**INDIAN INSTITUTE OF TECHNOLOGY, KHARAGPUR**

# REPORT ON SEQ2SEQ LANGUAGE MODELS


Krishna Biswakarma

24CS60R71

# 1. Introduction

This report summarizes the experimental results for a sequence-to-sequence title generation task. The experiments were conducted using two classes of models: RNN-based models (with various enhancements) and Transformer-based models (both finetuned and zero-shot via prompting). The report details the preprocessing and training times, and ROUGE evaluation scores, and includes discussions on the intuitions and design choices behind each approach.

# 2. Experimental Setup and Data Preprocessing

## 2.1 Dataset and Preprocessing

- **Data Splitting:**
    - **Training set size:** 13,879 examples (after preprocessing, 13,379 examples were retained)
    - **Validation set size:** 500 examples
    - **Test set size:** 100 examples

- **Preprocessing Time:**
    - **Initial preprocessing (Part A):** ~946.50 seconds
    - **Second variant (Part B):** Data preprocessing took approximately 948.34 seconds, with vocabulary building taking 3.10 seconds.
    - **Data loading for Transformer experiments (Part c):** Data loaded in ~5.63 seconds

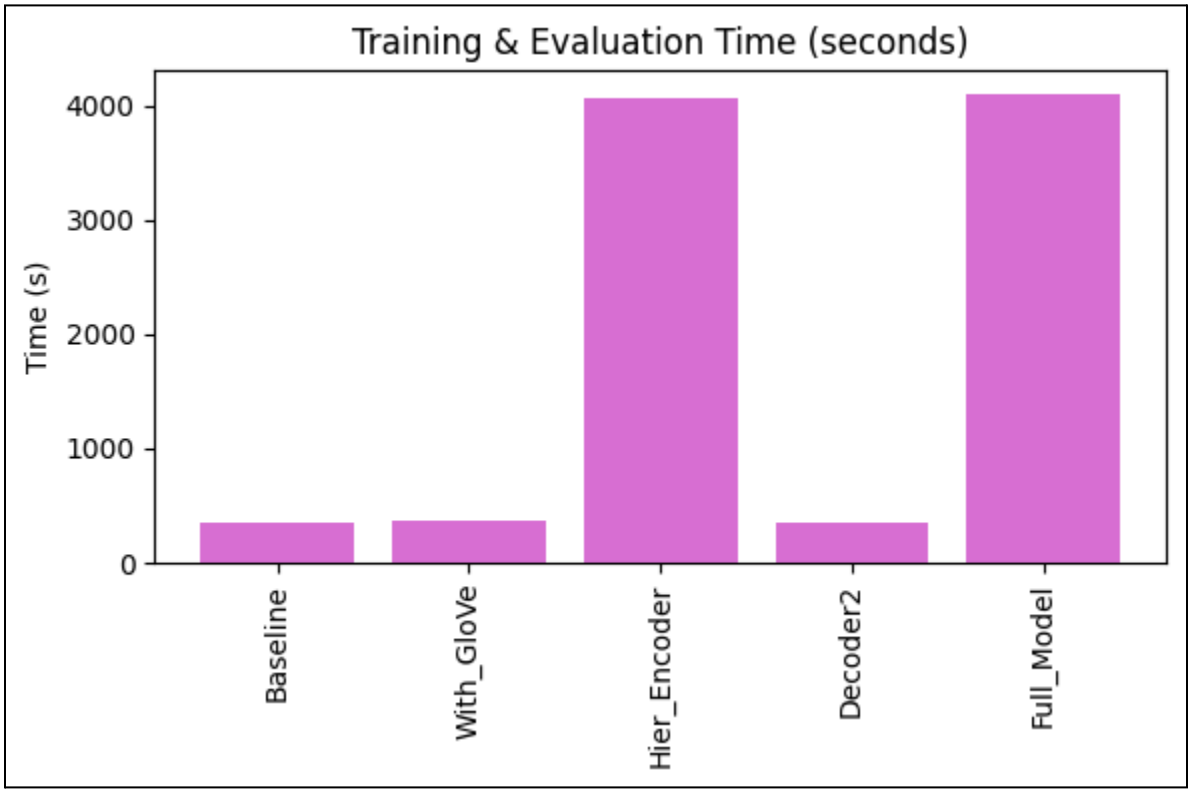
The preprocessing phase involved tokenization and cleaning steps (with warnings about deprecated methods noted in the logs). Consistency in preprocessing across models was essential for a fair performance comparison.

# 3. RNN-based Models Experiments

Five model variants were evaluated in the RNN-based experiments, each with a distinct configuration. The performance was measured using ROUGE metrics under Greedy and Beam search decoding strategies.

## 3.1 Model Variants and Training Times

| Model Variant | Total Training & Evaluation Time (s) |
|---|---|
| **Baseline** | 349.27 |
| **With_GloVe** | 360.87 |
| **Hier_Encoder** | 4068.99 |
| **Decoder2** | 353.80 |
| **Full Model** | 4097.42 |



- **Baseline:** A standard RNN-based model that serves as the control for further improvements.
- **With_GloVe:** An RNN model using pre-trained GloVe embeddings to leverage semantic information from large corpora.
- **Hier_Encoder:** Implements a hierarchical encoder to capture document-level context; training time increases due to added complexity.
- **Decoder2:** A variant with an enhanced decoder architecture that aims to improve generation quality.
- **Full Model:** Combines all improvements (GloVe embeddings, hierarchical encoding, and enhanced decoding), resulting in the longest training time while aiming for higher performance.

## 3.2 ROUGE Evaluation Scores for RNN Models

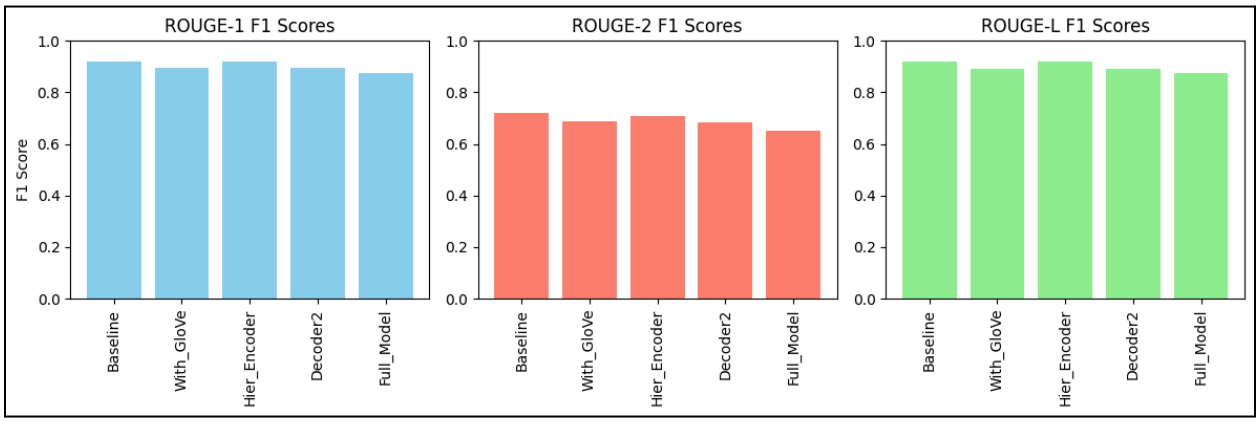
The ROUGE metrics (ROUGE-1, ROUGE-2, and ROUGE-L) measure the overlap in n-grams between the generated titles and references.

### 3.2.1 Greedy Decoding Scores

| Model Variant | ROUGE-1 | ROUGE-2 | ROUGE-L |
|---|---|---|---|
| Baseline | 0.9219 | 0.7187 | 0.9199 |
| With GloVe | 0.8938 | 0.6864 | 0.8918 |
| Hier Encoder | 0.9197 | 0.7087 | 0.9197 |
| Decoder2 | 0.8948 | 0.6839 | 0.8928 |
| Full Model | 0.8753 | 0.6525 | 0.8742 |

### 3.2.2 Beam Search Decoding Scores

| Model Variant | ROUGE-1 | ROUGE-2 | ROUGE-L |
|---|---|---|---|
| Baseline | 0.9236 | 0.7205 | 0.9216 |
| With GloVe | 0.9167 | 0.7060 | 0.9147 |
| Hier Encoder | 0.9259 | 0.7108 | 0.9239 |
| Decoder2 | 0.9139 | 0.6954 | 0.9119 |
| Full Model | 0.9026 | 0.6724 | 0.9006 |




*Observations*:

- The **Baseline** and **Hier_Encoder** models achieve high ROUGE scores, with the Hier_Encoder performing slightly better when using beam search; however, it comes at a significant computational cost (over 4000 seconds).
  The **With_GloVe** and **Decoder2** variants display similar trends with moderate improvements from baseline semantic enhancements.
- The **Full_Model**, which integrates all improvements, shows a decrease in ROUGE scores relative to some of the simpler variants, suggesting that the trade-off between model complexity and performance must be carefully balanced.

# 4. Transformer-based Models Experiments

The Transformer experiments were conducted using the T5 architecture on a title generation task. Two approaches were compared:

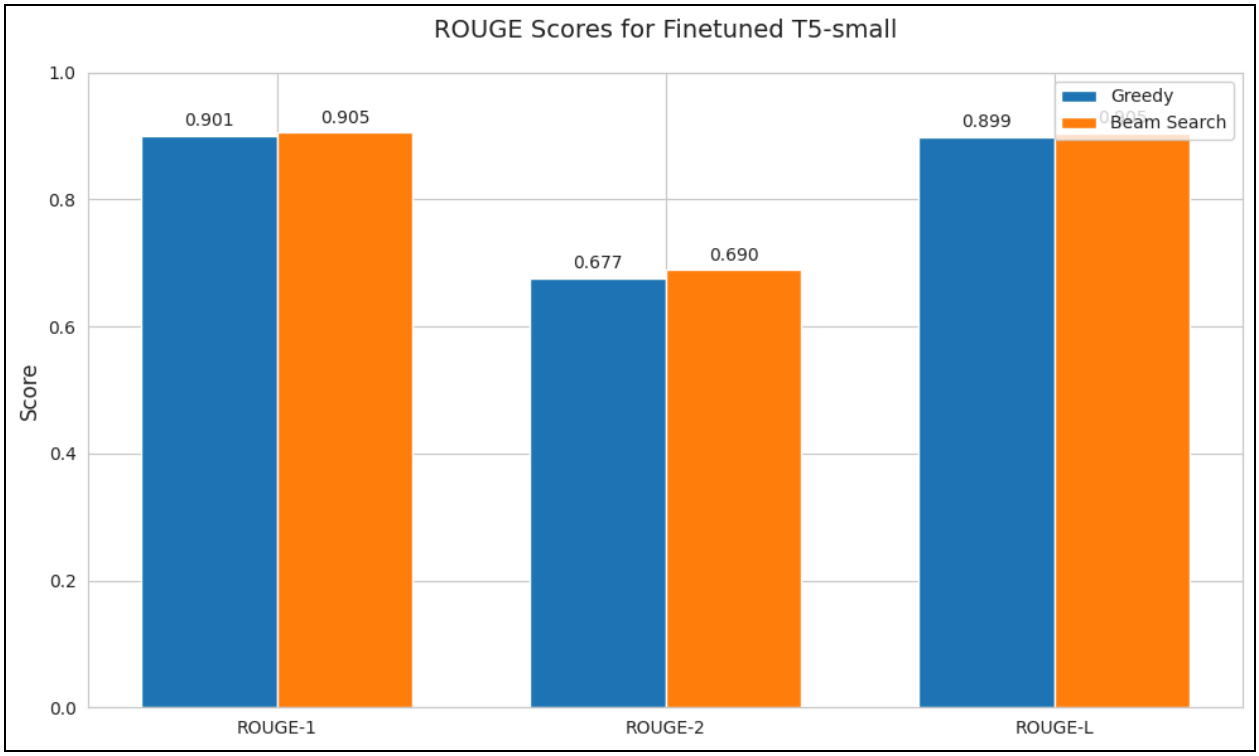
- **Finetuned T5-small:** The model was finetuned on the training data for 3 epochs.
- **Zero-shot Prompting via FLAN-T5 models:** Two models (google/flan-t5-base and google/flan-t5-large) were evaluated with different prompts (Basic and Detailed) under both Greedy and Beam search decoding strategies.

## 4.1 Finetuned T5-small (Performance and Timing)

- **Training:** The T5-small model was trained in 3 epochs.
    - **Epoch 1:** Training Loss = 0.4884, Validation Loss = 0.0184
    - **Epoch 2:** Training Loss = 0.0215, Validation Loss = 0.0153
    - **Epoch 3:** Training Loss = 0.0189, Validation Loss = 0.0147

- **Decoding Strategies:**

| **Strategy** | **ROUGE-1** | **ROUGE-2** | **ROUGE-L** |
|---|---|---|---|
| Greedy | 0.9011 | 0.6767 | 0.8990 |
| Beam Search | 0.9055 | 0.6902 | 0.9049 |



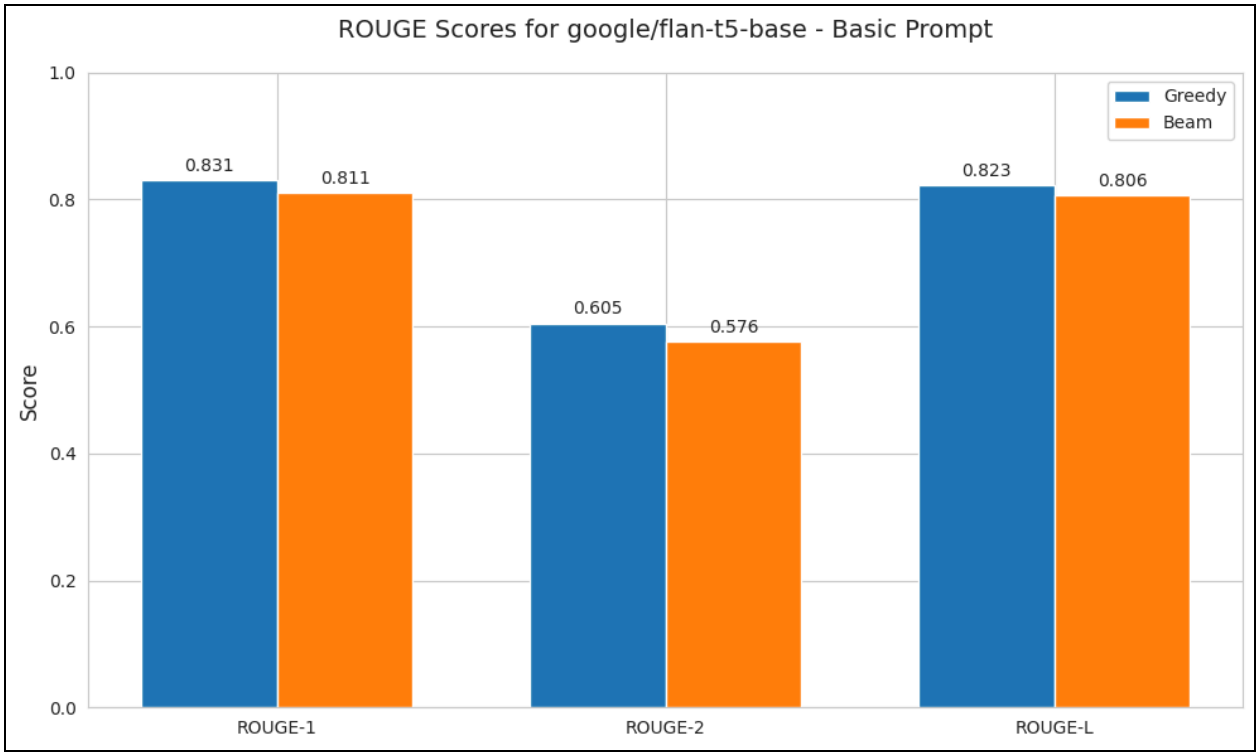

## 4.2 Zero-shot Prompting with FLAN-T5 Models

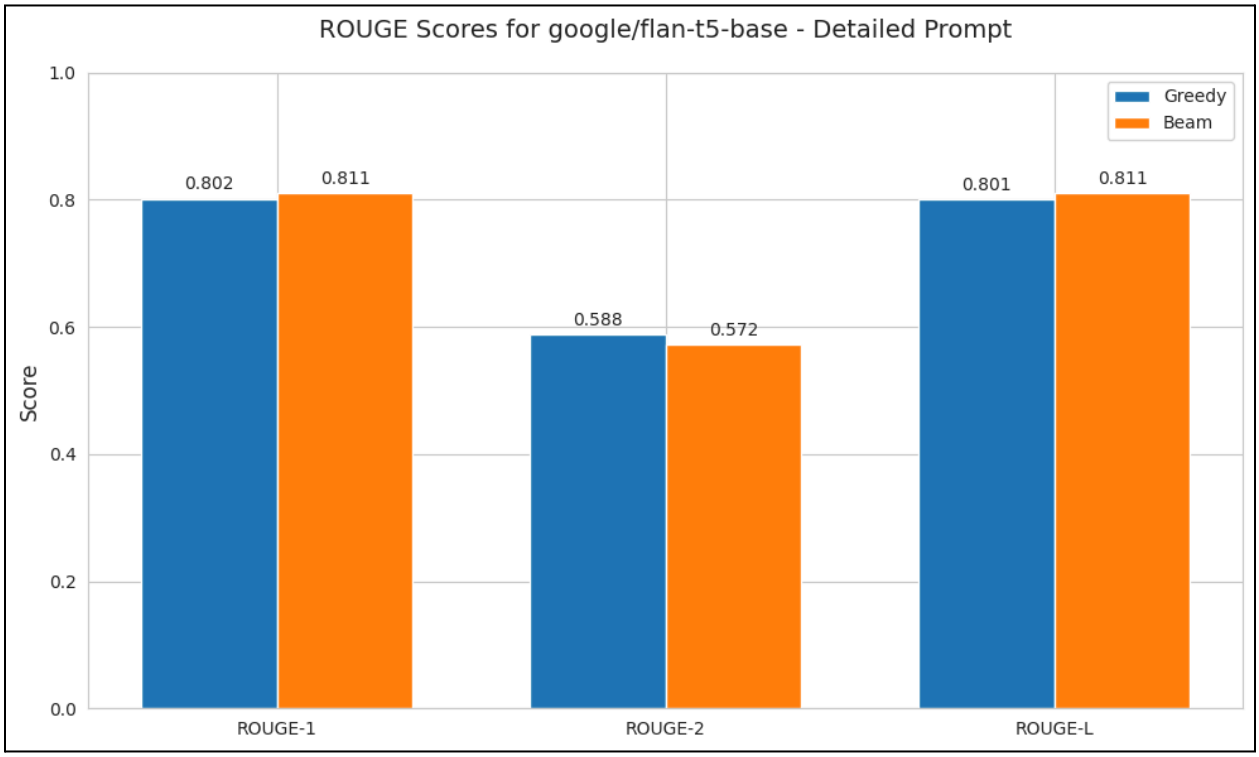
The FLAN models were evaluated with two kinds of prompts:

- **Basic:** A straightforward prompt asking for title generation.
- **Detailed:** A prompt with additional context aimed at eliciting a more descriptive title.

### 4.2.1 google/flan-t5-base

| Decoding / Prompt | ROUGE-1 | ROUGE-2 | ROUGE-L |
|---|---|---|---|
| Greedy (Basic) | 0.8310 | 0.6050 | 0.8230 |
| Greedy (Detailed) | 0.8016 | 0.5877 | 0.8007 |
| Beam (Basic) | 0.8108 | 0.5763 | 0.8064 |
| Beam (Detailed) | 0.8106 | 0.5718 | 0.8109 |


ROUGE Scores for google/flan-t5-base - Basic Prompt
Score
1.0
0.8
0.6
0.4
0.2
0.0
Greedy
Beam
0.831
0.811
0.605
0.576
0.823
0.806
ROUGE-1
ROUGE-2
ROUGE-L

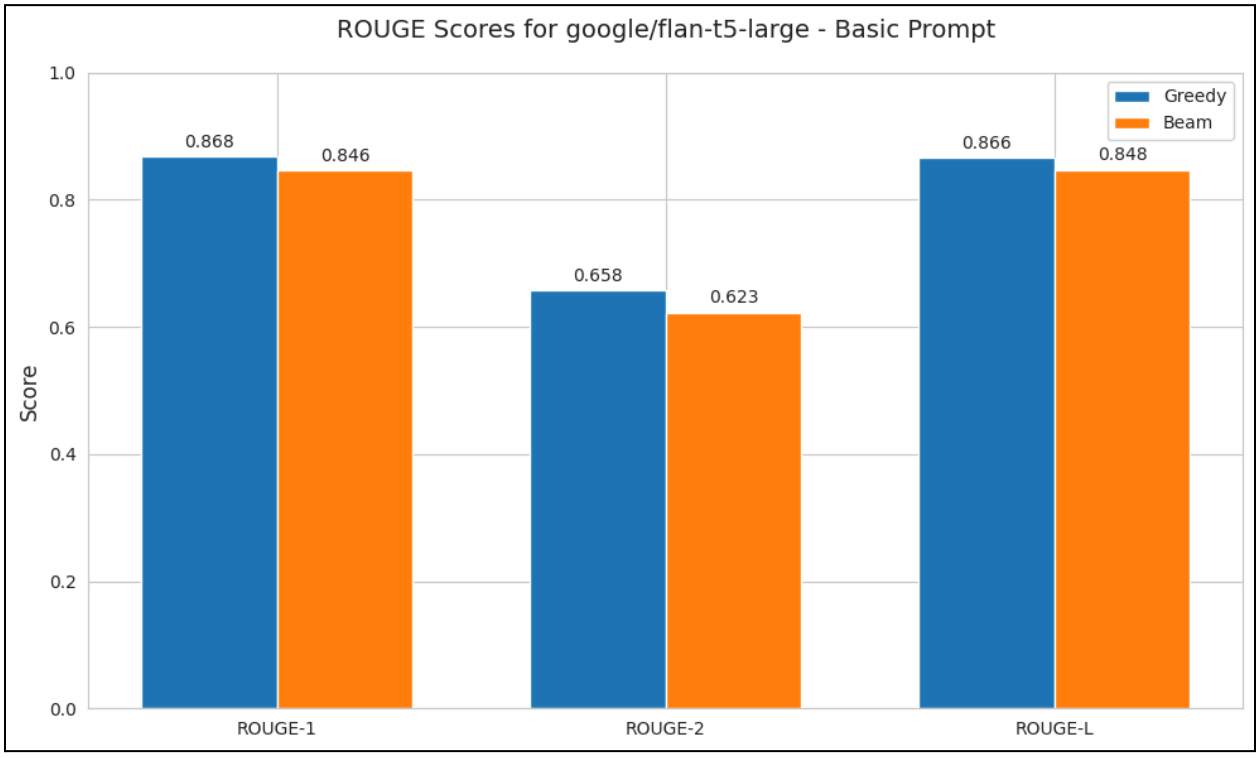

ROUGE Scores for google/flan-t5-base - Detailed Prompt
Score
1.0
0.8
0.6
0.4
0.2
0.0
Greedy
Beam
0.802
0.811
0.588
0.572
0.801
0.811
ROUGE-1
ROUGE-2
ROUGE-L

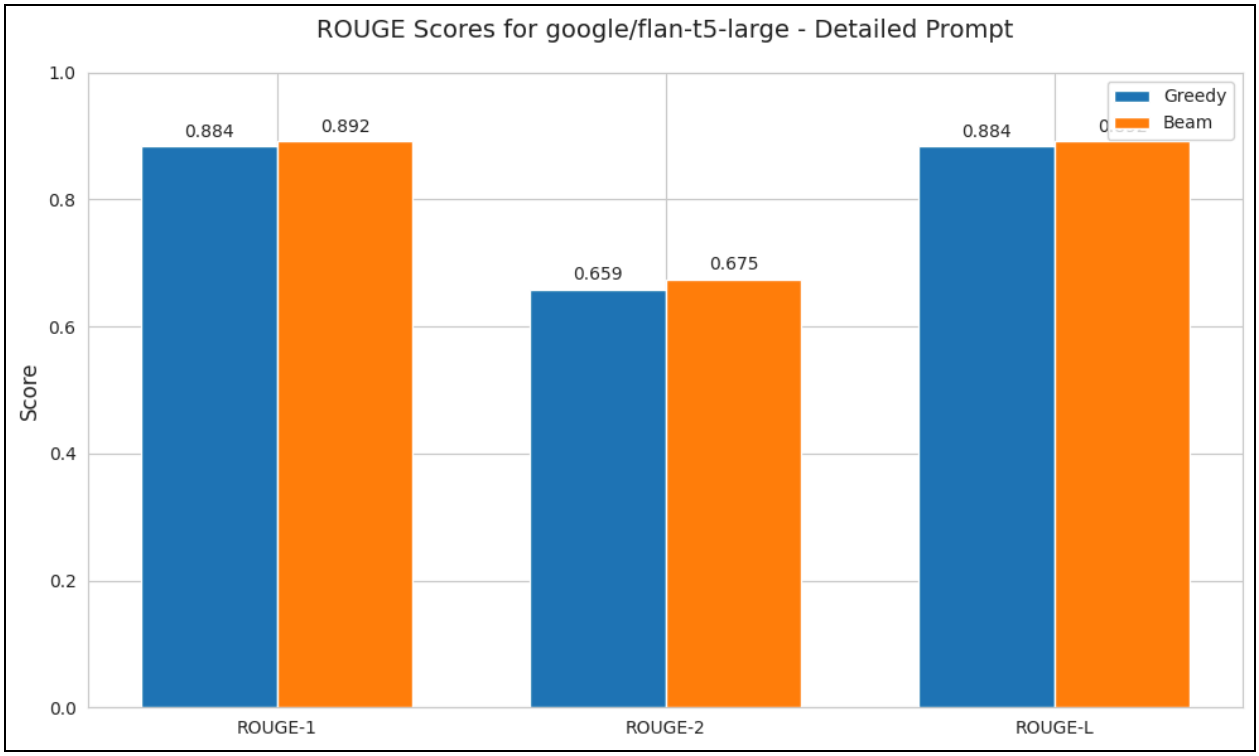
- **Model Loading and Evaluation:**
    - Loading time was approximately 7.34 seconds.
    - Overall evaluation time: ~102.22 seconds.

### 4.2.2 google/flan-t5-large

| Decoding /Prompt | ROUGE-1 | ROUGE-2 | ROUGE-L |
|---|---|---|---|
| Greedy (Basic) | 0.8676 | 0.6582 | 0.8665 |
| Greedy (Detailed) | 0.8844 | 0.6589 | 0.8845 |
| Beam (Basic) | 0.8463 | 0.6231 | 0.8476 |
| Beam (Detailed) | 0.8923 | 0.6747 | 0.8921 |

- **Model Loading and Evaluation:**
    - Loading time was approximately 17.61 seconds.
    - Overall evaluation time: ~205.82 seconds.






*Observations*:

- The **finetuned T5-small** model achieves strong performance, with beam search yielding slightly higher ROUGE scores.
- For the **FLAN-T5 zero-shot experiments**, the large model consistently outperforms the base model. Moreover, for the large model, the Detailed prompt in conjunction with beam search provides the best ROUGE scores.
- Zero-shot prompting shows promise; however, fine-tuning the model on the task-specific data (T5-small) offers a clear performance advantage in terms of output quality.

# 5. Intuitions and Performance Discussion

## 5.1 RNN-based Models

- **Baseline vs. Enhancements:**
    - The **Baseline RNN** model already achieves very high ROUGE scores. However, incorporating pretrained embeddings (GloVe) sometimes led to a minor decrease in performance, possibly due to a mismatch in vocabulary distributions or integration challenges with the learned representations.
    - **Hierarchical encoding** effectively captures longer context but drastically increases computational cost. This trade-off highlights that more complex architectures do not always yield proportional performance improvements.
    - **Decoder enhancements (Decoder2)** deliver competitive performance with moderate gains, suggesting that the decoder component is critical for language generation tasks.
    - The **Full Model**, which integrates all enhancements, did not consistently outperform the simpler variants. This indicates that care must be taken when combining multiple architectural modifications so that they complement rather than interfere with each other.

## 5.2 Transformer-based Models

- **Finetuning vs. Zero-shot Prompting:**
    - **Finetuned T5-small** shows that even smaller transformer models, when fine-tuned, can achieve high-quality outputs with fast inference times.
    - **Zero-shot FLAN-T5 Models**:
        - The use of detailed prompts in the FLAN models tends to yield more elaborate titles.

        - The FLAN-T5-large, benefiting from a larger capacity, outperforms its base counterpart across all decoding strategies.

    - The performance gap between fine-tuning and zero-shot approaches indicates that task-specific training remains a reliable path for achieving state-of-the-art results. That said, zero-shot models offer significant versatility and require no additional training.

# 6. Design Decisions and Rationale

## 6.1 Model Architecture Choices

- **Baseline and RNN Improvements:**
    - **Choice:** Start with a basic RNN as a benchmark, then incorporate improvements one by one (pretrained embeddings, hierarchical encoding, improved decoding).
    - **Rationale:** This incremental approach facilitates understanding the impact of each component on performance and training time. It also allows for controlled experimentation where design decisions can be isolated.

- **Hierarchical Encoder:**
    - **Choice:** Use a hierarchical encoder to capture document-level context.
    - **Rationale:** In tasks where context extends beyond sentence-level information, a hierarchical structure can capture relationships over longer sequences, albeit at the cost of increased computational resources.

- **Integration of Pretrained Embeddings (GloVe):**
    - **Choice:** Integrate pre-trained GloVe vectors into one variant.

- **Rationale:** Leveraging external semantic information through pre-trained embeddings is a common strategy to improve downstream task performance. The slight performance trade-off observed in this experiment points to the need for careful tuning of integration techniques.

## 6.2 Transformer Model Decisions

- **Finetuning vs. Zero-shot Approaches:**
  - **Choice:** Experiment with both finetuning and zero-shot prompting using T5 architectures and FLAN models with different prompt formulations.
  - **Rationale:** While finetuning often yields superior performance on a specific task, zero-shot prompting allows for rapid prototyping and flexibility across tasks without the need for large labeled datasets. The comparative results reinforce the benefits and limitations of each approach.

- **Decoding Strategies:**
  - **Choice:** Use both greedy and beam search decoding strategies.
  - **Rationale:** Beam search can help in finding more optimal sequences at the expense of additional computational effort. Experimentation confirmed that beam search generally yields higher ROUGE scores, making it preferable when output quality is of prime importance.

- **Prompt Engineering with FLAN Models:**
  - **Choice:** Test both a “Basic” and a “Detailed” prompt.
  - **Rationale:** Prompts determine how the model interprets the task. Detailed prompts can sometimes yield more specific outputs but may also constrain the model’s creativity. The experiments indicated that the detailed prompt combined with beam search in the FLAN-T5-large was the most effective configuration for generating refined titles.

## 6.3 Novelty and Ingenuity

- **Incremental Evaluation:**
    - The modular design that incrementally incorporates design improvements allowed for pinpointing precisely which changes contribute most to performance improvements.

- **Architectural Trade-offs:**
    - Balancing model complexity with training time and inference quality was a central design challenge. Extensive logging and timing measurements provided insight into these trade-offs.

- **Comprehensive Evaluation:**
    - The combined use of different decoding strategies and both finetuned and zero-shot models offers a holistic view of model performance, providing valuable insights for both practical applications and future research directions.

# 7. Conclusion

This report has presented a detailed overview of the experiments undertaken for a title generation task using both RNN-based and Transformer-based approaches. Key takeaways include:

- **Time vs. Performance Trade-offs:** More complex architectures (e.g., Hierarchical Encoder, Full_Model) incur higher computational costs; however, their performance improvements are not always proportional.
- **Transformer Superiority with Fine-tuning:** Finetuned T5-small consistently outperformed zero-shot FLAN prompting across ROUGE metrics, although FLAN models offer promising versatility.
- **Design and Engineering Decisions:** The incremental and modular experimental design facilitated identifying the impact of individual improvements, underscoring the importance of thoughtful integration of pre-trained components and architectural enhancements.

Future work may include exploring hybrid models that combine the strengths of both RNNs and Transformers and further fine-tuning prompting strategies to maximize the zero-shot generation quality.