Extractive Text Summarization using SummaRuNNer

Kanishk Dad, Dhruv Limbad, Dhyey Patel, Aarzoo Khambhoo

Abstract

This paper explores extractive text summarization using SummaRuNNer, a Recurrent Neural Network (RNN) based sequence classifier. The model is applied to the task of summarizing large datasets of news articles, and its performance is evaluated using standard metrics such as ROUGE scores. The paper highlights the implementation, challenges, results, and learnings from the project.

Keywords

Extractive summarization, SummaRuNNer, Text summarization, Recurrent Neural Networks

1. Introduction

Extractive text summarisation is a critical task in natural language processing that involves selecting the most relevant sentences from a document to create a concise summary. SummaRuNNer is a sequence classifier that uses Recurrent Neural Networks (RNNs) for this purpose. This report provides an overview of the implementation of SummaRuNNer, discusses the dataset, evaluation metrics, results, key challenges faced, and the learnings from the process.

2. Problem

The primary goal is to create a robust extractive summarisation model that can:

- Identify the most salient sentences in a document.
- Handle large datasets efficiently.
- Achieve state-of-the-art performance with interpretable decisions.

The problem was balancing content richness, salience, novelty, and positional importance for each sentence.

3. Dataset

The CNN/Daily Mail dataset was used for training and evaluation. The dataset comprises:

- **Training set:** 287,113 documents.
- **Validation set:** 13,368 documents.
- **Test set:** 11,490 documents.

Each document includes an article and a human-generated abstractive summary, which was converted into extractive labels using a greedy approach maximizing ROUGE scores.

Introduction to Information Retrieval (IT550), Autumn 2024, DAIICT, India

202103005@daiict.ac.in,202101263@daiict.ac.in,202103053@daiict.ac.in,202103026@daiict.ac.in
(K. D. D. L. D. P. A. Khambhoo)
0000-0002-1234-5678 (K. D. D. L. D. P. A. Khambhoo)

4. Evaluation Metrics

The performance of the model was evaluated using ROUGE scores:

- **ROUGE-1:** Measures unigram overlap.
- **ROUGE-2:** Measures bigram overlap.
- **ROUGE-L:** Measures the longest common subsequence.

These metrics provide a quantitative evaluation of the model's ability to generate summaries similar to the ground truth.

5. Implementation Changes

The SummaRuNNer model was implemented with the following modifications:

Listing 1: Code Snippet: Preprocessing and Model Adjustments

```
# Data Preprocessing
def preprocess_data(data, max_sentences=30, max_sentence_length=50):
    # Tokenizing and padding sentences
    for entry in data:
        sentences = entry['article'].split(". ")
        tokenized_sentences = [
            tokenizer.encode(sent, truncation=True, max_length=
                max_sentence_length, padding="max_length")
            for sent in sentences
        ]
        # Additional preprocessing to handle edge cases
        ...
    return tokenized_articles, tokenized_labels

# Model Adjustments
class SummaRuNNer(nn.Module):
    def __init__(self, embedding_dim, hidden_dim, vocab_size):
        super(SummaRuNNer, self).__init__()
        # Added dropout to prevent overfitting
        self.dropout = nn.Dropout(0.3)
        ...
```

These changes improved model stability and generalization.

6. Results

The following results were achieved on the test set:

- **ROUGE-1:** 27.59%
- **ROUGE-2:** 12.55%
- **ROUGE-L:** 18.64%

Compared to previous approaches, SummaRuNNer showed similar ROUGE scores, for all three Rouge-1, Rouge-2, and Rouge-L.

7. Key Challenges and Learnings

7.1. Challenges

- **Dataset Imbalance:** Handling the imbalance between summary and non-summary sentences was difficult.
- **High Computational Cost:** Training required significant computational resources.
- **Extractive Labels:** Generating extractive labels from abstractive summaries introduced noise.

7.2. Learnings

- Incorporating positional embeddings improved model performance.
- Using dropout layers helped mitigate overfitting in large datasets.
- Visualization of abstract features enabled better interpretability of model decisions.

8. Conclusion

SummaRuNNer [1] provides an effective and interpretable solution for extractive text summarization. With further optimizations, such as better label generation methods, it has the potential to outperform existing approaches consistently.

References

[1] R. Nallapati, F. Zhai, B. Zhou, Summarunner: A recurrent neural network based sequence model for extractive summarization of documents, arXiv preprint arXiv:1611.04230 (2016). Available at https://arxiv.org/abs/1611.04230.