TechSSN at SemEval-2025 Task 10: A Comparative Analysis of Transformer Models for Dominant Narrative-Based News Summarization

Pooja Premnath, Venkatasai Ojus Yenumulapalli, Parthiban Mohankumar, Rajalakshmi Sivanaiah and Angel Deborah Suseelan
Department of Computer Science and Engineering,
Sri Sivasubramaniya Nadar College of Engineering,
Chennai - 603110, Tamil Nadu, India
{pooja2110152, venkatasai2110272, parthiban2110207}@ssn.edu.in,
{rajalakshmis, angeldeborahs}@ssn.edu.in

Abstract

This paper presents an approach to Subtask 3 of Task 10 of SemEval 2025, which focuses on summarizing English news articles using a given dominant narrative. The dataset comprises news articles on the Russia-Ukraine war and climate change, introducing challenges related to bias, information compression, and contextual coherence. Transformer-based models, specifically BART variants, are utilized to generate concise and coherent summaries. Our team TechSSN, achieved 4th place on the official test leaderboard with a BERTScore of 0.74203, employing the DistilBART-CNN-12-6 model.

1 Introduction

In recent years, automated narrative extraction has gained significant attention in natural language processing (NLP), particularly in understanding complex socio-political events. A central challenge in this area is ensuring that the extracted summary faithfully captures the key information from the original narrative without losing context or meaning. This study focuses on Task 10 in SemEval 2025, which aims to extract, justify and summarize dominant narratives from news articles (Piskorski et al., 2025). The task is based upon shared tasks focusing on persuasion techniques and subjectivity, like the labs organized as part of SemEval 2023 (Piskorski et al., 2023), SemEval 2020 (Da San Martino et al., 2020), and the CheckThat! (Barrón-Cedeño et al., 2024) Labs that are part of the CLEF tasks. Specifically, we work on Subtask 3, with English data, which involves generating a concise, free-text explanation that supports the selection of a given dominant narrative. The dataset comprises articles related to the Russia-Ukraine war and climate change. Framed as a text-to-text generation problem, this subtask requires models to produce well-structured, contextually relevant explanations.

Our work primarily investigates the effectiveness of transformer-based models for summarization, with a particular focus on BART. We evaluate the variants of BART, with the subtasks' official metric- BERTScore (Zhang* et al., 2020). TechSSN achieved the 4th place on the official test leaderboard, achieving a BERTScore of **0.74203** using **distilbart-cnn-12-6**. Table 4 shows the leaderboard for this subtask. We notice that nearly all of the BART models, show fairly similar results. The system demonstrates superior performance with a BART-based architecture compared to a T5 model, highlighting a clear distinction in effectiveness. This can be attributed to BART's bidirectional encoder-decoder architecture, which enables more comprehensive contextual understanding, whereas T5 follows a standard encoder-decoder paradigm with a unidirectional decoder. However, within the BART variants, no single model consistently outperforms others across different types of BART models. The code for our work can be found in this repository.

2 Related Work

Text summarization research has evolved through distinct methodological paradigms, primarily categorized into extractive approaches, which select and concatenate salient text segments, and abstractive methods, which generate novel paraphrases to synthesize coherent summaries, with the latter gaining prominence due to their capacity to distill complex information into fluent narratives. Early efforts in abstractive summarization focused on sequence-to-sequence (seq2seq) architectures enhanced with attention mechanisms, as exemplified by (Sheik and Nirmala, 2021), who leveraged pretrained language models like BERT and GPT to address the intricate syntax and domain-specific terminology of legal texts, demonstrating the superiority of abstractive methods over extractive ones

Proceedings of the The 19th International Workshop on Semantic Evaluation (SemEval-2025), pages 2205–2212
July 31 - August 1, 2025 ©2025 Association for Computational Linguistics

while proposing a hybrid framework that combined extractive preprocessing (to identify key sentences) with abstractive generation (to refine coherence); however, this approach lacked rigorous empirical validation against standardized benchmarks, leaving its scalability and generalizability uncertain. Subsequent innovations sought to enhance semantic fidelity and structural coherence: (Song et al., 2019) introduced the ATSDL framework, integrating LSTM networks for sequential context modeling with CNNs for local feature extraction, guided by a Multiple Order Semantic Parsing (MOSP) method—a phrase extraction technique that hierarchically identifies multi-level semantic units (e.g., clauses, propositions) to steer abstractive generation—though its reliance on phrase-level processing limited scalability for long documents, as recursive segmentation introduced redundancy. To address length constraints, (Wilman et al., 2024) adapted the BART model with a chunking strategy, splitting inputs into segments shorter than 1,024 tokens, summarizing each incrementally and aggregating results, achieving competitive ROUGE-1 and ROUGE-2 scores (43.02% and 20.57%, respectively) on the CNN/DailyMail benchmark; however, their dependency on pretraining data from news articles hindered generalization to highly abstractive domains like XSum, where brevity and conceptual synthesis are paramount. Parallel efforts explored summarization's utility for downstream tasks: (Tran and Kruschwitz, 2022) combined extractive summarization (via DistilBART) with abstractive generation (using T5-3B) for cross-lingual fake news detection, translating German articles to English summaries to train classifiers, achieving a modest F1 score of 28.99% on German subsets—highlighting the challenges of language-specific biases and information loss during cross-lingual transfer. Despite these advancements, critical gaps persist: hybrid frameworks remain under-validated, long-document methods prioritize technical scalability (e.g., token limits) over conceptual nuance (e.g., preserving socio-political context), and existing models inadequately address bias propagation, particularly in contested domains like geopolitics or climate science, where dominant narratives can skew summary tone and factual balance. Our work bridges these gaps by adapting BART to generate concise (80-word), domain-aware summaries for topics such as the Russia-Ukraine war and climate change, explicitly incorporating narrative context (e.g., dominant perspectives, subnarratives) during generation to study bias modulation. Unlike prior studies fixated on metrics like ROUGE, we prioritize correctness (factual alignment with source content) and unbiasedness (neutral framing of contentious claims), tackling the unique challenge of compressing complex, ideologically charged narratives into balanced summaries—a task demanding both architectural innovation and ethical rigor, as models must disentangle factual reporting from rhetorical framing without amplifying systemic biases inherent in pretraining data.

3 Dataset

The objective of this subtask is to generate a concise, free-text explanation (up to 80 words) that elaborates on a dominant narrative within an article. Table 1 shows the data distribution. The dataset for this subtask is divided into training, development, and test sets. The training set comprises 203 instances, each containing four key fields: article_id, dominant_narrative, dominant_subnarrative, and explanation. Similarly, the development set consists of 30 instances with the same structure. The test set includes 68 instances, providing all fields except the explanation, which serves as the target output. Here, article_id represents the filename of the input article, dominant_narrative denotes the main narrative conveyed in the article, and dominant_subnarrative corresponds to its specific subnarrative. The explanation is a free-text justification that supports the identified dominant narrative (Stefanovitch et al., 2025).

Subtask 3 requires models to generate explanations that align closely with established ground truth. The official evaluation metric for this subtask is the BERTScore, which measures the average similarity between the predicted explanations and the gold-standard references. This ensures that the generated explanations are not only semantically accurate but also contextually aligned with human annotations.

4 System Overview

Summarization methods in general, can be grouped into Abstractive Summarization (Shi et al., 2020), Extreme Summarization (Cachola et al.), and Dialogue Summarization (Feng et al.). Each of these serve a different purpose in terms of the nature of the summarized text. Abstractive summarization generates summaries that may contain words and

Dataset	Total Entries	CC Count	URW Count
Train	203	93	110
Dev	30	17	13
Test	68	34	34

Table 1: Distribution of CC and URW in Train, Dev, and Test datasets.

phrases that are not present in the original text. This allows for greater flexibility in language as well as coherence. On the contrary, extreme summarization produces highly compressed summaries, often consisting of a single sentence that captures the core idea of the input text. Dialogue summarization summarizes conversational text while maintaining contextual coherence. Models for dialogue summarization are usually trained on corpora like DialogSum or Samsum (Gliwa et al., 2019).

For this study, we employed a range of transformer-based summarization models with varying architectures, depths, and training objectives. Our selection includes models based on BART (Bidirectional and Auto-Regressive Transformer), DistilBART (a distilled version of BART), T5 (Text-to-Text Transfer Transformer), and FalconAI's summarization model. Below, we provide an architectural overview of these models.

4.1 BART and its Variants

BART is a denoising autoencoder that combines a bidirectional encoder, similar to BERT, with an autoregressive decoder (Lewis et al., 2019). The bidirectional encoder enables full contextual understanding of the input text by processing tokens in both directions, while the autoregressive decoder generates outputs sequentially, ensuring fluency—a critical feature for abstractive summarization. This architecture makes BART highly effective for sequence-to-sequence tasks such as text summarization. Pretraining involves corrupting input text (e.g., through masking, sentence shuffling, or token deletion) and training the model to reconstruct the original text. This denoising objective aligns closely with summarization, as both tasks require condensing and rephrasing content while preserving meaning. In this study, we use bart-large-cnn, trained on the CNN corpus (Lins et al., 2019) for abstractive summarization, bart-large-xsum, trained on the XSum corpus (Narayan et al., 2018) for extreme summarization, and bart-large-cnn-samsum for dialogue summarization.

4.2 DistilBART

DistilBART is a distilled version of BART that retains much of its summarization capability while significantly reducing computational overhead. By using knowledge distillation, DistilBART achieves efficiency gains without a substantial drop in performance (Adhik et al., 2024). The models employed in this study include distilbart-cnn-12-6, distilbart-6-6-cnn, distilbart-xsum-12-1, distilbart-xsum-6-6, distilbart-xsum-12-3, distilbart-xsum-9-6, and distilbart-xsum-12-6. The numerical notation in DistilBART model names corresponds to the number of encoder and decoder layers, with the first number representing the encoder layers and the second indicating the decoder layers. For example, distilbart-cnn-12-6 contains 12 encoder layers and 6 decoder layers, striking a balance between computational efficiency and summarization performance (Yadav et al., 2023). In contrast, distilbart-6-6-cnn features 6 encoder layers and 6 decoder layers, making it a lighter model suited for constrained environments. Meanwhile, distilbart-xsum-12-1 retains 12 encoder layers but reduces the decoder to a single layer, optimizing it for short-text summarization.

4.3 T5

T5 reframes NLP tasks into a text-to-text format, making it highly adaptable for summarization. Unlike BART, which reconstructs text through a denoising objective, T5 is trained using a span corruption task, in which continuous chunks of text are randomly selected and replaced with special mask tokens such as $< extra_id_0 >$ and $< extra_id_1 >$. The model then learns to reconstruct the missing spans. While this approach enables T5 to handle diverse tasks, the span prediction objective prioritizes local coherence over global contextual synthesis, potentially limiting its effectiveness for abstractive summarization compared to BART. In this study, we use T5-small and mT5-small (Xue et al., 2020), the latter being a multilingual extension of T5 trained on the mC4 corpus (Dodge et al., 2021). Additionally, we include FalconAI/Text-Summarization, a model derived from T5-small but trained on a proprietary corpus. Notably, T5's text-to-text framework requires task-specific prefixes (e.g., "summarize:") during inference, adding minor overhead compared to BART's more direct sequence-to-sequence mapping.

4.4 Architectural Trade-offs

BART's pretraining aligns closely with summarization tasks due to its focus on reconstruction and fluency, whereas T5's span corruption objective emphasizes versatility across NLP tasks. While both models use encoder-decoder architectures, BART's bidirectional encoder captures richer contextual relationships, making it particularly suited for abstractive summarization where rephrasing and coherence are critical. In contrast, T5's strength lies in its unified text-to-text approach, which simplifies adaptation to multiple tasks but may sacrifice summarization-specific optimization. Additionally, BART's larger default size (e.g., 12 encoder/decoder layers in bart-large) contributes to higher computational costs but enables deeper contextual processing, while T5-small trades capacity for efficiency with fewer parameters.

Among these models, the bart-large variants are the most computationally intensive, requiring significant resources for training and inference. Models such as distilbart-12-6 and distilbart-xsum-12-6 offer a balance between computational efficiency and summarization performance. Lighter models, including distilbart-6-6, distilbart-xsum-6-6, T5-small, and mT5-small, are more suitable for environments with constrained computational resources. These differences in model architecture and computational requirements enable the selection of an appropriate model based on document length, and the nature of the generated summary. The structural differences in the models allow for varying trade-offs between processing speed, memory consumption, and summary quality.

5 Experimental Setup

This section details the steps taken in the implementation of the summarization models, including preprocessing, model training and hyperparameters.

5.1 Data Preprocessing

The first step in data preprocessing involved handling the tab-separated text file containing annotations. Since the news article texts and their corresponding annotations were stored separately, processing them efficiently required unifying them into a single JSON structure. This unified JSON file included the filename, article text, dominant narrative, and dominant sub-narrative. The annotations file was parsed, and for each entry, the corresponding news article text file was identified and combined.

The next step involved refining the text by removing specific prefixes that indicated the article type - `"URW:"` for Ukraine-Russia war articles and `"CC:"` for climate change articles. This ensured a cleaner input for subsequent processing. Following this, tokenization was performed using the Hugging Face tokenizer to prepare the text for model training. The tokenizer was applied to the input text with a maximum length of 1,024 tokens, truncating longer sequences. Similarly, the target summaries were tokenized with a maximum length of 512 tokens. The processed tokens were then structured into model inputs, with the tokenized summaries assigned as labels. The processed data looks as follows:

```
{
    "file": "EN_CC_100013.txt",
    "text": "Bill Gates Says He
       Is  The  Solution
       ...",
    "dominant_narrative": "CC:
       Criticism of climate
       movement",
    "dominant_subnarrative":
       "CC: Criticism of climate
       movement: Ad hominem
       attacks on key activists",
    "summary": "The text accuses
       climate activist Bill
       Gates for his alleged
       hypocritical behavior as
       he flies in private jets
       that pollute the
       environment while
       advocating for the
       climate cause."
}
```

The dominant_narrative, dominant_subnarrative, and the summary are concatenated together.

5.2 Model Training

The training process fine-tunes pre-trained summarization models using the Seq2SeqTrainer from the Hugging Face transformers library. The corpora that each model is trained on is listed in Table A1 in the Appendix.

The tokenized training data is fed into the model with a learning rate of 2e-5, a batch size of 16, and four training epochs. Weight decay is applied for regularization, and mixed-precision training (fp16) is enabled for efficiency. The training was conducted on an NVIDIA A100 GPU with 40GB of VRAM. The training loop includes evaluation at each epoch using the BERTScore metric, which assesses the generated summaries' precision, recall, and F1 score against reference texts. Table 2 summarizes the hyperparameters.

Hyperparameter	Value
Learning Rate	2e-5
Train Batch Size	16
Eval Batch Size	16
Weight Decay	0.01
Number of Epochs	4
FP16	True

Table 2: Hyperparameter Configuration for Training

6 Results

This section presents the results of the models on both the development and test sets. Table A2 in the Appendix illustrates the performance of various models on the development set. The best-performing model is distilbart-cnn-12-6, achieving a BERTScore of **0.74459**. This is the model that we submitted to the official test leaderboard.

Similarly, Table 3 summarizes the performance of different models on the test set, where the best-performing model is facebook/bart-large-xsum, with a BERTScore of **0.74707**. Both tables highlight the performance of BART-based models in comparison to other transformer models. In contrast, models such as T5, mT5, and FalconAI perform only marginally better than the baseline and exhibit significantly lower performance.

7 Conclusion

This work primarily explores the use of transformer-based models for news article summarization, demonstrating their effectiveness in generating concise and coherent summaries. The findings highlight the strong performance of BART-based models compared to other transformer models.

For future work, integrating large language models (LLMs) into the summarization process offers a promising direction, given their advanced capabilities and adaptability across diverse domains. Leveraging LLMs can enhance the quality and coherence of generated summaries, particularly in domain-specific and real-world applications. This integration opens up new possibilities for building more effective and context-aware summarization systems.

References

Chintalwar Adhik, Sonti Sri Lakshmi, and C Muralidharan. 2024. Text summarization using bart. In *AIP conference proceedings*, volume 3075. AIP Publishing.

Alberto Barrón-Cedeño, Firoj Alam, Tanmoy Chakraborty, Tamer Elsayed, Preslav Nakov, Piotr Przybyła, Julia Maria Struß, Fatima Haouari, Maram Hasanain, Federico Ruggeri, Xingyi Song, and Reem Suwaileh. 2024. The clef-2024 checkthat! lab: Check-worthiness, subjectivity, persuasion, roles, authorities, andnbsp;adversarial robustness. In *Advances in Information Retrieval: 46th European Conference on Information Retrieval, ECIR 2024, Glasgow, UK, March 24–28, 2024, Proceedings, Part V*, page 449–458, Berlin, Heidelberg. Springer-Verlag.

Isabel Cachola, Kyle Lo, Arman Cohan, and Daniel S Weld. Tldr: Extreme summarization of scientific documents. *arXiv preprint arXiv:2004.15011*.

Giovanni Da San Martino, Alberto Barrón-Cedeño, Henning Wachsmuth, Rostislav Petrov, and Preslav Nakov. 2020. SemEval-2020 task 11: Detection of propaganda techniques in news articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1377–1414, Barcelona (online). International Committee for Computational Linguistics.

Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner. 2021. Documenting large webtext corpora: A case study on the colossal clean crawled corpus. *arXiv preprint arXiv:2104.08758*.

Xiachong Feng, Xiaocheng Feng, and Bing Qin. A survey on dialogue summarization: Recent advances and new frontiers. *arXiv preprint arXiv:2107.03175*.

Model	BERTScore	Precision	Recall
distilbart-cnn-12-6	0.74203	0.73886	0.74568
distilbart-6-6-cnn	0.74045	0.73665	0.74481
distilbart-xsum-12-1	0.73424	0.74556	0.72380
distilbart-xsum-6-6	0.74551	0.75327	0.73839
distilbart-xsum-12-3	0.74518	0.75195	0.73900
distilbart-xsum-9-6	0.74518	0.74435	0.73994
distilbart-xsum-12-6	0.74191	0.75417	0.74174
bart-large-cnn	0.73994	0.73120	0.74939
bart-large-xsum	**0.74707**	**0.74687**	**0.74783**
bart-large-cnn-samsum	0.74187	0.73299	0.75150
T5/small	0.66684	0.65609	0.67889
mT5-small	0.62781	0.63135	0.62522
FalconAI/text-summarization	0.67638	0.66452	0.68931
Baseline	0.66690	0.65144	0.68344

Table 3: Model Performance Scores (Test)

Rank	Team	BERTScore	Precision	Recall
1	KyuHyunChoi	0.75040	0.76686	0.73517
2	WordWiz	0.74551	0.75464	0.73705
3	GPLSICORTEX	0.74280	0.75375	0.73274
4	**TechSSN**	**0.74203**	**0.73886**	**0.74568**
5	NarrativeNexus	0.73085	0.71991	0.74267
14	Baseline	0.66690	0.65144	0.68344

Table 4: Official Test Set Leaderboard

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. SAMSum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 70–79, Hong Kong, China. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Rafael Dueire Lins, Hilario Oliveira, Luciano Cabral, Jamilson Batista, Bruno Tenorio, Rafael Ferreira, Rinaldo Lima, Gabriel de França Pereira e Silva, and Steven J. Simske. 2019. The cnn-corpus: A large textual corpus for single-document extractive summarization. In *Proceedings of the ACM Symposium on Document Engineering 2019*, DocEng '19, New York, NY, USA. Association for Computing Machinery.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! Topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium.

Jakub Piskorski, Tarek Mahmoud, Nikolaos Nikolaidis, Ricardo Campos, Alípio Jorge, Dimitar Dimitrov, PurificaÃ§Ã£o Silvano, Roman Yangarber, Shivam Sharma, Tanmoy Chakraborty, Nuno GuimarÃ£es, Elisa Sartori, Nicolas Stefanovitch, Zhuohan Xie, Preslav Nakov, and Giovanni Da San Martino. 2025. SemEval-2025 task 10: Multilingual characterization and extraction of narratives from online news. In *Proceedings of the 19th International Workshop on Semantic Evaluation*, SemEval 2025, Vienna, Austria.

Jakub Piskorski, Nicolas Stefanovitch, Giovanni Da San Martino, and Preslav Nakov. 2023. SemEval-2023 task 3: Detecting the category, the framing, and the persuasion techniques in online news in a multi-lingual setup. In *Proceedings of the 17th Interna-*

tional Workshop on Semantic Evaluation (SemEval-2023)*, pages 2343–2361, Toronto, Canada. Association for Computational Linguistics.

Reshma Sheik and S. Jaya Nirmala. 2021. Deep learning techniques for legal text summarization. In *2021 IEEE 8th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, pages 1–5.

Tian Shi, Yaser Keneshloo, Naren Ramakrishnan, and Chandan K. Reddy. 2020. Neural abstractive text summarization with sequence-to-sequence models. *Preprint*, arXiv:1812.02303.

Shengli Song, Haitao Huang, and Tongxiao Ruan. 2019. Abstractive text summarization using lstm-cnn based deep learning. *Multimedia Tools and Applications*, 78(1):857–875.

Nicolas Stefanovitch, Tarek Mahmoud, Nikolaos Nikolaidis, Jorge Alípio, Ricardo Campos, Dimitar Dimitrov, Purificação Silvano, Shivam Sharma, Roman Yangarber, Nuno Guimarães, Elisa Sartori, Ana Filipa Pacheco, Cecília Ortiz, Cláudia Couto, Glória Reis de Oliveira, Ari Gonçalves, Ivan Koychev, Ivo Moravski, Nicolo Faggiani, and 19 others. 2025. Multilingual Characterization and Extraction of Narratives from Online News: Annotation Guidelines. Technical Report JRC141322, European Commission Joint Research Centre, Ispra (Italy).

Hoai Nam Tran and Udo Kruschwitz. 2022. ur-iw-hnt at checkthat!-2022: Cross-lingual text summarization for fake news detection. In *CLEF (Working Notes)*, pages 740–748.

Pascal Wilman, Talia Atara, and Derwin Suhartono. 2024. Abstractive english document summarization using bart model with chunk method. *Procedia Computer Science*, 245:1010–1019.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*.

Hemant Yadav, Nehal Patel, and Dishank Jani. 2023. Fine-tuning bart for abstractive reviews summarization. In *Computational Intelligence: Select Proceedings of InCITe 2022*, pages 375–385. Springer.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A Appendix

The Appendix contains details of the corpora that the BART, DistilBART, T5 and mT5 models are trained upon. This is present in Table A1. Table A2 shows the BERTScores of the summarizers on the development dataset.

Model	Training Corpora
distilbart-cnn-12-6	CNN/DailyMail
distilbart-6-6-cnn	CNN/DailyMail
distilbart-12-6-cnn	CNN/DailyMail
distilbart-xsum-12-1	xsum
distilbart-xsum-6-6	xsum
distilbart-xsum-12-3	xsum
distilbart-xsum-9-6	xsum
distilbart-xsum-12-6	xsum
bart-large-cnn	CNN/DailyMail
bart-large-xsum	xsum
bart-large-cnn-samsum	SAMSum
T5/small	C4
mt5-small	mC4
FalconAI/text-summarization	Not available

Table A1: Pre-trained datasets for different summarization models

Model	BERTScore	Precision	Recall
distilbart-cnn-12-6	**0.74459**	**0.75019**	**0.73945**
distilbart-6-6-cnn	0.73666	0.73798	0.73562
distilbart-xsum-12-1	0.72287	0.74048	0.70656
distilbart-xsum-6-6	0.73900	0.75293	0.72582
distilbart-xsum-12-3	0.73425	0.74488	0.72414
distilbart-xsum-9-6	0.74154	0.75452	0.72926
distilbart-xsum-12-6	0.73879	0.75253	0.72580
bart-large-cnn	0.73870	0.73575	0.74189
bart-large-xsum	0.73730	0.74157	0.73339
bart-large-cnn-samsum	0.73122	0.73197	0.73076
T5/small	0.67528	0.66615	0.68509
mT5-small	0.68125	0.69401	0.67036
FalconAI/text-summarization	0.67827	0.66915	0.68862
Baseline	0.66690	0.65144	0.68344

Table A2: Model Performance Scores (Development)