MIT | Academy of Engineering
(An Autonomous Institute Affiliated to Savitribai Phule Pune University)

A Comparative Study of Encoder-Decoder Architectures for Paraphrase Generation

Abstract

- This study explores and compares three encoder-decoder architectures—without attention, with Bahdanau/Luong attention, and Transformer-based models—for the task of paraphrase generation.
- Each model is trained on a curated sentence pair dataset to evaluate its effectiveness in generating syntactically and semantically diverse paraphrases.
- The research focuses on comparative analysis using performance metrics such as BLEU, ROUGE, and METEOR, along with qualitative assessment of output fluency and coherence.

Methodology

- **Dataset Preparation**

A parallel corpus of sentence pairs was preprocessed by tokenization, lowercasing, and padding to ensure uniform input length for all models.

- **Model Architectures**

1. **Without Attention:** Basic encoder-decoder using LSTM/GRU layers.
2. **With Attention:** Enhanced with Bahdanau and Luong attention mechanisms to focus on relevant input parts during decoding.
3. **Transformer:** Utilized multi-head self-attention and positional encoding for parallelized sequence learning.

- **Training and Optimization**

All models were trained using categorical cross-entropy loss with Adam optimizer. Early stopping and validation-based checkpointing were applied.

- **Inference Strategy**

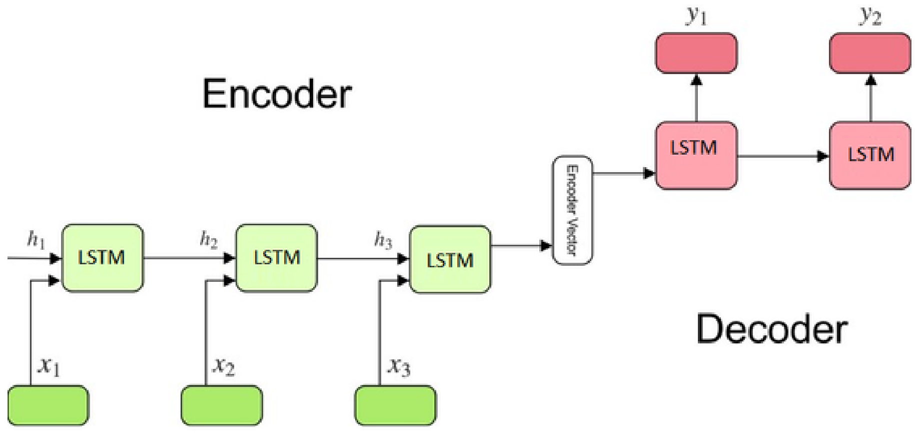
Greedy decoding was used to generate paraphrases from test sentences, with start and end tokens guiding the sequence.

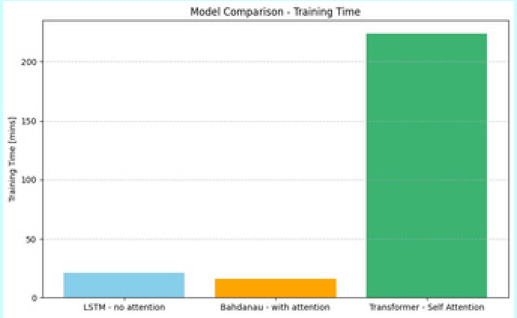
- **Evaluation Metrics**

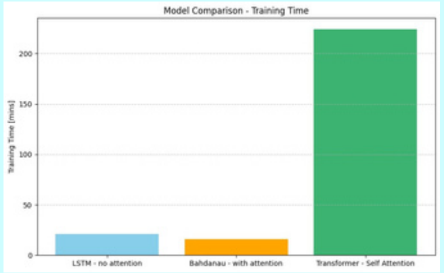
BLEU, ROUGE, METEOR, and qualitative assessment of syntactic variation and semantic preservation were used for performance comparison.

Without Attention (Basic Encoder-Decoder)

- The model encodes the entire input sentence into a single fixed-length vector, regardless of its length. This often leads to information loss, especially in longer or complex sentences, resulting in incomplete or generic paraphrases.
- The decoder generates outputs based only on the compressed context vector, which can cause paraphrases to be grammatically correct but semantically shallow or too similar to the input, lacking rephrasing diversity.
- While this architecture is computationally efficient and easy to train, its simplistic design does not handle varied sentence structures or nuanced meanings as effectively as more advanced models.







Key Takeways

LSTM trains the fastest due to its simpler architecture:

- With fewer parameters and no attention mechanism, LSTM models typically require less computational time per epoch.

Bahdanau attention increases training time moderately:

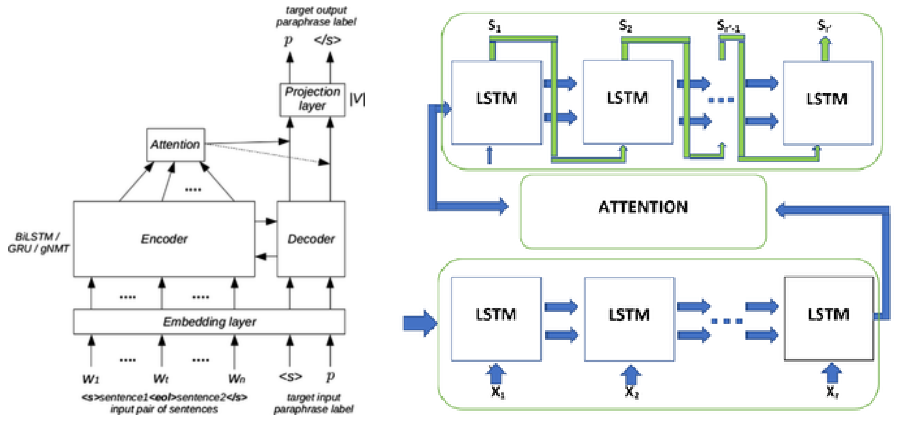
- Adding the attention mechanism leads to longer training times than plain LSTM, due to the need to compute attention weights at each decoding step.

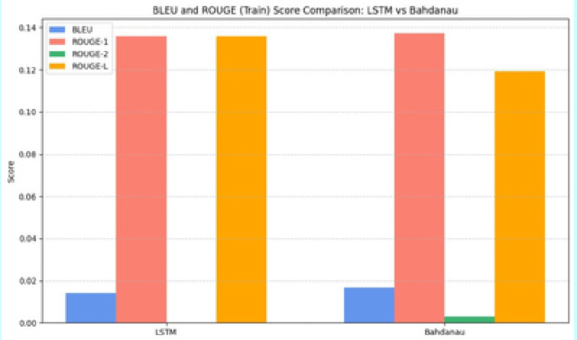
Transformers (if included) train slower per step but scale better:

- Although Transformers may have higher per-step computational cost, they parallelize well, making them faster on large datasets or modern GPUs compared to RNN-based models.

With Attention (Bahdanau/Luong)

- Attention mechanisms allow the decoder to selectively focus on specific parts of the input sequence during generation. This helps the model retain and prioritize relevant words or phrases, significantly improving semantic accuracy.
- Unlike the fixed context vector in basic models, attention enables the decoder to access the full encoder output at each time step. This leads to better paraphrasing of lengthy or complex sentences.
- While attention increases model complexity and training time, it brings substantial gains in paraphrase quality, generating more coherent, fluent, and meaning-preserving outputs compared to the baseline model.





Key Takeways

- **BLEU Score Improvement with Bahdanau:**

The BLEU score is slightly higher for the Bahdanau model than the plain LSTM, indicating better n-gram overlap and overall improved fluency or accuracy in generated sequences.

- **ROUGE-1 is consistent across both models:**

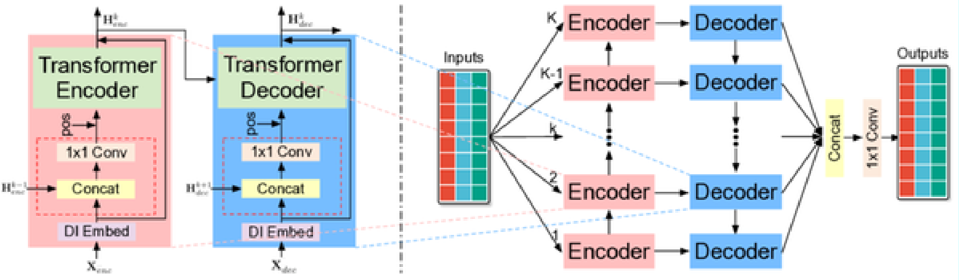
Both models achieve nearly identical ROUGE-1 scores, suggesting they capture similar levels of unigram recall, i.e., basic content word overlap with reference sequences.

- **Bahdanau shows a gain in ROUGE-2, but a slight drop in ROUGE-L:**

The Bahdanau model improves ROUGE-2, indicating better bigram-level coherence, but it shows a slightly lower ROUGE-L than LSTM, which may reflect less optimal longest subsequence overlap (i.e., slightly worse structure or phrasing).

Transformer (Self-Attention)

- Transformers use self-attention to model dependencies between all words in a sequence simultaneously. This allows the model to capture both short- and long-range contextual relationships with greater precision.
- The transformer's parallel architecture generates paraphrases that are more fluent, diverse in structure, and semantically rich, often rephrasing input in novel but accurate ways.
- Although more memory-intensive, transformers enable faster training through parallelization and are well-suited for large datasets. Their scalability makes them ideal for production-grade paraphrase generation tasks.



Conclusion

The comparative study of the three encoder-decoder models—without attention, with attention, and Transformer—highlights the evolving effectiveness of neural architectures in paraphrase generation. The basic model performs adequately but lacks contextual sensitivity. Adding attention significantly improves sequence alignment and meaning retention, while the Transformer surpasses both in fluency and accuracy due to its parallelized self-attention mechanism. Overall, as architectural complexity increases, so does the model's ability to generate more coherent and contextually accurate paraphrases.

Group Members

1. Kaustubh Wagh	202201070021
2. Jayesh Deshmukh	202201040203
3. Alvin Abraham	202201070132

Guided By Dr. Sunita Barve