Urdu Abstractive Text Summarization

NLP Project Report

Session: 2021-2025

Submitted by:

Messam Naqvi

Submitted to:
Dr. Quratul Ain Akram

Department of Computer Science, New Campus
University of Engineering and Technology
Lahore, Pakistan

Contents

Chapter 1

1 Report Synopsis

1.1 Introduction

The capacity to automatically produce succinct and useful summaries from massive amounts of text data has grown in value in today's information-driven environment. Abstractive summarization is a capability that is used in a variety of fields, including content creation, document summarization, and news aggregation. Although abstractive summarization has advanced significantly in languages like English, Urdu and other languages with limited resources and tools still face difficulties in this regard.

The objective of this project is to close the gap in Urdu abstractive summarization by utilising pre-trained models and cutting edge natural language processing (NLP) approaches. To create a system that can provide relevant and cogent summaries in Urdu, we combine data preprocessing, fine-tuning, and model training. The project makes use of pre-trained transformer-based models, including BART (Bidirectional and Auto-Regressive Transformers), and the power of deep learning frameworks, especially PyTorch. In order to accomplish our goal, we first preprocess an Urdu text dataset by cleaning and tokenizing the source documents as well as the summaries that go with them. Next, to enable efficient data loading and processing during model training, we build a new dataset class. The preprocessed Urdu data is used to fine-tune our selected model architecture, BART, making it suitable for the purpose of abstractive summarization in Urdu. To guarantee that the model produces high-quality summaries, we use suitable loss functions and optimise important hyperparameters during the training phase.

By the project's conclusion, we want to have created a reliable and adaptable system that can produce abstractive summaries in Urdu, making it easier to extract information and summarise content in one of the most commonly spoken languages in the world. By means of this project, we foster inclusivity and accessibility in the natural language processing area by furthering the field's research in natural language processing and opening the door for the creation of comparable systems in other languages with little resources.

1.2 Problem Statement

This project explores the application of abstractive summarization techniques for Urdu talkshows. We will leverage existing datasets to fine-tune a pre-trained model and evaluate its performance on unseen talkshow data.

1.3 Objectives

To realize our project's vision, we have outlined the following concrete objectives to structure our work and measure our progress:

- Develop an understanding of abstractive summarization techniques and their applications in natural language processing (NLP).
- Gain proficiency in working with pre-trained language models such as mT5 for text summarization tasks.

- Create Urdu Summarization dataset using the provided guidelines.
- Explore and utilize existing datasets of Urdu summarization for model training and evaluation.
- Implement and fine-tune the selected model using the provided raw data of Urdu talk shows to generate abstractive summaries.
- Evaluate the performance of the model using appropriate evaluation metrics and conduct subjective analysis of the generated summaries.

1.4 Proposed Methodology/System

- **Preprocessing**:
  - Excel script files were converted into text format.
  - The provided raw data of Urdu talk shows was cleaned and tokenized.
  - The data was organized into scripts and summaries.
  - The cleaned data was saved into a suitable format for further processing.
- **Dataset Preparation**:
  - A custom dataset class (UrduDataset) was created to handle the cleaned data.
  - Scripts and summaries were paired correctly for input to the model.
- **Model Selection**:
  - A pre-trained language model suitable for abstractive summarization tasks in Urdu, mirfan899/usum, was selected.
  - The model is a sequence-to-sequence transformer-based model from the Hugging Face library, likely similar to T5 or BART.
  - Exploration of options such as mT5 or other models trained on Urdu language data was also conducted.
- **Fine-tuning the Model**:
  - The selected model, mirfan899/usum, and tokenizer were initialized.
  - Hyperparameters such as batch size, learning rate, and number of epochs were defined.
  - The model was trained on the prepared dataset by fine-tuning its weights on the task of generating abstractive summaries.
- **Evaluation**:
  - The trained model's performance was assessed using metrics like ROUGE scores.
  - Subjective analysis of generated summaries was conducted to evaluate their quality and coherence.

1.5 Training Results

Training Loss	Epoch	Step
3.405	1	1
3.210	1	2
4.056	1	3
3.202	1	4
3.193	1	5
3.064	1	6
3.007	1	7
3.018	1	8
2.982	1	9
3.021	1	10
3.076	1	11
3.022	1	12
2.916	1	13
4.031	2	1
3.666	2	2
2.811	2	3
2.471	2	4
3.680	2	5
3.467	2	6
3.206	2	7
3.120	2	8
2.201	2	9
3.028	2	10
3.668	2	11
3.886	2	12
1.854	2	13
2.828	3	1
3.644	3	2
2.612	3	3
4.182	3	4
3.780	3	5
3.216	3	6
3.400	3	7
2.735	3	8
2.769	3	9
3.553	3	10
3.927	3	11
2.922	3	12
2.256	3	13

Key Findings:

- The training loss decreases gradually over epochs and steps, indicating that the model is learning from the data.

- The loss decreases significantly in some steps, indicating that the model is making substantial improvements at those points.
- Overall, the model seems to be converging as the training progresses.

1.6 Evaluation Results

Reference File Name	Candidate File Name	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore
105.txt	105u_summary.txt	0.2542	0.0507	0.1356	0.6547
77.txt	77u_summary.txt	0.3687	0.1321	0.2120	0.7253
79.txt	79u_summary.txt	0.3229	0.0692	0.2083	0.7116
80.txt	80u_summary.txt	0.2222	0.0598	0.0972	0.6639
78.txt	78u_summary.txt	0.2395	0.0588	0.1557	0.6746

1.7 Overall Model Performance

- The model exhibits varying degrees of performance across different summaries. For instance, summaries generated for certain files might demonstrate higher ROUGE and BERTScore values compared to others, indicating disparities in the model's effectiveness in capturing the essence of the input data.
- While ROUGE-1 scores tend to be relatively higher compared to ROUGE-2 and ROUGE-L scores across most summaries, indicating that the model is proficient in capturing individual words, it might struggle with maintaining coherence and fluency in longer sequences or pairs of words.
- The BERTScore values suggest that the semantic similarity between the generated summaries and the reference summaries is moderate, but this varies across different summaries. Some summaries might exhibit closer semantic alignment with the reference summaries compared to others.
- Among the summaries, the one associated with **77.txt / 77u_summary.txt** stands out as having the highest ROUGE and BERTScore values, indicating better overall quality and semantic similarity compared to the others. Conversely, **80.txt / 80u_summary.txt** appears to have the lowest scores, suggesting that the generated summary for this file might be less coherent and less semantically similar to the reference summary.

1.8 Conclusion

- The model performs moderately well, with room for improvement especially in capturing bigrams and sequences as indicated by the lower ROUGE-2 and ROUGE-L scores.
- The BERTScore suggests that while the summaries are somewhat semantically similar, there is still a significant gap to be closed for higher quality summaries.
- To improve, consider enhancing the training dataset, using more advanced models, or applying more sophisticated fine-tuning techniques.