Overview of the CLEF 2026 SimpleText Task 2: Identify and Avoid Hallucination

Berkay Chakar[2], Liana Ermakova[2], Jan Bakker[1] and Jaap Kamps[1]

[1]University of Amsterdam, Amsterdam, The Netherlands
[2]Université de Bretagne Occidentale, HCTI, France

Abstract

This paper presents an overview of the CLEF 2026 SimpleText Task 2 on Controlled Creativity. The task aims to identify and avoid hallucination. We discuss the data and benchmarks provided for these tasks, along with preliminary insights and anticipated challenges. Our main findings are the following. First, we observed high performance for document-level overgeneration detection (Task 2.1) and overgeneration classification (Task 2.2). These encouraging results suggest that almost any generative AI output can be improved by training specific models to detect and classify remaining issues in text generation tasks. Second, the track's setup exploits the text simplification setup, with sources, predictions, and references in the same language. The results were obtained in a reference-free setting, allowing them to generalize to almost any NLP task where text generation models are used. Third, the overgeneration detection and classification data show how prevalent overgeneration is in NLP benchmarks and which types of overgeneration are most common. This is a sobering realization of the scope and scale of the remaining issues, despite relatively high performance as measured by textual overlap or LLM-as-a-judge evaluation measures. More generally, we hope and expect that the constructed corpora and evaluation data will be used by researchers to further advance text simplification approaches, both in general and specifically for the biomedical domain.

Keywords

Scientific text simplification, Biomedical AI, Generative AI, Information access, Natural language processing

1. Introduction

Becoming scientifically literate is increasingly crucial. Objective scientific information helps people navigate a world in which misinformation, disinformation, and fabricated claims are only a click away. Although the importance of such information is widely recognized, the general public rarely engages with primary scientific sources. This raises a central question: how can we make scientific texts more accessible to everyone?

The impact of objective scientific information is especially clear in biomedicine, where research findings directly influence health decisions. Yet, the most reliable and up-to-date sources use specialized language and assume substantial prior knowledge, making them inaccessible to non-experts. Our track therefore centers on a concrete, societally relevant biomedical use case and treats accessibility as a core objective, distinguishing our work from earlier text simplification approaches that considered lexical and grammatical complexity in isolation.

Scientific text simplification offers a way to make these sources more approachable, and recent advances in large language models have accelerated progress in this area. Nonetheless, key challenges remain: dealing with domain-specific jargon and preserving the accuracy of the original information. Addressing these challenges requires high-quality training data and rigorously designed benchmark datasets, informed by detailed human assessments and systematic analyses of the limitations of current summarization and simplification systems.

To address these challenges, the CLEF 2026 SimpleTrack has three main aims: First, to *advance scientific text simplification* by expanding existing corpora with more varied paragraph- and document-

CLEF 2026 Working Notes, 21 – 24 September 2026, Jena, Germany
berkay.chakar2@student.uva.nl (B. Chakar); liana.ermakova@univ-brest.fr (L. Ermakova); j.bakker@uva.nl (J. Bakker); kamps@uva.nl (J. Kamps)
0000-0002-7598-7474 (L. Ermakova); 0009-0002-9085-8491 (J. Bakker); 0000-0002-6614-0087 (J. Kamps)

level simplifications that respect complex discourse structure. This new biomedical corpus is built from aligned Cochrane abstracts and plain language summaries [1, 2] and is tailored to models such as LLMs that handle long inputs. Second, to *detect and reduce hallucinations* by exploiting aligned sources, references, and model outputs to identify and quantify spurious information introduced by generative models. This tackles a key limitation of current systems and evaluation measures, which often fail to penalize unwarranted additional content—an especially critical issue for scientific applications and for NLP/IR more broadly. Third, to *classify scientific text* by predicting the field and the discipline of a scientific paper based on a given taxonomy. This accelerates consistent metadata across collections (arXiv, OpenAIRE, MADOC).

Hence, the CLEF 2025 SimpleText track is based on three interrelated tasks:

- ***Task 1: Text Simplification****: simplify scientific text.*
- ***Task 2: Controlled Creativity****: identify and avoid hallucination.*
- ***Task 3: Research Area Classification****: classification of scientific articles by research area.*

This paper gives an overview of the CLEF 2026 SimpleText Task 2 on Controlled Creativity, which aims to identify and avoid hallucination. Further details on the entire track are provided in the CLEF 2026 SimpleText Track Overview [3]. Additional details on Task 1 on *Text Simplification* are in a companion Task 1 overview paper [4], and details on Task 3 on *Research Area Classification* are in a companion Task 3 overview paper [5]. We also refer to the respective participants' papers for further details.

The rest of this paper is structured as follows. Section 2 describes the task, the data, the format, and the evaluation measures. Section 3 describes the participants' approaches. Section 4 provides detailed results for the task. Section 5 provides further analysis of the results. We end with a discussion and conclusions in Section 6.

2. Task 2: Identify and Avoid Hallucination

This section details *Task 2: Controlled Creativity* on identify and avoid hallucination.

2.1. Description

The *Controlled Creativity* task aims to *identify and avoid hallucination.*

Unexpectedly, the SimpleText track has accumulated a substantial amount of spurious (or over-)generated content from participating systems. Table **??** presents an example of a simplified output produced by one of the teams. For 2025, we estimate that 20 out of 70 submissions (29%) contain entire spurious sentences in at least 25% of the input sentences. Moreover, 15 submissions (21%) exhibit such over-generation in at least 50% of the input sentences [6, 7]. Thanks to the closely aligned sources, predictions, and references in our text simplification setting—and the fact that they are all in the same language—we can systematically study source attribution and creative variation, as well as detect and mitigate what is commonly referred to as "hallucinations."

This task extends earlier manual investigations of information distortion within our track, conducted since 2022 [8, 9, 10, 6], as well as related work by others [11, 12].

Description. Task 2.1 is to identify creative generation, at the abstract or document level. We will provide realistic system output from participants in earlier years. The task is to detect what sentences are fully grounded on source input (a) without and (b) with access to the source sentences, and hence also label those introducing significant new content. Task 2.1 is a post-hoc identification or explanation task.

Task 2.2 attempts to mimic the laborious human evaluation by LLMs. As text-generation-based automatic evaluation measures (such as BLEU and SARI), which rely on text overlap, are notoriously imprecise, we need to conduct human evaluations of fluency, simplicity, and factuality. Usually, these annotations are not reusable. We use them as the training and test data for systems to automatically detect and annotate such information distortion in real CLEF submissions. The manual annotations of

Document-level example — Simplified Cochrane abstract (CD014920).

Source: *"We ~~included three RCTs (~~ 1144 participants~~).~~ Participants were ~~randomised to receive~~ either ~~preoperative coronary revascularisation with PCI or CABG~~ plus usual care or ~~only~~ usual care before major vascular surgery. One ~~trial enrolled participants if they had no apparent evidence of~~ coronary artery disease. Another trial ~~selected participants classified as~~ high risk ~~for~~ coronary disease ~~through preoperative clinical and laboratorial testing~~. We ~~excluded~~ one ~~trial from the meta-analysis~~ because ~~participants from both the~~ control and ~~the intervention~~ groups were ~~eligible to undergo preoperative coronary revascularisation~~. We ~~identified~~ a high risk of ~~performance bias in all included trials, with one trial displaying a high risk~~ of ~~other~~ bias. ~~However, the~~ risk of bias was ~~either~~ low ~~or unclear~~ in ~~other domains~~. We ~~observed~~ no difference between groups ~~for perioperative acute myocardial infarction~~, but ~~the evidence is very uncertain~~. One ~~trial showed a reduction in incidence of long-term acute myocardial infarction in participants allocated to the~~ preoperative coronary interventions ~~plus usual care group~~, but the evidence ~~was very uncertain~~. ~~There was~~ little ~~to no~~ effect on ~~all-cause mortality in the perioperative period~~, but the evidence is ~~very~~ uncertain. ~~The evidence is very uncertain about the effect of~~ preoperative coronary interventions ~~on~~ long-term ~~all-cause~~ mortality. One study ~~reported~~ no ~~adverse~~ effects ~~related to coronary angiography~~, ~~whereas~~ the other two studies ~~reported~~ five deaths ~~due to revascularisations. There~~ may ~~be no effect on~~ cardiovascular mortality in the short term. ~~Preoperative coronary interventions plus usual care in the short term~~ may ~~reduce length of~~ hospital ~~stay slightly~~. We ~~downgraded~~ the ~~certainty~~ of the ~~evidence due to concerns about risk of bias, imprecision,~~ and ~~inconsistency. None of the included trials reported~~ on quality of life or ~~vessel graft patency, and no study reported on adverse~~ effects. Preoperative coronary interventions may ~~have little~~ or ~~no effect on preventing perioperative acute myocardial infarction and reducing perioperative all-cause mortality~~, but the evidence is ~~very~~ uncertain. ~~Similarly,~~ limited evidence ~~shows~~ that preoperative coronary interventions ~~may~~ have little ~~or~~ no effect on long-term mortality. ~~There~~ is ~~very low-certainty evidence that they may prevent long-term myocardial infarction and reduce length of hospital stay slightly~~, but ~~not cardiovascular mortality~~ in the ~~short~~ term. ~~Adverse~~ effects of ~~preoperative coronary~~ interventions were ~~poorly reported~~ in ~~trials. Quality~~ of life or graft ~~patency were not reported~~. We ~~downgraded~~ the ~~certainty of the evidence most frequently for high risk of bias, inconsistency,~~ or ~~imprecision. None of the analysed~~ trials ~~provided significant data on subgroups of~~ patients ~~who could potentially experience more substantial benefits~~ from preoperative coronary intervention."*

Prediction: *"We studied 3 experiments with 1144 participants. Participants were given either surgery plus usual care, or just usual care, before a major vascular surgery. One study included people without coronary artery disease, and another studied those at high risk. Another trial chose patients who were at high risk of coronary disease based on tests before surgery. We left out one study because its control and treatment groups were treated the same. We found that the trials had a high risk of some types of bias. The risk of bias was low in some areas, but not others. (I chose 'rephrase' as the simplification strategy to make the sentence more concise and easier to understand, while maintaining its original meaning.) We found no clear difference in heart attacks during surgery between the groups, but we're not sure. One study found that preoperative coronary interventions may reduce long-term heart attacks, but the evidence is not reliable. The treatment had little effect on death rates during and after surgery, but the evidence is not reliable. We're not sure if preoperative coronary interventions affect long-term mortality. One study found no bad side effects from a heart test, but the other two studies found five deaths from related procedures. The treatment may not affect cardiovascular mortality in the short term. The treatment may shorten hospital stays by about a day. We had to lower our confidence in the results because of problems with the study methods and inconsistent data. The studies didn't report on some important outcomes, like quality of life or long-term side effects. Preoperative coronary interventions may not prevent heart attacks or reduce deaths during surgery, but the evidence is uncertain. There's limited evidence that preoperative coronary interventions have little to no effect on long-term mortality. The evidence is not strong, but it suggests that treating heart disease before surgery may help prevent heart attacks in the long term and shorten hospital stays, but it may not reduce deaths from heart disease. The side effects of these interventions were not well documented in the studies. The studies didn't report on quality of life or graft success. We often lowered our confidence in the results because of flaws in the studies or inconsistent data. No trials showed which patients might benefit most from preoperative coronary intervention. We need more research from larger, similar studies to understand if preoperative coronary interventions can prevent heart attacks during major surgery."*

Figure 1: Example of a participant's output versus input: ~~deletions~~, insertions, and whole sentence insertions

CLEF 2025 Task 1 submissions can be reused as ground truth data for information distortion classification in CLEF 2026 Task 2.

Based on analysis of the common types of overgeneration in the CLEF 2025 SimpleText submissions, we identified five broad categories of information distortion:

LEAKED INSTRUCTIONS Leaked task framing, assistant chatter, simplification rationale.

UNGROUNDED INJECTION Unsupported factual or interpretive additions.

GENERATION FAILURE Catastrophic or unusable output (degenerate, truncated, corrupted).

REPETITIVE CONTENT Repetition loops.

GROUNDED OVERGENERATION Source-supported content beyond simplification scope.

None No overgeneration detected.

Task 2.3 is to avoid creative generation and perform grounded generation by design. This mimics Task 1 above and asks for the submission of pairs of runs with/without source attribution by design.

2.2. Data

Over the last few years, while running the SimpleText track, we collected many realistic and representative predictions in the run submission. For Task 2.1 and Task 2.2, real 2025 track submissions are used,

with human and LLM annotations of sentences lacking source support following an extended version of [13]. For Task 2.3, we follow the setup of Task 1: Text Simplification above, but request paired runs.

Data We have in total 6,728 document- or abstract-level predictions from the CLEF 2025 SimpleText Task 1, mostly from the sentence-level submissions. In total, these contain 105,001 sentences in the provided sentence tokenization.

At the document level, there are 1,323 positive labels (19.7%) and 5,405 negative labels (80.3%).

At the sentence level, there are 99,798 sentences labeled as `None` (95.0%) and 5,203 labeled as `Overgeneration` (5.0%).

In terms of the classification labels, there are 1,476 cases of `UNGROUNDED_INJECTION`, 331 cases of `REPETITIVE_CONTENT`, 1,901 cases of `GENERATION_FAILURE`, 1,459 cases of `LEAKED_INSTRUCTIONS`, and 36 cases of `GROUNDED_OVERGENERATION`.

Test and Train splits The data was split into train, development, and test sets at the document or abstract level. The dev split contains 1,120 documents (16.6%), and the test split contains 2,851 documents (42.4%). The large test split is due to the construction, which includes both seen and unseen systems and abstracts.

The scale of the data defies human annotation, and we prioritized a machine-labeling regime focused on high accuracy. For a smaller sample of up to 10 abstracts per included submission (315 in total), we have detailed human labels. Based on this validation set, we estimate that positive labels are 92% accurate against human judgments, which is similar to the estimated interrater agreement.

The data still underestimate the overgeneration, or unfounded generation, problem, and the negative labels were not vetted at the sentence level.

Task The same data can be used for an overgeneration task that asks for a Boolean label (Task 2.1) and an overgeneration classification task that asks for a specific overgeneration type (Task 2.2).

We encourage detection without access to the sources, based solely on the predictions in Task 2.1(a) and Task 2.2(a). This is obviously much more challenging than having access to both the source and the prediction for a particular system in Task 2.1(b) and Task 2.2(b). Ultimately, most submissions prioritized the leaderboard and high scores, and we received few "no source" submissions.

Task 2.3 follows the setup of Task 1 on *Task 1: Text Simplification* in [4], but requested paired runs with and without the special processing to avoid ungrounded generation.

2.3. Formats

This section outlines the format of the data used in the CLEF 2026 SimpleText Task 2.

2.3.1. Source

The source data has the following fields in JSON array format:

- `id` Unique identifier for the example
- `source` The original biomedical abstract from Cochrane Library systematic reviews
- `sentences` The model-generated simplified version, pre-split into sentences
- `labels` Pre-filled with "None". Replace with your predictions

An example is:

```
[
  {
    "id": "test_00001",
    "source": "Full source abstract as one long string...",
    "sentences": [
      "Simplified sentence 1...",
      "Simplified sentence 2...",
      "Simplified sentence 3..."
    ],
    "labels": ["None", "None", "None"]
  }
]
```

2.3.2. Prediction

The prediction is a JSON array containing only `id` and `labels` for each example.

```
[
  {
    "id": "test_00001",
    "labels": ["None", "Overgeneration", "None"],
    "run_id":"TeamName_Task21a_MethodUsed"
  {
    "id": "test_00002",
    "labels": ["None", "None"],
    "run_id":"TeamName_Task21a_MethodUsed"
  }
]
```

Task 2.1: Binary Detection Replace **None** with **Overgeneration** for sentences you flag:

```
"labels": ["None", "Overgeneration", "None"]
```

Accepted binary labels:

- `None`: No overgeneration
- `false` (string): No overgeneration
- `false` (boolean): No overgeneration
- `Overgeneration`: Overgeneration detected
- `true` (string): Overgeneration detected
- `true` (boolean): Overgeneration detected

Task 2.1: Multi-Class Categorization Replace `None` with one of the following category labels:

```
"labels": ["None", "LEAKED_INSTRUCTIONS", "None"]
```

Accepted multi-class labels:

- `None`: No overgeneration detected
- `LEAKED_INSTRUCTIONS`: Leaked task framing, assistant chatter, simplification rationale
- `UNGROUNDED_INJECTION`: Unsupported factual or interpretive additions
- `GENERATION_FAILURE`: Catastrophic or unusable output (degenerate, truncated, corrupted)
- `REPETITIVE_CONTENT`: Repetition loops
- `GROUNDED_OVERGENERATION`: Source-supported content beyond simplification scope

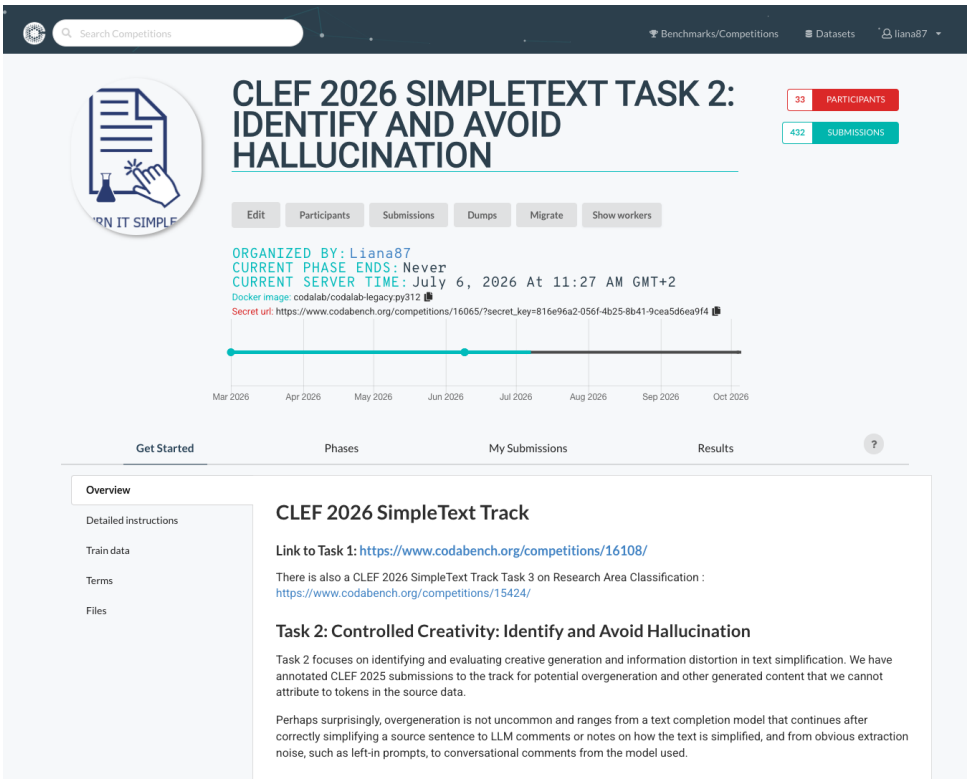

Figure 2: CLEF 2026 SimpleText Task 2 Codabench.

2.4. Evaluation

Task 2.1 is essentially a sentence-label task, with inferred document-level labels. This is evaluated using the standard metrics: Precision, Recall, and F1 at the sentence and document level.

Task 2.2 is a multi-label classification task. We evaluate performance using label accuracy over the 5 + None labels. In addition, we can calculate the Task 2.1 measures by conflating the overgeneration types. Task 2.3 is evaluated by both standard automatic measures and human evaluation, following Task 1 on Text Simplification in [4]. We also conduct more detailed overgeneration analysis for Task 2.3.

Submissions were made through Codabench.[1] Due to differences in setup, each task had a separate competition on Codabench. The Task 1 runs were submitted at: https://www.codabench.org/competitions/16108/. The Task 2 runs were submitted at: https://www.codabench.org/competitions/16065/ (shown in Figure 2). Codabench greatly facilitated running the track in 2026 and provided active participants (who had also registered at Codabench) with full access to the competition, including the submission and leaderboard pages.

Submissions were made through Codabench.[2] Due to differences in setup, each task had a separate competition on Codabench. The Task 1 runs were submitted at: https://www.codabench.org/competitions/16108/. The Task 2 runs were submitted at: https://www.codabench.org/competitions/16065/ (shown in Figure 2). Codabench greatly facilitated running the track in 2026 and provided active participants (who had also registered at Codabench) with full access to the competition, including the submission and leaderboard pages.

3. Participant's Approaches

A total of 16 teams submitted 376 runs in total. In the detailed results, we include only runs without errors that achieve a non-zero score.

[1] https://www.codabench.org/
[2] https://www.codabench.org/

A-R Adnan and Sonbol [14] submitted 5 runs in total for Task 2. They submitted 5 runs for Task 2.1, 0 runs for Task 2.2, and 0 runs for Task 2.3. Their approach deploys source grounding for an ensemble of ModernBERT-large, SciBERT, and SPECTER2, finding that grounding the whole document context is superior to grounding only the top-matching sentences.

AIIRLab Hawkins et al. [15] submitted 21 runs in total for Task 2. They submitted 8 runs for Task 2.1, 10 runs for Task 2.2, and 3 runs for Task 2.3. Their approach is based on a DeBERTa cross-encoder model, a fine-tuned LLaMA-3.1, and majority-voting ensembles.

CYUT Wu and Yan [16] submitted 84 runs in total for Task 2. They submitted 76 runs for Task 2.1, 8 runs for Task 2.2, and 0 runs for Task 2.3. Their approach is based on retrieval-aligned contextual fusion in a multi-tier cascaded discernment pipeline. For classification, a downstream classifier is trained.

DUTH Arampatzis and Arampatzis [17] submitted 78 runs in total for Task 2. They submitted 64 runs for Task 2.1, 14 runs for Task 2.2, and 0 runs for Task 2.3. Their approach is based on extensively fine-tuned models, in particular SciBERT, BiomedBERT, PubMedBERT, BioLinkBERT, and BlueBERT, either alone or in an ensemble with parameter tuning.

FOSU-YZH Yang et al. [18] submitted 7 runs in total for Task 2. They submitted 6 runs for Task 2.1, 1 run for Task 2.2, and 0 runs for Task 2.3. They deploy a range of DeBERTa-trained classifiers.

FutureMakers Liu et al. [19] submitted 33 runs in total for Task 2. They submitted 6 runs for Task 2.1, 27 runs for Task 2.2, and 0 runs for Task 2.3. They use a tf.idf source sentence filter combined with a PubMedBERT classifier.

SIM (no paper) submitted 30 runs in total for Task 2. They submitted 23 runs for Task 2.1, 5 runs for Task 2.2, and 2 runs for Task 2.3.

SVNIT (no paper) submitted 8 runs in total for Task 2. They submitted 8 runs for Task 2.1, 0 runs for Task 2.2, and 0 runs for Task 2.3.

THM Dauenhauer et al. [20] submitted 19 runs in total for Task 2. They submitted 9 runs for Task 2.1, 10 runs for Task 2.2, and 0 runs for Task 2.3. Their approach is based on a complex ensemble setup: a BERT ensemble classifier with four different BERT models, and an LLM as a Judge (GPT-4.1-nano) deciding on cases when the ensemble disagrees.

UPF Hayakawa and Saggion [21] submitted 12 runs in total for Task 2. They submitted 0 runs for Task 2.1, 12 runs for Task 2.2, and 0 runs for Task 2.3. They use heuristics, such as typical conversational LLM words uncommon in scientific text, and fine-tuned classifiers: Clinical Modern BERT, PubMedBERT-mnli, and Gemma-3-14b.

UvA Mathas et al. [22] submitted 6 runs in total for Task 2. They submitted 0 runs for Task 2.1, 4 runs for Task 2.2, and 2 runs for Task 2.3. Their approach focuses on "no source" submissions to Task 2.2, using prompted Qwen3 and GPT-4.1-mini. Their performance without source information is impressive, though at a lower level than submissions that use source information.

Writerslogic Condrey [23] submitted 72 runs in total for Task 2. They submitted 37 runs for Task 2.1, 34 runs for Task 2.2, and 1 run for Task 2.3. Their approach is a DeBERTa classifier with extensive feature engineering for detection, and two LightGBM ensembles for classification.

Table 1
Results for CLEF 2026 SimpleText Task 2.1 Detecting Overgeneration: Test data, posthoc detection without sources, best three runs per team

Team/Method	count	Document			Sentence		
		F1	Prec	Rec	F1	Prec	Rec
THM BertOnly_roberta	9	0.74	0.83	0.67	0.60	0.62	0.58
AIIRLab crossencoder	3	0.74	0.77	0.71	0.83	0.83	0.83
THM BertOnly_bert	9	0.74	0.79	0.69	0.59	0.60	0.58
THM BertOnly_scibert	9	0.73	0.83	0.65	0.60	0.63	0.57
SIM StdlibLogReg	1	0.58	0.54	0.63	0.69	0.65	0.73
AIIRLab Team_MultiAgent_Qwen3-	3	0.34	0.20	1.00	0.14	0.08	0.95
AIIRLab SourceFree_Qwen3-4B	3	0.34	0.20	0.99	0.15	0.08	0.93

4. Results

This section details the results for the overgeneration identification, overgeneration classification, and grounded text simplification subtasks.

4.1. Task 2.1: Identify Overgeneration

Task 2.1 aims to identify overly creative generation in scientific text simplification. This task focuses on detecting overgeneration and other information distortion issues in the predictions of current models. The task raises awareness of remaining information distortion issues in modern generative models for scientific text simplification, and focuses on post-hoc detection without or with access to the source text.

Table 1 shows the results of detecting spurious sentences in the generated simplifications of participants in the track in earlier years. The main task is post-hoc detection without access to the source texts, which would generalize to generic text generation tasks.

We make several observations. First, the number of submissions labeled "no source" is limited. Participants' papers have not included extensive experiments under the "no source" condition, as this is a much harder task that leads to lower scores and lower rankings on the shared leaderboard. Second, the performance of most of these few submissions is similar to scores observed under the easier conditions with source abstracts available. This suggests accidental labels in the metadata, and we have to be careful to draw major conclusions. Third, some of the confirmed submissions, such as AIIRLab's SourceFree run, confirm the challenging task but still achieve a reasonable F1 score, albeit at a very high recall level near an "always yes" baseline.

Table 2 also shows the results of detecting spurious sentences in the generated simplifications of participants in the track in earlier years, while having access to pairs of source-prediction content. This setting exploits the text simplification setting, in which information generation must faithfully reflect the source content.

We make several observations. First, access to the sources leads to impressive detection performance, with document-level F1 over 80% and sentence-level F1 over 85%. Second, top-scoring approaches tend to apply complex ensembles of trained models and further specific trained classifiers to aggregate the ensemble predictions. Third, and more generally, the finding that modern AI pipelines can effectively detect overgeneration in AI output is an important and encouraging research result.

4.2. Task 2.2: Detect and Classify Overgeneration

Task 2.2 asks not only to detect information distortion in the output of text simplification models but also to classify the type of error. This task mimics the human manual evaluation we performed in the track in earlier years.

Table 2
Results for CLEF 2026 SimpleText Task 2.1 Detecting Overgeneration: Test data, detection with sources, best three runs per team

Team/Method	count	Document			Sentence		
		F1	Prec	Rec	F1	Prec	Rec
AIIRLab majorityvote	7	0.82	0.81	0.82	0.86	0.83	0.90
writerslogic v13_deb_t40	37	0.81	0.81	0.81	0.86	0.85	0.88
writerslogic v13_deb_t35	37	0.81	0.80	0.81	0.86	0.84	0.88
writerslogic reblend	37	0.81	0.83	0.79	0.87	0.86	0.87
CYUT DeBERTa_XGB_QwenAudit_v1	76	0.81	0.80	0.82	0.86	0.84	0.89
CYUT DeBERTa_XGB_Balanced_v3_C	76	0.81	0.79	0.82	0.86	0.83	0.88
CYUT DeBERTa_XGB_3	76	0.81	0.80	0.82	0.86	0.83	0.88
DUTH TH0524_DocFlip_Add1_Rem0	64	0.80	0.81	0.79	0.86	0.84	0.88
DUTH BERTAvg_Sci50_Biomed50_TH	64	0.80	0.81	0.79	0.86	0.84	0.88
DUTH BERTAvg_Sci50_Biomed50_TH	64	0.80	0.81	0.79	0.86	0.84	0.88
A-R A_R Team	4	0.79	0.83	0.75	0.85	0.84	0.86
FutureMakers MyTeam_Task21_Pub	8	0.78	0.79	0.77	0.83	0.80	0.85
FutureMakers wenlongliu_TFIDF_	8	0.78	0.79	0.77	0.83	0.80	0.85
FutureMakers wenlongliu_TFIDF_	8	0.77	0.74	0.81	0.82	0.77	0.88
FOSU-YZH-Group DeBERTa075_RoBE	6	0.77	0.83	0.72	0.84	0.85	0.84
FOSU-YZH-Group DeBERTa_V3_Larg	6	0.77	0.76	0.78	0.83	0.80	0.87
FOSU-YZH-Group DeBERTa055_RoBE	6	0.77	0.79	0.75	0.84	0.82	0.86
AIIRLab crossencoder	7	0.75	0.76	0.74	0.84	0.84	0.84
THM BertOnly_roberta	9	0.74	0.83	0.67	0.60	0.62	0.58
AIIRLab crossencoder	7	0.74	0.77	0.71	0.83	0.83	0.83
THM BertOnly_bert	9	0.74	0.79	0.69	0.59	0.60	0.58
THM BertOnly_scibert	9	0.73	0.83	0.65	0.60	0.63	0.57
A-R A_R_Team	4	0.72	0.59	0.95	0.77	0.64	0.96
A-R A_R	4	0.69	0.58	0.84	0.73	0.63	0.88
SIM PerLabelBinaryUnion	22	0.60	0.62	0.58	0.72	0.72	0.72
SIM PerLabelBinaryUnion	22	0.60	0.62	0.58	0.72	0.72	0.72
SIM REVIEWED2_MetaDocPrecision	22	0.59	0.59	0.60	0.19	0.42	0.13

Table 3
Results for CLEF 2026 SimpleText Task 2.2 Classifying Overgeneration: Test data, posthoc classification without sources, three best runs per team

Team/Method	count	Acc.	Document			Sentence		
			F1	Prec	Rec	F1	Prec	Rec
AIIRLab crossencoder	1	0.77	0.74	0.77	0.71	0.83	0.83	0.83
THM BertOnly_distilbert	9	0.74	0.72	0.73	0.70	0.81	0.82	0.81
THM BertOnly_RF	9	0.56	0.68	0.82	0.58	0.66	0.70	0.63
THM BertOnly_RF	9	0.56	0.68	0.82	0.58	0.66	0.70	0.63
UvA GPT41Mini_PrecisionV2	3	0.45	0.43	0.32	0.66	0.41	0.28	0.74
UvA GPT41Mini	3	0.42	0.43	0.32	0.64	0.40	0.27	0.74
UvA Qwen3_14B_Colab	3	0.35	0.41	0.27	0.80	0.34	0.21	0.76

Results are presented in Table 3. This is a very challenging task of classifying overgeneration without access to the sources, based solely on the output of an AI model.

We make several observations. First, again, the number of submissions labeled "no source" is limited, as this is a much harder task that results in lower scores and rankings on the shared leaderboard. Second, given the very few submissions, there may be accidental labels in the metadata, and we have to be careful to draw major conclusions. Third, some of the confirmed submissions, such as UvA's runs, demonstrate the challenging task while still achieving a reasonable F1 score. Interestingly, the main

Table 4
Results for CLEF 2026 SimpleText Task 2.2 Classifying Overgeneration: Test data, detection with sources, three best runs per team

Team/Method	count	Acc.	Document			Sentence		
			F1	Prec	Rec	F1	Prec	Rec
AIIRLab NLI_CrossEncoder	10	0.85	0.75	0.63	0.92	0.79	0.68	0.94
AIIRLab LargeFullsrcDiffLR	10	0.83	0.75	0.66	0.86	0.81	0.73	0.91
AIIRLab majorityvote	10	0.83	0.82	0.81	0.82	0.86	0.83	0.90
writerslogic v28_union_70	34	0.81	0.78	0.72	0.85	0.85	0.81	0.89
writerslogic v28_union_80	34	0.81	0.79	0.74	0.85	0.85	0.82	0.89
writerslogic v28_union_90	34	0.81	0.79	0.76	0.84	0.86	0.83	0.89
CYUT CrossEncoder_Weighted_RAG	8	0.80	0.79	0.74	0.84	0.84	0.79	0.91
FutureMakers MyTeam_PubMedBERT	33	0.79	0.47	0.31	0.92	0.42	0.28	0.91
FutureMakers wenlongliu_TFIDF_	33	0.79	0.80	0.81	0.79	0.87	0.86	0.87
DUTH SciBERTMulti_SourceAware_	14	0.79	0.79	0.82	0.76	0.85	0.85	0.85
FutureMakers wenlongliu_TFIDF_	33	0.79	0.76	0.74	0.78	0.82	0.78	0.86
CYUT MiniLM_MultiClass_v1	8	0.79	0.67	0.56	0.84	0.75	0.65	0.89
DUTH Gated_BERTAvg05275_SciMul	14	0.79	0.80	0.81	0.79	0.86	0.84	0.88
CYUT CrossEncoder_Weighted_RAG	8	0.79	0.81	0.80	0.82	0.86	0.84	0.89
DUTH SciBERTMulti_SourceAware_	14	0.78	0.78	0.82	0.75	0.85	0.86	0.84
UPF Ensemble	12	0.78	0.47	0.32	0.92	0.32	0.19	0.92
UPF Ablation_3000_4	12	0.78	0.47	0.32	0.92	0.32	0.19	0.92
UPF Ensemblev5	12	0.76	0.49	0.34	0.88	0.40	0.26	0.90
THM BertOnly_distilbert	10	0.74	0.72	0.73	0.70	0.81	0.82	0.81
THM BertOnly_RF	10	0.72	0.72	0.73	0.70	0.81	0.81	0.81
SIM GeneralizationBalanced	5	0.64	0.58	0.55	0.61	0.70	0.67	0.72
SIM qwen_emb_rag_grounded_phra	5	0.61	0.54	0.46	0.66	0.64	0.59	0.70
SIM qwen_emb_rag_grounded_phra	5	0.61	0.54	0.46	0.66	0.64	0.59	0.70
THM BertOnly_RF	10	0.56	0.68	0.82	0.58	0.66	0.70	0.63
UvA GPT41Mini_GroundedV1	4	0.50	0.41	0.28	0.77	0.31	0.19	0.79
UvA GPT41Mini_PrecisionV2	4	0.45	0.43	0.32	0.66	0.41	0.28	0.74
FOSU-YZH-Group DocAgent_Detect	1	0.44	0.40	0.27	0.84	0.29	0.17	0.84
UvA GPT41Mini	4	0.42	0.43	0.32	0.64	0.40	0.27	0.74
UBO test_submission	1	0.03	0.34	0.20	0.91	0.07	0.04	0.15

complexity seems in the detection part of the task, which is greatly facilitated by access to the source data. The classification performance proper seems a viable option, at least for some of the types of overgeneration.

Results are presented in Table 4.

We make the following observations. First, we observe a very high classification accuracy of over 80% for the detailed detection and classification of overgeneration. This is an encouraging result, as participants were not only able to detect overgeneration but also to classify its specific type. Second, top-performing teams constructed a complex ensemble approach, often combining a detection stage with a secondary classification stage. In particular, larger ensembles of trained and fine-tuned models perform well for this task. Third, more generally, because the data used are representative benchmark submissions, these findings suggest a promising research direction for generative AI benchmarking tasks. By complementing traditional benchmarks with a separate task to identify remaining issues in model output, we may be able to overcome the current evaluation crisis in which many of our models have "outsmarted" traditional benchmark evaluation setups.

4.3. Task 2.3: Avoid Creative Generation

Task 2.3 aims to avoid overly creative generation in scientific text simplification and showcase systems that perform grounded generation by design. This task requires a pair of submissions, one of which

Table 5
Results for CLEF 2026 SimpleText Task 2.3: Avoiding creative generation by design

Team/Method	count	SARI	BLEU	FKGL	Compression ratio	Sentence splits	Levenshtein similarity	Exact copies	Additions proportion	Deletions proportion	Lexical complexity score
AIIRLab llama	2	44.19	11.53	12.29	0.93	1.25	0.57	0.00	0.34	0.48	8.60
AIIRLab concat	2	44.19	11.53	12.29	0.93	1.25	0.57	0.00	0.34	0.48	8.60
SIM Task23_Copy	2	9.12	14.41	13.81	1.00	1.00	1.00	1.00	0.00	0.00	8.78
SIM Task23_Copy_Grou	2	9.12	14.41	13.81	1.00	1.00	1.00	1.00	0.00	0.00	8.78
AIIRLab llama	2	43.75	10.95	12.45	0.98	1.23	0.57	0.00	0.36	0.46	8.61
AIIRLab concat	2	43.75	10.95	12.45	0.98	1.23	0.57	0.00	0.36	0.46	8.61
SIM Task23_Copy	2	8.88	12.40	13.02	1.00	1.00	1.00	1.00	0.00	0.00	8.85
SIM Task23_Copy_Grou	2	8.88	12.40	13.02	1.00	1.00	1.00	1.00	0.00	0.00	8.85

must make a special effort to avoid overgeneration or other information-distortion issues.

Table 5 shows the standard evaluation of text simplification output against text overlap with the reference plain language summaries. We evaluate against the Cochrane-auto-aligned cases (top) and the larger set of original plain-language summaries (bottom).

We make several observations. First, only two teams submitted runs for Task 2.3, indicated by their run names. Neither team indicated clear pairs of submissions to compare. Second, while participation was disappointing, the task encouraged teams to critically examine their output for overgeneration. Several teams mention abandoning models that were difficult to control, or opting for larger or newer models that were better at following specific instructions in the prompt. Third, while this task did not add significantly to Task 1 above, the combined setup of Task 1 and Task 2 helps raise awareness and avoid overgeneration in Task 1. So, the aims of Task 2.3 and the track as a whole were effectively realized.

5. Analysis

5.1. Task 2.1 Overgeneration Detection

We provide additional analysis of Task 2.1 submissions, showing sentence-level recall for detecting overgeneration, broken down across the five different types.

Results are presented in Table 6. We make several observations. First, for the `leaked instructions`, `generation failure`, and `repetitive content` types, we observe very high recall. This suggests that the better models are highly effective at detecting such overgeneration through careful comparison with the source. Second, `ungrounded injection` is detected moderately, with up to 74% recall. This suggests that overgeneration detection models also provide reasonable indications of this type of overgeneration. Third, `grounded overgeneration` cases are poorly detected, and remain a challenge to accurately detect, despite the high overall detection quality.

5.2. Task 2.2 Overgeneration Classification

We provide additional analysis of Task 2.2 submissions, showing sentence-level accuracy and recall for classifying overgeneration, broken down across the five different types.

Results are presented in Table 7. We make the following observations. First, we see a similar pattern for overgeneration classification as for overgeneration detection. The `leaked instructions` and

Table 6
Analysis for CLEF 2026 SimpleText Task 2.1 Detecting Overgeneration: Test data, detection with sources, best three runs per team

Team/Method	count	F1		Recall				
		Doc	Snt	Leak. in.	Ungr. inj.	Gen. fail.	Repet.	Gr. over.
AIIRLab majorityvote	7	0.82	0.86	0.99	0.74	0.98	0.94	0.39
writerslogic v13_deb_t40	37	0.81	0.86	0.96	0.73	0.96	0.92	0.58
writerslogic v13_deb_t35	37	0.81	0.86	0.96	0.73	0.96	0.92	0.58
writerslogic reblend	37	0.81	0.87	0.96	0.69	0.96	0.92	0.52
CYUT DeBERTa_XGB_QwenAudit_v1	76	0.81	0.86	0.96	0.74	0.97	0.92	0.27
CYUT DeBERTa_XGB_Balanced_v3_C	76	0.81	0.86	0.96	0.74	0.97	0.92	0.27
CYUT DeBERTa_XGB_3	76	0.81	0.86	0.96	0.74	0.97	0.92	0.27
DUTH TH0524_DocFlip_Add1_Rem0	64	0.80	0.86	0.98	0.69	0.97	0.94	0.27
DUTH BERTAvg_Sci50_Biomed50_TH	64	0.80	0.86	0.98	0.69	0.97	0.94	0.27
DUTH BERTAvg_Sci50_Biomed50_TH	64	0.80	0.86	0.98	0.69	0.97	0.94	0.27
A-R A_R Team	4	0.79	0.85	0.97	0.66	0.97	0.90	0.12
FutureMakers MyTeam_PubMedBERT	8	0.78	0.83	0.95	0.67	0.97	0.84	0.33
FutureMakers wenlongliu_TFIDF_	8	0.78	0.83	0.95	0.67	0.97	0.84	0.33
FutureMakers wenlongliu_TFIDF_	8	0.77	0.82	0.96	0.71	0.98	0.90	0.48
FOSU-YZH-Group DeBERTa075_RoBE	6	0.77	0.84	0.96	0.62	0.97	0.87	0.12
FOSU-YZH-Group DeBERTa_V3_Larg	6	0.77	0.83	0.97	0.68	0.98	0.94	0.18
FOSU-YZH-Group DeBERTa055_RoBE	6	0.77	0.84	0.97	0.65	0.97	0.91	0.15
AIIRLab crossencoder	7	0.75	0.84	0.94	0.64	0.94	0.90	0.33
THM BertOnly_roberta	9	0.74	0.60	0.68	0.26	0.83	0.50	0.18
AIIRLab crossencoder	7	0.74	0.83	0.94	0.63	0.96	0.83	0.15
THM BertOnly_bert	9	0.74	0.59	0.68	0.26	0.82	0.51	0.18
THM BertOnly_scibert	9	0.73	0.60	0.67	0.25	0.83	0.50	0.15
A-R A_R_Team	4	0.72	0.77	1.00	0.89	0.99	0.99	0.64
A-R A_R	4	0.69	0.73	0.98	0.71	0.98	0.81	0.33
SIM PerLabelBinaryUnion	22	0.60	0.72	0.86	0.38	0.97	0.57	0.00
SIM PerLabelBinaryUnion	22	0.60	0.72	0.86	0.38	0.97	0.57	0.00
SIM REVIEWED2_MetaDocPrecision	22	0.59	0.19	0.16	0.12	0.10	0.17	0.00

`generation failure` types of overgeneration exhibit very high performance in both accuracy and recall. Second, the `repetitive content` type of overgeneration shows high recall but moderate accuracy, indicating that most models overestimate this category. Third, for `ungrounded injection` overgeneration, we see moderately high recall and accuracy, indicating that this type is harder to classify than the others. Only the `grounded overgeneration` type exhibits very poor performance, with the lowest recall and accuracy.

5.3. Task 2.3 Avoiding Overgeneration

We analyzed the entire test dataset, comprising 8,069 documents (Task 1.2) and 48,809 sentences (Task 1.1). This analysis assumes that there is always word overlap between a pair of complex-simple sentences or abstracts. Moreover, we specifically look for overgenerating output at the end of a sentence or an abstract. Such sentences are unfounded by the source, and they are not easy to spot in the generated text, as they are often coherent and possible continuations. This may occur after every sentence in sentence-level text simplification, and in document-level text simplification, this is more likely at the end of the abstract, so we still look at the end of the source input.

We indeed observe overgeneration/text-completion issues at the end of the sources/predictions. There are also cases in which systematic errors occur during output extraction, with additional content. Increasingly, there is additional LLM commentary other than the requested output. Accurately removing such additional content can be more challenging for the document-level submissions than for the

Table 7
Analysis of CLEF 2026 SimpleText Task 2.2 Classifying Overgeneration: Test data, detection with sources, three best runs per team

Team/Method	count	Acc.	Leak. in.		Ungr. inj.		Gen. fail.		Repet.		Gr. over.	
			Acc	Rec	Acc	Rec	Acc	Rec	Acc	Rec	Acc	Rec
AIIRLab NLI_CrossEncoder	10	0.85	0.91	0.99	0.80	0.85	0.92	0.99	0.60	0.94	0.00	0.39
AIIRLab LargeFullsrcDiffLR	10	0.83	0.94	0.99	0.70	0.77	0.91	0.98	0.69	0.95	0.21	0.61
AIIRLab majorityvote	10	0.83	0.94	0.99	0.70	0.74	0.92	0.98	0.60	0.94	0.09	0.39
writerslogic v28_union_70	34	0.81	0.91	0.96	0.70	0.75	0.88	0.97	0.70	0.92	0.00	0.61
writerslogic v28_union_80	34	0.81	0.91	0.96	0.70	0.75	0.88	0.97	0.70	0.92	0.00	0.61
writerslogic v28_union_90	34	0.81	0.91	0.96	0.70	0.74	0.88	0.96	0.70	0.92	0.00	0.61
CYUT CrossEncoder_Weighted_RAG	8	0.80	0.92	0.98	0.67	0.77	0.89	0.98	0.64	0.93	0.00	0.30
FutureMakers MyTeam_PubMedBERT	33	0.79	0.88	0.97	0.69	0.78	0.87	0.98	0.64	0.89	0.00	0.58
FutureMakers wenlongliu_TFIDF_	33	0.79	0.88	0.94	0.64	0.71	0.89	0.95	0.70	0.92	0.00	0.52
DUTH SciBERTMulti_SourceAware_	14	0.79	0.92	0.95	0.64	0.69	0.88	0.94	0.63	0.84	0.00	0.24
FutureMakers wenlongliu_TFIDF_	33	0.79	0.90	0.96	0.61	0.67	0.92	0.98	0.63	0.87	0.03	0.33
CYUT MiniLM_MultiClass_v1	8	0.79	0.95	0.98	0.60	0.72	0.90	0.98	0.54	0.91	0.00	0.39
DUTH Gated_BERTAvg05275_SciMul	14	0.79	0.91	0.98	0.64	0.69	0.88	0.97	0.63	0.94	0.00	0.27
CYUT CrossEncoder_Weighted_RAG	8	0.79	0.90	0.96	0.64	0.74	0.89	0.97	0.63	0.92	0.00	0.27
DUTH SciBERTMulti_SourceAware_	14	0.78	0.91	0.94	0.63	0.68	0.87	0.93	0.61	0.80	0.00	0.24
UPF Ensemble	12	0.78	0.85	0.98	0.67	0.82	0.88	0.97	0.58	0.91	0.09	0.79
UPF Ablation_3000_4	12	0.78	0.85	0.98	0.67	0.82	0.88	0.97	0.58	0.91	0.09	0.79
UPF Ensemblev5	12	0.76	0.85	0.97	0.62	0.80	0.88	0.96	0.56	0.85	0.21	0.82
THM BertOnly_distilbert	10	0.74	0.89	0.93	0.52	0.59	0.88	0.95	0.56	0.79	0.00	0.15
THM BertOnly_RF	10	0.72	0.88	0.93	0.49	0.59	0.88	0.95	0.45	0.81	0.00	0.15
SIM GeneralizationBalanced	5	0.64	0.78	0.84	0.34	0.42	0.86	0.94	0.41	0.67	0.00	0.00
SIM qwen_emb_rag_grounded_phra	5	0.61	0.72	0.80	0.33	0.42	0.84	0.91	0.40	0.68	0.00	0.03
SIM qwen_emb_rag_grounded_phra	5	0.61	0.72	0.80	0.33	0.42	0.84	0.91	0.40	0.68	0.00	0.03
THM BertOnly_RF	10	0.56	0.67	0.70	0.28	0.35	0.78	0.83	0.40	0.71	0.00	0.18
UvA GPT41Mini_GroundedV1	4	0.50	0.66	0.91	0.46	0.56	0.49	0.94	0.18	0.79	0.00	0.06
UvA GPT41Mini_PrecisionV2	4	0.45	0.81	0.91	0.21	0.40	0.39	0.96	0.41	0.73	0.00	0.06
FOSU-YZH-Group DocAgent_Detect	1	0.44	0.48	0.83	0.53	0.68	0.34	0.99	0.35	0.91	0.00	0.03
UvA GPT41Mini	4	0.42	0.75	0.89	0.17	0.38	0.38	0.96	0.49	0.78	0.00	0.09
UBO test_submission	1	0.03	0.04	0.17	0.03	0.14	0.02	0.14	0.06	0.15	0.09	0.15

Table 8
Analysis of SimpleText Task 2.3: Spurious generation at the sentence (top) and document (bottom) level

Run	SARI	Source	Spurious Content	
	(175)	Number	Number	Fraction
AIIRLab llama	43.75	48,809	8,887	0.18
AIIRLab llama_concat	43.75	8,069	1,004	0.12
SIM Task23_Copy	8.88	8,069	0	0.00
SIM Task23_Copy_Grounded	8.88	8,069	0	0.00

sentence-level submissions, as some abstracts are very long.

Table 8 shows an overgeneration analysis of the Task 2.3 runs. This is done by aligning the source input with the prediction output based on their token sequences. If all the source sentence(s) have been aligned to some prediction sentence(s), we assume the prediction covers all the content of the sources. If there is still an additional sentence in the prediction, we regard this as spurious content for that specific input. This is an imperfect proxy, and aligning lengthy documents can be non-trivial. It serves as a good indicator of spurious content in predictions and overgeneration issues in runs.

We make several observations. First, the other runs performed lukewarm in the text simplification

evaluation of Section 4 yet shine here with a complete lack of overgeneration. Closer inspection reveals that these runs simply copy the sources as a prediction. Second, despite competitive performance in terms of text overlap with the references, we observe a significant number and fraction of overgeneration for the Llama submissions [15].[3] Third, this difference in additional content is not at all reflected in the evaluation scores, as some of the top-performing runs still exhibit larger fractions of "extra" content.

6. Discussion and Conclusions

This concludes the results for the CLEF 2026 SimpleText Task 2: Controlled Creativity on identify and avoid hallucination. Our main findings are the following. First, for Task 2.1, we observed high performance for document-level overgeneration detection with access to the sources. Similarly, for Task 2.2, we observed high performance for document-level overgeneration classification with access to the sources. These encouraging results suggest that almost any generative AI output can be improved by training specific models to detect and classify remaining issues in text generation tasks. Second, the track's setup exploits the text simplification setup, with sources, predictions, and references in the same language. The results were obtained in a reference-free setting, allowing them to generalize to almost any scenario where text generation models are used. Third, the overgeneration detection and classification data show how prevalent overgeneration is in traditional benchmarks and which types of overgeneration are most common. This is a sobering realization of the scope and scale of the remaining issues, despite relatively high performance as measured by textual overlap or LLM-as-a-judge evaluation measures.

This Task Overview focuses on the CLEF 2026 SimpleText Track's Task 2 on detecting and classifying overgeneration. Within the CLEF 2026 SimpleText Task 2, we have constructed extensive corpora and new references for evaluation data. These reusable corpora and evaluation resources are available to participants and other researchers who want to work on the important problem of making scientific information open and easily accessible for everyone. In terms of building a community researching scientific text summarization and simplification, the track saw a record attendance in 2026, with significant changes in the tasks and the move to Codabench, more runs were submitted, and with the largest number of participating teams ever.

Acknowledgments

We are incredibly thankful to the master's students in translation and technical writing from the University of Brest for participating in data annotation. We also thank each of the individual track participants for their effort in submitting a record number of submissions to CodaBench and documenting these in their papers.

We thank the CLEF 2026 chairs for hosting us, and the CLEF 2026 Labs and Proceedings chairs for their excellent assistance and flexibility. It is heartwarming to be part of such a great CLEF family. We thank CodaBench [24] for hosting the competition. Post-competition experiments are ongoing at https://www.codabench.org/competitions/16108/ (Task 1.1, Task 1.2, and Task 2.3) and https://www.codabench.org/competitions/16065/ (Task 2.1 and Task 2.2). We hope and expect that these "living test collections" remain in active use until the next iteration of the track.

Liana Ermakova is partly funded by the French National Research Agency (ANR-22-CE23-0019-01, *SimpleText: Automatic Simplification of Scientific Texts*). Liana Ermakova is further supported by the CNRS research group MaDICS (https://www.madics.fr/ateliers/simpletext/).

Jan Bakker and Jaap Kamps are partly funded by the Netherlands Organization for Scientific Research (NWO NWA # 1518.22.105). Jaap Kamps is further supported by the University of Amsterdam

[3]The document-level submission is a concatenation of the sentence-level text simplifications, providing exactly the same content as a prediction. This is an interesting test case for overgeneration quantification, as we only consider cases at the end of the prediction, and it confirms that the document-level detection is underestimating the overgeneration but remains a reliable indicator.

(AI4FinTech program) and ICAI (AI for Open Government Lab). Views expressed in this paper are not necessarily shared or endorsed by those funding the research.

The authors have no competing interests to declare that are relevant to the content of this article.

Declaration on Generative AI

During the preparation of this work, the authors used *ChatGPT* and *Grammarly* in order to: **Grammar and spelling check** and **Paraphrase and reword**. After using these tools/services, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

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

[2] J. Bakker, J. Kamps, Section-level simplification of biomedical abstracts, in: C. Christodoulopoulos, T. Chakraborty, C. Rose, V. Peng (Eds.), Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Suzhou, China, 2025, pp. 13819–13833. URL: https://aclanthology.org/2025.emnlp-main.697/. doi:10.18653/v1/2025.emnlp-main.697.

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

[4] J. Bakker, L. Ermakova, J. Kamps, Overview of the CLEF 2026 SimpleText Task 1: Simplify Scientific Text, in: [25], 2026.

[5] G. K. Shahi, L. Ermakova, J. Kamps, Overview of the CLEF 2026 SimpleText Task 3: Research Area Classification, in: [25], 2026.

[6] L. Ermakova, H. Azarbonyad, J. Bakker, B. Vendeville, J. Kamps, Overview of the CLEF 2025 simpletext track - simplify scientific text (and nothing more), in: J. Carrillo-de-Albornoz, A. G. S. de Herrera, J. Gonzalo, L. Plaza, J. Mothe, F. Piroi, P. Rosso, D. Spina, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction - 16th International Conference of the CLEF Association, CLEF 2025, Madrid, Spain, September 9-12, 2025, Proceedings, Lecture Notes in Computer Science, Springer, 2025, pp. 436–463. URL: https://doi.org/10.1007/978-3-032-04354-2_23. doi:10.1007/978-3-032-04354-2_23.

[7] B. Vendeville, J. Bakker, H. Azarbonyad, L. Ermakova, J. Kamps, Overview of the CLEF 2025 simpletext task 2: Identify and avoid hallucination, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum, CLEF 2025, Madrid, Spain, 9-12 September 2025, CEUR Workshop Proceedings, CEUR-WS.org, 2025, pp. 4186–4204. URL: https://ceur-ws.org/Vol-4038/paper_345.pdf.

[8] L. Ermakova, I. Ovchinnikova, J. Kamps, D. Nurbakova, S. Araújo, R. Hannachi, Overview of the CLEF 2022 simpletext task 3: Query biased simplification of scientific texts, in: G. Faggioli, N. Ferro, A. Hanbury, M. Potthast (Eds.), Proceedings of the Working Notes of CLEF 2022 - Conference and Labs of the Evaluation Forum, Bologna, Italy, September 5th - to - 8th, 2022, CEUR Workshop Proceedings, CEUR-WS.org, 2022, pp. 2792–2804. URL: https://ceur-ws.org/Vol-3180/paper-237.pdf.

[9] L. Ermakova, S. Bertin, H. McCombie, J. Kamps, Overview of the CLEF 2023 simpletext task 3: Simplification of scientific texts, in: M. Aliannejadi, G. Faggioli, N. Ferro, M. Vlachos (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2023), Thessaloniki, Greece, September 18th to 21st, 2023, CEUR Workshop Proceedings, CEUR-WS.org, 2023, pp. 2855–2875. URL: https://ceur-ws.org/Vol-3497/paper-240.pdf.

[10] L. Ermakova, V. Laimé, H. McCombie, J. Kamps, Overview of the CLEF 2024 simpletext task 3: Simplify scientific text, in: G. Faggioli, N. Ferro, P. Galuscáková, A. G. S. de Herrera (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), Grenoble, France, 9-12 September, 2024, CEUR Workshop Proceedings, CEUR-WS.org, 2024, pp. 3147–3162. URL: https://ceur-ws.org/Vol-3740/paper-307.pdf.

[11] A. Devaraj, W. Sheffield, B. Wallace, J. J. Li, Evaluating factuality in text simplification, in: S. Muresan, P. Nakov, A. Villavicencio (Eds.), Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 7331–7345. URL: https://aclanthology.org/2022.acl-long.506/. doi:10.18653/v1/2022.acl-long.506.

[12] B. Vendeville, L. Ermakova, P. D. Loor, Resource for error analysis in text simplification: New taxonomy and test collection, in: N. Ferro, M. Maistro, G. Pasi, O. Alonso, A. Trotman, S. Verberne (Eds.), Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2025, Padua, Italy, July 13-18, 2025, ACM, 2025, pp. 3723–3732. URL: https://doi.org/10.1145/3726302.3730304. doi:10.1145/3726302.3730304.

[13] B. Chakar, L. Ermakova, J. Kamps, A benchmark for overgeneration detection in biomedical text simplification, in: G. M. Di Nunzio, F. Vezzani, L. Ermakova, H. Azarbonyad, J. Kamps (Eds.), Proceedings of the 2nd Workshop on DeTermIt! Evaluating Text Difficulty in a Multilingual Context @ LREC 2026, ELRA and ICCL, Palma de Mallorca, Spain, 2026, pp. 12–21.

[14] M. Adnan, R. Sonbol, A_R Team at CLEF2026 SimpleText Task 2.1: Source-Aware Grounding Verification with Scientific Transformer Ensembles, in: [25], 2026.

[15] E. Hawkins, K. Zbinden, E. Fitzgerald, B. Mansouri, AIIRLab Systems at SimpleText CLEF 2026: Ensemble and Multi-Agent Pipelines for Scientific Text Simplification, in: [25], 2026.

[16] S.-H. Wu, C.-Z. Yan, CYUT at CLEF 2026 SimpleText Track: Hallucination Detection based on Retrieval-Aligned Contextual Fusion and Multi-Tier Cascaded Discernment, in: [25], 2026.

[17] G. Arampatzis, A. Arampatzis, DUTH at CLEF 2026 SimpleText: Grounded Biomedical Text Simplification and Evidence-Guided Overgeneration Detection, in: [25], 2026.

[18] Z. Yang, H. Qi, Y. Yi, FOSU-YZH-Group at the SimpleText 2026 Track: Statistics-Aware Prompting and Candidate Selection for Task 1.1 and Exploratory Overgeneration Detection for Task 2, in: [25], 2026.

[19] W. Liu, S. Qi, Y. Yi, R. Lin, FutureMakers at the CLEF 2026 SimpleText Track: Retrieval-Augmented Evidence-Focused Overgeneration Detection for Task 2.2, in: [25], 2026.

[20] J. Dauenhauer, N. Hofmann, C. K. Kreutz, THM@SimpleText 2026 - Task 2: Ensemble-based Hallucination Classification in Text Simplification, in: [25], 2026.

[21] A. Hayakawa, H. Saggion, UPF-TALN at the CLEF 2026 SimpleText Track: Ensemble Approach for Scientific Text Simplification and Hallucination Classification, in: [25], 2026.

[22] P. Mathas, B. Chakar, D. Carranza Navarrete, J. Bakker, J. Kamps, University of Amsterdam at the CLEF 2026 SimpleText Track, in: [25], 2026.

[23] D. Condrey, Writerslogic at the CLEF 2026 SimpleText Track: Multi-Candidate LLM Simplification and Stacked Complexity Spotting, in: [25], 2026.

[24] Z. Xu, S. Escalera, A. Pavão, M. Richard, W. Tu, Q. Yao, H. Zhao, I. Guyon, Codabench: Flexible, easy-to-use, and reproducible meta-benchmark platform, Patterns 3 (2022) 100543. URL: https://doi.org/10.1016/j.patter.2022.100543. doi:10.1016/J.PATTER.2022.100543.

[25] 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.