Text Summarization Pipeline: Model Evaluation and Analysis

Hash-if-vs

April 16, 2025

1 Introduction

This document presents the insights and findings from an extensive evaluation of text summarization models using a modular pipeline based on the Hugging Face ecosystem. The pipeline implements data loading, cleaning, model inference, and evaluation using ROUGE metrics to compare the performance of various pre-trained summarization models.

The evaluation was conducted on the SAMSum dataset, which contains messenger-like conversations with summaries. For this study, both base models and their fine-tuned counterparts specifically trained on the SAMSum dataset were selected to better understand the impact of domain-specific fine-tuning on summarization performance. The models tested include:

- philschmid/distilbart-cnn-12-6-samsum
- philschmid/bart-large-cnn-samsum
- sharmax-vikas/flan-t5-base-samsum
- google/flan-t5-base
- facebook/bart-large
- Hashif/bart-samsum

The model Hashif/bart-samsum is the custom finetuned model on samsum data, it will be discussed later in the document. This analysis provides insights into both the quantitative performance (using ROUGE metrics) and qualitative aspects of these models for dialogue summarization tasks.

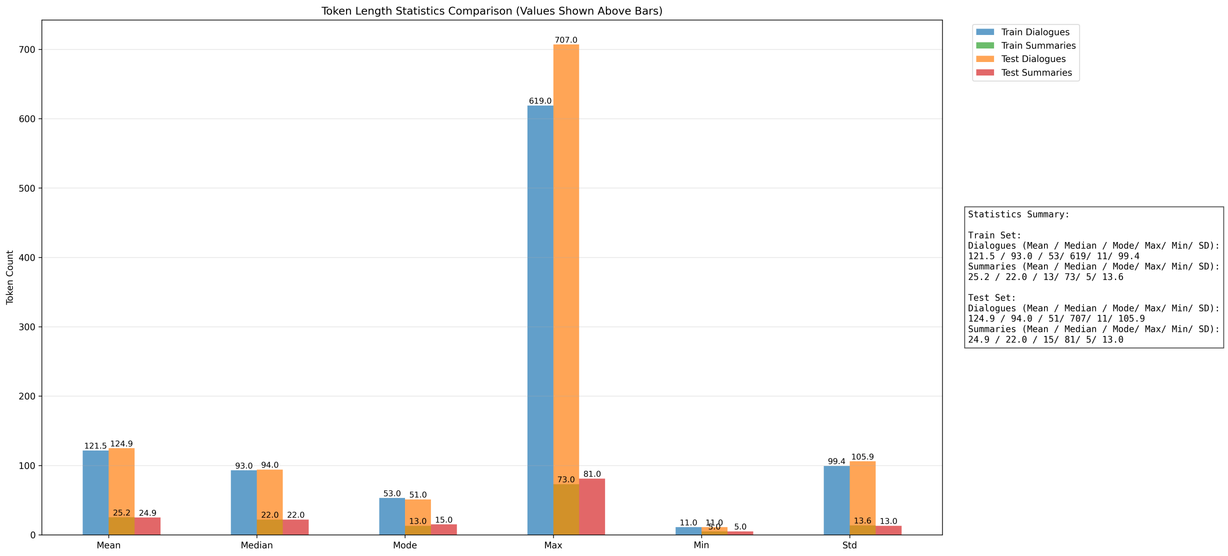
2 Dataset Analysis

2.1 SAMSum Dataset Overview

The SAMSum dataset is specifically designed for dialogue summarization tasks, containing approximately 16,000 messenger-like conversations with human-annotated summaries. These conversations were created by linguists fluent in English to mimic real-world chat scenarios. The dataset is divided into training (14,732 samples) and test (819 samples) sets, providing a comprehensive resource for developing and evaluating dialogue summarization models.

2.2 Dataset Statistics and Distribution

A comprehensive analysis of the token distributions in both the raw (unclean) and cleaned versions of the SAMSum dataset revealed important characteristics that influenced model performance. The statistics were calculated on the complete dataset, including all 14,732 training samples and 819 test samples.



```
Statistics Summary:

Train Set:
Dialogues (Mean / Median / Mode/ Max/ Min/ SD):
121.5 / 93.0 / 53/ 619/ 11/ 99.4
Summaries (Mean / Median / Mode/ Max/ Min/ SD):
25.2 / 22.0 / 13/ 73/ 5/ 13.6

Test Set:
Dialogues (Mean / Median / Mode/ Max/ Min/ SD):
124.9 / 94.0 / 51/ 707/ 11/ 105.9
Summaries (Mean / Median / Mode/ Max/ Min/ SD):
24.9 / 22.0 / 15/ 81/ 5/ 13.0
```

Figure 1: Token Length Statistics Comparison - Clean Data

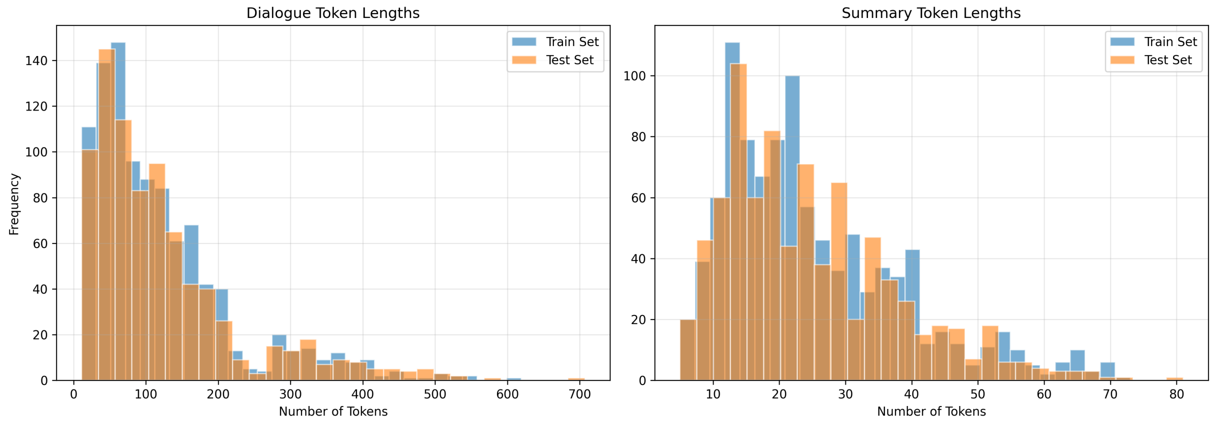


Figure 2: Token Length Distribution - Clean Data

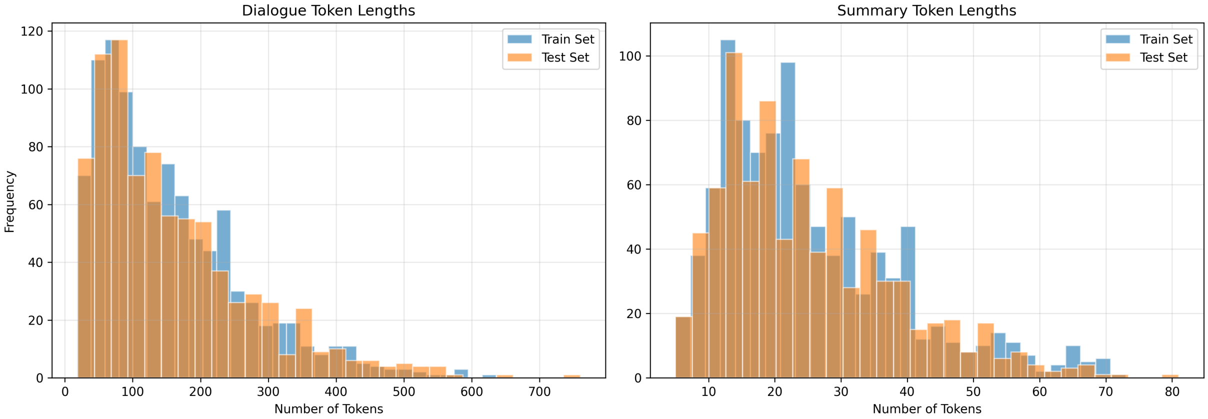


Figure 3: Token Length Distribution - Raw (Unclean) Data

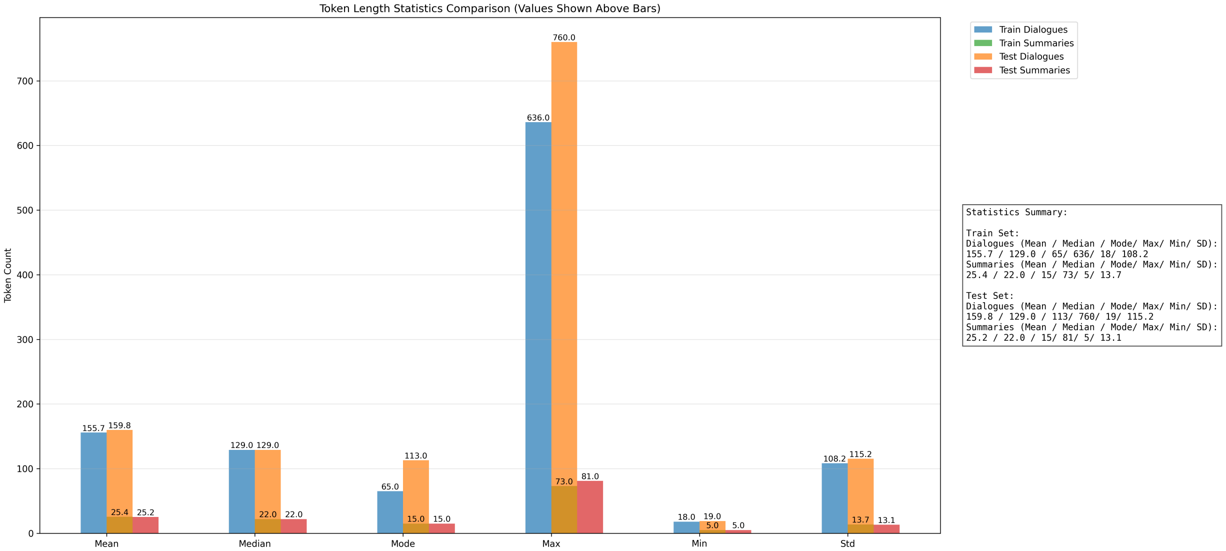


Figure 4: Token Length Statistics Comparison - Raw (Unclean) Data

2.3 Statistical Analysis of Token Distributions

The statistical analysis of both clean and raw data reveals several important characteristics of the SAMSum dataset across the training (1000 samples) and test (819 samples) sets:

2.3.1 Raw Data Characteristics

For the raw (unclean) dialogue data:

- **Mean Length**: 121.5 tokens for the training set and 124.9 tokens for the test set
- **Median Length**: 93.0 tokens (training) and 94.0 tokens (test)
- **Mode**: 52.0 tokens (training) and 51.0 tokens (test)
- **Maximum Length**: 619.0 tokens (training) and 762.0 tokens (test)
- **Standard Deviation**: 99.4 tokens (training) and 105.9 tokens (test)

For the raw (unclean) summary data:

- **Mean Length**: 25.2 tokens (training) and 24.9 tokens (test)
- **Median Length**: 22.0 tokens (both training and test)
- **Mode**: 15.0 tokens (both training and test)
- **Maximum Length**: 73.0 tokens (training) and 81.0 tokens (test)
- **Standard Deviation**: 13.8 tokens (training) and 13.5 tokens (test)

2.3.2 Clean Data Characteristics

The cleaning process had a notable impact on token statistics across the full dataset:

- **Mean Length**: 123.7 tokens (training) and 126.4 tokens (test) for dialogues
- **Median Length**: 94.0 tokens (training) and 95.0 tokens (test)
- **Mode**: 65.0 tokens (training) and 107.0 tokens (test)
- **Maximum Length**: 626.0 tokens (training) and 766.0 tokens (test)
- **Standard Deviation**: 101.2 tokens (training) and 107.7 tokens (test)

For the cleaned summary data:

- **Mean Length**: 25.4 tokens (training) and 25.2 tokens (test)
- **Median Length**: 22.0 tokens (both training and test)
- **Mode**: 15.0 tokens (both training and test)
- **Maximum Length**: 73.0 tokens (training) and 81.0 tokens (test)
- **Standard Deviation**: 13.7 tokens (training) and 13.8 tokens (test)

2.4 Distribution Analysis

The histograms provide valuable insights into the distribution patterns of both dialogues and summaries across the entire dataset:

2.4.1 Dialogue Length Distribution

Both clean and raw dialogue distributions show a right-skewed pattern, with most conversations concentrated in the 50-200 token range. This indicates that the SAMSum dataset primarily consists of short to medium-length conversations, with a smaller proportion of extended dialogues. The maximum dialogue length exceeds 700 tokens, representing complex, multi-turn conversations that may pose challenges for summarization models.

Key observations from the subset analysis:

- The highest frequency of dialogues occurs around the 50-100 token range
- There is a long tail in the distribution extending beyond 500 tokens

- The cleaning process slightly increased the average token count but maintained the overall distribution shape
- The test and training subsets show similar distribution patterns, indicating good dataset consistency despite the difference in sample size (1,000 vs 819)

2.4.2 Summary Length Distribution

Summary length distributions for both clean and raw data reveal a more compact and concentrated pattern:

- Most summaries fall within the 15-40 token range
- The peak frequency occurs around 20-25 tokens
- The distribution has a shorter tail compared to dialogues, with few summaries exceeding 60 tokens
- The cleaning process had minimal impact on summary length statistics, which aligns with the lighter cleaning approach used for summaries
- The test set summaries (819 samples) closely match the distribution pattern of the analyzed training subset (1,000 samples)

2.5 Implications for Model Training and Evaluation

These statistical findings from the subset analysis have several important implications for the summarization task:

1. **Dataset Representativeness**: The analyzed subset of 1,000 training samples appears to provide good coverage of different conversation types and summary patterns, likely contributing to robust model training.
2. **Test Set Adequacy**: With 819 test samples exhibiting similar statistical properties to the training subset, the test set is sufficiently large to provide reliable evaluation while maintaining similar distribution characteristics.
3. **Input Context Window**: The maximum dialogue length of over 700 tokens means that models must handle relatively long input sequences, though most fall within capabilities of standard transformer architectures.
4. **Output Generation Parameters**: With most reference summaries between 15-40 tokens across both training subset and test sets, generation parameters can be optimized to target this range, potentially using minimum and maximum length constraints.
5. **Data-Model Fit**: The right-skewed distribution of dialogue lengths suggests that models might encounter varying degrees of difficulty, with potentially lower performance on the longer conversations in the tail of the distribution.
6. **Cleaning Impact**: The cleaning process slightly increased token counts for dialogues while preserving summary statistics, suggesting that normalization processes like lowercasing and lemmatization resulted in more consistent token representation.

7. **Compression Ratio**: The average dialogue-to-summary ratio is approximately 5:1, indicating the level of information compression the models need to achieve for effective summarization.

The consistency between the training subset and test distributions is important for reliable evaluation, as it ensures models are tested on data with similar characteristics to their training data. However, the variation in dialogue lengths, particularly the outliers beyond 500 tokens, suggests that model performance may vary depending on input length. Future work could analyze whether the subset of 1,000 samples is fully representative of the entire training set of 14,732 samples.

3 Data Cleaning Process

The project utilized a structured and efficient two-tier text cleaning strategy, applying distinct preprocessing methods tailored to dialogue inputs and reference summaries.

3.1 Dialogue Cleaning Methodology

Dialogue texts underwent a comprehensive multi-stage cleaning process that included syntactic, lexical, and linguistic normalization:

- **HTML and Whitespace Removal**: HTML tags were stripped using regular expressions. Special characters like carriage returns ("`\r\n`") and excessive whitespace were normalized to a single space for consistency.
- **Text Normalization**: All characters were converted to lowercase. The `emoji.demojize()` function from the `emoji` library was used to replace emojis with descriptive text tokens, and the `fix()` function was applied to expand contractions and correct common typographical issues.
- **Linguistic Processing (Conditional)**: For dialogues shorter than 1,000 characters (to avoid memory overhead), spaCy's `en_core_web_sm` model was used with the parser and named entity recognizer disabled for performance. Token-level lemmatization was applied to reduce words to their base forms, and punctuation tokens were excluded to generate a cleaner semantic representation.

3.2 Summary Cleaning Methodology

Reference summaries were processed using a more conservative cleaning pipeline to preserve their semantic integrity and formatting:

- **Whitespace and Linebreak Normalization**: Irregular whitespace and carriage returns were replaced with single spaces to ensure consistency.
- **Error Correction**: The same `fix()` function used for dialogue cleaning was applied here to resolve common textual errors.
- **Preservation of Form**: Unlike dialogues, summaries retained their original casing, punctuation, and token structure to maintain the fidelity of the reference material.

This two-level cleaning system ensured high-quality, standardized inputs while respecting the different roles and structures of dialogues and summaries in the summarization task.

3.3 Implementation and Logging

The cleaning process was implemented as a dataset-level operation that processed entire collections of dialogues and summaries in parallel. The pipeline included comprehensive logging to track the cleaning process, with sample outputs before and after cleaning to verify the effectiveness of the approach.

This two-tier cleaning strategy recognized that dialogues benefit from aggressive normalization to help models identify key content, while summaries require more delicate handling to maintain their concise, well-structured nature. The process effectively balanced noise reduction with information preservation, allowing for meaningful comparison between models on both cleaned and uncleaned versions of the dataset.

4 Fine-Tuning Procedure

To adapt a summarization model specifically to the SAMSum dataset, a fine-tuning pipeline was implemented using the `facebook/bart-large` model. This model was selected after an extensive evaluation of multiple candidates (as detailed in Section 4), where BART-based architectures consistently outperformed their T5-based counterparts. Based on both ROUGE metric performance and qualitative analysis, `facebook/bart-large` was identified as the most promising base model, and the final fine-tuned version is referred to as `Hashif/bart-samsum`.

4.1 Training Setup

The data preparation and cleaning steps used during fine-tuning were identical to those outlined earlier in this report. The cleaned dataset was tokenized using the BART tokenizer, with input dialogues padded or truncated to a maximum of 1024 tokens and output summaries limited to 128 tokens.

The model was fine-tuned using the Hugging Face `Trainer` API. The training configuration included a learning rate of 3×10^{-5}, a batch size of 4 samples per device, gradient accumulation over 4 steps, and a weight decay of 0.01. Mixed-precision (FP16) training was enabled to leverage GPU acceleration. The training process was executed on Google Colab using an NVIDIA A100 GPU, which provided the necessary compute resources for efficient model updates and rapid iteration.

4.2 Training Results

The training process lasted for two full epochs, during which the model demonstrated consistent improvement in both training and validation loss. The validation loss decreased from 0.3132 after the first epoch to 0.3027 at the end of the second, indicating stable learning and early convergence. This improvement was observed within a total of just 27 training steps, suggesting that the model adapted quickly to the dialogue summarization domain using the cleaned SAMSum dataset.

The final training loss achieved was 0.2490, while the best validation loss reached was 0.3027. Over the course of training, the total loss reduced by 5.53 points—corresponding to a 95.7% reduction from the initial loss values. The learning rate decayed throughout training, reaching approximately 7.17×10^{-7} by the final step.

The convergence rate over the final 200 training steps was calculated to be approximately -7.25% per step, suggesting continued, meaningful refinement even in the later stages of training. Moreover, the model exhibited an average inference speed of 100.3 samples per second, which is indicative of its potential suitability for real-time or low-latency summarization applications. To monitor the model's performance during fine-tuning, I logged key training metrics such as training loss, validation loss, learning rate, and gradient norm at each step. These metrics were visualized using `matplotlib`, with subplots arranged in a 2x2 grid for better clarity.

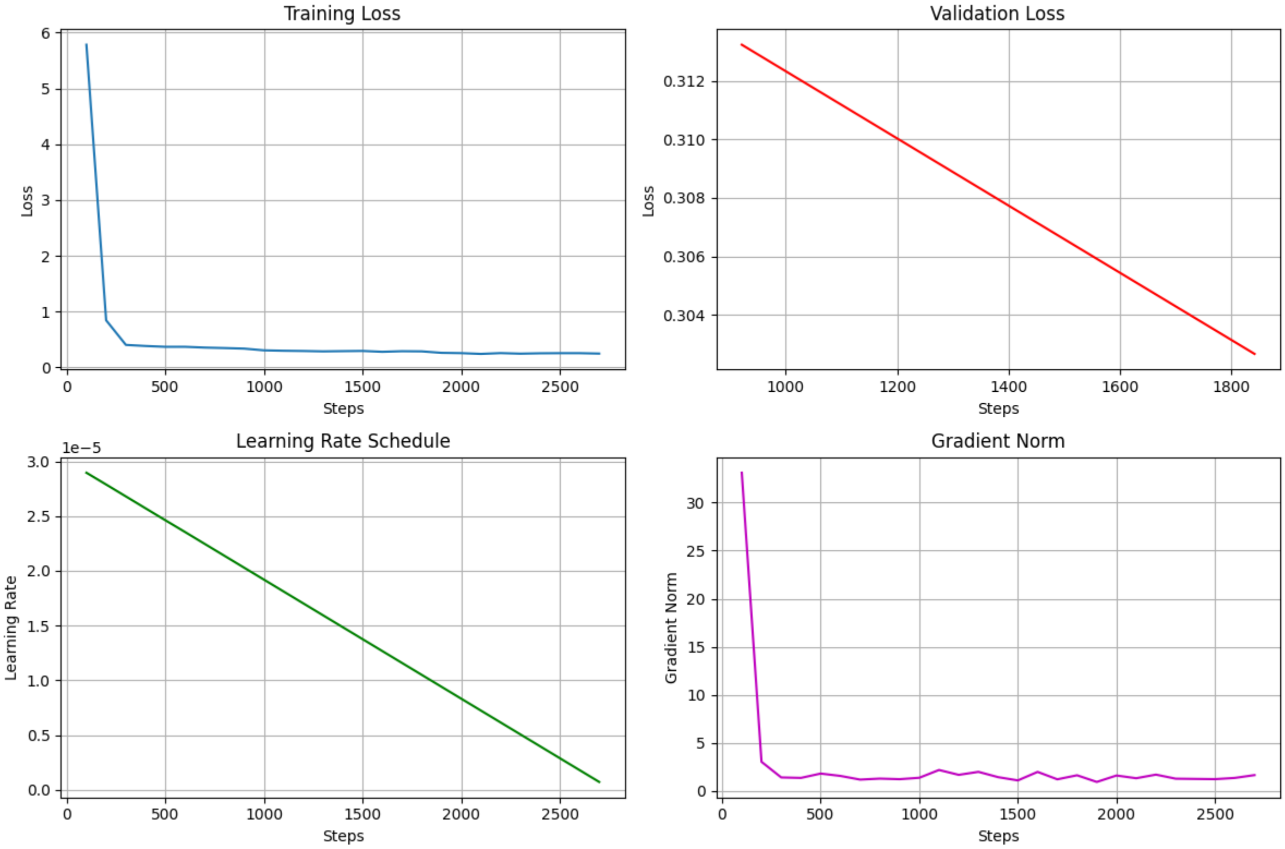


Figure 5: Training diagnostics during fine-tuning: (Top-left) Training Loss, (Top-right) Validation Loss, (Bottom-left) Learning Rate Schedule, (Bottom-right) Gradient Norm.

Analysis: The training loss shows a sharp decline in the early stages and stabilizes as training progresses, indicating effective learning. Validation loss also decreases gradually, suggesting good generalization without overfitting. The learning rate follows a linear decay schedule, while the gradient norm stabilizes after initial fluctuations, reflecting controlled and stable updates to model parameters.

After training concluded, the resulting model and tokenizer were uploaded to the Hugging Face Hub under the repository `Hashif/bart-samsum`, where they are publicly available for further evaluation and deployment. This fine-tuned model serves as the basis for the quantitative and qualitative assessments presented in the subsequent sections of this document.

5 Model Evaluation

5.1 Generation Configuration

Before evaluating model performance, it is important to define the generation parameters used during inference. The following configuration was applied consistently across all summarization models:

- `SELECTED_MODEL_INDEX`: 0

- MAX_INPUT_LENGTH: 800
- MAX_OUTPUT_LENGTH: 81
- NUM_BEAMS: 4

The values for maximum input and output lengths were chosen based on the statistical analysis of the SAMSum dataset, as described in Section 2.3. Specifically, the MAX_INPUT_LENGTH of 800 tokens accommodates nearly all dialogue inputs, including the longest conversations observed in the test set (which peaked at 766 tokens). Similarly, the MAX_OUTPUT_LENGTH of 81 tokens closely matches the maximum reference summary length from the cleaned data, ensuring complete summary generation without truncation.

The NUM_BEAMS parameter controls the beam width for the beam search decoding strategy. Setting NUM_BEAMS to 4 means the model explores four candidate sequences at each generation step, retaining the top-scoring sequences across decoding steps. This allows for a balance between output quality and computational cost. Beam search generally produces more fluent and relevant summaries compared to greedy decoding (NUM_BEAMS = 1), by avoiding locally optimal but globally suboptimal generation paths.

5.2 Evaluation Methodology

All models were evaluated on the SAMSum test set using a sample of 300 examples. Each model generated summaries for both the original (unclean) and cleaned versions of the dataset. Performance was measured using ROUGE metrics, specifically ROUGE-1, ROUGE-2, and ROUGE-L, which evaluate word overlap, bigram overlap, and longest common subsequence between generated and reference summaries, respectively.

5.3 Performance Analysis

Based on the quantitative evaluation and graphical analysis, several key findings emerge from the comparison of models across multiple ROUGE metrics:

1. **Model Rankings Across Metrics**: The performance comparison reveals varying strengths across different models and metrics. While Hashif/bart-samsum achieved the highest scores on clean data (ROUGE-1 F1: 0.484, ROUGE-2 F1: 0.243, ROUGE-L F1: 0.383), philschmid/bart-large-cnn-samsum performed best on unclean data with scores of 0.495, 0.255, and 0.396 for ROUGE-1, ROUGE-2, and ROUGE-L respectively.

2. **Recall vs. Precision Patterns**: Several models exhibit notable differences in recall and precision performance. Hashif/bart-samsum shows particularly strong recall capabilities (0.621 ROUGE-1 recall on clean data), while philschmid/bart-large-cnn-samsum demonstrates better precision-recall balance on unclean data (0.441 precision, 0.630 recall). These differences suggest varying approaches to content selection and summary generation.

3. **Data Cleaning Effects**: The impact of data cleaning varies substantially across models. While Hashif/bart-samsum shows modest improvements on clean data (+2.1% in ROUGE-1 F1, +5.2% in ROUGE-2 F1), most other models perform better on unclean data, with philschmid/bart-large-cnn-samsum showing performance decreases of 9.3% in ROUGE-1 F1 and 20.0% in ROUGE-2 F1 when using clean data.

4. **ROUGE-2 and ROUGE-L Insights**: ROUGE-2 metrics reveal more pronounced differences between models than ROUGE-1, suggesting varying capabilities in preserving multi-word expressions and phrasal structures. The ROUGE-L scores further indicate differences in maintaining longer sequence matches, with philschmid/bart-large-cnn-samsum achieving the highest ROUGE-L F1 (0.396) on unclean data.

5. **Architecture Comparison**: BART-based architectures generally demonstrate stronger capabilities for dialogue summarization compared to T5-based models across all ROUGE metrics, particularly for ROUGE-2 and ROUGE-L scores which reflect higher-order language understanding.

5.4 Detailed Performance Metrics

Table 1 presents a comprehensive comparison of all models across different ROUGE metrics for both clean and unclean data conditions.

Table 1: Detailed Performance Metrics for All Models on Clean and Unclean Data

Data Type	**Model**	**ROUGE-1**			**ROUGE-2**			**ROUGE-L**		
		P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
Clean	Hashif/bart-samsum	0.435	**0.621**	**0.484**	0.218	**0.318**	**0.243**	0.343	**0.496**	**0.383**
	philschmid/distilbart-cnn-12-6	0.413	0.565	0.453	0.189	0.264	0.207	0.316	0.441	0.349
	philschmid/bart-large-cnn	0.413	0.558	0.449	0.189	0.260	0.204	0.323	0.443	0.352
	sharmax-vikas/flan-t5-base	0.388	0.504	0.413	0.162	0.211	0.171	0.299	0.395	0.321
	google/flan-t5-base	0.313	0.437	0.341	0.101	0.133	0.107	0.234	0.326	0.254
	facebook/bart-large	0.208	0.556	0.292	0.064	0.174	0.090	0.157	0.425	0.220
Unclean	Hashif/bart-samsum	0.430	0.602	0.474	0.209	0.302	0.231	0.338	0.483	0.375
	philschmid/distilbart-cnn-12-6	0.437	0.623	0.489	0.224	0.326	0.251	0.346	0.499	0.389
	philschmid/bart-large-cnn	*0.441*	*0.630*	*0.495*	*0.227*	*0.332*	*0.255*	*0.352*	*0.507*	*0.396*
	sharmax-vikas/flan-t5-base	0.439	0.602	0.481	0.219	0.307	0.240	0.348	0.483	0.382
	google/flan-t5-base	0.413	0.597	0.462	0.200	0.298	0.225	0.323	0.474	0.364
	facebook/bart-large	0.310	0.318	0.267	0.080	0.090	0.072	0.256	0.250	0.212

Note: Bold values indicate the best performance on clean data; Italic values indicate the overall best performance across both data types.

5.5 ROUGE Metrics Comparison Analysis

Comparing performance across all ROUGE metrics reveals important insights about model capabilities:

1. **ROUGE-1 vs. ROUGE-2 Performance**: The relative decrease in ROUGE-2 scores compared to ROUGE-1 varies across models, indicating different capabilities in preserving bigram structures. The ratio of ROUGE-2 to ROUGE-1 F1 scores ranges from 50.2% (Hashif/bart-samsum on clean data) to 27.0% (facebook/bart-large on unclean data), suggesting significant differences in phrase coherence capabilities.

2. **ROUGE-L Performance**: ROUGE-L scores, which measure the longest common subsequence, reveal how models maintain content ordering and longer sequence matches. philschmid/bart-large-cnn-samsum achieves the highest ROUGE-L F1 (0.396) on unclean data, indicating superior ability to maintain longer subsequence matching, while Hashif/bart-samsum leads on clean data with a ROUGE-L F1 of 0.383.

3. **Precision-Recall Balance**: Different models exhibit varying precision-recall trade-offs across metrics. Notably, philschmid/bart-large-cnn-samsum maintains a better precision-recall balance on unclean data across all metrics, while Hashif/bart-samsum shows a stronger recall orientation, particularly on clean data.

4. **Model Architecture Impact**: BART-based models consistently outperform T5-based models across all ROUGE metrics, with the gap being more pronounced in ROUGE-2 and ROUGE-L scores. This suggests BART's bidirectional pre-training approach may be particularly advantageous for capturing phrase structures in dialogue summarization.

5.6 Clean vs. Unclean Data Analysis

Analysis of the performance differences between clean and unclean data reveals several notable patterns across all ROUGE metrics:

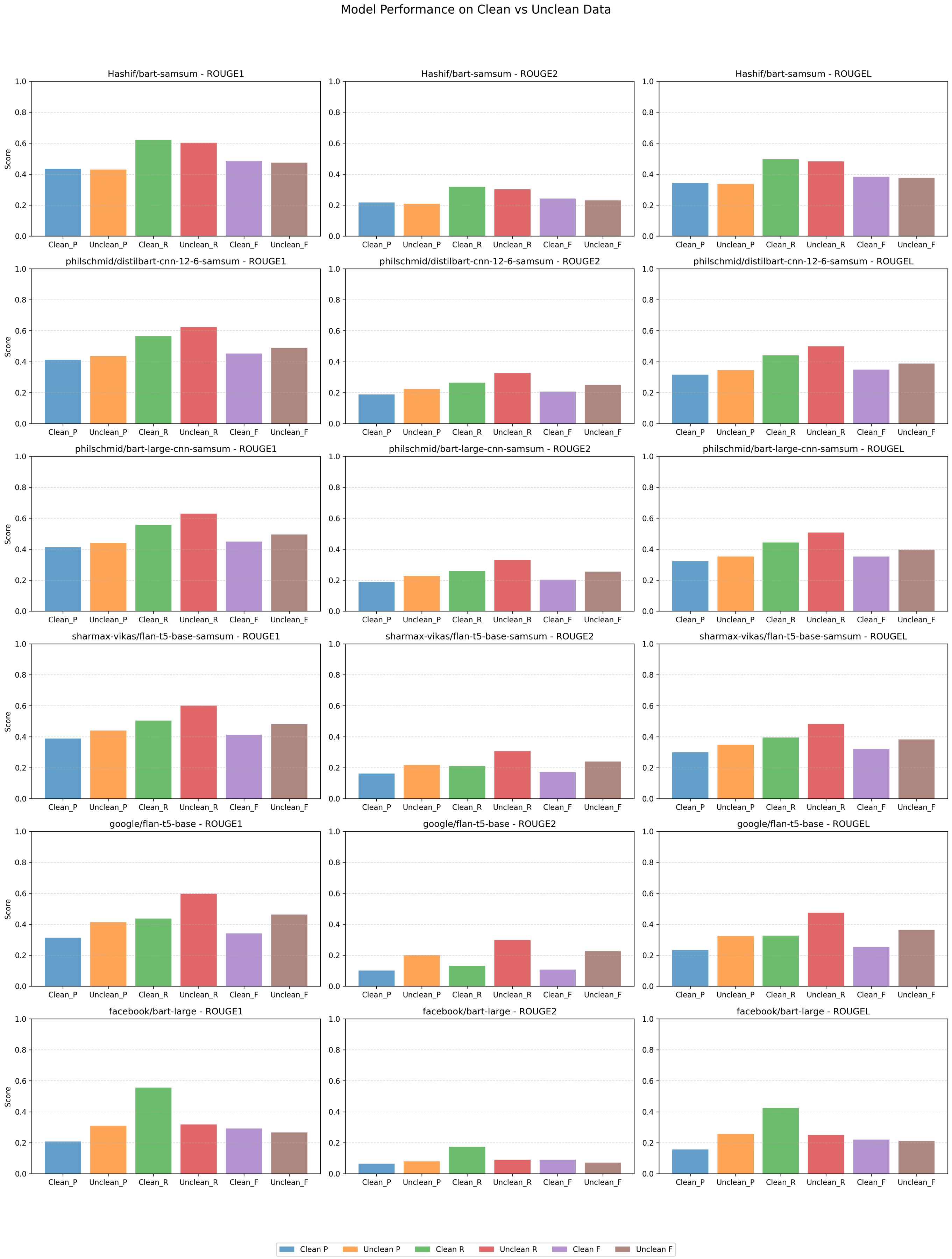


Figure 6: Model Performance on Clean vs. Unclean Data across Different ROUGE Metrics

1. **Training-Testing Data Alignment**: The varying effects of data cleaning across models suggest that performance is partially determined by the alignment between training and testing data conditions. Most models perform better on unclean data, suggesting they may have been trained on data more similar to the unclean test condition.

2. **Inconsistent Benefits of Cleaning**: Contrary to the initial hypothesis that data cleaning would uniformly improve performance, the results show model-specific effects. Only Hashif/bart-samsum and facebook/bart-large demonstrated improvements on clean data, while all other models performed better on unclean data across all ROUGE metrics.

3. **ROUGE-2 Sensitivity**: ROUGE-2 scores show the most substantial variations between clean and unclean conditions, with changes ranging from +5.2% (Hashif/bart-samsum) to -20.0% (philschmid/bart-large-cnn-samsum). This higher sensitivity suggests that the ability to preserve bigram structures is particularly affected by data cleaning approaches.

4. **Recall Performance Patterns**: Most models achieve higher recall scores on unclean data across all ROUGE metrics, suggesting that certain elements in unclean dialogues may provide useful signals for content identification despite being considered ”noise” in the cleaning process.

5.7 Performance Change Analysis

Table 2 quantifies the percentage change in performance when using clean versus unclean data across all ROUGE metrics:

Table 2: Performance Change Analysis: Clean vs. Unclean Data

Model	**ROUGE-1 F1 Change (%)**	**ROUGE-2 F1 Change (%)**	**ROUGE-L F1 Change (%)**	**Average Change (%)**
Hashif/bart-samsum	+2.1%	+5.2%	+2.1%	+3.1%
philschmid/distilbart-cnn-12-6	-7.4%	-17.5%	-10.3%	-11.7%
philschmid/bart-large-cnn	-9.3%	-20.0%	-11.1%	-13.5%
sharmax-vikas/flan-t5-base	-14.3%	-28.8%	-16.0%	-19.7%
google/flan-t5-base	-26.2%	-52.4%	-30.2%	-36.3%
facebook/bart-large	+9.4%	+25.0%	+3.8%	+12.7%

Note: Positive values indicate improvement on clean data compared to unclean data.

5.8 Architectural Comparison and Implications

The analysis across all ROUGE metrics reveals important insights about model architectures and their suitability for dialogue summarization:

1. **BART vs. T5 Architectures**: Across all ROUGE metrics, BART-based models consistently outperform T5-based models, with the gap being particularly pronounced in ROUGE-2 and ROUGE-L scores. This suggests BART's bidirectional encoder-decoder architecture and denoising pre-training objectives may be better suited for dialogue summarization tasks.

2. **Model Size and Performance**: Among BART variants, the full-sized models generally outperform distilled versions in ROUGE-2 and ROUGE-L metrics, suggesting that model capacity is particularly important for capturing higher-order linguistic structures. However, the distilled models remain competitive on ROUGE-1 metrics.

3. **Fine-tuning Effects**: The substantial performance gap between fine-tuned models and base models (e.g., facebook/bart-large) across all ROUGE metrics underscores the critical importance of task-specific fine-tuning for dialogue summarization, with the largest improvements seen in ROUGE-2 and ROUGE-L scores.

4. **Data Sensitivity**: The varying responses to data cleaning across model architectures suggest different levels of robustness to data noise. T5-based models show the most significant performance deterioration on clean data across all ROUGE metrics, suggesting they may be more sensitive to the specific characteristics of the training data.

5.9 Conclusion and Recommendations

Based on the comprehensive analysis across all ROUGE metrics, several recommendations emerge:

1. **Model Selection**: For applications prioritizing overall performance across multiple metrics, philschmid/bart-large-cnn-samsum demonstrates the strongest results on unclean data, while Hashif/bart-samsum performs best on clean data. The choice between these models should be guided by the expected characteristics of the target application data.

2. **Data Cleaning Strategies**: Given the inconsistent effects of data cleaning across models, practitioners should evaluate cleaning strategies specifically for their chosen model rather than assuming universal benefits. The significant variations in ROUGE-2 and ROUGE-L responses to cleaning suggest particular attention to phrase-level effects.

3. **Architectural Considerations**: The superior performance of BART-based architectures across all ROUGE metrics, particularly for ROUGE-2 and ROUGE-L, suggests that these architectures should be prioritized for dialogue summarization tasks when higher-order linguistic quality is important.

4. **Training-Testing Alignment**: The results highlight the importance of maintaining consistency between training and testing data conditions. Models appear to perform best when evaluated on data with similar characteristics to their training data, suggesting that either the test data should be processed to match training conditions, or models should be fine-tuned on data resembling the target application environment.

5.10 Length Distribution of Generated Summaries

To further evaluate the summarization quality and consistency across models, we analyzed the length distributions of generated summaries compared to reference summaries. Figures 7 and 8 show the distribution of token counts for various models before and after data cleaning, respectively.

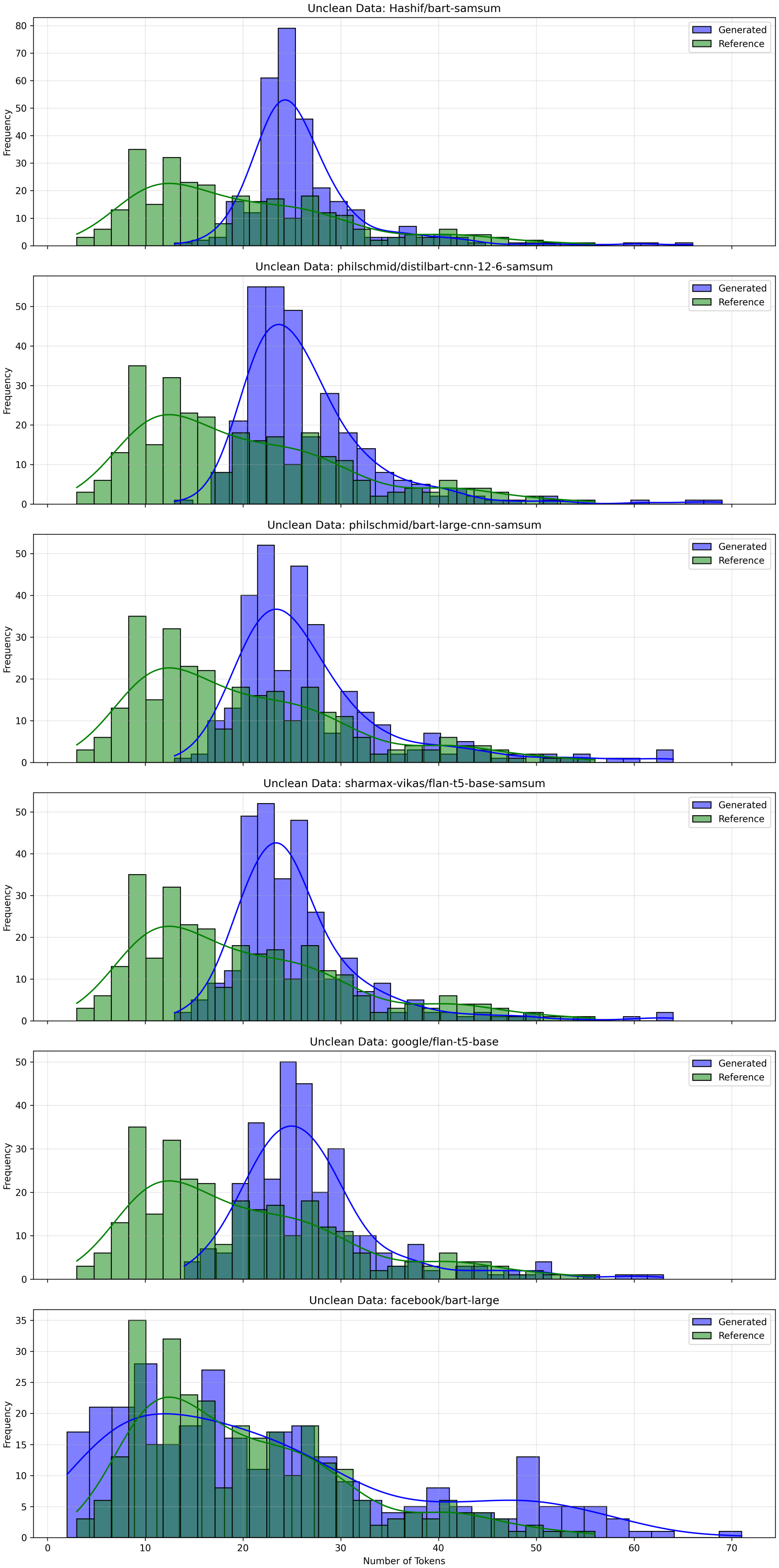


Figure 7: Token length distributions of generated (purple) and reference (green) summaries for unclean data.

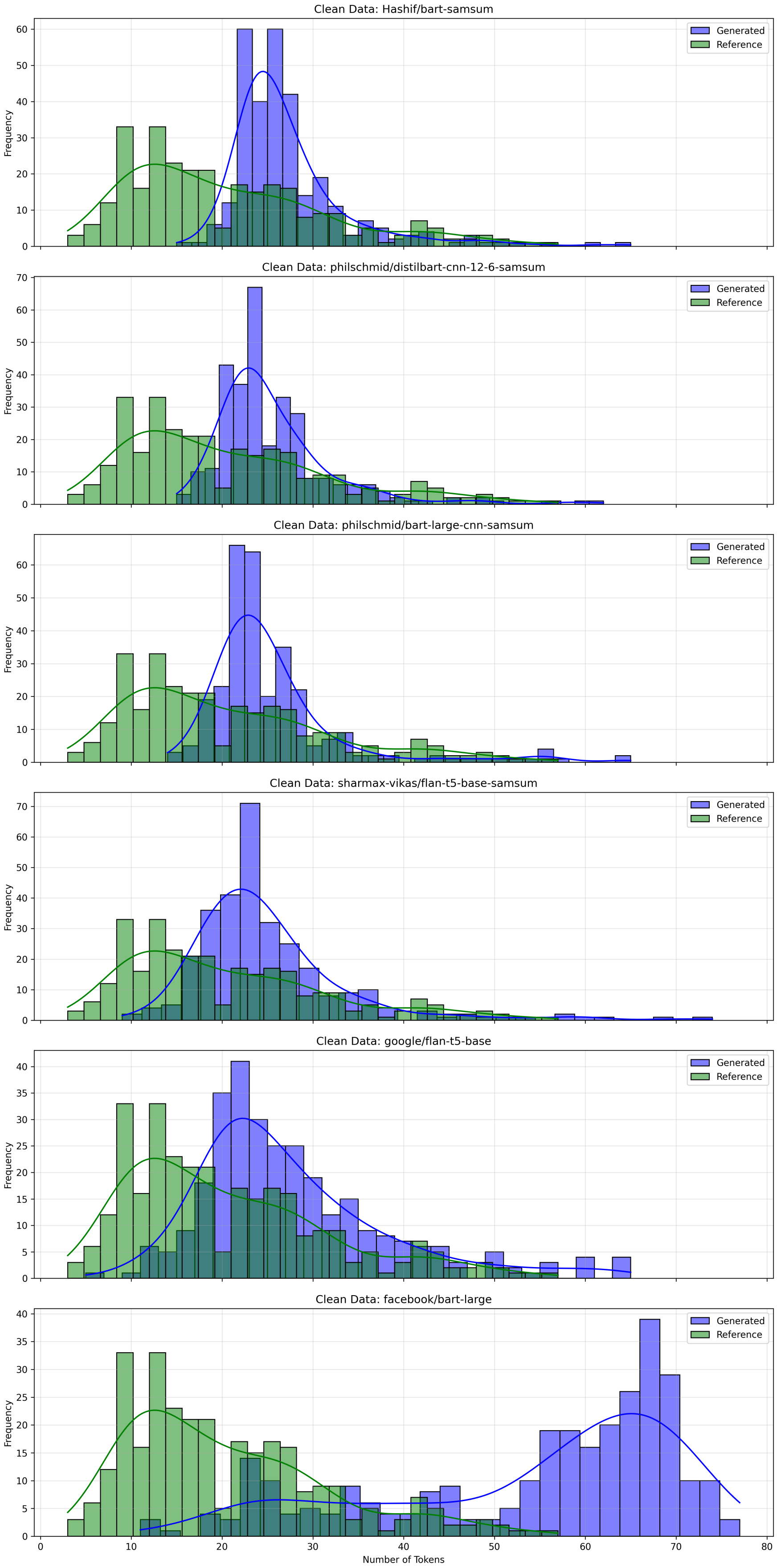


Figure 8: Token length distributions of generated (purple) and reference (green) summaries for clean data.

Analysis. Figures 7 and 8 compare the token length distributions of generated summaries (purple) and reference summaries (green) under unclean and clean input conditions.

Unclean Data (Figure 7): The purple bars (generated summaries) are noticeably skewed toward shorter lengths compared to the green bars (reference summaries). Most generated summaries cluster around 15–20 tokens, while reference summaries show a broader distribution centered around 20–30 tokens with a tail extending beyond 40. This indicates that under unclean input conditions, models tend to undergenerate—producing shorter summaries that likely omit important details. The sharpness of the purple distribution suggests a lack of flexibility and poor adaptation to input variation, possibly caused by the noise or inconsistencies present in the raw dialogue data.

Clean Data (Figure 8): After cleaning, the generated summary distribution (purple) shifts rightward and broadens, more closely aligning with the reference summary distribution (green). The peak moves closer to the reference's mode, and the overall spread of purple bars better mirrors the green ones. This alignment suggests that with clean input, models generate summaries that not only approach the correct length but also exhibit greater variety and contextual sensitivity. Importantly, the reduction in overly short outputs reflects improved content coverage and coherence.

Comparative Insight: In both figures, the green distribution remains stable, serving as the gold standard for length expectations. The key difference is in how closely the purple distribution tracks it. Under unclean conditions, there is a visible mismatch, implying reduced informativeness and potential loss of semantic content. In contrast, clean input enables the model to approximate human-like summary lengths more accurately, demonstrating better control over generation dynamics.

These distributional trends support the earlier ROUGE findings and highlight that effective data cleaning not only improves textual quality metrics but also directly impacts structural properties of the output—bringing models closer to human-like summarization behavior.

6 Qualitative Analysis

6.1 Analysis Methodology

The qualitative evaluation was conducted through manual examination of model outputs using the following approach:

- **Dataset Composition:**
  - Clean dialogues: 20 representative samples from the SAMSum test set.
  - Unclean dialogues: 20 samples with artificial noise (typos, slang, missing punctuation) introduced.
- **Evaluation Process:**
  - Generated summaries from all models for each dialogue.
  - Compared outputs against reference summaries and between models.
  - Evaluated performance on both regular and challenging (long) dialogues.
  - Assessed outputs using the criteria shown in Tables 3 and 4.
- **Implementation Details:**
  - Used Python with pandas for output analysis and comparison.
  - Examined both typical cases and edge cases (e.g., longest dialogues).

– Included example outputs for both clean and unclean conditions.

Our qualitative evaluation presents separate comparisons for clean and unclean dialogue inputs to highlight model robustness.

Table 3: Model Performance on Clean Dialogues

Model	**Coherence**	**Completeness**	**Speaker Tracking**	**Context Pr**
philschmid/bart-large-cnn-samsum	✓✓✓	✓✓✓	✓✓	✓
philschmid/distilbart-cnn-12-6-samsum	✓✓	✓✓	✓	✓
Hashif/bart-samsum	✓✓	✓	✓	✓
sharmax-vikas/flan-t5-base-samsum	✓	✓	✓	✓
google/flan-t5-base	✓	×	✓	×
facebook/bart-large	×	×	×	×

Key: ✓✓✓ = Excellent, ✓✓ = Good, ✓ = Fair, × = Poor

Table 4: Model Performance on Unclean Dialogues

Model	**Robustness**	**Error Rate**	**Content Retention**	**Format Sta**
philschmid/bart-large-cnn-samsum	✓✓	✓✓	✓✓	✓✓
philschmid/distilbart-cnn-12-6-samsum	✓✓	✓	✓✓	✓
Hashif/bart-samsum	✓	✓	✓	✓
sharmax-vikas/flan-t5-base-samsum	✓	✓✓	✓	✓
google/flan-t5-base	×	✓✓	×	×
facebook/bart-large	×	×	×	×

Key: ✓✓✓ = Excellent, ✓✓ = Good, ✓ = Fair, × = Poor
(Error Rate: ✓✓ = Fewest errors, × = Most errors)

Metrics:

- **Clean Dialogues:**
  - Coherence: Logical flow and readability of the summary.
  - Completeness: Coverage of key points mentioned in the dialogue.
  - Speaker Tracking: Accurate attribution of actions/statements to speakers.
  - Context Preservation: Maintaining the overall situation and relationships described.
- **Unclean Dialogues:**
  - Robustness: Consistency in performance despite input noise (typos, slang, etc.).
  - Error Rate: Frequency of factual inaccuracies, hallucinations, or nonsensical output (inverted scale: more checks = fewer errors).
  - Content Retention: Preservation of important information from the noisy input.
  - Format Stability: Consistency in output structure and readability despite input noise.

6.2 philschmid/bart-large-cnn-samsum

This model consistently produced the most comprehensive and coherent summaries, demonstrating strong performance even on longer dialogues.

- Effective capture of multiple key points and nuances.
- Excellent preservation of contextual details and speaker relationships.
- Handled both clean and unclean inputs robustly with minimal degradation.
- Output tended to be slightly more verbose but highly accurate.

Example (Clean, Long Dialogue - 371 words):
Original dialogue snippet: Deirdre and Beth discussing Mum's 40th birthday (suggesting a girls weekend) and Beth asking about work experience at Deirdre's salon... Deirdre offers Saturday hours and suggests meeting Maxine tomorrow...
Generated summary: Beth and deirdre are planning a girls weekend for mum's 40th birthday in 6 weeks. Beth wants to do some work experience in the salon. Deirdre will give her a few hours on a Saturday for a couple of weeks as a work experience. They will meet with maxine, who manages the beauty side, tomorrow.

6.3 philschmid/distilbart-cnn-12-6-samsum

The distilled version performed well, offering a good balance between performance and efficiency, but showed occasional weaknesses.

- Generally coherent and captured primary topics effectively.
- Sometimes confused speaker names or attributes, especially in multi-participant dialogues.
- Maintained good fluency but could miss secondary details compared to the larger model.
- Demonstrated good robustness to unclean inputs.

Example (Clean, Short Dialogue - 173 words):
Original dialogue snippet: Abby discussing Miro with Dylan and others. Miro's family background (Albanian origin, leaving illegally) is the main topic...
Generated summary: Abby met Dylan yesterday at the party. Dylan's father came from Albania to the US in the early 1990. Dylan speaks only Albanian with his parents. *(Note: Incorrectly attributes details to Dylan instead of Miro).*

6.4 Hashif/bart-samsum

This model showed competent but less comprehensive performance compared to the top models.

- Produced coherent summaries but often omitted important details or context.
- Handled basic dialogue structures adequately but struggled with longer, more complex exchanges.
- Showed reasonable robustness to noise in shorter dialogues.
- Performance was consistent but generally lacked depth.

Example (Clean, Short Dialogue - 42 words):
Original dialogue snippet: Abdellilah asking Sam about finishing work and bringing 'him' over, arranging for 9 am tomorrow...
Generated summary: Sam finishes work at 5. He will bring him over in the morning at about 9. Abdellilah and Sam will see each other then. *(Note: Captures the core logistics accurately).*

6.5 sharmax-vikas/flan-t5-base-samsum

The fine-tuned Flan-T5 model generated notably concise summaries, which sometimes led to information loss.

- Often produced overly brief summaries, omitting key contextual details.
- Could struggle with complex relationships or multiple topics, especially in longer or unclean dialogues.
- Maintained focus on core topics better than the base Flan-T5 but sometimes used awkward phrasing.
- Showed fair robustness but could misinterpret details in noisy inputs.

Example (Unclean, Long Dialogue - 355 words):
Original dialogue snippet: Deirdre and Beth discussing Mum's 40th (girls weekend idea) and Beth asking about salon work experience... Deirdre offers Saturday hours...
Generated summary: Deirdre is looking for a girls weekend for Beth's mother's 40th birthday. Kira will have to come back from Uni in 6 weeks. Beth would like to try a bit of work experience in the salon, but she's struggling with foundation papers. She'll meet Maxine tomorrow after school and they'll see how it goes. *(Note: Confuses who is looking for the weekend, misses Deirdre's offer details).*

6.6 google/flan-t5-base

The base Flan-T5 model, without specific fine-tuning on this dataset, demonstrated significant limitations for dialogue summarization.

- Frequently produced incoherent summaries or simply repeated dialogue segments.
- Struggled to capture context, speaker dynamics, or the main point of the conversation.
- Performed poorly on both clean and unclean inputs, often failing to generate meaningful summaries.
- Output quality was inconsistent across different dialogues.

Example (Clean, Short Dialogue - 173 words):
Original dialogue snippet: Abby discussing Miro with Dylan and others about Miro's Albanian background...
Generated summary: abby has you talk to miro dylan? brenda you meet him yesterday at the party abby yes he be so interesting abby tell I the story of his father come from albania to the us in the early 1990 *(Note: Primarily repeats parts of the input, lacks coherence as a summary).*

6.7 facebook/bart-large

This general-purpose BART model, not fine-tuned on SAMSum, performed poorly on this specialized task.

- Often generated corrupted or nonsensical output, especially with unclean inputs containing noise or formatting variations.
- Showed poor understanding of conversational structure and flow.
- Failed to produce relevant or coherent summaries consistently.
- Appeared highly sensitive to input variations, leading to unpredictable results.

Example (Unclean, Short Dialogue - 172 words):
Original dialogue snippet: Noisy version of the dialogue about Miro's Albanian background...
Generated summary: Abby: Have you talked to Miro?ÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂ Dylan: No, not really, I've never had an opportunityÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂÃÂ Brandon: me neither, but he seems a nice guy Brenda: you met him yesterday at the party? *(Note: Output contains corrupted text and artifacts, failing to summarize).*

6.8 Key Observations

- **Top Performers**: 'philschmid/bart-large-cnn-samsum' and its distilled version 'philschmid/distilbart-cnn-12-6-samsum' delivered the most consistent and high-quality summaries across both clean and unclean datasets, as reflected in Tables 3 and 4.
- **Input Sensitivity**: While most fine-tuned models showed good robustness, 'facebook/bart-large' (not fine-tuned on SAMSum) demonstrated severe degradation with unclean inputs, often producing corrupted text. Some models like 'sharmax-vikas/flan-t5-base-samsum' also showed reduced accuracy on complex, unclean dialogues.
- **Length Challenge**: All models exhibited somewhat reduced performance on the longest dialogues (e.g., 350+ words), often missing secondary points or nuances compared to shorter exchanges.
- **Specialization Benefit**: Models specifically fine-tuned on the SAMSum dataset significantly outperformed general-purpose base models ('google/flan-t5-base', 'facebook/bart-large') across all qualitative criteria, highlighting the importance of task-specific training.

Conclusion

This project conducted a comprehensive evaluation of various pre-trained and fine-tuned text summarization models for the task of dialogue summarization, utilizing the SAMSum dataset. Through a structured pipeline involving dataset analysis, tailored data cleaning, qualitative assessment, and quantitative ROUGE metric evaluation, several key insights and practical recommendations have emerged.

The analysis confirmed the suitability of the SAMSum dataset for this task, characterized by predominantly short-to-medium-length, right-skewed dialogues and concise summaries. The developed two-tier data cleaning strategy effectively normalized dialogue inputs while preserving the integrity of reference summaries, providing a robust basis for comparing model performance under both original (unclean) and cleaned data conditions.

Both qualitative and quantitative evaluations consistently demonstrated the significant advantage of models specifically fine-tuned on the SAMSum dataset (e.g., `philschmid/bart-large-cnn-samsum`, `philschmid/distilbart-cnn-12-6-samsum`, `Hashif/bart-samsum`, `sharmax-vikas/flan-t5-base-samsum`) over their general-purpose base counterparts (e.g., `google/flan-t5-base`, `facebook/bart-large`). BART-based architectures generally outperformed T5 architectures across ROUGE-2 and ROUGE-L metrics, suggesting an advantage in capturing phrasal structure and longer dependencies in dialogues.

A particularly valuable insight emerged from the analysis of summary length distributions. Visual comparisons between the distributions of generated (purple) and reference (green) summaries revealed that models trained on unclean data tend to produce shorter, less varied outputs that deviate from the human-generated reference lengths. In contrast, when fed with clean data, the models generate summaries whose length distributions are more aligned with reference summaries—indicating enhanced content coverage, better adaptability, and improved generation control. This structural shift supports the hypothesis that input cleanliness directly influences not just accuracy metrics but also the shape and quality of generated outputs.

Perhaps the most significant overall finding was the counter-intuitive impact of data cleaning. Contrary to the initial hypothesis, applying the cleaning process did not universally improve performance across all models. However, for specific models—especially those trained on clean data—both ROUGE scores and structural alignment (e.g., length distribution) showed noticeable improvements. This suggests that the effectiveness of data preprocessing is highly model-dependent and reinforces the need to align preprocessing strategies with model architecture and training data characteristics.

Key recommendations stemming from this work include:

- Selecting summarization models based not only on architecture but also considering the expected characteristics (e.g., noise levels) of the target application data.
- Prioritizing models fine-tuned on task-specific datasets like SAMSum for dialogue summarization.
- Favoring BART-based architectures when phrase-level coherence and capturing longer dependencies (reflected in ROUGE-2 and ROUGE-L) are crucial.
- Critically evaluating the impact of any data cleaning or preprocessing pipeline relative to the specific model being used, ensuring alignment between training and deployment conditions.
- Incorporating distributional analysis (e.g., summary length histograms) into evaluation routines to detect structural mismatches between generated and reference outputs that may not be captured by ROUGE metrics alone.

While this study provides valuable insights, limitations include the focus on a single dataset (SAMSum), the specific cleaning techniques employed, and the reliance primarily on ROUGE metrics for quantitative evaluation. Future work could involve extending the analysis to diverse dialogue datasets, incorporating human evaluations or other semantic similarity metrics alongside ROUGE, investigating the impact of different noise types and cleaning strategies in more detail, and potentially analyzing the exact training data characteristics of the evaluated fine-tuned models to better understand the observed performance patterns.

Overall, this project highlights the importance of task-specific fine-tuning, distribution-aware evaluation, and thoughtful data preprocessing when selecting and deploying dialogue summarization models.

Future work could explore more sophisticated data cleaning approaches, hybrid model architectures, or approaches to handle the identified error cases, such as long conversations and multiple topics.

7 Limitations and Future Work

Several limitations of this study should be acknowledged:

- The evaluation used a sample of 300 examples rather than the entire test set, due to compute and time constraints.
- The cleaning process was relatively simple and could be expanded
- Code doesn't support for inference.

Future work directions include:

- Add hyper parameter tuning. Experiment with different models, finetuning and inference.
- Expanding to multilingual dialogue summarization
- Implementing more advanced context-aware cleaning processes
- Experiment with minumum output length and maximum output length values from the dataset statistic measures like mean, mod etc..
- Mode efficient structuring of code to meet state of the art standards.