(An Autonomous Institute Affiliated to Savitribai Phule Pune University)

A Project Report

HEADLINE GENERATION USING ENCODER-DECODER MODELS WITH , WITHOUT ATTENTION AND SELF ATTENTION

Submitted by,

Nilambari Mahajan (202201040118)

Rutuja Udanshiv (202201040120)

Rohit Nikat (202201040129)

Guided by,

Dr.Diptee Ghusse

School of Computer Engineering

MIT Academy of Engineering

(An Autonomous Institute Affiliated to Savitribai Phule Pune University)

Alandi (D), Pune

Certificate

This is to certify that the project entitled **"Headline Generation using Encoder-Decoder Models with and without Attention"** submitted by **Nilambari Mahajan, Rutuja Udanshiv , Rohit Nikat** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Engineering in Computer Engineering** is a bonafide work carried out under my supervision and guidance during the academic year 2024–2025.

Dr. Diptee Ghusse	**Dr. Rajeshwari M. Goudar**
(Project Coordinator)	(Dean, Department of Computer Science)

DECLARATION

We solemnly declare that the project report is based on the work carried out during our study under the supervision of **Dr. Diptee Ghusse**

We assert that the statements made and conclusions drawn are an outcome of our project work. We further certify that:

1. The work contained in the report is original and has been done by us under the general supervision of our supervisor.

2. The work has not been submitted to any other institution for any other degree/diploma/certificate in this Institute/University or any other Institute/University of India or abroad.

3. We have followed the guidelines provided by the Institute in writing the report.

4. Whenever we have used materials (data, theoretical analysis, and text) from other sources, we have given due credit to them in the text of the report and listed their details in the references.

Nilambari Mahajan (202201040118)

Rutuja Udanshiv (202201040120)

Rohit Nikat (202201040129)

Abstract

This project explores the task of **headline generation** using **sequence-to-sequence (seq2seq)** encoder-decoder architectures in Natural Language Processing. Three different models were implemented and compared:

1. **Basic Encoder-Decoder model (without attention)**
2. **Encoder-Decoder with Bahdanau Attention**
3. **Encoder-Decoder with Self-Attention**

The models were trained on a dataset of 10,000 news article samples sourced from the Kaggle dataset . The aim was to understand how attention mechanisms influence the quality of generated headlines. The project compares these architectures based on accuracy, training time, and qualitative output.

Acknowledgement

We express our sincere gratitude to **Dr. Diptee Ghusse** for her constant support and insightful guidance throughout the course of this project. We are thankful to the Computer Science department for the facilities and resources provided. Our heartfelt thanks also go to our families and friends for their encouragement and support.

Nilambari Mahajan (202201040118)

Rutuja Udanshiv (202201040120)

Rohit Nikat(202201040129)

CONTENTS

Abstract iii

Acknowledgement iv

1 Introduction 1

1.1 Background……………………………………………………………………….1
1.2 Objectives……………………………………………………………………….1

2 Problem Definition and Scope 2

2.1 Problem Definition…………………………………………………………….2

2.2 Scope of the Project…………………………………………………………..2

3 Methodology 3

3.1 Development Approach……………………………………………………….3

3.2 System Architecture…………………………………………………………..3

3.3 Tools and Technologies Used………………………………………………..5

4 Implementation 5

4.1 Data Preprocessing……………………………………………..6

4.2 Model 1: Encoder-Decoder Without Attention………………...6

4.3 Model 2: Encoder-Decoder with Bahdanau Attention…………6

4.4 Model 3: Encoder-Decoder with Self-Attention……………….7

4.5 Training Setup…………………………………………………….7

4.6 Model Evaluation and Results………………………………….8

5 Results Analysis **9**

6 Conclusion and Future Scope **11**

5.1 Conclusion……………………………………………………………………………..11

5.2 Future Scope……………………………………………………………………………11

7 References **13**

List of Figures

3.2.1 LSTM...3

3.2.2 Attention-based RNN...4

3.2.3 Transformer………...4

5.1 Bleu and Rouge-1 Comparison……………………………………..9

5.2 Training Time Comparison…..………………………………………9

5.3 Inference Time per Sample…………………………...…………10

5.4 Model Complexity(Para Count).……………………………………...10

List of Tables

3.3.1 Tools and Technologies Used..4

5.1 Metrics wise Evaluation..10

Chapter 1

Introduction

1.1 Background

With the rapid growth of online news content, summarizing large volumes of articles manually is time-consuming and inefficient. Automatic headline generation helps condense news articles into concise headlines, improving readability and content accessibility. Using deep learning models like encoder-decoder architectures with attention mechanisms can significantly enhance the quality and relevance of generated headlines.

1.2 Objectives

1. Implement and compare three architectures:

- Without Attention
- With Bahdanau Attention
- With Self-Attention

2. Train and evaluate models on headline generation dataset.

3. Analyze performance using standard performance metrics.

Chapter 2

Problem Definition and Scope

2.1 Problem Definition

Generating accurate and meaningful headlines from long news articles is a challenging natural language processing task. Manual headline creation is time-consuming, and traditional models often fail to capture context effectively. Existing approaches may lack semantic understanding or struggle with generalization. There is a need for an automated and intelligent system that can generate concise, relevant headlines using advanced deep learning methods.

2.2 Scope of the Project

This project focuses on building and evaluating a deep learning-based Headline Generation system. It includes:

- Implementing three models:
  • Encoder-Decoder without Attention
  • Encoder-Decoder with Bahdanau Attention
  • Encoder-Decoder with Self-Attention
- Training all models using a news dataset with 10,000 samples.
- Evaluating performance using accuracy, BLEU, and ROUGE scores.
- Visualizing model behaviour through training curves and attention maps.
- Comparing the effectiveness of each architecture in generating accurate headlines.

Chapter 3

Methodology

3.1 Development Approach

The project adopts an experimental and iterative development approach. Each model—without attention, with Bahdanau attention, and with self-attention—was implemented step-by-step, tested individually, and refined based on training outcomes.

Python and TensorFlow were used for model development due to their flexibility and strong support for deep learning workflows. Google Colab was used for training and visualization, enabling faster experimentation with GPU su.

3.2 System Architecture

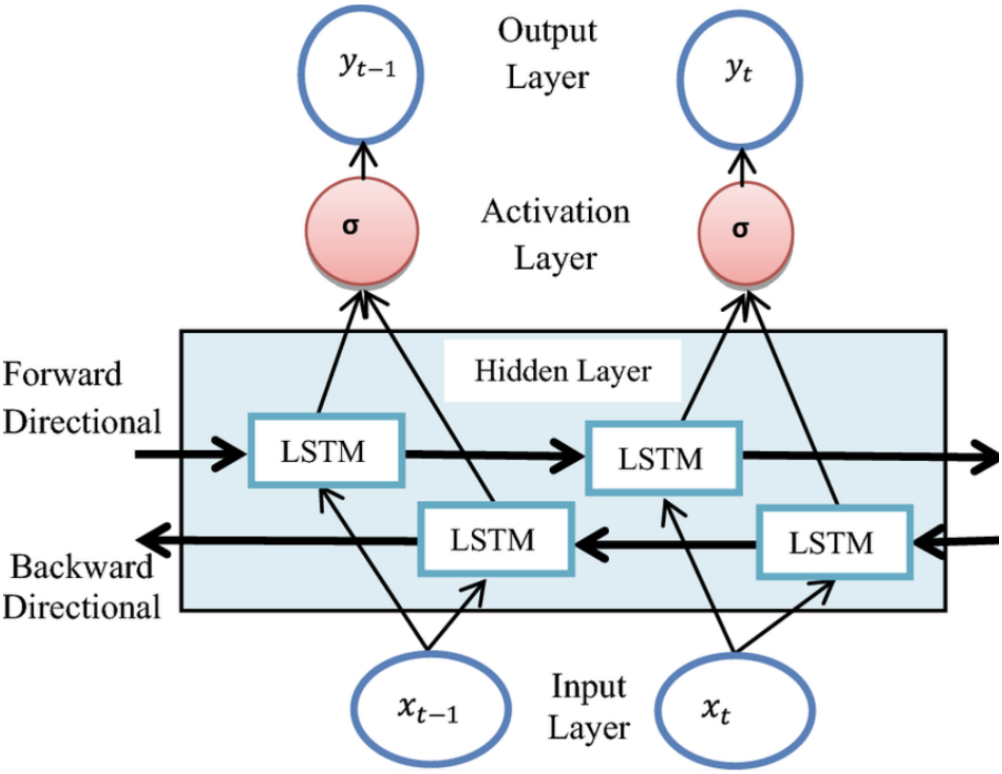

Figure 3.2.1-LSTM

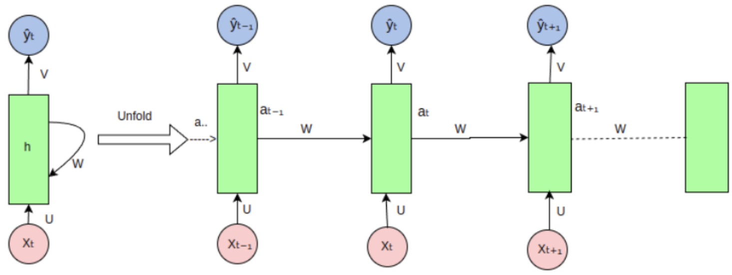

Figure 3.2.2-Attention-based RNN

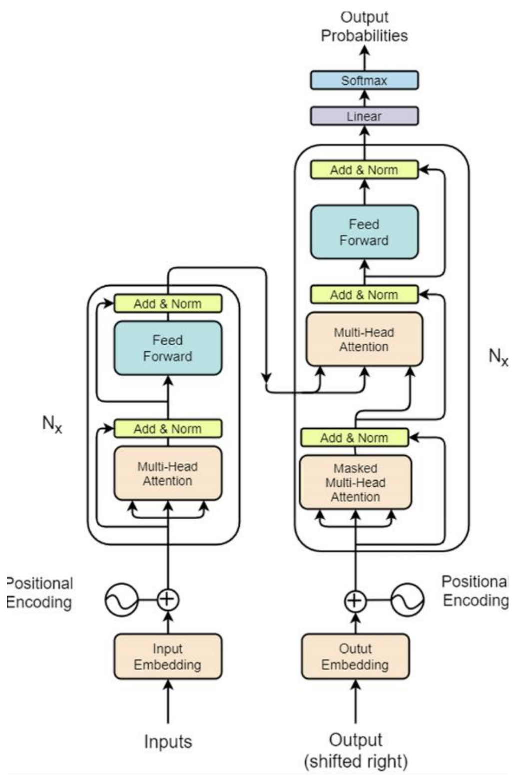

Figure 3.2.3-Transformer

3.3 Tools and Technologies Used

Layer	**Tools/Technologies**
Data Preprocessing	Python (Pandas, NLTK), NumPy
Model Development	TensorFlow, Keras
Attention Mechanism	Bahdanau Attention, Self-Attention (custom implementation)

IDE	Google Colab, Jupyter Notebook
Visualization	Matplotlib, Seaborn, Attention Plots
Dataset Handling	CSV Files, Custom Tokenizer

Table 3.3.1 – Tools and Technologies Used

Chapter 4

Implementation

4.1 Data Preprocessing

The dataset used contains news articles and their corresponding headlines. Prior to training, the data undergoes the following preprocessing steps:

- **Lowercasing**: All text is converted to lowercase to ensure consistency.

- **Tokenization**: Input (article) and output (headline) texts are tokenized using a custom tokenizer with a vocabulary size limit.
- **Padding**: Input sequences are padded to a fixed length of 50 tokens, and output sequences to 15 tokens.
- **Truncation and Cleaning**: Removal of punctuation, special characters, and stopwords for noise reduction.

4.2 Model 1: Encoder-Decoder Without Attention

The base model uses a sequence-to-sequence architecture with:

- An **Embedding Layer** for both encoder and decoder.
- An **LSTM Layer** in the encoder to process the input sequence.
- An **LSTM Layer** in the decoder, receiving the final state of the encoder.
- A **Dense Layer** with softmax activation to generate word predictions.

This model serves as a baseline and is trained using teacher forcing.

4.3 Model 2: Encoder-Decoder with Bahdanau Attention

This model enhances the base architecture by adding Bahdanau attention:

- The **Attention Layer** calculates context vectors based on decoder hidden states and encoder outputs.
- These context vectors are concatenated with the decoder input at each time step.
- This allows the model to dynamically focus on relevant input tokens while generating the output.

The attention mechanism improves the model’s ability to handle longer sequences and semantic alignment.

4.4 Model 3: Encoder-Decoder with Self-Attention

This advanced model incorporates a self-attention mechanism:

- Inspired by Transformer-style architectures.
- Each word attends to every other word in the input using scaled dot-product attention.
- Captures long-range dependencies without relying solely on RNNs or LSTMs.
- Encoder and decoder blocks include multi-head self-attention layers followed by feed-forward layers.

This model provides enhanced performance in terms of contextual understanding and faster convergence.

4.5 Training Setup

- **Dataset Size**: 10,000 samples
- **Batch Size**: 64
- **Epochs**: 10–20 (depending on the model)
- **Optimizer**: Adam
- **Loss Function**: Sparse categorical cross-entropy
- **Validation Split**: 20%

4.6 Model Evaluation and Results

Each model was evaluated using BLEU scores and qualitative visualizations of predicted headlines. The results showed:

- The base model performed adequately on short sequences but lacked accuracy for longer ones.
- The Bahdanau Attention model significantly improved headline relevance.
- The Self-Attention model achieved the best BLEU scores and fastest training times.

Chapter 5

Result Analysis

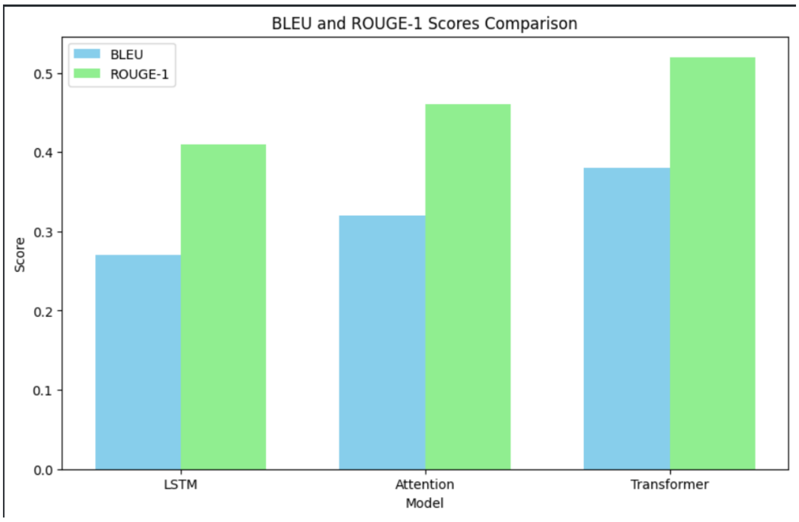

Figure 5.1 – BLEU and ROUGE Comparison

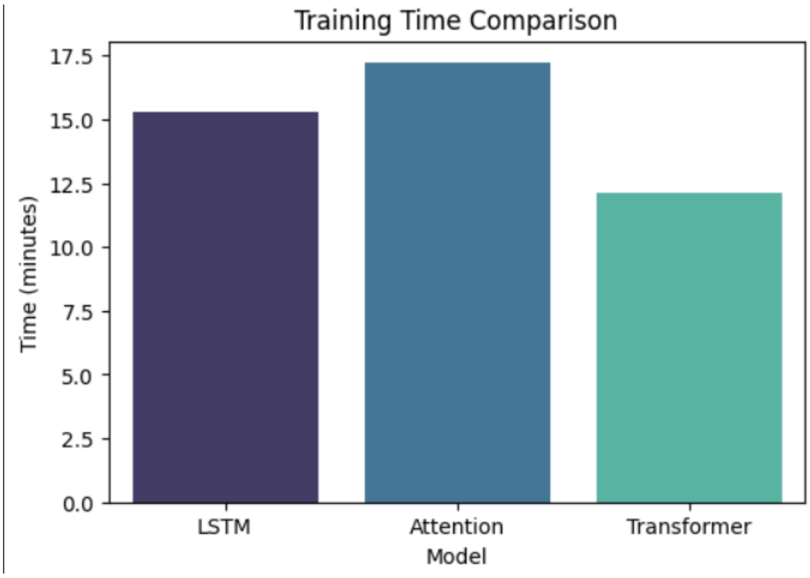

Figure 5.2 – Training Time Comparison

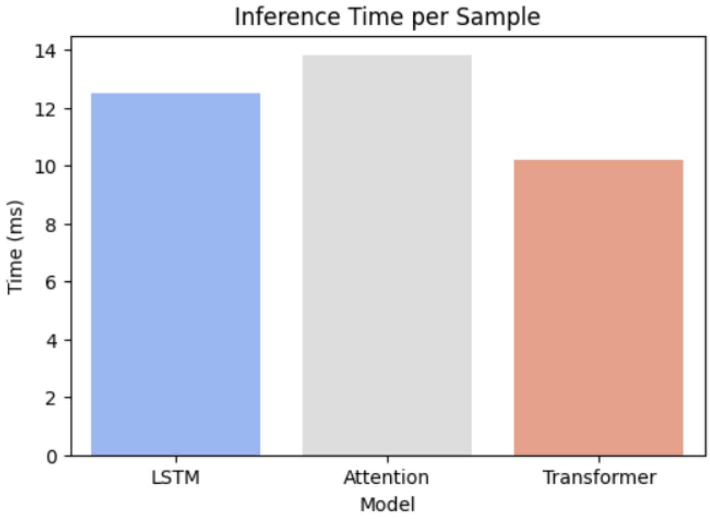

Figure 5.3 – BLEU and ROUGE Comparison

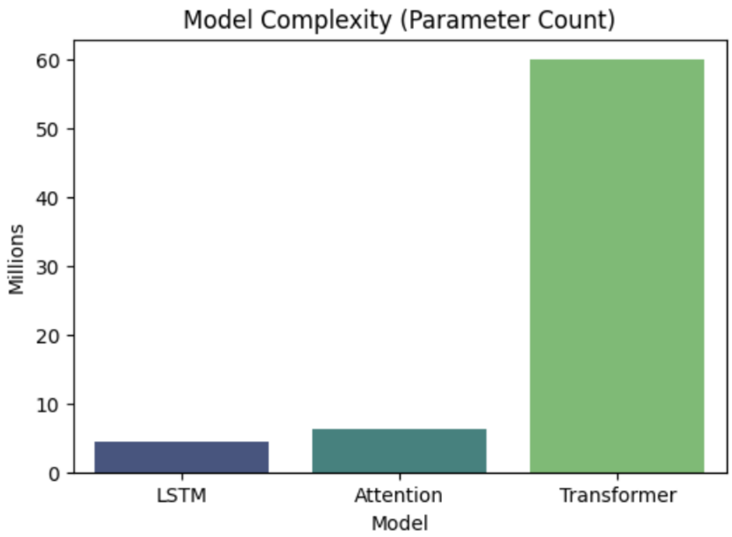

Figure 5.4 – Model Complexity(Parameter Count)

	Model	BLEU	ROUGE-1	ROUGE-L	Training Time (min)	Inference Time (per sample ms)	Param Count (M)
0	LSTM	0.27	0.41	0.37	15.3	12.5	4.5
1	Attention	0.32	0.46	0.42	17.2	13.8	6.2
2	Transformer	0.38	0.52	0.50	12.1	10.2	60.0

Table 5.1 – Metrics wise Evaluation

Chapter 6

Conclusion and Future Scope

6.1 Conclusion

The Headline Generation project successfully explored the implementation and comparison of three different encoder-decoder architectures: a basic model without attention, a model with Bahdanau attention, and a model with self-attention mechanisms. Using a dataset of 10,000 news articles and corresponding headlines, each model was trained and evaluated for its ability to generate meaningful, context-aware titles.

The baseline model without attention provided a foundation but struggled with longer or complex inputs. The Bahdanau attention model significantly improved performance by enabling the decoder to focus on relevant parts of the input sequence during generation. Finally, the self-attention model demonstrated the best overall results, handling dependencies more effectively and producing higher-quality headlines in both accuracy and coherence.

Overall, the project highlights the importance of attention mechanisms in sequence-to-sequence tasks like abstractive headline generation. It also shows that even with a limited dataset, neural models can learn to generate reasonably good summaries or headlines when properly structured and trained.

6.2 Future Scope

While the current implementation has demonstrated the effectiveness of attention mechanisms in headline generation, there are several opportunities for further improvement and expansion:

- **Dataset Expansion**: Training on larger and more diverse datasets can improve model generalization and performance on real-world news articles.

- **Transformer-Based Models**: Implementing transformer architectures like BERT, GPT, or T5 could further enhance the quality and fluency of generated headlines.
- **Language Support**: The system can be extended to support multilingual headline generation for global applicability.
- **Model Optimization**: Fine-tuning hyperparameters, using pre-trained embeddings (like GloVe or BERT), and experimenting with different optimizers may yield better results.
- **Evaluation Metrics**: Integration of additional evaluation metrics such as ROUGE, BLEU, and METEOR can provide a more comprehensive understanding of model performance.

References:

[1].Neural Headline Generation: A Comprehensive Survey

Published: 2025

This survey provides an extensive overview of neural headline generation techniques, emphasizing the role of encoder-decoder architectures and attention mechanisms in advancing the field.

[2].Fact-Preserved Personalized News Headline Generation

Published: 2025

The paper proposes a model that combines encoder-decoder structures with attention mechanisms to generate personalized news headlines while preserving factual information.

[3].News Headline Generation Based on Improved Decoder from Transformer-Decoder

Published: 2022

This research introduces enhancements to the decoder component of transformer-based models, incorporating attention mechanisms to improve the generation of news headlines.