- Shyam Devani (SJD210003)
- Rudy Rajput (RXR210083)

TEXT SUMMARIZATION USING TRANSFORMER MODELS

Introduction

The goal of this project is to use a pre-trained transformer model to summarize an English book from Project Gutenberg. I picked *The Blue Castle* by L. M. Montgomery in UTF 8 text format and downloaded it from the public link as required.

Text summarization means turning a long piece of text into a shorter version while keeping the main ideas and important details. Modern transformer models like BART and T5 work well for this because they can handle long context and create fluent sentences.

In this report I explain how transformers work for summarization, the model I used, the hyperparameters, the ROUGE evaluation results, and how well the summaries match the original text.

- **Source:** Project Gutenberg
- **Book:** *The Blue Castle* (https://www.gutenberg.org/cache/epub/67979/pg67979.txt)
- **Format:** Plain-text UTF-8
- **Total Characters:** ~416,444

After loading the text using Python's requests library, the Project Gutenberg header and footer were removed. The resulting cleaned text begins directly at CHAPTER I, ensuring the model only summarizes novel content.

To fit within transformer sequence limits, the text was divided into approximately 340 chunks, each ~1200 characters.

How Transformers Work for Text Summarization

Transformers rely on three core components:

1. Self-Attention

Allows each word to attend to every other word in the sequence.
This is essential for summarization since the model must understand context across paragraphs, not just locally.

2. Encoder–Decoder Structure

BART (Bidirectional and Auto-Regressive Transformers) uses:

- Encoder: Reads the full text chunk and creates contextual embeddings.
- Decoder: Generates the summary, one token at a time, using attention over encoder outputs.

3. Positional Embeddings

Because transformers do not process text sequentially like RNNs, positional embeddings give the model information about word order.

Model Used: BART (facebook/bart-large-cnn)

We used the pretrained summarization model:

facebook/bart-large-cnn

Key Architecture Details

- 12 encoder layers
- 12 decoder layers
- 406M parameters
- Pretrained on CNN/DailyMail summarization dataset
- Designed specifically for long-document summarization

Why BART?

- Strong performance on abstractive summarization
- High ROUGE scores compared to other models
- Handles long, narrative text better than small models (like T5-small)

Methodology

Preprocessing

- Removed Gutenberg licensing text
- Stripped newline formatting
- Split text into chunks using textwrap.wrap(cleaned, 1200)
- Total chunks: 340

Summarization Pipeline

Using HuggingFace pipeline:

summarizer = pipeline(

```
    "summarization",
    model="facebook/bart-large-cnn",
    tokenizer="facebook/bart-large-cnn"
)
```

Hyperparameters

Used default BART summarization settings:

- max_length = 150
- min_length = 60
- do_sample = False (greedy decoding)
- Summarized first 30 chunks due to computation limits on CPU

Evaluation Using ROUGE

ROUGE measures overlap between model summary and original text:

- ROUGE-1: unigrams
- ROUGE-2: bigrams
- ROUGE-L: longest common subsequence

The rouge-score library was used.

Results

Example Summary Output

For the first chunk of the novel:

Generated Summary:

> “Valancy Stirling's whole life would have been entirely different if it had not rained on a certain May morning. She would have gone with her family to Aunt Wellington’s engagement picnic. But the rain forces her to reflect alone, revealing her lifelong emotional struggles, social pressure from her mother, and her quiet hopes for romance that have not yet materialized.”

This summary captures the primary themes:

- Rain changing events
- Introduction of Valancy

- Isolation
- Emotional conflict

ROUGE Score Table (First 5 Chunks)

Chunk	ROUGE-1	ROUGE-2	ROUGE-L
0	0.4129	0.3836	0.3319
1	0.4397	0.4143	0.4397
2	0.3360	0.3065	0.3200
3	0.3779	0.3175	0.2598
4	0.4370	0.4254	0.3777

Interpretation

- ROUGE-1 scores between 0.33–0.44 indicate strong coverage of important words.
- ROUGE-2 around 0.30–0.42 is very good for literary text (which is harder to summarize than news).
- ROUGE-L scores show that summaries capture narrative flow reasonably well.

Overall, the model performed well given the long, descriptive writing style of early 20th-century fiction.

Discussion & Analysis

Strengths

- BART produced coherent, faithful summaries, even though the text is complex and not similar to its training data.
- The model captured:
    - Key events
    - Character emotions
    - Scene descriptions
    - Narrative themes
- ROUGE scores indicate high-quality summarization for multiple chunks.

Limitations

- Summaries sometimes simplify character details too much.

- Because only 30 chunks were summarized (due to runtime limits), the project does not cover the full novel.
- Literary fiction has less “compressible” structure compared to news articles, making ROUGE scores naturally lower.

Conclusion

This project successfully applied the BART transformer to perform abstractive summarization on *The Blue Castle*. The model generated coherent summaries and achieved solid ROUGE metrics across multiple chunks. The results demonstrate the effectiveness of transformer architectures for summarizing long-form narrative text.

Overall, this work shows how modern NLP techniques and pretrained transformers can be applied directly to real-world datasets with minimal preprocessing while producing strong analytical and quantitative results.