SSN Tokatrons at the CLEF 2026 SimpleText Track: Plan-Guided BART and Zero-Shot LLM Approaches to Biomedical Text Simplification*

Notebook for the SimpleText Lab at CLEF 2026

Sujith M[1], Sree Krishna S[1], Varghese K James[1] and Prabavathy Balasundaram[1]

[1]Sri Sivasubramaniya Nadar College of Engineering, Chennai, India

Abstract

This paper presents the Tokatrons team's participation in CLEF 2026 SimpleText Task 1, focusing on the automated simplification of biomedical texts at the sentence level. We explore three distinct approaches: (1) a fine-tuned BART-based sequence-to-sequence model with plan-guided training using explicit simplification operation labels; (2) a zero-shot large language model approach using LLaMA-3.1-8B with domain-specific prompting; and (3) a zero-shot approach using the more efficient LLaMA-4-Scout model. Our results show a trade-off between simplification quality and computational cost. The plan-guided BART model achieved an official test SARI score of 34.30, showing that explicit plan labels improve the training process. The zero-shot LLaMA-3.1-8B model achieved a superior validation SARI of 36.29, but its computational requirements made it impractical for full test-set inference within our budget constraints. The more efficient LLaMA-4-Scout model achieved only 17.92 SARI, substantially underperforming both alternatives. Our findings highlight that fine-tuned medium-sized models with structured training strategies remain competitive alternatives to large language models for text simplification, especially when computational resources are constrained.

Keywords

Text simplification, biomedical NLP, BART, LLaMA, plan-guided generation, CLEF SimpleText

1. Introduction

Biomedical text simplification is the task of transforming complex medical and scientific language into more accessible forms without losing critical information. It addresses a significant barrier in healthcare communication: patients, caregivers, and even non-specialist healthcare providers often struggle to understand the dense technical language found in medical literature, clinical guidelines, and systematic reviews [1].

The CLEF 2026 SimpleText Task 1 [2] formalizes this challenge as a sentence-level simplification task. Given a complex sentence from a biomedical source, the goal is to produce a simplified version that preserves the core meaning while improving readability for a general audience. The task uses the Cochrane Database of Systematic Reviews as its source domain, which contains highly technical sentences summarizing medical evidence.

In this paper, we describe the three approaches developed by the Tokatrons team for this task. First, we fine-tune a BART-large-CNN model [3] with plan-guided training, where explicit simplification operation labels (rephrase, delete, copy, merge, split) are prepended to source sentences during training.

CLEF 2026 Working Notes, 21 – 24 September 2026, Jena, Germany

*Submitted as a working note to the SimpleText Task 1 shared task at CLEF 2026. Code and models are publicly available at https://github.com/sujith0613/tokatrons-clef2026-simpletext.

Sujithmaris@gmail.com (S. M); sreekrishnassk166@gmail.com (S. K. S); varghesekj18@gmail.com (V. K. James); prabavathyb@ssn.edu.in (P. Balasundaram)

https://github.com/sujith0613 (S. M); https://github.com/SSK166 (S. K. S); https://github.com/vicfic18 (V. K. James); https://www.ssn.edu.in/computer-science-and-engineering-department/faculty/dr-b-prabavathy-associate-professor/ (P. Balasundaram)

0009-0001-0671-6195 (S. M); 0009-0000-2362-165X (S. K. S); 0009-0002-9977-768X (V. K. James); 0000-0003-3903-0410 (P. Balasundaram)

Second, we evaluate zero-shot LLaMA-3.1-8B [4] with domain-specific prompting via the Groq API. Third, we assess zero-shot LLaMA-4-Scout [5] as a more computationally efficient alternative.

Our results compare three approaches: BART fine-tuning achieves 34.30 SARI on the official test set; LLaMA-8B achieves 36.29 SARI on validation but was not evaluated on the test set due to API costs; and LLaMA-4-Scout achieves 17.92 SARI. We discuss the trade-offs between quality and cost across these approaches.

2. Related Work

Text simplification at CLEF has shifted toward LLM-based approaches. The SimpleText 2024 track overview [6] documents that 15 teams submitted 83 runs for Task 3, with the majority adopting decoder-only LLMs rather than fine-tuned encoder-decoder models.

The CLEF 2023 Best of Labs paper [7] (UZH Pandas, University of Zurich) fine-tuned Alpaca LoRA 7B on simplification data and generated candidates using six prompt templates, re-ranked via Minimum Bayes Risk decoding with LENS [8]. Their ablation study showed that the reference-less Simplicity Level Estimate (SLE) metric fails to penalise ungrammatical output.

In CLEF 2024, the USM AIIR team [9] (University of Southern Maine) applied LLaMA3 and Mistral across all SimpleText tasks, finding that prompt design (system prompt and task framing) was the key differentiator, not model scale. The DS@GT team [10] (Georgia Tech, CLEF 2025) later proposed a two-stage LLM framework: generate a structured plan, then execute plan-driven simplification; architecturally similar to our plan-guided BART, but using the LLM itself as the planner rather than an external classifier.

Encoder-decoder models [11] (BART, T5) use bidirectional encoding with autoregressive decoding, requiring task-specific fine-tuning. Decoder-only LLMs (LLaMA, GPT, Mistral) use pure autoregressive generation with in-context learning. The BLESS benchmark [12] showed that LLM-based simplification achieves higher human-judged quality than fine-tuned encoder-decoder models. Our results are consistent: LLaMA-8B improves FKGL from 13.03 to 11.31 while BART's FKGL of 16.00 represents a degradation. BART deletes only 7.3% of source tokens versus 33.7% for LLaMA-8B against a training distribution of 35.3% delete labels.

3. Task Description

SimpleText Task 1 [2] targets sentence-level biomedical text simplification. Given a source sentence s from a biomedical text, the system must generate a simplified sentence t that:

- Preserves the core factual content of the original sentence.
- Reduces lexical and syntactic complexity to improve readability for a non-specialist audience.
- Maintains grammatical correctness and fluency.

The evaluation framework uses four primary metrics. SARI [13], the main ranking metric, measures the quality of simplification operations (add, keep, delete) by comparing the system output against the source and reference simplifications. BLEU [14] measures n-gram overlap with the reference. BERTScore [15] computes semantic similarity using contextual embeddings from BERT. FKGL [16] estimates the grade level required to understand the text. Additional secondary metrics include compression ratio, deletion proportion, and the fraction of outputs identical to the source.

The official test set comprises 48,809 sentence pairs from the Cochrane database. Submissions are evaluated on the CodaBench platform, which computes all metrics automatically upon submission.

Table 1
Distribution of simplification operation labels in the training set.

Operation	Count	Percentage
Rephrase	5,212	45.3%
Delete	4,064	35.3%
Copy	967	8.4%
Merge	748	6.5%
Split	519	4.5%
Total	11,510	100.0%

4. Dataset

4.1. Cochrane-Auto Corpus

We use the Cochrane-auto dataset provided by the SimpleText organizers. The dataset consists of sentence-level (complex, simplified) pairs extracted from the Cochrane Database of Systematic Reviews. The training split contains 11,510 pairs, with a validation set of 9,160 sentences (derived from the SimpleText 2025 test set). The official 2026 test set contains 48,809 sentences for blind evaluation.

Training source sentences average 25.1 tokens (median 22, std 16.4, range 1–215). Target simplifications average 14.5 tokens (median 13, std 15.3, range 0–145). The median compression ratio is 1.55. The source vocabulary comprises 24,198 unique tokens, of which 15,002 appear in the simplified targets. Empty targets (where the simplified sentence is an empty string) account for 38.5% of the raw data. After filtering and label reassignment (described in §4.3), this proportion reduces to 28.8% as some empty-target entries are reassigned to rephrase, merge, or split labels through the alignment process.

4.2. Plan Labels

Following the plan-guided generation paradigm [17, 18], we annotate each training pair with a simplification operation label. The labels are derived by aligning the source and target sentences using a rule-based alignment algorithm provided by the SimpleText organizers. Five operation types are identified:

- **Rephrase**: The target paraphrases the source using simpler vocabulary or syntax.
- **Delete**: The target removes content present in the source.
- **Copy**: The source and target are identical.
- **Merge**: The target combines information from multiple source sentences.
- **Split**: The target breaks a single source sentence into multiple sentences.

The dominance of rephrase and delete operations (80.6% combined, Table 1) reflects that biomedical text simplification primarily involves lexical simplification and selective information removal. The 8.4% copy instances correspond to cases where the source is already simple enough. The 4.5% split and 6.5% merge operations handle complex cases requiring structural reorganisation.

4.3. Label Derivation

For 7,225 training pairs (62.8%), the reference simplification is available and labels are derived by alignment. For 967 pairs (8.4%), the source and target are identical, receiving the copy label. For 3,318 pairs (28.8%) where the target simplification is empty, the delete label is assigned, reflecting that the source content should be entirely removed.

Table 2
Training hyperparameters for the BART model.

Parameter	Value
Base model	`bart-large-cnn`
Optimizer	AdamW
Learning rate	3e-5
Weight decay	0.01
Epochs	5
Warmup ratio	0.1
Batch size	8 (gradient accumulation: 2)
Max source length	512
Max target length	128
FP16	True
Gradient clipping	1.0
Label smoothing	0.1
Random seed	42
Inference (best config)	
num_beams	2
length_penalty	1.0
no_repeat_ngram_size	0

5. Methods

5.1. Approach 1: Plan-Guided BART

5.1.1. Base Model

We use `facebook/bart-large-cnn` [3], a sequence-to-sequence transformer with approximately 400 million parameters. BART uses a denoising autoencoding objective: it corrupts text with a noising function and learns to reconstruct the original. The large-CNN variant is fine-tuned on CNN/DailyMail for text summarization, a related abstractive compression task.

5.1.2. Plan-Guided Training

We extend BART with plan-guided training [17]: five special tokens (`<rephrase>`, `<delete>`, `<copy>`, `<merge>`, `<split>`) are added to the vocabulary and embedding layer. During training, the correct operation token is prepended to the source sentence: `<rephrase> The patient exhibited...`. The model learns to use this token as a planning signal that conditions the generation process on the expected operation type.

We fine-tune for 5 epochs with plan tokens added to the vocabulary (Table 2).

5.1.3. Inference Tuning

We perform a grid search over inference hyperparameters on the validation set, exploring:

- Beam size: 1–5
- Length penalty: 0.6–1.4
- No-repeat n-gram size: 0–3

The best configuration (num_beams=2, length_penalty=1.0, no_repeat_ngram_size=0) achieves an oracle SARI of 35.86 when the correct plan label is provided.

5.1.4. Label Classifier

At inference time, the plan label is not available. We train a separate RoBERTa-base classifier [19] to predict the operation label from the source sentence alone. The classifier is fine-tuned on the 11,510 training pairs for 3 epochs with a learning rate of 2e-5 and batch size of 16. It achieves 52.6% accuracy across the five classes, above the random baseline (20%). Nearly half of all predictions select the wrong operation label.

5.2. Approach 2: Zero-Shot LLaMA-3.1-8B

5.2.1. Model and Setup

We evaluate LLaMA-3.1-8B [4] in a zero-shot setting via the Groq API. The model is accessed as `llama-3.1-8b-instant` and receives no fine-tuning or in-context examples.

5.2.2. Prompting Strategy

The system prompt defines the task and output constraints:

> "You are a medical text simplifier. Rewrite the sentence in simpler language for a general audience. RULES: - Output ONLY the rewritten sentence, nothing else - Keep sentences short and clear - Preserve all important numbers, statistics, and measurements"

The user prompt provides the source sentence to be simplified.

5.2.3. Inference Configuration

Generation uses temperature 0.3 (with automatic fallback to 0.7 if the output is identical to the input, triggering a single regeneration), top-p 0.9, and a maximum of 128 tokens. The validation set of 9,160 sentences (the SimpleText 2025 test set, reused as the 2026 validation split) is processed through the Groq free-tier API, which imposes rate limits of approximately 30 requests per minute.

5.3. Approach 3: Zero-Shot LLaMA-4-Scout

5.3.1. Rationale

Given the API cost and rate limitations of LLaMA-3.1-8B, we evaluate LLaMA-4-Scout [5] as a more computationally efficient alternative. LLaMA-4-Scout (`meta-llama/llama-4-scout-17b-16e-instruct`) is a mixture-of-experts (MoE) model with approximately 17 billion active parameters.

5.3.2. Inference Pipeline

The same prompt strategy is used as for LLaMA-8B, with temperature 0.3 and max tokens 128. The full test set of 48,809 sentences is processed via the Groq API.

6. Experimental Setup

6.1. Computational Resources

All fine-tuning and validation experiments are conducted on a single NVIDIA T4 GPU (16 GB VRAM) provided through Google Colab. LLaMA inference is performed exclusively through the Groq API. BART test-set inference is performed on a T4 GPU (Google Colab GPU runtime).

Table 3
Comparison of BART and LLaMA-3.1-8B on the validation set.

Metric	BART	LLaMA-3.1-8B
SARI	33.23	**36.29**
BLEU	**6.89**	2.21
BERTScore	**0.634**	0.610
FKGL	16.00	**11.31**
Compression Ratio	**2.85**	1.18
Deletion Proportion	0.073	**0.337**
Identical to Source	0.005	**0.000**

Table 4
Official CodaBench test set results with post-hoc analysis.
SARI from official CodaBench evaluation. Remaining metrics computed on English subset of test data.

Submission	SARI	BLEU	BERTScore	FKGL	Comp.
BART	**34.30**	2.10	**0.622**	16.35	**2.18**
LLaMA-4-Scout	17.92	**2.29**	0.510	**12.03**	1.00

6.2. Evaluation Metrics

We use the official SimpleText evaluation suite, which computes:

- **SARI** [13]: The primary metric, comparing system output against source and reference along three axes: added n-grams, kept n-grams, and deleted n-grams. SARI is the arithmetic mean of F1 scores for these three operations, computed at the n-gram level (n=1,2,3,4).
- **BLEU** [14]: Measures n-gram precision against the reference, with a brevity penalty.
- **BERTScore** [15]: Computes pairwise cosine similarity between token embeddings of system output and reference using BERT.
- **FKGL** [16]: The Flesch-Kincaid Grade Level, estimating the US school grade required to understand the text.
- **Compression Ratio**: The ratio of source length to system output length (in tokens).
- **Deletion Proportion**: The fraction of source tokens absent from the system output.
- **Identical to Source**: The fraction of system outputs that are identical to the source sentence.

6.3. CodaBench Submission

Official test submissions are managed through the CodaBench platform. Two submissions are made:

1. `tokatrons_task11_BART_20261.zip`: Plan-guided BART predictions.
2. `tokatrons_task11_LLM.zip`: LLaMA-4-Scout predictions.

7. Results and Discussion

7.1. Results

BART achieves 33.23 SARI with predicted plan labels. LLaMA-8B achieves 36.29, a gain of 3.06 points. LLaMA-8B produces simpler text (FKGL 11.31 vs. 16.00) with more aggressive deletion (33.7% vs. 7.3%). BART stays closer to the source vocabulary and structure, achieving higher BLEU (6.89) and BERTScore (0.634). BART's FKGL of 16.00 exceeds the source average of 13.03. The model increases reading difficulty despite training on simplification pairs.

Table 5
Grid search over BART inference hyperparameters on the validation set.

Beams	Len. Penalty	No-repeat Ngram	SARI
1	0.6	0	34.12
2	**1.0**	**0**	**35.86**
3	1.0	2	35.21
4	1.0	3	34.98
5	0.8	0	34.67

BART achieves 34.30 SARI on the official test (Table 4), confirming the validation result. LLaMA-4-Scout achieves 17.92 SARI. Its BERTScore of 0.510 shows substantial semantic drift from the reference. LLaMA-3.1-8B was evaluated only on the validation set.

A grid search over BART inference hyperparameters identified beam size 2, length penalty 1.0, and no-repeat n-gram size 0 as the optimal configuration.

This configuration reaches 35.86 oracle SARI when the correct plan label is provided.

7.2. Analysis

BART achieves FKGL 16.00 on validation and 16.35 on the test set, both above the source mean of 13.03. The model compresses and rephrases using biomedical vocabulary but does not replace complex terms with simpler alternatives. LLaMA-8B reduces FKGL to 11.31.

BART deletes 7.3% of source tokens. The training set is 35.3% delete labels. The RoBERTa plan classifier achieves 52% accuracy. Misclassifications cause the model to under-delete, particularly on inputs that should receive the delete label. The test run bypassed the classifier entirely, using a fixed `[REPHRASE]` label with greedy decoding.

LLaMA-8B's deletion proportion is 33.7%, close to the training distribution of 35.3%. The model sometimes deletes information that should be preserved. The BLEU score is 2.21. LLaMA-8B paraphrases aggressively, producing output with low n-gram overlap with the reference.

At Groq's paid-tier rates ($0.05 per million input tokens and $0.08 per million output tokens), processing the full 48,809-sentence test set costs approximately $0.27 per pass. Free-tier rate limits prevent iterative development at this scale.

LLaMA-4-Scout achieves 17.92 SARI, below both other models. The compression ratio of exactly 1.00 means the model produces output of the same length as the input and defaults to near-identity copying. The MoE routing may distribute biomedical tokens across general-purpose experts rather than domain-specific ones.

The architectural difference between encoder-decoder and decoder-only models explains these patterns. BART requires an explicit external mechanism to guide operation selection. The RoBERTa classifier's 52% accuracy causes errors that cascade into incorrect BART outputs. LLMs integrate operation selection intrinsically through the prompt context, avoiding this cascading error. The CLEF 2025 DS@GT approach [10] extends this further by using the LLM to generate explicit structured plans, replacing the external classifier with the LLM's own reasoning capabilities.

BART achieves higher BLEU (6.89 vs. 2.21) and BERTScore (0.634 vs. 0.610) because its encoder constrains the output to stay close to the source representation. The encoder constraints preserve faithfulness but limit simplification. LLMs achieve higher SARI (36.29 vs. 33.23) through more aggressive paraphrases that align better with human simplification patterns. The UZH Pandas study [7] showed that this trade-off can be managed through multi-prompt generation and MBR decoding.

Table 6 summarises the practical comparison. BART provides competitive SARI at low computational cost. LLaMA-8B achieves higher SARI but requires API access and has latency constraints. LLaMA-4-Scout underperforms on this task despite being a newer architecture.

Table 6
Cost–quality comparison across approaches.

Approach	SARI	Compute Cost	Scalability
BART	34.30	$0 (Colab free)	Full test
LLaMA-3.1-8B	36.29*	~$0.05 (Groq)	Limited
LLaMA-4-Scout	17.92	~$0.88 (Groq)	Full test

*Validation only; not submitted to test.

7.3. Positioning Among Prior Work

The plan-guided BART model follows the paradigm established by the UZH Pandas team [7]: incorporate structured planning into the generation process. Where they used multi-prompt MBR decoding with LLMs, we use explicit operation labels with a fine-tuned encoder-decoder model, a design choice motivated by the computational constraints of a single T4 GPU. Our LLaMA-8B results confirm the trend observed across all recent SimpleText editions [6, 9]. Zero-shot LLMs outperform fine-tuned task-specific models on the primary metric (SARI) but introduce new failure modes: semantic drift, cost, and scalability. The LLaMA-4-Scout result shows that MoE architectures may not transfer well to domain-specific tasks without adaptation.

Table 3 shows that LLaMA-8B achieves higher SARI (36.29 vs. 33.23), while BART maintains higher BERTScore (0.634 vs. 0.610) and BLEU (6.89 vs. 2.21) at lower computational cost. The DS@GT approach [10] of using LLMs as plan generators for smaller decoders is a direction we identify as future work.

8. Conclusion and Future Work

We presented three approaches to CLEF 2026 SimpleText Task 1: plan-guided BART (34.30 SARI), zero-shot LLaMA-3.1-8B (36.29 SARI on validation), and zero-shot LLaMA-4-Scout (17.92 SARI). Our results demonstrate that fine-tuned medium-sized models with structured training strategies remain competitive for biomedical text simplification, particularly when computational resources are constrained. The BART model's primary limitation is its failure to improve readability (FKGL 16.00 vs. source 13.03 on validation). Future work should address direct optimisation of simplification quality metrics. The LLaMA-8B results show the promise of large language models for this task, but practical deployment is limited by API costs and rate constraints. The LLaMA-4-Scout results show that architectural efficiency does not guarantee task performance for domain-specific simplification. We release our code and plan-label dataset at https://github.com/sujith0613/tokatrons-clef2026-simpletext and our trained models on the Hugging Face Hub: BART (standard plan) [1], BART (plan-guided) [2], BART (plan-guided extended) [3], RoBERTa plan classifier [4], DeBERTa plan classifier [5], DistilBERT plan classifier [6], and the balanced RoBERTa classifier [7] to support reproducibility and future research.

Improving plan classifier accuracy with larger architectures (e.g., DeBERTa-v3) or multi-task learning could reduce the classifier bottleneck. Directly optimising readability through FKGL-aware loss functions or reinforcement learning from readability feedback could address the FKGL degradation in BART. Exploring hybrid approaches that use LLaMA-8B as a plan label annotator for BART training could bridge the gap between the two approaches. Domain-adapting LLaMA-4-Scout could assess whether MoE performance stems from misalignment or fundamental limitations. Extending to document-level simplification (SimpleText Task 1.2) remains for future work.

[1] https://huggingface.co/winner0613/tokatrons-bart-cochrane-11
[2] https://huggingface.co/winner0613/tokatrons-bart-plan-guided
[3] https://huggingface.co/winner0613/tokatrons-bart-plan-guided-v3
[4] https://huggingface.co/winner0613/tokatrons-roberta-plan-classifier
[5] https://huggingface.co/winner0613/tokatrons-deberta-plan-classifier
[6] https://huggingface.co/winner0613/tokatrons-distilbert-plan-classifier
[7] https://huggingface.co/winner0613/tokatrons-roberta-balanced-classifier

Acknowledgments

We thank the CLEF SimpleText organisers for providing the dataset, evaluation framework, and computational support through the CodaBench platform. We are grateful to Google Colab for providing free GPU resources for model training. We acknowledge Groq for providing API access for LLaMA model inference.

Declaration on Generative AI

During the preparation of this work, the authors used *GitHub Copilot* in order to: **Formatting assistance**. Further, the authors used *Claude by Anthropic* in order to: **Grammar and spelling check** and **Drafting content**. After using these tools/services, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] L. Ermakova, H. Azarbonyad, J. Bakker, B. Chakar, G. K. Shahi, J. Kamps, Overview of the CLEF 2026 SimpleText track: Simplify scientific texts (and nothing more), in: M. Hagen, M. Potthast, B. Stein, P. Schaer, E. Zangerle, S. MacAvaney, J. M. Struß, E. Sánchez Salido, A. Barrón Cedeño, A. García Seco de Herrera (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Seventeenth International Conference of the CLEF Association (CLEF 2026), Lecture Notes in Computer Science, Springer, 2026.

[2] J. Bakker, L. Ermakova, J. Kamps, Overview of the CLEF 2026 SimpleText Task 1: Simplify Scientific Text, in: Working Notes of CLEF 2026, CEUR Workshop Proceedings, CEUR-WS.org, 2026.

[3] 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, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, 2020, pp. 7871–7880.

[4] AI@Meta, The Llama 3 Herd of Models, arXiv preprint arXiv:2407.21783 (2024).

[5] Meta AI, Llama 4: The Llama 4 Family of Models, 2025. URL: https://ai.meta.com/blog/llama-4-multimodal-intelligence/, accessed: 2026-06-06.

[6] L. Ermakova, S. Huet, E. SanJuan, J. Kamps, H. Bretel, O. Zubarev, P. Braslavski, I. Ovchinnikova, Overview of the SimpleText Lab at CLEF 2024: Simplify, Explain and Generate Scientific Text, in: Working Notes of CLEF 2024, volume 3740 of *CEUR Workshop Proceedings*, 2024.

[7] A. Michail, P. S. Andermatt, T. Fankhauser, SimpleText Best of Labs in CLEF-2023: Scientific Text Simplification Using Multi-Prompt Minimum Bayes Risk Decoding, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction. CLEF 2024, volume 14958 of *Lecture Notes in Computer Science*, Springer, 2024, pp. 227–253. doi:10.1007/978-3-031-71736-9_17.

[8] M. Maddela, Y. Dou, D. Heineman, W. Xu, LENS: A Learnable Evaluation Metric for Text Simplification, in: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Toronto, Canada, 2023, pp. 16383–16408.

[9] N. Largey, R. Maarefdoust, S. Durgin, B. Mansouri, Notebook for the SimpleText Lab at CLEF 2024: Application of Large Language Models for Scientific Text Simplification, in: Working Notes of CLEF 2024, volume 3740 of *CEUR Workshop Proceedings*, 2024. URL: https://ceur-ws.org/Vol-3740/paper-316.pdf.

[10] K. C. Marturi, H. H. Elwazzan, LLM-Guided Planning and Summary-Based Scientific Text Simplification: DS@GT at CLEF 2025 SimpleText, in: Working Notes of CLEF 2025, CEUR Workshop Proceedings, 2025. ArXiv:2508.11816.

[11] D. S. Nielsen, K. Enevoldsen, P. Schneider-Kamp, Encoder vs Decoder: Comparative Analysis of En-

coder and Decoder Language Models on Multilingual NLU Tasks, arXiv preprint arXiv:2406.13469 (2024).

[12] T. Kew, A. Chi, L. Vásquez-Rodríguez, S. Agrawal, D. Aumiller, F. Alva-Manchego, M. Shardlow, BLESS: Benchmarking Large Language Models on Sentence Simplification, in: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2023, pp. 13291–13309.

[13] W. Xu, C. Napoles, E. Pavlick, Q. Chen, C. Callison-Burch, Optimizing Statistical Machine Translation for Text Simplification, Transactions of the Association for Computational Linguistics 4 (2016) 401–415.

[14] K. Papineni, S. Roukos, T. Ward, W.-J. Zhu, BLEU: A Method for Automatic Evaluation of Machine Translation, in: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, 2002, pp. 311–318.

[15] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, Y. Artzi, BERTScore: Evaluating Text Generation with BERT, in: Proceedings of the 8th International Conference on Learning Representations (ICLR), 2020.

[16] J. P. Kincaid, R. P. F. Jr., R. L. Rogers, B. S. Chissom, Derivation of New Readability Formulas (Automated Readability Index, Fog Count, and Flesch Reading Ease Formula) for Navy Enlisted Personnel, Technical Report Research Branch Report 8-75, Naval Technical Training Command, Memphis, TN, 1975.

[17] L. Martin, A. Fan, É. de la Clergerie, A. Bordes, B. Sagot, MUSS: Multilingual Unsupervised Sentence Simplification by Mining Paraphrases, in: Proceedings of the Thirteenth Language Resources and Evaluation Conference (LREC 2022), European Language Resources Association, Marseille, France, 2022, pp. 1651–1664.

[18] S. Narayan, Y. Zhao, J. Maynez, G. Simões, V. Nikolaev, R. McDonald, Planning with learned entity prompts for abstractive summarization, Transactions of the Association for Computational Linguistics 9 (2021) 1475–1492. doi:10.1162/tacl_a_00438.

[19] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, RoBERTa: A Robustly Optimized BERT Pretraining Approach, arXiv preprint arXiv:1907.11692 (2019).