DUTH at CLEF 2026 SimpleText: Grounded Biomedical Text Simplification and Evidence-Guided Overgeneration Detection

Georgios Arampatzis[1,*], Avi Arampatzis[1]

[1]Democritus University of Thrace, Department of Electrical and Computer Engineering, Xanthi, Greece

Abstract

This paper presents the participation of Team DUTH in the CLEF 2026 SimpleText track, addressing two complementary biomedical scientific text processing tasks: text simplification and overgeneration detection. Task 1 focuses on sentence-level and document-level biomedical text simplification, where complex scientific content must be rewritten in simpler language while preserving factual meaning, numerical findings, comparison groups, outcomes, uncertainty expressions, and evidence certainty. For this task, we use grounded prompting with open-source instruction-tuned large language models, primarily Gemma-2-9B-IT. At sentence level, we develop an EasySplit prompting strategy that encourages simple wording, short sentence structure, and faithful preservation of biomedical information. At document level, we introduce a sentence-to-document hybrid strategy that aggregates high-quality sentence-level simplifications into document-level outputs and falls back to a direct document-level prediction when the aggregated output falls outside a safe length range. Our best official sentence-level run achieved 46.09 SARI, while the best grounded EasySplit variant achieved 46.04 SARI. For document-level simplification, our best official run achieved 46.16 SARI, while the strongest sentence-to-document hybrid variant achieved 46.13 SARI.

Task 2 focuses on detecting and classifying overgeneration in simplified biomedical texts, where the goal is to identify unsupported, distorted, repetitive, or prompt-leaked content. We formulate this task as source-aware sentence classification: each simplified sentence is paired with evidence retrieved from the corresponding source abstract, allowing the model to judge whether the generated content is grounded in the biomedical source. We evaluate lexical baselines, tree-based models, zero-shot large language model judging, and supervised domain-specific Transformer encoders. Our strongest binary overgeneration detection system combines SciBERT and BiomedBERT through probability-level ensembling and threshold calibration, achieving an official document-level F1 score of 0.7994. For fine-grained overgeneration classification, the best document-level F1 score was also 0.7994, while the highest-accuracy multiclass system achieved 0.7878 accuracy and 0.7901 document-level F1. Across both tasks, the results show that biomedical simplification and overgeneration detection benefit from grounded, evidence-aware, and domain-specific methods, combining constrained LLM prompting, cautious post-processing, sentence-to-document hybridization, explicit source evidence, biomedical Transformer pretraining, probability-level ensembling, and calibrated decision thresholds.

Keywords

Biomedical Text Simplification, Scientific Text Simplification, Plain-Language Generation, Grounded Prompting, Large Language Models, Gemma-2-9B-IT, Sentence-to-Document Hybridization, Overgeneration Detection, Hallucination Detection, Biomedical NLP, SciBERT, BiomedBERT, Transformer Ensembles, CLEF SimpleText

1. Introduction

Scientific and biomedical texts are often difficult for non-expert readers because they contain dense terminology, long syntactic constructions, numerical findings, domain-specific concepts, and implicit background knowledge. This creates an accessibility barrier for patients, citizens, journalists, educators, and policy makers who need to understand reliable scientific evidence. Automatic text simplification aims to reduce this barrier by rewriting complex texts into clearer and more accessible language while preserving the original meaning [1, 2].

Biomedical text simplification is more constrained than general-domain simplification. A system must

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

*Corresponding author.

geoaramp@ee.duth.gr (G. Arampatzis); avi@ee.duth.gr (A. Arampatzis)

0009-0003-3840-4537 (G. Arampatzis); 0000-0003-2415-4592 (A. Arampatzis)

not only simplify wording and syntax, but also preserve factual meaning, evidence strength, comparison groups, outcomes, numerical values, and uncertainty expressions. For example, changing expressions such as *may reduce*, *probably has little effect*, or *low-certainty evidence* can alter the interpretation of a biomedical finding. Therefore, simplification in this domain should be treated as grounded rewriting rather than open-ended summarization.

The CLEF SimpleText track provides a shared evaluation setting for improving access to scientific texts and for studying automatic simplification, scientific communication, and reliability of generated outputs [3, 4]. In the 2026 edition, the SimpleText track includes complementary tasks that address both generation and verification. Task 1 focuses on biomedical scientific text simplification at sentence and document level, while Task 2 focuses on detecting and classifying overgeneration in simplified biomedical texts. These two tasks are closely connected: simplification systems should produce accessible text, but their outputs must also remain faithful to the source.

Recent large language models have become strong candidates for text simplification because they can follow natural-language instructions and produce fluent rewrites without task-specific fine-tuning. However, their use in scientific and biomedical domains raises an important risk. Neural generation models may introduce unsupported information, omit important qualifiers, overstate uncertain conclusions, or add plausible but ungrounded explanations. This problem is related to hallucination and factual inconsistency in natural language generation [5, 6, 7]. In biomedical communication, such errors are particularly problematic because even small changes can distort scientific conclusions.

This paper presents the participation of Team DUTH in the CLEF 2026 SimpleText track. We address two complementary tasks. For Task 1, we focus on sentence-level and document-level biomedical text simplification. Our system uses grounded prompting with open-source instruction-tuned large language models, primarily Gemma-2-9B-IT [8]. The prompts explicitly require the model to preserve biomedical meaning, numerical findings, comparison groups, outcomes, uncertainty expressions, and evidence certainty. At document level, we introduce a sentence-to-document hybridization strategy that aggregates high-quality sentence-level simplifications into document-level outputs and falls back to a direct document-level prediction when the aggregated output is outside a safe length range.

For Task 2, we address biomedical overgeneration detection and classification. We formulate the task as source-aware sentence classification. Each simplified sentence is paired with evidence retrieved from the corresponding source abstract, allowing the classifier to decide whether the generated sentence is grounded in the original biomedical content. We compare lexical baselines, tree-based classifiers, zero-shot large language model judging, and supervised biomedical Transformer encoders. Our strongest systems use domain-specific pretrained models, including SciBERT and BiomedBERT, combined through probability-level ensembling and document-level threshold calibration [9, 10].

The two tasks are unified by the same methodological principle: grounded and evidence-aware processing. For simplification, the system should make the text easier to understand without changing the biomedical claim. For overgeneration detection, the system should compare each simplified sentence against source evidence and identify unsupported or distorted content. Across both tasks, we avoid relying only on open-ended generation and instead use constraints, evidence, calibration, and validation to improve reliability.

Our main contributions are as follows. First, we show that grounded prompting with open-source instruction-tuned LLMs can achieve strong biomedical simplification performance without task-specific fine-tuning. Second, we demonstrate that sentence-level simplification outputs can be reused effectively for document-level simplification through a simple sentence-to-document hybridization strategy. Third, we show that source-aware biomedical Transformer ensembles substantially outperform lexical, tree-based, and zero-shot approaches for overgeneration detection. Finally, we provide a unified analysis of biomedical simplification and overgeneration detection as two parts of the same reliability-oriented scientific communication problem.

The remainder of this paper is organized as follows. Section 2 discusses related work on text simplification, biomedical simplification, evaluation metrics, hallucination detection, and biomedical pretrained language models. Section 3 provides an overview of the DUTH participation in the CLEF 2026 SimpleText track. Section 4 describes our Task 1 system for biomedical text simplification. Section 5

presents our Task 2 system for evidence-guided overgeneration detection. Section 6 discusses the main findings across both tasks, and Section 7 concludes the paper.

2. Related Work

2.1. Text Simplification and Scientific Text Accessibility

Automatic text simplification aims to transform complex texts into simpler and more accessible versions while preserving the original meaning [1]. Early approaches relied on lexical substitution, syntactic rewriting, sentence splitting, and deletion, whereas more recent methods have framed simplification as monolingual text-to-text generation [2, 11]. Neural sequence-to-sequence models improved fluency and enabled more flexible rewriting, but they also introduced challenges related to meaning preservation and factual consistency.

Scientific and biomedical text simplification is particularly challenging because source texts often contain technical terminology, domain-specific abbreviations, numerical findings, study designs, comparison groups, outcomes, and uncertainty statements. Unlike general-domain simplification, biomedical simplification must preserve evidence strength and avoid altering medically relevant claims. For example, removing uncertainty markers or strengthening cautious conclusions can distort the interpretation of biomedical evidence. This makes biomedical simplification a grounded rewriting problem rather than a free-form summarization problem.

The CLEF SimpleText track has provided a shared benchmark for improving access to scientific texts and evaluating simplification systems in a reproducible setting [3, 4]. In the 2026 edition, Task 1 focuses on biomedical scientific text simplification using sentence-level and document-level inputs, while Task 2 focuses on detecting overgeneration and information distortion in simplified biomedical texts. These tasks are complementary because simplification quality depends not only on readability, but also on whether the generated output remains faithful to the source.

2.2. Controllable Simplification and Large Language Models

Controllable simplification methods aim to guide the output toward desired properties such as compression, lexical simplicity, syntactic simplicity, and meaning preservation. Zhang and Lapata [12] formulated sentence simplification as a reinforcement learning problem, optimizing for fluency, adequacy, and simplicity. Martin et al. [13] showed that simplification can be controlled by explicitly guiding generation toward target attributes such as length, lexical complexity, and syntactic structure.

Large language models have recently become strong candidates for simplification because they can follow natural-language instructions and produce fluent rewrites without task-specific fine-tuning. Instruction-tuned models such as FLAN-T5, BART-based systems, Gemma, Qwen, and Llama provide strong general-purpose generation capabilities [14, 15, 8, 16, 17]. However, open-ended generation is risky in biomedical domains, where fluent outputs may introduce unsupported additions, alter uncertainty, omit clinically relevant details, or overstate evidence. For this reason, biomedical simplification requires controlled generation settings that prioritize factual preservation and source faithfulness in addition to readability.

2.3. Evaluation of Text Simplification

Text simplification evaluation usually combines reference-based, semantic, and readability-oriented metrics. SARI is widely used because it evaluates whether a system appropriately adds, deletes, and keeps n-grams with respect to both the source and reference simplifications [18]. This makes it more directly aligned with simplification than standard machine translation metrics. BLEU is also commonly reported, although it primarily measures lexical overlap with reference texts and may reward outputs that preserve too much of the original input [19].

Semantic metrics such as BERTScore compare contextual embeddings of generated and reference texts and provide a complementary view of meaning preservation [20]. Learned simplification metrics such as LENS aim to better correlate with human judgments of simplification quality [21]. In biomedical simplification, however, automatic metrics remain incomplete because they may not fully detect factual distortions, missing uncertainty markers, or unsupported medical claims. Therefore, automatic simplification scores should be interpreted together with factuality and overgeneration analysis.

2.4. Overgeneration, Hallucination, and Factual Consistency

Overgeneration in simplified text is closely related to hallucination and factual inconsistency in natural language generation. Hallucination occurs when a model produces content that is unsupported by, or contradictory to, the input document [7]. This problem has been widely studied in abstractive summarization, where neural systems can generate fluent summaries that do not remain faithful to the source [5, 6].

In text simplification, overgeneration may appear as unsupported explanations, causal interpretations, added background information, repeated content, leaked instructions, or conversational comments. Such errors are especially serious in scientific and biomedical communication because they can change the meaning of evidence or introduce claims that were not present in the source. Traditional metrics such as BLEU, ROUGE, and SARI are not sufficient to identify these errors reliably, because they mainly compare the generated output with references or source overlap rather than checking whether each claim is grounded.

The CLEF 2026 SimpleText Task 2 addresses this issue by evaluating whether simplified biomedical outputs contain unsupported, distorted, repetitive, or prompt-leaked information [22]. Task 2.1 focuses on binary overgeneration detection, while Task 2.2 extends the setting to fine-grained classification of distortion types. This connects text simplification evaluation with factuality assessment and evidence-grounded classification.

2.5. Biomedical Pretrained Language Models and Factuality Classification

Transformer-based pretrained language models have become a dominant approach for sentence classification, semantic matching, and factuality-related tasks. BERT introduced bidirectional Transformer pretraining and provided strong representations for many downstream NLP tasks [23]. However, general-domain pretraining may be suboptimal for scientific and biomedical language, where terminology, discourse structure, abbreviations, and entity distributions differ from open-domain text.

Domain-specific models address this limitation by pretraining on scientific or biomedical corpora. SciBERT is pretrained on scientific publications and improves performance on scientific NLP tasks [9]. PubMedBERT, also referred to as BiomedBERT, is pretrained from scratch on biomedical text and provides strong representations for biomedical language understanding [10]. BioLinkBERT extends biomedical pretraining by incorporating document-link information, which can help capture knowledge across related biomedical documents [24]. These models are therefore relevant for biomedical overgeneration and factuality detection, where systems must distinguish source-supported statements from unsupported, distorted, or misleading simplified content.

2.6. Previous DUTH Shared-Task Participation

The present work also builds on previous DUTH participation in shared tasks and related NLP systems. In CLEF 2025 SimpleText, DUTH addressed scientific text simplification and hallucination detection, providing a direct precursor to the generation-and-verification setting studied in the present paper [25]. Previous DUTH systems have also explored multilingual classification, retrieval, reranking, sparse–neural fusion, retrieval-augmented generation, and shared-task system design.

Work on SemEval intimacy analysis, EXIST sexism detection, online polarization detection, stance prediction, and constrained multilingual generation used ensemble modelling, multilingual representations, Transformer-based classification, and prompt-based generation strategies [26, 27, 28, 29, 30].

These studies provide broader methodological background for classification, calibration, multilingual modelling, and controlled generation.

DUTH work on news credibility, justification retrieval, product search, passage retrieval, topic-oriented retrieval, JOKER, and humour-aware retrieval provides additional background for evidence retrieval, ranking, fusion, candidate selection, and retrieval-augmented reasoning [31, 32, 33, 34, 35, 36, 37]. These earlier systems are not direct solutions to biomedical simplification, but they inform the reliability-oriented design principles used in the present SimpleText participation.

3. Overview of the DUTH Participation

Team DUTH participated in two CLEF 2026 SimpleText tasks: Task 1 on biomedical scientific text simplification and Task 2 on overgeneration detection in simplified biomedical texts. Although the tasks have different outputs, they are methodologically connected. Task 1 requires the generation of simpler biomedical text, while Task 2 evaluates whether simplified outputs remain faithful to the source. We therefore treat both tasks as reliability-oriented biomedical NLP problems, where generated or classified outputs should remain grounded in the source evidence.

For Task 1, we formulate biomedical simplification as grounded rewriting. Our systems use open-source instruction-tuned large language models, primarily Gemma-2-9B-IT [8], with prompts designed to preserve factual meaning, numerical findings, comparison groups, outcomes, uncertainty expressions, and evidence certainty. At sentence level, we use grounded EasySplit-style prompting, while at document level we combine direct document-level simplification with sentence-to-document hybridization and length-based fallback control.

For Task 2, we formulate overgeneration detection as source-aware sentence classification. Each simplified sentence is paired with evidence retrieved from the corresponding source abstract, allowing the classifier to judge whether the generated content is supported by the biomedical source. We compare lexical baselines, tree-based classifiers, zero-shot LLM judging, and supervised domain-specific Transformer encoders. The strongest systems use SciBERT and BiomedBERT [9, 10], combined through probability-level ensembling and document-level threshold calibration.

Table 1: Overview of the DUTH participation in CLEF 2026 SimpleText.

Task	Setting	Main approach	Best result
Task 1.1	Sentence-level biomedical text simplification	Gemma-2-9B-IT prompting with grounded EasySplit-style instructions and light plain-language post-processing	46.09 SARI
Task 1.2	Document-level biomedical text simplification	Gemma-2-9B-IT document-level simplification with sentence-to-document hybridization, length-ratio fallback, and light post-processing	46.16 SARI
Task 2.1	Binary overgeneration detection	Source-aware SciBERT–BiomedBERT probability ensemble with document-level threshold calibration	0.7994 doc-F1
Task 2.2	Fine-grained overgeneration classification	Calibrated binary overgeneration mask combined with source-aware multiclass label assignment	0.7994 doc-F1; 0.7878 Acc.

Table 1 summarizes the two SimpleText tasks, the main DUTH approach, and the best official results. Overall, our participation shows that biomedical simplification and overgeneration detection benefit from the same core principles: grounding, evidence awareness, domain-specific modelling, and calibrated decision-making.

4. Task 1: Grounded LLM Prompting and Sentence-to-Document Hybridization for Biomedical Text Simplification

4.1. Dataset

The CLEF 2026 SimpleText Task 1 focuses on biomedical scientific text simplification at sentence and document level [38]. The task is divided into two complementary subtasks: Task 1.1, which targets

sentence-level simplification, and Task 1.2, which targets document-level simplification. Both subtasks are based on the Cochrane-auto biomedical corpus, which contains biomedical scientific texts derived from Cochrane systematic reviews and their corresponding plain-language summaries. The goal is to rewrite complex biomedical content into simpler and more accessible language while preserving factual meaning, evidence strength, uncertainty, comparison groups, outcomes, and numerical findings.

Scientific and biomedical text simplification differs from general-domain simplification because the input often contains technical terminology, study designs, statistical results, treatment comparisons, population descriptions, and uncertainty expressions. In this setting, simplification cannot be treated as unrestricted summarization. A system may improve readability, but it must not alter the meaning of medical claims. For example, changing an expression such as *may reduce* into *reduces* or removing phrases such as *low-certainty evidence* may distort the interpretation of a biomedical finding. This motivates grounded simplification methods that explicitly preserve the source meaning and evidence status [1, 5, 6].

For Task 1.1, the input consists of complex biomedical sentences. In the final test input used for submission, the sentence-level source file contained 48,809 records. Each record includes a complex source sentence, metadata such as the pair identifier, language, and source collection, and requires one simplified prediction. The sentence-level setting is suitable for controlled rewriting because each input unit contains limited local context and can be simplified with explicit constraints on meaning preservation and readability.

For Task 1.2, the input consists of full biomedical documents or abstracts. In the final test input, the document-level source file contained 8,069 records. Each record corresponds to a document-level simplification instance and includes the complex source text and associated metadata. Compared with sentence-level simplification, document-level simplification is more challenging because the system must preserve discourse structure, coherence across sentences, paragraph-level organization, and the balance between simplification and summarization. Document-level outputs should be readable, but they should also preserve enough content to remain faithful to the original biomedical source.

Table 2: Dataset statistics for CLEF 2026 SimpleText Task 1.

Subtask	**Input level**	**Test records**
Task 1.1	Sentence-level simplification	48,809
Task 1.2	Document-level simplification	8,069

The official test data do not include reference simplifications. Therefore, submissions are evaluated blindly through the official evaluation platform. The official evaluation uses automatic simplification and generation metrics, including SARI, BLEU, LENS, and BERTScore, together with human assessment of selected submissions [18, 19, 20, 21]. Our systems generate one prediction for every input record and package the outputs in the required JSON-in-ZIP submission format.

4.2. Models Used

We experimented with several open-source instruction-tuned language models for biomedical text simplification. The main model used in our final systems was Gemma-2-9B-IT, an open instruction-tuned large language model from the Gemma 2 family [8]. Gemma-2-9B-IT was selected as the primary model because it provided the best balance between generation quality, controllability, and computational feasibility on our available hardware. In our experiments, it followed simplification instructions reliably and produced fluent outputs without requiring task-specific fine-tuning.

In addition to Gemma-2-9B-IT, we performed exploratory experiments with other open-source instruction-tuned models, including Qwen2.5-14B-Instruct, Qwen2.5-7B-Instruct, Gemma-3-12B-IT, and Llama-3-8B-Instruct [16, 17]. These models were used to test whether larger models, alternative instruction-tuning recipes, or more guarded prompting strategies could improve biomedical simplification quality. However, the official results showed that the strongest submitted systems were based

on Gemma-2-9B-IT, while Qwen-based and guarded Gemma-3 variants did not outperform the best Gemma-2-9B-IT configurations.

For Task 1.1, Gemma-2-9B-IT was used with sentence-level grounded prompting and an EasySplit-style prompt. This prompt encouraged simple wording, faithful preservation of biomedical meaning, and controlled sentence splitting. We also applied light plain-language post-processing to selected sentence-level outputs. The best official sentence-level run was the Gemma-2-9B-IT plain post-processing variant, which achieved 46.09 SARI, while the strongest grounded EasySplit variant remained very close, achieving 46.04 SARI.

For Task 1.2, Gemma-2-9B-IT was used in two ways. First, we used direct document-level prompting, where the full biomedical document was simplified in one generation step. Second, we reused sentence-level simplification outputs and aggregated them into document-level predictions using a sentence-to-document hybrid strategy with length-ratio fallback control. This hybrid strategy aimed to combine the local controllability of sentence-level simplification with document-level safeguards against outputs that were too short, too long, or insufficiently aligned with the source.

The final best-performing Task 1 systems were therefore based primarily on Gemma-2-9B-IT. The best official document-level run was the Gemma-2-9B-IT plain post-processing variant, which achieved 46.16 SARI. The strongest sentence-to-document hybrid configuration achieved 46.13 SARI, showing that hybridization was highly competitive even though the best final score was obtained after light post-processing. Overall, the results indicate that model choice, grounded prompt design, moderate post-processing, and sentence-to-document fallback control were all important for strong biomedical simplification performance.

4.3. Methodology

Our approach formulates biomedical simplification as grounded rewriting. Instead of allowing the model to freely summarize, reinterpret, or expand the source text, we constrain the generation process through explicit instructions. The prompts require the model to preserve biomedical meaning, numerical findings, comparison groups, outcomes, uncertainty expressions, and evidence certainty. This design is motivated by previous findings that neural generation systems can produce fluent but factually inconsistent outputs, especially in abstractive rewriting and summarization settings [5, 6, 7].

4.3.1. Sentence-Level Grounded Prompting

For Task 1.1, we used sentence-level grounded prompting. Each source sentence was rewritten in plain language for non-expert readers, while preserving the biomedical information required for factual interpretation. In particular, the prompt instructed the model to keep numbers, treatment comparisons, population groups, outcomes, and uncertainty expressions such as *may*, *probably*, *likely*, *uncertain*, and *little to no difference*. The prompt also encouraged the model to replace technical expressions with simpler wording when this could be done without changing the meaning.

The strongest grounded sentence-level variant was `Gemma2_9B_SentenceEasySplit`. This prompt emphasized short and clear sentence structures and allowed sentence splitting when it improved readability. At the same time, it explicitly prohibited unsupported additions and instructed the model to return only the simplified sentence or sentences, without markdown, headings, bullet points, citations, or explanations. This restriction was important because the official evaluation expects direct simplification outputs rather than explanatory text.

The EasySplit prompt was designed to balance readability and faithfulness. If the model simplified too conservatively, the output remained close to the original biomedical sentence and received weaker simplification scores. If the model simplified too aggressively, it risked deleting important evidence, altering numerical information, or changing uncertainty. The final prompt therefore encouraged simpler wording and shorter syntax, while explicitly grounding the output in the source sentence.

4.3.2. Direct Document-Level Prompting

For Task 1.2, we first explored direct document-level prompting. In this setting, the model receives a full biomedical document or abstract and produces a simplified document-level output in a single generation step. We tested several prompt variants, including detailed, less-compressed, balanced, and sentence-splitting-oriented prompts.

The direct document-level experiments showed that overly aggressive simplification can reduce performance. Prompts that encouraged excessive compression or too much sentence splitting sometimes produced outputs that were shorter and easier to read, but less faithful to the original biomedical source or less aligned with reference-style simplifications. The strongest direct document-level configuration used a balanced prompt that preserved biomedical findings while applying moderate compression and clearer sentence structure.

This result shows that document-level simplification is not simply a longer version of sentence-level simplification. At document level, the system must preserve coherence, maintain the relationship between sentences, avoid deleting important findings, and keep enough information to remain faithful to the source. This motivated the sentence-to-document hybridization strategy described below.

4.3.3. Sentence-to-Document Hybridization

To improve document-level performance, we introduced a sentence-to-document hybridization strategy. The main idea is to reuse strong Task 1.1 sentence-level simplification outputs and aggregate them into document-level predictions. For each document, sentence-level predictions are grouped by document identifier and ordered according to paragraph and sentence metadata. The simplified sentences are then concatenated into paragraph-level and document-level outputs.

This strategy combines two complementary strengths. Sentence-level simplification is easier to control because each input is short and local, which reduces the risk of long unsupported additions. Direct document-level prompting, however, can better preserve broader coherence and avoid fragmentary outputs. The hybrid method therefore uses sentence-level clarity when aggregation is safe, while retaining a document-level fallback when the aggregated output is unsuitable.

Direct aggregation can sometimes produce outputs that are too short, too long, or too close to the source. Therefore, we used a length-ratio filtering mechanism. Let r be the ratio between the word count of the aggregated sentence-level output and the word count of the original document:

$$r = \frac{\text{words}(\text{aggregated})}{\text{words}(\text{source})}. \tag{1}$$

If r falls within a predefined safe range, the aggregated sentence-level output is used as the document prediction. Otherwise, the system falls back to a selected direct document-level prediction. We tested both stricter and wider safe ranges. The best hybrid configuration used a wider safe range, allowing the system to benefit more often from strong sentence-level outputs while still avoiding extreme length mismatches.

The strongest sentence-to-document hybrid configuration was `HybridWideMaxLexLite`. This run extended the hybrid output with light post-processing. The post-processing removed duplicate text and applied limited plain-language lexical cleanup, while avoiding aggressive removal of biomedical or statistical information. This design produced a highly competitive document-level result and showed that sentence-level simplification can be effectively reused for document-level biomedical simplification when combined with fallback control and conservative post-processing.

4.4. Implementation and Environment

All Task 1 experiments were implemented in Python 3.10. Neural generation was performed using PyTorch and the Hugging Face Transformers library [39, 40]. Experiments were executed on a compute node equipped with an NVIDIA RTX A6000 GPU.

The local environment used a Python virtual environment under the project directory. The Hugging Face model cache was reused across experiments in order to avoid repeated downloads of large model checkpoints. During final experimentation, models were loaded from local cache when possible, improving reproducibility and reducing dependency on external model updates.

The input data were stored in compressed archives containing the official JSON files for the sentence-level and document-level subtasks. The main generation scripts loaded the corresponding source file according to the selected subtask. For Task 1.1, the script loaded the sentence-level records. For Task 1.2, the script loaded the document-level records. For each input record, the system generated one prediction, attached the corresponding run identifier, and wrote the output to a JSON file.

Long-running experiments were executed with background processes so that generation could continue even if the terminal session disconnected. Intermediate predictions were written to checkpoint files, enabling progress monitoring and reducing the risk of losing outputs during long generation runs. GPU usage was monitored during generation.

Before submission, all JSON and ZIP files were validated. The validation checks confirmed that the number of predictions matched the number of input records, that no predictions were missing, that run identifiers were present, and that each ZIP archive contained the expected JSON file. These checks were important because format errors can lead to failed or invalid submissions independently of model quality.

4.5. Results

4.5.1. Evaluation Metrics

The official CLEF 2026 SimpleText Task 1 evaluation uses automatic metrics for text simplification, including SARI, BLEU, LENS, and BERTScore, together with human assessment of selected submissions [18, 19, 20, 21]. In the official leaderboard, SARI is the primary ranking metric, while additional diagnostic metrics such as BLEU, FKGL, compression, sentence splitting, Levenshtein similarity, exact-copy rate, addition, deletion, average length, and LENS provide further insight into system behaviour.

SARI is the main metric used to evaluate simplification quality. It measures how well a system performs the three core simplification operations: adding appropriate information, deleting unnecessary or overly complex content, and keeping useful content with respect to the source and reference simplifications [18]. This makes SARI particularly suitable for simplification, because good outputs should not only overlap with the reference, but should also perform appropriate rewriting operations.

BLEU measures n-gram overlap between the system output and reference texts [19]. Although BLEU was originally designed for machine translation, it remains useful as a secondary indicator of lexical overlap with reference simplifications. However, high BLEU does not necessarily imply good simplification, because a system may preserve too much of the original input or avoid the rewriting operations rewarded by SARI.

FKGL, the Flesch–Kincaid Grade Level, estimates readability from sentence length and word complexity. Lower FKGL values generally indicate easier text, although biomedical simplification sometimes requires retaining technical terms for factual correctness. Compression measures the length ratio between the simplified output and the original input, while the sentence-splitting metric captures the degree to which the system breaks complex sentences into shorter sentences. Levenshtein similarity provides a surface-level indication of how much the generated output differs from the source text, and the exact-copy rate shows how often the system output is identical to the input.

The Add and Delete scores reported in the official results provide a more detailed view of the rewriting behaviour captured by SARI. The Add score reflects whether the system introduces appropriate simplification-oriented phrasing, while the Delete score reflects whether it removes unnecessary or overly complex content without losing important meaning. In our Task 1.1 sentence-level results, we also report the average output length, which helps characterize how compact the generated simplifications are. In our Task 1.2 document-level results, we additionally report LENS, a learned simplification metric designed to better correlate with human judgments of simplification quality [21]. Together, these

metrics help characterize whether a system behaves as a faithful simplifier, a conservative paraphraser, or an overly aggressive summarizer.

4.5.2. Sentence-Level Results

Table 3 presents representative official DUTH results for Task 1.1, the sentence-level biomedical simplification task. Our best sentence-level configuration was the Gemma-2-9B-IT variant with plain post-processing, which achieved the highest SARI score among our submitted runs, with a score of **46.09**. This run obtained a BLEU score of 8.85 and an FKGL value of 11.75, while maintaining a compression ratio of 0.87, a split ratio of 1.02, and a Levenshtein similarity score of 0.51. The exact-copy rate was 0.00, indicating that the system did not simply copy the original input. The Add and Delete operation scores were 0.43 and 0.56, respectively, and the average output length was 8.28 tokens, showing that the system produced compact simplifications while preserving the main biomedical content.

Table 3: Representative official DUTH results for Task 1.1 sentence-level simplification.

Rank	LLM / Strategy	SARI	BLEU	FKGL	Compr.	Split	Lev.	Exact	Add.	Del.	AvgLen
1	Gemma-2-9B-IT + plain post-processing	**46.09**	8.85	11.75	0.87	1.02	0.51	0.00	0.43	0.56	8.28
2	Gemma-2-9B-IT sentence-level EasySplit prompting	46.04	8.84	11.87	0.87	1.02	0.51	0.00	0.43	0.56	8.30
3	Gemma-2-9B-IT grounded sentence-level prompting	46.04	8.84	11.87	0.87	1.02	0.51	0.00	0.43	0.56	8.30
4	Gemma-2-9B-IT + plain selector over top candidates	46.02	8.88	11.83	0.87	1.02	0.51	0.00	0.42	0.56	8.30
5	DeBERTa-based candidate selection	45.98	8.82	11.57	0.89	1.07	0.52	0.00	0.43	0.55	8.31
6	Gemma-3-12B-IT guarded rewriting	45.96	8.93	11.85	0.87	1.02	0.52	0.00	0.42	0.56	8.31
7	Gemma-2-9B-IT balanced sentence-level prompting	45.77	9.19	10.93	0.77	0.98	0.50	0.00	0.37	0.60	8.29
8	Gemma-2-9B-IT reference-style batch rewriting	45.37	7.86	9.86	1.02	1.45	0.52	0.00	0.50	0.49	8.36
9	Gemma-2-9B-IT source-rescue plainification	45.28	9.31	11.63	0.85	1.02	0.53	0.00	0.39	0.55	8.34
10	Gemma-3-12B-IT guarded rewriting, alternative prompt	41.31	10.11	10.86	0.70	0.99	0.63	0.00	0.16	0.50	8.66
11	Lexical simplification baseline	28.80	16.52	12.29	0.80	0.99	0.88	0.01	0.01	0.26	8.82
12	Grounded lexical simplification baseline	12.34	12.50	12.51	0.99	1.00	0.98	0.06	0.02	0.02	8.84

The next two configurations, the EasySplit and grounded sentence-level prompting variants, achieved almost identical performance, both reaching a SARI score of 46.04, with BLEU 8.84, FKGL 11.87, compression 0.87, split ratio 1.02, Levenshtein similarity 0.51, Add 0.43, Delete 0.56, and average output length 8.30. These results show that grounded prompting and controlled sentence-level rewriting were highly competitive with the best post-processed variant. The plain selector over top candidates also remained close to the best result, reaching SARI 46.02, BLEU 8.88, and FKGL 11.83. This indicates that candidate selection helped maintain stable simplification quality, although it did not improve over the best plain post-processing strategy.

The DeBERTa-based candidate selection run achieved SARI 45.98, BLEU 8.82, and FKGL 11.57. Compared with the highest-ranked configuration, it produced slightly stronger compression and rewriting behavior, with a compression score of 0.89 and a split ratio of 1.07, while keeping the Levenshtein score close at 0.52. Similarly, the Gemma-3-12B-IT guarded rewriting configuration obtained SARI 45.96, BLEU 8.93, FKGL 11.85, compression 0.87, split ratio 1.02, Levenshtein 0.52, Add 0.42, Delete 0.56, and AvgLen 8.31. These results suggest that larger or guarded models did not necessarily outperform the more stable Gemma-2-9B-IT configurations for this task.

The balanced sentence-level prompting run achieved a lower SARI score of 45.77 but a higher BLEU score of 9.19 and a lower FKGL of 10.93. Its compression score was 0.77, with split ratio 0.98, Levenshtein 0.50, Add 0.37, Delete 0.60, and AvgLen 8.29. This suggests that the balanced prompt produced outputs with stronger lexical overlap and somewhat simpler readability, but its edit operations were less aligned with the reference simplifications according to SARI. In contrast, the reference-style batch rewriting run achieved SARI 45.37, BLEU 7.86, and FKGL 9.86, with a much higher split ratio of 1.45 and Add score of 0.50. This indicates more aggressive rewriting and sentence restructuring, which improved readability but reduced alignment with the official simplification references.

The lower-ranked runs further illustrate the importance of controlled generation. The source-rescue plainification variant achieved SARI 45.28 and the highest BLEU among the strong Gemma-2 runs, with BLEU 9.31, but it did not improve SARI. The second guarded Gemma-3-12B-IT run dropped to

SARI 41.31 despite having BLEU 10.11 and Levenshtein 0.63, suggesting that high lexical overlap alone was not sufficient for successful simplification. Finally, the lexical simplification baselines performed substantially worse, with SARI scores of 28.80 and 12.34. Although these baselines produced relatively high BLEU or high Levenshtein scores, their very low Add and Delete scores show that they failed to perform the balanced rewriting operations required by the task.

Overall, the Task 1.1 results show that the strongest DUTH submissions were based on Gemma-2-9B-IT with light post-processing, grounded prompting, and controlled sentence-level simplification. The best system achieved **46.09** SARI, while the strongest systems remained within a narrow SARI range between 46.09 and 45.96 and maintained stable BLEU, FKGL, compression, splitting, Levenshtein, Add, Delete, and average-length behavior. The results also show that SARI was more informative than BLEU alone for this task: some runs with higher BLEU scores did not achieve higher simplification quality, whereas the best SARI configurations balanced meaning preservation, rewriting, deletion, and output length more effectively.

4.5.3. Document-Level Results

Table 4 presents representative official DUTH results for Task 1.2, the document-level biomedical simplification task. To avoid reporting many near-identical submissions with the same score, we report one representative configuration from each group of similar runs and include both high-performing hybrid/post-processed variants and lower-scoring direct prompting baselines. Our best document-level configuration was the Gemma-2-9B-IT variant with plain post-processing, which achieved the highest SARI score among our Task 1.2 submissions, with a score of **46.16**.

The best run achieved SARI 46.16, BLEU 8.93, FKGL 11.75, compression 0.86, split ratio 1.01, Levenshtein similarity 0.51, exact-copy rate 0.00, Add 0.43, Delete 0.57, and LENS 8.28. These values indicate that the system produced compact document-level simplifications while avoiding exact copying and maintaining a balanced amount of insertion and deletion operations. The very low exact-copy rate shows that the outputs were not simple repetitions of the source documents, while the Add and Delete scores suggest that the model performed meaningful rewriting rather than only shortening the text.

Table 4: Representative official DUTH results for Task 1.2 document-level simplification.

Rank	LLM / Strategy	SARI	BLEU	FKGL	Compr.	Split	Lev.	Exact	Add.	Del.	LENS
1	Gemma-2-9B-IT + plain post-processing	**46.16**	8.93	11.75	0.86	1.01	0.51	0.00	0.43	0.57	8.28
2	Gemma-2-9B-IT + statistics-lite post-processing	46.15	8.93	11.75	0.86	1.01	0.51	0.00	0.43	0.57	8.28
3	Gemma-2-9B-IT hybrid sentence-to-document	46.13	8.91	11.77	0.86	1.01	0.51	0.00	0.43	0.57	8.28
4	Candidate-bank selection over generated variants	46.12	8.91	11.75	0.86	1.01	0.51	0.00	0.43	0.57	8.28
5	Gemma-2-9B-IT + split-read filtering	46.11	8.90	11.73	0.86	1.02	0.51	0.00	0.43	0.57	8.28
6	Majority-based lightweight selector	46.10	8.90	11.72	0.86	1.02	0.51	0.00	0.43	0.57	8.28
7	Gemma-2-9B-IT + split-read filtering	46.09	8.89	11.42	0.86	1.05	0.51	0.00	0.43	0.57	8.28
8	Gemma-2-9B-IT hybrid sentence-to-document	46.06	9.36	11.63	0.77	0.94	0.49	0.00	0.37	0.61	8.32
9	Gemma-3-12B-IT guarded rewriting	45.96	8.93	11.85	0.87	1.02	0.52	0.00	0.42	0.56	8.31
10	Gemma-2-9B-IT hybrid sentence-to-document	45.87	9.18	11.47	0.69	0.88	0.47	0.00	0.33	0.65	8.33
11	Gemma-2-9B-IT guarded hybrid output	45.72	8.86	11.76	0.87	1.02	0.51	0.01	0.43	0.56	8.29
12	Gemma-2-9B-IT sentence-to-document aggregation	45.39	7.98	11.15	0.58	0.78	0.44	0.00	0.27	0.69	8.35
13	Gemma-2-9B-IT direct grounded document prompt	42.56	2.85	10.61	0.32	0.45	0.34	0.00	0.12	0.81	8.43
14	Gemma-2-9B-IT direct balanced-short document prompt	42.06	2.63	9.82	0.31	0.47	0.34	0.00	0.13	0.82	8.41
15	Qwen2.5-7B-Instruct direct more-split prompt	41.62	2.90	8.42	0.33	0.57	0.36	0.00	0.13	0.80	8.42

The second-ranked configuration, Gemma-2-9B-IT with statistics-lite post-processing, obtained almost identical results, with SARI 46.15, BLEU 8.93, FKGL 11.75, compression 0.86, split ratio 1.01, Levenshtein 0.51, Add 0.43, Delete 0.57, and LENS 8.28. This very small difference shows that light post-processing variants were highly stable. The sentence-to-document hybrid configuration followed closely, achieving SARI 46.13, BLEU 8.91, FKGL 11.77, compression 0.86, split ratio 1.01, Levenshtein 0.51, exact-copy rate 0.00, Add 0.43, Delete 0.57, and LENS 8.28. Therefore, the hybrid sentence-to-document strategy remained one of the strongest approaches, even though the best final score was obtained after additional post-processing.

The next high-performing configurations also remained within a very narrow SARI range. Candidate-bank selection over generated variants reached SARI 46.12, while split-read filtering and the majority-based lightweight selector achieved SARI scores of 46.11 and 46.10, respectively. Their BLEU scores remained close, between 8.90 and 8.91, and their FKGL values ranged from 11.72 to 11.75. Compression remained stable at 0.86, with split ratios between 1.01 and 1.02 and Levenshtein similarity 0.51. This indicates that most of the strongest systems produced very similar simplification behavior, with only small differences in final ranking.

The lower-ranked hybrid variants show the effect of more aggressive document-level filtering and aggregation. The Gemma-2-9B-IT hybrid sentence-to-document variant ranked eighth achieved SARI 46.06 and the highest BLEU score in the table, 9.36, but its compression dropped to 0.77, its split ratio to 0.94, and its Add score to 0.37, while Delete increased to 0.61. This suggests that this variant preserved more lexical overlap with the references but performed fewer addition operations and more deletion operations. Another hybrid variant achieved SARI 45.87 with BLEU 9.18, FKGL 11.47, compression 0.69, split ratio 0.88, Levenshtein 0.47, Add 0.33, Delete 0.65, and LENS 8.33. Although it had relatively high BLEU, its lower SARI shows that stronger compression and deletion did not necessarily lead to better simplification quality under the official evaluation.

The guarded hybrid and aggregation variants further demonstrate this trade-off. The guarded hybrid output obtained SARI 45.72, BLEU 8.86, FKGL 11.76, compression 0.87, split ratio 1.02, Levenshtein 0.51, exact-copy rate 0.01, Add 0.43, Delete 0.56, and LENS 8.29. The sentence-to-document aggregation variant achieved SARI 45.39, BLEU 7.98, FKGL 11.15, compression 0.58, split ratio 0.78, Levenshtein 0.44, Add 0.27, Delete 0.69, and LENS 8.35. This indicates that aggregation alone was not sufficient when it produced overly compressed outputs and reduced the amount of useful rewriting.

The direct document-level prompting baselines were clearly weaker than the best post-processed and hybrid configurations. The Gemma-2-9B-IT direct grounded document prompt achieved SARI 42.56, BLEU 2.85, FKGL 10.61, compression 0.32, split ratio 0.45, Levenshtein 0.34, Add 0.12, Delete 0.81, and LENS 8.43. The direct balanced-short document prompt achieved SARI 42.06, BLEU 2.63, FKGL 9.82, compression 0.31, split ratio 0.47, Levenshtein 0.34, Add 0.13, Delete 0.82, and LENS 8.41. Similarly, the Qwen2.5-7B-Instruct direct more-split prompt reached SARI 41.62, BLEU 2.90, FKGL 8.42, compression 0.33, split ratio 0.57, Levenshtein 0.36, Add 0.13, Delete 0.80, and LENS 8.42. These direct prompting baselines produced lower FKGL values and slightly higher LENS scores, but their much lower SARI and BLEU scores show that they were too aggressive, deleting too much content and failing to preserve enough reference-aligned information.

Overall, the Task 1.2 results show that document-level biomedical simplification benefits from controlled post-processing and sentence-to-document hybridization. The best DUTH configuration achieved SARI 46.16, while the strongest hybrid sentence-to-document configuration achieved 46.13, only 0.03 SARI points lower. In contrast, the best direct document-level prompt achieved 42.56 SARI, meaning that the strongest post-processed configuration improved over direct grounded prompting by 3.60 SARI points. The results also show that higher BLEU or lower FKGL alone did not guarantee better simplification quality. The best-performing configurations achieved a better balance across SARI, BLEU, FKGL, compression, splitting, Levenshtein similarity, exact-copy rate, Add, Delete, and LENS, while the direct prompting baselines tended to over-compress the documents and delete too much information.

4.5.4. Additional Plain and Grounded Simplification Runs

We also evaluated representative simplification outputs in two variants: grounded and plain. The grounded variant preserves more of the original biomedical and statistical information, while the plain variant applies additional lexical simplification and limited removal of highly technical statistical expressions. Table 5 reports these submitted runs and their official scores.

Table 5: Summary comparison of representative plain and grounded DUTH simplification variants using the official Task 1.1 and Task 1.2 metrics.

Run	Variant	SARI	BLEU	FKGL	Compr.	Split	Lev.	Exact	Add.	Del.	AvgLen
Arampageos_Task12_PlainBestV1	Document plain	46.16	8.93	11.75	0.86	1.01	0.51	0.00	0.43	0.57	8.28
Arampageos_Task12_GroundedBestV1	Document grounded	46.13	8.91	11.77	0.86	1.01	0.51	0.00	0.43	0.57	8.28
Arampageos_Task11_PlainBestV1	Sentence plain	46.09	8.85	11.75	0.87	1.02	0.51	0.00	0.43	0.56	8.28
Arampageos_Task11_GroundedBestV1	Sentence grounded	46.04	8.84	11.87	0.87	1.02	0.51	0.00	0.43	0.56	8.30

The comparison in Table 5 shows that the plain and grounded variants achieved very similar official scores. At document level, the best plain run, Arampageos_Task12_PlainBestV1, achieved the highest SARI score, 46.16, while the corresponding grounded document-level run achieved 46.13. The difference is small, but it suggests that light plain-language rewriting and limited post-processing can slightly improve the automatic simplification score without substantially changing the other metrics.

The document-level plain and grounded variants had almost identical behaviour across the remaining metrics. Both obtained compression 0.86, split ratio 1.01, Levenshtein similarity 0.51, exact match 0.00, addition score 0.43, deletion score 0.57, and average length 8.28. The plain variant had a slightly higher BLEU score, 8.93 compared with 8.91, while the grounded variant had a slightly higher FKGL value, 11.77 compared with 11.75. These differences indicate that the two document-level strategies produced outputs with very similar structure and length, while the plain version achieved a marginally better balance under SARI.

A similar pattern appears at sentence level. The plain sentence-level run achieved 46.09 SARI, while the grounded sentence-level run achieved 46.04. The BLEU scores were also nearly identical, 8.85 and 8.84, respectively. Both runs had the same compression score 0.87, split ratio 1.02, Levenshtein similarity 0.51, exact match 0.00, addition score 0.43, and deletion score 0.56. The grounded sentence-level run had a slightly higher FKGL value, 11.87 compared with 11.75, and a slightly higher average length, 8.30 compared with 8.28, suggesting that it preserved somewhat more complex wording or structure.

Overall, the results suggest that grounded prompting is a strong and safe strategy for biomedical simplification, but the official automatic metrics slightly favoured the plain variants in both sentence-level and document-level settings. The gains are modest, so the main conclusion is not that plain rewriting is clearly superior, but that moderate lexical simplification can improve SARI while preserving similar compression, splitting, addition, and deletion behaviour. For biomedical simplification, this supports a cautious plain-language strategy: simplify terminology and sentence form where possible, but avoid aggressive deletion of numerical findings, uncertainty expressions, comparison groups, or evidence-related information.

5. Task 2: Evidence-Guided Biomedical Overgeneration Detection with Domain-Specific Transformer Ensembles

5.1. Task Description and Dataset

The CLEF 2026 SimpleText Task 2 focuses on detecting and classifying overgeneration in simplified biomedical texts [22]. Overgeneration occurs when a generated simplification introduces information that is unsupported by the original source document, distorts biomedical findings, modifies uncertainty statements, repeats content, leaks prompt instructions, or hallucinates facts that are not present in the evidence. This problem is closely related to hallucination and factual inconsistency in natural language generation, where fluent generated text may still be unsupported by the input document [5, 6, 7]. In biomedical communication, such errors are particularly problematic because they may alter the interpretation of scientific results and clinical evidence.

The task is divided into two complementary subtasks. Task 2.1 is a binary classification task that determines whether a simplified sentence contains overgenerated content. Task 2.2 extends this setting to fine-grained multiclass classification, where the system must identify the specific type of distortion. The official label space includes faithful simplifications, generation failures, grounded overgeneration,

leaked instructions, repetitive content, and ungrounded injections. This setting is connected to the broader goals of the CLEF SimpleText track, which studies scientific text simplification, accessibility, and reliability of simplified scientific communication [3, 4].

To formulate the problem as supervised classification, we convert the official data into sentence-level source-aware instances. Each simplified sentence is paired with evidence retrieved from the corresponding source abstract. This allows the classifier to compare generated content directly against the biomedical evidence rather than relying only on surface-level linguistic properties. This source-aware formulation is motivated by factuality-oriented approaches in generation evaluation, where generated claims are assessed against the source document [6, 5].

After preprocessing and flattening the training data, the binary Task 2.1 setup contains 350,583 sentence-level instances, including 26,881 positive overgeneration examples and 323,702 negative examples. The development set contains 18,638 sentence-level instances, while the hidden test set contains 105,001 sentence-level instances.

Table 6: Dataset statistics for CLEF 2026 SimpleText Task 2.1.

Split	Instances	Positive	Negative
Training	350,583	26,881	323,702
Development	18,638	–	–
Test	105,001	Hidden	Hidden

The dataset is highly imbalanced, with substantially more faithful sentences than overgenerated ones. Consequently, simple accuracy is not an appropriate evaluation measure. Instead, the official evaluation emphasizes document-level F1, rewarding systems that can reliably identify rare but important overgeneration cases. This makes threshold calibration and probability estimation critical components of a successful system.

5.2. Models Used

We evaluated both classical machine-learning baselines and neural source-aware classifiers for overgeneration detection. The classical baselines included TF–IDF logistic regression, ExtraTrees, and LightGBM classifiers. These systems used lexical overlap features, sentence statistics, numerical overlap features, and source–sentence similarity signals. TF–IDF provides a strong and interpretable lexical baseline for text classification, while LightGBM is a strong gradient-boosted decision-tree method for feature-based classification tasks [41, 42].

For neural classification, we fine-tuned several pretrained Transformer encoders, including SciBERT, BiomedBERT, BioLinkBERT, BlueBERT, and general-domain BERT models. BERT introduced bidirectional Transformer pretraining and became a standard architecture for sentence classification and semantic matching tasks [23]. However, general-domain pretraining may be suboptimal for biomedical overgeneration detection, because scientific and biomedical texts contain specialized terminology, numerical expressions, evidence statements, and domain-specific discourse patterns.

Domain-specific pretrained encoders address this limitation. SciBERT is pretrained on scientific publications and is designed for scientific NLP tasks [9]. BiomedBERT, also known as PubMedBERT, is pretrained from scratch on biomedical text and provides strong representations for biomedical language understanding [10]. BioLinkBERT extends biomedical pretraining by incorporating document-link information, which can help represent knowledge across related scientific documents [24]. These models are well suited to our setting because the classifier must compare simplified biomedical sentences with source evidence and determine whether the generated content is supported.

In addition to encoder-based classifiers, we explored LLM-based alternatives, including zero-shot instruction-tuned LLM judging and LLaMA-3.1 QLoRA source-aware classification variants. The zero-shot judge was used to test whether an instruction-following model could directly identify unsupported

content from the simplified sentence and source evidence. The QLoRA variants were used to examine whether lightly adapted generative models could behave as source-aware classifiers. In the official results, however, the supervised biomedical encoder systems were more stable and generally stronger than the zero-shot judging setup, supporting previous findings that hallucination and factual consistency evaluation require explicit grounding in the source document rather than relying only on generation fluency [5, 7].

For Task 2.1, the strongest systems were based on source-aware Transformer encoders and probability-level ensembling. The best submitted binary run used a BERT-average selection strategy with additional post-processing, while the best pure SciBERT–BiomedBERT average ensemble achieved nearly identical performance. These systems combine the positive-class probabilities produced by individual source-aware classifiers and then apply document-level threshold calibration on the development set.

For Task 2.2, we trained multiclass source-aware classifiers using the same sentence–evidence input format. We also evaluated gated variants that combine a calibrated binary overgeneration mask with multiclass label assignment. In this setting, the binary detector first decides whether a sentence should be treated as overgenerated, and the multiclass classifier then assigns the fine-grained distortion category. This design reduces over-prediction of rare distortion labels and reuses the stronger binary overgeneration signal.

5.3. Methodology

Our methodology is based on source-aware sentence classification. Instead of judging each simplified sentence in isolation, the system compares it with evidence retrieved from the corresponding source abstract. This formulation treats overgeneration detection as a grounding problem: a simplified sentence should be classified as faithful only if its content is supported by the biomedical source. This is consistent with prior work on factual consistency and hallucination detection, where generated statements are evaluated against the input document rather than only against surface-level fluency [5, 6, 7].

For each simplified sentence, we first retrieve the most relevant evidence sentences from the source abstract. Relevance is estimated using lexical overlap between the simplified sentence and each sentence of the source abstract. We also consider numerical overlap, because biomedical abstracts often contain important numerical findings, percentages, sample sizes, confidence intervals, and statistical values. A simplified sentence that preserves such values is more likely to remain grounded in the original evidence. The top retrieved source sentences are concatenated and used as evidence.

For Transformer-based models, each input instance is encoded as a sentence pair. The first segment contains the simplified sentence, and the second segment contains the retrieved source evidence. Conceptually, the input has the following form:

```
simplified sentence [SEP] retrieved source evidence
```

The classifier then predicts whether the simplified sentence is faithful or contains overgeneration. In the multiclass setting, the same input representation is used, but the classifier predicts one of the official distortion categories. This source-aware design allows the model to use both the generated sentence and the supporting biomedical context.

For Task 2.1, the models output a probability for the positive overgeneration class. Because the official evaluation is performed at document level, we do not optimize the threshold using sentence-level accuracy. Instead, we tune the decision threshold on the development set using document-level F1. A document is considered positive if at least one of its simplified sentences is predicted as overgenerated. This calibration step is important because the dataset is highly imbalanced and because the official score rewards correct document-level detection.

The main probability-level ensemble combines SciBERT and BiomedBERT predictions. The SciBERT classifier produces one positive-class probability, and the BiomedBERT classifier produces another positive-class probability. The final ensemble probability is computed as the average of the two probabilities:

$$p_E = \frac{p_S + p_B}{2}, \tag{2}$$

where p_E is the ensemble probability, p_S is the SciBERT probability, and p_B is the BiomedBERT probability.

The ensemble probability is then converted into a binary prediction using the calibrated threshold selected on the development set. This ensemble improves robustness because SciBERT and BiomedBERT capture complementary representations. SciBERT is pretrained on broad scientific text, while BiomedBERT is specialized for biomedical language [9, 10].

For Task 2.2, we extend the same source-aware architecture to multiclass classification. Instead of predicting only faithful versus overgenerated, the classifier predicts the exact distortion type. The final classification-oriented systems combine a strong binary overgeneration mask with multiclass label assignment. In this setting, the calibrated binary detector first determines which sentences should be considered overgenerated. Then, the multiclass classifier assigns a fine-grained label to the positive sentences. This design prevents the multiclass model from over-predicting rare distortion categories and makes use of the stronger binary detector.

Overall, the methodology consists of four steps: evidence retrieval, source-aware input construction, Transformer-based probability estimation, and calibration or label projection. This pipeline is simple, reproducible, and directly aligned with the central requirement of the task: identifying whether simplified biomedical sentences remain grounded in their original source evidence.

5.4. Implementation and Environment

All Task 2 experiments were implemented in Python. Classical machine-learning models were trained using scikit-learn and LightGBM [42]. Transformer-based models were implemented using PyTorch and the Hugging Face Transformers library [39, 40]. The final systems used supervised fine-tuning with cross-entropy loss, AdamW optimization, linear learning-rate scheduling, gradient accumulation, and a maximum input length of 384 tokens.

For the neural models, source-aware instances were tokenized as sentence pairs. The simplified sentence was used as the first segment and the retrieved source evidence as the second segment. This allowed the encoder to attend jointly to the generated sentence and to the biomedical evidence. Classification was performed using the final pooled representation followed by a task-specific classification head.

The main Transformer models were trained for one epoch using a batch size of 16 and gradient accumulation of 2. We used a learning rate of 5×10^{-6}, which provided stable fine-tuning for the biomedical encoders. Model selection and threshold calibration were performed on the development set using document-level F1, matching the official Task 2.1 evaluation protocol.

Experiments were run on a server equipped with an NVIDIA RTX A6000 GPU with 48 GB of memory. The software environment included Python, PyTorch with CUDA support, Hugging Face Transformers, scikit-learn, LightGBM, NumPy, pandas, and tqdm. GPU acceleration was required for the Transformer models, especially because the flattened dataset contained hundreds of thousands of sentence-level instances.

The final submissions were produced as JSON files following the official submission format and were then compressed into ZIP archives. Before submission, each file was validated to ensure that all required identifiers were present, that the number of predictions matched the expected number of instances, and that the ZIP archive contained the correct prediction file. This validation step was necessary to avoid format-related submission errors.

5.5. Results

5.5.1. Evaluation Metrics

The official evaluation of CLEF 2026 SimpleText Task 2 is performed primarily at document level, although participating systems produce sentence-level predictions. For Task 2.1, the goal is binary overgeneration identification. A document is considered positive if at least one of its simplified sentences is predicted as overgenerated. Therefore, the main evaluation metric for Task 2.1 is document-level F1.

We report precision, recall, and F1 for binary identification at both document and sentence level. Document-level precision measures the proportion of predicted positive documents that are truly positive, while document-level recall measures the proportion of true positive documents that are successfully detected by the system. F1 is the harmonic mean of precision and recall:

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{3}$$

This metric is appropriate for Task 2.1 because the dataset is highly imbalanced, with substantially more faithful sentences than overgenerated sentences. Accuracy alone would not adequately reflect the ability of a system to detect rare but important overgeneration cases. During development, we therefore tuned binary decision thresholds using document-level F1 rather than sentence-level accuracy.

In addition to document-level scores, we also report sentence-level precision, recall, and F1 as diagnostic metrics. These scores show how well the system identifies individual overgenerated sentences before predictions are aggregated to the document level. Sentence-level metrics are useful for analysing local detection behaviour, while document-level metrics better reflect the official task setting, where a document-level decision is required.

For Task 2.2, systems must assign fine-grained labels to simplified sentences. The label set includes faithful simplification, generation failure, grounded overgeneration, leaked instructions, repetitive content, and ungrounded injection. This setting is more difficult than binary identification because the system must not only detect whether overgeneration is present, but also distinguish among different types of information distortion.

For the fine-grained setting, we report accuracy together with document-level and sentence-level precision, recall, and F1. Accuracy measures the proportion of instances assigned to the correct fine-grained class, while the F1-based metrics show how well the system detects and aggregates overgeneration-related decisions. This distinction is important because the system with the highest document-level F1 is not necessarily the same as the system with the highest multi-class accuracy. Therefore, Task 2.2 results should be interpreted using both accuracy and F1-based metrics.

5.5.2. Binary Overgeneration Detection

Table 7 presents representative official Task 2.1 binary overgeneration detection results, ordered by document-level F1. When multiple submitted runs achieved the same or nearly identical official score, only representative runs are shown in order to keep the comparison concise. The table includes classical lexical baselines, tree-based models, single Transformer encoders, probability-level Transformer ensembles, and post-processing variants.

The official Task 2.1 results show that the strongest systems are the source-aware Transformer ensembles, especially the variants combining SciBERT and BiomedBERT. The best submitted run, the BERT-average leak-tuned selection with AddP2, achieved the highest document-level F1 score of **0.7994**, with document-level precision 0.8085 and recall 0.7906. At sentence level, the same run achieved an F1 score of 0.8606, with precision 0.8443 and recall 0.8775. This indicates that the system was able to identify overgenerated content with a balanced trade-off between precision and recall, while also maintaining strong sentence-level detection performance.

The next group of runs confirms the stability of the SciBERT–BiomedBERT ensemble. Threshold variants between 0.520 and 0.5425 all achieved very similar document-level F1 scores, ranging from 0.7957 to 0.7991. This small variation shows that the ensemble was not highly sensitive to minor

threshold changes. The best pure SciBERT–BiomedBERT average ensemble, with threshold 0.5225, achieved 0.7991 document-level F1, only 0.0003 below the best overall run. Its document-level precision was 0.8077 and its recall was 0.7906, showing nearly the same precision–recall balance as the top-ranked system.

Table 7: Representative official DUTH results for Task 2.1 binary overgeneration detection, covering both top-performing and diverse model variants.

Rank	Model / Strategy	$\mathbf{F1}_{\mathrm{doc}}$	$\mathbf{Prec.}_{\mathrm{doc}}$	$\mathbf{Recall}_{\mathrm{doc}}$	$\mathbf{F1}_{\mathrm{snt}}$	$\mathbf{Prec.}_{\mathrm{snt}}$	$\mathbf{Recall}_{\mathrm{snt}}$
1	BERT-average leak-tuned selection with AddP2	**0.7994**	0.8085	0.7906	0.8606	0.8443	0.8775
2	SciBERT/BiomedBERT average ensemble, TH=0.5225	0.7991	0.8077	0.7906	0.8602	0.8436	0.8775
3	SciBERT/BiomedBERT average ensemble, TH=0.520	0.7987	0.8069	0.7906	0.8599	0.8429	0.8775
4	SciBERT/BiomedBERT average ensemble, TH=0.525	0.7983	0.8081	0.7888	0.8601	0.8442	0.8767
5	SciBERT/BiomedBERT average ensemble, TH=0.5285	0.7981	0.8087	0.7879	0.8603	0.8450	0.8761
6	SciBERT/BiomedBERT weighted average ensemble	0.7978	0.7985	0.7971	0.8588	0.8405	0.8778
7	SciBERT/BiomedBERT average ensemble, TH=0.5325	0.7974	0.8091	0.7860	0.8601	0.8457	0.8750
8	SciBERT/BiomedBERT average ensemble, TH=0.5375	0.7964	0.8120	0.7815	0.8601	0.8474	0.8731
9	BERT ensemble with SciBERT/BioBERT weighting, TH=0.5975	0.7959	0.8178	0.7750	0.8567	0.8448	0.8690
10	SciBERT/BiomedBERT average ensemble, TH=0.5425	0.7957	0.8124	0.7796	0.8605	0.8485	0.8728
11	BERT-average strict-length rescue, TH=220	0.7956	0.7996	0.7916	0.8593	0.8420	0.8772
12	LLaMA-3.1 QLoRA light source-aware selection, calibrated variant	0.7952	0.7989	0.7916	0.8591	0.8418	0.8772
13	BERT ensemble with SciBERT/BiomedBERT/BioLinkBERT weighting	0.7947	0.8155	0.7750	0.8597	0.8470	0.8728
14	SciBERT/BiomedBERT average ensemble, TH=0.5025	0.7939	0.8028	0.7851	0.8550	0.8361	0.8748
15	BERT-average strict-length rescue, TH=200	0.7923	0.7912	0.7934	0.8582	0.8394	0.8778
16	BlueBERT source-aware selection	0.7246	0.7588	0.6933	0.8257	0.8328	0.8186
17	LLaMA-3.1 QLoRA light source-aware selection, alternative variant	0.6296	0.7171	0.5611	0.6917	0.7720	0.6265
18	LLaMA-3.1 QLoRA regularized source-aware selection	0.6143	0.6607	0.5739	0.6818	0.7417	0.6309
19	LightGBM fast source-aware baseline	0.6074	0.5631	0.6593	0.7102	0.6667	0.7597
20	LightGBM memorized strict source-aware baseline	0.5789	0.5105	0.6685	0.6615	0.5834	0.7638
21	LightGBM source-aware baseline, TH=0.70	0.5648	0.7314	0.4601	0.7475	0.8130	0.6917
22	ExtraTrees source-aware baseline, TH=0.35	0.5603	0.5550	0.5657	0.6919	0.6748	0.7099
23	TF–IDF logistic regression baseline	0.4570	0.3409	0.6933	0.2271	0.1322	0.8062
24	TF–IDF logistic regression baseline, TH=0.55	0.4563	0.3462	0.6694	0.2300	0.1344	0.7960
25	Qwen2.5-7B LLM-as-a-judge selection	0.4046	0.2728	0.7833	0.3422	0.2377	0.6111

Threshold calibration mainly affected the balance between precision and recall. For example, the SciBERT–BiomedBERT average ensemble with threshold 0.5375 achieved higher document-level precision, 0.8120, but lower recall, 0.7815, leading to a slightly lower F1 score of 0.7964. Similarly, the BERT ensemble with SciBERT/BioBERT weighting and threshold 0.5975 achieved one of the highest precision values, 0.8178, but its recall decreased to 0.7750, resulting in 0.7959 document-level F1. These results suggest that stricter thresholds reduce false positives but may miss more true overgeneration cases. In contrast, the best-performing runs retained recall close to 0.7906 while keeping precision above 0.806, which produced the strongest F1 scores.

The sentence-level metrics are consistently higher than the document-level metrics for the top systems. The best run achieved 0.8606 sentence-level F1, while the corresponding document-level F1 was 0.7994. This difference is expected because document-level evaluation is stricter: a system must make consistent decisions across all relevant sentences in a document. Nevertheless, the top systems maintained high sentence-level recall, around 0.8775, showing that they were effective at detecting many overgenerated sentences. Their sentence-level precision, around 0.844, also indicates that the models did not simply over-predict the positive class.

Single-model and non-ensemble Transformer variants performed below the best ensemble systems. BlueBERT source-aware selection achieved 0.7246 document-level F1, with precision 0.7588 and recall 0.6933. Although this result is clearly better than the weaker baselines, it remains substantially below the SciBERT–BiomedBERT ensembles. This suggests that combining complementary biomedical encoders at probability level provides a more robust grounding signal than relying on a single biomedical Transformer model.

The LLaMA-3.1 QLoRA source-aware variants showed mixed behaviour. One light source-aware selection run reached 0.7952 document-level F1, which is close to the best ensemble group, with precision 0.7989 and recall 0.7916. However, other QLoRA variants were much weaker, with document-level F1 scores of 0.6296 and 0.6143. This gap indicates that generative or lightly fine-tuned LLM-based

classification can be competitive in some configurations, but it is less stable than the supervised SciBERT–BiomedBERT probability ensembles.

The LightGBM source-aware baseline achieved 0.6074 document-level F1, with precision 0.5631 and recall 0.6593. Its higher recall than precision suggests that the model identified many possible overgeneration cases, but also produced more false positives. Compared with this baseline, the best SciBERT–BiomedBERT ensemble improved document-level F1 by 0.1920 absolute points, corresponding to a relative improvement of approximately 31.6%. This confirms that lexical and tree-based features are useful but insufficient for reliable biomedical overgeneration detection.

The additional baseline runs further clarify the limitations of lexical and tree-based source-aware methods. The LightGBM memorized strict variant achieved 0.5789 document-level F1, with precision 0.5105 and recall 0.6685. This result shows relatively high recall but lower precision, suggesting that the model identified many possible overgeneration cases but also introduced false positives. The thresholded LightGBM variant with TH=0.70 obtained 0.5648 document-level F1, with much higher precision 0.7314 but lower recall 0.4601. This confirms the expected effect of a stricter threshold: the system becomes more conservative, but misses more true overgeneration cases.

The ExtraTrees source-aware baseline achieved 0.5603 document-level F1, with a more balanced precision–recall profile of 0.5550 and 0.5657, respectively. Although this was competitive with the lower LightGBM variants, it remained far below the Transformer ensembles. The TF–IDF logistic regression baselines were weaker, achieving 0.4570 and 0.4563 document-level F1. Both TF–IDF variants had relatively high recall, 0.6933 and 0.6694, but low precision, 0.3409 and 0.3462. This indicates that surface lexical overlap can help retrieve many candidate overgeneration cases, but it is not precise enough to distinguish unsupported biomedical content from faithful simplification.

Overall, these additional baseline runs support the same conclusion as the main comparison: engineered lexical and tree-based methods provide useful reference points, but they lack the semantic grounding capacity required for reliable biomedical overgeneration detection. The large gap between these baselines and the best source-aware Transformer ensembles shows that domain-specific pretrained encoders and calibrated probability-level ensembling are essential for strong Task 2.1 performance.

The zero-shot Qwen2.5-7B LLM-as-a-judge run obtained the weakest result among the submitted variants, with 0.4046 document-level F1. Its recall was relatively high, 0.7833, but its precision was very low, 0.2728. This means that the zero-shot judge tended to mark many sentences as overgenerated, producing many false positives. The same pattern appears at sentence level, where recall was 0.6111 but precision was only 0.2377. These results show that generic instruction-following ability alone is not sufficient for this task. Reliable biomedical overgeneration detection requires explicit source evidence, supervised training, and careful calibration.

Overall, the results demonstrate that Task 2.1 benefits most from source-aware biomedical Transformer modelling. The best systems combine three important components: retrieved evidence from the source abstract, domain-specific pretrained encoders, and calibrated probability-level ensembling. These components allow the classifier to distinguish supported simplifications from unsupported or distorted generated content more reliably than lexical baselines, tree-based classifiers, or zero-shot LLM judging.

5.5.3. Fine-Grained Overgeneration Classification

Table 8 reports representative official Task 2.2 multiclass classification results. Task 2.2 is more difficult than binary identification because the system must not only detect whether overgeneration is present, but also assign the correct fine-grained distortion category.

The Task 2.2 results show that fine-grained overgeneration classification is more difficult than binary identification, because systems must both detect unsupported generated content and assign an appropriate distortion category. The strongest runs were source-aware Transformer-based systems and gated variants that reused the binary overgeneration signal from Task 2.1.

The highest accuracy was achieved by the SciBERT multi-class source-aware classifier with threshold 0.65, which obtained **0.7878** accuracy, 0.7901 document-level F1, 0.8222 document-level precision, and

0.7603 document-level recall. At sentence level, the same system achieved 0.8518 F1, with precision 0.8541 and recall 0.8494. This shows that a direct source-aware SciBERT multi-class classifier can provide a strong and balanced solution for assigning fine-grained overgeneration labels.

Table 8: Representative official DUTH results for Task 2.2 fine-grained overgeneration classification, covering both top-performing and diverse model variants.

Rank	Model / Strategy	Acc.	$F1_{doc}$	$Prec._{doc}$	$Recall_{doc}$	$F1_{snt}$	$Prec._{snt}$	$Recall_{snt}$
1	SciBERT multi-class source-aware classifier, TH=0.65	**0.7878**	0.7901	0.8222	0.7603	0.8518	0.8541	0.8494
2	Gated BERT-average with SciBERT fallback	0.7870	0.7981	0.8087	0.7879	0.8602	0.8447	0.8761
3	SciBERT source-aware classifier, TH=0.70	0.7798	0.7844	0.8241	0.7484	0.8473	0.8587	0.8362
4	Gated BERT-average with SciMulti threshold 0.625	0.7729	0.7981	0.8087	0.7879	0.8602	0.8447	0.8761
5	Same-mask classification labels with AddP2	0.7707	0.7994	0.8085	0.7906	0.8606	0.8443	0.8775
6	Gated BERT-average with SciMulti fallback grounding	0.7702	0.7981	0.8087	0.7879	0.8602	0.8447	0.8761
7	Gated BERT-average with threshold 0.5325 and SciMulti fallback	0.7699	0.7974	0.8091	0.7860	0.8601	0.8457	0.8750
8	Gated BERT-average with SciMulti strict selection	0.7691	0.7770	0.8529	0.7135	0.8504	0.8733	0.8288
9	Gated BERT-average with SciMulti threshold 0.675	0.7677	0.7981	0.8087	0.7879	0.8602	0.8447	0.8761
10	LightGBM multiclass classifier, TH=0.50	0.7030	0.5898	0.4944	0.7309	0.6839	0.6005	0.7941
11	LightGBM multiclass classifier, TH=0.62	0.6741	0.5934	0.5755	0.6125	0.7250	0.6989	0.7531
12	Gated BERT-average with DeBERTa positive labeler	0.3278	0.7981	0.8087	0.7879	0.8602	0.8447	0.8761

The gated BERT-average variants were also highly competitive. The gated BERT-average system with SciBERT fallback achieved 0.7870 accuracy and 0.7981 document-level F1, with precision 0.8087 and recall 0.7879. Several related gated variants, including the SciMulti threshold and fallback configurations, obtained very similar document-level F1 scores around 0.7981. These results suggest that combining a strong binary overgeneration mask with a separate multi-class source-aware labeler is an effective strategy. The binary component helps decide whether a sentence is problematic, while the multi-class component assigns the specific distortion type.

The run based on same-mask classification labels with AddP2 achieved the highest document-level F1 score in the table, 0.7994, with precision 0.8085 and recall 0.7906. Its sentence-level F1 was also the highest, 0.8606, with precision 0.8443 and recall 0.8775. However, its accuracy was 0.7707, lower than the top SciBERT multi-class classifier. This difference indicates that optimizing for document-level F1 and optimizing for overall multi-class accuracy do not always select the same system. For this reason, the results should be interpreted using both accuracy and F1-based metrics.

Thresholding again affected the precision–recall balance. The SciBERT source-aware classifier with threshold 0.70 achieved high document-level precision, 0.8241, but lower recall, 0.7484, resulting in 0.7844 document-level F1. Similarly, the strict gated BERT-average configuration achieved the highest document-level precision in the table, 0.8529, but its recall dropped to 0.7135. These patterns show that stricter decision rules make the classifier more conservative, reducing false positives but missing more true overgeneration cases.

The sentence-level scores were consistently higher than the document-level scores for the best systems. For example, the gated BERT-average with SciBERT fallback achieved 0.8602 sentence-level F1 compared with 0.7981 document-level F1. This mirrors the Task 2.1 results and suggests that document-level evaluation is stricter, since errors across multiple sentences can affect the final document-level decision. Nevertheless, the top systems maintained strong sentence-level recall, around 0.876–0.878, indicating that they detected many overgenerated sentences before assigning fine-grained labels.

The classical LightGBM multi-class baselines were clearly weaker than the Transformer-based systems. With threshold 0.50, LightGBM achieved 0.7030 accuracy but only 0.5898 document-level F1, with low precision 0.4944 and higher recall 0.7309. This suggests that the model detected many positive cases but produced many false positives. Increasing the threshold to 0.62 improved precision to 0.5755 and produced a slightly higher document-level F1 of 0.5934, but accuracy decreased to 0.6741 and recall dropped to 0.6125. Thus, threshold adjustment improved the precision–recall balance but did not close the gap with source-aware Transformer models.

The DeBERTa positive-labeler variant produced an unusual result: it retained a high document-level F1 score of 0.7981 but achieved very low accuracy, 0.3278. This indicates that the positive/negative overgeneration mask remained strong, but the fine-grained class assignment was not reliable across all

categories. Therefore, good binary detection alone is not sufficient for Task 2.2; the system must also assign the correct distortion label.

Overall, the Task 2.2 results confirm that fine-grained biomedical overgeneration classification benefits from the same principles as Task 2.1: explicit source evidence, domain-specific Transformer encoders, calibrated thresholding, and careful separation between binary detection and multi-class label assignment. The best-performing systems were not simple lexical or tree-based classifiers, but source-aware Transformer and gated ensemble variants that combined strong detection with more informed category assignment.

5.5.4. Summary of Task 2 Findings

Overall, the results confirm that source-aware biomedical Transformer classifiers are effective for identifying and classifying overgeneration in simplified scientific texts. Our best Task 2.1 system achieved an official document-level F1 score of 0.7994. For Task 2.2, the same-mask AddP2 variant achieved the highest document-level F1 score of 0.7994, while the highest-accuracy multiclass system achieved 0.7878 accuracy and 0.7901 document-level F1.

The main finding is that explicit source evidence, domain-specific pretraining, probability-level ensembling, and document-level threshold calibration are key components for reliable biomedical hallucination and overgeneration detection. Classical lexical models and tree-based classifiers provided useful baselines, but they were not sufficient for the semantic grounding required by the task. Zero-shot LLM judging was also not competitive, indicating that supervision, source-aware input construction, and calibrated decision thresholds are essential in this biomedical reliability setting.

6. General Discussion

The two SimpleText tasks addressed in this paper are closely connected. Task 1 focuses on biomedical text simplification, while Task 2 focuses on detecting unsupported or distorted content in simplified biomedical outputs. Together, they show that biomedical simplification is not only a readability problem, but also a factuality and grounding problem. A useful simplification system must preserve factual meaning, evidence strength, numerical findings, comparison groups, outcomes, and uncertainty expressions, because small wording changes can alter the interpretation of biomedical evidence [1, 4].

A central finding is that grounding is essential for both generation and classification. In simplification, grounding constrains fluent rewriting so that readability does not come at the cost of factual distortion. In overgeneration detection, grounding allows simplified sentences to be judged against source evidence rather than in isolation. This supports prior work showing that hallucination and factual inconsistency should be evaluated with respect to the input document, not only through fluency or reference overlap [5, 6, 7].

The results indicate that controlled systems were more reliable than unconstrained ones. For Task 1, instruction-tuned LLMs were effective when combined with factuality-oriented prompts, moderate post-processing, and conservative sentence-to-document aggregation. For Task 2, lexical and tree-based baselines captured useful surface signals, but the strongest results came from supervised biomedical models using source evidence and calibrated decisions. This suggests that reliability in biomedical NLP depends more on grounding, domain-specific modelling, and task-specific control than on model size alone [9, 10].

The evaluation results also highlight the limitations of automatic metrics. SARI is useful for measuring simplification operations, and document-level F1 is useful for overgeneration detection [18]. However, these metrics do not fully capture the biomedical consequences of altered uncertainty, missing qualifiers, or unsupported causal claims. Complementary metrics such as BERTScore and LENS can provide additional signals, but biomedical simplification still requires evaluation settings that consider readability and factual correctness together [20, 21].

Overall, the DUTH participation in CLEF 2026 SimpleText shows that biomedical simplification and overgeneration detection should be treated as two sides of a reliability-oriented pipeline. The generator

should simplify without adding unsupported content, and the detector should identify when such content appears. Grounded prompting, evidence-aware modelling, and conservative decision-making therefore emerge as key principles for robust biomedical text simplification.

7. Conclusion

This paper presented Team DUTH's participation in the CLEF 2026 SimpleText track, addressing two complementary biomedical NLP tasks: biomedical text simplification and overgeneration detection. Task 1 focused on sentence-level and document-level simplification, while Task 2 focused on detecting unsupported, distorted, repetitive, or prompt-leaked content in simplified biomedical texts. Across both tasks, our systems followed a common reliability-oriented principle: generated or classified outputs should remain grounded in the original biomedical evidence.

For Task 1, we treated biomedical simplification as grounded rewriting. The results show that open-source instruction-tuned language models can be effective when generation is constrained through prompts, moderate post-processing, and sentence-to-document aggregation. For Task 2, we treated overgeneration detection as evidence-aware sentence classification. The strongest results came from supervised biomedical Transformer models that used source evidence, probability-level ensembling, and calibrated document-level decisions.

Overall, the findings show that biomedical simplification and overgeneration detection should be studied together. A useful simplification system must improve readability without introducing unsupported claims, while an overgeneration detector must identify cases where simplified text changes, adds, repeats, or distorts biomedical information. This is especially important in biomedical domains, where small changes in uncertainty, numerical findings, comparison groups, or outcomes can alter the interpretation of evidence.

A central lesson from our participation is that controlled and evidence-aware methods were more reliable than unconstrained generation or zero-shot judging. Grounded prompting helped constrain simplification, while source-aware classification helped detect unsupported content. These results support the broader view that biomedical NLP systems should combine fluency and readability with explicit factuality checks against the source document.

Future work will explore stronger validation and calibration methods for both tasks. For simplification, this includes coherence-aware sentence-to-document aggregation, factuality-aware post-processing, and better control of uncertainty and evidence strength. For overgeneration detection, future work should improve evidence retrieval through semantic matching, cross-encoder reranking, or natural language inference. A further direction is to integrate simplification generation and overgeneration detection into a single reliability-oriented pipeline, where simplified outputs are automatically checked before use.

Together, the DUTH systems show that reliable biomedical text simplification requires more than fluent rewriting. It also requires careful preservation and verification of biomedical meaning through grounding, domain-specific modelling, evidence-aware classification, and conservative decision-making.

8. Acknowledgments

We thank the organizers of the CLEF 2026 SimpleText track for providing the task data, evaluation platform, and shared-task infrastructure.

9. Declaration on Generative AI

Large language models and generative AI systems were used as part of the experimental systems described in this paper. In particular, instruction-tuned models were used for biomedical text simplification, zero-shot judging, and LLM-based experimental variants. All outputs, experiments, results, and conclusions were checked and approved by the authors.

References

[1] H. Saggion, Automatic Text Simplification, Synthesis Lectures on Human Language Technologies, Morgan & Claypool Publishers, 2017. doi:10.2200/S00700ED1V01Y201602HLT032.

[2] S. Nisioi, S. Štajner, S. P. Ponzetto, L. P. Dinu, Exploring neural text simplification models, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Volume 2: Short Papers, Association for Computational Linguistics, 2017, pp. 85–91. doi:10.18653/v1/P17-2014.

[3] L. Ermakova, E. SanJuan, S. Huet, H. Azarbonyad, G. M. Di Nunzio, F. Vezzani, J. D'Souza, J. Kamps, Overview of the CLEF 2024 SimpleText track: Improving access to scientific texts for everyone, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction, Lecture Notes in Computer Science, Springer, 2024. doi:10.1007/978-3-031-71908-0_13.

[4] J. Bakker, B. Vendeville, L. Ermakova, J. Kamps, Overview of the CLEF 2025 SimpleText task 1: Simplify scientific text, in: Working Notes of CLEF 2025 – Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.

[5] J. Maynez, S. Narayan, B. Bohnet, R. McDonald, On faithfulness and factuality in abstractive summarization, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, 2020, pp. 1906–1919. doi:10.18653/v1/2020.acl-main.173.

[6] W. Kryscinski, B. McCann, C. Xiong, R. Socher, Evaluating the factual consistency of abstractive text summarization, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2020, pp. 9332–9346. doi:10.18653/v1/2020.emnlp-main.750.

[7] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Chen, W. Dai, H. S. Chan, A. Madotto, P. Fung, Survey of hallucination in natural language generation, ACM Computing Surveys 55 (2023) 1–38. doi:10.1145/3571730.

[8] Gemma Team, Gemma 2: Improving open language models at a practical size, arXiv preprint arXiv:2408.00118 (2024).

[9] I. Beltagy, K. Lo, A. Cohan, SciBERT: A pretrained language model for scientific text, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2019, pp. 3615–3620. doi:10.18653/v1/D19-1371.

[10] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Domain-specific language model pretraining for biomedical natural language processing, ACM Transactions on Computing for Healthcare 3 (2021) 1–23. doi:10.1145/3458754.

[11] F. Alva-Manchego, C. Scarton, L. Specia, Sentence simplification by monolingual machine translation, in: Proceedings of the 27th International Conference on Computational Linguistics, Association for Computational Linguistics, 2018, pp. 346–357.

[12] X. Zhang, M. Lapata, Sentence simplification with deep reinforcement learning, Transactions of the Association for Computational Linguistics 5 (2017) 595–610.

[13] L. Martin, A. Fan, E. de la Clergerie, A. Bordes, B. Sagot, MUSS: Multilingual unsupervised sentence simplification by mining paraphrases, in: Proceedings of the 12th Language Resources and Evaluation Conference, 2020, pp. 1651–1664.

[14] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, et al., Scaling instruction-finetuned language models, arXiv preprint arXiv:2210.11416 (2022).

[15] 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, 2020, pp. 7871–7880.

[16] Qwen Team, Qwen2.5 technical report, arXiv preprint arXiv:2412.15115 (2025).

[17] A. Dubey, A. Jauhri, A. Pandey, A. Kadian, et al., The llama 3 herd of models, arXiv preprint arXiv:2407.21783 (2024).

[18] W. Xu, C. Napoles, E. Pavlick, Q. Chen, C. Callison-Burch, Optimizing statistical machine trans-

lation for text simplification, in: Transactions of the Association for Computational Linguistics, volume 4, 2016, pp. 401–415. doi:10.1162/tacl_a_00107.
[19] 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. doi:10.3115/1073083.1073135.
[20] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, Y. Artzi, BERTScore: Evaluating text generation with BERT, in: International Conference on Learning Representations, 2020.
[21] T. Goyal, L. Martin, D. Kumar, G. Durrett, LENS: A learned evaluation metric for text simplification, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, 2022.
[22] B. Chakar, J. Bakker, L. Ermakova, J. Kamps, Overview of the CLEF 2026 SimpleText Task 2: Identify and Avoid Hallucination, in: [43], 2026.
[23] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, 2019, pp. 4171–4186.
[24] M. Yasunaga, J. Leskovec, P. Liang, Linkbert: Pretraining language models with document links, arXiv preprint arXiv:2203.15827 (2022).
[25] G. Arampatzis, A. Arampatzis, DUTH at CLEF 2025 SimpleText track: Tackling scientific text simplification and hallucination detection, in: Working Notes of CLEF 2025 – Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.
[26] G. Arampatzis, V. Perifanis, S. Symeonidis, A. Arampatzis, DUTH at SemEval-2023 task 9: An ensemble approach for twitter intimacy analysis, in: Proceedings of the 17th International Workshop on Semantic Evaluation, Association for Computational Linguistics, 2023.
[27] G. Arampatzis, V. Perifanis, S. Symeonidis, A. Arampatzis, DUTH at EXIST 2025: Multilingual sexism detection with soft labels and transformers, in: Working Notes of CLEF 2025 – Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.
[28] G. Arampatzis, A. Arampatzis, DUTH at SemEval-2026 task 9: Joint multilingual fine-tuning for online polarization detection, in: Proceedings of the 20th International Workshop on Semantic Evaluation (SemEval-2026), Association for Computational Linguistics, 2026.
[29] G. Arampatzis, A. Arampatzis, DUTH at SemEval-2026 task 3: Multilingual transformer models for dimensional stance prediction across tasks, in: Proceedings of the 20th International Workshop on Semantic Evaluation, Association for Computational Linguistics, 2026.
[30] G. Arampatzis, A. Arampatzis, DUTH at SemEval-2026 task 1: Prompt-based zero-shot large language models for constrained multilingual humor generation, in: Proceedings of the 20th International Workshop on Semantic Evaluation, Association for Computational Linguistics, 2026.
[31] G. Arampatzis, I. Maslaris, A. Arampatzis, Llm-based question generation and retrieval-augmented reporting for news credibility, in: The Thirty-Fourth Text REtrieval Conference Proceedings (TREC 2025), NIST, 2025.
[32] G. Arampatzis, V. Perifanis, A. Arampatzis, Justification retrieval with llms, retrieval-augmented generation, and hybrid labels, in: Proceedings of the Thirty-Fourth Text REtrieval Conference (TREC 2025), NIST, 2025.
[33] G. Arampatzis, S. Symeonidis, A. Arampatzis, Precision by design: Rm3 and fusion in product search, in: Proceedings of the Thirty-Fourth Text REtrieval Conference (TREC 2025), NIST, 2025.
[34] G. Arampatzis, K. Safouri, A. Arampatzis, Bridging lexical and neural ranking for topic-oriented retrieval, in: Proceedings of the Thirty-Fourth Text REtrieval Conference (TREC 2025), NIST, 2025.
[35] G. Arampatzis, A. Arampatzis, Hybrid sparse-neural fusion for passage retrieval, in: Proceedings of the Thirty-Fourth Text REtrieval Conference (TREC 2025), NIST, 2025.
[36] G. Arampatzis, A. Arampatzis, DUTH at CLEF JOKER 2025 tasks 2 and 3: Translating puns and proper names with neural approaches, in: Working Notes of CLEF 2025 – Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2025.
[37] G. Arampatzis, A. Arampatzis, Multilingual humour-aware retrieval with dense and re-ranking models, arXiv preprint arXiv:2605.25165 (2026).

[38] J. Bakker, L. Ermakova, J. Kamps, Overview of the CLEF 2026 SimpleText Task 1: Simplify Scientific Text, in: [43], 2026.
[39] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, PyTorch: An imperative style, high-performance deep learning library, in: Advances in Neural Information Processing Systems, volume 32, 2019.
[40] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, A. M. Rush, Transformers: State-of-the-art natural language processing, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, 2020, pp. 38–45.
[41] G. Salton, C. Buckley, Term-weighting approaches in automatic text retrieval, Information Processing & Management 24 (1988) 513–523. doi:10.1016/0306-4573(88)90021-0.
[42] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, LightGBM: A highly efficient gradient boosting decision tree, in: Advances in Neural Information Processing Systems, volume 30, 2017.
[43] E. Sánchez Salido, A. Barrón-Cedeño, A. García Seco de Herrera, S. MacAvaney, J. M. Struß (Eds.), Working Notes of CLEF 2026: Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS.org, 2026.