UM-MDAS: A Unified Model Framework for Multi-Document Abstractive Summarization

Reshma Khan[1], Sarwar Shah Khan[1], Ijaz Ali[1], Muhammad Saad Salman[2], Abdur Rahman[3], Mian Saeed Akber[4], Muhammad Yousaf[4]
[1] Department of Computer Science, Iqra National University, Swat Campus, Pakistan
[2] Minor Bugs, Software House, Lahore, Pakistan
[3] Go Jins, Software House, Lahore, Pakistan
[4] Department of Computer Science, University of Engineering and Technology, Mardan, Pakistan
*Correspondence:sskhan0092@gmail.com

Citation | Khan. R, Khan. S. S, Ali. I, Salman. M. S, Rahman. A, Akber. M. S, Yousaf. M, "UM-MDAS: A Unified Model Framework for Multi-Document Abstractive Summarization", IJIST, Vol. 8 Issue. 2 pp 504-519, April 2026
Received| February 28, 2026 **Revised|** March 30, 2026 **Accepted|** April 03, 2026 **Published|** April 07, 2026.

The volume of information across diverse sources has been increasing rapidly, and thus, there has been a need for effective multi-document summarization systems. Multi-Document Abstractive Summarization (MDAS) aims to generate concise and coherent summaries that capture common core ideas between related documents. Despite recent improvements in transformer-based models in the transformer-based models, the current methodologies of MDAS still struggle with cross-document redundancy, inconsistent content selection, and the inability to control redundancy in the salience of a summary. The article introduces a framework, UM-MDAS, which is a unified and modular architecture of multi-document abstractive summarization to overcome these challenges with a structured pipeline architecture. The framework builds sentence-level semantic representations based on SBERT and combines the representations into document-level representations through min-max pooling to identify salient and extreme semantic features. The cosine similarity is used to measure semantic relevance and eliminate redundancy across document clusters, whereas the abstractive summary generation is done through the PEGASUS sequence-to-sequence model. The experiments carried out with MultiNews data indicate that UM-MDAS enhances semantic coverage, fluency, and coherence. Evaluation results show ROUGE-1 (53.83), ROUGE-2 (21.00), ROUGE-L (23.54), and BERTScore (86.41), indicating that the proposed framework performs well in capturing both surface-level and contextual similarities. The article introduces a framework, UM-MDAS, which is a unified and modular architecture of multi-document abstractive summarization to overcome these challenges with a structured pipeline architecture.

Keywords: Multi-Document Summarization; SBERT; PEGASUS; NLP; Deep Learning

Introduction:

Text is an essential medium of conveying information, ideas, and knowledge, both through simple and complicated sentences as well as written and spoken communication, including books, research articles, and numerous other documents [1]. As the digital content grows faster, it has become more difficult to control, interpret, and analyze a great deal of textual information. Text summarization also provides a good solution in that large volumes of textual information are summarized in a small and meaningful form, retaining the main message and context of the text used. Summarization can effectively condense information because it highlights the important information and eliminates redundancy, facilitating decision-making in academic, professional, and industrial settings [2].

Text summarization may also be done manually or automatically, as per the magnitude, urgency, and accuracy with which the task needs to be done. Manual summarization has the advantage of human comprehension and interpretation; nevertheless, it is time-intensive, expensive, and infeasible in terms of large datasets or continuously changing datasets. Automatic Text Summarization (ATS), on the other hand, is based on computational means to create summaries. It can be used to summarize news articles, scientific literature, and legal documents in large volumes. The amount of textual information keeps increasing beyond what humans can handle; ATS has become a mandatory technology rather than a convenience [3].

According to the information representation, text summarization methods can be categorized into extractive and abstractive methods. Extractive summarization identifies and merges significant sentences, explicitly coping with the source text but producing summaries that are factually faithful but often lack fluency and coherence [4]. In contrast, abstractive summarization forms new sentences as a paraphrastic and synthesizing expression of the original meaning, and is more similar to human-style summarization. More recent works, like the use of deep learning and transformer-based networks, such as sequence-to-sequence and pre-trained models, such as T5, BART, and PEGASUS, have shown notable improvements in the fluency, coherence, and readability of abstractive summaries [4].

We can also divide summarization according to the number of documents that are input. Single-document summarization concerns the creation of a summary of a single source, whereas multi-document summarization involves synthesizing information from a number of related documents into one summary [5]. Multi-document summarization is even more problematic, since redundancy, finding complementary and contradictory information, and source coherence must be maintained. These challenges are especially prominent in abstractive multi-document summarization, when the development of factually correct, coherent, and short summaries involves the use of sophisticated semantic comprehension as well as appropriate content synthesizing policies.

Despite notable progress in transformer-based models, recent studies highlight that MDAS still suffers from critical limitations, including inadequate handling of cross-document redundancy, inconsistent identification of salient content, and limited control over information diversity and summary focus. Many existing approaches either rely heavily on end-to-end architectures without explicit redundancy control or lack structured mechanisms for integrating semantic representations across multiple documents. As a result, the generated summaries may include repetitive information, miss important themes, or lack coherence when synthesizing content from diverse sources.

There is a need for a structured and modular framework that can effectively integrate semantic representations, control redundancy, and ensure coherent content selection across multiple documents.

To address this gap, the proposed research focuses on developing an advanced multi-document abstractive summarization framework that improves redundancy reduction,

enhances content selection, and generates more fluent and coherent summaries. By incorporating recent natural language processing techniques along with robust evaluation measures, this work aims to contribute to more effective knowledge synthesis and improved accessibility of information in both academic and real-world applications. Following the main contributions:

This study introduces a novel unified and modular framework (UM-MDAS) for multi-document abstractive summarization, specifically designed to address key challenges such as redundancy, inconsistent content selection, and lack of coherence.

The framework uses SBERT to generate sentence-level embeddings and introduces a min–max pooling strategy to construct document-level representations, effectively capturing both salient and extreme semantic features.

The proposed framework follows a systematic pipeline consisting of preprocessing, embedding, summarization, post-processing, and evaluation, enabling better control and interpretability compared to end-to-end approaches.

Literature Review:

The section provides a review of current studies on multi-document text summarization, especially those that focus on abstractive techniques and transformer-based models. There is an extensive set of techniques investigated by prior studies to solve such problems as reducing redundancy, selecting content, and coherence in summaries. The practice of the analysis of these works will be useful in locating the existing restrictions in the process of multi-document abstractive summarization and emphasizing the gaps in the research that will encourage the given structure.

The strategy that [6] support is an unsupervised multi-document summarization system called SummPip, which can effectively operate in a low training data situation. The algorithm constructs sentence graphs on the foundation of linguistic and semantic relationships and spectral clustering to cluster sentences and consequently generate summaries through compression of clusters.

The Abstractive Summarization Framework by [7], whose model is based on Semantic Link Networks (SLNs), is used to model documents using the interrelated concepts and events. It is a technique that attempts to identify salient concepts and semantic consistency between two or more documents.

[8] introduced a neural abstractive MDS model that uses explicit document graphs representations through similarity and discourse graphs to model cross-document relationships. Graph-based encoding facilitates coherence and informality in summaries generated. An approach by [9] for summarizing news, implementing models XLNet and GPT-2 with other models such as BART, T5, Pegasus, LED, Big Bird, and Distill BERT on the BBC News dataset from the Hugging Face library. XLNet is a bi-directional model that analyses all words in a sequence to consider interdependencies, whereas GPT-2 generates diverse and coherent text based on context learned from a large corpus of data. The results suggest that GPT-2 performed better due to its capacity to generate coherent and contextually appropriate text across various categories and tasks

[10] introduced SgSum, a graph-based framework that formulates multi-document summarization as a subgraph selection problem, capturing intra- and cross-document sentence relations to produce coherent summaries. [11]proposed a collaborative learning approach that leverages single-document summarization data to improve multi-document summarization using shared encoder–decoder parameters, achieving strong results but requiring high computational resources and high-quality SDS data.

[12] proposed a multi-granularity interaction network that combines extractive and abstractive summarization by modeling interactions at both word and sentence levels. Attention mechanisms enable information transfer across granularities to improve coherence

and informativeness, but the model increases computational complexity and requires careful attention design and monitoring.

[13] presented EMSum, an entity-conscious abstractive MDS framework that incorporates entity nodes into heterogeneous graphs. The explicit modeling of entity relationships in documents enhances saliency and redundancy in EMSum. The strategy, however, makes the model more complex and dependent on the correct recognition of entities, which can be restrictive in domain-level robustness. [14] have proposed PEGASUS-XL, a saliency-based MDS model that combines semantic scoring, redundancy reduction, and sparse attention. On Multi-News and XSum, the model has reached high performance, but it is a computationally expensive model, which points to the current lack of scalability.

[15] conducted a comparative evaluation of extractive and abstractive approaches for news text summarization using the CNN-Daily Mail dataset. To point out that, even though transformer-based abstractive models like PEGASUS can produce high-quality summaries because they have been pre-trained on a wide variety of text collections, extractive models are faster and more efficient, and are therefore more appropriate in real-time use cases. The study also focuses on the practical shortcomings of the implementation of abstractive models because they demand massive computing resources to train and infer. The main weaknesses identified are that the abstractive summarization models are computationally expensive and time-consuming, which limits their application in real-time or resource-constrained systems. [16] introduced PRIMERA, a pre-trained model designed for multi-document summarization that minimizes the need for dataset-specific architectures and large amounts of labeled fine-tuning data. PRIMERA employs a Longformer-based encoder-decoder transformer along with a novel entity-based sentence masking pre-training objective, enabling the model to effectively connect and aggregate information across multiple documents. Extensive experiments on six multi-document summarization datasets from three different domains, under zero-shot, few-shot, and full-supervised settings, show that PRIMERA outperforms prior state-of-the-art pre-trained and dataset-specific models, which is further corroborated by human evaluations.

Specifically, we now compare existing approaches based on key dimensions such as scalability, computational complexity, coherence, redundancy handling, and dependency on annotated data. For instance, while graph-based methods effectively model inter-document relationships, they often suffer from high computational overhead and complex graph construction. Similarly, transformer-based models demonstrate strong performance in generating coherent summaries but are limited by their high resource requirements and lack of efficiency in low-resource or real-time settings. Moreover, hybrid and collaborative approaches improve performance by leveraging multi-level interactions, yet they introduce additional architectural complexity and dependency on high-quality training data. Entity-aware models enhance semantic consistency but rely heavily on accurate entity recognition, which may limit their robustness across domains.

Methodology:

This paper proposes UM-MDAS, a unified pipeline framework for multi-document abstractive summarization designed to generate concise, coherent, and factually consistent summaries from multiple related documents. The framework addresses key challenges in MDAS, including semantic redundancy, inconsistent content selection, and loss of important information, while producing fluent and readable summaries suitable for practical applications.

UM-MDAS is a five-step pipeline that includes preprocessing, embedding, summarization, post-processing, and evaluation. Preprocessing cleans the text, normalizes the text, and segments the document to eliminate noise and standardize heterogeneous inputs. During the embedding part, sentence-level semantic representations generated by

SBERT are pooled together into document-level embeddings through min-max pooling to obtain salient and extreme semantic representations. Cosine similarity is used to model semantic relevance and find overlapping content in the documents.

The PEGASUS neural sequence-to-sequence transformer model is used to do the abstractive summary generation and generates sensible summaries that capture the main information spread among multiple sources [17]. Additional post-processing steps apply MMR-based redundancy removal on generated summaries to eliminate redundancy via the Maximal Marginal Relevance (MMR), keep crucial factual information via entity preservation, and rearrange sentences at will to achieve better logical flow. Lastly, ROUGE and BERTScore are used as methods to compare the quality of the summary, which offer two complementary lexical and semantic scores. The general UM-MDAS workflow is shown in Figure 1. The pipeline is applied to the dataset and consists of the following stages:

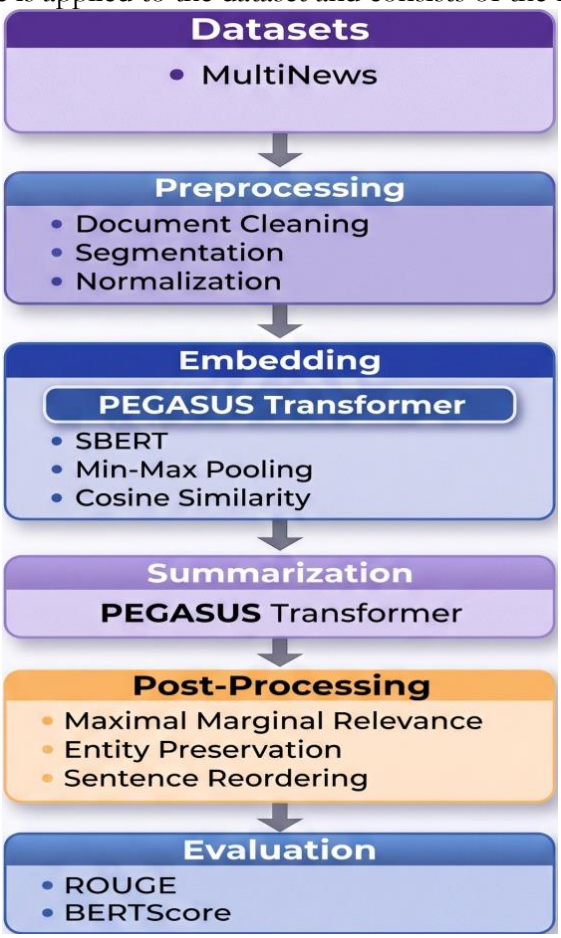

Figure 1. Multi-Document Abstractive Summarization Workflow

Data Preprocessing:

Preprocessing is an important step in multi-document abstractive summarization as it ensures clean, structured, and model-ready input for embedding and summarization stages. Multi-document datasets differ in the length of documents and their structure and formatting, so preprocessing is the key to the attainment of robustness, reproducibility, and semantic consistency in the generated summaries.

A text cleaning is used to eliminate noise and non-informative data like emojis, symbols, URLs, email addresses, and non-ASCII characters that may interfere with tokenization and lower embedding quality [18]. Also, whitespace normalization and punctuation standardization are done to guarantee constant boundaries of the sentences and also, consistent generation of embedded data. After cleaning, the document cluster is divided

into coherent document segments to maintain document boundaries, minimize semantic contamination, and minimize redundancy in summarization. The segmentation is based on both dataset-specific delimiters or natural paragraph breaks, and very short or empty ones are filtered. Generally, preprocessing converts heterogeneous multi-document inputs into these regular and semantically unified representations, which facilitate the effective generation of embedding and abstractive summarization.

Embedding:

The embedding stage is a critical step of the specified Multi-Document Abstractive Summarization (MDAS) pipeline since it is called to transform textual data into the form of compact numbers without semantics loss and contextual relationships. It refers to the process of converting discrete units of text to continuous spaces of vectors to enable computation of semantic similarity, relevance, and structural associations using numerical operations [19]. The embedding step helps downstream summarization models to be able to operate with semantically high-level representations rather than with raw texts through the computation of the organized text form. This work makes use of embeddings generated using such a model as Sentence-BERT (SBERT), which is a transformer-based model that is specifically created to generate semantically meaningful sentence-level representations [20]. SBERT provides strong contextual and semantic sentence representations and, therefore, is suitable for identifying paraphrases, quantifying semantic similarity, and the content overlap across multiple documents. In order to generate document-level representations of sentence embeddings, minmax pooling is used across sentence vectors and allows the model to learn both dominant and extreme semantic features of each document group.

Cosine similarity between the pooled embeddings is calculated to model semantic relevance and redundancy across documents. The similarity measure facilitates useful detection of salient contents and redundant information that is shared by groups of documents, a requirement for redundancy-sensitive multi-document summarization. The resulting embeddings give semantically consistent and low-dimensional representations of multi-document inputs, which enables correct content selection, reduction of redundancy, and coherent generation of abstractive summaries at later steps of the MDAS pipeline.

SBERT-Based Embedding with Min–Max Pooling and Concise Similarity:

In this embedding approach, each document is first segmented into sentences and encoded using Sentence-BERT (SBERT), a transformer-based model designed to generate semantically meaningful contextual embeddings [20] is shown in Figure 2. For a document containing T tokens, SBERT produces token-level embeddings:

$$h_t \in R^d, \quad t = 1, 2, \dots, T \quad (1)$$

which are stacked into a token embedding matrix:

$$H = [h_1, h_2, \dots, h_T]^\top \in R^{T \times d} \quad (2)$$

To handle variable-length documents, padding tokens are introduced and excluded from computation using an attention mask $M \in \{0,1\}^T$where $M_t = 1$ indicates a valid token and $M_t = 0$ denotes padding. This ensures that only semantically meaningful tokens contribute to document representations.

Min–Max pooling is then applied across valid tokens to obtain a fixed-length document embedding. For each embedding dimension i, the pooled values are computed as:

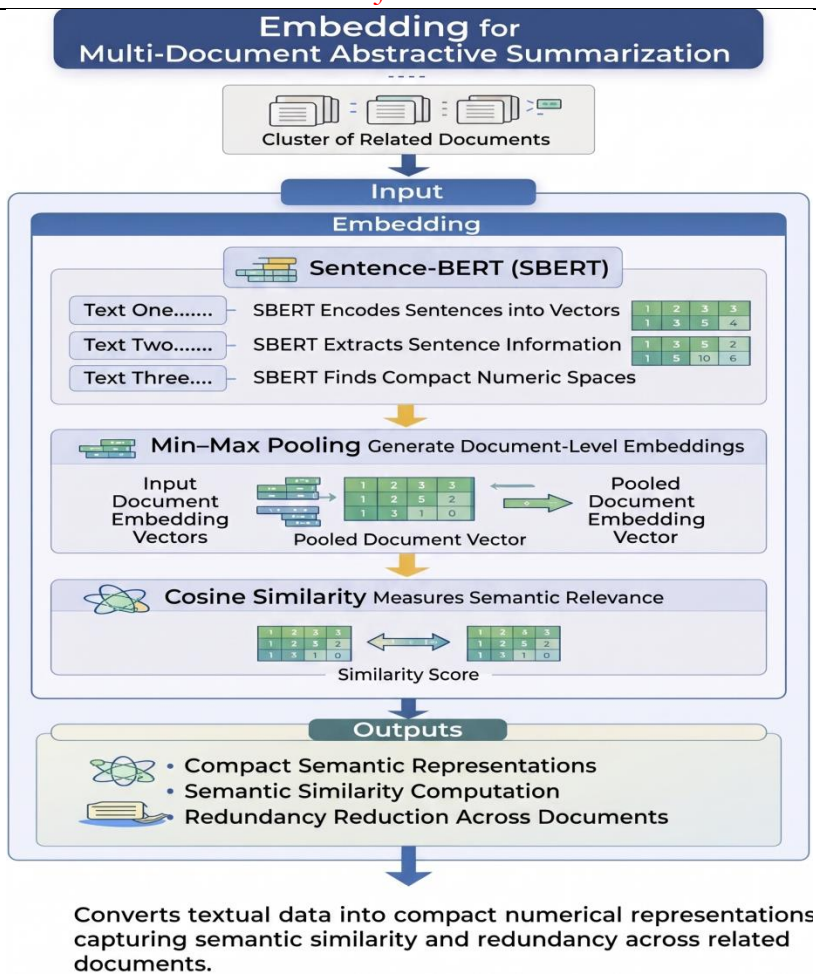

Figure 2. Embedding for multi-document abstractive summarization using SBERT, min-max pooling, and cosine similarity.

$$d_{min}(i) = \min_{t:M_t=1} H_{t,i}, d_{max}(i) = \max_{t:M_t=1} H_{t,i}, i = 1,2,\dots,d \quad (3)$$

The minimum vector captures rare or subtle semantic cues, while the maximum vector represents dominant and salient features. These vectors are concatenated to form the final document embedding:

$$d = [d_{min} \mid d_{max}] \in R^{2d} \quad (4)$$

To ensure numerical stability, any $\pm\infty$ values arising from masking are replaced with zeros. The semantic variability of each dimension is implicitly represented by the range.$d_{max}(i) - d_{min}(i)$.Computing embeddings for all documents, semantic relationships are quantified using cosine similarity:

$$sim(d_i, d_j) = \frac{d_i \cdot d_j}{|d_i||d_j|} \quad (5)$$

The resulting similarity matrix highlights semantically related documents and supports redundancy reduction, clustering, and informed abstractive summarization.

For example, given three pooled document embeddings:

D1= [0.2,0.3,0.5‖0.7,0.8,0.9], D2= [0.1,0.4,0.6‖0.6,0.9,1.0], D3= [0.3,0.2,0.4‖0.8,0.7,0.9]

Computing cosine similarity between these embeddings produces a symmetric similarity matrix:

$$S = \begin{bmatrix} 1.0 & 0.92 & 0.88 \\ 0.92 & 1.0 & 0.85 \\ 0.88 & 0.85 & 1.0 \end{bmatrix} \quad (6)$$

Clearly identifying semantic overlap among documents.

This similarity computation and embedding algorithm can be summed up in the following way:

Algorithm: SBERT with Min–Max Pooling and Concise Similarity

Input

- A set of documents $\boldsymbol{D} = \{\boldsymbol{d_1}, \boldsymbol{d_2}, \dots, \boldsymbol{d_N}\}$
- Maximum token length L
- SBERT encoder and tokenizer

Output

- Document embeddings $\boldsymbol{d_i} \in \boldsymbol{R}^{\mathbb{2}\boldsymbol{d}}$
- Pairwise similarity matrix $\boldsymbol{S} \in \boldsymbol{R}^{\boldsymbol{N}\times\boldsymbol{N}}$

Step 1: For each document $\boldsymbol{d_i}$**∈D**:

a. Tokenize $\boldsymbol{d_i}$ and create an attention mask M_i to identify valid tokens.

b. Encode tokens using SBERT to obtain token embeddings $\boldsymbol{H_i} \in \boldsymbol{R}^{\boldsymbol{T}\times\boldsymbol{d}}$

c. Compute masked element-wise minimum: $\boldsymbol{d_{i,min}(j)} = \min_{t:M_{i,t}=1} \boldsymbol{H_{i,t,j}}$d

d. Compute masked element-wise maximum: $\boldsymbol{d_{i,max}(j)} = \max_{t:M_{i,t}=1} \boldsymbol{H_{i,t,j}}$

e. Replace ±∞ values with zeros for numerical stability.

f. Concatenate $\boldsymbol{d_{i,min}}$ and $\boldsymbol{d_{i,max}to}$ form $\boldsymbol{d_i} = \left[\boldsymbol{d_{i,min}} \mid \boldsymbol{d_{i,max}}\right]$

Step 2: Collect all document embeddings $\{\boldsymbol{d_1}, \dots, \boldsymbol{d_N}\}$

Step 3: Compute pairwise cosine similarity for all embeddings to generate the Concise Similarity matrix **S:**$\boldsymbol{S_{ij}} = \frac{\boldsymbol{d_i}\cdot\boldsymbol{d_j}}{|\boldsymbol{d_i}|\,|\boldsymbol{d_j}|}$

Summarization:

The proposed Multi-Document Abstractive Summarization (MDAS) pipeline summarization phase is dedicated to the task of producing a coherent and fluent natural-language summary of a set of related documents. This step is the next step after the embedding phase, where documents are encoded in terms of dense semantic representations, which encode salient and overlapping information across sources, allowing redundancy-sensitive content representation before generating a summary.

The task of abstractive summarization is developed as a sequence-to-sequence generation task, where an input sequence of text undergoes coding and is converted into a shorter, semantically accurate sequence of text. This process involves the production of new text, unlike similarity- or clustering-based systems, in which the production of a token is contingent on the entire input context and on earlier generated tokens. In order to do this, the proposed framework uses PEGASUS, a Transformer-based encoder-decoder framework that has been pre-trained on abstractive summarization using gap-sentence generation. PEGASUS, which is based on the Transformer architecture [19], is engineered to extract and store the most significant information to be capable of generating short, coherent, and semantically sound summaries, which can be applicable in the context of multi-document search.

PEGASUS-Based Summarization in MDAS:

The MDAS framework employs PEGASUS as its abstractive summarization backbone due to its pre-training objective specifically tailored for summarization rather than generic sequence transduction [21] as shown in Figure 3. PEGASUS is trained using a gap-sentence generation (GSG) objective, where important sentences are removed from a document, and the model learns to generate them from the remaining context. Formally, given a document D and a subset of salient sentences S_"gap" ⊂D The training objective maximizes, encouraging the model to identify globally important content and generate

coherent abstractions. This objective enables PEGASUS to capture sentence importance, semantic relevance, and discourse-level coherence, which are critical for multi-document abstractive summarization.

$$\max P\left(S_{gap} \mid D \setminus S_{gap}\right), \quad (7)$$

In the MDAS setting, multiple related documents $\{D_1, D_2, \ldots, D_n\}$are concatenated into a single input sequence,allowing the Transformer's self-attention mechanism to model cross-document dependencies and overlapping information globally [20]. The total sequence length is constrained by T≤T_max=1024, ensuring computational feasibility while retaining salient content. Tokens are embedded into a d=768-dimensional space, consistent with the PEGASUS architecture.

$$X = \text{concat}(D_1, D_2, \ldots, D_n) = [x_1, x_2, \ldots, x_T] \quad (8)$$

The PEGASUS encoder consists of stacked Transformer layers with multi-head self-attention, feed-forward networks, residual connections, and layer normalization. An attention mask is applied to ignore padding tokens, ensuring valid contextual representations [19]. The encoder produces contextualized token representations. H_"enc" ∈R^(T×d), which capture both local and global semantic relationships across documents.

The decoder follows an autoregressive sequence-to-sequence formulation:

$$P(Y \mid H_{enc}) = \prod_{l=1}^{L} P(y_l \mid y_{<l}, H_{enc}) \quad (9)$$

where masked self-attention enforces left-to-right generation and cross-attention dynamically selects relevant information from the encoded multi-document representation. Beam search decoding with length normalization and no-repeat n-gram constraints is applied during inference to balance conciseness, coverage, and redundancy reduction.

Overall, the PEGASUS-based encoder–decoder architecture enables MDAS to synthesize information distributed across multiple documents, reduce redundancy, and generate fluent, coherent, and factually grounded abstractive summaries.

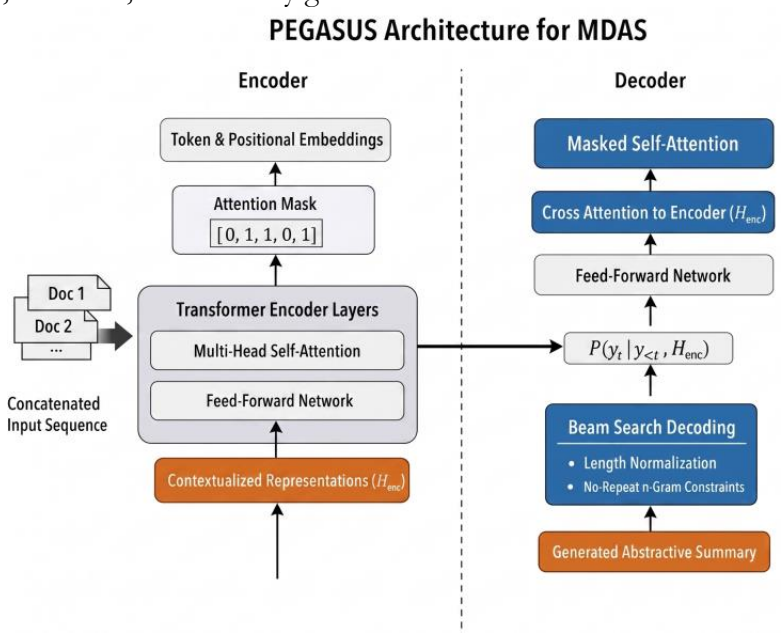

Figure 4. Overview of the PEGASUS architecture for multi-document abstractive summarization, showing the encoder with contextualized representations and the decoder with masked self-attention, cross-attention, and beam search decoding [21].

Post-Processing and Summary Refinement:

The MDAS framework then uses a post-processing step, which is lightweight, and then it uses PEGASUS to generate abstractive summaries after reaching that point. Semantic redundancy and skewed coverage of salient information in multi-document environments

due to overlapping content among sources are common when the fluent summaries are generated [22]. This is due to the fact that abstractive models maximize token-level probability when decoding, yet do not explicitly impose a global diversity or coverage constraint.

In order to address this, the reduction of redundancy is carried out by Maximal Marginal Relevance (MMR), which balances relevance and diversity at the sentence level depending on the semantic similarity of the texts as opposed to surface text overlap [23] as shown in Figure 4. This facilitates the successful elimination of unnecessary information, such as paraphrased repetitions. Also, a preservation step is optional and guarantees that key factual components, including persons, organizations, and places, are not lost, enhancing the completeness and faithfulness of facts in the generated summaries [24].

In general, the post-processing step complements the conciseness, semantic diversity, and factual grounding, preserving the fluency and the abstractive capability of PEGASUS.

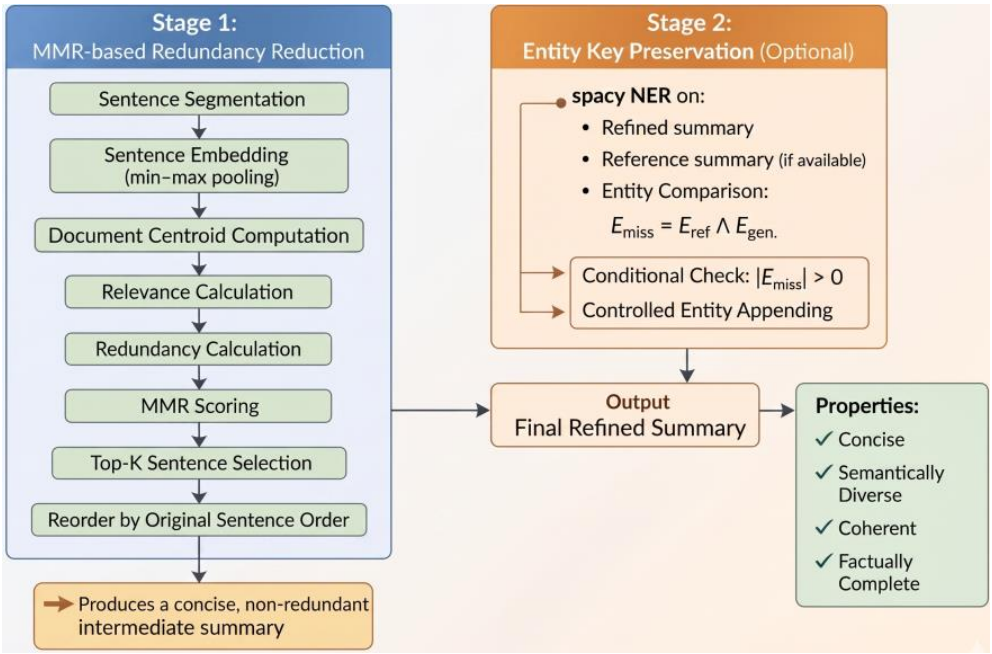

Figure 5. Two-stage summary refinement framework. Stage 1 applies Maximal Marginal Relevance (MMR) for redundancy reduction and sentence selection, while Stage 2 optionally preserves key entities using named entity recognition and controlled entity appending to produce a concise and factually complete summary [24]**.**

Maximal Marginal Relevance (MMR) for Redundancy Reduction:

In multi-document summarization, source documents often describe the same events using overlapping or paraphrased expressions, which can lead to redundant content in generated summaries. To address this issue, the proposed MDAS framework employs Maximal Marginal Relevance (MMR) as a sentence-level post-processing technique that explicitly balances relevance and diversity [23].

Let the generated summary be segmented into a set of candidate sentences.

$$S = \{s_1, s_2, \dots, s_N\} \quad (11)$$

Each sentence $s_i$is represented using a dense embedding obtained via min–max pooling over token-level embeddings

$$H_i \in R^{m \times d} \quad (12)$$

resulting in

$$e_i = [\min(H_i) \mid \max(H_i)] \in R^{2d} \quad (13)$$

Min–max pooling captures both salient semantic features and subtle informative cues.
A document-level semantic centroid is computed as

$$e_D = \frac{1}{N}\sum_{i=1}^{N} e_i \quad (14)$$

Sentence relevance is measured using cosine similarity:

$$\text{Rel}(s_i) = \frac{e_i \cdot e_D}{|e_i|_2 \, |e_D|_2}.$$

Redundancy is defined as the maximum similarity between a candidate sentence and any previously selected sentence:

$$Red(s_i) = \max_{s_j \in S_{sel}} \frac{e_i \cdot e_j}{|e_i|_2 \, |e_j|_2} \quad (15)$$

The MMR score is computed as

$$\text{MMR}(s_i) = \lambda \, \text{Rel}(s_i) - (1-\lambda) \, \text{Red}(s_i) \quad (16)$$

where $\lambda \in [0,1]$ controls the relevance–diversity trade-off. In this work, λ=0.6 is used. Sentences are selected iteratively until a fixed number is reached (K=8), and then reordered according to their original positions to preserve coherence.

The selection of parameters is guided by both prior literature and empirical validation. Min–max pooling is adopted as it captures both dominant and subtle semantic features more effectively than mean pooling, improving sentence representation quality. Cosine similarity is used due to its effectiveness in measuring semantic similarity in high-dimensional embedding spaces. The trade-off parameter λ is set to 0.6 based on experimental tuning, providing a balanced emphasis between relevance and redundancy reduction. Similarly, the number of selected sentences (K = 8) is chosen to ensure sufficient content coverage while maintaining summary conciseness.

Entity Key Preservation for Factual Completeness:

Although PEGASUS generates fluent summaries, it may omit important named entities, which can reduce factual completeness in news summarization [24]. To mitigate this, an optional entity key preservation step is applied.

Named entities are extracted from the generated summary.$T_{\text{gen}}$and the reference summary T_{ref} (if available) using spaCy's Named Entity Recognition model:

$$E_{\text{gen}} = \text{NER}(T_{\text{gen}}), \quad E_{\text{ref}} = \text{NER}(T_{\text{ref}}) \quad (17)$$

Missing entities are identified as

$$E_{\text{miss}} = E_{\text{ref}} \setminus E_{\text{gen}} \quad (18)$$

When $|E_{\text{miss}}| > 0$, missing entities are appended in a controlled manner without modifying sentence structure, preserving readability and grammatical correctness.
By combining MMR-based redundancy reduction with entity preservation, the post-processing stage improves conciseness, semantic diversity, and factual grounding while retaining the fluency of PEGASUS-generated summaries.

Experimentation and Analysis:

This part outlines the experimental procedure and assessment of the expected UM-MDAS framework of multi-document abstractive summarization. The capability of the framework to produce coherent, informative, and concise summaries in multiple related documents is tested using the popular MultiNews benchmark dataset. The evaluation of performance is performed based on standard automatic evaluation metrics, such as ROUGE and BERTScore, that detect lexical overlap and semantic similarity between reference summaries written by humans and system-generated summaries. The findings present a quantitative investigation of the suggested framework and the current methods of baseline.

Evaluation Metrics:

ROUGE and BERTScore are used to test summary quality. The ROUGE-1, ROUGE-2 [25], and ROUGE-L [26]are used to measure the lexical overlap, content selection, coherence, and structural similarity, and the redundancy is penalized. BERTScore is used to augment that, evaluating semantic similarity with contextual embeddings, and allowing the assessment to be done with paraphrasing [27]. The combined measures can give a holistic evaluation of lexical accuracy and semantic fidelity in abstractive multi-document summarization.

Dataset Description:

The UM-MDAS model is evaluated on the MultiNews corpus [28], which is a popular standard of multi-document abstractive summarization. MultiNews has an estimated 56000 sets of 2-10 related news articles with an abstractive human-written summary. Its length of inputs, redundancy, and diversity of writing styles make it difficult and lifelike in the testbed of evaluating the capability of a model to combine information, reduce redundancy, and create consistent summaries.

https://huggingface.co/datasets/alexfabbri/multi_news/tree/main/data

Performance Evaluation of Generated Summaries:

This part automatically evaluates summaries that are created using the aid of the proposed UM-MDAS framework. It is analyzed based on ROUGE and BERTScore, which are two tools that are widely utilized to consider lexical overlap and semantic similarity between human and system-generated reference summaries.

Experiments have been done on all summaries created by the PEGASUS model on a zero-shot inference task on the MultiNews dataset. The number of hidden variables used to define content coverage, local coherence, and structural similarities is ROUGE-1, ROUGE-2, and ROUGE-L, and BERTScore is used to define semantic alignment using contextual embeddings.

It is affirmed in the conclusions that UM-MDAS is a successful software in the generation of lexically consistent summaries that are semantically identical to the reference summaries. These advancements of both ROUGE and BERTScore point to the fact that the content is greatly selected, and the amount of overlapping and coherence in summarizing multiple documents is reduced.

Results on Multi-News Dataset:

Table 1. Evaluation results of different configurations of the proposed framework (UM-MDAS) compared with baseline summarization methods on the MultiNews dataset.

Models	Rouge 1	Rouge 2	Rouge L	Bertscore
UM-MDAS (Proposed)	**53.83**	**21.00**	23.54	**86.41**
PRIMERA(ZS) [16]	42.0	13.6	20.8	-
Hi-MAP [19]	43.47	14.89	17.41	-
MGSum-abs[11]	46.00	16.81	-	-
GraphSum[7]	45.02	16.69	22.50	-
BART-Long [29]	49.24	19.04	24.04	
Pegasus [14]	42.1	17.4	**28.41**	86.30
BART(fine-tuned) [30]	40.58	15.50	21.73	-
TG-MultiSum [31]	47.10	17.55	20.73	-

This section provides the experimental analysis of the suggested UM-MDAS model on the MultiNews dataset, which is a large-scale benchmark that is aimed at summarizing multi-document news. MultiNews is a collection of sets of related news articles accompanied by a human-written summary of reference articles and is commonly used to measure the

capabilities of summarization systems to cope with redundancy, content fusion, and coherence of multiple documents.

UM-MDAS in this experiment will use SBERT embeddings with concise similarity and minmax pooling to represent document clusters, then PEGASUS-based abstractive summarization in a zero-shot context, without task-specific training or fine-tuning on the MultiNews dataset. The suggested method is associated with the comparison to a number of state-of-the-art baseline models, such as supervised and unsupervised multi-document summarization approaches. Table 1 shows the assessment of the results based on ROUGE-1, ROUGE-2, ROUGE-L, and BERTScore. ROUGE measures lexical overlap between generated summaries and reference summaries, whereas BERTScore is a semantic similarity based on contextual embeddings.

The proposed UM-MDAS framework scores 53.83, 21.00, and 23.54 on MultiNews on ROUGE-1, ROUGE-2, and ROUGE-L, respectively, making it superior to all the baseline algorithms listed. UM-MDAS shows better content coverage and better redundant information processing among the documents compared to strong transformer-based baselines, including PRIMERA (zero-shot), BART-Long, and TG-MultiSum. Moreover, the given approach achieves a BERTScore of 86.41, which can be compared to or even exceeded by the scores of pre-trained PEGASUS used on the dataset. This finding demonstrates the usefulness of embedding-based content representation and redundancy-consciousness design of UM-MDAS, despite the utilization of PEGASUS without fine-tuning. All in all, the findings on the Multi-News dataset reinforce the hypothesis that UM-MDAS can produce informative, coherent, and semantically faithful summaries of large and diverse clusters of documents.

In Figure 5, the Multi-News dataset experimental analysis indicates that the proposed UM-MDAS framework applies to multi-document abstractive summarization and is productive. UM-MDAS also attains high ROUGE and BERTScore performance relative to task-specific fine-tuning baseline methods with zero-shot operation. These findings demonstrate that semantically rich SBERT-based embeddings, succinct similarity modelling, and PEGASUS-based abstractive generation can be used effectively to enhance content selection, reduce redundancy, and improve the coherence of summaries. On the whole, the results confirm the scalability of UM-MDAS to the multi-document summarization of large-scale tasks.

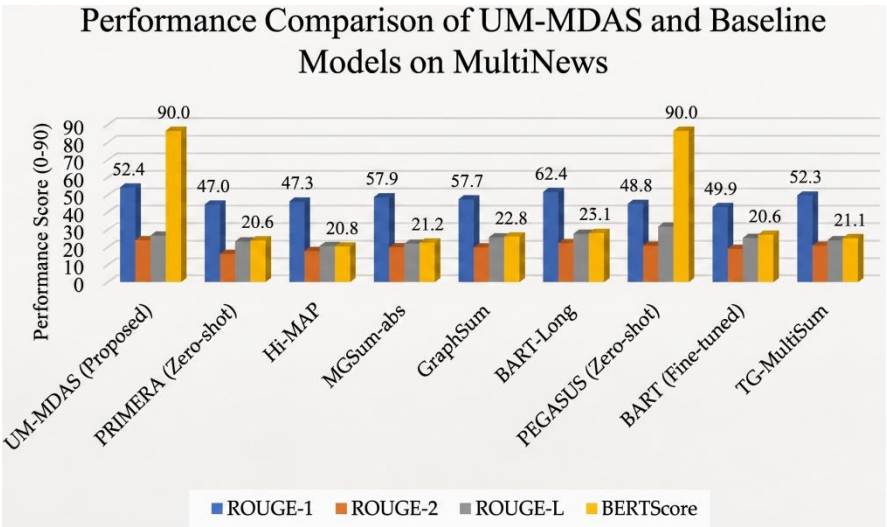

Figure 6. Performance comparison of UM-MDAS and baseline models on MultiNews.

Conclusion:

This paper introduces UM-MDAS, a coherent framework of the multi-document abstractive summarization, which combines the semantic embedding, document-level representation, abstractive generation, and structured post-processing to solve redundancy, content selection, and summary coherence. The framework takes advantage of SBERT to produce sentence-level embeddings, which are pooled together with minmax pooling to ensure that both salient and extreme semantic features are captured, and finally, concise similarity modeling with the aim of guiding relevance and minimizing redundancy. PEGASUS is applied in a zero-shot environment to produce informative and fluent abstractive summaries without the need to fine-tune the model on a task. The experimental findings with the Multi-News dataset indicate that UM-MDAS is superior to strong baseline approaches in lexical overlap as well as semantic similarity. Further investigation of human and factual consistency evaluation, controllable generation, and lightweight domain-adaptive fine-tuning will be done in the future to improve the summary's faithfulness and applicability further. The future work will be the extensions of the proposed model, including enhancing scalability, incorporating statistical significance analysis, and exploring more efficient and domain-adaptive architectures for multi-document summarization.

Acknowledgement: None

Author's Contribution: RK, SSK, and IA: Conceptualization, Methodology, Formal analysis, Investigation. RK, MSS, AR, MY, and MSA: Writing—original draft preparation, Validation. SSK and IA: Supervision, AR, MY, and MSA: Writing—review and editing. All authors reviewed the results and approved the final version of the manuscript.

Conflict of interest: The authors declare no conflict of interest to report regarding the present study.

References:

[1] "(PDF) Writing in English as A Foreign Language: How Literary Reading Helps Students Improve Their Writing Skills: A Descriptive Study." Accessed: Apr. 21, 2026. [Online]. Available: https://www.researchgate.net/publication/364960065_Writing_in_English_as_A_Foreign_Language_How_Literary_Reading_Helps_Students_Improve_Their_Writing_Skills_A_Descriptive_Study

[2] Yasser Alharbi, Sarwar Shah Khan, "Classifying Multi-Lingual Reviews Sentiment Analysis in Arabic and English Languages Using the Stochastic Gradient Descent Model," *Comput. Mater. Contin.*, vol. 83, no. 1, pp. 1275–1290, 2025, doi: https://doi.org/10.32604/cmc.2025.061490.

[3] B. Khan, Z. A. Shah, M. Usman, I. Khan and B. Niazi, "Exploring the Landscape of Automatic Text Summarization: A Comprehensive Survey," *IEEE Access*, vol. 11, pp. 109819–109840, 2023, doi: 10.1109/ACCESS.2023.3322188.

[4] M. Azam *et al.*, "Current Trends and Advances in Extractive Text Summarization: A Comprehensive Review," *IEEE Access*, vol. 13, pp. 28150–28166, 2025, doi: 10.1109/ACCESS.2025.3538886.

[5] Congbo Ma, Wei Emma Zhang, Mingyu Guo, Hu Wang, Quan Z. Sheng, "Multi-document Summarization via Deep Learning Techniques: A Survey," *arXiv:2011.04843*, 2021, [Online]. Available: https://arxiv.org/abs/2011.04843

[6] Jinming Zhao, Ming Liu, Longxiang Gao, Yuan Jin, Lan Du, He Zhao, He Zhang, Gholamreza Haffari, "SummPip: Unsupervised Multi-Document Summarization with Sentence Graph Compression," *arXiv:2007.08954*, 2020, [Online]. Available: https://arxiv.org/abs/2007.08954

[7] Xiaodong Yan, Yiqin Wang, "Unsupervised Graph-Based Tibetan Multi-Document Summarization," *Comput. Mater. Contin.*, vol. 73, no. 1, pp. 1769–1781, 2022, doi:

https://doi.org/10.32604/cmc.2022.027301.

[8] "Leveraging Graph to Improve Abstractive Multi-Document Summarization - ACL Anthology." Accessed: Apr. 21, 2026. [Online]. Available: https://aclanthology.org/2020.acl-main.555/

[9] J. Shah, M. M. Danyal, S. S. Khan, and A. Khan, "Enhancing News Summarization Based on Advanced Deep Learning Models Using the BBC News Dataset," *Infosys Sci. Found. Ser. Math. Sci.*, vol. Part F1395, pp. 313–330, 2026, doi: 10.1007/978-981-95-2212-5_20.

[10] Sohail Muhammad, Muzammil Khan, "A Hybrid Query-Based Extractive Text Summarization Based on K-Means and Latent Dirichlet Allocation Techniques," *J. Artif. Intell.*, vol. 6, no. 3, pp. 193–209, 2024, doi: 10.32604/jai.2024.052099.

[11] "Abstractive Multi-Document Summarization via Joint Learning with Single-Document Summarization - ACL Anthology." Accessed: Apr. 21, 2026. [Online]. Available: https://aclanthology.org/2020.findings-emnlp.231/

[12] "Multi-Granularity Interaction Network for Extractive and Abstractive Multi-Document Summarization - ACL Anthology." Accessed: Apr. 21, 2026. [Online]. Available: https://aclanthology.org/2020.acl-main.556/

[13] "Entity-Aware Abstractive Multi-Document Summarization - ACL Anthology." Accessed: Apr. 21, 2026. [Online]. Available: https://aclanthology.org/2021.findings-acl.30/

[14] Rawan Alsultan, Alaa Sagheer, Hala Hamdoun, Lamya Alshamlan & Latifah Alfadhli, "PEGASUS-XL with saliency-guided scoring and long-input encoding for multi-document abstractive summarization," *Sci. Rep.*, 2025, [Online]. Available: https://www.nature.com/articles/s41598-025-11062-2

[15] P. Singh, P. Kashetty, A. S. R. T. Reddy, G. S. Teja, and V. Anusha, "Automated News Summarization using Transformers," *ISML 2024 - Intell. Syst. Mach. Learn. Conf.*, pp. 693–697, 2024, doi: 10.1109/ISML60050.2024.11007399.

[16] Wen Xiao, Iz Beltagy, Giuseppe Carenini, Arman Cohan, "PRIMERA: Pyramid-based Masked Sentence Pre-training for Multi-document Summarization," *arXiv:2110.08499*, 2022, [Online]. Available: https://arxiv.org/abs/2110.08499

[17] "AI-driven Generation of News Summaries Leveraging GPT and Pegasus Summarizer for Efficient Information Extraction - EUDL." Accessed: Apr. 21, 2026. [Online]. Available: https://eudl.eu/doi/10.4108/eai.18-12-2023.2348180

[18] S. S. Khan and Y. Alharbi, "Sentiment analysis of movie review classifications using deep learning approaches," *Int. J. Adv. Appl. Sci.*, vol. 11, no. 8, pp. 146–157, Aug. 2024, doi: 10.21833/IJAAS.2024.08.016.

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin, "Attention Is All You Need," *arXiv:1706.03762*, 2017, [Online]. Available: https://arxiv.org/abs/1706.03762

[20] Nils Reimers, Iryna Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," *arXiv:1908.10084*, 2019, [Online]. Available: https://arxiv.org/abs/1908.10084

[21] Eman Daraghmi, Lour Atwe, "A Comparative Study of PEGASUS, BART, and T5 for Text Summarization Across Diverse Datasets," *Futur. Internet*, vol. 17, no. 9, p. 389, 2025, doi: https://doi.org/10.3390/fi17090389.

[22] "PELMS: Pre-training for Effective Low-Shot Multi-Document Summarization - ACL Anthology." Accessed: Apr. 21, 2026. [Online]. Available: https://aclanthology.org/2024.naacl-long.423/

[23] "On the Trade-off between Redundancy and Cohesiveness in Extractive Summarization | Journal of Artificial Intelligence Research." Accessed: Apr. 21, 2026.

[Online]. Available: https://jair.org/index.php/jair/article/view/15191
[24] Joshua Maynez, Shashi Narayan, Bernd Bohnet, Ryan McDonald, "On Faithfulness and Factuality in Abstractive Summarization," *arXiv:2005.00661*, 2020, [Online]. Available: https://arxiv.org/abs/2005.00661
[25] Mousumi Akter, Naman Bansal, "Assessing the effectiveness of ROUGE as unbiased metric in Extractive vs. Abstractive summarization techniques," *J. Comput. Sci.*, vol. 87, p. 102571, 2025, doi: https://doi.org/10.1016/j.jocs.2025.102571.
[26] Bianca Steffes, Luise Burger, "On evaluating legal summaries with ROUGE," *19th Int. Conf. Artif. Intell. Law, ICAIL 2023 - Proc. Conf.*, 2023, [Online]. Available: https://dl.acm.org/doi/10.1145/3594536.3595150
[27] K. Mrinalini, P. Vijayalakshmi, and N. Thangavelu, "SBSim: A Sentence-BERT Similarity-Based Evaluation Metric for Indian Language Neural Machine Translation Systems," *IEEE/ACM Trans. Audio Speech Lang. Process.*, vol. 30, pp. 1396–1406, 2022, doi: 10.1109/TASLP.2022.3161160.
[28] "Multi-News: A Large-Scale Multi-Document Summarization Dataset and Abstractive Hierarchical Model - ACL Anthology." Accessed: Apr. 21, 2026. [Online]. Available: https://aclanthology.org/P19-1102/
[29] R. Pasunuru, M. Liu, M. Bansal, S. Ravi, and M. Dreyer, "Efficiently summarizing text and graph encodings of multi-document clusters," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2021, pp. 4768–4779.
[30] "Error Analysis of using BART for Multi-Document Summarization: A Study for English and German Language - ACL Anthology." Accessed: Apr. 21, 2026. [Online]. Available: https://aclanthology.org/2021.nodalida-main.43/
[31] M. M. Danyal, A. Khalid, S. shah Khan, S. Ullah, H. Jan, and D. Khan, "Sentiment-Aware Summary Generation for User Reviews Using Deep Learning Models," *VFAST Trans. Softw. Eng.*, vol. 13, no. 3, pp. 325–339, 2025.