VNU Journal of Science: Com. Science & Com. Eng., Vol. 42, No. 2 (2026) 64–86

VNU Journal of Science:
Computer Science and Communication Engineering

Journal homepage: http://www.jcsce.vnu.edu.vn/index.php/jcsce

Original Article

Improving Biomedical Multi-document Abstractive Summarization Model with Syntax Tree Pruning and Generative Pre-training Adaptation

Thanh-Tam Doan[1], Tu-Phuong Mai[2], Quoc-Hung Duong[2], Duy-Cat Can[2], Thi-Hai-Yen Vuong[2], Mai-Vu Tran[2*]

[1]*Viettel Group, Hanoi, Vietnam*
[2]*VNU University of Engineering and Technology, Hanoi, Vietnam*

Received 17th September 2025
Revised 22nd February 2026; Accepted 18th May 2026

Abstract: Biomedical multi-answer summarization presents critical challenges for healthcare applications, where standard transformer-based models face input length limitations, factual inconsistency risks, and inadequate query-driven content selection mechanisms. We propose SAMSum, a Syntax-aware Adaptive Transformer-based Model for MAS, which integrates three innovative components to address these fundamental limitations. Our approach combines an adaptive BART architecture with extractive preprocessing to mitigate information loss, query-conditioned formatting to ensure medical question relevance, and dynamic length prediction for optimal information density. A syntax tree pruning mechanism employs supervised gradient boosting classification to systematically eliminate redundant phrases while preserving medical content integrity and grammatical structure. Comprehensive evaluation on the MEDIQA-MAS 2021 dataset demonstrates that SAMSum achieves state-of-the-art performance across all evaluation metrics, with ROUGE-2 F1 score of 17.3% and BERTScore F1 of 66.8%, substantially outperforming existing baselines and challenging participants. Data and code are available at: https://github.com/candleMind/SAMSum.

Keywords: biomedical text summarization, multi-answer summarization, syntax tree pruning, abstractive summarization, generative models, healthcare systems, clinical decision support

*Corresponding author.
E-mail address: vutm@vnu.edu.vn

https://doi.org/10.25073/2588-1086/vnucsce.5797

1. Introduction

Biomedical question-answering systems face a fundamental challenge in synthesizing information from multiple sources to provide comprehensive and accurate responses to complex medical queries. Multi-answer summarization (MAS) targets aggregation of multiple relevant answers to a biomedical question into one concise and relevant answer, addressing the critical need to combine diverse perspectives and complementary information from various medical sources [1]. However, different answers can bring complementary perspectives that are likely to benefit the users of Question-answeing (QA) systems, making MAS particularly valuable in clinical decision support and patient education systems where comprehensive information synthesis is essential.

The biomedical domain presents unique challenges for traditional summarization approaches. Current transformer-based models, while demonstrating remarkable performance in general-domain summarization, face three critical limitations when applied to biomedical MAS. First, standard abstractive models like BART and PEGASUS are constrained by input length limitations, often requiring truncation of multiple biomedical documents that results in substantial information loss [2]. Domain-specific datasets are scarce for training robust biomedical summarization models, while text summarization is a difficult problem and always attracts attention from the research community, especially working on biomedical text data which lacks supporting tools and techniques. Second, these models frequently generate fluent but factually inconsistent or medically inappropriate content, a critical concern in healthcare applications where accuracy directly impacts patient safety. Third, existing approaches lack effective mechanisms for query-driven content selection, often producing generic summaries that fail to address the specific information needs embedded in medical questions.

To address these fundamental challenges, we propose SAMSUM (**S**yntax-aware **A**daptive Transformer-based Model for **M**ulti-answer **Sum**marization), a novel model that bridges the gap between extractive reliability and abstractive fluency through three main contributions:

- The *adaptive BART* component employs transformer-based generation enhanced with extractive preprocessing for computational efficiency, query-conditioned formatting for relevance, and dynamic length prediction for optimal information density.
- The *syntax tree pruning* stage applies gradient boosting classification to eliminate redundant phrases while preserving medical content integrity through supervised learning that leverages dataset-specific characteristics.
- In comprehensive evaluation on the MEDIQA-MAS 2021 dataset, SAMSUM achieves *state-of-the-art performance* across multiple metrics, validating the effectiveness of our integrated methodology.

2. Related Work

This section contextualizes our research within the landscape of automatic text summarization, with emphasis on biomedical multi-document abstractive summarization.

Extractive Summarization. Extractive summarization has evolved from frequency-driven methods (TF-IDF) and graph-based algorithms (LexRank) to contemporary neural approaches, significantly enhancing biomedical document processing [3]. Contemporary neural approaches use transformer-based architectures with hierarchical encoding to handle long documents, while hybrid frameworks combining neural topic models and graph neural networks refine sentence selection through semantic relationships [4]. Transformer-based extractive models now incorporate hierarchical encoding

with domain-adaptive attention mechanisms specifically designed for long clinical documents [5]. However, these complicated encoding models prevent fast inference ability, especially when the extractive summarization phase is solely the middle stage of a hybrid model.

Abstractive Summarization. Abstractive methods have progressed from early encoder-decoder models to transformer-based frameworks (BART, PEGASUS) and large language models (GPT-3, T5) that leverage generative pretraining [6]. Recent work demonstrates that modern LLMs exhibit significant memorization of medical training data, raising concerns about factual consistency and privacy in clinical applications [7]. Multi-step reasoning frameworks have emerged to enhance LLM performance in complex medical question-answering tasks, employing chain-of-thought prompting and self-consistency mechanisms [8]. Retrieval-Augmented Generation (RAG) architectures ground generation in external knowledge sources, with recent biomedical implementations incorporating dynamic retrieval strategies and knowledge distillation for improved efficiency [9, 10]. Multi-modal approaches now extend beyond text to integrate whole-slide imaging and clinical notes for comprehensive patient summarization [11]. Hybrid frameworks leverage extractive modules for content selection coupled with abstractive components for surface realization, balancing factual precision with linguistic quality [12].

Biomedical QA Summarization. Biomedical summarization faces domain-specific challenges, including terminology precision, evidence synthesis, and query-driven content selection. MEDIQA-AnS systems typically process retrieved answers to consumer health questions, demanding question-based multi-document summarization with high factual accuracy. Approaches incorporate knowledge graphs and multi-stage abstraction, outperforming conventional methods by explicitly modeling semantic connections [13]. Knowledge graph integration also enables explicit modeling of medical relationships, with graph-based neural networks outperforming sequence-only models on multi-document biomedical summarization [14]. Recent benchmarks systematically evaluate LLM performance across clinical reasoning tasks, revealing significant gaps in specialized medical knowledge despite strong general capabilities [15]. Domain-adaptive pretraining on biomedical corpora (PubMed, MIMIC-III) improves terminology handling, with fine-tuned medical LLMs demonstrating superior performance on clinical summarization benchmarks [16, 17].

Phrase Pruning Techniques. Phrase pruning reduces redundancy through syntactic or learned compression strategies. Tree-based approaches leverage grammatical structures for deterministic subtree removal [18]. Contemporary syntax-aware frameworks integrate constituency parsing with semantic role labeling, enabling context-sensitive removal of non-essential phrases. Biomedical adaptations incorporate UMLS-based node importance scoring, retaining clinically relevant concepts during compression [4]. On the other hand, learning-based methods using neural network classifiers for sequence labeling [12]. Neural networks such as attention mechanisms enable soft pruning through differentiable masking, allowing gradient-based optimization of compression rates [19]. However, most pruning techniques operate independently of the generation process, limiting their ability to adapt to query-specific information needs.

Research Gaps. Despite significant progress, current biomedical summarization systems exhibit three critical limitations. First, they lack integrated frameworks that balance extractive efficiency with abstractive fluency while

maintaining medical accuracy. Second, most approaches do not adequately incorporate query-driven content selection throughout the summarization pipeline. Third, existing methods rarely address optimal summary length determination, leading to either information loss or excessive redundancy. Our work proposes an adaptive Transformer-based summarization phase that incorporates question-driven encoding and length optimization, using extractive pre-processing to remove redundancy and improve the input quality. While tree-based methods ensure grammaticality, learning approaches better handle domain shifts through representation learning. Our work bridges this divide through syntax-informed neural pruning optimized for clinical discourse.

3. Proposed Model

Our proposed model integrates deep learning-based abstractive summarization with syntax-aware refinement to enhance coherence and informativeness in multi-document biomedical summarization. The framework, illustrated in Figure 1, comprises four sequential components that progressively transform raw documents into polished, query-focused summaries.

The *pre-processing* stage normalizes biomedical documents and extracts domain-specific information through ScispaCy tokenization, BioBERT embeddings, and named entity recognition to ensure clean, structured input. The *adaptive BART* component employs transformer-based generation enhanced with extractive preprocessing for computational efficiency, query-conditioned formatting for relevance, and dynamic length prediction for optimal information density. The *syntax tree pruning* stage applies gradient boosting classification to eliminate redundant phrases while preserving medical content integrity. Finally, *post-processing* ensures fluency, readability, and factual consistency in the generated summaries.

3.1. Pre-processing

To improve the quality of biomedical query-based multi-document summarization and ensure that the input documents are cleaned, structured, and enriched with domain-specific information, we employ a three-step pre-processing pipeline:

- *Normalization*: We remove HTML artifacts, redundant whitespace, and enumerated list inconsistencies to prevent fragmented or incoherent sentences.
- *Segmentation*: Using ScispaCy [20], we tokenize biomedical texts, extract POS tags, lemmatized forms, and stopwords, which are later used to enhance relevance-based summarization.
- *Augmentation*: Named Entities Recognition (NER) is done to perceive named entities, especially the biomedical aspects like medicines, diseases, treatments, and biomedical abbreviations. The keywords of sentences and the question are simply extracted by picking lemmatized words that are not stopwords and tagged as nouns or verbs. BioBERT embeddings [21] are also integrated to prioritize medically relevant content, computing query Q relevance via cosine similarity.

3.2. Adaptive BART Abstractive Summarization

This subsection presents our adaptive approach to BART-based abstractive summarization, addressing three critical challenges in biomedical multi-document summarization: input length limitations, query-driven content selection, and optimal summary length determination. Our method enhances the standard BART architecture through extractive summary input to manage computational constraints, query-conditioned input formatting to ensure relevance, and dynamic length prediction to optimize information density. The input consists of multiple biomedical

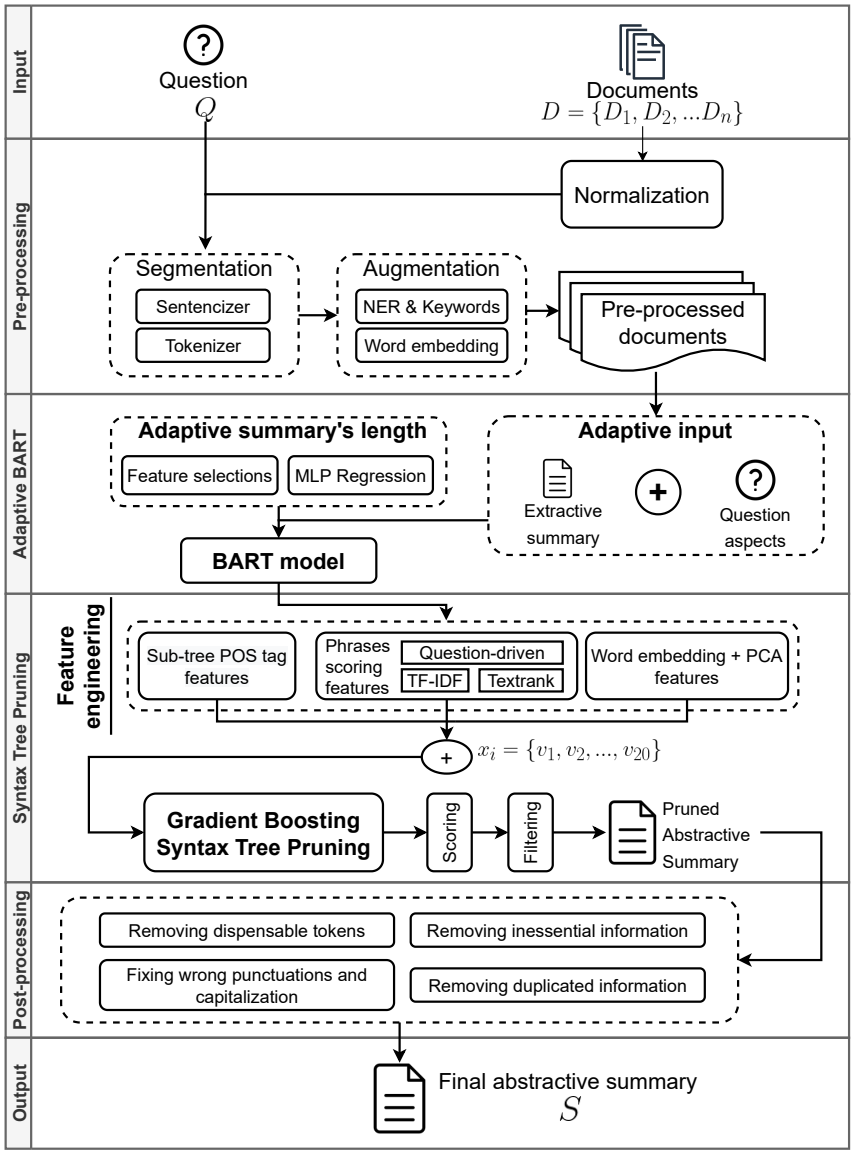


Figure 1. Overview of the Proposed Multi-Document Abstractive Summarization Model.

documents and a query, while the output is a coherent, query-focused abstractive summary of appropriate length.

3.2.1. Extractive Summary Input

Direct encoding of lengthy biomedical documents exceeds BART's 1024-token limit and introduces substantial noise that degrades generation quality. To address computational constraints and select the most informative input partials, we implement an extractive summarization stage that generates a concise summary by selecting top-ranked sentences through a weighted combination of complementary scoring mechanisms. This approach preserves essential information while reducing hallucinations and computational overhead. Our sentence selection mechanism combines three scoring approaches to capture different aspects of content importance.

TF-IDF score. The TF-IDF score [22] quantifies statistical term importance within the document collection. Given a set of documents $D = \{d_1, d_2, ..., d_n\}$ and current document $d = \{s_1, s_2, ..., s_m\}, d \in D$, the score of sentence s_i is calculated by:

$$\text{tf-idf}(s_i) = \sum_{w \in s} \text{tf}(w, d) \times \text{idf}(w, D) \quad (1)$$

where Term Frequency (TF) and Inverse Document Frequency (IDF) are defined as:

$$\text{tf}(w, d) = \frac{f(w, d)}{max\{f(t, d) : t \in d\}} \quad (2)$$

$$\text{idf}(w, D) = \log \frac{|D|}{|\{d \in D : w \in d\}|} \quad (3)$$

To mitigate the bias toward lengthy sentences, we consider only unique word sets and apply thresholding to eliminate low-scoring terms.

LexRank score. The LexRank score [23] captures global sentence centrality through graph-based analysis, representing sentences as nodes connected by weighted edges based on semantic similarity. The centrality score is computed iteratively:

$$p^0(u) = \frac{1}{N}, \forall u \quad (4)$$

$$p^t(u) = \frac{d}{N} + (1 - d) \sum_{v \in adj_u} \frac{\text{sim}(u, v)}{\sum_{z \in adj_v} \text{sim}(z, v)} p^{t-1}(v)$$

where N represents the sentence count, d is the damping factor (typically 0.1-0.2), and $\text{sim}(u, v)$ denotes cosine similarity between sentence embeddings. Iteration continues until convergence: $\left\|p^t - p^{t-1}\right\| < \varepsilon$.

Question-relevant score. The question-relevant score ensures query relevance by incorporating both semantic similarity and entity overlap. For sentence s_i and query Q, the score combines similarity and keyword matching:

$$\text{question}(s_i) = \lambda \text{sim}(s_i, Q) + (1 - \lambda)\text{keys}(s_i, Q) \quad (5)$$

where λ controls the contribution balance (default 0.5), $\text{sim}(s_i, Q)$ represents BioBERT embedding cosine similarity, and $\text{keys}(s_i, Q)$ measures entity and keyword overlap through Intersection over Union:

$$\text{keys}(s_i, Q) = \frac{[\text{Overlap}(s_i)] \cap [\text{Overlap}(Q)]}{\text{Overlap}(s_i) \cup \text{Overlap}(Q)} \quad (6)$$

$$\text{Overlap}(s) = \text{Entities}(s) \cup \text{Keywords}(s) \quad (7)$$

3.2.2. Question-driven BART Encoding

Standard BART generates generic summaries without considering specific information needs. To enable query-tailored summarization that prioritizes relevant medical content, we modify the input structure to explicitly condition generation on the provided query. This approach ensures that the model attends to query-relevant information while maintaining coherent abstractive generation.

We integrate query conditioning by prepending the query Q to the extracted summary using special separator tokens that enable the transformer to distinguish between query context and source content:

$$\texttt{[CLS]}\, Q\, \texttt{[SEP]}\, s_1\, \texttt{[SEP]}\, s_2 \ldots s_n\, \texttt{[SEP]} \quad (8)$$

This structured input format ensures that BART's attention mechanism considers query relevance throughout the generation process. The separator tokens enable the model to maintain distinct representations for query and content while allowing cross-attention between these components, resulting in summaries that directly address the specified information needs rather than providing generic overviews.

3.2.3. Adaptive Summary Length

Spontaneous summary lengths of BART generation often produce suboptimal results, with

short summaries missing critical information and lengthy ones introducing redundancy. To dynamically optimize summary length for maximum BART's length constraint based on content characteristics, we develop a regression model predicting optimal output length using document statistics and content density measures. This approach prevents over-truncation and excessive repetition while ensuring information completeness.

Table 1. Input features for adaptive summary length

Feature		Explain
Statistical feature	Sum of tokens	Accumulating the number of tokens of all documents
	Average tokens per document	Mean of the numbers of tokens per document
	Sum of sentences	Accumulating the number of sentences of all documents
	Average sentences per document	Mean of the numbers of sentences per document
	Sum of documents	The number of documents
Scoring feature	TF-IDF score	Mean of the sums of top 3 sentences with highest TF-IDF score in all documents
	Lexrank score	Mean of the sums of top 3 sentences with highest Lexrank score in all documents
	Question-relevant score	Mean of the sums of top 3 sentences with highest Question-relevant score in all documents

We train a Multi-Layer Perceptron (MLP) regression model to predict optimal summary length based on document features that correlate with information density and content complexity:

$$\hat{L} = f(\mathbf{x}) = \mathbf{W} \cdot \mathbf{x} + b \tag{9}$$

where $\mathbf{x}$ is the document's statistical feature vector, and $\hat{L}$ is the predicted summary length.

Table 1 presents the statistical features used to predict the optimal summary length, incorporating both statistical and scoring-based attributes. These features capture document structure, lexical importance, and query relevance, ensuring that length prediction aligns with content density and informativeness.

3.3. Syntax Tree Pruning

Neural abstractive summarization frequently generates redundant phrases and syntactically complex constructions that impede readability while providing minimal informational value. To enhance conciseness and improve the clarity of generated summaries, we implement a learning-based syntax tree pruning mechanism that systematically removes unnecessary phrases while preserving grammatical structure and medical content relevance. Our approach addresses the limitations of statistical pruning methods through supervised learning that leverages dataset-specific characteristics. The input consists of constituency parse trees from generated summaries, while the output comprises pruned summaries with improved syntactic quality and reduced redundancy. Figure 2 illustrates the training pipeline, where these features are extracted, processed, and used to optimize the pruning decision for enhanced summary quality.

Gradient Boosting Phrase Classification. Leveraging comprehensive feature representations, we develop a gradient boosting classification model that learns optimal pruning decisions from training data. Our approach transforms syntax tree pruning into a supervised learning problem where each phrase constituent in parsed summary trees serves as a classification instance.

The training dataset construction involves creating phrase-level records where each phrase p in the constituency parse tree corresponds

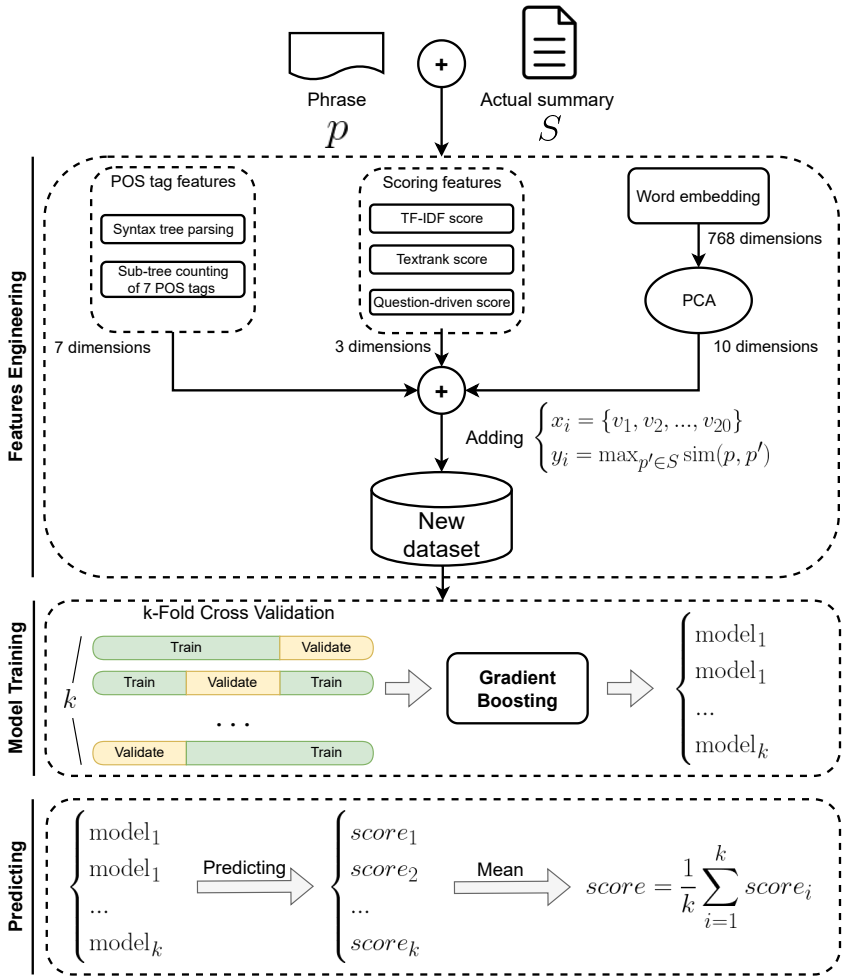


Figure 2. Gradient Boosting model for phrase scoring in syntax tree pruning.

to a training instance (x_i, y_i). The feature vector x_i represents linguistic and contextual characteristics of phrase p, while the target label y_i indicates phrase retention likelihood computed through similarity alignment with human reference summaries:

$$y_i = \max_{p' \in S} \text{sim}(p, p') \tag{10}$$

where p' represents reference phrases in the gold standard summary S, and $\text{sim}(p, p')$ denotes cosine similarity between phrase embeddings. This formulation enables the model to learn which phrase types contribute meaningfully to high-quality summaries.

Feature Engineering. For the model's inputs, our feature engineering approach combines multiple linguistic representation levels to capture phrase importance comprehensively. Table 2 lists the features used for gradient boosting-based phrase scoring. The feature set encompasses part-of-speech tag frequencies within phrase subtrees, capturing syntactic patterns that correlate with retention decisions. Lexical scoring features incorporate phrase-level importance measures derived from our extractive preprocessing stage. High-dimensional phrase embeddings provide semantic representations

that are dimensionality-reduced through Principal Component Analysis from 768 to 10 dimensions, preserving essential semantic information while maintaining computational efficiency.

Table 2. Gradient Boosting input features

Features	Explain	Dimension
Frequencies	The frequency of 7 tags including NP, VP, WH*, ADJP, ADVP, CONJ, PP	7
Scorings	The TF-IDF, Textrank and Question-relevant scores of the phrase	3
Word embedding	The embedding vector of the phrase, reduced from 768 to 10 dimensions	10
	Total	20

Training Procedure. The model training procedure employs k-fold cross-validation to maximize utilization of available training data while ensuring robust generalization. The dataset undergoes random partitioning into k folds, with each iteration designating one fold as validation data and remaining k-1 folds as training data. We utilize LightGBM, a state-of-the-art gradient boosting framework, to train k independent models across validation folds. This ensemble approach enhances prediction stability and reduces overfitting risks inherent in single-model approaches.

Inference Procedure. During inference, phrase scoring involves ensemble prediction where k trained models generate individual phrase scores that undergo averaging to produce final retention probabilities. Phrases receiving scores below a predefined threshold undergo removal, resulting in syntactically simplified sentences that replace original constructions in the abstractive summary. This learning-based approach enables adaptive pruning decisions that reflect dataset-specific patterns while maintaining grammatical coherence and medical content integrity. The pseudo-code of the inference process is described in Algorithm 1.

3.4. Post-processing

To refine summaries and address artifacts, redundancy, and formatting inconsistencies, we apply a post-processing pipeline with three key steps:

- *Token removal and formatting corrections*: We remove residual HTML tags, citations, and metadata while correcting punctuation and capitalization errors introduced by pruning.

Algorithm 1 Gradient Boosting Inference and Syntax Tree Pruning

Require: Abstractive summary $\hat{s}$; ensemble $\mathcal{M} = \{M_1, \ldots, M_k\}$; PCA model Φ; threshold δ; parser $\mathcal{P}$; encoder $\phi(\cdot)$
Ensure: Pruned summary $\hat{s}'$

$\hat{s}' \leftarrow [\,]$
for each sentence $s \in \hat{s}$ **do**
  $T \leftarrow \mathcal{P}(s)$ ▷ Parse constituency tree
  $\mathcal{R} \leftarrow \emptyset$ ▷ Set of subtrees to prune
  for each phrase constituent p in T **do**
    if $|p| < \tau_{\min}$ **then**
      continue
    end if
    $\mathbf{f}_{\text{tag}} \leftarrow \text{TagFreq}(p)$
    $\mathbf{f}_{\text{score}} \leftarrow [\text{TFIDF}(p),\ \text{TextRank}(p),\ \text{Query}(p)]$
    $\mathbf{e}_p \leftarrow \Phi(\phi(p))$ ▷ Reduce 768-d → 10-d
    $\mathbf{x} \leftarrow [\mathbf{f}_{\text{tag}} \mid \mathbf{f}_{\text{score}} \mid \mathbf{e}_p]$
    $\text{score}(p) \leftarrow \frac{1}{k} \sum_{j=1}^{k} M_j(\mathbf{x})$ ▷ Ensemble mean prediction
    if $\text{score}(p) < \delta$ **then**
      $\mathcal{R} \leftarrow \mathcal{R} \cup \{p\}$
    end if
  end for
  $s' \leftarrow \text{Reconstruct}(T, \mathcal{R})$ ▷ Emit leaves not covered by any pruned subtree
  Append s' to $\hat{s}'$
end for
return $\hat{s}'$

- *Irrelevant content filtering*: We remove contact details, URLs, metadata, and non-essential concluding segments using domain-specific rules and named-entity filtering to ensure summaries remain focused and relevant.

- *Duplicate reduction*: To address repetitions and paraphrasing, we apply maximal marginal relevance score [24] for each sentence s_i as follow:

$$f(s_i) = \text{sim}(s_i, Q) - \lambda \max_{j \neq i} \Big(\text{sim}(s_i, s_j)\Big)$$

where λ balances relevance and diversity. Low-scoring sentences are removed to enhance conciseness and informativeness.

4. Experimental Results

This section presents a comprehensive evaluation of our proposed SAMSUM model on the MEDIQA-MAS 2021 dataset, including detailed comparisons with state-of-the-art baselines, thorough analysis of the results across multiple evaluation metrics, and ablation studies to assess individual component contributions.

4.1. Dataset

Experiments were conducted on the MEDIQA-Answer Summarization (MEDIQA-AnS) dataset [25]. The training data consisted of 156 health questions asked by consumers, corresponding answers to these questions, and expert-created summaries of these answers by both extractive and abstractive approaches were used for training. The validation and test sets are automatically generated by the consumer health question answering system CHiQA[1], which searches for answers from only trustworthy medical information sources. They contain 50 and 80 questions, respectively, including multiple documents and the provided summaries. The overall statistics of training data are shown in Table 3, which highlights the differences between the training, validation, and testing sets.

Table 3. Dataset Statistics

Statistics	Training		Valid-ation	Testing
	Article	Section		
Average				
Answer / question	3.54	3.54	3.85	3.80
Tokens / answer	532.83	152.35	219.44	240.22
Token / single sum*	70.51	70.51	-	-
Token / multi sum†	119.04	119.04	81.18	-
Compression ratio‡				
Single sum*	0.07	0.32	-	-
Multi sum†	0.04	0.13	0.15	-

*Single-answer summary. †Multi-answer summary.
‡The summary-to-answer length ratio.

4.2. Evaluation metrics

For the automatic text summarization tasks, there is the need to compare the similarity between the predicted summary (hypothesis) and the actual summary (reference). `ROUGE` [26], an official metric in the MEDIQA 2021 contest, is most widely used. Moreover, the semantic-based metric called `BERTScore` [27] has received increasing attention recently, especially when used to evaluate abstractive summaries. To complement these reference-based metrics, we additionally employ *LLM-as-a-Judge* evaluation [28], which leverages large language models as automated evaluators to assess summary quality along multiple human-aligned dimensions.

`ROUGE`. `ROUGE` evaluation metric includes `ROUGE-1`, `ROUGE-2` and `ROUGE-L`. While `ROUGE-1` and `ROUGE-2` consider the existence of single words or bigrams between the hypothesis and reference texts, `ROUGE-L` concerns the order of those words by evaluating based on the Long Common Subsequence (LCS) between two texts. Those types of `ROUGE` involving Recall (R), Precision (P) and F1. The predicted summary should be both informative and accurate, producing the high F1 score.

[1] https://chiqa.nlm.nih.gov

BERTScore. The ROUGE metric requires that words used in the hypothesis and the reference must be an exact match. Because the generated abstractive summary can be rewritten and paraphrased from the original document that still retains the same meanings, ROUGE is only partially suitable. By using BERTScore [27], the semantic factors are taken into account for evaluation. Leveraging the pre-trained contextual embeddings of BERT-based models, BERTScore (i) creates word embedding for both two compared texts and (ii) matches words between the hypothesis and reference by the cosine similarity of their embedding vectors. BERTScore also involves in Recall (R), Precision (P) and F1.

LLM-as-a-Judge. Reference-based metrics such as ROUGE and BERTScore measure lexical and semantic overlap against a single gold reference, but do not directly capture human-perceived quality aspects such as factual faithfulness or informativeness [29]. To address this limitation, we adopt the *LLM-as-a-Judge* paradigm [28], prompting a capable LLM (Gemini 3 Flash) to score each predicted summary along four dimensions drawn from the SUMMEVAL framework [29]. Following Lee et al. [**?**], who show that decomposing high-level dimensions into fine-grained sub-dimensions substantially improves inter-evaluator agreement and reduces rating variance, we adopt the sub-dimension checklist as a structured reasoning prompt:

- *Coherence*: topic maintenance, logical flow, and consistent point of view.
- *Consistency*: factual accuracy, absence of hallucinated information, and preservation of the source intent.
- *Fluency*: formatting correctness, grammaticality, sentence completeness, and readability.
- *Relevance*: coverage of key points, terminological consistency, use of domain-specific phrases, and importance of included content.

However, we retain the 1–5 Likert scoring to ensure comparability with prior work and to avoid scoring discontinuities that arise when dimensions have unequal numbers of sub-questions. We report scores for both reference summaries and predicted summaries to contextualize model performance relative to human-written references [29].

4.3. Comparative models

We compare the proposed SAMSUM's results with our baseline transformer-based summarization models, including PEGASUS [30], BART [31], and BioBART [32], and concurrent large language models [17], such as GPT-3.5, PaLM-2, Claude-2, and LLaMa-2. We also compare our model with the top four teams from the MEDIQA 2021 MAS abstractive summarization challenge [1]:

- paht_nlp: a multi-grained, query-focused MAS model using pre-trained RoBERTa and Markov Chain model with incremental question-driven factors to evaluate sentences locally and globally;
- UCSD-Adobe: a transfer learning BART model incorporating answer sentence selection that trains on the HealthCareMagic question summarization dataset;
- UETfishes: a hybrid extractive-abstractive summarization approach with a query-driven filtering phase to choose useful information from the input document automatically;
- MNLP: PEGASUS model with multiple pre-processing techniques, including question focus identification on the input and the development of an ensemble method to combine question focus.

Table 4. Model performance and Comparison on MEDIQA-MAS 2021 challenge (%)

Models	ROUGE-1 F1	ROUGE-2 F1	ROUGE-L F1	BERT† F1
SAMSUM‡	**41.2** (±0.5)	**17.3** (±1.0)	22.7 (±0.8)	86.7 (±0.7)
Baseline transformer-based models				
BioBART	32.9	11.3	**29.3**	86.1
BART	30.1	11.5	13.9	80.4
PEGASUS	28.6	10.1	12.5	80.7
Large language models				
GPT-3.5	38.9	14.6	22.1	**87.9**
PaLM-2	15.3	8.6	13.5	85.2
Claude-2	13.4	6.2	11.1	85.6
LLaMA-2	13.7	11.2	13.2	86.6
Top 4 models on MEDIQA-MAS 2021 challenge				
paht_nlp	32.2	16.2	19.1	65.3
UCSD-Adobe	38.4	16.0	21.2	63.3
UETfishes	31.2	15.0	18.9	64.7
MNLP	34.9	11.7	20.5	57.6

†*BERT: BERTScore using the RoBERTa-Large model.*
‡*: Our results are mean ± standard deviation over 10 runs. The highest result in each column is in* ***bold*** *font.*

4.4. Overall Results and Analysis

Table 4 presents the evaluation results of our proposed SAMSUM model against established baselines and top-performing systems from the MEDIQA-MAS 2021 challenge. Our model achieves the highest performance in ROUGE-2 F1 score of 17.3% and second-best performance in BERTScore F1 of 86.7%. When compared to baseline transformer models, SAMSUM demonstrates substantial improvements in both content relevance and semantic coherence. The model outperforms BART and BioBART by approximately 6% in ROUGE-2 F1. Similarly, our approach surpasses PEGASUS with a 7.2% improvement in ROUGE-2 F1 and a 6% gain in BERTScore F1.

The model performance also surpasses concurrent large language models in terms of all ROUGE metrics, and ranks second in BERTScore F1 with the difference of 1.2%. The comparison with top-performing MEDIQA-MAS 2021 challenge systems further confirms the superiority of our approach. SAMSUM outperforms the best system paht_nlp by 1.1% in ROUGE-2 F1 and by 9% in ROUGE-1 F1. These consistent improvements across top-performance systems demonstrate that our syntax tree pruning mechanism and adaptive length prediction effectively bridge the gap between extractive and abstractive methods while maintaining superior semantic coherence in clinical discourse summarization.

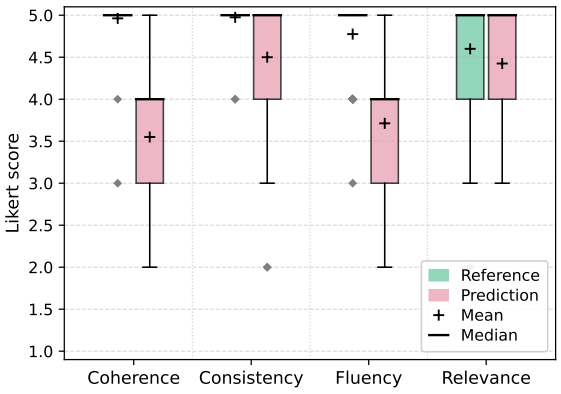


Figure 3. LLM-as-a-Judge scores (1–5) of predicted vs. reference summaries across four quality dimensions. Boxes show the interquartile range; + marks the mean; horizontal lines denote the median.

LLM-as-a-Judge Analysis. Figure 3 reports *LLM-as-a-Judge* scores for SAMSUM predictions alongside the reference summaries across the four SUMMEVAL dimensions.

Overall, predicted summaries score competitively with references on most dimensions, indicating that SAMSUM produces clinically plausible and linguistically well-formed outputs. Most notably, *Relevance* scores are the strongest across both conditions — the prediction achieves a mean of approximately 4.4, closely matching the reference mean of 4.6 — demonstrating that the model reliably captures salient clinical information, a critical requirement in biomedical summarization where missing key findings can have serious consequences. *Consistency* scores are also encouraging: while prediction shows a slightly wider spread than references (mean ≈4.5 vs. 5), the majority of

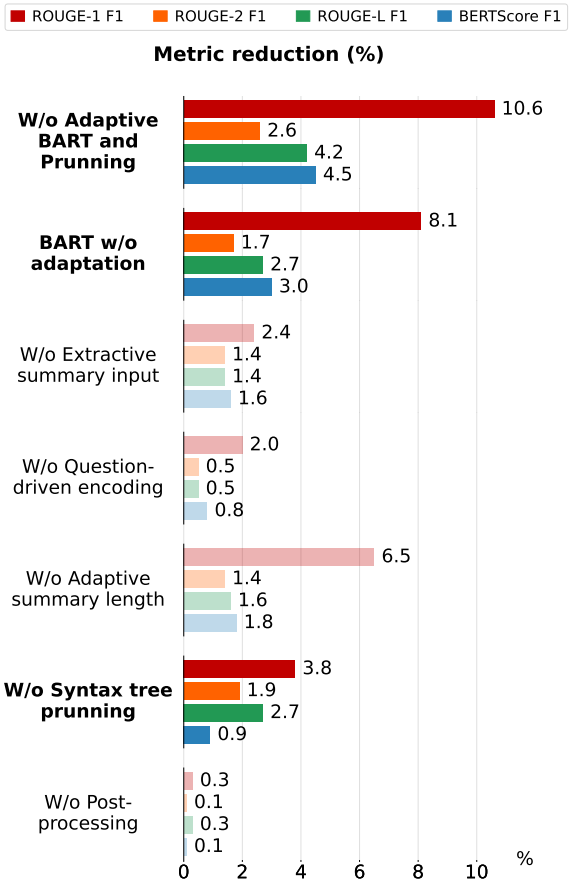


Figure 4. Model Contribution Analysis.

predictions remain factually faithful to the source. *Coherence* and *Fluency* follow expected patterns, with prediction scoring slightly below references (median 4.0), reflecting the inherent difficulty of maintaining discourse structure across multiple source notes — an area we leave for future work.

4.5. Model Components Contribution Analysis

To evaluate the individual contribution of each component in our SAMSum architecture, we conducted comprehensive ablation studies by systematically removing or modifying key components. Figure 4 presents the performance degradation when each component is excluded from the complete system, providing insights into the relative importance of our proposed modules.

The ablation analysis reveals that the adaptive BART component with syntax tree pruning represents the most critical element of our architecture. Removing both adaptive BART modifications and pruning mechanisms results in substantial performance degradation across all metrics, with `ROUGE-1` F1 declining by 10.6%, `ROUGE-L` F1 by 4.2%, and `BERTScore` F1 by 4.5%. This significant impact validates our core hypothesis that bridging extractive preprocessing with syntax-informed neural processing is essential for effective biomedical summarization.

Similarly, removing BART adaptations while maintaining other components leads to considerable performance loss, with `ROUGE-1` F1 decreasing by 8.1% and `BERTScore` F1 by 3.0%. The adaptive summary length component shows notable influence on model performance, particularly affecting `ROUGE-1` F1 scores with a 6.5% reduction when removed. The extractive summary input component demonstrates moderate but consistent impact across all evaluation metrics, with performance reductions ranging from 1.4% to 2.4%. The question-driven encoding component also shows the impact with performance decreases of 0.5-2.0% across metrics. These results demonstrate the critical role of our extractive summary input, query-conditioned formatting, and dynamic length prediction mechanisms in the adaptive BART architecture.

The syntax tree pruning module, when removed independently, causes a 3.8% decline in `ROUGE-1` F1 and 2.7% in `ROUGE-L` F1, confirming its effectiveness in eliminating redundancy while preserving semantic coherence. Interestingly, the impact on `BERTScore` F1 is relatively minimal (0.9% reduction), suggesting that syntax tree pruning primarily affects surface-level redundancy rather than semantic content preservation. The post-processing component exhibits the smallest performance impact when removed, with metric reductions of only 0.1-0.3%. This minimal influence suggests that the primary

Table 5. Heuristical pruning and preserving rules

Rule	POS tag	Explain
Pruning Relative Clause	WH*	Relative Clause modifies a noun or noun phrase and provides additional information about the noun it modifies.
Pruning Adjective Phrase	ADJP	An Adjective Phrase is a phrase that acts as an adjective by describing a noun or noun phrase and provides additional characteristics about the noun it described.
Pruning Adverbial Phrase	ADVP	An Adverbial Phrase modifies a verb or verb phrase and provides additional information about the verb it modifies.
Pruning Conjunction Phrase	CONJP	A Conjunction Phrase is attached to several types of phrases. In many cases in the datasets, the secondary phrase is typically shorter and less informative than the primary phrase.
Pruning Preposition Phrase	PP	A Prepositional Phrase is a group of words containing a preposition, a noun or pronoun object of the preposition, and any modifiers of the object.
Preserving Noun Phrase	NP	A Noun Phrase is a phrase that has a noun or pronoun as its head or performs the same grammatical function as a noun.
Preserving Verb Phrase	VP	A Verb Phrase is a phrase that has at least one verb and its dependents as its head or performs the same grammatical function as a verb.

quality improvements are achieved through the core adaptive BART and syntax tree pruning mechanisms, with post-processing serving as a refinement stage rather than a fundamental performance driver.

5. Discussion

5.1. Insights from Different Syntax Tree Pruning approaches

Heuristical Pruning. From the earlier research [33], the syntax tree pruning is divided into two categories:

- *Syntax-driven pruning* merely focuses on simplifying the syntax structure regardless of the importance and question relevance of the content. The rules for POS tags removing or preserving introduced by the authors are listed in Table 5.
- *Relevancy-driven pruning* optimizes the above by taking the question into consideration, only phrases that are low relevant to the question are discarded.

However, those pruning rules encounter several weaknesses:

- *Combination of both approaches.* Both approaches have their own merits and hindrances. Requires a method to let the two approaches complement each other.
- *Shallow sub-tree analysis.* For each sub-tree, their work only considers the tag of the sub-tree without finding more information from child nodes, which may contain vital information.
- *Fixed tags pruning rules.* They use some `IF` conditions to check whether the sub-tree's tag must be preserved, should be removed or none of both cases.

To overcome the above drawbacks, the phrase scoring that aggregates both aforementioned and some new sub-tree features is applied instead of simple pruning rules:

$$\begin{aligned}\text{score}(p) = {} & W_1 \times \text{tag}(p) + W_2 \times \text{tf-idf}(p) \\ & + W_3 \times \text{lexrank}(p) + W_4 \times \text{question}(p)\end{aligned} \tag{11}$$

whereas W_1, W_2, W_3, W_4 are the weights of the *Sub-tree tag* scoring, *TF-IDF* scoring, *LexRank* scoring and *Question-relevant* scoring, respectively.

Finally, phrases whose scores are under the pre-defined threshold are discarded. This phrase pruning process is iteratively executed to all sentences in the abstractive summary generated by the BART model to produce a shorter and higher accurate summary of which irrelevant phrases are cut off.

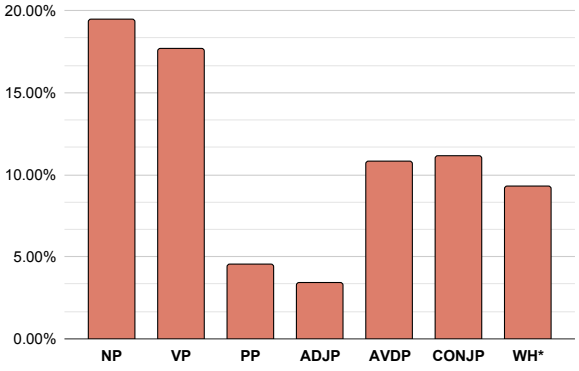


Figure 5. The ratio of POS tags existing in the abstractive summaries.

Statistical Pruning. The apparent difficulty of the heuristical pruning approach is the *weight tuning*. Many given parameters are manually selected by experience, which can bias the phrase scoring. Furthermore, the experience can vary enormously depending on the characteristics of the datasets. For better weight selection for POS tags and scoring functions based on the observation of a specific dataset, a statistical pruning approach is proposed.

Firstly, the appearance of POS tags within the training reference abstractive summaries is statistically analyzed to choose the appropriate range of POS tags' weights. Figure 5 demonstrates the average ratio of phase count in the reference summary, and the total of phrases in the input text under each POS tag. Then, the weight calibration process is iteratively executed by random sampling of POS tags and phase scores' weights, and hill-climbing with simulated annealing for choosing the best explored state of weights. The pseudo-code of the fine-tuned process is described in Algorithm 2.

Algorithm 2 Hill-Climbing Algorithm

Require: $num\_iters > 0$, $0 < c < 1$, weight scoring function $f(S)$
Ensure: The best weight set S_{best}
$S_{best} \leftarrow$ Random vector
for $iter \leftarrow 1$ to num_iters **do**
  $S \leftarrow$ In-range random sampling vector
  while termination criterion is not satisfied **do**
    $S_{next} \leftarrow S + \Delta S$
    if $f(S_{next}) > f(S)$ **then**
      $S \leftarrow S_{next}$
    else
      $\Delta f \leftarrow f(S_{next}) - f(S)$
      $r \leftarrow random(0, 1)$
      if $r > e^{-\Delta f/T}$ **then**
        $S \leftarrow S_{next}$
      end if
    end if
    if $f(S_{best}) < f(S)$ **then**
      $S_{best} \leftarrow S$ ▷ Update best weight
    end if
    $T \leftarrow c \times T$ ▷ Temperature decrease periodically
  end while
end for
return S_{best} ▷ Return the best weights set

Gradient Boosting Pruning. However, the statistical pruning still has evident drawbacks:

- *Time-consuming* due to the vast iterations in random sampling hill-climbing.
- *Hardly finding the global maximum* due to the random-relying learning strategy.
- *Limited knowledge extraction from the dataset* as the statistics are not optimal

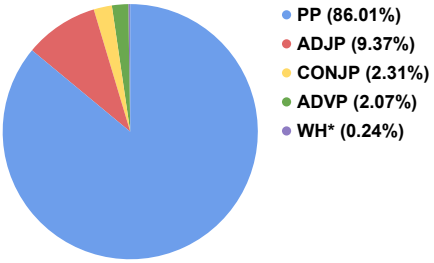
Table 6. The performance of syntax tree pruning methods (%)

Model	ROUGE-1			ROUGE-2			ROUGE-L			Pruning
	R	P	F1	R	P	F1	R	P	F1	ratio
Heuristical Pruning	75.5	22.6	33.7	35.0	10.0	15.0	45.4	13.5	20.1	11.53
Statistical Pruning	76.1	**23.9**	**35.2**	34.8	10.9	15.9	46.2	**14.6**	**21.3**	11.64
Gradient Boosting Pruning	**78.2**	22.9	34.5	**38.1**	**11.0**	**16.6**	**48.1**	14.1	21.2	11.12

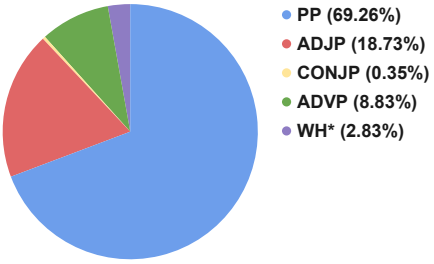
The highest results in each column are bold highlighted.

enough to explore more characteristics of the dataset.

For those reasons, a learning-based Gradient Boosting Pruning approach is proposed to find optimal weights for each feature, based on the characteristics of the dataset.



(a) Statistical Pruning



(b) Learning-based GB Pruning

Figure 6. The ratios of pruned POS tags.

Table 6 shows the comparative performance of three syntax tree pruning methods evaluated by the ROUGE metric. From the table, while the Statistical Pruning acquires good results in terms of ROUGE-1 and ROUGE-L, the Learning-based Gradient Boosting Pruning has witnessed a considerable increase in the ROUGE-2, achieving a large pruning ratio of approximately 11%.

To further explore the pruning effects by the Statistical and Gradient Boosting methods, Figure 6 compares their proportions of pruned POS tags. From that illustration, the Preposition (PP) tag is majorly pruned, followed by the Adjective (ADJP) tag. Changing from the Statistical Pruning to the Gradient Boosting Pruning, a decrease in the PP tag ratio is observed, while the ADJP and Adverbial (ADVP) ratios witness considerable growth. This emphasizes the learned patterns from the training dataset of the Gradient Boosting Pruning approach.

5.2. *Error Analysis*

To provide deeper insights into our model's performance and limitations, we conducted a comprehensive error analysis using representative examples from the MEDIQA-MAS 2021 dataset. Table 7 and 8 present four illustrative cases that demonstrate both the strengths and weaknesses of our SAMSum approach compared to reference abstractive summaries.

Content Preservation and Medical Accuracy. Our analysis reveals that SAMSum consistently maintains medical accuracy while preserving essential clinical information (in Question 1 and 3). The model's ability to maintain factual consistency across complex medical topics validates the effectiveness of our syntax tree pruning mechanism in preserving medical content integrity.

Information Synthesis and Redundancy Management. The examples demonstrate SAMSUM's capacity for effective information

Table 7. Examples of SAMSUM's summaries compared to reference abstractive summaries (1)

Data & Prediction		Summary
Question 1	Question	How should the parents boost delayed puberty in a 14-year-old girl?
	Reference	Normally, **puberty in girls starts when they are between 8 to 15 years old**, which explains why **children in 7th grade may look as young children or almost grown up**. **Problems with puberty are caused by hormones**. **Puberty is delayed or doesn't start at all if there are too few hormones, which could be caused by different problems**.
	SAMSUM	**Most girls go through puberty somewhere between being 8 to 15 years old**. Puberty is a wide age range when puberty starts. **Problems with puberty usually involve hormones**. There are many **different hormone problems that result in too few hormones so that puberty is delayed or doesn't start at all**. **Some kids in 7th grade still look like young children and others look really grown up**.
Question 2	Question	How should the parents boost delayed puberty in a 14-year-old girl?
	Reference	**Ropinirole may cause side effects: - nausea - vomiting - stomach pain (...) - sweating or flushing** - confusion - difficulty remembering or concentrating - anxiety - uncontrolled, sudden body movements - shaking of a part of your body (...) - difficulty urinating or pain when urinating - **in men, difficulty achieving or maintaining an erection - back, muscle, or joint pain - pain, burning, numbness, or tingling in the hands or feet** - swelling of the hands, arms, feet, ankles, or lower legs - dry mouth. **Some side effects can be serious:** - **hallucinations (seeing things or hearing voices that do not exist) - fainting (...) - double vision or other changes in vision**.
	SAMSUM	**Ropinirole may cause side effects: nausea, vomiting, stomach pain (...), sweating or flushing**. **Some side effects can be serious:** pounding in the ears, rapid weight gain, sensation of spinning**hallucinations seeing things or hearing voices that do not exist, fainting (...), double vision or other changes in vision**. **In men, difficulty achieving or maintaining an erection, back, muscle, or joint pain, pain, burning, numbness and tingling in the hands or feet**.

The content in some text has been shortened by replacing with (...)

synthesis, particularly evident in Question 2 concerning Ropinirole side effects. The model successfully consolidates an extensive list of potential adverse reactions while maintaining clinical relevance and readability. However, the analysis also reveals areas for improvement in redundancy elimination. In Question 3, both the reference and SAMSUM summaries contain some repetitive information about physical characteristics, suggesting that our syntax tree pruning approach could benefit from enhanced semantic similarity detection to further reduce content duplication.

Query-Response Alignment and Structural Coherence. A notable strength of our approach is demonstrated in Question 4, where SAMSUM effectively addresses the specific query about zoledronic acid contraindications in patients with scleroderma. This example illustrates the effectiveness of our query-conditioned formatting component in ensuring response relevance. However, comparison with the reference summary reveals that our model occasionally provides more comprehensive information than strictly necessary, suggesting opportunities for refinement in length optimization algorithms to achieve more precise information density control. Besides, there are also some misspellings due to the generative models.

5.3. Ethical Considerations

As a biomedical summarization system intended for clinical decision support, SAMSUM presents several ethical considerations that must be addressed before deployment.

Bias and Fairness. The MEDIQA-MAS training dataset may contain demographic biases reflecting underrepresentation of certain patient populations, geographic regions, or medical conditions. Our model's performance may vary across different demographic groups and medical specialties not well-represented in training data. Future work should conduct systematic bias audits examining model outputs across diverse patient demographics, disease categories, and

Table 8. Examples of SAMSUM's summaries compared to reference abstractive summaries (2)

Data and prediction		Summary
Question 3	Question	What are the symptoms of cri-du-chat syndrome?
	Reference	**Cri-du-chat syndrome** could be inherited or **caused by a random mutation** that results in **the intellectual disability and health problems**. The syndrome is called cri-du-chat because one of the **symptoms is a characteristic high-pitched, mewing cry that sounds like a kitten crying. This cry starts after birth and lasts several weeks before disappearing**. The symptoms also include a low birth weight and a small head with **a round face, small jaw, (...) and abnormally shaped ears set low in the head**. The infant may seem limp and have **webbed fingers and toes and heart defects**.
	SAMSUM	**Cri du chat syndrome is caused by a missing piece of chromosome** (...) About 10 percent of people with cri-du-chat syndrome inherit the chromosome abnormality. An infant with this syndrome may have **a round face, small jaw, (...) and abnormally shaped ears set low in the head**. **Webbed fingers and toes syndactyly and heart defects are common**. There are **significant limitations in mental and physical development**. Many children with the syndrome survive but have substantial disabilities. The deletion occurs as a random event during the formation of reproductive cells eggs (...) **The symptoms of the syndrome include a characteristic high-pitched, mewing cry that sounds like a kitten crying. The cry may be heard immediately after birth, lasts several weeks, and then disappears**.
Question 4	Question	Can I take zoledronic acid infusion if my mother has scleroderma?
	Reference	Your doctor might **advise against taking Zoledronic Acid if you have kidney problems**; have trouble absorbing minerals in your stomach; are pregnant or breastfeeding or plan to become pregnant; had surgery that removed your parathyroid or thyroid glands, or part of the intestines; if you are sensitive to aspirin, cannot take calcium supplements, **have low blood calcium**. People with scleroderma have calcium deposits in their tissues.
	SAMSUM	Zoledronic Acid ZOE-le-DRON-ik AS-id Injection is a prescription medicine used to: Treat certain men and women who have Paget's disease of the bone. **Do not take zoledronic acid injection if you: Have low levels of calcium in your blood Have kidney problems** Are allergic to zoledronics acid or any of its ingredients. Severe kidney problems may happen when you take Zoledronicacid injection. Patients should tell their doctor if they have kidney problems (...). See FDA-Approved Medication Guide Information for Patients for more information (...).

The content in some text has been shortened by replacing with (...)

socioeconomic contexts to identify and mitigate potential disparities in summary quality.

Factual Accuracy and Clinical Safety. While our evaluation demonstrates strong performance on automatic metrics and LLM-based quality assessment, neural summarization models remain susceptible to factual errors and hallucinations. Generated summaries may occasionally contain medically inaccurate information, omit critical details, or introduce misleading content not present in source documents. Such errors could lead to inappropriate clinical decisions if summaries are used without verification. We strongly emphasize that SAMSUM should augment rather than replace clinical expertise and human judgment.

Deployment Guidelines. We recommend the following safeguards for responsible deployment:

- Human-in-the-loop validation: All generated summaries should undergo review by qualified healthcare professionals before clinical use.
- Source traceability: Systems should provide access to original source documents enabling clinicians to verify summary content.
- Uncertainty indication: Future versions should incorporate confidence scores or uncertainty estimates to flag potentially unreliable outputs.
- Continuous monitoring: Deployed systems require ongoing performance monitoring and periodic retraining to maintain accuracy as medical knowledge evolves.

Limitations and Transparency. We acknowledge that our system has inherent limitations: it is trained on consumer health question-answering data and may not generalize to specialized clinical contexts; it processes only text inputs and cannot integrate multimodal clinical data; and its length prediction and pruning mechanisms reflect dataset-specific patterns that may not transfer to other applications. Users should be clearly informed of these limitations to prevent inappropriate reliance on system outputs beyond its validated scope.

These considerations underscore the importance of treating SAMSUM as a clinical support tool requiring expert oversight rather than an autonomous decision-making system.

6. Conclusion

This work introduces SAMSUM, a model that addresses fundamental limitations in biomedical multi-answer summarization through adaptive transformer architectures with syntax-informed neural processing. Our approach bridges the gap between extractive reliability and abstractive fluency that has hindered existing summarization systems in healthcare applications. The syntax tree pruning mechanism demonstrates how supervised learning approaches enhance abstractive outputs while preserving medical content integrity. Experimental validation on the MEDIQA-MAS 2021 dataset proves our model's substantial improvements over baselines and top-performing challenge participants. Results demonstrate that our hybrid architecture addresses core challenges of input length limitations, factual consistency risks, and query-driven content selection in existing biomedical summarization systems. Future research includes extending our framework to other specialized domains and exploring knowledge graph integration for enhanced factual consistency verification.

Limitations

Despite the promising results, our approach still has limitations that provide directions for future improvements. The syntax tree pruning mechanism relies on predefined heuristics and feature-based scores that may not generalize well across diverse biomedical text structures, suggesting the need for dynamic, data-driven pruning methods such as reinforcement learning approaches. Additionally, our length predictor is trained on a specific biomedical dataset, which may limit generalization to new corpora with different length distributions, indicating opportunities for meta-learning techniques or dataset-adaptive prediction models. Finally, our evaluation is limited to a single biomedical dataset and lacks comprehensive human evaluation from domain experts, which restricts our understanding of the model's practical usability in authentic clinical settings.

Acknowledgement

This work has been supported by VNU University of Engineering and Technology under project number CN25.12.

References

[1] A. Ben Abacha, Y. Mrabet, Y. Zhang, C. Shivade, C. Langlotz, D. Demner-Fushman, Overview of the MEDIQA 2021 Shared Task on Summarization in the Medical Domain, D. Demner-Fushman, K. B. Cohen, S. Ananiadou, J. Tsujii (Eds.), Proceedings of the 20th Workshop on Biomedical Language Processing, Association for Computational Linguistics, Online, 2021, pp. 74–85, https://doi.org/10.18653/v1/2021.bionlp-1.8.

[2] W. Guan, I. Smetannikov, M. Tianxing, Survey on Automatic Text Summarization and Transformer Models Applicability, Proceedings of the 2020 1st International Conference on Control, Robotics and Intelligent System, CCRIS '20, Association for Computing Machinery, New York, NY, USA, 2021, p. 176–184, https://doi.org/10.1145/3437802.3437832.

[3] S. Gupta, A. Sharaff, N. K. Nagwani, Biomedical Text Summarization: a Graph-based Ranking Approach, Applied Information Processing Systems: Proceedings of ICCET 2021, Springer, 2021, pp. 147–156, https://doi.org/10.1007/978-981-16-2008-9_14.

[4] Q.-A. Nguyen, Q.-H. Duong, M.-Q. Nguyen, H.-S. Nguyen, H.-Q. Le, D.-C. Can, T. D. Thanh, M.-V. Tran, A Hybrid Multi-answer Summarization Model for the Biomedical Question-Answering System, 2021 13th International Conference on Knowledge and Systems Engineering (KSE), 2021, pp. 1–6, https://doi.org/10.1109/KSE53942.2021.9648640.

[5] T. Wang, C. Yang, M. Zou, J. Liang, D. Xiang, W. Yang, H. Wang, J. Li, A Study of Extractive Summarization of Long Documents Incorporating Local Topic and Hierarchical Information, Scientific Reports, Vol. 14, No. 1, 2024, pp. 10140, https://doi.org/10.1038/s41598-024-60779-z.

[6] H. Shakil, A. Farooq, J. Kalita, Abstractive Text Summarization: State of the Art, Challenges, and Improvements, Neurocomputing, Vol. 603, 2024, pp. 128255, https://doi.org/10.1016/j.neucom.2024.128255.

[7] Y. Yang, J. Qi, Memorization Capability of Medical Large Language Models and User Privacy Data Disclosure Willingness, International Journal of Human–Computer Interaction2025, pp. 1–25, https://doi.org/10.1080/10447318.2025.2586693.

[8] K. Singhal, T. Tu, J. Gottweis, R. Sayres, E. Wulczyn, M. Amin, L. Hou, K. Clark, S. R. Pfohl, H. Cole-Lewis, et al., Toward Expert-level Medical Question Answering with Large Language Models, Nature medicine, Vol. 31, No. 3, 2025, pp. 943–950, https://doi.org/10.1038/s41591-024-03423-7.

[9] J. He, B. Zhang, H. Rouhizadeh, Y. Chen, R. Yang, J. Lu, X. Chen, N. Liu, I. Li, D. Teodoro, Retrieval-Augmented Generation in Biomedicine: A Survey of Technologies, Datasets, and Clinical Applications, arXiv2025, pp. arXiv–2505, https://doi.org/10.48550/arXiv.2505.01146.

[10] H. Sakai, S. S. Lam, KDH-MLTC: Knowledge Distillation for Healthcare Multi-Label Text Classification, arXiv2025, pp. 2505, https://doi.org/10.48550/arXiv.2505.07162.

[11] R. AlSaad, A. Abd-Alrazaq, S. Boughorbel, A. Ahmed, M.-A. Renault, R. Damseh, J. Sheikh, Multimodal Large Language Models in Health Care: Applications, Challenges, and Future Outlook, Journal of medical Internet research, Vol. 26, 2024, pp. e59505, https://doi.org/10.2196/59505.

[12] S. Gurajada, S. Seufert, I. Miliaraki, M. Theobald, Using Graph Summarization for Join-Ahead Pruning in a Distributed RDF Engine, SWIM'14, Association for Computing Machinery, New York, NY, USA, 2014, p. 1–4, https://doi.org/10.1145/2630602.2630610.

[13] L. Guan, Y. Huang, J. Liu, Biomedical Question Answering via Multi-Level Summarization on a Local Knowledge Graph, arXiv2025, pp. 2504, https://doi.org/10.48550/arXiv.2504.01309.

[14] Q. Xie, P. Tiwari, S. Ananiadou, Knowledge-Enhanced Graph Topic Transformer for Explainable Biomedical Text Summarization, IEEE Journal of Biomedical and Health Informatics, Vol. 28, No. 4, 2024, pp. 1836–1847, https://doi.org/10.1109/JBHI.2023.3308064.

[15] K. Singhal, S. Azizi, T. Tu, S. S. Mahdavi, J. Wei, H. W. Chung, N. Scales, A. Tanwani, H. Cole-Lewis, S. Pfohl, et al., Large Language Models Encode Clinical Knowledge, Nature, Vol. 620, No. 7972, 2023, pp. 172–180, https://doi.org/10.1038/s41586-023-06291-2.

[16] Z. Cao, V. K. Keloth, Q. Xie, L. Qian, Y. Liu, Y. Wang, R. Shi, W. Zhou, G. Yang, J. Zhang, et al., The Development Landscape of Large Language Models for Biomedical Applications, Annual Review of Biomedical Data Science, Vol. 8, 2025, https://doi.org/10.1146/annurev-biodatasci-102224-074736.

[17] I. Jahan, M. T. R. Laskar, C. Peng, J. X. Huang, A Comprehensive Evaluation of Large Language Models on Benchmark Biomedical Text Processing Tasks, Computers in biology and medicine, Vol. 171, 2024, pp. 108189, https://doi.org/10.1016/j.compbiomed.2024.108189.

[18] H. Xu, S. Fan, Y. Wang, Z. Huang, H. Xu, P. Xie, Tree2tree Structural Language Modeling for Compiler Fuzzing, International Conference on Algorithms and Architectures for Parallel Processing, Springer, 2020, pp. 563–578, https://doi.org/10.1007/978-3-030-60245-1_38.

[19] Z. Li, S. Ghodrati, A. Yazdanbakhsh, H. Esmaeilzadeh, M. Kang, Accelerating Attention Through Gradient-based Learned Runtime Pruning, Proceedings of the 49th Annual International Symposium on Computer Architecture, ISCA '22, Association for Computing Machinery, New York, NY, USA, 2022, p. 902–915, https://doi.org/10.1145/3470496.3527423.

[20] M. Neumann, D. King, I. Beltagy, W. Ammar, ScispaCy: Fast and Robust Models for Biomedical Natural Language Processing, Proceedings of the 18th BioNLP Workshop and Shared Task, Association for Computational Linguistics, Florence, Italy, 2019, pp. 319–327, https://doi.org/10.18653/v1/W19-5034.

[21] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, BioBERT: a Pre-trained Biomedical Language Representation Model for Biomedical tText Mining, Bioinformatics, Vol. 36, No. 4, 2020, pp. 1234–1240, https://doi.org/10.1093/bioinformatics/btz682.

[22] I. M. K. Karo, A. Perdana, S. Dewi, Automatic Text Review Summarization of Digital Library System Application using TextRank Algorithm and TF-IDF, 2024 4th International Conference of Science and Information Technology in Smart Administration (ICSINTESA), 2024, pp. 570–575, https://doi.org/10.1109/ICSINTESA62455.2024.10747952.

[23] S. M. Jijo, D. Panchal, J. Ardeshana, U. Chaudhari, Text Summarization using Textrank, Lexrank and Bart model, 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT), 2024, pp. 1–7, https://doi.org/10.1109/ICCCNT61001.2024.10725676.

[24] A. Onan, H. Alhumyani, Knowledge-Enhanced Transformer Graph Summarization (KETGS): Integrating Entity and Discourse Relations for Advanced Extractive Text Summarization, Mathematics, Vol. 12, No. 23, 2024, pp. 3638, https://doi.org/10.3390/math12233638.

[25] M. Savery, A. B. Abacha, S. Gayen, D. Demner-Fushman, Question-driven Summarization of Answers to Consumer Health Questions, Scientific Data, No. 1, 2020, pp. 1–9, https://doi.org/10.6084/m9.figshare.12676655.

[26] C.-Y. Lin, ROUGE: A Package for Automatic Evaluation of Summaries, Text Summarization Branches Out, Association for Computational Linguistics, Barcelona, Spain, 2004, pp. 74–81, https://aclanthology.org/W04-1013/ Accessed on June 01, 2024.

[27] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, Y. Artzi, BERTScore: Evaluating Text Generation with BERT, International Conference on Learning Representations (ICLR), 2020, pp. 1–14, https://openreview.net/forum?id=SkeHuCVFDr Accessed on December 01, 2022.

[28] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. P. Xing, H. Zhang, J. E. Gonzalez, I. Stoica, Judging LLM-as-a-judge with MT-bench and Chatbot Arena, Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23, Curran Associates Inc., Red Hook, NY, USA, 2023, https://dl.acm.org/doi/abs/10.5555/3666122.3668142 Accessed on June 01, 2025.

[29] A. R. Fabbri, W. Kryściński, B. McCann, C. Xiong, R. Socher, D. Radev, SummEval: Re-evaluating Summarization Evaluation, Transactions of the Association for Computational Linguistics, Vol. 9, 2021, pp. 391–409, https://doi.org/10.1162/tacl_a_00373.

[30] J. Zhang, Y. Zhao, M. Saleh, P. J. Liu, PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization, Proceedings of the 37th International Conference on Machine Learning, ICML'20, JMLR.org, 2020.

[31] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 7871–7880, https://doi.org/10.18653/v1/2020.acl-main.703.

[32] H. Yuan, Z. Yuan, R. Gan, J. Zhang, Y. Xie, S. Yu, BioBART: Pretraining and Evaluation of A Biomedical Generative Language Model, D. Demner-Fushman, K. B. Cohen, S. Ananiadou, J. Tsujii (Eds.), Proceedings of the 21st Workshop on Biomedical Language Processing, Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 97–109, https://doi.org/10.18653/v1/2022.bionlp-1.9.

[33] P. Perera, L. Kosseim, Evaluating Syntactic Sentence Compression for Text Summarisation, E. Métais, F. Meziane, M. Saraee, V. Sugumaran, S. Vadera (Eds.), Natural Language Processing and Information Systems, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 126–139, https://doi.org/10.1007/978-3-642-38824-8_11.

A. Training Environment and Hyperparameter Configurations

This appendix documents the computational environment used for all experiments and the complete hyperparameter configurations for each model component.

A.1. Training Environment

All experiments were conducted on Google Colaboratory (Colab) with the following environment:

`GPU`	NVIDIA Tesla T4 (16 GB VRAM, provided by Colab)
`RAM`	12.7 GB system RAM (Colab standard allocation)
`Python`	3.10
`PyTorch`	2.10 (CUDA backend)

`Transformers`	Hugging Face `transformers` 4.9
`LightGBM`	4.6
`scikit-learn`	1.6
`NLTK`	3.9
`spaCy`	3.8 with `en_core_sci_lg` (ScispaCy)

A.2. Extractive Summarization Hyperparameters

The extractive pre-processing stage linearly combines four sentence-level scoring signals. The following parameters govern scoring and sentence selection:

`TFIDF_WEIGHT`	1.0
`LEXRANK_WEIGHT`	1.0
`KEYWORDS_WEIGHT`	1.0
`QUERY_BASED_WEIGHT`	5.0
`Max output sentences`	30
`LexRank threshold`	0.3 (Min cosine similarity for LexRank graph edge)
`TF-IDF threshold`	0.2 (Min token TF-IDF value to accumulate)

A.3. BART Abstractive Summarization Hyperparameters

The adaptive BART component is built on `facebook/bart-large-cnn`. Dynamic output length is controlled by comparing extractive and reference token counts at inference time:

`Backbone model`	`bart-large-cnn`
`Max input tokens`	1024
`Beam size`	5
`No-repeat` n`-gram`	3
`RANGE_RATIO`	0.4 (Offset added to the dynamic length lower bound)

A.4. Gradient Boosting Phrase Classification Hyperparameters

Feature Engineering

Phrase representations are encoded with `dmis-lab/biobert-base-cased-v1.2` (mean-pooled, 768 d) and compressed via PCA before concatenation with syntactic and lexical features:

`Encoder model`	`biobert-base -cased-v1.2`
`Raw embedding dim`	768
`PCA target dim`	10
`POS tag groups`	7
`Scoring features`	3
`Total feature dim`	20
`Min phrase tokens`	2
`Max sentence length`	300

LightGBM Training

Models are trained with k-fold cross-validation using LightGBM in regression mode, predicting continuous phrase retention scores:

`Number of folds` k	5
`n_estimators`	500
`learning_rate`	0.05
`num_leaves`	31
`min_child_samples`	20
`subsample`	0.8
`colsample_bytree`	0.8
`Pruning threshold` δ	0.5 (Phrases below this score are removed)

Table 9. Inference Latency Benchmarks for the MAS Pipeline

Hardware Setup	Device	Mean (s/sample)	Std Dev (s)	Samples/Min
Colab Default	CPU (2 vCPUs)	21.5	± 4.5	2.8
Kaggle/Colab Free	NVIDIA T4	2.1	± 0.3	28.6

B. Inference-Time Performance Analysis

To assess the practical viability of the proposed multi-stage summarization pipeline, we conducted an empirical evaluation of inference latency across different hardware configurations commonly used in research environments (Kaggle and Google Colab).

The benchmarking was performed using the MEDIQA-Answer Summarization (MAS) dataset. The pipeline configuration involved an extractive cutoff of 30 sentences ($C = 30$) followed by a BART-large abstractive model utilizing beam search with a width of 5 (*num_beams* = 5). Time measurements exclude model loading and I/O overhead, focusing strictly on the transformation from raw input to the final abstractive summary.

As illustrated in Table 9, the transition from CPU-only to GPU-accelerated environments yields a significant performance gain. We categorize the latency into two primary phases:

1. *Syntactic and Extractive Phase:* This phase includes pre-processing, LexRank/TF-IDF scoring, and the *Gradient Boosting Phrase Classification*. Since these operations are primarily sequential and involve tree traversal (NLTK/spaCy), they are CPU-bound. In our tests, this accounts for approximately 15% to 25% of the total latency on GPU setups.
2. *Abstractive Generation Phase:* The BART-large model serves as the primary computational bottleneck. On standard CPUs, autoregressive decoding with beam search is prohibitively slow due to the high dimensionality of the Transformer layers. The NVIDIA T4 GPU reduces this latency by nearly 85%, making real-time summarization feasible.

The standard deviation (σ) is largely influenced by the input sequence length. Since MAS dataset answers can vary in length, the extractive step occasionally feeds a maximum of 1024 tokens to BART, leading to the observed variance in processing time.