massive dataset of text and code. This pre-training exposes the model to a wide range of sentence structures, vocabulary, and concepts. The model's understanding of diverse language helps it learn effective simplification strategies.BART's pre-training involves understanding both left and right context, making it effective for understanding complex sentences.This project seeks to make sure that all kinds of information is easy to comprehend and understand by everybody. We use a dataset called "ASSET" It has pairs of complex sentences and its simpler versions. We train the model on the dataset to learn how to change complicated sentences into simpler ones while retaining the meaning. It is implemented by intially tokenizing the complex sentences given as input into words which is obtained by passing it through the T5Tokenizer which which convert it into token embeddings which is then passed to the encoder T5, after that it gets compared with the target simpler sentences that is passed along the decoder. This is the basic background behind the implementation of Simplify. It will later be opitimized and evaluated using the Bert score,Google blue,Sari score,Blue score,Rouge Metrics and Sacredblue metrics.

1.2 Problem Definition

Many individuals face challenges in reading or understanding complex language, it may include people with learning disabilities, people who have English as their second language, or those with limited reading skills, this limits their ability to comprehend information. Our project aims to simplify such text into simpler, more understandable text while retaining the original meaning thereby enhancing accessibility for diverse audiences.

1.3 Scope and Motivation

Our project focuses on utilizing Natural Language Processing (NLP) techniques, particularly the T5 model, for text simplification tasks. Specifically, we aim to train the model on a specialized dataset, "'ASSET 'consisting of pairs of complex sentences and their simplified counterparts. The scope of our project encompasses fine-tuning T5 on this dataset to generate simplified versions of text while preserving the original semantics. Additionally, we will explore the effectiveness of T5 in capturing the language patterns to produce accurate and contextually appropriate simplifications.It promotes inclusivity by making information accessible to a wider range of people where all can benefit from the clearer and more concise language. It enhances readability and comprehension as it allows everyone to grasp information more easily and save time struggling with challenging text. There's a need for information to be accessible to a wider audience especially those who struggle to grasp complex texts. This software tackles this challenge by making information more approachable for everyone.

. This can save time and resources compared to doing it by hand. Our project aims to use the latest technology to break down language barriers and empower people to engage with text more confidently

1.4 Objectives

Simplify aims to bridge the gap between complex text and diverse audiences.It helps users to grasp information more efficiently, regardless of their literacy level or native language. This promotes equal access to information regardless of language proficiency.This website

not only saves time for busy professionals and students, but also promotes equal access to knowledge and education.

1.5 Challenges

T5 needs a lot of memory to learn so we require a large dataset.Also, it is necessary to ensure that the model understands the text accurately without distorting the original text meanings.

1.6 Assumptions

The main assumptions that are made for developing Simplify includes the input text is always written in English and it should be input as a single sentence.

1.7 Societal / Industrial Relevance

Simplify is a website designed to simplify complex text with a primary focus on enhancing accessibility for various user groups.People can use it every day to understand complex texts in news articles, study materials, or even emails better.This software not only makes reading easier but also saves time. It could be a valuable tool for busy professionals, students with tight schedules.

1.8 Organization of the Report

The report has been organised in a way to provide a comprehensive overview of the proposed 'Simplify' project in 5 different chapters. The project's purpose and aim are laid out in detail in the introduction, which includes background information, a definition of the problem, scope and motivation, challenges, objectives, assumptions and societal industrial relevance. The second chapter explores the software requirement specification which includes the body of current literature, highlighting relevant research publications and their major conclusions. The comprehensive requirements chapter that follows describes the functional requirements as well as the hardware and software requirements, guaranteeing implementation clarity and viability. The chapter on system architecture dwells further into technical issues including an architectural design, an overview and in-

depth module breakdowns to help understand the inner working of Simplify. In the next chapter, an overview on the results are discussed, details about the testing, quantitative results and graphical analysis. Through the conclusion we provide summary of the main elements and an exciting section on future scope and growth opportunities.

Chapter 2

Software Requirements Specification

2.1 Introduction

2.1.1 Purpose

The purpose of this document is to outline the objectives and scope of the Text Simplification Tool, a website referred to as "Simplify". The software aims to address the need for making information more accessible to diverse groups, including individuals with cognitive disabilities, language learners, and those with limited literacy skills. This document is intended for both software's developers and stakeholders.

2.1.2 Product Scope

Simplify is a website designed to simplify complex text using Natural Language Processing (NLP) techniques, with a primary focus on enhancing accessibility for various user groups. People can use it every day to understand complex texts in news articles, study materials, or even emails better. This software not only makes reading easier but also saves time. It could be a valuable tool for busy professionals, students with tight schedules.

2.2 Overall Description

2.2.1 Product Perspective

Simplify is a website that makes difficult text easier to understand. It's great for students, professionals, or anyone who wants to grasp complex information without feeling overwhelmed. With Simplify, you just paste in the text you want to simplify, and it does the hard work for you. It keeps the important parts but makes everything clearer and

simpler to read. It's like having a personal English language helper right at your fingertips.

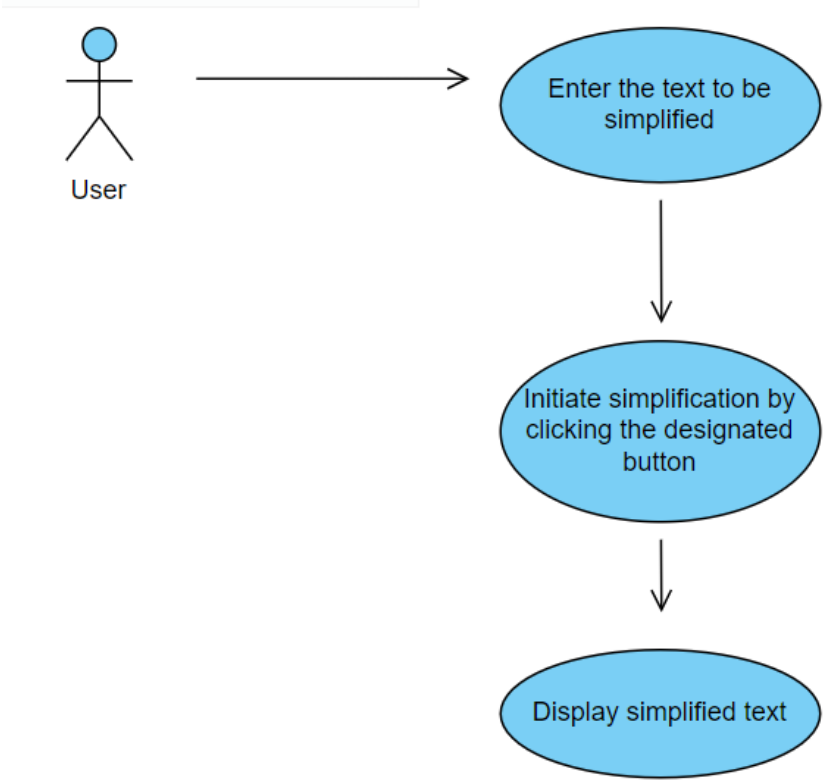


Figure 2.1: Use Case Diagram of Simplify

2.2.2 Product Functions

Text Simplification for Accessibility (Priority: High)

- Automatically simplifies complex text to improve readability for people with low literacy levels, or those for whom English is not a first language.
- This could have a significant impact on access to information and education as comprehensible and easily understandable information will be to all.

Writing Assistant Tools (Priority: Medium)

- The model's understanding of simplification could be used to develop writing assistant tools that suggest simpler alternatives to complex phrases.

- This could be helpful for writers who want to improve the clarity and conciseness of their writing.

2.2.3 Operating Environment

Since Simplify is a website, there are no major requirements with regard to the operating environment. A computer running windows 8 or above with 8 gigabytes of RAM should be enough.

2.2.4 Assumptions and Dependencies

Some of the assumptions that are made for developing Simplify include:- These include:-

- The input text is written in English.
- The input text should be given as a single sentence.
- The field of Natural Language Processing (NLP) is constantly evolving. Hence emergence of new, more efficient T5 models might require adjustments to the training process or model deployment strategy.

2.3 External Interface Requirements

2.3.1 User Interfaces

The software's user interfaces will be designed using HTML and CSS.The software must have a web-based user interface such as Google Chrome. The interfaces to be implemented include the main screen for interactions, an input interface for entering data and an output interface for displaying results and a button to initiate the simplification process. Users will enter their text to be simplified according to the word limit specification, one sentence at a time. T5 will be employed to simplify any text displayed to users. Flask, a web framework within python programming language will manage the layout and functionality of these interfaces, ensuring they align with website standards and deliver a smooth and efficient user experience.Flask manages the communication between the frontend interface and the backend processing components.

2.3.2 Hardware Interfaces

The hardware requirements include a computer or a server capable of running web server software. This computer or server should have sufficient processing power preferably a dual Core Processor, a memory of minimum 8 GB RAM, and storage to handle the website's operations efficiently. The computational demands of models and datasets for text simplification require substantial computing power, which a good GPU can provide which can be obtained from tools like Google Colab. Additionally, reliable internet connectivity is necessary for hosting the website and serving it to users accessing it over the internet. The hardware infrastructure supporting the text simplification website is robust, reliable, and capable of meeting the demands of both users and the application itself.

2.3.3 Software Interfaces

The website is constructed using Python and Flask, enabling interactions over the web. We use various python libraries such as Pandas, which makes handling input sentences and processing output simplifications much easier, Scikit-learn, which helps us with machine learning tasks such as preparing our data before we feed it into our models, or cleaning up the results afterwards, PyTorch, which is a tool for building deep learning models .The encoder-decoder model uses T5 to understand our input sentences and BART to generate simplified versions. To work with these models, we use Transformers from Hugging Face. It gives us access to pre-trained models like T5 and BART, so we don't have to start from scratch. Flask, a python framework is used to create user-friendly web interfaces. This lets people interact with our project easily through their web browser. Overall, these tools work together to make sure our project runs smoothly, making it easier for us to work with data and build models.

2.3.4 Communication Interfaces

The web browser component of the website requires communication protocols such as HTTP (Hypertext Transfer Protocol) for transmitting requests and receiving responses from web servers hosting the text simplification service. Additionally, it may utilize network server communications protocols for exchanging data with remote servers hosting

additional resources or services related to text simplification. Moreover, communication with the Transformer and BART model occurs internally within the website. Since communication security and encryption are crucial aspects, we implement protocols like HTTPS (Hypertext Transfer Protocol Secure) to ensure the integrity of data transmitted over the network.

2.4 System Features

This section describes the features provided by Simplify: Text simplification using T5 and explains how they can be used, and how it will be useful to the user.

2.4.1 Simplification

2.4.1.1 Description and Priority

- This website aims to enhance accessibility by simplifying text using Tranformer model which consists of T5 as the encoder and BART as the decoder.
- It converts complex text into simpler language, benefiting users with cognitive disabilities or those who prefer simple language.
- Priority: High

2.4.1.2 Stimulus/Response Sequences

- Users input complex text into the system one sentence at a time in english .
- System processes the input using the Transformer model.
- System generates simplified text as output on the website.
- Users receive simplified text for enhanced accessibility.

2.4.1.3 Functional Requirements

- 2.4.1.3.1 The input text should be in english. Other language text will not be accepted by the system.
  REQ-1:English Language

- 2.4.1.3.2 If T5 fails to simplify the text, the system should provide the original text as output.

2.4.2 User Interface

2.4.2.1 Description and Priority

- Enhances the user interface for better usability
- Priority: High

2.4.2.2 Stimulus/Response Sequences

- User action: Inputting data
- System response: Instant validation and Response Sequences

2.4.2.3 Functional Requirements

- REQ-1: Provide real-time validation for user inputs.

2.5 Other Nonfunctional Requirements

2.5.1 Performance Requirements

Our text simplification project is required to deliver prompt results while maintaining accuracy in the simplified output text. The goal is to ensure that users receive timely responses without any significant slowdowns in performance.

2.5.2 Safety Requirements

We implement measures to protect user data privacy, including data encryption both in transit and at rest.

2.5.3 Security Requirements

Using strong encryption methods to protect data during transmission and storage is essential. We test thoroughly to make sure everything runs smoothly and doesn't make any errors. We have to ensure that we do not distort the original text meanings.

2.5.4 Software Quality Attributes

Our project aims to provide a great experience for users and make it easy for developers to maintain. For users, we're aiming for a simple interface that responds quickly, ideally within 5 seconds. For developers, our goal is to write clear code that's easy to understand, with a design that's split into parts and well-documented. Accuracy in simplifying text is very important, and we're targeting a score of over 80 on BLEU score and SARI metrics.

Chapter 3

System Architecture and Design

3.1 System Overview

Simply aims to reduce the linguistic complexity of a text, while still maintaining its original information content and meaning. Given an English sentence as input, the model aims to rearrange the words or substitute words or phrases to make it easier to comprehend. There are many reasons why such a task is needed. Our project helps those with low literacy levels, second language readers, those suffering from various types of reading comprehension problems, as well as children.

Input Interface:

The Simplify website, built using Flask, provides a user-friendly interface for users to input English sentences one at a time into an input text box. The simplification process is initiated by pressing a button. The interface ensures a smooth user experience.

Pre-processing:

English sentences in the ASSET dataset undergo pre-processing to identify complex and challenging vocabulary. We employ a process called tokenization to break down sentences into words ,this prepares the sentences for further simplification. **Simplification Algorithm:**

A sophisticated algorithm restructures sentences by rearranging words, substituting complex phrases with simpler alternatives, and replacing difficult vocabulary with more common terms. The algorithm aims to retain the original meaning while improving readability.

Backend Processing:

Python serves as the primary programming language for the project. Backend processing involves several Python libraries and also an end-to-end pipeline leveraging state-of-the-art (SOTA) transformer models:

- Pandas: Utilized for data manipulation tasks, such as handling input sentences and processing output simplifications.
- Scikit-learn (SKlearn): Used for machine learning tasks, potentially for pre-processing or post-processing steps in the simplification pipeline.
- PyTorch: Employed for deep learning models, specifically for the encoder-decoder architecture utilizing T5 as the encoder.
- Transformers from Hugging Face: This library is utilized for NLP tasks, providing access to pre-trained models like T5 and BART.
- T5Tokenizer: Normal sentences are tokenized using the T5 tokenizer, converting them into token embeddings.
- Encoder-Decoder Model: Token embeddings are passed through an encoder-decoder model. T5 acts as the encoder, capturing contextual information from the input sentences. The decoder component, utilizing BART, generates token embeddings for simplified sentences.
- BARTTokenizer: Token embeddings for simplified sentences are converted back into human-readable sentences using the BART tokenizer.

Model Training:

The project utilizes supervised learning techniques to train the transformer models. The models are trained on normal and simplified sentences provided by the WASSET dataset to learn the mapping between them effectively. It undergoes fine-tuning.

Evaluation:

The Bert score, Google blue, Sari score, Blue score, Rouge Metrics and Sacred blue metric evaluates the quality and accuracy of simplified text generated. The evaluation process ensures the effectiveness of the simplification model in retaining the meaning while enhancing readability.

Deployment:

The Flask framework facilitates the deployment of Simplify, providing a reliable platform for users to access the simplification service. Flask manages the communication between the frontend interface and the backend processing components.

Output Generation:

Simplified sentences are presented to users in a clear and concise format in the output text box, making them easier to comprehend while preserving the underlying information.

3.2 Architectural Design

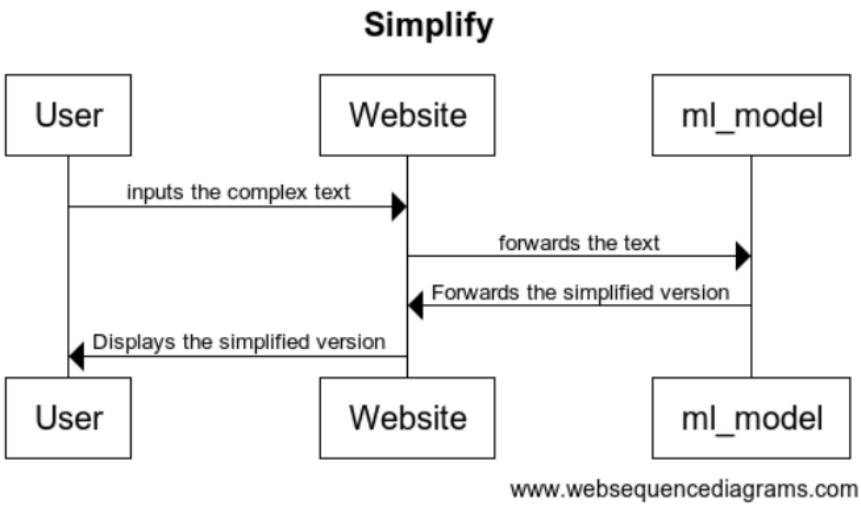


Figure 3.1: Sequence diagram of Simplify

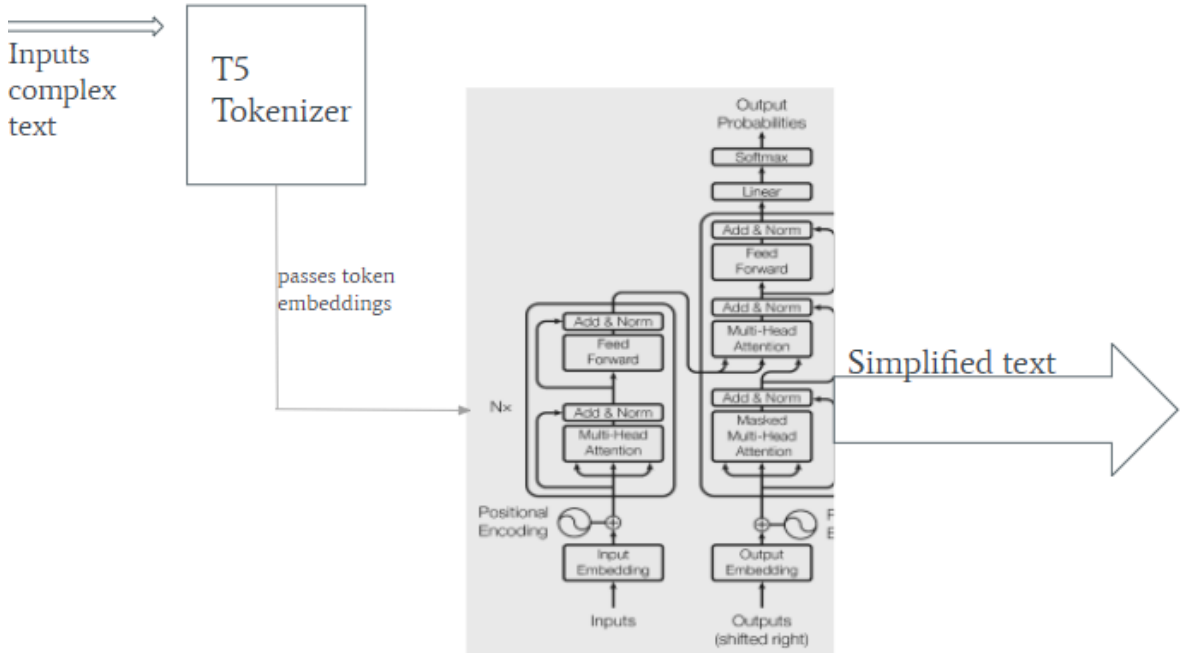


Figure 3.2: Architecture of Simplify

3.3 Dataset identified

The ASSET Dataset is used for the project.It is a curated collection of texts specifically chosen for their complexity, along with simplified versions of those texts. It was used in a text simplification project to train models on how to transform complex sentences into simpler ones. By providing examples of both complex and simplified texts, the dataset helps the models learn how to simplify language effectively, making information more accessible to a wider audience.

3.4 Proposed Methodology/Algorithms

Following complex word identification(CWI), substitute generation, filtering and substitute ranking steps, the overall simplification algorithm is given below. Given the sentence S and complexity threshold t, we first identify named entity using entity identification system . We add entities into ignore list which means these words do not need to be simplified.In algorithm 2, in the model, we identify all complex words in sentence s using CWI step excluding ignore list (line 1). If the number of complex words in the sentence s is larger than 0 (line 2), the model will try to simplify the top complex word w (line 3). LST5 calls substitute generation (line 4) and substitute ranking (line 5) in turn.The

model chooses the top substitute (line 6). When the model performs the simplification, it will replace w into top (line 8) and add the word top into ignore list (line 9). After completing the simplification of one word, we will iteratively call the model (line 10 and line 12). If the number of complex words in S equals to 0, we will stop calling the model (line 15).

Algorithm 1 :Lexical simplification framework

S ← Input Sentence
t ← Complexity threshold
ignore list ← Named Entity Identification(S)
LST5(S, t, ignore list)

Algorithm 2: Lexical Simplification Algorithm

complex words ← CWI(S, t) - ignore list
if number(complex words) > 0 then
w ← head(complex words)
subs ← Substitution Generation(S, w)
subs ← Substitute Ranking(subs)
top ← head(subs)
if fre(top) > fre(w) then
Replace(S, w, top)
ignore list.add(w)
LST5(S, t, ignore list)
else
LST5(S, t, ignore list)
end if
else
return S
end if

3.5 User Interface Design

The user interface (UI) design for text simplification is designed as a single-page layout website page, it is done so to ensure simplicity and efficiency in the user experience. This page acts as both the place where users enter their text and where they see the simplified version. It makes the whole process of simplifying text smooth and easy by combining everything into one place. The page also consists of a button which upon pressing initiates the simplification process. At the top section of the page, users encounter a text input field where they can type or paste the text they want to simplify. This input field is accompanied by clear instructions, guiding users on how to input text and initiate the simplification process. The design adopts a minimalist approach, prioritizing a clutter-free interface . Once users have inputted their text, they can trigger the simplification process by clicking the "Simplify" button located below the input field. Upon clicking the button, the simplified version of the input text is promptly displayed below. The output text is formatted for readability, ensuring that users can easily understand the simplified content.

The front-end development of the UI is created using HTML and CSS. On the back-end, Python along with its various libraries are use to handle text processing and simplification tasks. Flask, a python framework manages communication between the front end and back end of the website, though it primarily focuses on the back end.

3.6 Description of Implementation Strategies

In this project , Python serves as the primary programming language. The project utilizes specific Python libraries, including Pandas for data manipulation, SKlearn for machine learning tasks, PyTorch for deep learning models, and Transformers from Huggingface for NLP tasks.The core methodology involves the transformation of complex text into token embeddings using the T5 tokenizer. BART then acts as the encoder within an encoder-decoder architecture, with BART serving as the decoder. Then simplified version of text is generated by BART tokenizer.The model is evaluated using a combination of Bert score, Google blue, Sari score, Blue score, Rouge Metrics and Sacred blue metrics. SARI evaluates the quality of the simplified text compared to the original, while BLEU measures the similarity between the generated text and reference text. This combined

approach provides a comprehensive evaluation of both precision and readability at the sentence level.

3.7 Module Division

3.7.1 Front-end Modules:

1. **Text Input Interface:** This module is responsible for managing the user interface for inputting text to be simplified.

3.7.2 Back-end Modules:

1. **Text Preprocessing Module:** This module handles the preprocessing of input text before simplification such as tokenizing the text.

2. **Simplification Model Integration Module:** This module integrates the text simplification model into the system for processing input text.

3. **Text Simplification Module:** The core module responsible for applying the simplification techniques to the input text(The Transformer model which consists of T5 and BART models).

4. **Evaluation Module:** This module uses the Bert score, Google blue, Sari score, Blue score, Rouge Metrics and Sacredblue metrics evaluate the quality of text generated.

5. **Output Interface Module:** The Output Interface Module is responsible for handling the display of the simplified text output generated by the text simplification system. It ensures that the output is presented to the user in a understandable and clutter-free manner.

3.8 Gantt Chart

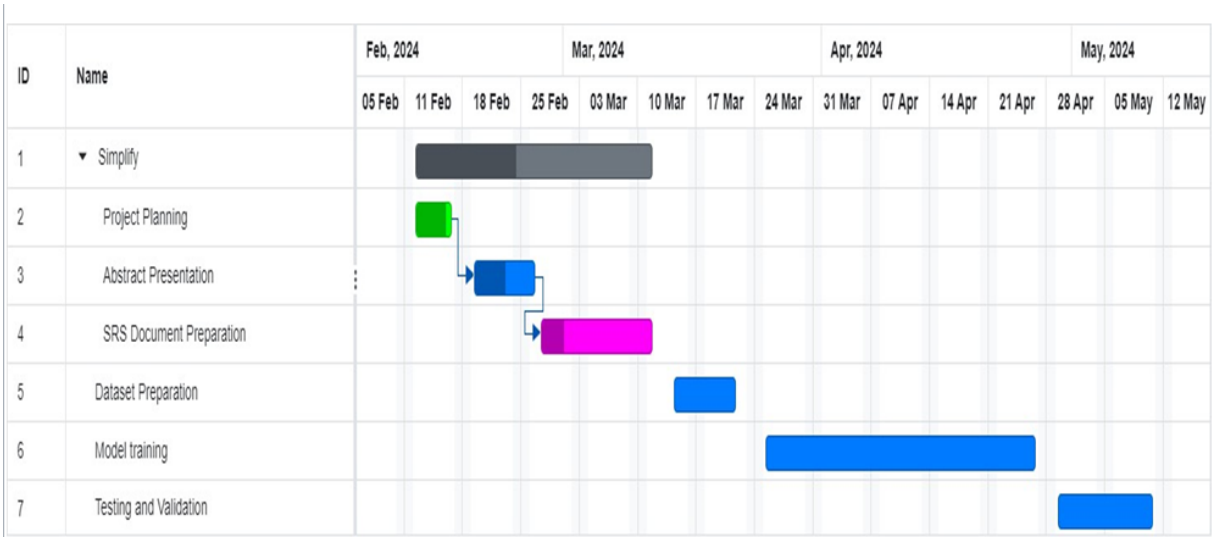


Figure 3.3: Work schedule of Simplify depicted using Gantt Chart

Chapter 4

Results and Discussions

4.1 Overview

This project addressed the challenge of information accessibility by developing "Simplify," a website that leverages Natural Language Processing (NLP) to simplify complex text. Our target audience includes individuals with learning disabilities, non-native English speakers, and those with limited literacy skills.

Simplify was developed using the Transformer model, combining T5 (encoder) and BART (decoder) functionalities. T5, pre-trained on the dataset 'ASSET', excels at understanding diverse language structures and vocabulary. BART is great at understanding complicated sentences and conversations from both sides and linking target sentences to the input sentences, effectively handles complex sentences.

4.2 Testing

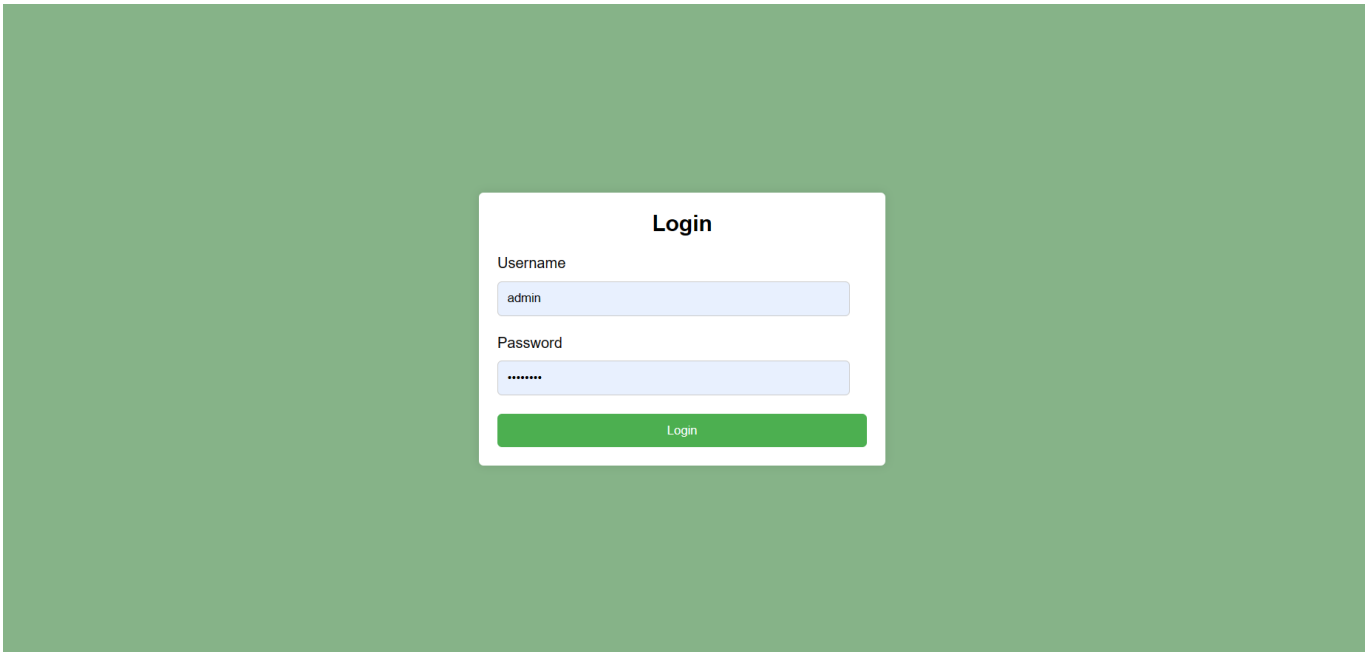


Figure 4.1: Login page

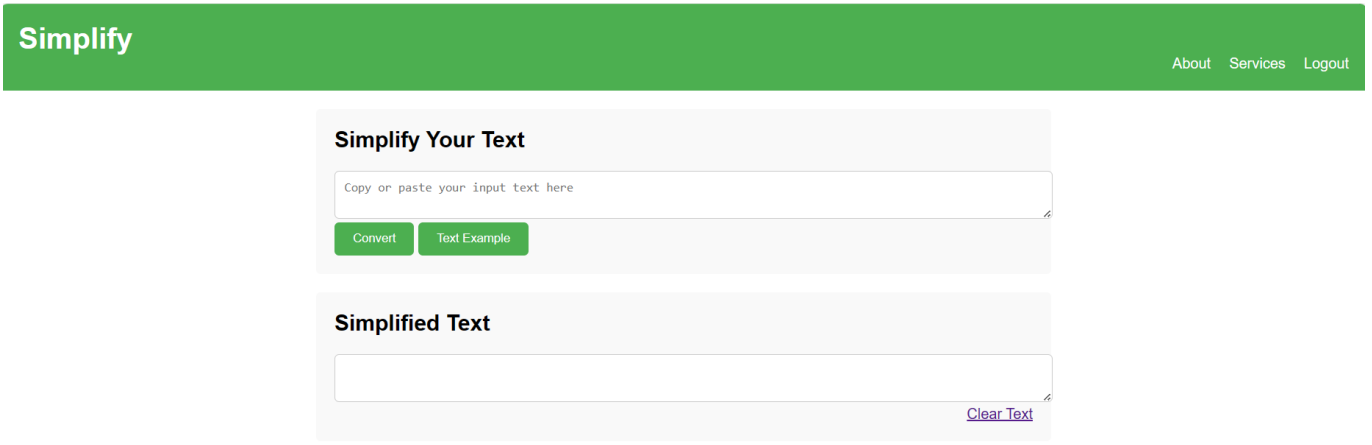


Figure 4.2: Main page

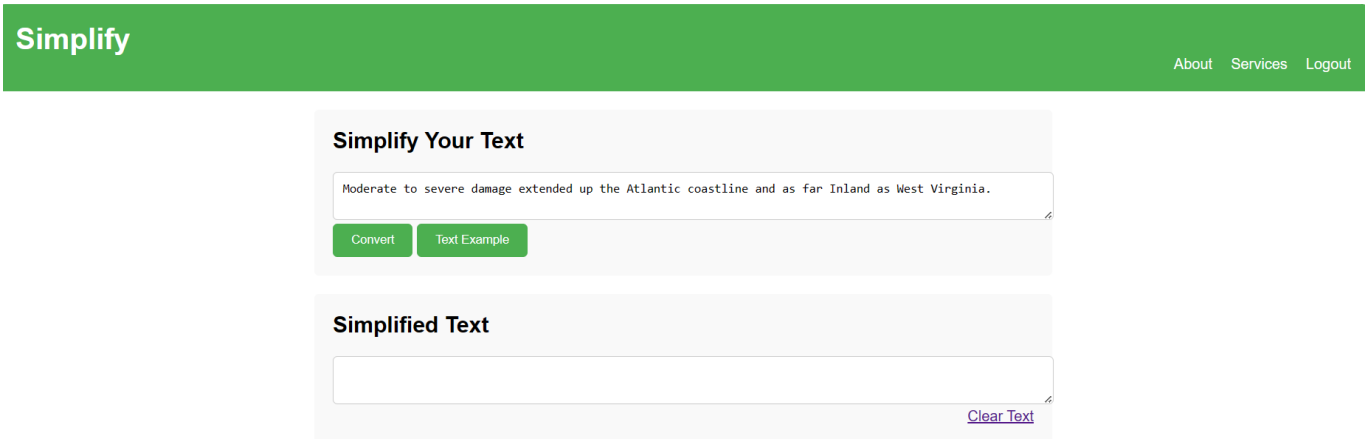


Figure 4.3: Input Text

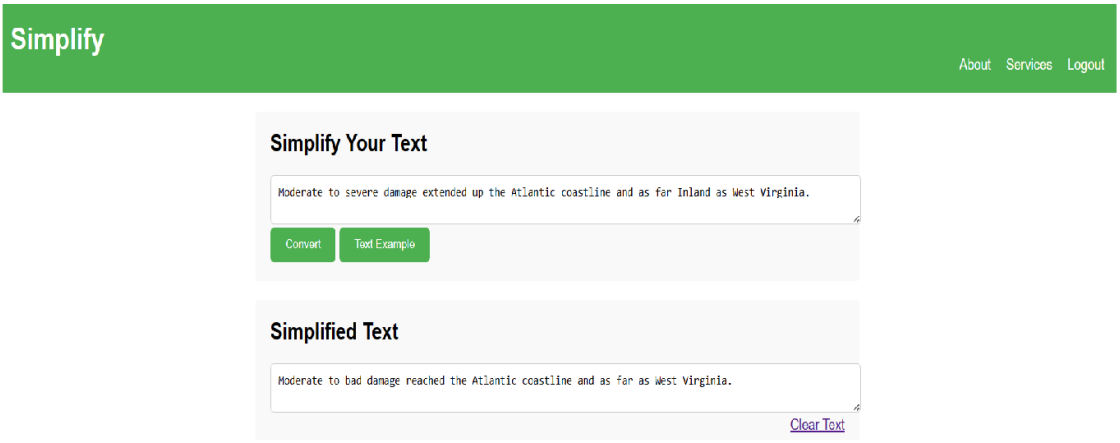
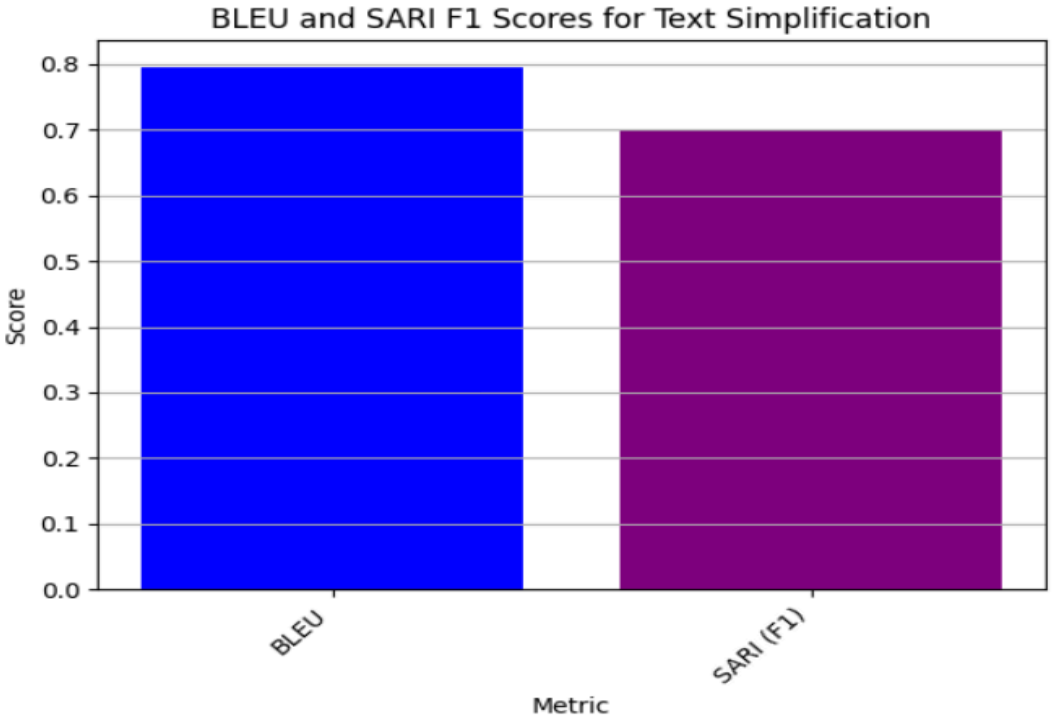
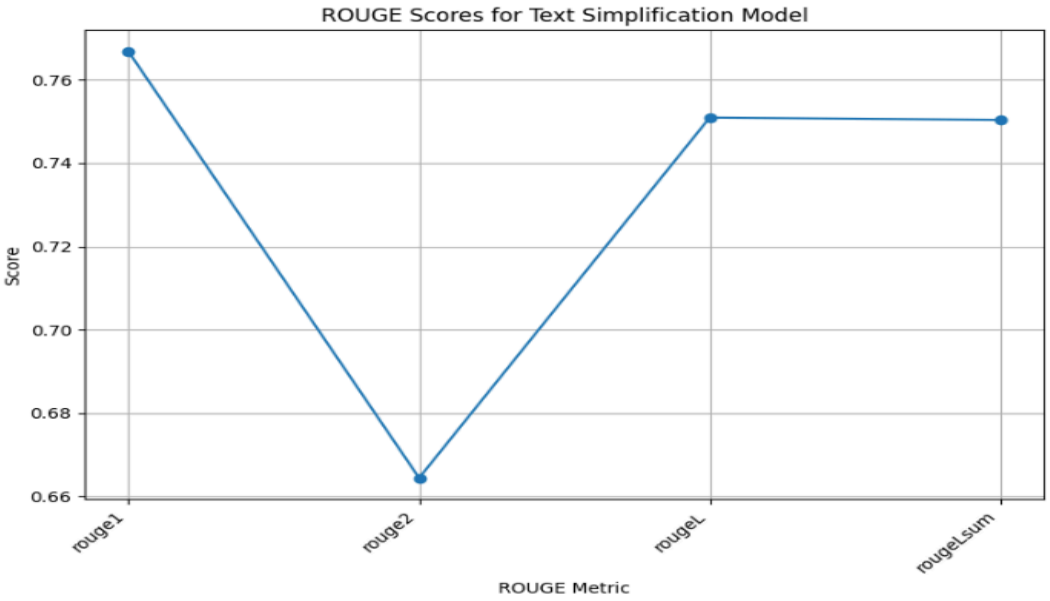
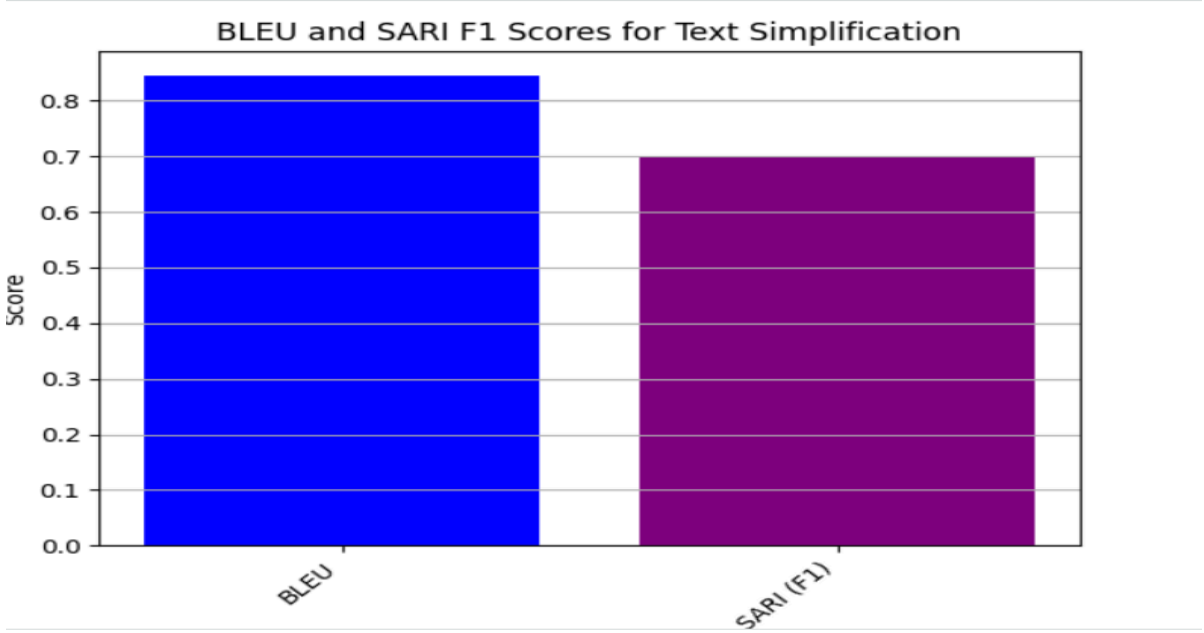


Figure 4.4: Output: The simplified text

4.3 Quantitative Results








```
Evaluating checkpoint-3375: 100%|██████| 359/359 [01:02<00:00,  5.75it/s]
tokenizer_config.json: 100%|██████| 25.0/25.0 [00:00<00:00, 1.06kB/s]
config.json: 100%|██████| 482/482 [00:00<00:00, 26.0kB/s]
vocab.json: 100%|██████| 899k/899k [00:00<00:00, 3.33MB/s]
merges.txt: 100%|██████| 456k/456k [00:00<00:00, 3.53MB/s]
tokenizer.json: 100%|██████| 1.36M/1.36M [00:00<00:00, 4.11MB/s]
model.safetensors: 100%|██████| 1.42G/1.42G [00:13<00:00, 40.2MB/s]
Some weights of RobertaModel were not initialized from the model checkpoint at roberta-large and are newly initialized: ['roberta.pooler.dense.bias', 'roberta.pooler.dense.weight']
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.
Results for checkpoint-3375:
result: {'rouge': {'rouge1': 0.7492083367435423, 'rouge2': 0.6335656982742597, 'rougeL': 0.7311808471277131, 'rougeLsum': 0.7313045589083585}, 'bleu': {'bleu': 0.7999286368094305, 'precisions': [0.9337410805300713, 0.8435987681478223,
```

'google_bleu': {'google_bleu': 0.48157999206063414}, 'bertscore': {'precision': 0.9573945260831356, 'recall': 0.9428519115142503, 'f1': 0.9463866514848154}, 'sacrebleu': {'score': 76.67162700643063, 'counts': [4580, 3835, 3186, 2615],

BLEU	ROUGUE-L	BERTScore	Google Bleu	SARI
0.845	0.751	0.961	0.505	41.432

Table 4.1: Evaluation accuracy with T5

BLEU	ROUGUE-L	BERTScore	Google Bleu	SARI
0.799	0.731	0.946	0.482	40.903

Table 4.2: Evaluation results with BART Base

4.3 Discussion

Simplify has been implemented in the form of a web app. Users are able to input their text and get their simplified text quickly. It only takes a few seconds.
Different models were trained, the BART base model with the ASSET dataset, the T5 BASE with ASSET dataset and the T5 small with the ASSET dataset. Sincd the T5 small with the ASSET dataset produces better accuracy and evaluation metrics we proceeded further with The Evaluation module works with great accuracy.

Chapter 5

Conclusion

5.1 Conclusion

In this project, we utilised Natural Language Processing (NLP) for simplifying text to improve information accessibility for a wider audience. We employed the T5 text-to-text transformer model, fine-tuned on a specialized dataset of complex sentences and their simplified counterparts called ASSET. This dataset provided the model with the necessary training data to learn the relationship between complex and simplified language structures.

The project's findings demonstrate the effectiveness of the T5 model in generating accurate and contextually appropriate simplifications. The model's ability to handle text-to-text formatting allows it to capture intricate language patterns, and generate resulting simplified text. It make information more accessible and readable for individuals with cognitive disabilities, language learners, and those with limited literacy skills. Overall, this project highlights the potential of NLP techniques in promoting inclusivity and improving the reach of information across diverse audiences.

5.2 Future Scope

- Expanding your project to simplify text in multiple languages broadens its usefulness and reach. Creating models that can simplify content across different languages makes it accessible to a wider audience.
- Simplify can be further developed to initiate simplification on a larger input, paragraphs at a time.
- Integrate Simplify with real-world applications, educational and assistive platforms

Bibliography

[1] Jipeng Qiang, Yun Li, Yi Zhu, Yunhao Yuan, and Xindong Wu. “LSBert: A Simple Framework for Lexical Simplification”. Submitted on 25 Jun 2020

[2] Sarah Alissa1 and Mike Wald “Text Simplification Using Transformer and BERT ” Submitted on January 2023

[3] Kang Liu, Jipeng Qiang ”Sentence Simplification Using Paraphrase Corpus for Initialization ”. Submitted on 31 May 2023

[4] Suha S. Al-Thanyyan, Aqil M. Azmi “Automated Text Simplification: A Survey”. Submitted on 05 March 2021

Appendix A: Presentation

SIMPLIFY

Ms. Asna P K

Adityakrishnan A
Ali Thalhathe
Anjana Raju
Anna Laju

Contents

INTRODUCTION

- Simplify mainly focuses on bridging the gap between complex information and its audience
- It promotes inclusivity by making information accessible to a wider range of people where all can benefit from the clearer and more concise language.
- It enhances readability and comprehension as it allows everyone to grasp information more easily and save time struggling with challenging text.
- There's a need for information to be accessible to a wider audience especially those who struggle to grasp complex texts. This software tackles this challenge by making information more approachable for everyone.

PROBLEM DEFINITION

Many individuals face challenges in reading or understanding complex language, it may include people with learning disabilities, people who have English as their second language, or those with limited reading skills, this limits their ability to comprehend information. Our project aims to simplify such text into simpler, more understandable text while retaining the original meaning thereby enhancing accessibility for diverse audiences.

Objectives

- **Collection of Data and its Preprocessing:**
  - ASSET dataset is used in our project. It contains 1,200 human-annotated simplifications with multiple rewriting transformations applied to the original sentence. The dataset is then preprocessed by cleaning and conversion into tokens and embeddings.
  - Wikilarge Dataset was also used which consists of sentence pairs extracted from Wikipedia articles.

- **Selection of a Model and its Fine-tuning:**
  - NLP models, T5 and BART is used to simplify the text.
  - Transformers Model is used in this project.ie. Encoder-Decoder architecture in which the pretrained model acts as the encoder and decoder.

Objectives

- **Evaluation for accuracy:**
  - Model evaluation is conducted using the Bert score, Rouge, Sari, Google_bleu and Sacrebleu which is used for assessing the performance of the simplified text.
  - Evaluate the fine-tuned model on held-out validation and test datasets.
- **Deployment and Accessibility:**
  - The trained T5 model is deployed on a website where we can enter an input and upon pressing the button we get the simplified text as output.
  - It is accessible to all diverse users
  - The language employed is English

Scope and Relevance

- The scope of our project encompasses fine-tuning T5 on this dataset to generate simplified versions of text while preserving the original semantics.
- People can use it every day to understand complex texts in news articles, study materials, or even emails better.
- It promotes inclusivity and enhances readability and comprehension as it allows everyone to grasp information more easily and save time struggling with challenging text
- It could be a valuable tool for busy professionals and students with tight schedules.

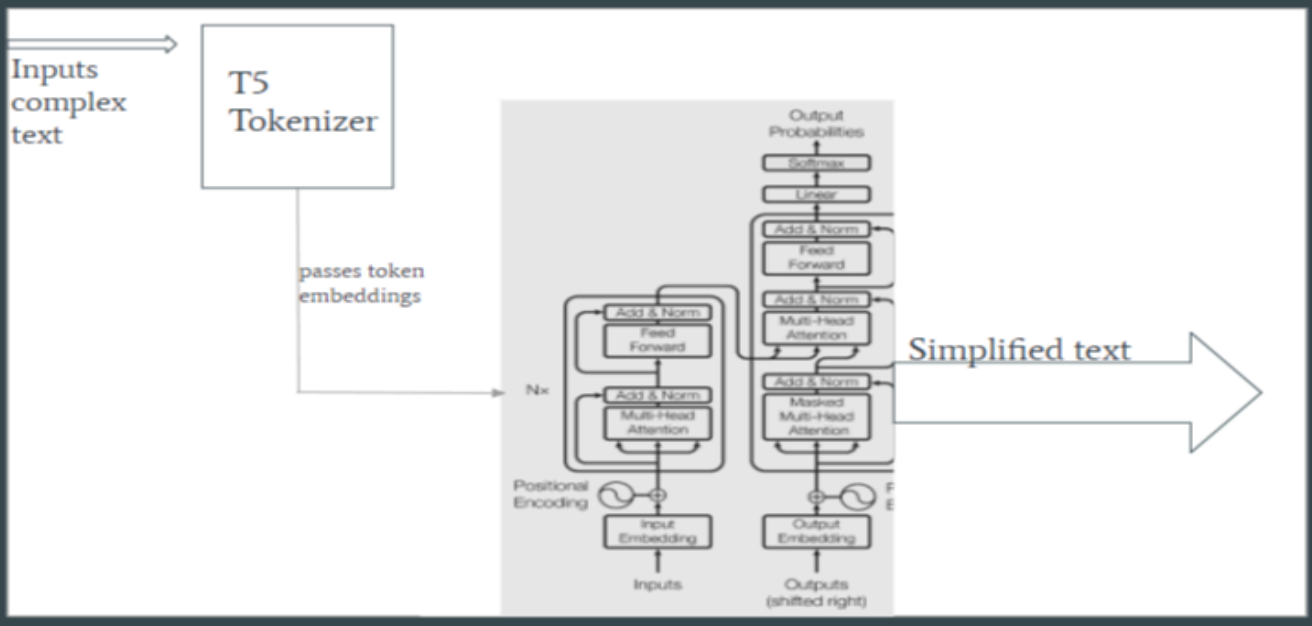
SYSTEM DESIGN-OVERVIEW

- **Input Interface**: The Simplify website provides a user-friendly interface for users to input English sentences one at a time and receive simplified versions in return.
- **Pre-processing:** Tokenization is employed to break down sentences into words and then into embeddings in order to train the model.
- **Algorithm**: T5 uses wordpiece algorithm to generate the vocabulary. Wordpiece algorithm generates subwords based on the likelihood of characters occurring together.
- **Backend Processing**: Python libraries like NLTK,Pandas, Scikit-learn, PyTorch and Transformers is used.

SYSTEM DESIGN-OVERVIEW

- **Model Training**: The models are trained on normal and simplified sentences provided by the ASSET dataset to learn the mapping between them effectively.
- **Evaluation**: Bert score, Rouge, Sari, Google_bleu and Sacrebleu evaluate the quality of text generated.
- **Deployment**: Website is created using HTML and CSS .Flask manages the communication between the frontend interface and the backend processing components.
- **Output Generation**: Simplified sentences are presented in a clear and concise format for easier comprehension while retaining the original meaning.

SYSTEM DESIGN-ARCHITECTURAL DESIGN



SYSTEM DESIGN-MODULES

- **Text Input Interface:** This module is responsible for managing the user interface for inputting text to be simplified.
- **Text Preprocessing Module:** This module handles the preprocessing of input text before simplification.
- **Text Simplification Module:** The core module responsible for applying the simplification techniques to the input text.

SYSTEM DESIGN-MODULES

- **Simplification Model Integration Module:** This module integrates the text simplification model into the system for processing input text.
- **Evaluation Module:** This module evaluates and measures the accuracy along with the quality of text generated.
- **Output Interface Module:** The Output Interface Module is responsible for handling the display of the simplified text output generated by the text simplification system. It ensures that the output is presented to the user in an understandable manner.

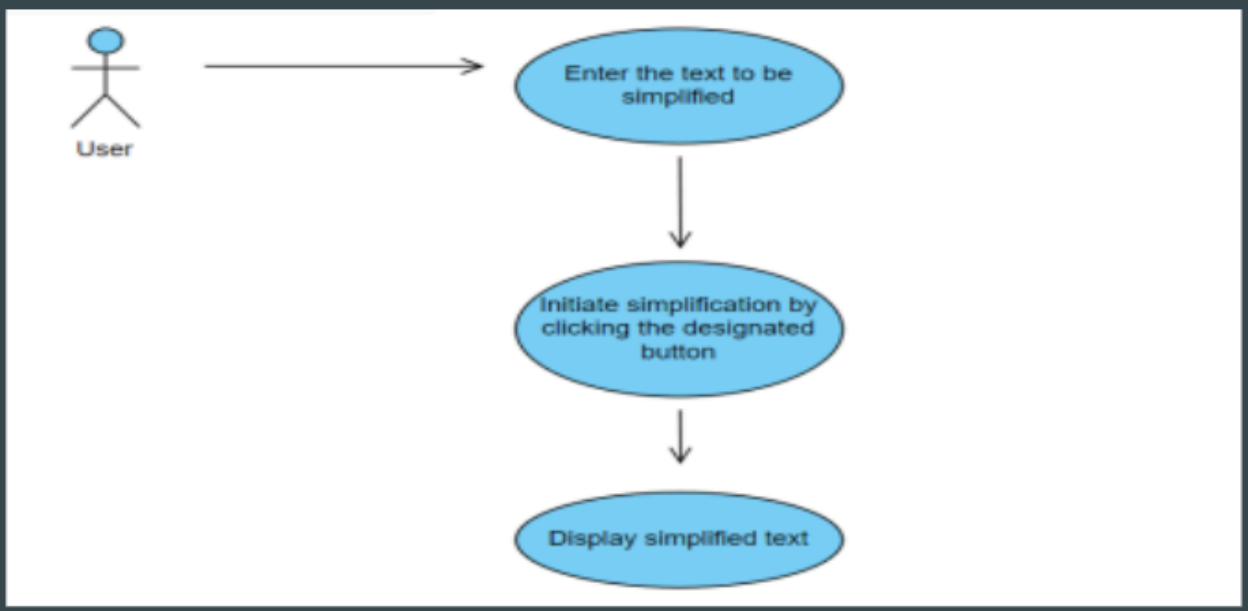
SYSTEM DESIGN-ALGORITHM

Algorithm for Text Simplification:

```
complex words ← CWI(S,t)-ignore list
if number(complex words)>0 then
        w ← head(complex words)
        subs ← Substitution Generation(S,w)
        subs ← Substitute Ranking(subs)
        top ← head(subs)
        if fre(top)>fre(w) then
                Replace(S,w,top)
                ignore list.add(w)
                LST5(S,t,ignore list)
         else
                LST5(S,t,ignore list)
          end if
else
        return S
end if
```

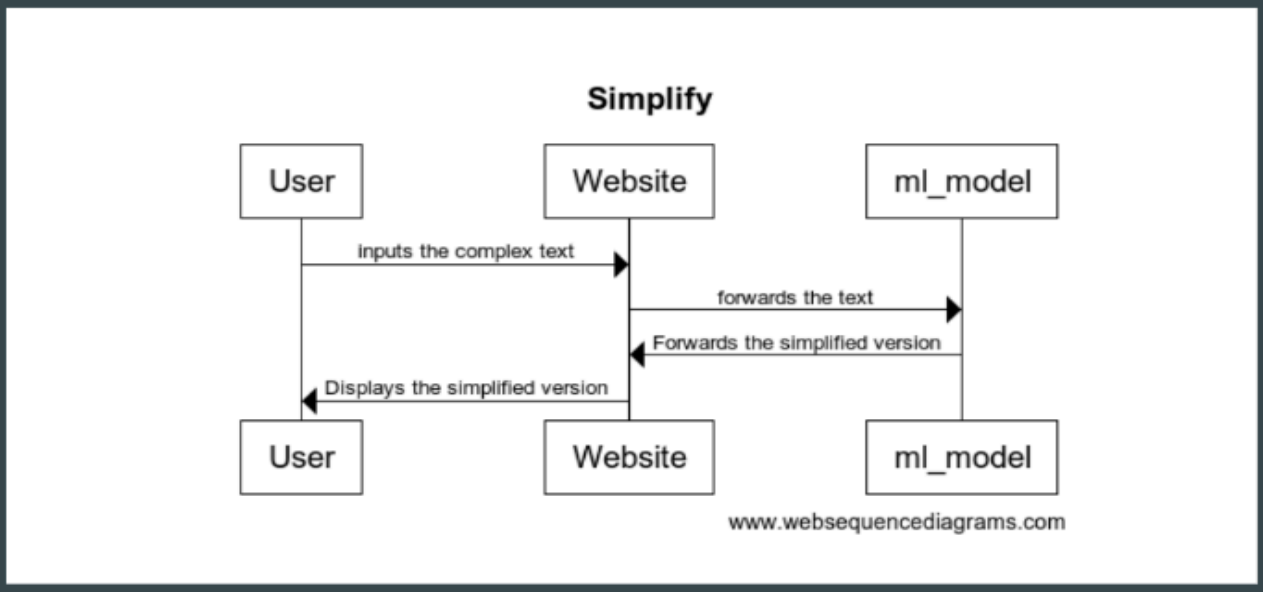

SYSTEM DESIGN-DESIGN MODELS

USE CASE DIAGRAM



SYSTEM DESIGN-DESIGN MODELS

SEQUENCE DIAGRAM



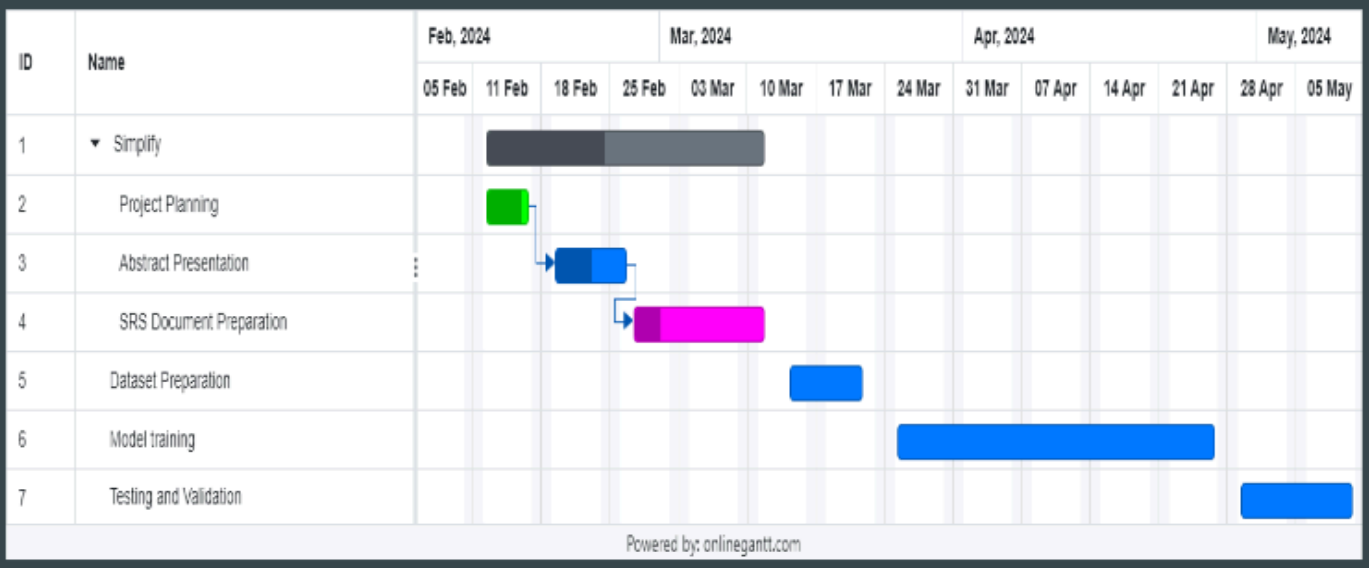
DATASETS

ASSET is a specific dataset used in the field of Natural Language Processing (NLP) for training and evaluating models that simplify sentences.

WikiLarge Dataset , a dataset consisting of pairs of complex sentences and their simplified counterparts is used. It is a collection of text data extracted from Wikipedia articles hence it has an extensive coverage of topics, spanning domains such as science, history, technology, literature, and more.

WORK DIVISION

GANTT CHART

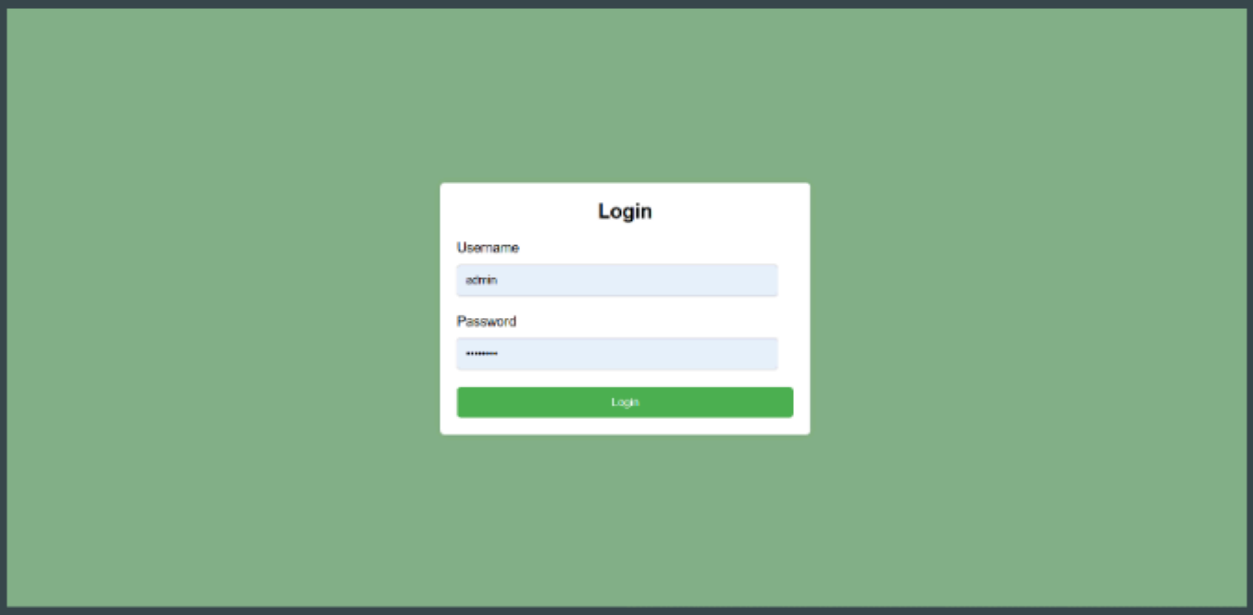


SOFTWARE/HARDWARE REQUIREMENTS

- SOFTWARE
  - Python programming language with flask.
  - T5 and BART model.
  - Transformers,pandas,pytorch and scikit learn libraries for text processing.
  - HTML/CSS

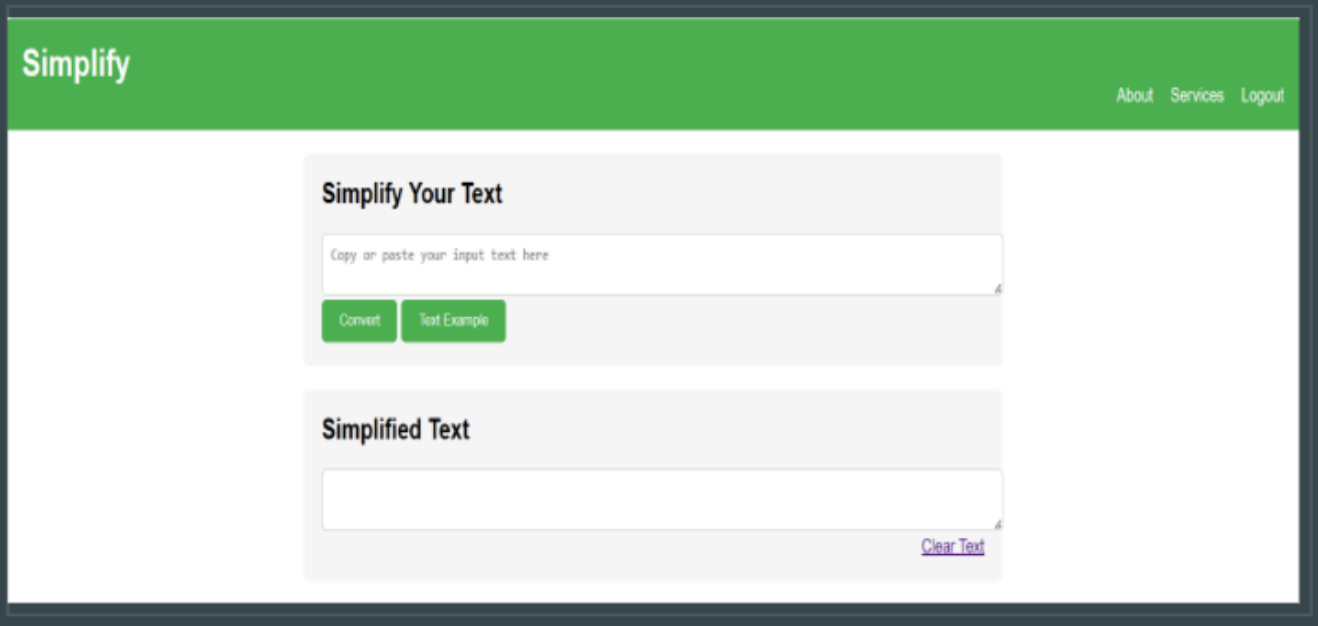
- HARDWARE
  - Standard computer hardware with sufficient processing power, a Dual Core Processor
  - Memory of 8 GB RAM
  - Storage to handle the web application's operation is required.
  - GPU obtained from Colab

RESULTS:- LOGIN PAGE

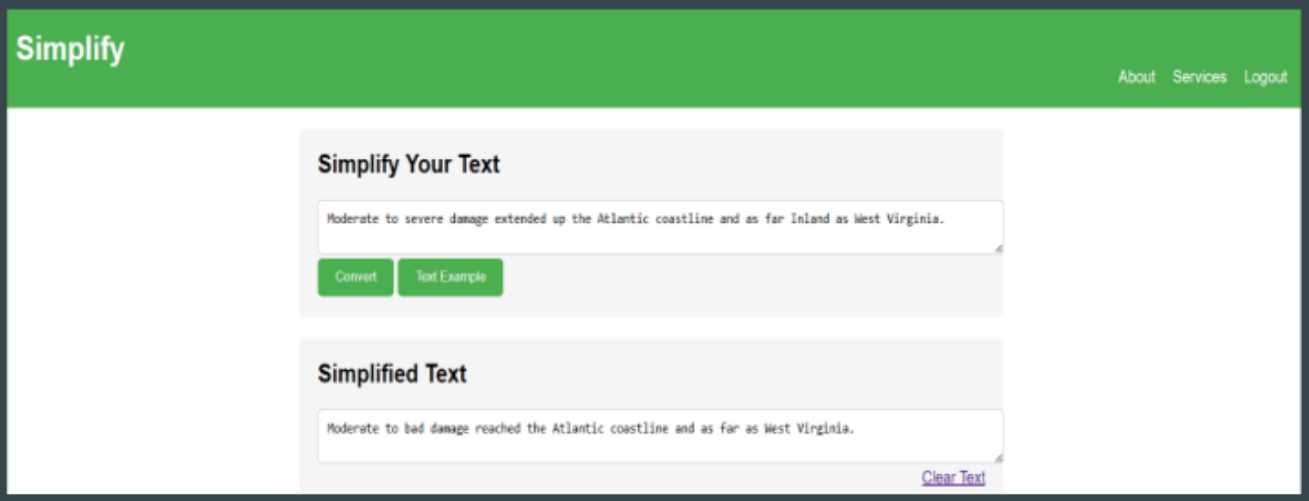


19

MAIN PAGE

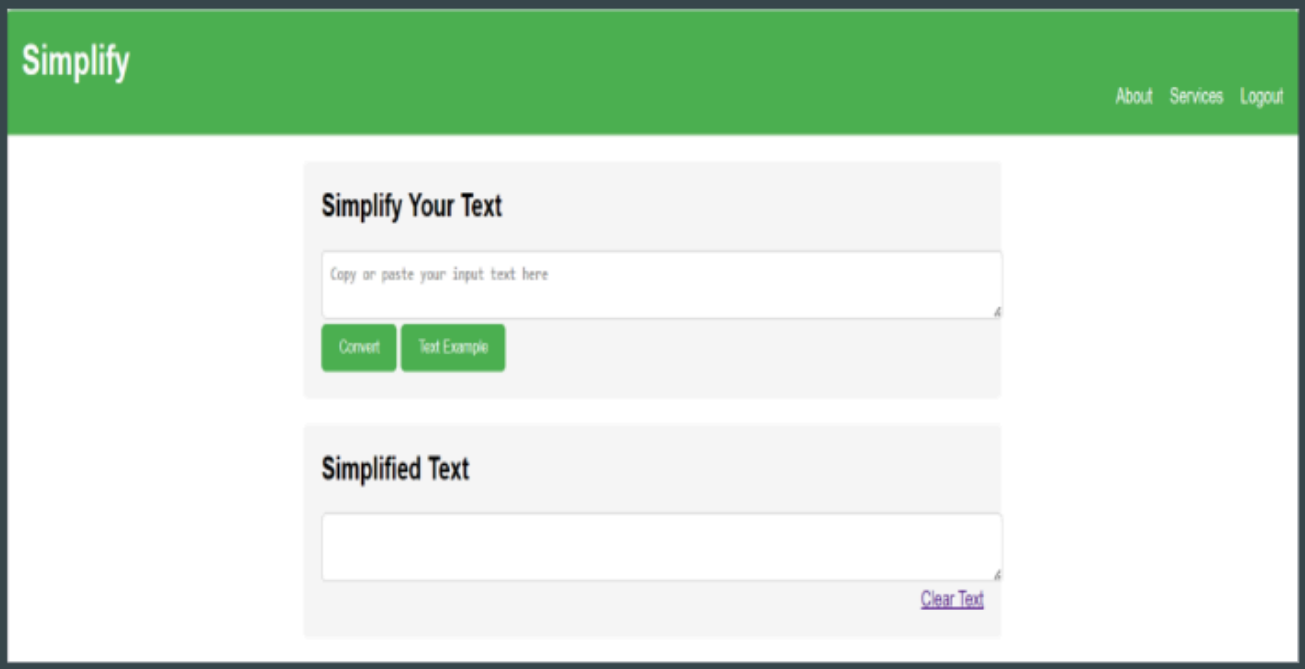


SIMPLIFIED TEXT



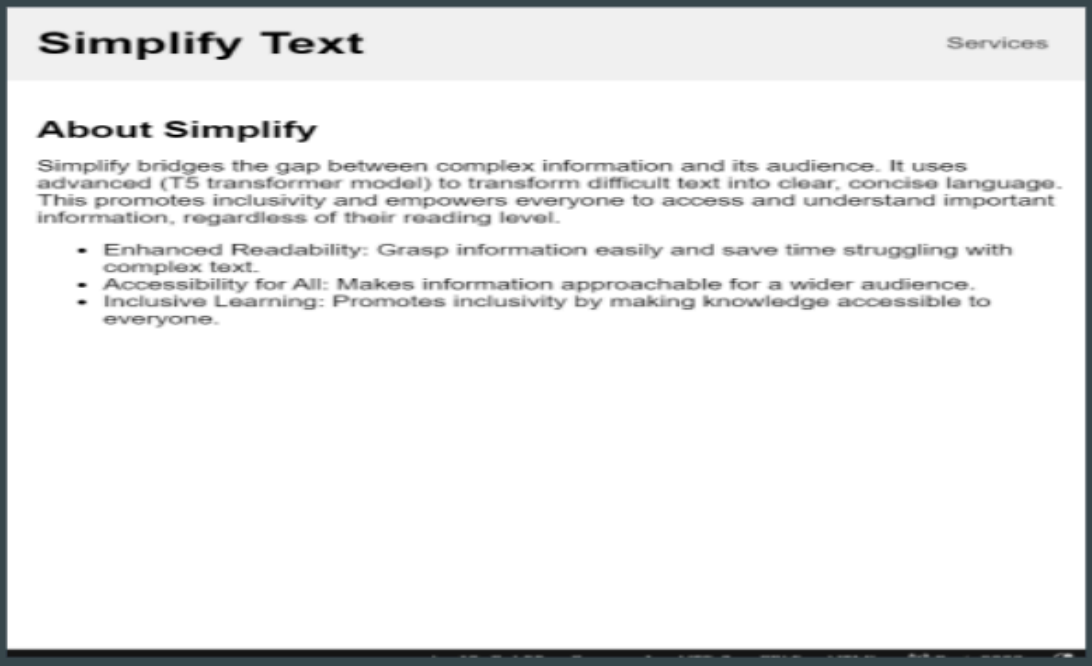
21

AFTER CLEARING



22

ABOUT



23

EVALUATION

Evaluation of T5-small model trained with ASSET Dataset

Evaluating checkpoint 4500: 100%| | 359/359 [02:28<00:00, 2.42it/s]

Some weights of RobertaModel were not initialized from the model checkpoint at roberta-large and are newly initialized: ['roberta.pooler.dense.bias', 'roberta.pooler.dense.weight']

You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.

Results for checkpoint 4500:

result: {'rouge': {'rouge1': 0.7667695102874497, 'rouge2': 0.664445747183364, 'rougeL': 0.7509370194842242, 'rougeLsum': 0.7503693527542086}, 'bleu': {'bleu': 0.8454418776747928, 'precisions': [0.9530487804878048, 0.883578162683622, 0.8143741075678248, 0.7449908925318761], 'brevity_penalty': 1.0, 'length_ratio': 1.1700356718192628, 'translation_length': 4920, 'reference_length': 4205}, 'google_bleu': {'google_bleu': 0.5053406123902208}, 'bertscore': {'precision': 0.9615679049558294, 'recall': 0.9464273902699144, 'f1': 0.9506411384739252}, 'sacrebleu': {'score': 80.71568081913723, 'counts': [4689, 4030, 3422, 2863], 'totals': [4920, 4561, 4202, 3843], 'precisions': [95.3048780487805, 88.3578162683622, 81.43741075678248, 74.49908925318762], 'bp': 0.954715905972489, 'sys_len': 4920, 'ref_len': 5148}, 'sari': {'sari': 41.431428190049665}}

24

EVALUATION

Evaluation of Bart-base model trained with ASSET Dataset

You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.

Results for checkpoint-3375:

result: {'rouge': {'rouge1': 0.7492083367435423, 'rouge2': 0.6335656982742597, 'rougeL': 0.7311808471277131, 'rougeLsum': 0.7313045589083585}, 'bleu': {'bleu': 0.7999286368094305, 'precisions': [0.9337410805300713, 0.8435987681478223, 0.7609266778122761, 0.6831243469174504], 'brevity_penalty': 1.0, 'length_ratio': 1.1664684898929845, 'translation_length': 4905, 'reference_length': 4205}, 'google_bleu': {'google_bleu': 0.4815799920603414}, 'bertscore': {'precision': 0.957394526031356, 'recall': 0.9428519115142503, 'f1': 0.9463866314848154}, 'sacrebleu': {'score': 76.67162700643063, 'counts': [4580, 3835, 3186, 2615], 'totals': [4905, 4546, 4187, 3828], 'precisions': [93.37410805300713, 84.35987681478223, 76.0926677812276, 68.31243469174504], 'bp': 0.9584808378937474, 'sys_len': 4905, 'ref_len': 5113}, 'sari': {'sari': 40.90290371227241}}

CONCLUSION

- This project aims at providing simpler alternatives to complex sentences.
- Its features include: Generation of simplified text,user-friendly interface for input and output,integration of evaluation metrics for quality assessment.

FUTURE ENHANCEMENTS

- Expanding your project to simplify text in multiple languages broadens its usefulness and reach. Creating models that can simplify content across different languages makes it accessible to a wider audience.
- Simplify can be further developed to initiate simplification on a larger input, paragraphs at a time.
- Integrate Simplify with real-world applications (e.g., educational platforms, assistive technology tools).

REFERENCES

- Jipeng Qiang, Yun Li, Yi Zhu, Yunhao Yuan, and Xindong Wu. LSBert: A Simple Framework for Lexical Simplification.
- Sarah Alissa1 and Mike Wald "Text Simplification Using Transformer and BERT"
- Kang Liu, Jipeng Qiang "Sentence Simplification Using Paraphrase Corpus for Initialization".
- Suha S. Al-Thanyyan, Aqil M. Azmi “Automated Text Simplification: A Survey"

Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
RAJAGIRI SCHOOL OF ENGINEERING & TECHNOLOGY (AUTONOMOUS)
RAJAGIRI VALLEY, KAKKANAD, KOCHI, 682039
(Affiliated to APJ Abdul Kalam Technological University)

Vision, Mission, Programme Outcomes and Course Outcomes

Institute Vision

To evolve into a premier technological institution, moulding eminent professionals with creative minds, innovative ideas and sound practical skill, and to shape a future where technology works for the enrichment of mankind.

Institute Mission

To impart state-of-the-art knowledge to individuals in various technological disciplines and to inculcate in them a high degree of social consciousness and human values, thereby enabling them to face the challenges of life with courage and conviction.

Department Vision

To become a centre of excellence in Computer Science and Engineering, moulding professionals catering to the research and professional needs of national and international organizations.

Department Mission

To inspire and nurture students, with up-to-date knowledge in Computer Science and Engineering, ethics, team spirit, leadership abilities, innovation and creativity to come out with solutions meeting societal needs.

Programme Outcomes (PO)

Engineering Graduates will be able to:

1. Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Conduct investigations of complex problems: Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and Team work: Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.

10. Communication: Communicate effectively with the engineering community and with society at large. Be able to comprehend and write effective reports documentation. Make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

Programme Specific Outcomes (PSO)

A graduate of the Computer Science and Engineering Program will demonstrate:

PSO1: Computer Science Specific Skills

The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas by understanding the core principles and concepts of computer science and thereby engage in national grand challenges.

PSO2: Programming and Software Development Skills

The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry.

PSO3: Professional Skills

The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur.

Course Outcomes

After the completion of the course the student will be able to:

CO1:

Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)

CO2:
Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)

CO3:
Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (Cognitive Knowledge Level: Apply)

CO4:
Prepare technical report and deliver presentation (Cognitive Knowledge Level: Apply)

CO5:
Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)

Appendix C: CO-PO-PSO Mapping

COURSE OUTCOMES:

After completion of the course the student will be able to

SL. NO	DESCRIPTION	Blooms' Taxonomy Level
CO1	Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO2	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO3	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO4	Prepare technical report and deliver presentation (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO5	Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)	Level 3: Apply

CO-PO AND CO-PSO MAPPING

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO3
CO1	3	3	3	3		2	2	3	2	2	2	3	2	2	2
CO2	3	3	3	3	3	2		3	2	3	2	3	2	2	2
CO3	3	3	3	3	3	2	2	3	2	2	2	3			2
CO4	2	3	2	2	2			3	3	3	2	3	2	2	2
CO5	3	3	3	2	2	2	2	3	2		2	3	2	2	2

3/2/1: high/medium/low

JUSTIFICATIONS FOR CO-PO MAPPING

MAPPING	LOW/ MEDIUM/ HIGH	JUSTIFICATION
101003/CS622T.1-PO1	**HIGH**	Identify technically and economically feasible problems by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS622T.1-PO2	**HIGH**	Identify technically and economically feasible problems by analysing complex engineering problems reaching substantiated conclusions using first principles of mathematics.
101003/CS622T.1-PO3	**HIGH**	Design solutions for complex engineering problems by identifying technically and economically feasible problems.
101003/CS622T.1-PO4	**HIGH**	Identify technically and economically feasible problems by analysis and interpretation of data.
101003/CS622T.1-PO6	**MEDIUM**	Responsibilities relevant to the professional engineering practice by identifying the problem.
101003/CS622T.1-PO7	**MEDIUM**	Identify technically and economically feasible problems by understanding the impact of the professional engineering solutions.
101003/CS622T.1-PO8	**HIGH**	Apply ethical principles and commit to professional ethics to identify technically and economically feasible problems.
101003/CS622T.1-PO9	**MEDIUM**	Identify technically and economically feasible problems by working as a team.
101003/CS622T.1-PO10	**MEDIUM**	Communicate effectively with the engineering community by identifying technically and economically feasible problems.
101003/CS622T.1-P011	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles by selecting the technically and economically feasible problems.
101003/CS622T.1-PO12	**HIGH**	Identify technically and economically feasible problems for long term learning.
101003/CS622T.1-PSO1	**MEDIUM**	Ability to identify, analyze and design solutions to identify technically and economically feasible problems.
101003/CS622T.1-PSO2	**MEDIUM**	By designing algorithms and applying standard practices in software project development and Identifying technically and economically feasible problems.
101003/CS622T.1-PSO3	**MEDIUM**	Fundamentals of computer science in competitive research can be applied to Identify technically and economically feasible problems.
101003/CS622T.2-PO1	**HIGH**	Identify and survey the relevant by applying the knowledge of mathematics, science, engineering fundamentals.

101003/CS622T.2-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems get familiarized with software development processes.
101003/CS622T.2-PO3	**HIGH**	Design solutions for complex engineering problems and design based on the relevant literature.
101003/CS622T.2-PO4	**HIGH**	Use research-based knowledge including design of experiments based on relevant literature.
101003/CS622T.2-PO5	**HIGH**	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes by using modern tools.
101003/CS622T.2-PO6	**MEDIUM**	Create, select, and apply appropriate techniques, resources, by identifying and surveying the relevant literature.
101003/CS622T.2-PO8	**HIGH**	Apply ethical principles and commit to professional ethics based on the relevant literature.
101003/CS622T.2-PO9	**MEDIUM**	Identify and survey the relevant literature as a team.
101003/CS622T.2-PO10	**HIGH**	Identify and survey the relevant literature for a good communication to the engineering fraternity.
101003/CS622T.2-PO11	**MEDIUM**	Identify and survey the relevant literature to demonstrate knowledge and understanding of engineering and management principles.
101003/CS622T.2-PO12	**HIGH**	Identify and survey the relevant literature for independent and lifelong learning.
101003/CS622T.2-PSO1	**MEDIUM**	Design solutions for complex engineering problems by Identifying and survey the relevant literature.
101003/CS622T.2-PSO2	**MEDIUM**	Identify and survey the relevant literature for acquiring programming efficiency by designing algorithms and applying standard practices.
101003/CS622T.2-PSO3	**MEDIUM**	Identify and survey the relevant literature to apply the fundamentals of computer science in competitive research.
101003/CS622T.3-PO1	**HIGH**	Perform requirement analysis, identify design methodologies by using modern tools & advanced programming techniques and by applying the knowledge of mathematics, science, engineering fundamentals.
101003/CS622T.3-PO2	**HIGH**	Identify, formulate, review research literature for requirement analysis, identify design methodologies and develop adaptable & reusable solutions.

101003/CS622T.3-PO3	**HIGH**	Design solutions for complex engineering problems and perform requirement analysis, identify design methodologies.
101003/CS622T.3-PO4	**HIGH**	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS622T.3-PO5	**HIGH**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools.
101003/CS622T.3-PO6	**MEDIUM**	Perform requirement analysis, identify design methodologies and assess societal, health, safety, legal, and cultural issues.
101003/CS622T.3-PO7	**MEDIUM**	Understand the impact of the professional engineering solutions in societal and environmental contexts and Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions.
101003/CS622T.3-PO8	**HIGH**	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions by applying ethical principles and commit to professional ethics.
101003/CS622T.3-PO9	**MEDIUM**	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
101003/CS622T.3-PO10	**MEDIUM**	Communicate effectively with the engineering community and with society at large to perform requirement analysis, identify design methodologies.
101003/CS622T.3-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering requirement analysis by identifying design methodologies.
101003/CS622T.3-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by analysis, identify design methodologies and develop adaptable & reusable solutions.
101003/CS622T.3-PSO3	**MEDIUM**	The ability to apply the fundamentals of computer science in competitive research and prior to that perform requirement analysis, identify design methodologies.
101003/CS622T.4-PO1	**MEDIUM**	Prepare technical report and deliver presentation by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS622T.4-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems by preparing technical report and deliver presentation.

101003/CS622T.4-PO3	**MEDIUM**	Prepare Design solutions for complex engineering problems and create technical report and deliver presentation.
101003/CS622T.4-PO4	**MEDIUM**	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions and prepare technical report and deliver presentation.
101003/CS622T.4-PO5	**MEDIUM**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and Prepare technical report and deliver presentation.
101003/CS622T.4-PO8	**HIGH**	Prepare technical report and deliver presentation by applying ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
101003/CS622T.4-PO9	**HIGH**	Prepare technical report and deliver presentation effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
101003/CS622T.4-PO10	**HIGH**	Communicate effectively with the engineering community and with society at large by prepare technical report and deliver presentation.
101003/CS622T.4-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work by prepare technical report and deliver presentation.
101003/CS622T.4-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by prepare technical report and deliver presentation.
101003/CS622T.4-PSO1	**MEDIUM**	Prepare a technical report and deliver presentation to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas.
101003/CS622T.4-PSO2	**MEDIUM**	To acquire programming efficiency by designing algorithms and applying standard practices in software project development and to prepare technical report and deliver presentation.
101003/CS622T.4-PSO3	**MEDIUM**	To apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs by preparing technical report and deliver presentation.
101003/CS622T.5-PO1	**HIGH**	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS622T.5-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems by applying engineering and management principles to achieve the goal of the project.

101003/CS622T.5-PO3	**HIGH**	Apply engineering and management principles to achieve the goal of the project and to design solutions for complex engineering problems and design system components or processes that meet the specified needs.
101003/CS622T.5-PO4	**MEDIUM**	Apply engineering and management principles to achieve the goal of the project and use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS622T.5-PO5	**MEDIUM**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO6	**MEDIUM**	Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities by applying engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO7	**MEDIUM**	Understand the impact of the professional engineering solutions in societal and environmental contexts, and apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO8	**HIGH**	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice and to use the engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO9	**MEDIUM**	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PSO1	**MEDIUM**	The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas. Apply engineering and management principles to achieve the goal of the project.

101003/CS622T.5-PSO2	**MEDIUM**	The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry and to apply engineering and management principles to achieve the goal of the project.
101003/CS622T.5-PSO3	**MEDIUM**	The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur and apply engineering and management principles to achieve the goal of the project.