# Overview of the CLEF 2025 SimpleText Task 2: Identify and Avoid Hallucination

Benjamin Vendeville[1,2], Jan Bakker[3], Hosein Azarbonyad[4], Liana Ermakova[1] and Jaap Kamps[3]

[1]Université de Bretagne Occidentale, HCTI, Brest, France
[2]Lab-STICC (UMR CNRS 6285), Brest, France
[3]University of Amsterdam, Amsterdam, The Netherlands
[4]Elsevier, The Netherlands

## Abstract

This paper presents an overview of the CLEF 2025 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 used aligned sources, predictions, and references in text simplification to detect and quantify hallucinations—spurious content introduced by generative models—highlighting a critical limitation of current evaluation metrics. Second, we found that overgeneration and information distortion in model outputs can be detected with high accuracy, even without access to the original source text, suggesting that automatic detection is a promising strategy. Third, while automatic methods show promise, the detailed classification of distortions remains difficult to replicate without human expertise, underscoring the continued importance of expert human evaluation and the research challenge of building effective classification models to match this. More generally, we hope and expect that the constructed corpora and evaluation data will be used by researchers to further advance information distortion detection and classification approaches, both in general and specifically for scientific text simplification models.

## Keywords

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

## 1. Introduction

Becoming science-literate is more important than ever before. Objective scientific information helps any user navigate a world where misinformation, disinformation, or generated and unfounded information is only a single mouse click away. Everyone acknowledges the importance of objective scientific information, but the general public seldom consults scientific sources. The value of objective scientific information cannot be overstated. Biomedical research can directly impact people's decisions about health. However, the most reliable and up-to-date sources in biomedicine contain complex language and assume a high degree of background knowledge, making them difficult for the general public to understand.

To address these challenges, the CLEF 2025 Simple Track has three aims. First, we push the research frontier in text simplification by further expanding the scientific text simplification corpora, focusing on true paragraph-level and document-level simplification with greater variation, and considering the complex discourse structure. This setup fits current models, such as LLMs, that operate on a long input. This new biomedical corpus is constructed from aligned Cochrane abstracts and plain language summaries [1]. Second, we exploit the text simplification setup with aligned sources, references, and the output of generative models to detect, quantify, and avoid spurious information introduced gratuitously by the generative model. This is what is informally referred to as "hallucinations," addressing the remaining limitations of large generative models is crucial for the scientific use case, as current

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

benjamin.vendeville@univ-brest.fr (B. Vendeville); j.bakker@uva.nl (J. Bakker); h.azarbonyad@elsevier.com (H. Azarbonyad); liana.ermakova@univ-brest.fr (L. Ermakova); kamps@uva.nl (J. Kamps)
0009-0003-5298-147X (B. Vendeville); 0009-0002-9085-8491 (J. Bakker); 0000-0003-0841-7516 (H. Azarbonyad); 0000-0002-7598-7474 (L. Ermakova); 0000-0002-6614-0087 (J. Kamps)

**Table 1**
CLEF 2025 Simpletext official run submission statistics

| Team | Task 1 | | Task 2 | | | Task 1 | Task 2 | Total runs |
|---|---|---|---|---|---|---|---|---|
| | **1.1** | **1.2** | **2.1** | **2.2** | **2.3** | | | |
| AIIRLab [4] | 4 | 2 | 5 | 5 | 0 | 6 | 10 | 16 |
| ASM [5] | 0 | 10 | 0 | 0 | 0 | 10 | 0 | 10 |
| DSGT [6, 7] | 2 | 1 | 6 | 6 | 3 | 3 | 15 | 18 |
| DUTH [8] | 3 | 0 | 2 | 2 | 0 | 3 | 4 | 7 |
| EngKh (no paper) | 2 | 0 | 0 | 0 | 0 | 2 | 0 | 2 |
| Fujitsu [9] | 19 | 0 | 0 | 0 | 0 | 19 | 0 | 19 |
| LIA [10] | 0 | 9 | 0 | 0 | 0 | 9 | 0 | 9 |
| Mtest (no paper) | 1 | 1 | 1 | 1 | 0 | 2 | 2 | 4 |
| PICT [11] | 1 | 1 | 0 | 0 | 0 | 2 | 0 | 2 |
| RECAIDS [12] | 1 | 1 | 1 | 1 | 0 | 2 | 2 | 4 |
| Scalar [13] | 10 | 1 | 0 | 0 | 1 | 11 | 1 | 12 |
| SINAI [14] | 2 | 2 | 15 | 15 | 0 | 4 | 30 | 34 |
| THM [15] | 22 | 0 | 0 | 0 | 0 | 22 | 0 | 22 |
| UBO [16] | 5 | 7 | 1 | 1 | 0 | 12 | 2 | 14 |
| UM-FHS [17] | 4 | 5 | 0 | 0 | 0 | 9 | 0 | 9 |
| UvA [18] | 5 | 9 | 0 | 0 | 0 | 14 | 0 | 14 |
| Unknown (no paper) | 2 | 0 | 0 | 0 | 0 | 2 | 0 | 2 |
| Total | 83 | 49 | 31 | 31 | 4 | 132 | 66 | 198 |

evaluation measures are "blind" and don't punish the unwarranted generation of additional content. This task addresses one of the main challenges in the Track, CLEF, and the fields of NLP and IR in general. Third, by popular demand, we will revisit and rerun some earlier tasks to ensure that the transition to the new track setup will retain the active track participants of earlier years.

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: SimpleText 2024 Revisited*** *selected tasks by popular request.*

This paper gives an overview of the CLEF 2025 SimpleText Task 2 on Controlled Creativity, which aims to identify and avoid hallucination. Further detail on the entire track is in the CLEF 2025 SimpleText Track Overview [2]. Additional details on Task 1 on *Text Simplification* are in a companion Task 1 overview paper [3]. We also refer to the respective participants' papers for further details.

A total of 74 teams registered for our SimpleText track at CLEF 2025. A total of 18 teams submitted 198 runs in total for Tasks 1 and 2. The statistics for these runs submitted are presented in Table 1.[1] However, some runs had problems that we could not resolve. We do not detail them in the rest of the paper and leave out the 0-scoring runs. More details about individual runs and experiments can be found in the participants' papers, also shown in Table 1.

The rest of this paper is structured in the following way. 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.

[1] The table includes submissions in the Tasks 1 and 2 Codabench evaluation platform, where we were privileged to have 29 (Task 1) and 13 (Task 2) participants.

**Table 2**
Example of a participant's output versus input: ~~deletions~~, insertions, and whole sentence insertions

| **Abstract G01.1_130055196** |
|---|
| *As various kinds of output devices emerged , such as highresolution printers or a display of PDA ( Personal Digital Assistant ) ~~, the~~ . The importance of high-quality resolution conversion has been increasing . \|This paper proposes a new method for enlarging an image with high quality . It will involve using a combination of high-speed imaging and high-resolution video . \|One of the ~~largest~~ biggest problems on image enlargement is the exaggeration of the jaggy edges . This is especially true when the image is enlarged , as in this case . \|To remedy this problem , we propose a new interpolation method ~~, which~~ . This method uses artificial neural network to determine the optimal values of interpolated pixels . \|The experimental results are shown and evaluated . The results are compared to other studies and found to be inconclusive . \|The effectiveness of our methods is discussed by comparing with the conventional methods . Our methods are designed to help people with mental health problems , not just as a way to cure them . \|* |

## 2.1. Description

The *Controlled Creativity* task aims to *identify and avoid hallucination.* To our own surprise, the SimpleText track has collected a massive collection of spurious or overgeneration content from its participants in earlier years of the track. Table 3 shows an example output simplification of one of the participating teams. For the CLEF 2024 task on text simplification, a total of 17 out of 36 submissions (47%) contain spurious whole sentences in at least 10% of the input sentences. In fact, 14 submissions (39%) have spurious sentences in at least 20% of the input, while 7 submissions (19%) have them in at least 50% of the input sentences [19]. Our text simplification setup has sources, predictions, and references that are closely aligned and in the same language. This design allows us to study source attribution and creative variation while also identifying and avoiding what is informally referred to as "hallucinations." This task builds on earlier manual analysis of information distortion in our track since 2022 [20, 21, 19], and similar work by others [22].

**Task 2.1** is to identify creative generation, at the abstract or document level. We will provide realistic system outputs from participants in previous years, along with some intentionally generated outputs from known models. The task is to identify which sentences are fully grounded in the source input: (a) without access to the source sentences and (b) with access to them. This also includes labeling sentences that introduce significant new content. Task 2.1 can be seen as a post-hoc identification task.

**Task 2.2** focuses on detecting and classifying information distortion in simplified sentences. Specifically, it is a multi-label text classification task in which participants are asked to identify the types of information distortion issues based on the annotation scheme introduced by Vendeville et al. [23]. This scheme discerns four broad categories of information distortion:

**A. Fluency** Is the answer provided in a correct form that a fluent speaker would speak?

**B. Alignment** Is the format of the answer correct?

**C. Information** Is the information provided accurate and relevant to the input?

**D. Simplification** Does the response focus on simplification?

Each group contains several fine-grained error types, for a total of 14 classes.[2] The test set is based on manual annotations, while the training set consists of synthetically generated simplifications containing targeted errors. Both datasets were constructed using runs submitted to previous editions of the SimpleText track.

[2] Our annotation scheme focuses on content and meaning preservation. Following [22], we use the word "error" as a general term for annotated issues. The term error is used for brevity, acknowledging that some cases can be considered acceptable in a text simplification context.

**Table 3**
Example of an input of Task 2.1 labeled *spurious* = **True** with ~~deletions~~ and insertions

<table>
<tr><th>Source abstract G05.2_99676222</th></tr>
<tr><td>Summary: We propose the in vivo/vitro use of prokaryotic adaptive immune systems for distributed learning. In the coming years synthetic biologists will learn to control, program, and modify such systems. We design an enhancement to CRISPR-Cas immune systems and demonstrate the learning potential of the modified system by showing it can approximate solutions to a computationally hard problem. To our knowledge this is the first proposed use of CRISPR-Cas systems for computational purposes.</td></tr>
<tr><td><b>Generated Simplification</b></td></tr>
<tr><td>~~Summary :~~ We ~~propose the~~ suggest that in vivo/vitro use of prokaryotic adaptive immune systems for distributed learning . |In the coming years synthetic biologists will learn to control , program , and modify such systems . |We ~~design an enhancement to~~ found that CRISPR-Cas immune systems ~~and demonstrate the learning potential of the modified system by showing it~~ can approximate solutions to a computationally hard problem . |To our knowledge this is the first proposed use of CRISPR-Cas systems for computational purposes .</td></tr>
</table>

**Task 2.3** Finally, we have a text alignment on avoiding creative generation and performing grounded generation by design. This task mirrors Task 1 on text simplification and requires the submission of pairs of runs, both with and without source grounding or source attribution by design.

## 2.2. Data

In running the SimpleText track over the last three years, we have collected an extensive set of realistic and representative predictions in the run submissions. For Tasks 2.1 and 2.2, we use this corpus of realistic generations to build datasets with appropriate labels. Task 2.1 focuses on identifying whether a generated sentence in the prediction is spurious. This is essentially a sentence label task, and the data was provided accordingly. We constructed the data using simplifications submitted in previous SimpleText labs. These simplifications were evaluated based on token-level alignment with the source document. If more than 10% of the generated tokens could not be aligned with the source, the corresponding sentence was labeled as "*spurious*".

From this, we identify 2 cases:

- "**sourced**": participants are tasked with labeling the generation *with* access the source
- "**posthoc**": participants are tasked with labeling the generation *without* access the source

We provide both test and train datasets, as well as a source dataset containing the abstracts for the *sourced* runs.

Task 2.2 mimics the human annotation of information distortion as done in earlier years of the track. Specifically, each simplification was labeled according to the annotation scheme of [23].

Finally, for Task 2.3, we use the same data as for tasks 1.1 and 1.2, and expect the same format of output.

**Train data** For Task 2.1, we selected 782 abstracts used at CLEF 2024 SimpleText Task 3 on Text Simplification. The Task 2.1 train data with sentence labels consisted of 13,341 sentences (posthoc) and 13,514 sentences (sourced). The prevalence was very high: 11,991 (89.9%) sentences were labeled spurious for posthoc and 12,115 (89.6%) sentences for sourced.

For Task 2.2, the train data is based on a synthetic dataset starting from simplifications previously annotated as error-free. We started with submissions from past years that we annotated as error-free. Then, we used a combination of formal algorithms and large language models (LLMs) for each error class in the taxonomy to generate variants of the simplification containing the targeted error class. This approach enabled us to create a large-scale training dataset without relying on time-expensive manual annotation. The set contained 42,392 sentences with a detailed information distortion label.

**Test data** For Task 2.1, the test data with sentence labels consisted of 3,336 sentences (posthoc) and 3,379 sentences (sourced). The prevalence was very high: 3,006 (90.1%) sentences were labeled as spurious for posthoc, and 3,033 (89.8%) sentences for sourced.

The test data for Task 2.2 consists of 2,659 sentences produced by participants in previous years of the SimpleText challenge. We manually annotated 2,659 sentences with an information distortion taxonomy of [23]. The submission will be evaluated against these manually annotated sentences. A total of 820 (30.1%) of sentences had no errors, and 1,839 were classified into four categories (Fluency, Alignment, Information, and Simplification issues) and 14 detailed types.

Task 2.3 follows the setup of *Task 1: Text Simplification* in [3], but requested paired runs with and without the special processing to avoid ungrounded generation.

## 2.3. Formats

### 2.3.1. Train data

**Task 2.1** The two instances—*posthoc* and *sourced*—differ mainly in the fields *gen_id* and *anon_gen_id*. For *sourced*, we construct the *gen_id* as:

`"<anonymised_run_id>//<abs_id>//<sentence_id>"`

We also provide the *abs_id* as a separate field to facilitate joining with the abstracts. For *posthoc*, we provide *anon_gen_id*, formatted similarly, but all components are anonymized:

`"<anonymised_run_id>//<anonymised_abs_id>//<anonymised_sentence_id>"`

In both cases, the data is provided in JSON Lines (jsonl) format.

**Example format for Task 2.1 (posthoc):**

```
{
  "sentence": "Here's the simplified sentence:\n\n'Sometimes, when you're playing on a computer
  ↪  or tablet, special tiny helpers called 'cookies' can follow you around.",
  "is_spurious": true,
  "anon_gen_id": "74704850//98491492//4"
}
```

**Example format for Task 2.1 (sourced):**

```
{
  "abs_id": "G10.1_2010209632",
  "sentence": "system and present our results.",
  "is_spurious": true,
  "gen_id": "35623979//G10.1_2010209632//7"
}
```

**Task 2.2** For this task, we provide a jsonl file containing the synthetically generated simplifications, annotated with error types and identifiers. Each entry includes:

- *snt_id*: Represents the source document and sentence.
- *simp_id*: Identifies the simplification and the error-generation algorithm; both are anonymized.

**Example format for Task 2.2:**

```
{
  "source sentence": "Compliance to the GDPR is a problem for organizations, it imposes strict
  ↪  constraints whenever they deal with personal data and, in case of infringement, it
  ↪  specifies severe consequences such as legal and monetary penalties.",
  "simplified sentence": "Organizations face challenges in complying with the GDPR, which sets
  ↪  strict rules for handling personal data and imposes penalties for violations.",
  "snt_id": "G15.3_2766353613_2",
  "simp_id": "429978-180325",
  "No error": false,
  "A1. Random generation": false,
  "A2. Syntax error": false,
  "A3. Contradiction": false,
  "A4. Simple punctuation / grammar errors": false,
  "A5. Redundancy": false,
  "B1. Format misalignment": false,
  "B2. Prompt misalignment": false,
  "C1. Factuality hallucination": false,
  "C2. Faithfulness hallucination": false,
  "C3. Topic shift": false,
  "D1.1. Overgeneralization": true,
  "D1.2. Overspecification of Concepts": false,
  "D2.1. Loss of Informative Content": false,
  "D2.2. Out-of-Scope Generation": false
}
```

**Task 2.3** follows the same format as tasks 1.1 and 1.2. **Example format for Task 2.3 sentence-level:**

```
{
    "pair_id": "CD009102",
    "complex": "However, the evidence is very uncertain.",
    "simple": "['As a result, we have little confidence in the evidence and the results of this
    ↪  outcome should be interpreted with caution.']"
 }
```

**Example format for Task 2.3 abstract-level:**

```
{
    "pair_id": "CD008996",
    "complex": "A total of 1437 adult patients participated in the five randomized parallel
    ↪  group studies, with treatment durations ranging from 8 to 16 weeks. The daily doses of
    ↪  eplerenone ranged from 25 mg to 400 mg daily. Meta-analysis of these studies showed
    ↪  [...]",
    "simple": "These studies followed patients for 8 to 16 weeks while on therapy. The doses of
    ↪  eplerenone used in these studies ranged from 25 mg to 400 mg daily. None of the studies
    ↪  reported on the clinically meaningful outcomes of eplerenone, such as whether
    ↪  eplerenone can reduce [...]"
 }
```

### 2.3.2. Test Data

**Task 2.1** The format is the same as for training but without the *is_spurious* field. In both cases, the data is provided in JSON Lines (jsonl) format.

**Example format for Task 2.1 (posthoc):**

```
{
  "sentence": "I explained the complex terms directly within the simplified sentence: *
  ↪  'Next-generation model' means a new and improved plan.",
  "anon_gen_id": "74704850//66348262//3"
}
```

**Example format for Task 2.1 (sourced):**

```
{
  "abs_id": "G01.1_1570837852",
  "sentence": "In this paper, we share our findings on how evolutionary algorithms and
  ↪  multi-agent systems can be used to understand a user's preferences while they interact
  ↪  with a digital assistant.",
  "gen_id": "11102757//G01.1_1570837852//1"
}
```

**Task 2.2** The format is the same as for training but without the labels in error field. The data is provided in JSON Lines (jsonl) format.

**Example format for Task 2.2:**

```
{
    "source sentence": "Two dependent variables (satisfaction and perceived sincerity of
    ↪  response) were measured.",
    "simplified sentence": "Two dependent variables ( satisfaction and perceived sincerity of
    ↪  response ) were measured .",
    "snt_id": "G03.1_2960901639_6",
    "simp_id": 427515.0,
    "No error": nan,
    "A1. Random generation": nan,
    "A2. Syntax error": nan,
    "A3. Contradiction": nan,
    "A4. Simple punctuation / grammar errors": nan,
    "A5. Redundancy": nan,
    "B1. Format misalignement": nan,
    "B2. Prompt misalignement": nan,
    "C1. Factuality hallucination": nan,
    "C2. Faithfulness hallucination": nan,
    "C3. Topic shift": nan,
    "D1.1. Overgeneralization": nan,
    "D1.2 Overspecification of Concepts": nan,
    "D2.1. Loss of Informative Content": nan,
    "D2.2. Out-of-Scope Generation": nan
 }
```

**Task 2.3** follows the same format as tasks 1.1 and 1.2. **Example format for Task 2.3 sentence-level:**

```
{
    "pair_id": "CD012520",
    "para_id": 0,
    "sent_id": 0,
    "complex": "We included seven cluster-randomised trials with 42,489 patient participants
    ↪  from 129 hospitals, conducted in Australia, the UK, China, and the Netherlands."
 }
```

**Example format for Task 2.3 abstract-level:**

```
{
    "pair_id": "CD012520",
    "source": "Cochrane",
    "complex": "We included seven cluster-randomised trials with 42,489 patient participants
    ↪  from 129 hospitals, conducted in Australia, the UK, China, and the Netherlands. Health
    ↪  professional participants (numbers not specified) included nursing, medical and allied
    ↪  health professionals. Interventions in all studies included [...]"
 }
```

### 2.3.3. Sources

For Task 2.1 *sourced*, we provide a source file containing all abstracts used. This file is also in jsonl format and includes the following fields:

```
{
    "query_id": "G07.1",
    "query_text": "misinformation",
    "doc_id": 2100028027,
    "abs_id": "G07.1_2100028027",
    "abs_source": "Inaccurate information, in the field of library and information science, is often
    ↪  regarded as a problem that needs to be corrected or simply understood as either
    ↪  misinformation or disinformation without [...]"
 }
```

### 2.3.4. Predictions

In all cases, we asked for JSON submissions, but during evaluation we tried to parse JSON, JSONL, CSV and TSV formats to fix any error by the participants. We also expected a run_id of the format:

*<team-name>_<task-name>_<method-used>*

In practice, some participants used "_" in the method names so we parsed everything after the task name into the method used.

**Task 2.1** For this task, we expected a JSON file containing the sentence, identifier (either *gen_id* or *anon_gen_id*), *is_spurious* label, and the run_id

**Example format for Task 2.1 sourced**

```
{
    "sentence":"In this paper, we share our findings on how evolutionary algorithms and
    ↪  multi-agent systems can be used to understand a user's preferences while they interact
    ↪  with a digital assistant.",
    "gen_id":"11102757//G01.1_1570837852//1",
    "is_spurious":false,
    "run_id":"UBOnlp_task21sourced_gpt4o"
}
```

**Example format for Task 2.1 posthoc**

```
{
    "sentence":"I explained the complex terms directly within the simplified sentence:\n\n*
    ↪  \"Next-generation model\" means a new and improved plan.",
    "anon_gen_id":"74704850//66348262//3",
    "is_spurious":false,
    "run_id":"UBOnlp_task21posthoc_gpt4o"
}
```

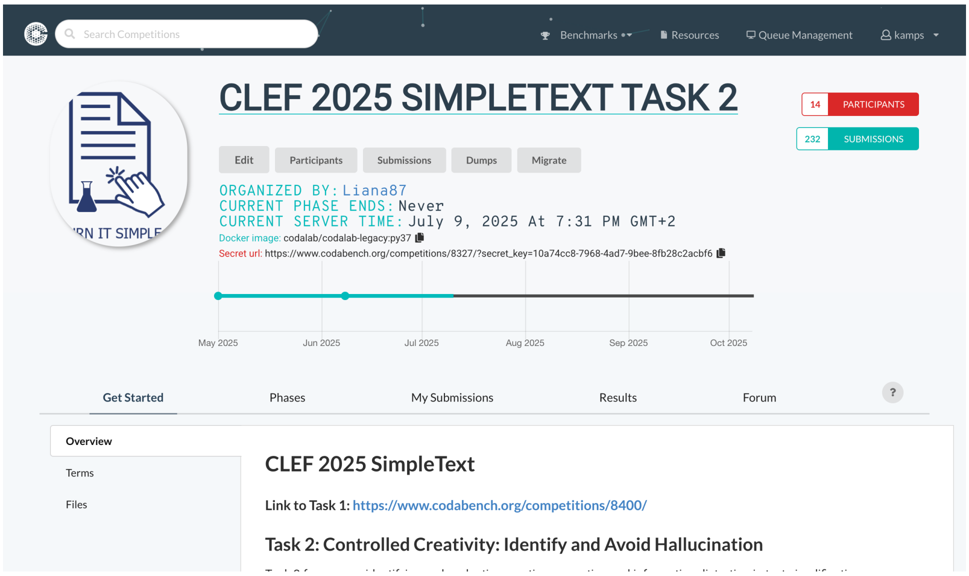
**Task 2.2** For this task, we expected a JSON file containing the source and simplified sentences, the snt_id and simp_id identifiers, the the run_id, and a label for each error class.

**Example format for Task 2.1 sourced**

```
{
    "source sentence":"Partners In Health (PIH) and its sister organization in Lima, Peru,
    ↪  Socios En Salud (SES), treat a majority of multidrug-resistant tuberculosis (MDR-TB)
    ↪  patients in Peru, in conjunction with the Peruvian National TB Program (NTP).",
    "simplified sentence":"Socios En Salud (SES) and its sister organization in Lima, Peru,
    ↪  treat a majority of multidrug-resistant tuberculosis (MDR-TB) patients in Peru, in
    ↪  conjunction",
    "snt_id":"G01.1_147704292_1",
    "simp_id":783909,
    "run_id":"UBOnlp_task22_gpt4o",
    "No error":false,
    "A1. Random generation":false,
    "A2. Syntax error":false,
    "A3. Contradiction":false,
    "A4. Simple punctuation / grammar errors":false,
    "A5. Redundancy":false,
    "B1. Format misalignement":false,
    "B2. Prompt misalignement":false,
    "C1. Factuality hallucination":false,
    "C2. Faithfulness hallucination":true,
    "C3. Topic shift":false,
    "D1.1. Overgeneralization":false,
    "D2.1. Loss of Informative Content":true,
    "D2.2. Out-of-Scope Generation":false,
    "D1.2. Overspecification of Concepts":false
}
```

**Task 2.3** **Example format for Task 2.3 Abstract level**

```
{
    "pair_id": "CD012520",
    "source": "Cochrane",
```



CLEF 2025 SIMPLETEXT TASK 2

ORGANIZED BY: Liana87
CURRENT PHASE ENDS: Never
CURRENT SERVER TIME: July 9, 2025 At 7:31 PM GMT+2
Docker image: codalab/codalab-legacy:py37
Secret url: https://www.codabench.org/competitions/8327/?secret_key=10a74cc8-7968-4ad7-9bee-8fb28c2acbf6

CLEF 2025 SimpleText

Link to Task 1: https://www.codabench.org/competitions/8400/

Task 2: Controlled Creativity: Identify and Avoid Hallucination

**Figure 1:** CLEF 2025 SimpleText Task 2 Codabench.

```
    "complex": "We included seven cluster-randomised trials with 42,489 patient participants
    ↪  from 129 hospitals, conducted in Australia, the UK, China, and the Netherlands. Health
    ↪  professional participants (numbers not specified) included [...]",
    "run_id": "AIIRLab_task12_Mistral_7b_base_grounded"
},
```

### Example format for Task 2.3 sentence level

```
{
    "pair_id":"CD012520","para_id":0,"sent_id":0,"complex":"We included seven
    ↪  cluster-randomised trials with 42,489 patient participants from 129 hospitals,
    ↪  conducted in Australia, the UK, China, and the Netherlands.",
    "prediction":"We studied seven trials with 42,489 patients from 129 hospitals in four
    ↪  countries.",
    "run_id":"dsgt_Task11_plan_guided_llama_grounded"
},
```

## 2.4. Codabench

Submissions were made through Codabench.[3] Due to the differences in the setup, each task had a designated separate competition on Codabench. The Task 1 runs were submitted at: https://www.codabench.org/competitions/8400/. The Task 2 runs were submitted at: https://www.codabench.org/competitions/8327/ (shown in Figure 1). The Codabench greatly facilitated running the track in 2025 and provided active participants (who had also registered at the Codabench) with full access to the competition, including the submission and leaderboard pages.

## 2.5. Evaluation

Task 2.1 is essentially a sentence label task, evaluated in the standard way with Precision, Recall, F1, and AUC. Task 2.2 is a multi-label classification task. We evaluate performance using both F1 score and AUC, computed for individual classes and aggregated across the four main classes. Task 2.3 will be evaluated by both standard automatic measures and human evaluation, following Task 1 on Text Simplification in [3]. We also conduct a more detailed overgeneration analysis for Task 2.3.

[3] https://www.codabench.org/

## 3. Participant's Approaches

A total of 9 teams submitted 66 runs in total. In the detailed results, we only include runs without errors, which got a non-zero score.

*AIIRLab* Largey et al. [4] submitted 10 runs in total for Task 2. They submitted five runs for Task 2.1, five runs for Task 2.2, and none for Task 2.3. They use a combination of four different methods for detecting spurious sentences: an abstract meaning representation, an encoder classifier, majority voting over three models (QWEN, Mistral, LLaMA), and extensive textual features. Furthermore, they use a trained RoBERTa classifier for multi-label prediction, and an ensemble of three models (LLaMA, Mistral, and Openchat) to classify each type of information distortion. Their Task 2.3 were submitted to Task 1.1, were they deployed their Task 1 approach with special precautions against noise and unwanted output.

*DSGT* Marturi and Elwazzan [7] submitted 15 runs in total for Task 2. They submitted six runs for Task 2.1, six runs for Task 2.2, and three runs for Task 2.3. The paper uses an advanced set of approaches, including classifiers, semantic similarity, entailment, and LLM as a Judge, for Task 2.1. They use DeBERTa and LLaMA classifiers for Task 2.2. Finally, for Task 2.3, they repurposed the Task 1 two-stage approach and added a third stage in which they use LLaMA to check and revise the output for content not in the source document. Details about the Task 1 approach are in a separate paper [6].

*DUTH* Arampatzis and Arampatzis [8] submitted four runs in total for Task 2. They submitted two runs for Task 2.1, two runs for Task 2.2, and none for Task 2.3. For Task 2.1, the paper uses a set of classifiers trained on lexical features to detect spurious sentences and obtains high performance. For Task 2.2, a multi-class classifier is trained on the embeddings of the sentence pairs to classify the pairs into the given labels.

*Mtest* (no paper) submitted two runs in total for Task 2. They submitted one run for Task 2.1, one run for Task 2.2, and none for Task 2.3.

*RECAIDS* Eugin et al. [12] submitted two runs in total for Task 2. They submitted one run for Task 2.1, one run for Task 2.2, and none for Task 2.3. They explore a T5 model for Tasks 2.1 and 2.2, using a straightforward T5 completion prompt, with a model fine-tuned on each task.

*SINAI* Collado-Montañez et al. [14] submitted 30 runs in total for Task 2. They submitted 15 runs for Task 2.1, 15 runs for Task 2.2, and none for Task 2.3. They use a rule-based approach to Task 2.1, exploiting some features or artifacts of the data and task setup, followed by an LLaMA model for final classification. Their results show high degrees of effectiveness under both conditions of having access to the source.

*UBO* Vendeville et al. [16] submitted two runs in total for Task 2. They submitted one run for Task 2.1, one for Task 2.2, and none for Task 2.3. The submissions were mostly test submissions, but the paper documents an interesting LLM approach to directly apply the annotation scheme for information distortion as used in the human evaluation of Task 2.2.

## 4. Results

This section details the task results for the overgeneration detection subtask, information distortion detection, classification subtask, and the grounded text simplification subtask.

### 4.1. Task 2.1: Identify Creative Generation

Task 2.1 aims to identify overly creative generation in scientific text simplification. This is a new task that 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 4**
Results for CLEF 2025 SimpleText Task 2.1 Detecting Overgeneration: Test data, posthoc detection without sources, best five runs per team

| Team/Method | count | Acc. | Prec | Rec | F1 | AUROC | AUPRC |
|---|---|---|---|---|---|---|---|
| SINAI basic-prefilter-all-true | 3,336 | 0.91 | 0.91 | 1.00 | 0.95 | 0.55 | 0.91 |
| DSGT bertclassifier | 3,336 | 0.91 | 0.93 | 0.97 | 0.95 | 0.64 | 0.93 |
| DSGT bert_nli_llm_ensemble | 3,336 | 0.90 | 0.93 | 0.97 | 0.95 | 0.64 | 0.93 |
| DSGT bertnlillmensemble | 3,336 | 0.90 | 0.93 | 0.97 | 0.95 | 0.64 | 0.93 |
| DUTH Task21posthoc_et | 3,336 | 0.90 | 0.92 | 0.97 | 0.95 | 0.62 | 0.92 |
| DUTH Task21posthoc_rf | 3,336 | 0.90 | 0.92 | 0.97 | 0.94 | 0.63 | 0.92 |
| DUTH Task21posthoc_svc | 3,336 | 0.79 | 0.94 | 0.83 | 0.88 | 0.66 | 0.93 |
| DUTH Task21posthoc_xgb | 3,336 | 0.79 | 0.94 | 0.81 | 0.87 | 0.69 | 0.94 |
| DUTH Task21posthoc_logreg | 3,336 | 0.77 | 0.95 | 0.79 | 0.86 | 0.70 | 0.94 |
| DSGT llm | 3,336 | 0.77 | 0.95 | 0.78 | 0.86 | 0.70 | 0.94 |
| DSGT nli_entailment | 3,336 | 0.45 | 0.95 | 0.41 | 0.57 | 0.61 | 0.92 |
| SINAI improved-prefilter-all-true | 3,336 | 0.37 | 0.94 | 0.32 | 0.47 | 0.57 | 0.91 |
| SINAI improved-prefilter-confidence-95 | 3,336 | 0.35 | 0.95 | 0.29 | 0.44 | 0.57 | 0.91 |
| UBOnlp gpt4o | 3,379 | 0.27 | 0.92 | 0.21 | 0.35 | 0.52 | 0.90 |

Table 4 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 scores are generally high, with many systems performing over 90% accuracy, F1, and AUC-PR. The test collection contains a variety of information distortion issues (see Task 2.2 and Task 2.3 for more details), including some clear "errors" such as leaving in prompts, or systematic errors in extracting the simplified content from the output of models. However, it also contains complex cases to detect (like the example in Table 3). Hence, the performance is encouraging. Second, it is interesting that trained classifiers such as encoders seem to outcompete larger and modern models as decoders for this task. This may result from the specific task setting, where effective training will pay off. Third, while the task was intended to present entire abstracts or documents, a sentence label task was more practical to run in this first year. This may have effectively reduced the task to a sentence-level task, which may have been easier than a long document-level task.

Table 5 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 would intuitively make the task far easier: human assessors generally rely on this to make their judgments. We see a notable increase in the performance of models, even in AUC-RO, which was lagging in Table 4 above. Second, similar to above, we see that trained or fine-tuned encoders are very effective, generally outcompeting larger decoder models with prompting and few-shot, in-context learning. Third, in the context of source-prediction pairs of sentences, the task is more straightforward than observing a long source document paired to a lengthy list of prediction sentences. Still, the near-perfect performance of the best submissions is very encouraging.

This completes the discussion of the Task 2.1 experiments. For the source-prediction pairs, we expected that the better systems would be able to perform close to perfection. These results indicate that it is possible to detect information distortion errors, such as overgeneration, in the output of current systems. Current evaluation measures based on the overlap with references are insensitive to such additions or redundant content freely generated by the models. Effective detection models can help identify and quantify these issues in the output of models, which is of great importance in further advancing scientific text simplification models.

**Table 5**
Results for CLEF 2025 SimpleText Task 2.1 Detecting Overgeneration: Test data, detection with sources, best five runs per team

| Team/Method | count | Acc. | Prec | Rec | F1 | AUROC | AUPRC |
|---|---|---|---|---|---|---|---|
| AIIRLab CrossEncoder | 3,379 | 0.98 | 0.99 | 0.99 | 0.99 | 0.95 | 0.99 |
| Mtest bartfinetuned | 3,379 | 0.97 | 0.99 | 0.97 | 0.98 | 0.96 | 0.99 |
| SINAI improved-prefilter-all-true | 3,379 | 0.96 | 1.00 | 0.95 | 0.98 | 0.98 | 0.99 |
| SINAI prefilter-all-true | 3,379 | 0.95 | 0.95 | 1.00 | 0.97 | 0.77 | 0.95 |
| AIIRLab RandomForest | 3,379 | 0.95 | 0.95 | 1.00 | 0.97 | 0.77 | 0.95 |
| SINAI improved-prefilter-confidence-99 | 3,379 | 0.93 | 1.00 | 0.93 | 0.96 | 0.96 | 0.99 |
| SINAI llama3.1-8b-instruct | 3,379 | 0.93 | 0.95 | 0.97 | 0.96 | 0.77 | 0.95 |
| DSGT bertclassifier | 3,379 | 0.91 | 0.93 | 0.98 | 0.95 | 0.65 | 0.93 |
| DSGT bertnlillmensemble | 3,379 | 0.91 | 0.93 | 0.97 | 0.95 | 0.68 | 0.93 |
| DUTH Task21sourced_et | 3,379 | 0.91 | 0.93 | 0.97 | 0.95 | 0.66 | 0.93 |
| DUTH Task21sourced_rf | 3,379 | 0.90 | 0.93 | 0.96 | 0.95 | 0.65 | 0.93 |
| DUTH Task21sourced_svc | 3,379 | 0.80 | 0.94 | 0.83 | 0.88 | 0.69 | 0.93 |
| SINAI improved-prefilter-confidence-95 | 3,379 | 0.81 | 1.00 | 0.79 | 0.88 | 0.89 | 0.98 |
| DUTH Task21sourced_ridge | 3,379 | 0.77 | 0.94 | 0.79 | 0.86 | 0.68 | 0.93 |
| DUTH Task21sourced_logreg | 3,379 | 0.77 | 0.94 | 0.79 | 0.86 | 0.69 | 0.93 |
| DSGT llm | 3,379 | 0.74 | 0.94 | 0.76 | 0.84 | 0.68 | 0.93 |
| UBOnlp gpt4o | 3,379 | 0.70 | 0.95 | 0.71 | 0.81 | 0.69 | 0.93 |
| RECAIDS T5 | 3,379 | 0.49 | 0.89 | 0.49 | 0.63 | 0.47 | 0.89 |
| DSGT nli_entailment | 3,379 | 0.35 | 0.92 | 0.31 | 0.46 | 0.53 | 0.90 |
| DSGT nli_contradiction | 3,379 | 0.20 | 0.90 | 0.12 | 0.21 | 0.50 | 0.90 |
| AIIRLab LLMs | 3,379 | 0.10 | 0.00 | 0.00 | 0.00 | 0.50 | 0.90 |
| AIIRLab LLMs | 3,379 | 0.10 | 0.00 | 0.00 | 0.00 | 0.50 | 0.90 |

## 4.2. Task 2.2: Detect and Classify Information Distortion Errors

Task 2.2 is a new task that 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.

We evaluate this task using a corpus of 2,659 manually annotated sentence–simplification pairs. Each simplified sentence may contain multiple error types, making this a multi-label classification problem. The error taxonomy is organized hierarchically into four categories (A–D), each comprising several fine-grained error types. For evaluation, predicted and gold error labels are aggregated at the group level: if any fine-grained error from a group is present, the group is considered active. Performance is then measured per group using both micro and macro F1 scores. We also consider the "No Error" class, indicating no errors were detected.

Results are presented in Table 6. The table includes only valid submissions, excluding 39 duplicates where teams submitted the same method multiple times, where we retain only the run with the highest F1 score on the *No Error* class. The results displayed here are limited to the best five runs per team, and are sorted by F1 score on *No Error* class.

From this, we make several observations. First, while some models were able to perform well on *No Error*, achieving over 0.65 F1 scores, performance quickly drops, and over half of them do not achieve 0.50 F1 scores. Second, results are quite low for all other groups. For Fluency issues (group A), the five best systems achieve an F1 score between 0.255 and 0.283. For Alignment (group B), only 55% of the systems achieved over 0.10 F1 scores, with 20% over 0.25 and up to 0.47. For Information issues (group C), only two systems achieved over 0.25 F1 scores (with 0.30 and 2.69), with the next 60% achieving between 0.10 and 0.17. Finally, for Simplification issues (group D), only the same two models were able to achieve F1 scores above 0.25 (with 0.37 and 0.30) while the next 60% achieved between 0.12 and 0.24. Third, more generally, the results suggest that detecting specific error categories remains a challenging task, especially under realistic conditions with a multi-label setting and imbalanced data. The relatively strong performance on the *No Error* class demonstrates that distinguishing error-free simplifications

**Table 6**
Model Performance on CLEF 2025 SimpleText Task 2.2. Results by Error Categories (Best Scores in Bold) for No error, Fluency (A), Alignment(B), Information (C), and Simplification (D) categories, with $F_1$ and AUC-PR, best five runs per team

| Team/Method | No Error | | A | | B | | C | | D | |
|---|---|---|---|---|---|---|---|---|---|---|
| | $F_1$ | AUC | $F_1$ | AUC | $F_1$ | AUC | $F_1$ | AUC | $F_1$ | AUC |
| DSGT DebertaLlmensemble | **0.763** | 0.561 | 0.283 | 0.133 | 0.354 | 0.173 | **0.301** | **0.156** | **0.374** | **0.224** |
| AIIRLab paraphrase_mpnet | 0.755 | **0.567** | 0.255 | 0.154 | 0.258 | 0.113 | 0.136 | 0.084 | 0.147 | 0.168 |
| AIIRLab mpnet | 0.744 | 0.557 | 0.255 | **0.156** | 0.218 | 0.099 | 0.150 | 0.091 | 0.147 | 0.167 |
| DSGT roberta | 0.694 | 0.491 | 0.233 | 0.121 | 0.249 | 0.101 | 0.114 | 0.089 | 0.128 | 0.164 |
| UBOnlp gpt4o | 0.680 | 0.505 | **0.322** | 0.150 | 0.381 | 0.192 | 0.250 | 0.122 | 0.292 | 0.189 |
| DSGT llama | 0.680 | 0.483 | 0.282 | 0.132 | 0.324 | 0.182 | 0.269 | 0.147 | 0.306 | 0.196 |
| AIIRLab OpenChat | 0.640 | 0.421 | 0.154 | 0.070 | 0.141 | 0.061 | 0.144 | 0.080 | 0.222 | 0.156 |
| AIIRLab MajorityVoting | 0.633 | 0.415 | 0.156 | 0.071 | 0.110 | 0.045 | 0.170 | 0.088 | 0.239 | 0.160 |
| AIIRLab Mistral | 0.563 | 0.357 | 0.158 | 0.069 | 0.104 | 0.040 | 0.116 | 0.070 | 0.176 | 0.144 |
| DSGT BERT | 0.515 | 0.330 | 0.214 | 0.133 | 0.208 | 0.103 | 0.167 | 0.095 | 0.129 | 0.161 |
| DUTH scibert | 0.436 | 0.321 | 0.088 | 0.045 | 0.035 | 0.025 | 0.100 | 0.066 | 0.145 | 0.135 |
| DUTH deberta-v3 | 0.404 | 0.322 | 0.003 | 0.044 | 0.051 | 0.026 | 0.006 | 0.064 | 0.093 | 0.136 |
| Mtest bartfinetuned | 0.404 | 0.322 | 0.270 | 0.143 | **0.472** | **0.265** | 0.078 | 0.074 | 0.128 | 0.167 |
| DSGT bert_llama_ensemble | 0.404 | 0.322 | 0.231 | 0.137 | 0.253 | 0.107 | 0.116 | 0.088 | 0.128 | 0.163 |
| DUTH roberta-base | 0.404 | 0.322 | 0.083 | 0.044 | 0.033 | 0.027 | 0.117 | 0.064 | 0.023 | 0.136 |
| RECAIDSTechTitans T5 | 0.404 | 0.322 | 0.022 | 0.046 | 0.000 | 0.026 | 0.004 | 0.065 | 0.000 | 0.136 |
| DUTH logreg | 0.404 | 0.322 | 0.000 | 0.044 | 0.000 | 0.026 | 0.000 | 0.064 | 0.000 | 0.136 |
| DUTH logreg_oversample | 0.404 | 0.322 | 0.021 | 0.046 | 0.000 | 0.026 | 0.004 | 0.064 | 0.000 | 0.136 |

is a realistic and tractable subtask. The gap between detecting no errors and identifying fine-grained error types remains an open research challenge, and the results of the track highlight the complexity of accurately modeling semantic information distortions in the output of current models.

This completes the discussion of the Task 2.2 experiments. The results are mixed. On the one hand, consistent with the results of Task 2.1, we saw that detecting that a prediction has information distortion issues is a viable task for current systems. On the other hand, fine-grained annotation of the types of information distortion remains challenging. This indicates that manual evaluation remains of great value for scientific text simplification and the automatic evaluation measures. Yet the effort and cost of manually annotating all output remains very high, and such human evaluation is not reusable and has to be repeated for every new prediction. One realistic option is to use a hybrid approach. The ability to automatically filter out the cases with no error and judge samples of the remaining predictions to assess the error types and distribution can be a pragmatic and more cost-effective way to scale up human evaluation.

## 4.3. Task 2.3: Avoid Creative Generation

Task 3.2 aims to avoid overly creative generation in scientific text simplification and showcase systems that perform grounded generation by design. This is a new task that asks for a pair of submissions, one of which must make a special effort to avoid overgeneration or other information distortion issues.

Two teams submitted runs for Task 2.3, indicated by "_grounded" in the run names. Some of these runs were specifically submitted to Task 2.3, and others were regular submissions to Tasks 1.1 and 1.2.

Table 7 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 tried to locate the matching baseline runs, indicated with ⋆ in the tables from the earlier results as displayed for Task 1 in [3].

We make several observations. First, the performance is generally competitive, and several runs are among the best-performing runs. This is reassuring, as any attempt to ground the predictions more closely to the source texts should not lead to a dramatic decrease in performance. Second, the

**Table 7**
Results for CLEF 2025 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 llama3.1_gro | 37 | 43.63 | 17.92 | 11.02 | 0.63 | 0.96 | 0.61 | 0.00 | 0.13 | 0.53 | 8.72 |
| AIIRLab llama3.1_cro | 37 | 43.24 | 17.48 | 11.16 | 0.63 | 0.96 | 0.61 | 0.00 | 0.13 | 0.53 | 8.71 |
| DSGT llama_summary_s | 37 | 41.25 | 15.00 | 12.74 | 0.76 | 0.85 | 0.57 | 0.00 | 0.23 | 0.48 | 8.76 |
| ⋆DSGT llama | 37 | 40.32 | 7.63 | 9.56 | 0.59 | 0.86 | 0.42 | 0.00 | 0.31 | 0.70 | 8.49 |
| DSGT plan_guided_lla | 37 | 37.33 | 18.27 | 12.87 | 0.91 | 1.09 | 0.71 | 0.00 | 0.18 | 0.31 | 8.79 |
| ⋆AIIRLab llama3.1-8b | 37 | 31.27 | 19.59 | 11.44 | 0.85 | 1.09 | 0.83 | 0.00 | 0.09 | 0.25 | 8.83 |
| ⋆DSGT llama | 217 | 42.92 | 5.32 | 9.94 | 0.49 | 0.72 | 0.39 | 0.00 | 0.24 | 0.75 | 8.55 |
| DSGT llama_summary_s | 217 | 42.06 | 9.89 | 12.81 | 0.62 | 0.72 | 0.50 | 0.00 | 0.19 | 0.59 | 8.82 |
| AIIRLab llama3.1_gro | 217 | 40.90 | 11.60 | 11.31 | 0.63 | 0.98 | 0.62 | 0.00 | 0.12 | 0.53 | 8.83 |
| AIIRLab llama3.1_cro | 217 | 40.82 | 11.60 | 11.28 | 0.63 | 0.98 | 0.62 | 0.00 | 0.11 | 0.53 | 8.83 |
| DSGT plan_guided_lla | 217 | 33.41 | 10.04 | 12.96 | 0.96 | 1.14 | 0.69 | 0.00 | 0.21 | 0.31 | 8.88 |
| ⋆AIIRLab llama3.1-8b | 217 | 29.80 | 11.32 | 11.19 | 0.83 | 1.10 | 0.80 | 0.00 | 0.10 | 0.29 | 8.93 |

baseline runs without any precautions observe the highest number of additions, indicating that the grounded runs are generally more conservative. Third, although we refer to the participants' papers of AIIRLab [4] and DSGT [6] for specific details, some of the grounding seems to involve more careful output processing, such as ensuring in the prompts that no extra information other than the text simplification is output by the model.

More generally, while the primary goal of prediction grounding is not a performance improvement, it is also the case that other runs with presumably redundant information are not performing less well. The standard measures based on textual overlap with the references are relatively insensitive to additional content in the predictions. This invites further analysis to investigate how well the source information grounds the predictions, and when they are not.

## 4.4. Findings

This concludes the results for the CLEF 2025 SimpleText Task 2: Controlled Creativity on identify and avoid hallucination. Our main findings are the following: First, for Task 2.1 on detecting creative generation, we observed very high performance for identifying overgeneration and other information distortion. This was hoped and expected for pairs of source-prediction content, but unexpected for post hoc detection on only the system's predictions. Second, for Task 2.2 on classifying the type of information distortion, we observed mixed results. Also here we saw solid performance for the "no error" cases, yet identifying the precise type of information distortion similar to human evaluation remains a challenging tasks for current models. Third, for Task 2.3 on avoiding creative generation and performing grounded generation by design, we observed that text simplification measures are immune to detecting overgeneration, and that this remains a serious issue in the predictions. More sensitive text simplification evaluation measures are needed to highlight these aspects and ensure that the research community further develops grounded generation approaches.

# 5. Analysis

## 5.1. Task 2.1 Analysis

Task 2.1 focused on identifying spurious sentences i.e. those that introduce content not grounded in the source text. Participants tackled this task in two settings: post-hoc, where only the prediction was available, and sourced, where both the source and the generated sentence were provided. This distinction simulates real-world scenarios where access to the source may or may not be available, and allows us to explore the limits of detection in both conditions.

Results were encouraging in both settings, but showed clear differences. In the post-hoc setting, several systems still reached high scores over 90% accuracy and F1, suggesting that many spurious sentences are detectable based on surface cues alone. Obvious cases like prompt leaks or formulaic overgeneration patterns were often caught even without source access. However, this setting is inherently more challenging, and performance varied more across teams.

In the sourced setting, access to the input significantly improved model performance. Top submissions achieved near-perfect results, with F1 scores up to 0.99 and very high precision and recall. Having the source allowed systems to make more reliable decisions about whether a sentence was actually grounded, especially in borderline or more nuanced cases.

Interestingly, in both settings, trained encoders and task-specific classifiers generally outperformed larger language models relying on in-context learning. This suggests that for this type of targeted detection task, fine-tuning on aligned examples still offers a strong advantage over general LLM prompting.

Another important aspect is task framing. While the original goal was to evaluate grounding at the document level, we focused on sentence-level labels for this first edition. This likely made the task more approachable, especially for systems that don't model discourse-level context.

These results suggest that while source access helps, it's still possible to detect many hallucinations post-hoc. Effective detection tools are valuable for both evaluating model outputs and flagging risky generations in practice.

## 5.2. Task 2.2 Analysis

Task 2.2 pushed systems beyond simple error detection by asking them to identify what kind of information distortion occurred, based on a 14-class taxonomy. This proved to be much harder than determining whether a sentence was error-free.

Most systems performed reasonably well on the No Error class where several reached F1 scores above 0.70. But performance dropped sharply for the error categories. For example, only a few models scored above 0.30 F1 on Fluency or Simplification issues, and many had near-zero scores for rarer types like Overspecification or Topic shift.

Several factors likely contributed to this. First, the fine-grained labels often overlap and can be subtle, even for human annotators. Second, systems were trained on synthetic data but evaluated on real, human-annotated outputs, which may have led to generalization issues. Third, many error types were underrepresented, and few approaches explicitly addressed this imbalance.

Interestingly, the systems that performed best on the No Error class also tended to score highest on Fluency errors (Group A), but this pattern didn't hold across other categories. For groups B–D, the correlation with No Error performance was much weaker.

The strongest results came from ensemble systems like DSGT's, which combined DeBERTa with LLaMA-based models, and AIIRLab's voting ensemble over several LLMs. In contrast, smaller models or rule-based classifiers struggled more, especially with semantically complex errors.

These results suggest that while identifying clean outputs is a realistic goal, explaining what went wrong remains an open challenge. A promising next step could be a hybrid setup: automatic filtering of error-free outputs, followed by targeted human review of potential issues. This could make evaluation more scalable without sacrificing quality.

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

| Run | SARI | Source | Spurious Content | |
|---|---|---|---|---|
| | (217) | Number | Number | Fraction |
| AIIRLab llama3_grounded | 40.90 | 9,160 | 17 | 0.00 |
| AIIRLab llama3_crossencoder_grounded2 | 40.82 | 9,160 | 15 | 0.00 |
| ⋆AIIRLab llama3-8b | 29.80 | 9,160 | 394 | 0.04 |
| ⋆DSGT plan_guided_llama | 42.98 | 9,160 | 206 | 0.02 |
| DSGT plan_guided_llama_grounded | 33.41 | 9,160 | 477 | 0.05 |
| ⋆DSGT llama_summary_simplification | 42.92 | 666 | 538 | 0.81 |
| DSGT llama_summary_simplification_grounded | 42.06 | 666 | 504 | 0.76 |

## 5.3. Task 2.3 Analysis

We analyzed the entire test data set, comprising 666 documents (Task 1.2) and 9,160 sentences (Task 1.1). This analysis assumes that there is always word overlap between a pair of complex-simple sentences or abstracts. Moreover, we look specifically for overgenerating output at the sentence's or abstract's end. This is typical of sequence-to-sequence models, which are asked to complete the input with a simplified version in standard text completion mode.

Assume we feed the model one long sentence extracted from an abstract, without further context. Now, due to sentence splitting, the output could contain multiple sentences. However, after the input sentence is fully simplified, the model wants to complete the text. Without access to the rest of the source abstract, the model may generate the most likely subsequent sentences. Such sentences are completely unfounded by the source, and it isn't easy to spot these cases in the generated text, as they are indeed coherent and possible continuations. This may occur after every sentence in sentence-level text simplification.

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 observe, indeed, overgeneration/text completion issues at the end of the sources/predictions. There are also cases in which there are systematic errors in extracting the output, 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 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 to the prediction output regarding 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 the predictions and of overgeneration issues in the runs.

We make several observations. First, despite competitive performance in terms of text overlap with the references, we see widely varying numbers of cases of overgeneration, ranging from a few percentage points to large fractions of the output. Second, 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. Some of these may be easily spotted as "noise," such as systematically left-in prompts. Other cases may be challenging to detect in the output by users of text simplification systems. Third, in the context of the task, we see some interesting examples, for example, AIIRLab [4] detected "noise" and changed the prompts to ensure only the simplified text, and nothing else, was in the model output.

# 6. Discussion and Conclusions

This paper describes the setup of the CLEF 2025 SimpleText track, which contains the following three tasks. Task 1 on *Text Simplification*: *simplify scientific text.* Task 2 on *Controlled Creativity*: *identify and* *avoid hallucination.* Task 3 on *SimpleText 2024 Revisited*: *selected tasks by popular request.* This Task overview focuses on the CLEF 2025 SimpleText Track's Task 2 on identifying and avoiding information distortion (or "hallucination"). The main aim of our track, and the CLEF evaluation forum as a whole, is i) to construct corpora and evaluation resources to stimulate research on scientific text summarization and simplification, and ii) to foster a community of IR, NLP, and AI researchers working together on the important task of making science more accessible for everyone.

Within the CLEF 2025 SimpleText Task 2, we have constructed extensive corpora and references for evaluation data. First, we exploited the text simplification setup with aligned sources, references, and the output of generative models to detect, quantify, and avoid spurious information introduced gratuitously by the generative model. This is what is informally referred to as "hallucinations." Addressing the remaining limitations of large generative models is crucial for the scientific use case, as current evaluation measures are "blind" and don't punish the unwarranted generation of additional content. Second, we observed very high accuracy in detecting overgeneration and other types of information distortion in the output of text simplification systems. This task was based on the real output of CLEF 2024 submissions, and the best systems could detect sentence-level information distortion in the predictions with near-perfect accuracy in the presence of the sources. Unexpectedly, the accuracy without access to the source was also very high, even though this may be partly due to the class imbalance in the data. This is a positive result, as the automatic detection of noise and overgeneration in the output of AI models appears to be a viable strategy. Third, detailed classification of information distortion, as is done in small-scale human evaluation, remains challenging to mimic. This may be partly attributed to human inter-annotator variation and the need for detailed, qualitative judgments that require extensive expertise and training. This highlights the remaining value of detailed human analysis in addition to automatic evaluation measures. At the same time, this presents a significant research challenge for future research to address.

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 for researching scientific text summarization and simplification, the track saw a record attendance in 2025, with significant changes in tasks and the move to Codabench. More runs were submitted, and the largest number of participating teams ever was achieved.

## 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 2025 chairs for hosting us, and the CLEF 2025 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/8400/ (Task 1.1, Task 1.2, and Task 2.3) and https://www.codabench.org/competitions/8327/ (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.

Benjamin Vendeville and Liana Ermakova are partly funded by the French National Research Agency (ANR-22-CE23-0019-01, *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 (NWO CI # CISC.CC.016), the University of Amsterdam (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.

## Disclosure of Interests

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

[3] J. Bakker, B. Vendeville, L. Ermakova, J. Kamps, Overview of the CLEF 2025 SimpleText Task 1: Simplify Scientific Text, in: [25], 2025.

[4] N. Largey, D. Wu, B. Mansouri, AIIRLab Systems for CLEF 2025 SimpleText: Cross-Encoders to Avoid Spurious Generation, in: [25], 2025.

[5] A. N. Djoudi, S. Nouali, M. Aabid, I. Badache, A.-G. Chifu, P. Bellot, LIS at the SimpleText 2025: Enhancing Scientific Text Accessibility with LLMs and Retrieval-Augmented Generation, in: [25], 2025.

[6] K. C. Marturi, H. H. Elwazzan, Hallucination Detection and Mitigation in Scientific Text Simplification using Ensemble Approaches: DS@GT at CLEF 2025 SimpleText, in: [25], 2025.

[7] K. C. Marturi, H. H. Elwazzan, LLM-Guided Planning and Summary-Based Scientific Text Simplification: DS@GT at CLEF 2025 SimpleText, in: [25], 2025.

[8] G. Arampatzis, A. Arampatzis, DUTH at CLEF 2025 SimpleText Track: Tackling Scientific Text Simplification and Hallucination Detection, in: [25], 2025.

[9] M. M. Agüero-Torales, C. Rodríguez-Abellán, C. A. C. Moraga, Sentence-level Scientific Text Simplification With Just a Pinch of Data, in: [25], 2025.

[10] Y. Gallina, T. Jiménez, S. Huet, University of Avignon at SimpleText 2025: Guided Medical Abstract Simplification, in: [25], 2025.

[11] A. Vora, T. Chaudhari, S. Hotha, S. Sonawane, S-3 Pipeline by PICT/Pune for Biomedical Text Simplification, in: [25], 2025.

[12] S. Eugin, A. Ms.Beula, V. Sathvikha, V. Sangamithra, SimpleText: Simplify Scientific Text, in: [25], 2025.

[13] A. A. Dongre, A. Vaadiraaju, A. K. Madasamy, NITK SCaLAR Lab at the CLEF 2025 SimpleText Track: Transformer-Based Models for Biomedical Sentence Simplification (Task 1.1), in: [25], 2025.

[14] J. Collado-Montañez, J. A. Ortiz-Zambrano, C. Espin-Riofrio, A. Montejo-Ráez, SINAI in SimpleText CLEF 2025: Simplifying Biomedical Scientific Texts and Identifying Hallucinations Using GPT-4.1 and Pattern Detection, in: [25], 2025.

[15] N. Hofmann, J. Dauenhauer, N. O. Dietzler, I. D. Idahor, C. K. Kreutz, THM@SimpleText 2025 Task 1.1: Revisiting Text Simplification based on Complex Terms for Non-Experts, in: [25], 2025.

[16] B. Vendeville, L. Ermakova, P. D. Loor, J. Kamps, UBONLP Report on the SimpleText lab, in: [25], 2025.

[17] P. Kocbek, G. Stiglic, UM-FHS at the CLEF 2025 SimpleText Track: Comparing No-Context and Fine-Tune Approaches for GPT-4.1 Models in Sentence and Document-Level Text Simplification, in: [25], 2025.

[18] T. Papandreou, J. Bakker, J. Kamps, University of Amsterdam at the CLEF 2025 SimpleText Track, in: [25], 2025.

[19] 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, volume 3740 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2024, pp. 3147–3162. URL: https://ceur-ws.org/Vol-3740/paper-307.pdf.

[20] 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, volume 3180 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2022, pp. 2792–2804. URL: https://ceur-ws.org/Vol-3180/paper-237.pdf.

[21] 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, volume 3497 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 2855–2875. URL: https://ceur-ws.org/Vol-3497/paper-240.pdf.

[22] 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.

[23] B. Vendeville, L. Ermakova, P. D. Loor, Resource for Error Analysis in Text Simplification: New Taxonomy and Test Collection, 2025. doi:10.1145/3726302.3730304. arXiv:2505.16392.

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