.

S-3 Pipeline for Biomedical Text Simplification

Notebook for the SimpleText Lab at CLEF 2025

Ansh Vora[1], Tanish Chaudhari[1], Sanjeev Hotha[1] and Dr. Sheetal Sonawane[1]

[1]*Pune Institute of Computer Technology (PICT), Pune, India*

Abstract

Text simplification is an important field in natural language processing that is aimed at improving the readability and understanding of complex text. This makes scientific and technical text more accessible for individuals with limited literacy and comprehension and aids patients in a healthcare setting. We present the S-3 Simplification system – an expert in biomedical text simplification that produces lexically and structurally simplified text, which is semantically fluent, accurate and easy to understand. The system integrated a semantic simplification using T5 models, AMR (Abstract Meaning Representation) -guided structural simplification and BERT-masked modelling with medical thesaurus for context-aware synonym substitution. This approach highlights the effectiveness of a hybridized model for maintaining meaning and fluency while achieving lexical and syntactic simplification.

Keywords

Text Simplification, Semantic, Syntactic, Lexical, Abstract Meaning Representation, BERT-Masking, Biomedical

1. Introduction

Increasing biomedical and clinical literature has led to a vast repository containing scientific and medical information. While this data carries critical information in the studies, its linguistic complexity tends to make it incomprehensible and inaccessible. These clinical reports, reviews, and other literature contain complex semantics and domain-specific jargon that are usually at a graduate level or above. Hence, in the biomedical domain, simplification helps make biomedical literature more accessible and encourage better health literacy. However, the general simplification of sentences may pose significant challenges in maintaining biomedical contexts, where subtle semantic deviations can have significant consequences during interpretation.

Recent advancements in the field of Natural Language Processing (NLP), especially in transformer-based architectures such as BERT, T5, Pegasus, and SciBERT, have enabled progress in tasks such as summarization and translation. In this study for SimpleText, we propose a comprehensive and modular method for performing biomedical text simplification. Our proposed method is devised such that the multidimensional nature of simplification, that is, the lexical, syntactical and semantic simplification is performed in an integrated manner. Lexical simplification is performed by leveraging BERT-based masked language modelling and WordNet filtering to replace complex terms with simpler synonyms. Semantic and syntactic simplification was achieved using pre-trained models like Pegasus and T5 for paraphrasing and sentence restructuring. And finally, intelligent output merging, which integrates multiple simplification outputs based on readability, fluency, and semantic consistency using automated metrics like SARI, BLEU, and FKGL. Additionally, for lexical simplification, we used an AMR–guided mechanism. By extracting biomedical entities from AMR graphs and replacing them using WordNet and masked language modelling with contextual scoring, we ensure that simplifications remain aligned with the original meaning. Furthermore, the pipeline is optimized with several enhancements such as frequency-aware synonym selection, context-sensitive scoring using pseudo-log-likelihoods, and GPU-accelerated evaluation. Through SimpleText, we aim for more accessible biomedical content, supporting not just researchers but also everyone involved in scientific and healthcare fields, and even the general public in engaging with medical information more effectively.

CLEF 2025 Working Notes, 9 – 12 September 2025, Madrid, Spain

2. Literature Survey

SimpleText 2025 [1] [2] [3] track at CLEF 2025 is focused on simplification of Biomedical Text. Recently, there have been numerous advances in the domain of biomedical text simplification. Attal et al. [4] introduced the PLABA dataset with 7,600 sentence pairs that are distributed over 750 abstracts and consist of complex and simplified sentence pairs, and Bakker et al. (2024) [5] proposed the Cochrane dataset that is a large parallel corpus of Cochrane biomedical reviews and corresponding summaries that are aligned at various textual levels. Basu et al. (2023) also proposed a crowdsourced dataset annotated with transformation types that can be used for controlled simplification, while Ren et al. (2023) [6] extended the Cochrane with MULTICHOCRANE that became the first multilingual simplification corpus in the biomedical domain, although there was a remarkable performance gap with English models far outperforming other languages.

Text simplification methods originally included procedural methods but have now shifted majorly towards data-driven neural network architectures, as highlighted in Ondov et al. (2022) [7]. The paper also underlined the challenges in text simplification such as low-resource settings and domain sensitivity. Li et al. (2024) [8] introduced domain-specific biomedical simplification using several Large Language Models (LLMs) including BART, T5, SciFive, and GPT-4. This research showcased how meaning preservation remained a critical challenge despite good results. Knappich et al. (2023) [9] then introduced a novel Llama-2-based architecture that incorporated token-level loss weighting for improved emphasis on simplification edits, while Flores et al. (2023) [10] optimized output by introducing unlikelihood training and ranked decoding. Phatak et al. (2022) [11] used an innovative approach incorporating reinforcement learning to optimize simplicity in the TESLEA model. Gill et al. (2023) [12] pioneered rule-based simplification using a knowledge-based approach (KITS) to enhance biomedical relation extraction.

The T5 models have always been favoured in text simplification, whether it be T5-based lexical simplification as demonstrated in Sheang et al. (2022) [13] or controllable simplification using T5 in Saggion et al. (2021) [14]. Lexical simplification methods could also be made context-aware, as shown in Qiang et al. (2020) [15], which introduces LSBert. Transformer approaches to text simplification generally involve embedding-based simplification, as demonstrated in Trucia et al. (2023) [16].

AMR-based structural subgraph extraction and simplification was explored in Yao et al. (2024) [17] and showed better preservation on semantic fidelity compared to LLMs. Zaman et al. (2024) [18] combined simplification with summarization using the SATS model for conciseness and clarity. Beauchemin et al. (2023) [19] introduced MeaningBERT, a simplification metric that improves upon the limitations of traditional metrics like BLEU and SARI.

3. Methodology

The proposed S3 model is a hybrid system designed to perform simplification of biomedical text at sentence level. It performs simplification at three distinct and complementary levels: sub-lexical, structural (syntactic) and semantic. The model combines rule-based resources for simplifying the structure of the sentence for greater understanding and replaces complex phrases at a tokenized level with simpler alternatives. The pipeline improves readability and accessibility of complex tests using a semantic approach where paraphrasing is done while preserving meaning and domain relevance. The architecture of the pipeline is given in Fig 1

The pipeline has five main phases- Preprocessing, Lexical Simplification using WordNet and BERT Masking, Syntactic Simplification using a graph-based approach (AMR construction and parsing), Semantic Simplification using T-5 paraphrasing models and finally merging of the outputs from the lexico-structural and semantic branches. Each stage is modular and has its advantages that are hybridized for balanced and simplified output.

- Sub-Lexical Simplification focuses on simplifying complex phrases, words and terms with simpler alternatives that are more readable. In the pipeline we use WordNet and UMLS Medical Thesaurus

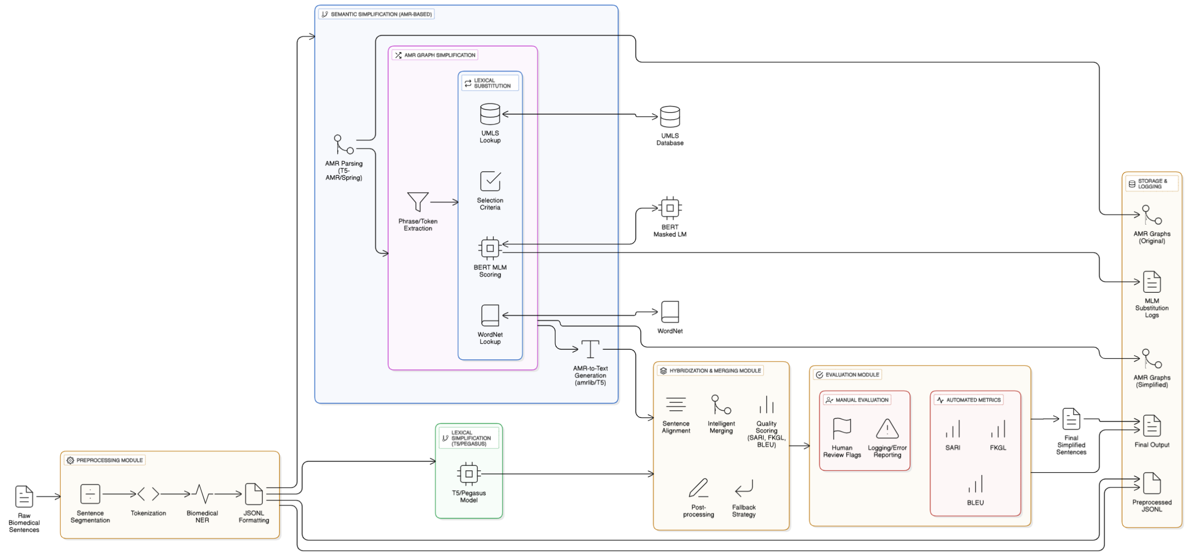

Figure 1: Architecture diagram of the S3 simplification pipeline

for finding simpler alternatives to complex medical terms. The appropriateness of the word is judged using BERT- Masking to select the most appropriate and simplified synonym for the complex term.

- Structural Simplification uses Abstract Meaning Representation graphs to manipulate the conceptual and syntactic of sentences. Once the graphs are created, we simplify the technical and complex terms to prune the unnecessary subtrees and then use various graph parsing techniques to reconstruct the sentence using AMR to Text generation. This step in hybridized strongly with Lexical simplification to discover the important terms in the graph and send them only for lexical substitution.
- Semantic Simplification is performed using a Pegasus (T5) model for sentence rewriting. It understands the meaning of a sentence and rewrites it in simpler phrases to enhance fluency. T5 models introduce semantic transformations that complement the lexico- structural transformation.

In the final stage we use various simplification metrics and gram-based overlap using evaluation metrics such as FKGL and SARI. The Hybridisation system merges phrases and segments with highest simplification efficacy by combining fine grained word level replacements with structural rearrangement while ensuring fluency with high level paraphrasing ensuring semantic cohesion. The S3-Pipeline has a balanced trade-off between structure, syntax, fluence and ensuring improved biomedical simplification.

4. Preprocessing Stage

Preprocessing in our pipeline is critical due to the domain-specificity of the vocabulary and complex sentence structures beacuse of the scientific nature of the text corpus. The document is split into sentences using tokenization techniques. These sentences are then tokenized at a word level and POS-tagged to facilitate with lexical replacement in the pipeline. The tokenization is performed using a combination of SCIBERT tokenizer [20] (for domain specificity) and SpaCy for NLP pipeline integration. This helps in simplifying the criteria for selecting candidates to be substituted as well as identifying the semantic blueprints for the AMR graphs using the AMRlib parser. Other steps in preprocessing involve removal of punctuation, lemmatization and citation markers.

5. Structural Simplification

Structural simplification is implemented in the pipeline using Abstract Meaning Representation (AMR), using a graphical method that is used to encode the semantics of a sentence independently of its syntax. The focus is on reconfiguring the structure of a sentence by identifying and modelling their conceptual scaffolding through AMR graph. This stage involves using semantic representations to rewrite sentences at the conceptual level, making the structure and complex terms simpler, while preserving the original meaning of the sentence.

5.1. AMR Graph Construction

Each sentence is parsed using the AMRlib pipeline which uses a Transformed based encoder-decoder architecture that is trained on the LDC AMR data to convert the sentence structure in a rooted, directed and acyclic graph that represents the syntactic structure of the sentence. The Nodes in the AMR graph represent the concepts that include entities, events and attributes while the edges represent the semantic roles like agents, patients and modifiers as well as temporal relations.

5.2. Concept Node Identification

After creation the AMR graph is traversed to identify key concept nodes that are most likely to carry added scientific complexity and require the most simplification. These include multiword scientific terms like myocardial infarction, predicates that represent complex relations and compound nominal modifiers.

These nodes are prioritized on the following principals-

- Graph role- This indicates whether they are the central arguments or are they peripheral modifiers that do not contribute much to the meaning of the sentence.
- Domain specificity – This indicates whether they can be generalised or need thesaurus-based domain specific substitution
- Lexical features- These features include NLP features like TF-IDF and relative importance across documents.

This approach ensures that the focus remains on the semantically complex part of the sentence that contributes most to the meaning, rather than simplifying the supporting phrases.

5.3. Lexical Simplification within AMR Graphs

We perform the lexical substitution using the AMR graphs only to ensure only the important phrases in the concept nodes are substituted so that we have maximum efficiency. The content nodes already store the most significant portions of the input text and thus require maximum simplification and by integrating lexical substitution within AMR parsing, we can make easily identify key phrases.

For each concept node in the AMR graph, the S-3 Pipeline:

- Identifies candidate lexical substitution phrases using Wordnet which is a lexical database that stores words as sets of synonyms called synsets. Only the synsets that match the POS- tag of the candidate phrase are used for the substitution i.e. nouns are replaced only by nouns and verbs are replaced only by verbs etc.
- The Synonyms that contain overly complex and technical terms or archaic terms are filtered out on basis of frequency and syllable complexity. They are also lemmatized to match the format of the candidate phrase in AMR graphs.
- To select the best candidate amongst the set of candidates we use contextual evaluation using BERT masked language model (MLM). In this model the original sentence is considered, and the concept node is replaced with a [MASK]. The BERT MLM assesses the correctness of each synonym in the place of [MASK] and assigns a SoftMax probability as a score. This score is a

combination of contextual fluency and grammatical correctness. Candidates with top-k scores are prioritized and further analysed on basis of simplicity heuristics. We calculate the PLL score as follows-

$$\mathrm{PLL}(x) = -\sum_{t=2}^{T} \log p(x_t \mid x_1, \ldots, x_{t-1}) \tag{1}$$

Where:

  - $x = (x_1, x_2, \ldots, x_T)$ is the tokenized sentence
  - $p(x_t \mid x_{<t})$ is the token-level conditional probability from BERT
  - PLL is the negative log-likelihood of the sentence under BERT's masked language model

- The top replacement candidates are now ranked on basis of simplicity- contextuality score that considers factors like – Word length based on syllabic complexity, psycholinguistic metrics like Age of Acquisition (AoA), familiarity and number of meanings (homophones) and lastly the BERT MLM score. The best candidate selection logic follows this formula:

$$\mathrm{BestCandidate} = \arg\max_{c \in \mathscr{C}} \mathrm{PLL}(x_{[p \to c]}) \tag{2}$$

Where:

  - $\mathscr{C}$ is the set of candidate substitutions (from WordNet/UMLS or any othe rmedical thesaurus or BERT top-k)
  - $x_{[p \to c]}$ is the sentence x with phrase p replaced by candidate c
  - $\mathrm{PLL}(\cdot)$ is defined in Equation 1

- Only the candidate synonyms that surpass both the simplicity threshold and the contextuality fit test are retained. If none of the synonyms pass this threshold then the original node remains unchanged to avoid loss of semantic meaning or fluency. Thus,

$$\mathrm{Accept}(c) = \begin{cases} \text{True}, & \text{if } \mathrm{PLL}(x_{[p \to c]}) > \mathrm{PLL}(x) \cdot (1 + \delta) \\ \text{False}, & \text{otherwise} \end{cases} \tag{3}$$

Where:

  - δ is the improvement threshold (set to 0.05)
  - x is the original sentence
  - $x_{[p \to c]}$ is the new sentence with candidate c

5.4. Graph Update and Node Substitution

- After a simplified synonym is selected for substituting the concept node, the AMR graph is updated with the new lexical substitution. Updating the graph ensures that the simplification is embedded directly within the semantic structure of the graph before we begin sentence reconstruction enabling preservation of integrity connectivity of the graph.
- Each concept node in a graph is represented using a surface string and a unique identifier variable. The pipeline locates the corresponding concept node in the Penman graph representation (AMR) and replaces it with the simplified synonym. Example: Node (m/myocardia-infarction) is simplified to (m/heart-attack).
- The multiword expressions that are used as replacements are made AMR compatible using hyphens and underscores to ensure the tokens remain a single concept in the graph. The updated term is lemmatized and normalised to fir AMR conventions. Special characters in the replacements are reformatted.
- Existing semantic edges like :ARGx, :mod etc are attached to the node and are preserved to ensure there is no alteration in the nides structural role in the graph and changes are only in the content of the nodes. This enables us to maintain the semantic relations in the data and prevents errors during sentence regeneration.

5.5. Surface Realization

- After the nodes of the AMR graph are simplified using node substitution, the final stage of this step is surface realization to convert the modified semantic graph back into a natural language sentence. The reconstructed sentence now contains the lexically simplified phrases.
- The sentence reconstruction is performed using AMR-to-text generation model based on a sequence-to-sequence Transformer architecture. The model linearizes the AMR graph into a string representation by using the PENAM notation and decodes it into well formatted text while handling ensuring fluency using word ordering, grammatical correctness and insertion of grammatical function words like determiners, prepositions etc.
- As only the node concepts were altered and the graph structure remained unaltered, the regenerated sentence retain the full semantic fidelity of the original sentences that were used to model the graph. Further the simplified terms are smoothly integrated adjusting tense and agreement as needed.
- Handling multiword expressions involves creation and alteration of the surrounding structure in the graph to ensure fluency. Example- For a replacement like "heart attack", the model contextually analyses the sentence and adjusts the surrounding to "suffering from a heart attack", depending on the surrounding structure of graph. The reconstructed sentence serves as the output of this stage.

6. Semantic Simplification

Parallelly to the lexico-syntactic processing, the S-3 Pipeline performs semantic-level simplification which involves transforming complex sentences into simpler versions by rephrasing and rewriting them at a high level of abstraction. We use Transformer-based encoder-decoder architecture, specifically a fine-tuned variant of the PEGASUS model for the paraphrasing. In this stage, we holistically work on the sentence meaning and capture the paraphrastic variation in the sentence to understand its core meaning. Once the meaning of the sentence (or intent) is understood, the model uses its knowledge base in natural language to find simplifications that express the intent while simplifying the sentence using more frequent vocabulary.

Pegasus is integrated in the pipeline to be used as a high-capacity neural simplifier, where its behaviour is guided through generation hyperparameters, and post-hoc evaluation heuristics. It enables the contextually natural rewriting of biomedical text. To maintain content fidelity, we employ controlled decoding strategies like beam search with Width 5 to ensures a diverse yet focused exploration of possible rewrites, maximum Length Constraint to avoid verbosity and enforce lexical economy and early stopping to halt generation once a high probability sequence is reached.

While the AMR-guided lexical and structural simplification phases help in improving precision and transparency, the semantic phase introduces holistic fluency which is missing in rule-based substitution systems. The deeper processing of core concepts is handled by the AMR- guided lexical approach while sentence structure and semantic simplification is handled by the Pegasus approach. This avoids unnecessary rewordings and allows Pegasus to focus on maximizing readability. The final output of the S-3 pipeline recognizes the individual strengths of the semantic approach and uses it for sentence framing, while lexico- graph output is used for phrase replacement.

7. Hybridization and Merging

To capture both structural fluency and semantic precision in simplification the S-3 pipeline adopts a hybrid simplification strategy that merges outputs from Pegasus-based semantic simplification and AMR-based structural-lexical rewriting. Both Pegasus and AMR-based models offer distinct advantages that can be used to improve quality of simplification-

- Pegasus is trained for paraphrasing and abstractive summarization and is adept at capturing semantic intent, paraphrasing complex constructions, and producing fluent output, but it struggles with over generalization and occasionally omits critical domain specific information. Sometimes if not tuned properly to the domain, it may oversimplify the document.
- AMR-based simplification, on the other hand, performs simplification in through structural and rule-based approach, making it structurally consistent and lexically transparent. It excels at preserving critical terms and decomposing them into simpler phrases, but its output may sound less fluent due to sentence reconstruction limitations.

The goal of merging is to combine the best of both approaches: Pegasus for semantic abstraction and fluency, and AMR-lexical simplification for domain faithfulness and structural precision. In most cases the lexical substitute candidate provides simpler phases but leaves the clause structure a bit rigid whereas the semantic candidate can rephrase more freely.

The fusion model must be fed the lexical and semantic inputs and must explicitly know which input did a token come from. This is done by using <L> and <S> tags for Lexical and Semantic tokens respectively and build an input sequence like Lexical: <L> All tokens <L: <S>Semantic<S> The model's encoder builds contextual embeddings that intermix information from Lexical and Semantic tokens. During decoding, when the model generates token yt, its attention heads can attend jointly to simplifying synonyms and fluent restructuring. Thus the merging works in three steps

A multi-source sequence-to-sequence generator (e.g., T5) implements the following abstract process:

- Encoding: Each token from either stream is mapped to a vector representation. As the tokens appear in the fusion prompt the encoder can attend across them.
- Contextualization: The model learns what parts of the <L> and <S> should be combined using stacked elf attention layers. It takes simplified phrases from <L> and structure from <S>. The models attention heads can link these two.
- Decoding: The decoding layer consults the combined encoded representation and decides what token is to be emitted (Ex: Copies, re-interpreted phases or unnecessary connectives). Greedy decoding is used via beam search to select top probable next tokens.

8. Evaluation and Selection

Though the fusion step itself produces a merged candidate M, the system compares MM against the individual outputs of semantic and lexical states to cover cases where strong vocabulary-based replacements or meaning based replacements might be more optimal.

8.1. Metrics Used:

1. BLEU (Bilingual Evaluation Understudy)
   BLEU measures n-gram overlap between two sentences. It counts the unigrams, bigrams, n-grams, etc., that occur in both the candidate and the reference. We use BLEU to compare the lexical and semantic simplifications to estimate consistency and mutual agreement. A high BLEU score indicates that the simplifications are consistent.
2. SARI (System output Against References and Input)
   SARI evaluates how well the simplified sentence has added, deleted, or retained content relative to the complex input sentence as well as reference simplifications. It measures the quality of additions, keeps, and deletions.
   - When evaluating the lexical output, the Pegasus output is treated as the reference.
   - For the semantic output, the AMR output is used as the reference.
   - For the final merged output, both Pegasus and AMR outputs are treated as references along with the original sentence.

   A high SARI score ensures a good balance between information retention and simplification.

3. FKGL (Flesch–Kincaid Grade Level)
   FKGL is used to measure the readability of the simplified sentence. It specifically indicates how many years of schooling are required to understand the sentence. It is based on the number of syllables and word length. A lower FKGL score indicates lower complexity.
4. Length Ratio
   This is simply the ratio of the length of the simplified sentence to the original complex sentence. Ideally, a ratio between 0.65 and 0.85 is desirable to maintain a balance between over-simplification and verbosity.
5. LM Fluency Score (Language Model Perplexity)
   This measures the fluency of a sentence according to a pretrained language model. It uses pseudo-perplexity (the negative log-likelihood of a token given the previous tokens). A lower perplexity score indicates a more fluent sentence. We evaluate the fluency of each token to avoid generating ungrammatical sentences.

8.2. Weighted Scoring

We aggregate these metrics into one single metric called "Universal Metric". Whichever candidate has the lowest universal score is selected. During several runs we found the hybrid output to be the most likely (84%) to be selected. A simplification candidate with high BLEU score, lower FKGL score, Length Ratio around 0.75 (to avoid aggressive overcutting and verbosity) and a high SARI score is adjudged to be a good simplification.

$$\begin{aligned}
\text{Score} = \;& 0.20 \cdot (1 - \text{BLEU}) + 0.20 \cdot \left(\frac{\text{FKGL}}{\text{FKGL} + 10}\right) \\
& + 0.10 \cdot |\text{LR} - 0.75| + 0.10 \cdot \left(\frac{\text{PPL}}{\text{PPL} + 100}\right) \\
& + 0.30 \cdot \left(1 - \frac{\text{SARI}}{100}\right) \\
\text{where:} \quad & \text{BLEU} = \text{BLEU similarity between candidates}, \\
& \text{FKGL} = \text{Flesch–Kincaid Grade Level (lower is simpler)}, \\
& \text{LR} = \text{Length Ratio} = \frac{\text{len(simple)}}{\text{len(complex)}}, \\
& \text{PPL} = \text{Pseudo-perplexity}, \\
& \text{SARI} = \text{SARI simplification score (0–100)}.
\end{aligned}$$

9. Results

9.1. Task 1.1: Sentence-Level Simplification

We evaluated our S-3 pipeline on the provided test datasets and assessed the performance using the metrics we defined above. We also calculated change in FKGL (ΔFKGL) to understand the improvement in meaning using our model and ROUGE-L.

As shown in Table 1, the system produces consistent and medically accurate simplifications. The dataset has highest ΔFKGL, indicating that it has high level of effective simplifications. In Biomedical systems a SARI of around 15 to 20 is deemed desirable to preserve meaning during simplification while maintaining terminology accuracy and not rephrasing medical terms. The stable SLR scores indicate structural and lexical integrity.

9.2. Task 1.2: Paragraph-Level Simplification

We further evaluated our S-3 model on paragraph-level input under Task 1.2. These results show that our model maintains strong simplification quality while preserving essential biomedical content over

Table 1
Biomedical Text Simplification Results across Dataset (Task 1.1)

Dataset	BLEU	SARI	FKGL	ΔFKGL	CR	SLR	ROUGE-L	Universal Score
1K	0.2782	17.72	10.38	2.58	0.770	0.752	0.534	0.543
5K	0.2696	17.30	10.08	2.76	0.772	0.751	0.531	0.544
Full	0.2806	17.77	10.32	3.14	0.776	0.750	0.544	0.547

longer spans of text. (Refer Table 2)

Table 2
Biomedical Text Simplification Results across Dataset (Task 1.2)

Model	Data Size	FKGL	ΔFKGL	SARI	Compression Ratio	ROUGE-L	Universal Score
Hybrid	1K	10.21	2.77	17.43	0.785	0.540	0.548
Hybrid	5K	10.11	2.74	17.21	0.787	0.546	0.549
Hybrid	Full	10.34	3.12	17.68	0.794	0.557	0.546

Table 3
MeaningBERT Scores for Sentence- and Paragraph-Level Simplification (Hybrid Model)

Task	Dataset	MeaningBERT Score
Sentence-Level (Task 1.1)	1K	0.7932
	5K	0.7845
	Full	0.7911
Paragraph-Level (Task 1.2)	1K	0.7998
	5K	0.8004
	Full	0.8038

9.3. Result Analysis

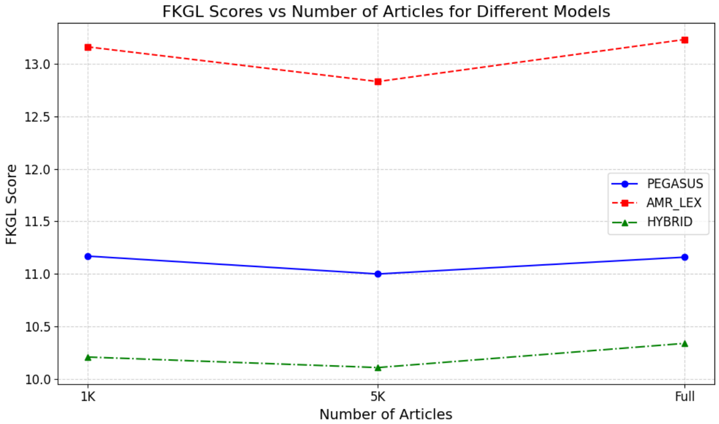

Figure 2: FKGL Scores for various methods on the 9226 sentences of the Task 1.2 Dataset

- **FKGL scores** (Refer Figure 2) remained consistently around 10. This suggests that the output is simple enough to be understood by non-domain experts like patients and general readers, while still preserving the source content. An FKGL around 10 is considered optimal for the biomedical domain, as excessive simplification could lead to a loss of precision.

Table 4
MeaningBERT Scores for Sentence- and Paragraph-Level Simplification (Hybrid Model)

Task	Dataset	MeaningBERT Score
Sentence-Level (Task 1.1)	1K	0.7932
	5K	0.7845
	Full	0.7911
Paragraph-Level (Task 1.2)	1K	0.7998
	5K	0.8004
	Full	0.8038

- **ΔFKGL scores**, ranging from 2.5 to 3.1, indicate a substantial improvement in readability across both subtasks. This increase—especially in the paragraph dataset—demonstrates that the S-3 pipeline is highly effective at simplifying longer and more complex sentences without compromising their meaning.
- **SARI scores** of around 17 across both tasks are ideal for the biomedical domain. While modest compared to general-domain simplification, they are appropriate here because aggressive editing is discouraged. A SARI range of 16–20 reflects a balanced simplification that maintains semantic integrity.
- **SLR scores** of approximately 0.75 show that although lexical simplification is applied, essential terminology, abbreviations, and structural elements are retained. This minimizes the risk of factual drift and preserves key biomedical meaning.
- **Compression ratio (CR)** values mostly fall within the ideal range of 0.65 to 0.85. This ensures that important information is not lost and the simplified output is not overly verbose. The moderate compression supports reduced complexity while maintaining content integrity. The stability of CR across datasets and tasks indicates controlled and consistent simplification.
- **MeaningBERT scores** (Table 4) are especially high—around 0.8—for paragraph-level simplification. This suggests the model excels at simplifying large or compound sentences. Combined with a solid compression ratio, this confirms that the S-3 pipeline performs well in preserving semantic clarity while reducing linguistic complexity.

10. Conclusion

In this paper we presented the **S-3 Pipeline** that is a hybrid system for biomedical text simplification that seamlessly integrates semantic, sub-lexical and structural simplification through T-5 models, AMR- parsing and pruning and BERT- guided lexical substitution. We evaluated the results across multiple metrics like FKGL, SARI and MeaningBERT that demonstrated the pipeline achieves a strong balance between information retrieval, simplification, conciseness and readability. The S-3 pipeline is a promising tool that can be used to improve accessibility and comprehension of bio-medical literature. Future work may explore tighter integration parameters for the structural and lexical modules and extending the system for multilingual biomedical texts.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools for the research and while writing this paper. 1 is partially generated using Generative-AI tools.

References

[1] L. Ermakova, H. Azarbonyad, J. Bakker, B. Vendeville, J. Kamps, Clef 2025 simpletext track: Simplify scientific text (and nothing more), in: European Conference on Information Retrieval, Springer, 2025, pp. 425–433.

[2] G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025: Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS.org, 2025.

[3] L. Ermakova, H. Azarbonyad, J. Bakker, B. Vendeville, J. Kamps, Overview of the CLEF 2025 SimpleText track: Simplify scientific texts (and nothing more), in: J. Carrillo de Albornoz, J. Gonzalo, L. Plaza, A. García Seco de Herrera, J. Mothe, F. Piroi, P. Rosso, D. Spina, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Sixteenth International Conference of the CLEF Association (CLEF 2025), Lecture Notes in Computer Science, Springer, 2025.

[4] K. Attal, B. Ondov, D. Demner-Fushman, A dataset for plain language adaptation of biomedical abstracts, Scientific Data 10 (2023) 8. URL: https://doi.org/10.1038/s41597-022-01920-3. doi:`10.1038/s41597-022-01920-3`.

[5] J. Bakker, J. Kamps, Cochrane-auto: An aligned dataset for the simplification of biomedical abstracts, in: M. Shardlow, H. Saggion, F. Alva-Manchego, M. Zampieri, K. North, S. Štajner, R. Stodden (Eds.), Proceedings of the Third Workshop on Text Simplification, Accessibility and Readability (TSAR 2024), Association for Computational Linguistics, Miami, Florida, USA, 2024, pp. 41–51. URL: https://aclanthology.org/2024.tsar-1.5/. doi:`10.18653/v1/2024.tsar-1.5`.

[6] S. Joseph, K. Kazanas, K. Reina, V. J. Ramanathan, W. Xu, B. C. Wallace, J. J. Li, Multilingual simplification of medical texts, 2023. URL: https://arxiv.org/abs/2305.12532. `arXiv:2305.12532`.

[7] B. Ondov, K. Attal, D. Demner-Fushman, A survey of automated methods for biomedical text simplification, Journal of the American Medical Informatics Association 29 (2022) 1976–1988. doi:`10.1093/jamia/ocac149`.

[8] Z. Li, S. Belkadi, N. Micheletti, L. Han, M. Shardlow, G. Nenadic, Large language models for biomedical text simplification: Promising but not there yet, 2024. URL: https://arxiv.org/abs/2408.03871. `arXiv:2408.03871`.

[9] V. Knappich, S. Razniewski, A. Friedrich, Boschai @ plaba 2023: Leveraging edit operations in end-to-end neural sentence simplification, arXiv preprint arXiv:2311.01907 (2023).

[10] L. J. Y. Flores, H. Huang, K. Shi, S. Chheang, A. Cohan, Medical text simplification: Optimizing for readability with unlikelihood training and reranked beam search decoding, in: Findings of EMNLP 2023, ACL, 2023, pp. 4859–4873.

[11] A. Phatak, D. W. Savage, R. Ohle, J. Smith, V. Mago, Medical text simplification using reinforcement learning (teslea): Deep learning–based text simplification approach, JMIR Medical Informatics 10 (2022) e38095. doi:`10.2196/38095`.

[12] J. Gill, M. Chetty, S. Lim, J. Hallinan, Knowledge-based intelligent text simplification for biological relation extraction, Informatics 10 (2023) 89. doi:`10.3390/informatics10040089`.

[13] K. C. Sheang, D. Ferrés, H. Saggion, Controllable lexical simplification for English, in: S. Štajner, H. Saggion, D. Ferrés, M. Shardlow, K. C. Sheang, K. North, M. Zampieri, W. Xu (Eds.), Proceedings of the Workshop on Text Simplification, Accessibility, and Readability (TSAR-2022), Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Virtual), 2022, pp. 199–206. URL: https://aclanthology.org/2022.tsar-1.19/. doi:`10.18653/v1/2022.tsar-1.19`.

[14] K. C. Sheang, H. Saggion, Controllable sentence simplification with a unified text-to-text transfer transformer, in: Proceedings of the 14th International Conference on Natural Language Generation, 2021, pp. 341–352. doi:`10.18653/v1/2021.inlg-1.38`.

[15] J. Qiang, Y. Li, Y. Zhu, Y. Yuan, X. Wu, Lsbert: A simple framework for lexical simplification, CoRR abs/2006.14939 (2020).

[16] C.-O. Truică, A.-I. Stan, E.-S. Apostol, Simplex: a lexical text simplification architecture, CoRR abs/2304.07002 (2023).

[17] P. Yao, K. Guzhva, D. Barbosa, Semantic graphs for syntactic simplification: A revisit from the age

of llm, in: Proceedings of TextGraphs-17, Association for Computational Linguistics, 2024, pp. 105–115.

[18] F. Zaman, F. Kamiran, M. Shardlow, S.-U. Hassan, A. Karim, N. R. Aljohani, Sats: Simplification aware text summarization of scientific documents, Frontiers in Artificial Intelligence 7 (2024) 1375419. doi:`10.3389/frai.2024.1375419`.

[19] D. Beauchemin, H. Saggion, R. Khoury, Meaningbert: assessing meaning preservation between sentences, Frontiers in Artificial Intelligence 6 (2023) 1223924.

[20] I. Beltagy, A. Cohan, K. Lo, Scibert: Pretrained contextualized embeddings for scientific text, CoRR abs/1903.10676 (2019). URL: http://arxiv.org/abs/1903.10676. `arXiv:1903.10676`.