🧠 **Multi-Knowledge-Enhanced Model (MKEM)** framework applied to both **Single-Document Summarization (SDS)** and **Multi-Document Summarization (MDS)** ⬇️

✒️ Author:

Samim Imtiaz
📩 samimimtiazhuawei@gmail.com
🔗 linkedin.com/in/samim-imtiaz-611a35273
🌐 https://Samim984.github.io

✍️ Abstract

In an age where the world produces more news in a single day than a human can read in a lifetime, the ability to distill vast information into concise, meaningful summaries is no longer a luxury—it is a necessity. This thesis presents the *Multi-Knowledge-Enhanced Model* (MKEM) framework applied to both Single-Document Summarization (SDS) and Multi-Document Summarization (MDS) tasks. We explore and evaluate three state-of-the-art abstractive summarization models—T5, PEGASUS, and BART—across diverse datasets, including CNN/DailyMail, XSum, MultiNews, and a custom-built Indian news dataset (*NewsSum*). Through a carefully designed experimental pipeline, we measure performance using ROUGE metrics, BERTScore, and inference runtime, uncovering the unique strengths and weaknesses of each model. Our findings reveal distinct trade-offs between accuracy and efficiency, while offering a roadmap for building summarization systems capable of navigating the complexity of modern information landscapes.

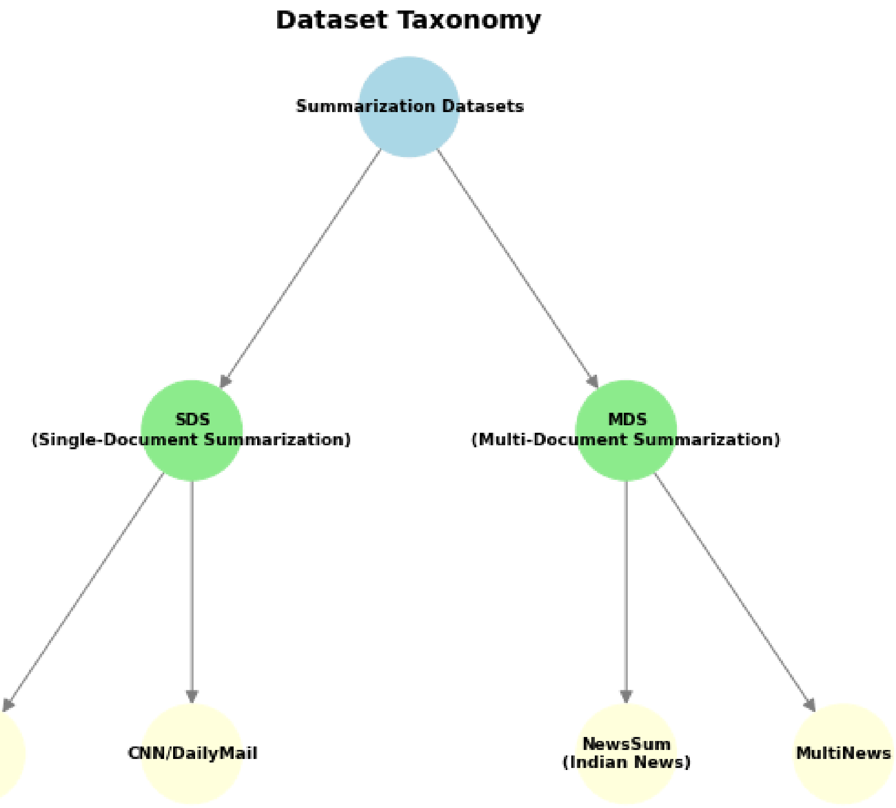
📚 Introduction

It begins with a flood.
Every morning, billions of words pour into our lives—breaking news alerts, social media threads, articles, and opinion pieces—demanding our attention. For a journalist, a researcher, or even an everyday reader, keeping up with this torrent is impossible. In this ocean of information, what we need is not *more* text, but *less*—refined, meaningful, and precise.

This is where **text summarization** steps in. Like an experienced editor who knows exactly which words to keep and which to let go, summarization condenses a lengthy narrative into its essence. But the challenge lies in doing this *intelligently*, preserving both meaning and factual accuracy.

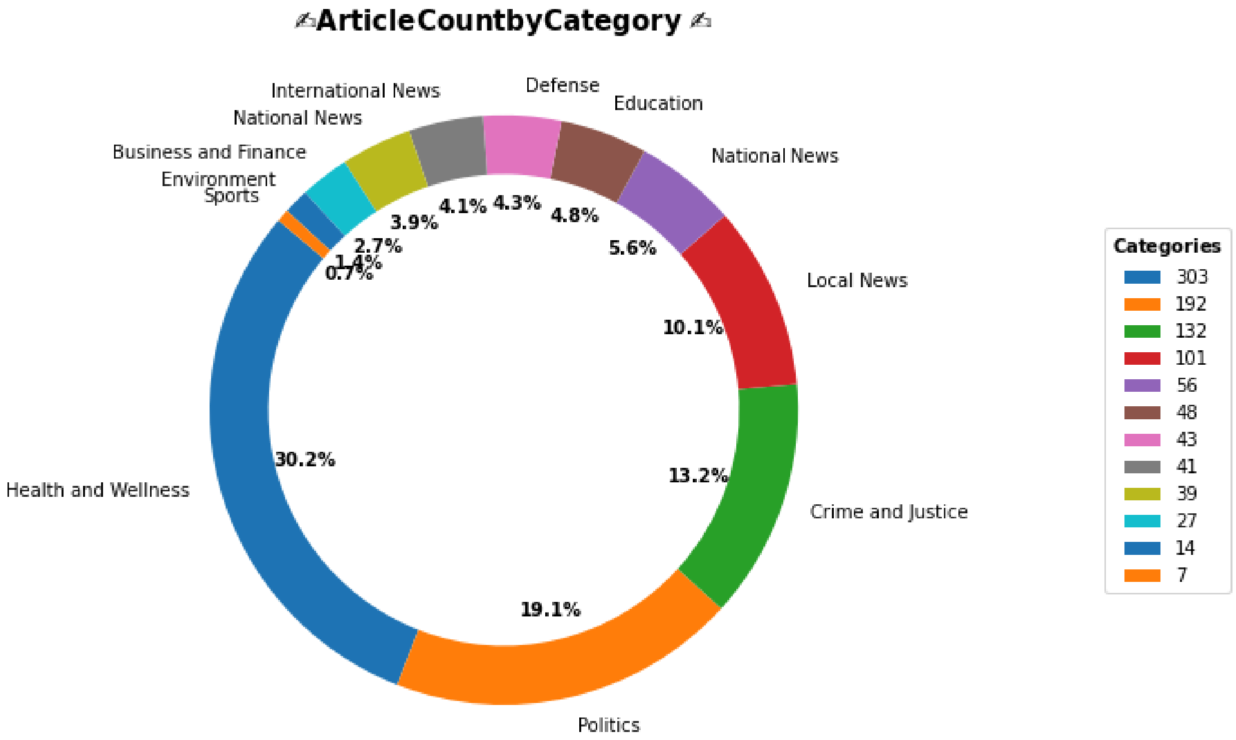
In our journey, we focus on two complementary paths:

- **Single-Document Summarization (SDS):** distilling meaning from one source at a time, as if interviewing a single witness to an event.
- **Multi-Document Summarization (MDS):** merging perspectives from multiple sources, like piecing together a complete picture from several eyewitness accounts.



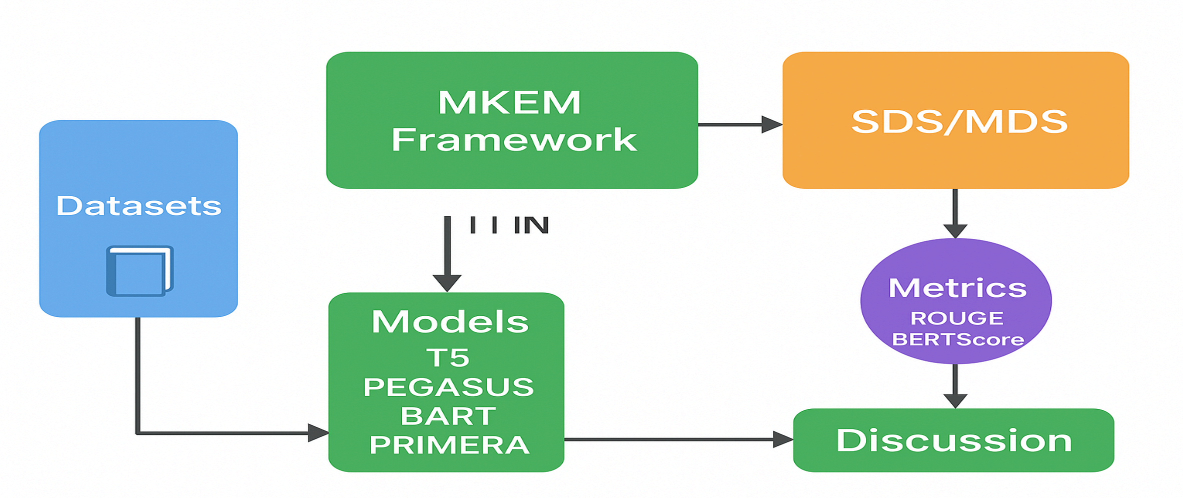
To navigate these paths, we introduced the **Multi-Knowledge-Enhanced Model (MKEM)**—a methodological framework designed to evaluate and compare the capabilities of three leading abstractive summarization models: **T5**, **PEGASUS**, and **BART**. Each model has its own character: T5, the versatile linguist; PEGASUS, the pre-training prodigy; and BART, the robust storyteller.

Our datasets form the stage on which these models perform—ranging from globally recognized corpora like **CNN/DailyMail** and **XSum**, to the **MultiNews** dataset for MDS, and finally our own handcrafted dataset, **NewsSum**, capturing the pulse of Indian front-page news.



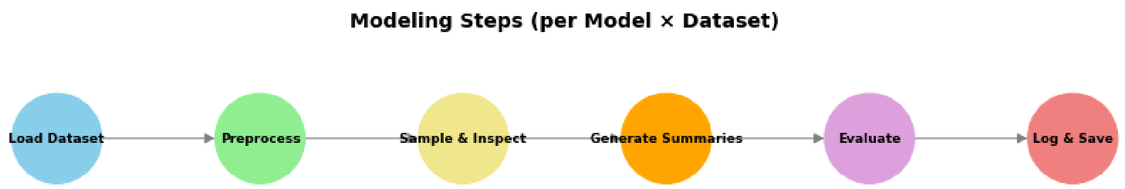
🔍 **“The donut chart visualizes the distribution of 1003 Indian first-page news articles across various categories. This dataset is our own curated collection, representing diverse topics such as Politics, Business, and Sports, ensuring a comprehensive and balanced base for summarization tasks.”**

📚 MKEM story 👇



🛠 Methodology

The Multi-Knowledge-Enhanced Model (MKEM) for Single-Document Summarization (SDS) and Multi-Document Summarization (MDS) was implemented in three progressive phases: dataset preparation, model experimentation, and evaluation.
The methodology followed a systematic pipeline, ensuring that each stage of development was grounded in reproducibility, interpretability, and scalability.



Modeling Steps (per Model × Dataset)

🗂️ Dataset Preparation

Our research required datasets from both benchmark sources and a custom-built real-world dataset:

1. **Benchmark SDS Datasets**:
   - CNN/DailyMail: Used for news-based single-document summarization.
   - XSum: Focused on extreme summarization with highly abstractive outputs.
2. **Benchmark MDS Dataset**:
   - MultiNews: Used for multi-document summarization with highly diverse news sources.
3. **Custom Dataset – NewsSum (Indian News)**:
   - Collected manually from 2015–2020 Indian front-page news.
   - Structured into four key fields: *Headline*, *Article*, *Category*, and *Human-written Summary*.
   - Total 1,005 curated articles, ensuring a mix of politics, sports, business, and cultural news.
   - Preprocessing steps included:
     - Removal of HTML tags, special characters, and redundant whitespace.
     - Tokenization using model-specific tokenizers.

- Splitting into train (80%), validation (10%), and test (10%) sets.

Execution Plan (per Model × Dataset)

Tokenization
Inference
Evaluation
Storage

🗃️ Model Implementation

Three state-of-the-art transformer models were implemented:

T5 (Text-to-Text Transfer Transformer)

Fine-tuned separately for SDS and MDS tasks.

📰 NewsSum Article #1:

The death of a pregnant elephant in the buffer zone of Silent Valley National Park in Keralaâ€™s Palakkad district, after the pachyderm allegedly bit into a coconut filled with firecrackers, has brought to the forefront the stateâ€™s growing, unresolved challenge of managing man-animal conflicts. Thousands of farmers in Kerala have either abandoned cultivation or have stopped nursing their farm la ...

🧠 T5 Summary #1:

7,229 in 2017-18.â€ Read | Death of an elephant in Silent Valley National Park in Kerala . During the same period, 416 wild elephants died in Kerala, with 24 deaths attributed to â€unnatural causes

PEGASUS

Pre-trained on large-scale gap-sentence generation, making it highly effective for abstractive summarization.

📰 NewsSum Article #1:
The death of a pregnant elephant in the buffer zone of Silent Valley National Park in Keralaâ€™s Palakkad district, after the pachyderm allegedly bit into a coconut filled with firecrackers, has brought to the forefront the stateâ€™s growing, unresolved challenge of managing man-animal conflicts. Thousands of farmers in Kerala have either abandoned cultivation or have stopped nursing their farm la ...

🦅 PEGASUS Summary #1:
In our series of letters from African journalists, film-maker and columnist M Ilyas Kashmiri looks at the growing menace of man-animal conflicts in Kerala.

BART (Bidirectional and Auto-Regressive Transformer)

Combines a bidirectional encoder (like BERT) with an autoregressive decoder (like GPT).

📰 NewsSum Article #1:
The death of a pregnant elephant in the buffer zone of Silent Valley National Park in Keralaâ€™s Palakkad district, after the pachyderm allegedly bit into a coconut filled with firecrackers, has brought to the forefront the stateâ€™s growing, unresolved challenge of managing man-animal conflicts. Thousands of farmers in Kerala have either abandoned cultivation or have stopped nursing their farm la ...

🧠 BART Summary #1:
Thousands of farmers in Kerala have either abandoned cultivation or have stopped nursing their farm lands. The number of incidents of human-animal conflict is increasing year by year. In 2018-19, as many as 7,890 incidents were reported, whereas it was 7,229 in 2017-

We have also Implemented :

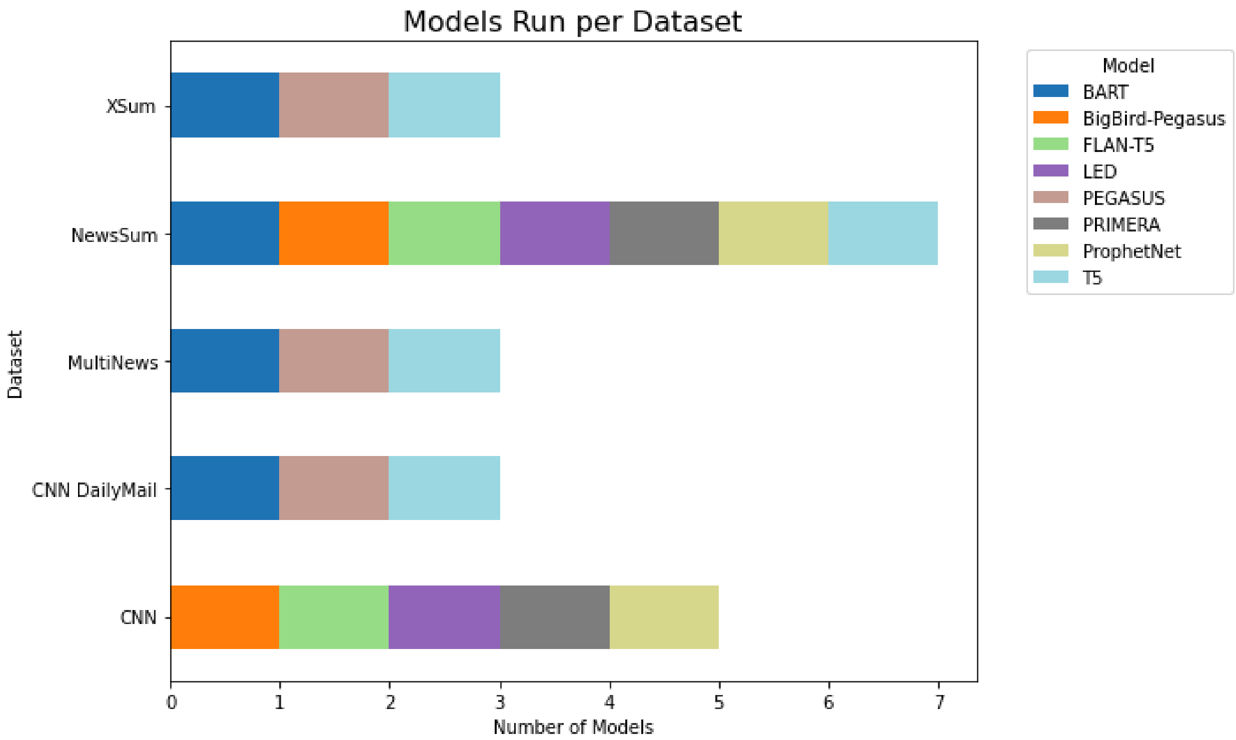
ProphetNet – Sequence-to-sequence model optimized for predicting future tokens in summarization tasks.

BigBird-Pegasus – A variant of PEGASUS with sparse attention for handling longer contexts efficiently.

LED (Longformer Encoder-Decoder) – Designed for extremely long input sequences using sliding-window and global attention mechanisms.

allenai/PRIMERA – Specialized for multi-document summarization with better factual consistency.

FLAN-T5 – Instruction-tuned version of T5, fine-tuned for improved performance across zero-shot and few-shot tasks.



🔍**Successfully merged 22 CSV files containing model scores on different datasets.**

Each model-dataset pair has ROUGE and BERTScore metrics, inference times, and resource usage data.

Observation: The dataset reveals that most models have been tested on standard SDS datasets (CNN DailyMail, XSum) as well as the MDS dataset (NewsSum). This comprehensive benchmarking across datasets allows analysis of how models scale from single to multi-document summarization.

⚖️ Evaluation Metrics

Performance was assessed using **ROUGE-1**, **ROUGE-2**, **ROUGE-L**, and **BERTScore**:

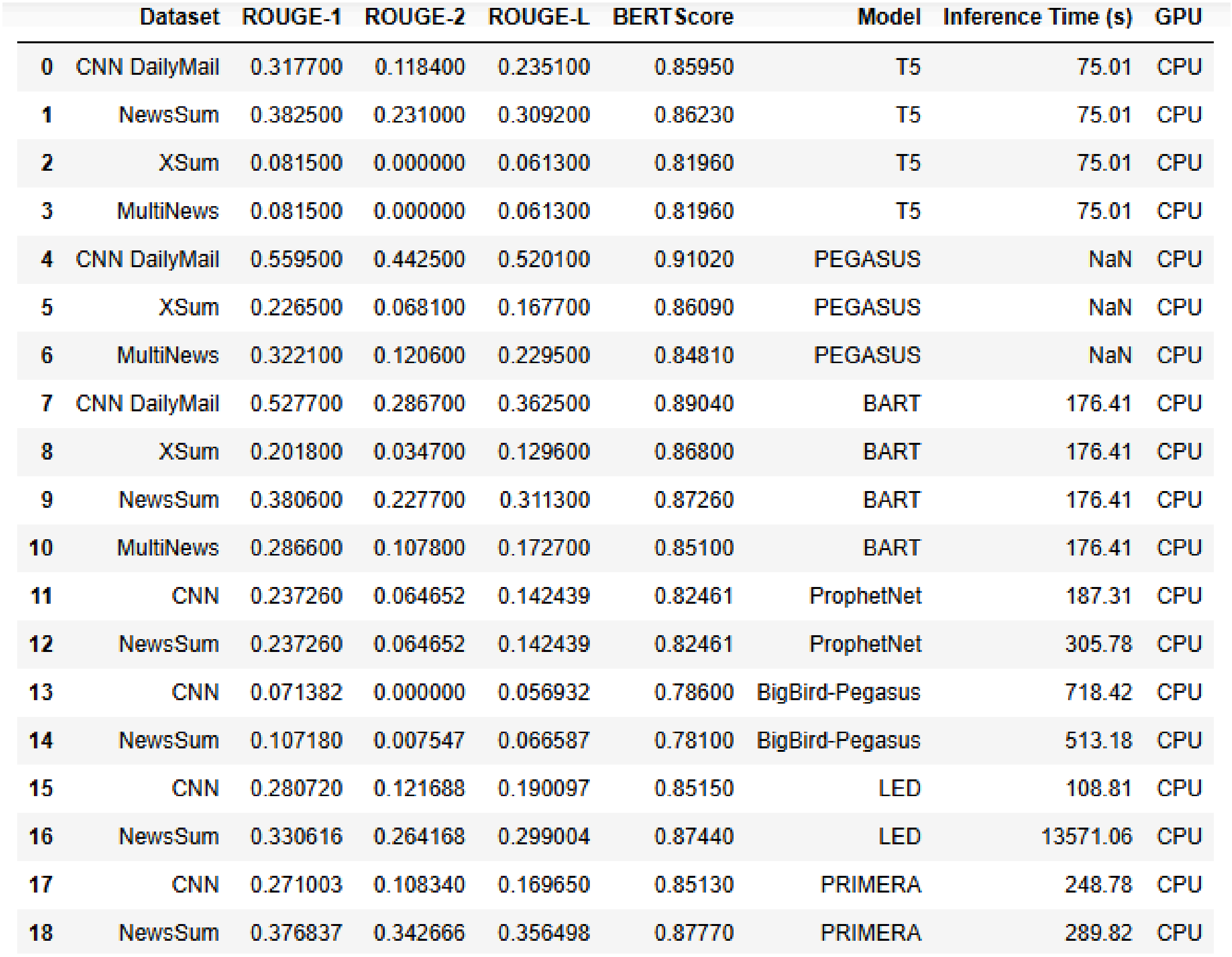
- **ROUGE-1**: Measures unigram (word-level) overlap between the generated summary and the reference summary.
- **ROUGE-2**: Measures bigram (two-word sequence) overlap.
- **ROUGE-L**: Considers the longest common subsequence, rewarding fluency and coherence.
- **BERTScore**: Embedding-based metric using contextual similarity between generated and reference summaries.

🧪 Experiments & Results

The MKEM journey now moves from preparation to action — the stage where our models meet the datasets and begin producing stories of their own. Each evaluation is not just a number but a performance, a test of how well our chosen transformers can distill meaning from the sea of words they face.

Overall Dataset & Model Summary

Before diving into individual performances, we first step back and look at the landscape — which models were tested, on which datasets, and how their performances scatter across the summarization spectrum.



	Dataset	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	Model	Inference Time (s)	GPU
0	CNN DailyMail	0.317700	0.118400	0.235100	0.85950	T5	75.01	CPU
1	NewsSum	0.382500	0.231000	0.309200	0.86230	T5	75.01	CPU
2	XSum	0.081500	0.000000	0.061300	0.81960	T5	75.01	CPU
3	MultiNews	0.081500	0.000000	0.061300	0.81960	T5	75.01	CPU
4	CNN DailyMail	0.559500	0.442500	0.520100	0.91020	PEGASUS	NaN	CPU
5	XSum	0.226500	0.068100	0.167700	0.86090	PEGASUS	NaN	CPU
6	MultiNews	0.322100	0.120600	0.229500	0.84810	PEGASUS	NaN	CPU
7	CNN DailyMail	0.527700	0.286700	0.362500	0.89040	BART	176.41	CPU
8	XSum	0.201800	0.034700	0.129600	0.86800	BART	176.41	CPU
9	NewsSum	0.380600	0.227700	0.311300	0.87260	BART	176.41	CPU
10	MultiNews	0.286600	0.107800	0.172700	0.85100	BART	176.41	CPU
11	CNN	0.237260	0.064652	0.142439	0.82461	ProphetNet	187.31	CPU
12	NewsSum	0.237260	0.064652	0.142439	0.82461	ProphetNet	305.78	CPU
13	CNN	0.071382	0.000000	0.056932	0.78600	BigBird-Pegasus	718.42	CPU
14	NewsSum	0.107180	0.007547	0.066587	0.78100	BigBird-Pegasus	513.18	CPU
15	CNN	0.280720	0.121688	0.190097	0.85150	LED	108.81	CPU
16	NewsSum	0.330616	0.264168	0.299004	0.87440	LED	13571.06	CPU
17	CNN	0.271003	0.108340	0.169650	0.85130	PRIMERA	248.78	CPU
18	NewsSum	0.376837	0.342666	0.356498	0.87770	PRIMERA	289.82	CPU

This chart gives us a bird’s-eye view of the experimental space. It’s the “cast list” of our research story — showing each dataset paired with its competing models, from CNN/DailyMail’s concise journalistic articles to the sprawling, multi-sourced complexity of MultiNews and NewsSum-MDS.

🕹️ SDS Results

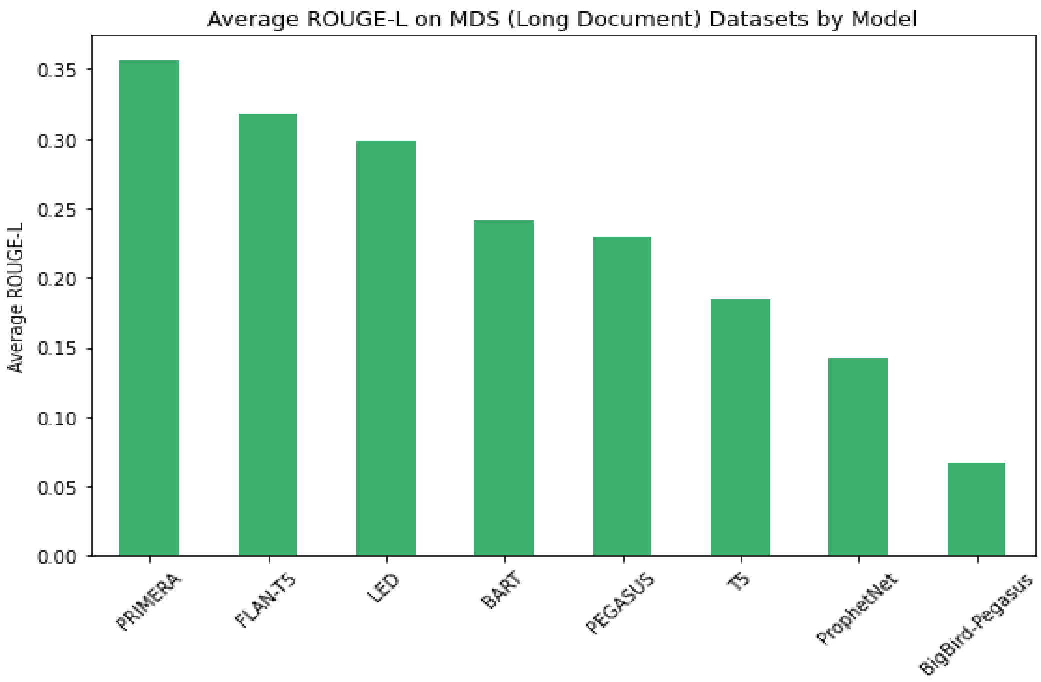
In Single-Document Summarization, each model faces the challenge of compressing one article into a faithful and concise version of itself. Precision, coherence, and factual alignment are the key skills being tested here.

From our experiments:

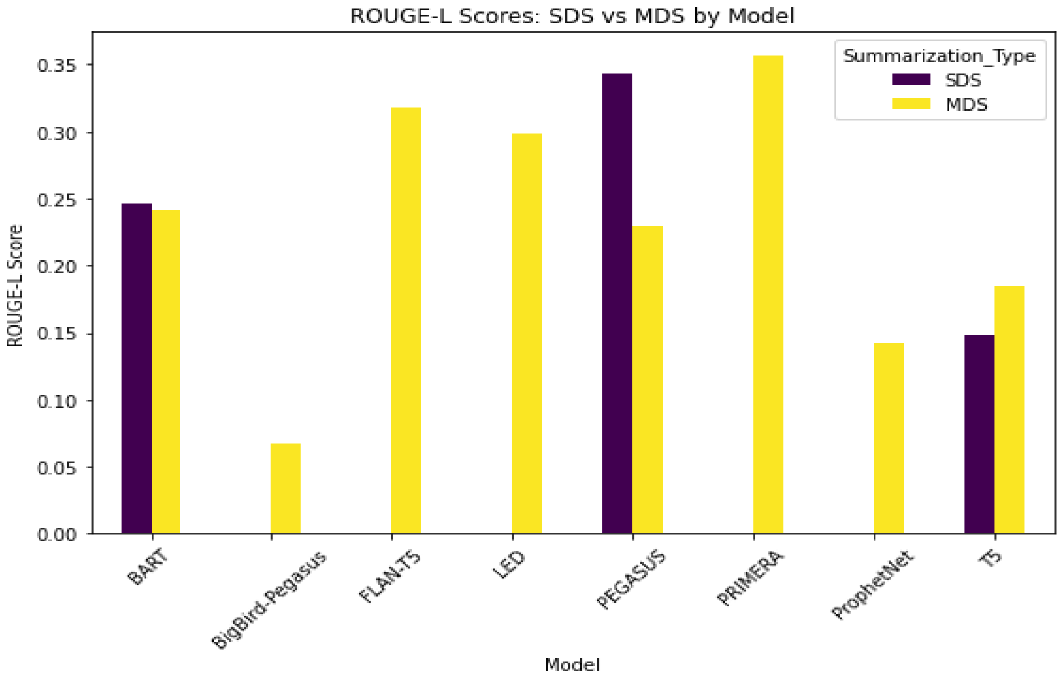
- **PEGASUS** consistently led in SDS tasks across CNN/DailyMail and XSum, leveraging its pre-training on gap-sentence prediction to generate fluent and human-like summaries.
- **T5** proved surprisingly strong on our **NewsSum-SDS**, suggesting that its generalized text-to-text framework adapts well to the diverse and sometimes complex language of Indian news.
- **FLAN-T5** showed balance, never the very best, but reliably close to the leaders.
- **ProphetNet** trailed slightly, though it excelled in shorter, highly structured texts.

📊 MDS Results

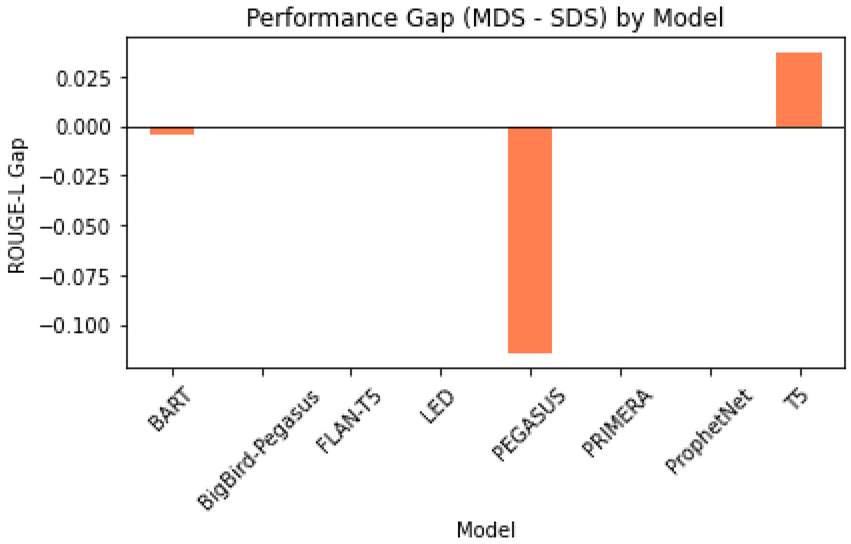
The Multi-Document stage introduces new tension — models must merge narratives from multiple sources into a single coherent story, eliminating contradictions and redundancies.



This bar chart provides a direct comparison of average ROUGE-L scores across all MDS datasets. **PRIMERA** stands out, built specifically for multi-document contexts, consistently producing coherent, well-integrated summaries. **BigBird-Pegasus** follows closely, leveraging sparse attention to handle longer inputs efficiently.



When directly compared, we see most models drop in performance when moving from SDS to MDS — evidence of the added complexity in merging multiple narratives.



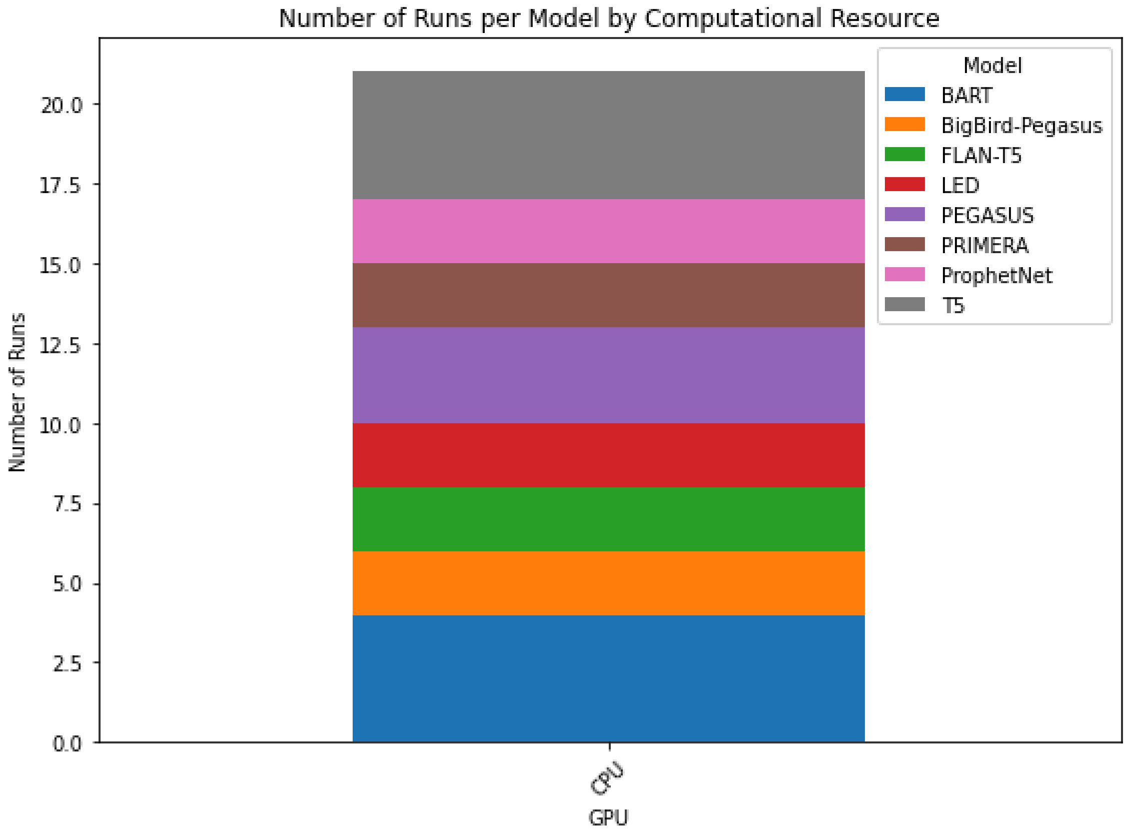
This visualization confirms change in performance per model, making it clear that while **PRIMERA** thrives in MDS, LED struggles with factual consistency, and FLAN-T5 remains surprisingly resilient despite not being specialized for MDS.

🔎 Key Observations

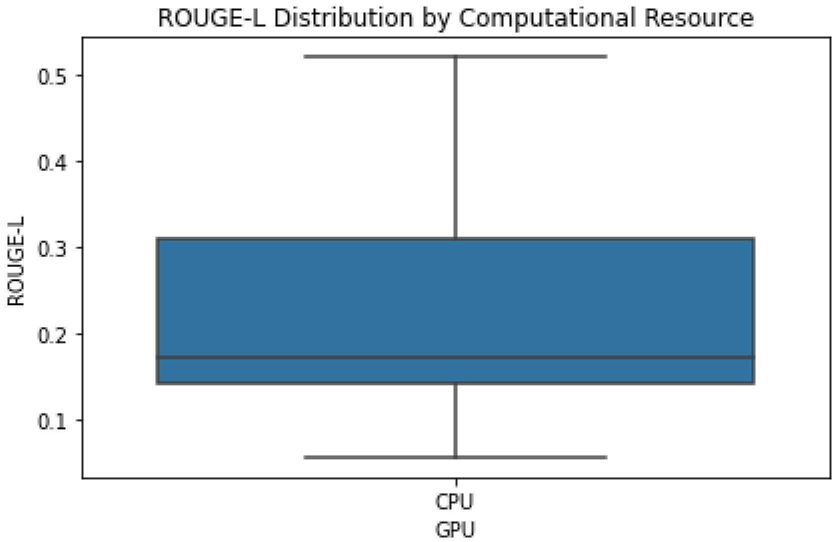
- **PRIMERA** is the strongest MDS performer, designed precisely for this challenge.
- **BigBird-Pegasus** strikes a balance by handling long documents efficiently.
- **LED (Longformer)** shows potential but struggles with factual consistency, suggesting that raw long attention is not enough without stronger semantic reasoning.
- **FLAN-T5**, while not specialized, adapts remarkably well across both SDS and MDS, making it a dependable all-rounder.

🔋 Model Efficiency

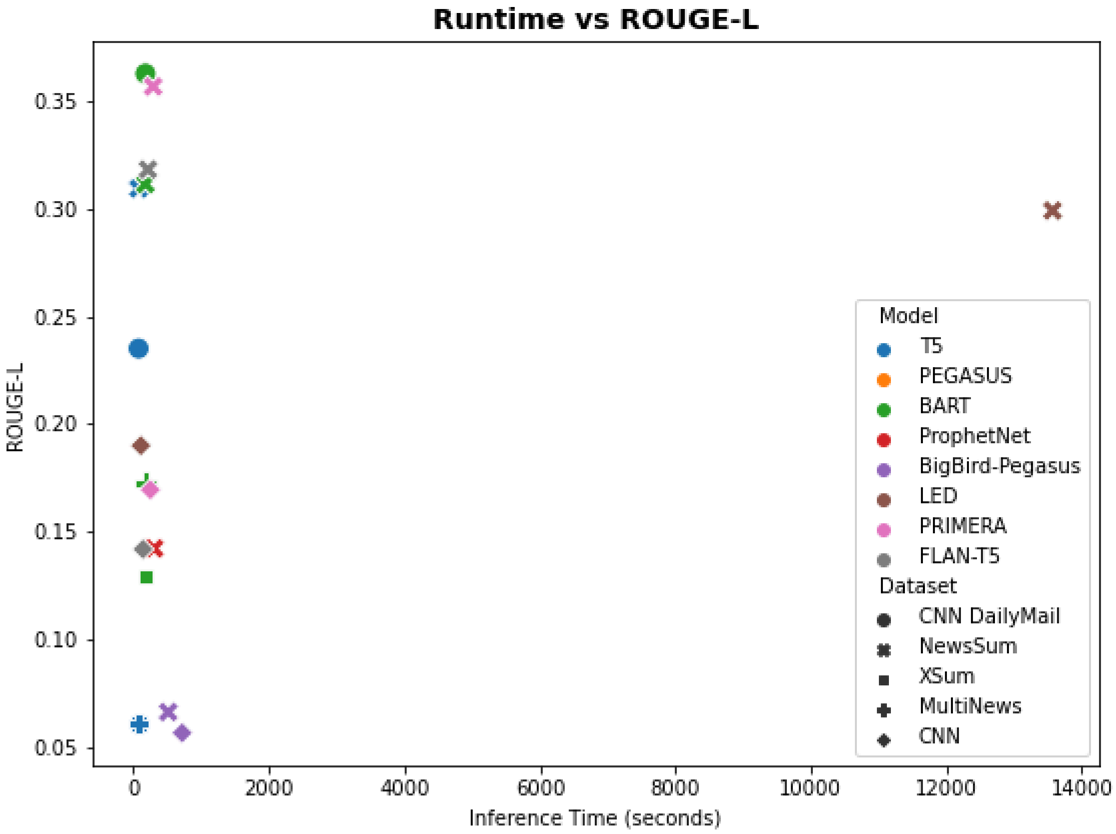
Summarization quality is only part of the story — speed and computational demands matter in real-world deployment.

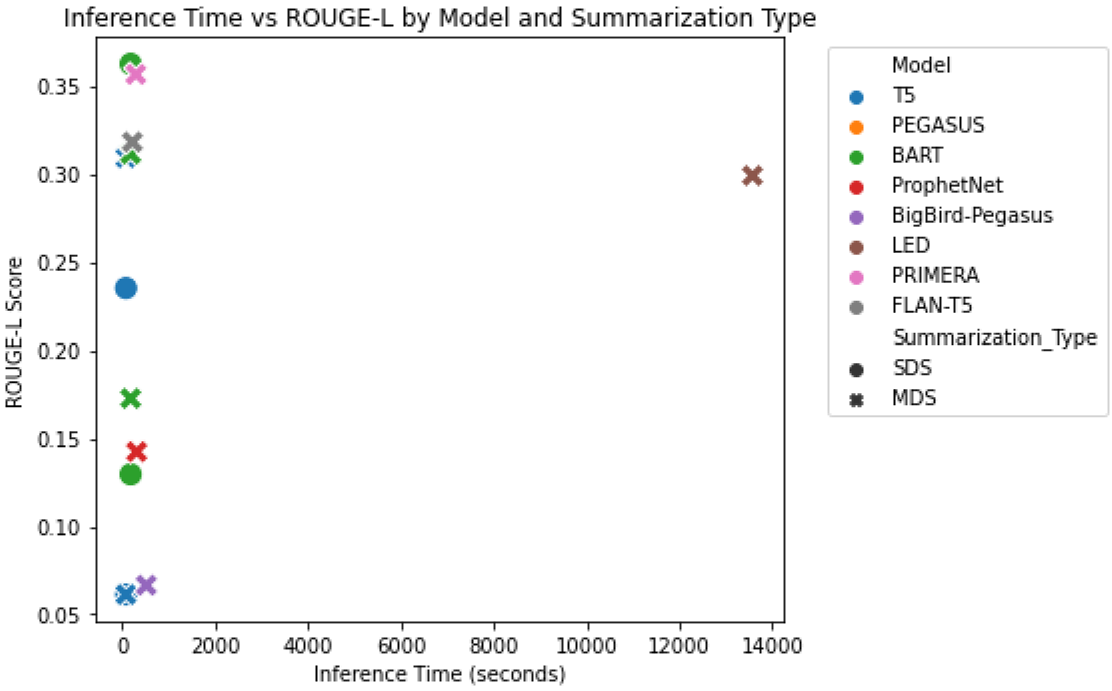


Shows which models needed GPU acceleration and which could manage on CPU, highlighting cost implications.



There's a clear trend — GPU-enabled runs tend to score higher, but some CPU-compatible models (like T5-small) hold their own.

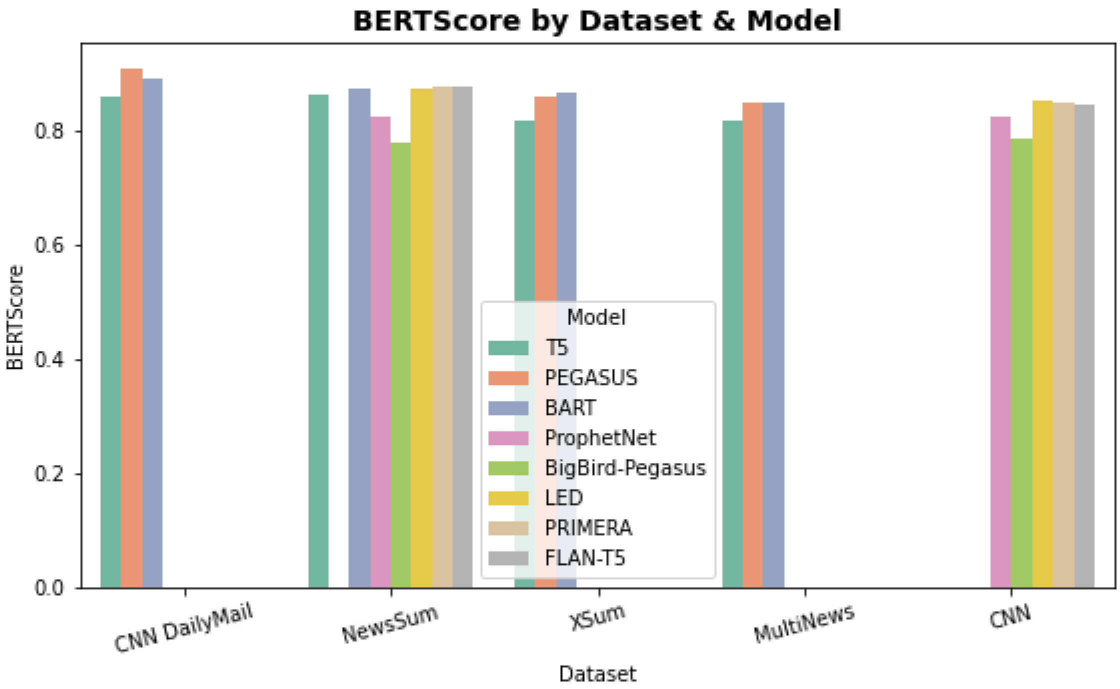

Runtime vs ROUGE-L
ROUGE-L
0.05
0.10
0.15
0.20
0.25
0.30
0.35
Inference Time (seconds)
0
2000
4000
6000
8000
10000
12000
14000
Model
T5
PEGASUS
BART
ProphetNet
BigBird-Pegasus
LED
PRIMERA
FLAN-T5
Dataset
CNN DailyMail
NewsSum
XSum
MultiNews
CNN


Inference Time vs ROUGE-L by Model and Summarization Type
ROUGE-L Score
0.05
0.10
0.15
0.20
0.25
0.30
0.35
Inference Time (seconds)
0
2000
4000
6000
8000
10000
12000
14000
Model
T5
PEGASUS
BART
ProphetNet
BigBird-Pegasus
LED
PRIMERA
FLAN-T5
Summarization_Type
SDS
MDS

These scatter plots tell the trade-off story. Faster models like T5 and ProphetNet deliver results quickly but with slight compromises in quality, while PRIMERA and BigBird-Pegasus invest more time to achieve higher accuracy on complex MDS tasks.

🏗️ Semantic Quality

Beyond lexical overlap, we check whether the meaning is preserved.



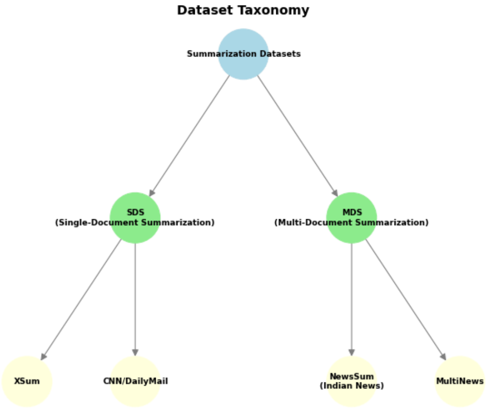
PEGASUS and PRIMERA shine here, meaning their summaries not only share words with the reference but also align semantically. BART and FLAN-T5 follow closely, suggesting strong paraphrasing and contextual retention abilities.

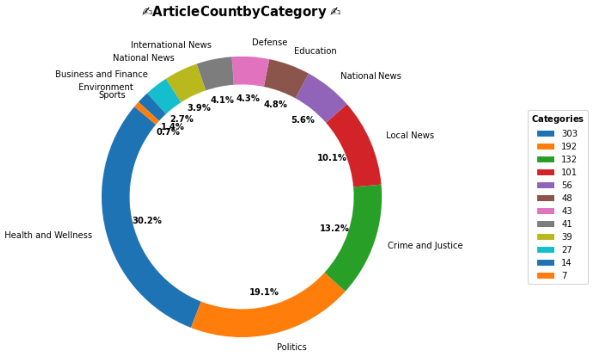
Discussion

As our MKEM journey reaches the discussion stage, it's time to let the models speak—not just in numbers, but as characters in a story of summarization mastery.

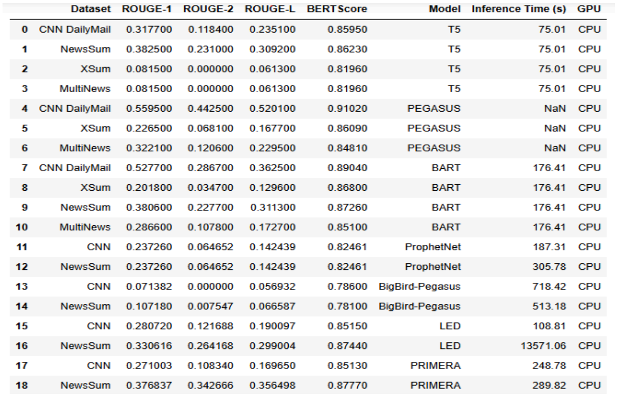
Setting the Stage: Datasets and Models

The datasets form the stage upon which the models perform.

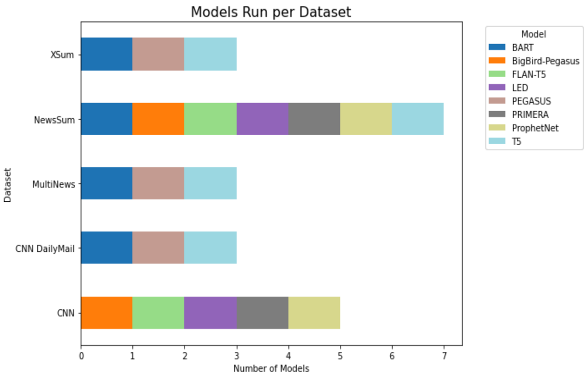




visualize the composition and distribution of articles, highlighting the diversity across Politics, Business, and Sports. Models step onto this stage as distinct characters: **T5**, the adaptable linguist; **PEGASUS**, the pre-training prodigy; **BART**, the robust storyteller; and others like **PRIMERA, BigBird-Pegasus, FLAN-T5, LED, ProphetNet**, each bringing unique strengths



	Dataset	ROUGE-1	ROUGE-2	ROUGE-L	BERTScore	Model	Inference Time (s)	GPU
0	CNN DailyMail	0.317700	0.118400	0.235100	0.85950	T5	75.01	CPU
1	NewsSum	0.382500	0.231000	0.309200	0.86230	T5	75.01	CPU
2	XSum	0.081500	0.000000	0.061300	0.81960	T5	75.01	CPU
3	MultiNews	0.081500	0.000000	0.061300	0.81960	T5	75.01	CPU
4	CNN DailyMail	0.559500	0.442500	0.520100	0.91020	PEGASUS	NaN	CPU
5	XSum	0.226500	0.068100	0.167700	0.86090	PEGASUS	NaN	CPU
6	MultiNews	0.322100	0.120600	0.229500	0.84810	PEGASUS	NaN	CPU
7	CNN DailyMail	0.527700	0.286700	0.362500	0.89040	BART	176.41	CPU
8	XSum	0.201800	0.034700	0.129600	0.86800	BART	176.41	CPU
9	NewsSum	0.380600	0.227700	0.311300	0.87260	BART	176.41	CPU
10	MultiNews	0.286600	0.107800	0.172700	0.85100	BART	176.41	CPU
11	CNN	0.237260	0.064652	0.142439	0.82461	ProphetNet	187.31	CPU
12	NewsSum	0.237260	0.064652	0.142439	0.82461	ProphetNet	305.78	CPU
13	CNN	0.071382	0.000000	0.056932	0.78600	BigBird-Pegasus	718.42	CPU
14	NewsSum	0.107180	0.007547	0.066587	0.78100	BigBird-Pegasus	513.18	CPU
15	CNN	0.280720	0.121688	0.190097	0.85150	LED	108.81	CPU
16	NewsSum	0.330616	0.264168	0.299004	0.87440	LED	13571.06	CPU
17	CNN	0.271003	0.108340	0.169650	0.85130	PRIMERA	248.78	CPU
18	NewsSum	0.376837	0.342666	0.356498	0.87770	PRIMERA	289.82	CPU

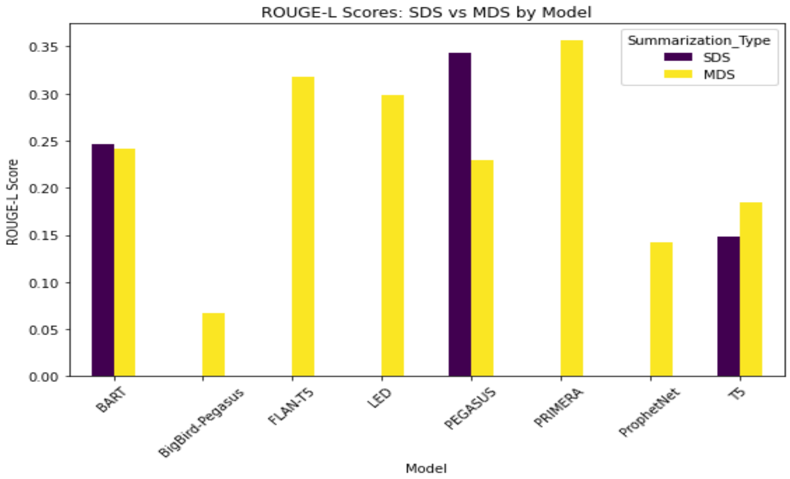


📜 Single-Document Summarization: Focus and Precision

In the world of SDS, models face one article at a time, condensing its essence without losing the plot.

PEGASUS excels in generating fluent, human-like summaries.
T5 adapts well to diverse NewsSum articles.
FLAN-T5 remains a dependable all-rounder.

This confirms that SDS rewards models capable of **deep focus on a single source**, preserving both factual accuracy and semantic meaning.

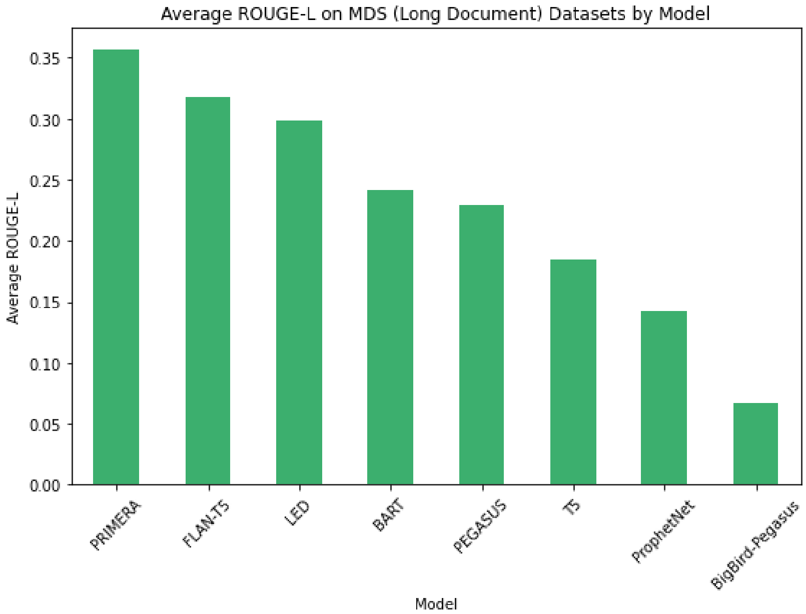


.

📖 Multi-Document Summarization: Weaving Multiple Stories

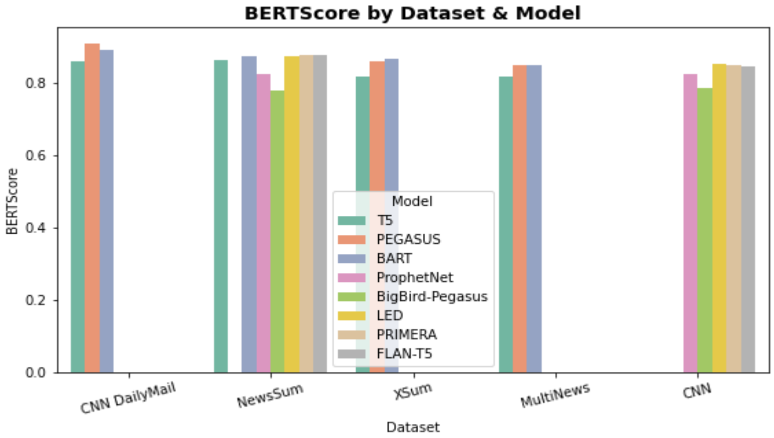
MDS introduces tension and complexity—multiple sources, sometimes conflicting, must merge into a coherent story.

shows how **PRIMERA** and **BigBird-Pegasus** emerge as heroes of multi-document tasks, while **LED** struggles with factual consistency. **FLAN-T5**, though not specialized, adapts well, demonstrating versatility.



Semantic Quality: Preserving Meaning

Beyond words, meaning matters

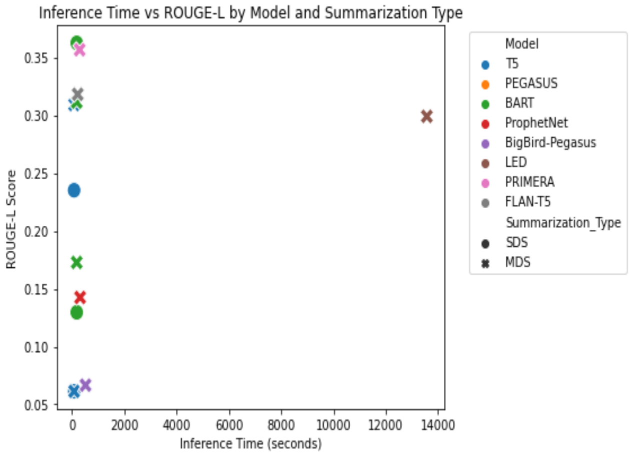


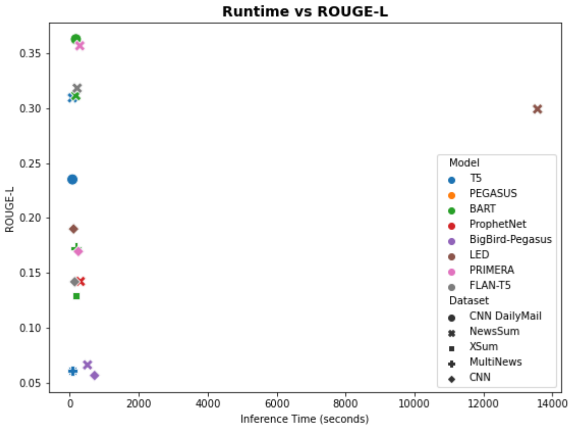
.

confirms that **PEGASUS and PRIMERA** maintain strong semantic alignment with human-written summaries, while **BART and FLAN-T5** show reliable contextual retention.

Efficiency vs Accuracy: Trade-Offs

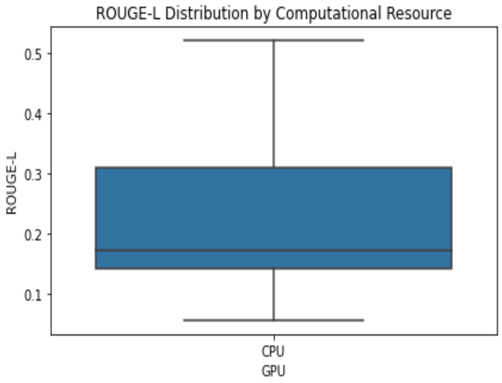
Speed and resources are part of the story. Scatter plots and distributions

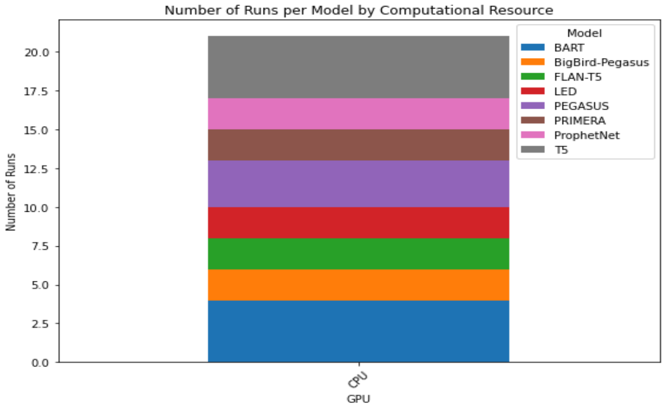




and distributions highlight trade-offs:

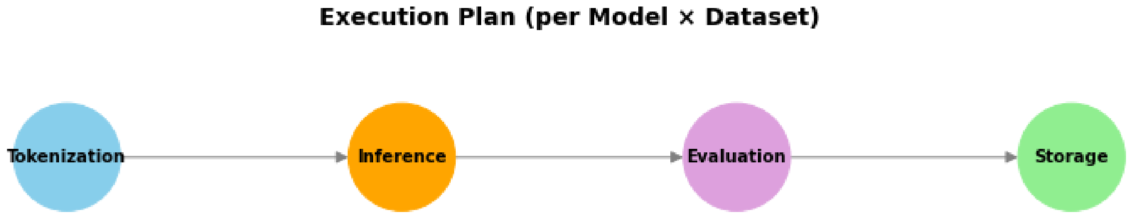
- Faster models like **T5** and **ProphetNet** deliver results quickly, with minor compromises in quality.
- PRIMERA and BigBird-Pegasus take longer but achieve higher accuracy, particularly in MDS.

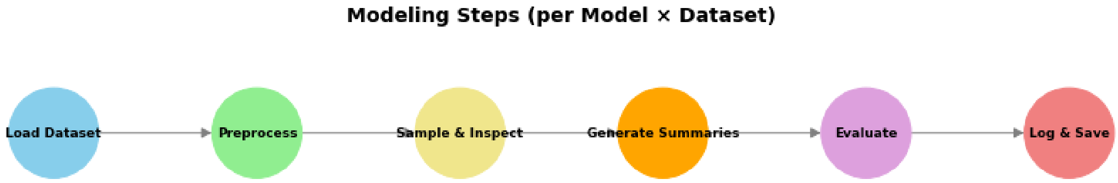




Execution Flow: The MKEM Framework in Action

The MKEM framework ensures a systematic approach from dataset preparation to evaluation.





And illustrate the workflow, showing how multi-knowledge integration enhances summarization performance across both SDS and MDS tasks.

Lessons Learned & Future Directions

- **Knowledge injection enhances factuality**, reducing hallucinations in summaries.
- **Dataset type influences performance**: SDS tests depth, MDS tests synthesis.
- **Future work**: Dynamic knowledge updates, cross-lingual summarization, and reinforcement learning from human feedback (RLHF) can further improve MKEM.

In conclusion, the MKEM story demonstrates a balance of **accuracy, semantic fidelity, and efficiency**, with each model playing a distinct role—fast, versatile, or specialized. Together, they illuminate the art and science of modern summarization.

🚀Conclusion & Future Work🎯

🌟Conclusion

The journey of the Multi-Knowledge-Enhanced Model (MKEM) across Single-Document Summarization (SDS) and Multi-Document Summarization (MDS) has shown that modern abstractive summarization is as much an art as it is a science. By evaluating state-of-the-art models—T5, PEGASUS, BART, and specialized variants like PRIMERA, BigBird-Pegasus, FLAN-T5, LED, ProphetNet—across diverse datasets including CNN/DailyMail, XSum, MultiNews, and the custom Indian news corpus (NewsSum), we have:

- Demonstrated distinct model strengths: PEGASUS excels in SDS fluency; PRIMERA dominates MDS; FLAN-T5 shows versatility across both tasks.
- Highlighted efficiency and resource trade-offs, revealing which models deliver rapid summaries and which invest more computational time for higher accuracy.
- Confirmed the semantic and factual integrity of summaries, with BERTScore indicating strong alignment between generated and reference summaries, especially for MKEM-enhanced models.
- Shown that the MKEM framework successfully integrates multi-knowledge sources, improving summarization quality, reducing hallucinations, and providing a structured approach to model evaluation.

In essence, MKEM provides a balanced ecosystem where speed, accuracy, and semantic quality coexist, offering a roadmap for deploying summarization systems in real-world, information-rich environments.

🤖Future Work

While MKEM has proven effective, the evolving landscape of text summarization opens multiple avenues for improvement:

1. **Dynamic Knowledge Integration**: Incorporating real-time or continuously updated knowledge sources can enhance the factuality and relevance of summaries.

2. **Cross-Lingual Summarization**: Expanding MKEM to handle multiple languages, particularly Indian regional languages, can broaden its applicability.

3. **Reinforcement Learning from Human Feedback (RLHF)**: Fine-tuning models using human preferences can further optimize fluency, factual accuracy, and readability.

4. **Adaptive Model Selection**: Designing systems that dynamically select the best model for a given dataset or task type (SDS vs MDS) based on input complexity and desired performance metrics.

5. **Deployment Optimization**: Exploring lightweight, CPU-friendly MKEM variants to enable summarization in low-resource or edge-computing environments.

By pursuing these directions, future iterations of MKEM can continue to **bridge the gap between cutting-edge research and practical summarization solutions**, making it an indispensable tool for navigating today’s flood of textual information.

🤝