Overview of the CLEF 2023 SimpleText Lab

Automatic Simplification of Scientific Texts

Ermakova, L.; SanJuan, E.; Huet, S.; Azarbonyad, H.; Augereau, O.; Kamps, J.

DOI
10.1007/978-3-031-42448-9_30

Publication date
2023

Document Version
Final published version

Published in
Experimental IR Meets Multilinguality, Multimodality, and Interaction

Overview of the CLEF 2023 SimpleText Lab: Automatic Simplification of Scientific Texts

Liana Ermakova[1(✉)], Eric SanJuan[2], Stéphane Huet[2], Hosein Azarbonyad[3], Olivier Augereau[4], and Jaap Kamps[5]

[1] Université de Bretagne Occidentale, HCTI, Brest, France
liana.ermakova@univ-brest.fr
[2] Avignon Université, LIA, Avignon, France
[3] Elsevier, Amsterdam, The Netherlands
[4] ENIB, Lab-STICC UMR CNRS 6285, Brest, France
[5] University of Amsterdam, Amsterdam, The Netherlands

Abstract. There is universal consensus on the importance of objective scientific information, yet the general public tends to avoid scientific literature due to access restrictions, its complex language or their lack of prior background knowledge. Academic text simplification promises to remove some of these barriers, by improving the accessibility of scientific text and promoting science literacy. This paper presents an overview of the CLEF 2023 SimpleText track addressing the challenges of text simplification approaches in the context of promoting scientific information access, by providing appropriate data and benchmarks, and creating a community of IR and NLP researchers working together to resolve one of the greatest challenges of today. The track provides a corpus of scientific literature abstracts and popular science requests. It features three tasks. First, *content selection* (what is in, or out?) challenges systems to select passages to include in a simplified summary in response to a query. Second, *complexity spotting* (what is unclear?) given a passage and a query, aims to rank terms/concepts that are required to be explained for understanding this passage (definitions, context, applications). Third, *text simplification* (rewrite this!) given a query, asks to simplify passages from scientific abstracts while preserving the main content.

Keywords: Scientific text simplification · (Multi-document) summarization · Contextualization · Background knowledge · Comprehensibility · Scientific information distortion

1 Introduction

Scientific literacy is an important ability for people. It is one of the keys to critical thinking, objective decision-making, and judgment of the validity and significance of findings and arguments, which allows discerning facts from fiction. Thus, having basic scientific knowledge may also help maintain one's health, both physiological and mental. The COVID-19 pandemic provides a good example of such a matter. Understanding the issue itself, choosing to use or avoid a particular treatment or prevention procedure can become crucial. However, the recent pandemic has also shown that simplification

A. Arampatzis et al. (Eds.): CLEF 2023, LNCS 14163, pp. 482–506, 2023.
https://doi.org/10.1007/978-3-031-42448-9_30

can be modulated by political needs and the scientific information can be distorted [13]. Thus, the evaluation of the alteration of scientific information during the simplification process is crucial but underrepresented in the state-of-the-art.

Digitization and open access have made scientific literature available to every citizen. While this is an important first step, there are several remaining barriers preventing laypersons to access objective scientific knowledge in the literature. In particular, scientific texts are often hard to understand as they require solid background knowledge and use tricky terminology. Although there were some recent efforts on text simplification (e.g. [18]), removing such understanding barriers between scientific texts and the general public in an automatic manner is still an open challenge. The CLEF 2023 SimpleText track[1] brings together researchers and practitioners working on the generation of simplified summaries of scientific texts. It is an evaluation lab that follows up on the CLEF 2021 SimpleText Workshop [10] and CLEF 2022 SimpleText Track [12]. All perspectives on automatic science popularisation are welcome, including but not limited to: Natural Language Processing (NLP), Information Retrieval (IR), Linguistics, Scientific Journalism, etc.

SimpleText provides data and benchmarks for discussion of challenges of automatic text simplification by bringing in the following interconnected tasks [11]:

Task 1: What is in (or out)? Select passages to include in a simplified summary, given a query.

Task 2: What is unclear? Given a passage and a query, rank terms/concepts that are required to be explained for understanding this passage (definitions, context, applications, ..).

Task 3: Rewrite this! Given a query, simplify passages from scientific abstracts.

A total of 74 teams registered for our SimpleText track at CLEF 2023. A total of 20 teams submitted 139 runs in total. The statistics for these runs submitted are presented in Table 1.

The bulk of this paper presents the tasks with the datasets and evaluation metrics used, as well as the results of the participants, in three self-contained sections: Sect. 2 on the first task about content selection, Sect. 3 on the second task about complexity spotting, and Sect. 4 on the third task about text simplification proper. We end with Sect. 5 discussing the results and findings, and lessons for the future.

2 Task 1: What is in (or Out)?

Given a popular science article targeted to a general audience, this task aims at retrieving passages that can help to understand this article, from a large corpus of academic abstracts and bibliographic metadata. Relevant abstracts should relate to any of the topics in the source article. These passages can be complex and require further simplification to be carried out in tasks 2 and 3. Task 1 focuses on content retrieval.

[1] https://simpletext-project.com.

Table 1. CLEF 2023 Simpletext official run submission statistics

Team	Task 1	Task 2.1	Task 2.2	Task 3	Total runs
Elsevier	10				10
Maine (Aiirlab)	10	3	3	2	18
uninib_DoSSIER	2				2
UAms	10	1		2	13
LIA	7				7
MiCroGerk		4	4	3	11
Croland		2	2		
NLPalma		1	1	1	3
Pandas				6	6
QH				3	3
SINAI		4	2		
irgc				4	4
CYUT				4	4
UOL-SRIS		1			1
Smroltra		10	10	1	21
TeamCAU		3	3	1	7
TheLangVerse		1	1	1	3
ThePunDetectives		2	2	2	6
UBO		7	1	1	9
RT				1	1
Total runs	39	39	29	32	139

2.1 Data

Corpus: DBLP Abstracts. We use the Citation Network Dataset: DBLP+Citation, ACM Citation network (12th version released in 2020).[2] This contains a total of 4,894,063 scientific articles. A JSON dump of the corpus is made available for participants. In addition, an ElasticSearch index is provided to participants with access through an API.

Topics: Press Articles. Topics are a selection of press articles from the technology section of *The Guardian*[3] newspaper (topics G01 to G20) and the *Tech Xplore*[4] website (topics T01 to T20). URLs to original articles and textual content of each topic are provided to participants. All passages retrieved from DBLP by participants are expected to have some overlap (lexical or semantic) with the article content.

[2] https://www.aminer.org/citation.
[3] https://www.theguardian.com/uk/technology.
[4] https://techxplore.com/.

Queries as Facets. For each popular news article, multiple keyword queries are provided, leading to a grand total of 114 requests. It has been manually checked that each query allows retrieving relevant articles related to the topic of the press article.

Qrels. Quality relevance of abstracts w.r.t. topics are given in both the train qrels (released prior to submissions) and the test qrels.

Train Qrels Relevance annotations are provided on a 0–2 scale (the higher the more relevant) for 29 queries associated with the first 15 articles from The Guardian (G01–G15). Specifically, it extends the 2022 qrels released with a significant increase in the depth of judgments of abstracts per query.

Test Qrels Relevance annotations are provided on a 0–2 scale (the higher the more relevant) for 34 queries associated with the 5 articles from The Guardian (G16–G20, 17 queries) and 5 articles from Tech Xplore (T01–T05, 17 queries). These qrels were based on pooling the submissions of 2023 participants.

2.2 Attended Results

Ad Hoc Passage Retrieval. Participants should retrieve, for each topic and each query, all passages from DBLP abstracts, related to the query and potentially relevant to be inserted as a citation in the paper associated with the topic. Some abstracts could be very complex for non-experts. We encourage participants to take into account passage complexity as well as its credibility/influentialness.

Output Format. Results should be provided in a TREC-style JSON or TSV format with the following fields:

run_id Run ID starting with: team_id_task_id_method_used, e.g. *UBO_task_1_TFIDF*
manual Whether the run is manual {0,1}
topic_id Topic ID
query_id Query ID used to retrieve the document (if one of the queries provided for the topic was used; 0 otherwise)
doc_id ID of the retrieved document (to be extracted from the JSON output)
rel_score Relevance score of the passage (higher is better)
comb_score General score that may combine relevance and other aspects: readability, credibility or authoritativeness
passage Text of the selected abstract.

For each query, the maximum number of distinct DBLP references (doc_id field) must be 100 and the total length of passages should not exceed 1,000 tokens. The idea of taking into account complexity is to have passages easier to understand for non-experts, while the credibility score aims at guiding them on the expertise of authors and the value of publication w.r.t. the article topic. For example, complexity scores can be evaluated using readability and credibility scores using bibliometrics.

An example of the output is shown in Table 2. For each topic, the maximum number of distinct DBLP references (_id json field) was 100 and the total length of passages was not to exceed 1,000 tokens.

Table 2. CLEF 2023 SimpleText Task 1 on content selection: example of output

Run	M/A	Topic	Query	Doc	Rel	Comb	Passage
ST1_task1_1	0	G01	G01.1	1564531496	0.97	0.85	A CDA is a mobile user device, similar to a Personal Digital Assistant (PDA). It supports the citizen when dealing with public authorities and proves his rights - if desired, even without revealing his identity ...
ST1_task1_1	0	G01	G01.1	3000234933	0.9	0.9	People are becoming increasingly comfortable using Digital Assistants (DAs) to interact with services or connected objects ...
ST1_task1_1	0	G01	G01.2	1448624402	0.6	0.3	As extensive experimental research has shown individuals suffer from diverse biases in decision-making ...

2.3 Evaluation Metrics

Passage relevance is assessed based on:

– manual relevance assessment of a pool of passages (relevance scores provided by participants is used to measure ranking quality)

In addition to topical relevance, additional aspects such as the text complexity and the credibility or importance of the retrieved results are key in the use-case of the track. Hence we provide additional analysis in terms of:

– readability level analysis of the retrieved results, providing an indication of the accessibility of the retrieved abstracts.
– manual assessment by non-expert users of credibility and complexity.

2.4 Participants' Approaches

Elsevier (*Elsevier*$*$ in the Table 3) [6] submitted a total of 10 runs to Task 1, exploring the effectiveness of a stream of neural rankers, both applied in a zero-shot way as well as with unsupervised fine-tuning on scientific documents.

University of Amsterdam (*UAms_*$*$) [16] submitted a total of 10 runs for Task 1. First, they submitted 3 baseline rankers to improve the pool of judgments: an elastic run using keyword (non-phrase) queries, and a cross-encoder reranking of the top 100 and top 1k results from Elastic. Second, they submitted 4 runs aiming to address the credibility of the retrieved results, taking into account the recency and number of citations of each paper. Third, they submitted 3 runs aiming to address the readability of the retrieved results.

University of Avignon (LIA_*) submitted a total of 7 runs to Task 1, using a range of lexical and neural ranking models. These runs were used to analyze pool diversity and reusability of the resulting test collection and to investigate the aggregation of several queries to their associated article.

University of Maine (AIIR Lab, maine_*) [19] submitted a total of 5 runs to Task 1, experimenting with several cross-encoders and bi-encoder models, in comparison to lexical models.

University of Milano Bicocca (unimibDoSSIER_*) [20] submitted a total of 2 runs to Task 1, with a range of domain-specific approaches for scientific documents, including probabilistic lexical ranking, hierarchical document classification, and pseudo-relevance feedback (PRF).

2.5 Results

Retrieval Effectiveness. Table 3 shows the results of the CLEF 2023 Simpletext Task 1, based on the 34 test queries. The main measure of the task is NDCG@10, and the table is sorted on this measure for convenience.

A number of observations stand out. First and foremost, we see in general that the top of the Table is dominated by neural rankers, in particular, cross-encoders trained on MSMarco applied in a zero-shot way (or variants thereof), perform well for ranking scientific abstracts on NDCG@10 and other early precision measures. Traditional lexical retrieval models perform reasonably but at some distance from the top-scoring runs, with the neural runs typically re-ranking such a lexical baseline run.

Second, looking at more recall-oriented measures, such as MAP and bpref, the picture is more mixed. This is indicating some approaches privilege precision over recall, whereas other approaches seem to promote all recall levels.

Third, some submissions aimed to balance the topical relevance with the readability or credibility of the results. We observe that these runs still achieve competitive retrieval effectiveness, despite removing or down-ranking highly relevant abstracts that have for example a high text complexity or are dated with low numbers of citations.

Analysis of Readability. Table 4 shows several statistics over to the top 10 results retrieved for the entire topic set for Task 1:

- citation analysis (impact factor based on ACM records and average number of references per document)
- textual analysis (document length and FKGL scores)

We make a number of observations.

First, it appears that the most effective ranking models tend to retrieve abstracts that are not only longer, but also exhibit greater length variability. These retrieved abstracts often have higher impact factors and extensive bibliographies. There also seems to be a discernible difference between the lengths of abstracts retrieved by lexical-based systems compared to those retrieved by neural-based systems.

Table 3. Evaluation of SimpleText Task 1 (Test qrels).

Run	MRR	Precision		NDCG		Bpref	MAP
		10	20	10	20		
ElsevierSimpleText_run8	0.8082	0.5618	0.3515	0.5881	0.4422	0.2371	0.1633
ElsevierSimpleText_run7	0.7136	0.5618	0.4103	0.5704	0.4627	0.2626	0.1915
maine_CrossEncoder1	0.7309	0.5265	0.4500	0.5455	0.4841	0.3337	0.2754
maine_CrossEncoderFinetuned1	0.7338	0.4971	0.4000	0.4859	0.4295	0.3443	0.2385
ElsevierSimpleText_run5	0.6600	0.4765	0.3838	0.4826	0.4186	0.2542	0.1828
ElsevierSimpleText_run2	0.7010	0.4676	0.4059	0.4791	0.4282	0.2528	0.1942
ElsevierSimpleText_run6	0.6402	0.4676	0.3853	0.4723	0.4185	0.2557	0.1809
ElsevierSimpleText_run4	0.6774	0.4529	0.3794	0.4721	0.4116	0.2485	0.1898
ElsevierSimpleText_run9	0.5933	0.4735	0.3176	0.4655	0.3595	0.1758	0.1238
ElsevierSimpleText_run1	0.6821	0.4588	0.3824	0.4626	0.4071	0.2573	0.1823
maine_CrossEncoderFinetuned2	0.7082	0.4706	0.3926	0.4617	0.4089	0.3259	0.2253
UAms_CE1k_Filter	0.6403	0.4765	0.3559	0.4533	0.3743	0.2727	0.1936
ElsevierSimpleText_run3	0.6502	0.4471	0.3779	0.4460	0.3994	0.2558	0.1785
UAms_ElF_Cred44	0.6888	0.4324	0.3338	0.4103	0.3499	0.2395	0.1719
UAms_CE100	0.6779	0.3971	0.3456	0.4016	0.3642	0.2658	0.1792
maine_Pl2TFIDF	0.5626	0.4176	0.2809	0.4014	0.3218	0.2155	0.1364
UAms_Elastic	0.6424	0.4059	0.3456	0.3910	0.3541	0.2501	0.1895
UAms_ElF_Cred53	0.6429	0.4088	0.3382	0.3883	0.3468	0.2454	0.1833
UAms_ElF_Cred44Read	0.6625	0.3971	0.3147	0.3723	0.3282	0.2123	0.1403
UAms_CE1k_Combine	0.5880	0.4147	0.3515	0.3706	0.3398	0.2700	0.1865
UAms_CE1k	0.5880	0.4147	0.3515	0.3706	0.3398	0.2700	0.1865
UAms_ElF_Read25	0.6076	0.3735	0.3074	0.3539	0.3190	0.2194	0.1522
UAms_ElF_Cred53Read	0.6088	0.3676	0.3059	0.3469	0.3153	0.2133	0.1456
maine_tripletloss	0.5502	0.3382	0.2176	0.3353	0.2561	0.1335	0.0696
unimib_DoSSIER_2	0.5201	0.2853	0.2515	0.2980	0.2683	0.1898	0.1141
unimib_DoSSIER_4	0.5202	0.2853	0.2441	0.2972	0.2632	0.1873	0.1111
run-LIA.bm25	0.4536	0.1912	0.1338	0.2192	0.1700	0.1384	0.0515
run-LIA.all-MiniLM-L6-v2.query	0.3505	0.2000	0.1662	0.2019	0.1767	0.1956	0.0667
run-LIA.all-MiniLM-L6-v2.query-topic	0.3655	0.1765	0.1485	0.1912	0.1647	0.2043	0.0591
run-LIA.all-mpnet-base-v2.query-topic	0.3506	0.1647	0.1294	0.1835	0.1517	0.2073	0.0523
run-LIA.all-mpnet-base-v2.query	0.3302	0.1647	0.1529	0.1802	0.1644	0.1956	0.0602
run-LIA.lda	0.3138	0.1824	0.1456	0.1666	0.1488	0.1402	0.0521
run-LIA.es	0.3056	0.1118	0.0912	0.1277	0.1080	0.1935	0.0342

Second, in terms of readability levels, the overwhelming majority of systems retrieve abstracts with an FKGL of around 14 – corresponding to university-level texts. This is entirely as expected since the corpus is based on scientific text, known to be written for experts with higher text complexity than for example newspaper articles.

Third, two systems retrieve abstracts with an FKGL of 11–12 – corresponding to the exit level of compulsory education, and the reading level of the average newspaper reader targeted by the use case of the track. These runs still achieved very reasonable retrieval effectiveness (NDCG@10 0.37–0.45 in Table 3) while only retrieving abstracts with the desirable readability level.

Table 4. Text Analysis of SimpleText Task 1 output.

Run	Impact	#Refs	Length		FKGL	
			Mean	Median	Mean	Median
ElsevierSimpleText_run1	1.88	0.95	965.02	921.00	13.80	13.80
ElsevierSimpleText_run2	2.24	1.36	1017.57	981.00	13.98	13.90
ElsevierSimpleText_run3	1.80	0.94	951.64	912.00	13.71	13.75
ElsevierSimpleText_run4	2.10	1.21	1011.10	994.00	13.95	13.90
ElsevierSimpleText_run5	1.78	0.71	993.14	972.50	13.76	13.80
ElsevierSimpleText_run6	1.59	0.65	995.65	975.50	13.75	13.90
ElsevierSimpleText_run7	2.37	0.94	1101.23	1075.50	13.87	13.80
ElsevierSimpleText_run8	0.60	0.50	1089.90	1045.00	14.09	14.00
ElsevierSimpleText_run9	0.71	0.54	1016.96	991.00	13.66	13.70
UAms_CE100	3.20	1.64	1028.78	975.00	14.59	14.50
UAms_CE1k	2.41	1.24	1071.67	985.50	14.70	14.60
UAms_CE1k_Combine	0.84	0.49	924.38	839.00	10.84	11.20
UAms_CE1k_Filter	1.09	0.62	988.00	913.50	12.40	12.70
UAms_ElF_Cred44	3.32	1.62	973.03	970.50	13.60	14.50
UAms_ElF_Cred44Read	1.85	1.34	799.29	851.00	13.18	14.20
UAms_ElF_Cred53	2.89	1.49	938.41	932.00	13.73	14.40
UAms_ElF_Cred53Read	1.70	1.28	774.76	823.00	13.29	14.30
UAms_ElF_Read25	1.60	1.25	767.70	819.00	13.09	14.20
UAms_Elastic	2.84	1.45	922.36	917.00	13.49	14.30
maine_CrossEncoder1	4.22	2.86	961.17	923.00	14.64	14.60
maine_CrossEncoderFinetuned1	4.41	3.37	1003.75	988.00	15.01	14.80
maine_CrossEncoderFinetuned2	3.49	3.04	988.86	951.50	14.95	14.80
maine_Pl2TFIDF	3.35	2.58	893.29	894.00	14.03	14.00
maine_tripletloss	4.76	3.29	969.09	973.50	14.69	14.60
unimib_DoSSIER_2	1.44	1.33	1024.48	994.00	14.77	14.60
unimib_DoSSIER_4	1.44	1.33	238.63	212.00	15.11	15.00

3 Task 2: What is Unclear?

The goal of this task is to identify key concepts that need to be contextualized with a definition, example, and/or use-case and provide useful and understandable explanations for them. Thus, there are two subtasks:

- to retrieve up to 5 difficult terms in a given passage from a scientific abstract
- to provide an explanation (one/two sentences) of these difficult terms (e.g. definition, abbreviation deciphering, example, etc.)

For each passage, participants should provide a ranked list of difficult terms with corresponding difficulty scores on a scale of 0–2 (2 to be the most difficult terms, while the meaning of terms scored 0 can be derived or guessed) and definitions (optional). Passages (sentences) are considered to be independent, i.e. difficult term repetition is allowed. Detected concept spans and term and term difficulty are evaluated.

3.1 Data

Datasets for Task 2.1. To build the **test set** for Task 2.1, 116,763 sentences from the DBLP abstracts were extracted. Then, a set of 1262 distinct sentences were manually evaluated to measure the performance of different models in terms of their ability in detecting difficult terms and their difficulty scores. A pooling mechanism is used to further annotate 5,142 distinct pairs sentence-term manually in which each evaluated source sentence contained the results of all participants.

Datasets for Task 2.2. A set of 203 difficult terms (within sentences from Task 1) with their ground truth annotations are provided in the training set for Task 2.2 for the definition generation part. For the evaluation of runs for this task, we use ∼800 terms with ground truth definitions. From this set, ∼300 terms are annotated using a pooling mechanism (based on the submitted runs) to make sure that the majority of runs have enough annotated samples in the test set. There are a total of 15,056 sentences containing at least one of these terms in our test set. For the abbreviation expansion evaluation, we manually annotate a set of ∼1K manually abbreviations. We additionally expand this dataset by mining 4,374 extra abbreviations from the sentences from Task 1. We use the Schwartz and Hearst [26] algorithm to extract these extra abbreviations and their expansion from the test set. There are 38,416 sentences in the test set containing at least one of the ∼5K abbreviations. We use this set of sentences for the final evaluation of this subtask.

Input Format. The train and the test data are provided in JSON and TSV formats with the following fields:

snt_id a unique passage (sentence) identifier
doc_id a unique source document identifier
query_id a query ID
query_text difficult terms should be extracted from sentences with regard to this query
source_snt passage text

Input example:

```
[{"query_id":"G14.2",
  "query_text":"end to end encryption",
  "doc_id":"2884788726",
  "snt_id":"G14.2_2884788726_2",
```

```
  "source_snt":"However, in information-centric networking (ICN)
   ↪ the end-to-end encryption makes the content caching
   ↪ ineffective since encrypted content stored in a cache is
   ↪ useless for any consumer except those who know the
   ↪ encryption key."},

 {"snt_id":"G06.2_2548923997_3",
  "doc_id":2548923997,
  "query_id":"G06.2",
  "query_text":"self driving",
  "source_snt":"These communication systems render self-driving
   ↪ vehicles vulnerable to many types of malicious attacks,
   ↪ such as Sybil attacks, Denial of Service (DoS), black
   ↪ hole, grey hole and wormhole attacks."}]
```


Output Format. Results should be provided in a TREC-style JSON or TSV format with the following fields:

run_id Run ID starting with (team_id)_(task_id)_(method_used), e.g. UBO_task_2.1_TFIDF
manual Whether the run is manual $\{0, 1\}$.
snt_id a unique passage (sentence) identifier from the input file.
term Term or another phrase to be explained.
term_rank_snt term difficulty rank within the given sentence.
difficulty difficulty scores of the retrieved term on the scale 0–2 (2 to be the most difficult terms, while the meaning of terms scored 0 can be derived or guessed)
definition (only used for Task 2.2) short (one/two sentence) explanations/definitions for the terms. For the abbreviations, the definition would be the extended abbreviation.

Output example Task 2.1:


```
[{"snt_id":"G14.2_2884788726_2",
  "term":"content caching",
  "difficulty":1.0,
  "term_rank_snt":1,
  "run_id":"team1_task_2.1_TFIDF",
  "manual":0}]
```


Output example Task 2.2:


```
[{"snt_id":"G14.2_2884788726_2",
  "term":"content caching",
  "difficulty":1.0,
  "term_rank_snt":1,
  "definition":"Content caching is a performance optimization
   ↪ mechanism in which data is delivered from the closest
   ↪ servers for optimal application performance.",
  "run_id":"team1_task_2.2_TFIDF_BLOOM",
  "manual":0}]
```

3.2 Evaluation Metrics

In this section, we describe different evaluation metrics used to evaluate the performance of submissions for Task 2.1 and Task 2.2.

Task 2.1. We have evaluated the performance of different submissions for Task 2.1 based on:

- correctness of detected term limits: this metric reflects whether the retrieved difficult terms are well limited or not. This is a binary label assigned to each retrieved term.
- difficulty scores: we used a three-scale terms difficulty score which reflects how difficult the term is in the context for an average user and how necessary it is to provide more context about the term: 0 score corresponds to an easy term (explanation might be given but not required); 1 corresponds to somewhat difficult (explanation could help); 2 corresponds to difficult (explanation is necessary).

Task 2.2. For this task, we use the following evaluation metrics:

- **BLEU** score [24] between the reference (ground truth definition) and the predicted definitions.
- **ROUGE L F-measure** [17] which measures the ROUGE F-measure based on the Longest Common Subsequence between the reference and the predicted definitions.
- **Semantic match** between the reference and predicted definitions measured using the *all-mpnet-base-v2*[5] sentence transformer model which is an advanced model for sentence similarity. This measure is the average semantic similarity between reference and predicted definitions for all detected terms.
- **Exact match** which is only used for the task of **abbreviation extension** in which we ask the participants to provide only extensions for the detected difficult abbreviations. This metric measures the number of exact matches between the reference and predicted extensions for abbreviations.
- **Partial match** which measures the number of non-identical abbreviation extensions (between reference and predicted extensions) which have a Levenshtein distance lower than 4 characters. This corresponds to slight variations (such as plural/non-plural) between reference and predicted abbreviation extensions.

3.3 Participants' Approaches

National Polytechnic Institute of Mexico (NLPalma) [23] submitted a total of 2 runs for Task 2, a single run for each of Task 2.1 and Task 2.2. They experimented with BLOOMZ to produce description-style prompts given by text input on a task and a binary classifier based on BERT-multilingual for term difficulty.

University of Amsterdam (UAms) [16] submitted a single run for Task 2 focusing on complexity spotting. Their approach aimed to demonstrate the relative effectiveness of simple and straightforward approaches, and made use of standard TF-IDF based term-weighting using the large test set as a source for within-domain term statistics.

[5] https://huggingface.co/sentence-transformers/all-mpnet-base-v2.

University of Cadiz/Split (Smroltra) [25] submitted a total of 20 runs for Task 2, with both 10 runs for Task 2.1 and 10 runs for Task 2.2. They experimented with a range of keyword extraction approaches (KeyBERT, RAKE, YAKE!, BLOOM, T5, TextRank) for the first task, and a Wikipedia extraction approach, BERT, and BLOOMZ for the second task.

University of Guayaquil/Jaén (SINAI) [22] submitted a total of 6 runs for Task 2, with 4 runs for Task 2.1 and 2 runs for Task 2.2. They investigated zero-shot and few-shot learning strategies over the auto-regressive model GPT-3, and in particular effective prompt engineering.

University of Kiel (TeamCAU) [4] submitted 6 runs for Task 2, based on three different large pre-trained language models (SimpleT5, AI21, and BLOOM). They made three and corresponding submissions to both Task 2.1 and 2.2, and also note the complexities of adapting models with limited train data.

University of Kiel/Split/Malta (MicroGerk) [7] submitted a total of 8 runs for Task 2, with 4 runs for Task 2.1 and 4 runs for Task 2.2. They experimented with a range of models (YAKE!, TextRank, BLOOM, GPT-3) for the first task, and a range of models (Wikipedia, SimpleT5, BLOOMZ, GPT-3) for the second task.

University of Southern Maine (Aiirlab) [19] submitted a total of 6 runs for Task 2, consisting 3 runs for Task 2.1 and 3 runs for Task 2.2. They experimented with keyword extraction approaches (YAKE!, KBIR) and IDF weighting for the first task, and definition detection in top-ranked documents based on a trained classifier.

University of Western Brittany (UBO) [8] submitted a total of 8 runs for Task 2, no less than 7 runs for Task 2.1 and a single run for Task 2.2. They experimented with a range of keyword extraction approaches (FirstPhrase, TF-IDF, YAKE!, TextRank, SingleRank, TopicRank, PositionRank) for the first task and a Wikipedia extraction approach for the second task.

University of Split (Croland) submitted a total of 4 runs for Task 2, specifically 2 runs for Task 2.1 and 2 runs for Task 2.2. They applied GPT-3 and TF-IDF for difficult term detection. They extracted definitions from Wikipedia and applied GPT-3 to generate explanations.

University of Liverpool (UOL-SRIS) submitted a single run for Task 2, specifically for Task 2.1 by applying KeyBERT.

University of Kiel/Cadiz/Gdansk (TheLangVerse) submitted a total of 2 runs for Task 2, a single run for both Task 2.1 and Task 2.2 using GPT-3.

3.4 Results

We evaluate the performance of the submissions separately for the difficult terms spotting (Task 2.1) and definition extraction/generation (Task 2.2) using separate test sets created per task. In this section, we describe the main results of different submissions per task.

Table 5. SimpleText Task 2.1: Results for the official runs

	Total	Evaluated		Score	
			+Limits		+Limits
SINAI_task_2.1_PRM_ZS_TASK2_1_V1	11081	1322	1185	556	507
UAms_Task_2_RareIDF	675090	1293	1145	309	241
SINAI_task_2.1_PRM_FS_TASK2_1_V1	10768	1235	1122	440	405
Smroltra_task_2.1_keyBERT_FKgrade	11099	1215	1061	379	341
Smroltra_task_2.1_keyBERT_F	11099	1215	1061	223	171
UOL-SRIS_2.1_KeyBERT	23757	1215	1061	0	0
MiCroGerk_task_2.1_TextRank	21516	1275	1002	482	391
Smroltra_task_2.1_TextRank_FKgrade	10056	1275	1002	456	363
SINAI_task_2.1_PRM_ZS_TASK2_1_V2	10952	1075	965	366	330
SINAI_task_2.1_PRM_FS_TASK2_1_V2	8836	1004	915	346	316
Smroltra_task_2.1_YAKE_D	11112	1576	905	627	422
MiCroGerk_task_2.1_YAKE	23790	1576	905	582	362
Smroltra_task_2.1_YAKE_Fscore	11112	1576	905	409	209
MiCroGerk_task_2.1_GPT-3	15892	968	889	487	459
UBO_task_2.1_FirstPhrases	14088	1032	831	210	161
UBO_task_2.1_PositionRank	13881	1071	825	237	181
UBO_task_2.1_SingleRank	14088	981	748	200	151
UBO_task_2.1_TfIdf	14340	1206	740	263	187
UBO_task_2.1_TextRank	14088	960	722	189	139
Smroltra_task_2.1_RAKE_AUI	10660	1016	713	378	288
Smroltra_task_2.1_RAKE_F	10660	1016	713	255	170
UBO_task_2.1_TopicRank	13912	824	663	174	144
UBO_task_2.1_YAKE	14337	1118	576	265	116
MiCroGerk_task_2.1_BLOOM	9600	608	535	235	218
Aiirlab_task_2.2_KBIR	4797	498	429	158	135
TeamCAU_task_2.1_ST5	2234	484	418	222	201
Smroltra_task_2.1_SimpleT5	2234	460	406	259	239
Smroltra_task_2.1_SimpleT5_COLEMAN_LIEAU	2234	460	406	168	152
TheLangVerse_task_2.2_openai-curie-finetuned	2234	445	391	0	0
ThePunDetectives_task_2.1_SimpleT5	152072	428	371	110	91
Aiirlab_task_2.2_YAKEIDF	4790	465	241	154	75
Aiirlab_task_2.2_YAKE	4790	486	234	169	78
TeamCAU_task_2.1_AI21	100	10	6	3	2
Smroltra_task_2.1_Bloom	100	4	2	1	1
TeamCAU_task_2.1_BLOOM	100	1	1	0	0

Task 2.1: Difficult Term Spotting. In this section, we describe the results of the submissions on Task 2.1. A total of 12 teams submitted runs for Task 2.1. There were in total 39 runs. Table 5 shows the results for different runs. We show the total

Table 6. SimpleText Task 2.2: Results for the official runs

Run	Evaluated	BLEU	ROUGE	Semantic
UBO_task_2.1_FirstPhrases_Wikipedia	393	29.73	0.41	0.80
Croland_task_2_PKE_Wiki	43	33.68	0.46	0.70
MiCroGerk_task_2.2_GPT-3_Wikipedia	932	26.38	0.41	0.75
Smroltra_task_2.2_Text_Wiki	547	17.59	0.33	0.75
Smroltra_task_2.2_RAKE_Wiki	337	16.95	0.32	0.74
Smroltra_task_2.2_YAKE_Wiki	436	16.94	0.32	0.73
TeamCAU_task_2.1_BLOOM	10	10.46	0.27	0.48
MiCroGerk_task_2.2_GPT-3_BLOOMZ	1,108	9.07	0.40	0.83
Smroltra_task_2.2_keyBERT_Wiki	302	8.60	0.23	0.69
MiCroGerk_task_2.2_GPT-3_GPT-3	1,108	7.73	0.38	0.83
NLPalma_task_2.2_BERT_BLOOMZ	537	7.22	0.39	0.76
Smroltra_task_2.2_Bloomz	23	7.15	0.30	0.69
TeamCAU_task_2.1_AI21	22	6.38	0.31	0.78
TheLangVerse_task_2.2_openai-curie-finetuned	444	5.03	0.25	0.74
Croland_task_2_GPT3	69	4.83	0.27	0.77
SINAI_task_2.1_PRM_FS_TASK2_2_V1	649	4.23	0.21	0.78
MiCroGerk_task_2.2_GPT-3_simpleT5	1,108	4.22	0.28	0.77
TeamCAU_task_2.1_ST5	379	3.33	0.20	0.60
Smroltra_task_2.2_SimpleT5	392	3.09	0.22	0.72
SINAI_task_2.1_PRM_ZS_TASK2_2_V1	649	3.08	0.19	0.69
Smroltra_task_2.2_keyBERT_dict	120	2.07	0.14	0.51
Smroltra_task_2.2_YAKE_WN	48	1.88	0.15	0.44
Aiirlab_task_2.2_KBIR	556	1.62	0.15	0.50
Smroltra_task_2.2_keyBERT_WN	328	1.33	0.14	0.45
Aiirlab_task_2.2_YAKEIDF	179	1.13	0.14	0.41
Aiirlab_task_2.2_YAKE	165	1.10	0.15	0.43
Smroltra_task_2.2_RAKE_WN	70	0.00	0.14	0.46

number of evaluated terms and the number of terms with correct term limits. We present results for correctly attributed scores regardless of the correctness of term limits and the number of correctly limited terms with correctly attributed scores (+Limits). The SINAI_task_2.1_PRM_ZS_TASK2_1_V1 run has the highest number of correctly detected terms and scores among all the runs for this task.

Participants used Large Language Models (LLMs) as well as unsupervised methods. We received many partial runs due to token constraints of LLMs or their execution time. We also observe that the results of the same methods depend heavily on implementation, fine-tuning, and/or used prompts. Results of difficult term detection by LLMs are comparable to RareIDF, TextRank and YAKE! Term difficulty scores assigned by models are quite different from the lay annotations.

Table 7. SimpleText Task 2.2: Results for the official runs on the abbreviation expansion task

Run	Evaluated	BLEU	ROUGE	Semantic	Exact	Partial
MiCroGerk_task_2.2_GPT-3_BLOOMZ	854	13.87	0.68	0.76	326	185
MiCroGerk_task_2.2_GPT-3_GPT-3	855	11.86	0.64	0.73	294	166
MiCroGerk_task_2.2_GPT-3_Wikipedia	855	4.68	0.43	0.60	205	109
MiCroGerk_task_2.2_GPT-3_Wikipedia	618	5.01	0.56	0.64	198	109
NLPalma_task_2.2_BERT_BLOOMZ	345	6.83	0.39	0.52	50	47
Smroltra_task_2.2_SimpleT5	185	0.00	0.12	0.39	8	7
TeamCAU_task_2.1_ST5	141	1.48	0.14	0.40	6	3
TheLangVerse_task_2.2_openai-curie-finetuned	204	1.60	0.14	0.42	1	2
SINAI_task_2.1_PRM_ZS_TASK2_2_V1	228	1.61	0.13	0.55	1	0
TeamCAU_task_2.1_AI21	10	1.87	0.14	0.38	0	0
SINAI_task_2.1_PRM_FS_TASK2_2_V1	228	1.35	0.10	0.53	0	0
UBO_task_2.1_FirstPhrases_Wikipedia	116	5.09	0.19	0.47	0	0
Aiirlab_task_2.2_KBIR	202	1.17	0.07	0.44	0	0
Smroltra_task_2.2_RAKE_Wiki	27	0.54	0.04	0.14	0	0
Smroltra_task_2.2_Bloomz	4	0	0.22	0.61	0	0
Aiirlab_task_2.2_YAKEIDF	19	0	0.10	0.40	0	0
Smroltra_task_2.2_keyBERT_WN	188	0	0.04	0.27	0	0
Smroltra_task_2.2_keyBERT_Wiki	163	0.21	0.02	0.13	0	0
Smroltra_task_2.2_keyBERT_dict	46	0	0.04	0.34	0	0
Smroltra_task_2.2_RAKE_WN	21	0	0.04	0.24	0	0
Smroltra_task_2.2_YAKE_WN	32	0	0.02	0.21	0	0
Smroltra_task_2.2_YAKE_Wiki	31	0	0.03	0.11	0	0
Smroltra_task_2.2_Text_Wiki	50	0	0.02	0.10	0	0
Aiirlab_task_2.2_YAKE	9	0	0.13	0.36	0	0
TeamCAU_task_2.1_BLOOM	3	0	0	0.14	0	0

Task 2.2: Difficult Term Explanation. For this task, 10 teams submitted 29 runs in total. The main results for Task 2.2 are shown in Table 6. The low number of evaluated sentences for most runs is due to the fact that most runs are done on a small set of sentences from the test set. The rest of the runs also achieved strong performance in terms of the semantic similarity of their provided definitions with the ground truth definitions. As the results show, UBO_task_2.1_-FirstPhrases_Wikipedia, Croland_task_2_PKE_Wiki, and MiCroGerk_task_2.2_GPT-3_-Wikipedia runs achieved a strong performance in terms of the BLEU score. This result shows that although these runs do not use the same set of words as the ground truth definitions to define difficult terms, they still provide an explanation for the terms that are semantically similar to the ground truth ones. The Wikipedia-based runs have the highest similarity with the ground truth definitions.

Table 7 shows the performance of the runs on the abbreviation expansion task. *MiCroGerk* run has the highest performance on this task. This best-performing model is

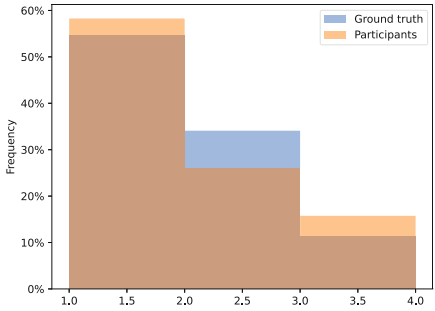

Fig. 1. Histogram of the difficulties of the definitions on a scale of 1–3 (1 - easy; 2 - difficult; 3 - very difficult)

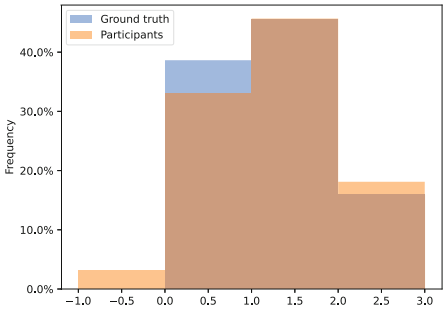

Fig. 2. Difference between term difficulty and definition difficulty on a scale of 1–3 (1 - easy; 2 - difficult; 3 - very difficult). Positive values on X axis show helpful definitions. 0 refers to unhelpful definitions. Negative values increase the difficulty.

able to provide an expansion for 326 identical expansions the true expansions and 185 partially correct expansions. In general, LLMs (BLOOMz, GPT-3) have the best performance for abbreviation expansion. Note, that the provided scores are averaged over the number of evaluated instances favoring small runs. Many partial runs are due to token/-time constraints of LLMs. Besides, evaluation results depend on the terms extracted in Task 2.1.

Analysis of Definitions' Difficulty. In order to analyze the helpfulness of the provided definitions, a master's student in translation and technical writing manually assigned scores of difficulty on a scale of 1–3 (1 - easy; 2 - difficult; 3 - very difficult) to 353 definitions for 82 distinct terms. The analyzed definitions are taken from participants' runs as well as from the ground truth.

Figure 1 shows the relative distribution of easy, difficult, and very difficult definitions in the participants' runs as well as in our ground truth. The figure provides evidence that in the majority of cases (more than 50% both in the runs and the ground truth), the definitions are considered by a non-expert in computer science to be easy.

In our ground truth, there are a slightly higher proportion of difficult definitions and a slightly lower proportion of very difficult definitions than in the participants' runs.

Although the majority of definitions are considered to be easy, this evidence is not enough to make a conclusion about their helpfulness. Therefore, we decided to compare the term difficulty and the corresponding definitions' difficulty. Figure 2 presents the histogram of the differences between term difficulty and definition difficulty. Positive values of the X axis show helpful definitions as the term difficulty is higher than the difficulty of the corresponding definition. 0 refers to an unhelpful definition as it has the same difficulty as the terms it should explain. Negative values on the X axis increase the difficulty, i.e. definition difficulty is higher than the difficulty of the corresponding term. The results suggest that 30%–40% of definitions are either unhelpful or even more difficult than the corresponding terms. Our ground truth does not have harmful definitions in contrast to the runs of the participants.

4 Task 3: Rewrite This!

This task aims to provide a simplified version of sentences extracted from scientific abstracts.

4.1 Data

As in 2022, we provide a parallel corpus of 648 manually simplified sentences as train data [12]. This year, we evaluated the submitted runs by comparing them against the new 245 manually simplified sentences extracted from relevant passages for Task 1.

Input Format. The train and the test data are provided in JSON and TSV formats with the following fields:

snt_id a unique passage (sentence) identifier
doc_id a unique source document identifier
query_id a query ID
query_text difficult terms should be extracted from sentences with regard to this query
source_snt passage text

Input example:

```
{"snt_id":"G11.1_2892036907_2",
 "source_snt":"With the ever increasing number of unmanned
  ↪ aerial vehicles getting involved in activities in the
  ↪ civilian and commercial domain, there is an increased need
  ↪ for autonomy in these systems too.",
 "doc_id":2892036907,
"query_id":"G11.1",
"query_text":"drones"}
```

Output Format. Results should be provided in a TREC-style JSON or TSV format with the following fields:

run_id Run ID starting with (team_id)_(task_3)_(method_used), e.g. UBO_BLOOM
manual Whether the run is manual $\{0, 1\}$.
snt_id a unique passage (sentence) identifier from the input file.
simplified_snt simplified passage .

Output example (JSON format):

```
{"run_id":"BTU_task_3_run1",
"manual":1,
"snt_id":"G11.1_2892036907_2",
"simplified_snt":"Drones are increasingly used in the civilian
 ↪  and commercial domain and need to be autonomous."}
```

4.2 Evaluation Metrics

To evaluate the simplification results, we used the EASSE implementation [2] of the following metrics:

- **FKGL:** Flesch-Kincaid Grade Level is a readability metric that relies on average sentence lengths and number of syllables per word [14];
- **SARI** metric compares the system's output to multiple simplification references and the original sentence based on the words added, deleted, and kept by a system [28];
- **BLEU** is a precision-oriented metric that relies on the proportion of shared n-gram in a system's output and references [24];
- **Compression ratio**;
- **Sentence splits**;
- **Levenshtein similarity** measures the number of edits (insertions, deletions, or substitutions) needed to transform one sentence into another;
- **Exact copies**;
- **Additions proportion**;
- **Deletions proportion**;
- **Lexical complexity score** computed by taking the log-ranks of each word in the frequency table [2].

4.3 Participants' Approaches

Chaoyang University of Technology (CYUT) [27] submitted four runs for Task 3, experimenting with the GPT-4 API provided by OpenAI. They experiment with three different prompts, even using GPT-4 to suggest better prompts for the task.

National Polytechnic Institute of Mexico (NLPalma) [23] submitted a single run for Task 3. They experimented with BLOOMZ with different prompts for generate text simplifications.

University of Amsterdam [16] submitted two runs (*UAms_**) for Task 3, using the zero-shot application of GPT-2 based text simplification model. Their approach aimed to address one of the main issues in text generation approaches, which are prone to 'hallucinate' and generate spurious content unwarranted by the input. Specifically, by post-processing the generated output to ensure grounding on input sentences, spurious generated output was identified and removed.

University of Applied Sciences, Cologne [9] submitted four runs (*irgc_**) for Task 3, with two runs using T5, one run using PEGASUS, and the final run exploiting ChatGPT. They perform detailed analysis

University of Cadiz/Split (Smroltra) [25] submitted a single run for Task 3. They experimented with a SimpleT5 model for text simplification.

University of Kiel [4] submitted a single run (*TeamCAU_**) for Task 3, based on the SimpleT5 pre-trained language model.

University of Kiel/Cadiz/Gdansk [21] submitted two runs for Task 3 (as *Pun Detective*). They used SimpleT5 and GPT-3 models under resource constrained conditions such as the limited task specific train data, and showed the SimpleT5 model outperforming GPT-3 in key metrics.

University of Kiel/Split/Malta (MicroGerk) [7] submitted a total of 3 runs for Task 3. They experimented with BLOOMZ, GPT-3, and SimpleT5 models for text simplification.

University of Southern Maine (AIIR Lab) [19] submitted a total of 2 runs for Task 3. They experimented with two models, a GPT-2 based model and an OpenAI DaVinci model for generating text simplifications.

University of Zurich (Andermatt) [3] submitted 6 runs (*Pandas_**) for Task 3, experimenting with four large pretrained language models: T5, Alpaca 5B, and Alpaca LoRA. They exploit Task 2 data as additional train data, and experiment with prompt engineering.

University of Zurich (Hou) [15] submitted three runs (*QH_**) for Task 3, adapting the Multilingual Unsupervised Sentence Simplification (MUSS) model to HuggingFace's BART, and using a T5-Large model. They experiment with a template consisting of 5 control tokens and also add the original request.

University of Kiel/Gdansk/Cadiz (TheLangVerse) submitted a single run for Task 3. They experimented with a finetuned OpenAI Curie model for text simplification.

University of Western Brittany (UBO) [8] submitted a single run for Task 3. They experimented with a SimpleT5 model (and with BLOOM) to generate simplifications.

Another team from the

University of Western Brittany (not in the Table) [5] experimented with ChatGPT for scientific text simplification, conducting a qualitative experiment with various analysis of the prompts and generated output.

Table 8. Results for task 3 (task number removed from the run_id)

run_id	count	FKGL	SARI	BLEU	Compression ratio	Sentence splits	Levenshtein similarity	Exact copies	Additions proportion	Deletions proportion	Lexical complexity score
Identity_baseline	245	13.64	15.09	26.22	1.00	1.00	1.00	1.00	0.00	0.00	8.64
AiirLab_task3_davinci	243	11.17	47.10	18.68	0.75	1.00	0.68	0.0	0.20	0.45	8.59
AiirLab_task3_run1	245	9.86	30.07	15.93	1.26	1.67	0.80	0.0	0.30	0.17	8.47
CYUT_run1	245	9.63	47.98	14.81	0.87	1.14	0.56	0.0	0.47	0.55	8.35
CYUT_run2	245	8.43	44.93	12.09	0.76	1.06	0.56	0.0	0.46	0.62	8.31
CYUT_run3	245	10.00	46.81	14.70	0.81	1.02	0.59	0.0	0.44	0.57	8.36
CYUT_run4	245	9.24	47.69	15.41	0.78	1.03	0.58	0.0	0.41	0.58	8.32
MiCroGerk_BLOOMZ	245	12.54	32.01	22.24	0.92	0.99	0.89	0.0	0.13	0.21	8.54
MiCroGerk_GPT-3	245	10.74	46.90	16.98	0.72	1.01	0.67	0.0	0.19	0.47	8.67
MiCroGerk_simpleT5	245	12.96	25.43	21.26	0.91	0.99	0.92	0.0	0.09	0.18	8.52
NLPalma_BLOOMZ	245	9.61	35.66	5.76	0.68	1.00	0.51	0.0	0.35	0.66	8.26
Pandas_alpaca-lora-alpaca-simplifier-alpaca-simplifier	245	10.96	38.31	17.88	0.74	1.00	0.77	0.0	0.10	0.36	8.51
Pandas_alpaca-lora-both-alpaca-normal-tripple	245	12.02	36.10	20.89	0.89	1.05	0.82	0.0	0.16	0.29	8.57
Pandas_alpaca-lora-both-alpaca-simplifier-tripple_10	244	11.71	36.38	19.62	0.89	1.07	0.78	0.0	0.16	0.31	8.55
Pandas_alpaca-lora-simplifier-alpaca-short	245	12.90	31.88	24.08	0.93	1.02	0.89	0.0	0.13	0.20	8.58
Pandas_clean-alpaca-lora-simplifier-alpaca-short	245	12.90	31.88	24.08	0.93	1.02	0.89	0.0	0.13	0.20	8.58
Pandas_submission_ensemble	245	10.51	40.25	17.40	0.77	1.09	0.73	0.0	0.15	0.40	8.52
QH_run1	245	12.45	26.46	21.23	0.94	1.07	0.92	0.0	0.11	0.17	8.50
QH_run2	245	13.05	24.40	21.33	0.96	1.03	0.92	0.0	0.12	0.15	8.48
QH_run3	245	12.74	27.56	20.24	0.90	1.01	0.91	0.0	0.09	0.19	8.50
Smroltra_SimpleT5	245	12.88	26.25	21.43	0.90	1.00	0.91	0.0	0.09	0.19	8.54
TeamCAU_ST5	245	12.77	27.19	21.06	0.90	1.00	0.91	0.0	0.10	0.20	8.52
TheLangVerse_openai-curie-finetuned	245	12.21	30.78	18.92	0.86	1.00	0.86	0.0	0.11	0.24	8.49
ThePunDetectives_GPT-3	245	7.52	41.56	6.10	0.46	0.97	0.50	0.0	0.16	0.68	8.46
ThePunDetectives_SimpleT5	245	12.92	25.87	21.79	0.91	0.99	0.92	0.0	0.09	0.18	8.53
UAms_Large_KIS150	245	10.50	33.02	14.59	1.26	1.48	0.76	0.0	0.34	0.20	8.45
UAms_Large_KIS150_Clip	245	11.12	33.47	16.59	1.01	1.23	0.82	0.0	0.24	0.23	8.48
UBO_SimpleT5	245	12.33	30.89	21.08	0.88	1.05	0.89	0.0	0.10	0.22	8.51
irgc_ChatGPT_2stepTurbo	245	12.31	46.98	16.86	0.94	1.04	0.63	0.0	0.37	0.46	8.46
irgc_pegasusTuner007plus_plus	245	12.74	23.28	17.42	1.23	1.28	0.83	0.0	0.22	0.15	8.55
irgc_t5	245	9.56	37.83	15.85	0.76	1.35	0.73	0.0	0.15	0.38	8.49
irgc_t5_noaron	245	9.55	37.84	15.84	0.76	1.35	0.73	0.0	0.15	0.38	8.49

4.4 Results

A total of 14 teams submitted 32 runs for Task 3, mainly LLMs. Table 8 presents the results of participants' runs according to the automatic evaluation listed in Section 4.2. Surprisingly, all systems modified the original sentences (Exact copies = 0). While many participants applied the same LLMs, such as GPT-3 and T5, their results differ a lot.

All runs improved the FKGL readability and lexical complexity scores with regard to the identity baseline (i.e. source sentences) suggesting that systems produced shorter sentences with simpler and shorter words on average. Note, that shorter words are not necessarily simpler as in the case of numerous abbreviations. Original sentences have an FKGL score of around 14 – corresponding to university-level texts. The majority of the submitted runs are scored lower than 11–12 according to FKGL – corresponding to the exit level of compulsory education.

All runs largely improved the SARI score compared to the original sentences. However, the source sentences have the highest vocabulary overlap with reference sentences.

Information Distortion. In order to analyze information distortion [12], a master student in translation and technical writing manually annotated 249 pairs of source sentences and simplifications submitted by the participants corresponding to 13 distinct source sentences. Sentences were assigned with binary labels corresponding to the

Table 9. Information distortion type statistics

Information distortion type	Instances	
	#	%
Incorrect syntax	9	3.61
Unresolved anaphora due to simplification	32	12.85
Unnecessary repetition/iteration	9	3.61
Spelling, typographic or punctuational errors	115	46.18
Contresens	18	7.22
Topic shift	3	1.20
Omission of essential details with regard to a query	45	18.07
Oversimplification	31	12.44
Insertion of false or unsupported information	8	3.21
Insertion of unnecessary details with regard to a query	3	1.20
Redundancy	3	1.20
Style	3	1.20
Non-sense	2	0.80

Table 10. Statistics on the levels of the difficulty of simplified sentences and information distortion severity on the scale of 1–7

	1	2	3	4	5	6	7
syntax complexity	230	19					
lexical complexity	54	114	62	19			
information loss severity	151	40	24	10	13	2	7
information loss severity %	60.64	16.06	9.63	4.01	5.22	0.80	2.81

occurrence of the information distortion types. Table 9 provides statistics on the information distortion identified in the participants' runs. The most common errors (46%) are spelling, typographic, and punctuational ones. It is followed by information loss (18%), unresolved anaphora due to simplification (13%), and oversimplification (12%). In 60% of cases, information loss was judged to be low (see Table 10).

4.5 Difficult Terms and Simplification

A master student in translation and technical writing manually assigned difficulty scores on a scale of 1–7 to the syntax and vocabulary of 249 simplified sentences from the

Table 11. Comparison of manually simplified and source sentences in Task 3

Metric (Avg)	Source snt	Simplified snt
FKGL	15.16	12.12
# Abbreviations	0.24	0.13
# Difficult terms	0.41	0.28

participants' runs corresponding to 13 distinct source sentences. Table 10 provides evidence that automatic simplification is effective in terms of reducing syntax difficulty. However, lexical difficulty, i.e. the presence of difficult scientific terms, is much higher, remaining the main barrier to understanding a scientific text.

In order to evaluate the quality of our train data (648 manually simplified sentences), we compared simplified and source sentences according to the following metrics:

- FKGL readability score that relies on average sentence lengths and number of syllables per word [14];
- Average number of abbreviations per sentence. The list of abbreviations was taken from Task 2.1.
- Average number of difficult terms per sentence. The list of difficult terms was constructed from the data used for the evaluation of Task 2.1.

Table 11 reports the scores of manually simplified and source sentences used in Task 3 according to these three metrics. The table provides evidence that our manual simplifications reduce text difficulty not only in terms of readability score, but our simplified sentences have more than 50% less difficult terms and abbreviations. These results also provide evidence that our tasks are closely interconnected.

5 Discussion and Conclusions

We introduced the CLEF 2023 SimpleText track, containing three interconnected shared tasks on scientific text simplification. Conceptually, we envisage a system pipeline retrieving relevant abstracts or passages for Task 1 (Content Selection); in order to detect difficult terms to be explained for Task 2 (Complexity Spotting); and simplify the ultimate selected sentences for Task 3 (Text Simplification). We evaluated the term difficulty, their explanations, and simplifications with regard to the queries from Task 1.

For Task 1, we created a large corpus of scientific abstracts, a set of popular science requests with detailed relevance judgments on the level of relevance of scientific abstracts to the request and the broader context of a newspaper article on this topic. The abstracts of scientific papers retrieved for these requests were used in the follow-up tasks. In 2023, we dramatically extended the qrels and introduced a additional evaluation measures that takes into account the complexity or credibility of the retrieved abstracts.

For Task 2 and 3, we created a corpus of sentences extracted from the abstracts of scientific publications, with manual annotations of term complexity and their definitions (Task 2). Our manual simplifications (Task 3) reduce text difficulty not only in terms of readability score but also have 50% less difficult terms and abbreviations than the source sentences. These results confirm the interconnection of the SimpleText tasks, and the value of researching their key dependencies.

We refer to the preceding sections for details of the different approaches to the tasks, and their effectiveness. A few general observations stand out. First, even when deploying similar models, the results of the same methods depend heavily on implementation, fine-tuning, and/or used prompts. Second, efficiency is of key importance in addition to effectiveness. We have received many partial runs due to token/time constraints of LLMs. Results of difficult term detection by LLMs are comparable to those of unsupervised methods. Third, robustness of the approaches remains challenging. Specifically, no less than 30%–40% of definitions are either unhelpful or even more difficult than the corresponding terms. Fourth, automatic simplification is effective in terms of reducing syntax difficulty and optimizing the FKGL score. However, lexical difficulty, i.e. the presence of difficult scientific terms, is much higher, remaining the main barrier to understanding a scientific text. The most common errors introduced in simplifications are spelling, typographic, and punctuational ones (46%), followed by information loss (18%), unresolved anaphora (13%), and oversimplification (12%).

So the general upshot of the CLEF 2023 SimpleText track is both that we observed great progress, but at the same time that there is also still a lot of room for improvements. In the future, we plan to classify difficult term explanations (definitions, examples, abbreviation deciphering etc.) and evaluate systems according to the usefulness and complexity of the provided explanations of scientific terms. We will further explore information distortion introduced by simplification.

Acknowledgments. *This research was funded, in whole or in part, by the French National Research Agency (ANR) under the project ANR-22-CE23-0019-01. We would like to thank Sarah Bertin, Radia Hannachi, Silvia Araújo, Pierre De Loor, Olga Popova, Diana Nurbakova, Quentin Dubreuil, Helen McCombie, Aurianne Damoy, Angelique Robert, and all other colleagues and participants who helped run this track.*

References

1. Aliannejadi, M., Faggioli, G., Ferro, N., Vlachos, M. (eds.): Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum. CEUR Workshop Proceedings. CEUR-WS.org (2023)
2. Alva-Manchego, F., Martin, L., Scarton, C., Specia, L.: EASSE: easier automatic sentence simplification evaluation. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations, pp. 49–54. Association for Computational Linguistics, Hong Kong (2019). https://doi.org/10.18653/v1/D19-3009, https://aclanthology.org/D19-3009
3. Andermatt, P.S., Fankhauser, T.: UZH_Pandas at SimpleTextCLEF-2023: alpaca LoRA 7B and LENS model selection for scientific literature simplification. In: [1] (2023)

4. Anjum, A., Lieberum, N.: Automatic simplification of scientific texts using pre-trained language models: a comparative study at CLEF symposium 2023. In: [1] (2023)
5. Bertin, S.: Scientific simplification, the limits of ChatGPT. In: [1] (2023)
6. Capari, A., Azarbonyad, H., Tsatsaronis, G., Afzal, Z.: Elsevier at simpletext: passage retrieval by fine-tuning GPL on scientific documents. In: [1] (2023)
7. Davari, D.R., Prnjak, A., Schmitt, K.: CLEF 2023 SimpleText task 2, 3: identification and simplification of difficult terms. In: [1] (2023)
8. Dubreuil, Q.: UBO team @ CLEF SimpleText 2023 track for task 2 and 3 - using IA models to simplify scientific texts. In: [1] (2023)
9. Engelmann, B., Haak, F., Kreutz, C.K., Nikzad-Khasmakhi, N., Schaer, P.: Text simplification of scientific texts for non-expert readers. In: [1] (2023)
10. Ermakova, L., et al.: Overview of SimpleText 2021 - CLEF workshop on text simplification for scientific information access. In: Candan, K.S., et al. (eds.) CLEF 2021. LNCS, vol. 12880, pp. 432–449. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-85251-1_27
11. Ermakova, L., SanJuan, E., Huet, S., Augereau, O., Azarbonyad, H., Kamps, J.: CLEF 2023 simpletext track - what happens if general users search scientific texts? In: Kamps, J., et al. (eds.) ECIR 2023. LNCS, vol. 13982, pp. 536–545. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-28241-6_62
12. Ermakova, L., et al.: Overview of the CLEF 2022 simpletext lab: automatic simplification of scientific texts. In: Barrón-Cedeño, A., et al. (eds.) CLEF 2022. LNCS, vol. 13390, pp. 470–494. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-13643-6_28
13. Ermakova, L.N., Nurbakova, D., Ovchinnikova, I.: COVID or not COVID? Topic shift in information cascades on Twitter. In: Linguistics, A.F.C. (ed.) 3rd International Workshop on Rumours and Deception in Social Media (RDSM) Collocated with COLING 2020. Proceedings of the 3rd International Workshop on Rumours and Deception in Social Media (RDSM), Barcelona (On line), Spain, pp. 32–37 (2020). https://hal.archives-ouvertes.fr/hal-03066857
14. Flesch, R.: A new readability yardstick. J. Appl. Psychol. **32**(3), 221–233 (1948). ISSN 0021-9010
15. Hou, R., Qin, X.: An evaluation of MUSS and T5 models in scientific sentence simplification: a comparative study. In: [1] (2023)
16. Hutter, R., Sutmuller, J., Adib, M., Rau, D., Kamps, J.: University of Amsterdam at the CLEF 2023 SimpleText track. In: [1] (2023)
17. Lin, C.Y., Hovy, E.: Automatic evaluation of summaries using N-gram co-occurrence statistics. In: Proceedings of the 2003 Conference of the North American Chapter of the ACL on Human Language Technology, vol. 1, pp. 71–78. ACL (2003)
18. Maddela, M., Alva-Manchego, F., Xu, W.: Controllable text simplification with explicit paraphrasing (2021). http://arxiv.org/abs/2010.11004
19. Mansouri, B., Durgin, S., Franklin, S., Fletcher, S., Campos, R.: AIIR and LIAAD labs systems for CLEF 2023 SimpleText. In: [1] (2023)
20. Mendoza, O.E., Pasi, G.: Domain context-centered retrieval for the content selection task in the simplification of scientific literature. In: [1] (2023)
21. Ohnesorge, F., Gutierrez, M.A., Plichta, J.: Scientific text simplification and general audience. In: [1] (2023)
22. Ortiz-Zambrano, J.A., Espin-Riofrio, C., Montejo-Ráez, A.: SINAI participation in SimpleText task 2 at CLEF 2023: GPT-3 in lexical complexity prediction for general audience. In: [1] (2023)
23. Palma, V.M., Preciado, C.P., Sidorov, G.: NLPalma @ CLEF 2023 SimpleText: BLOOMZ and BERT for complexity and simplification task. In: [1] (2023)
24. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: BLEU: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting on ACL, pp. 311–318. ACL (2002)

25. Dadić, P., Popova,O.: CLEF 2023 SimpleText tasks 2 and 3: enhancing language comprehension: addressing difficult concepts and simplifying ccientific texts using GPT, BLOOM, KeyBert, simple T5 and more. In: [1] (2023)
26. Schwartz, A.S., Hearst, M.A.: A simple algorithm for identifying abbreviation definitions in biomedical text. In: Biocomputing 2003, pp. 451–462 (2002)
27. Wu, S.H., Huang, H.Y.: A prompt engineering approach to scientific text simplification: CYUT at SimpleText2023 task3. In: [1] (2023)
28. Xu, W., Napoles, C., Pavlick, E., Chen, Q., Callison-Burch, C.: Optimizing statistical machine translation for text simplification. Trans. ACL **4**, 401–415 (2016)