University of Amsterdam at the CLEF 2024 SimpleText Track

Jan Bakker, Göksenin Yüksel and Jaap Kamps

University of Amsterdam, Amsterdam, The Netherlands

Abstract

This paper reports on the University of Amsterdam's participation in the CLEF 2024 SimpleText track. Our overall goal is to investigate and remove barriers that prevent the general public from accessing scientific literature, hoping to promote science literacy among the general public. Our specific focus is to investigate the relation between the *topical relevance* and the *text complexity* of scientific text, as well as develop *text simplification* approaches for scientific text. Our main findings are the following. First, for lay person scientific passage retrieval, both lexical and zero-shot retrieval models perform well, with only marginal loss of performance for complexity-aware models avoiding the retrieval of passages with low readability. Second, for spotting complex concepts, relative simple approaches based on corpus statistics show competitive precision but low recall. Third, for scientific text simplification different models generate different simplifications with all reasonable overlap with human reference simplifications. Fourth, document or abstract level text simplification incorporate discourse structure and make sentence deletions, which hold great promise to improve the output quality and succinctness for lay users of scientific text.

Keywords

Information Storage and Retrieval, Natural Language Processing, Wordplay translation, Humor retrieval, Humor classification

1. Introduction

While the advent of the internet and social media has given us access to an unprecedented volume of information, it also comes with unprecedented risks due to potential misinformation and disinformation spreading easily. The traditional antidote against misinformation is scientifically grounded information, and everyone agrees on the value and importance of science literacy. In practice, lay persons avoid consulting scientific sources, due to its presumed complexity. Hence, removing any access barriers for lay persons to consult scientific text are of paramount importance.

The CLEF 2024 SimpleText track investigates the barriers that ordinary citizens face when accessing scientific literature head-on, by making available corpora and tasks to address different aspects of the problem. For details on the exact track setup, we refer to the Track Overview paper CLEF 2024 LNCS proceedings [1] as well as the detailed task overviews in the CEUR proceedings [2, 3, 4, 5].

We conduct an extensive analysis of the three tasks of the track: Task 1 on *Content Selection*; Task 2 on *Complexity Spotting*; and Task 3 on *Text Simplification*. We submitted in total seven runs for Task 1, focussing both on retrieval effectiveness for popular requests, as well as on the text complexity of the retrieved abstracts. We submitted three baseline runs for Task 2, focusing on straightforward locating rare terms, and on matching scientific text to definitions of terminology. We submitted ten runs for Task 3, exploring three different text simplification models (GPT-2, Wiki, Cochrone) and three levels of simplification (sentence, paragraph, document or abstract).

The rest of this paper is structured as follows. Next, in Section 2 we discuss our experimental setup and the specific runs submitted. Section 3 discusses the results of our runs and provides a detailed analysis of the corpus and results for each tasks. We end in Section 4 by discussing our results and outlining the lessons learned.

CLEF 2024: Conference and Labs of the Evaluation Forum, September 9–12, 2024, Grenoble, France

kamps@uva.nl (J. Kamps)

0000-0002-6614-0087 (J. Kamps)

2. Experimental Setup

In this section, we will detail our approach for the three CLEF 2024 SimpleText track tasks.

2.1. Experimental Data

For details of the exact task setup and results we refer the reader to the detailed overview of the track in [6]. The basic ingredients of the track are:

Corpus The CLEF 2024 SimpleTrack Corpus consists of 4.9 million bibliographic records, including 4.2 million abstracts, and detailed information about authors/affiliations/citations.

Context There are 40 popular science articles, with 20 from *The Guardian*[1] and 20 from *Tech Xplore*.[2]

Requests For Task 1, there are 176 requests, 109 requests are based on The Guardian and 67 on TechXplore. Abstracts retrieved for these requests form the corpus for the remaining Tasks 2 and 3. This expands the topic set with 1-4 word queries using earlier years with 64 verbose questions on the Guardian articles.

Train Data For Task 1, there are relevance judgments for 64 requests (corresponding to 20 Guardian articles, G01–G20, and 5 TechExplore articles, T01–T05), with 61 queries having 10 or more relevant abstracts.

For Task 2, there are 576 train sentences with ground truth on complex terms/concepts for a total of 2,579 terms, and 317 test sentences (4.5 per query). For Task 2.3, an additional set of 3,815 other sentences is provided.

For Task 3, there are 958 train sentences with human simplifications, matching to 175 train abstracts with human simplifications. There are 4,797 test sentences, and a matching set of 182 test abstracts.

Test Data For Task 1, the ultimate test collection consists of 30 queries G1.C1–G10.C1 (10 on the Guardian), T06–T11 (20 on Tech Xplore). with a total of 4,854 judgments (128.5 per query). All 30 queries have 29 or more relevant abstracts.

For Task 2, there are 313 test sentences with ground truth on complex terms/concepts for a total of 1,440 terms (4.6 per query).

For Task 3, there are 578 test sentences with human simplifications, matching to 103 test abstracts with human simplifications.

2.2. Official Submissions

We created runs for all the three tasks of the track, which we will discuss in order.

Task 1 *This task asks to retrieve passages to include in a simplified summary.*

We submitted six runs in total, shown in Table 1. We first submitted four baseline runs focusing on regular information retrieval effectiveness. Two are vanilla baseline runs on an Anserini index, using either BM25 or BM25+RM3 with default settings [7].[3] The other two runs are neural cross-encoder rerankings of these runs, based on zero-shot application of an MSMARCO trained ranker, reranking the top 100 of either the BM25 or the BM25+RM3 baseline run.[4]

We submitted two further runs that filter for median FKGL in the runs, both for the top 100 and top 1K crossencoder reranker, following the *Complexity Aware Ranking* approach of [8]. These runs

[1] https://www.theguardian.com/science
[2] https://techxplore.com/
[3] https://github.com/castorini/pyserini
[4] https://huggingface.co/cross-encoder/ms-marco-MiniLM-L-12-v2

Table 1
CLEF 2024 SimpleText Track Submissions

Task	Run	Description
1	UAms_Task1_Anserini_bm25	BM25 baseline (Anserini, stemming)
1	UAms_Task1_Anserini_rm3	RM3 baseline (Anserini, stemming)
1	UAms_Task1_CE100	Cross-encoder top 100
1	UAms_Task1_CE1K	Cross-encoder top 1,000
1	UAms_Task1_CE100_CAR	Cross-encoder top 100 + Complexity filter
1	UAms_Task1_CE1K_CAR	Cross-encoder top 1,000 + Complexity filter
2.1	UAms_Task2-1_RareIDF	Up to 5 rarest terms on idf from test-large 2023
2.3	UAms_Task2-3_Anserini_bm25	BM25 baseline (Anserini, stemming)
2.3	UAms_Task2-3_Anserini_rm3	RM3 baseline (Anserini, stemming)
3.1	UAms_Task3-1_GPT2	GPT-2 Sentence level
3.1	UAms_Task3-1_GPT2_Check	GPT-2 Sentence level, Source checked
3.2	UAms_Task3-2_GPT2_Check_Snt	GPT-2 Sentence level, Source checked, merged into abstracts
3.2	UAms_Task3-2_GPT2_Check_Abs	GPT-2 Abstract level, Source checked
3.1	UAms_Task3-1_Wiki_BART_Snt	Wikiauto trained BART sentence level simplification
3.1	UAms_Task3-1_Cochrane_BART_Snt	Cochrane trained BART sentence level simplification
3.2	UAms_Task3-2_Wiki_BART_Par	Wikiauto trained BART paragraph level simplification
3.2	UAms_Task3-2_Cochrane_BART_Par	Cochrane trained BART paragraph level simplification
3.2	UAms_Task3-2_Wiki_BART_Doc	Wikiauto trained BART document level simplification
3.2	UAms_Task3-2_Cochrane_BART_Doc	Cochrane trained BART document level simplification

simply filters out the most complex abstract per request, using a standard readability measure. The run is aiming to remove up to 50% of the results, with the remaining abstracts in the same relevance order as in the original run.

As the train data is limited, and none of the approaches above are specific to scientific text, we also experimented with domain adaptation approaches in post-submission experiments.

Task 2 *This task asks to identify and explain difficult concepts.*

We submitted three runs, also shown in Table 1. For Task 2.1 on complexity spotting, we submitted a single run. As sentences have a limited number of words, we observed that naive baseline approaches can obtain reasonable performance already. Hence, our submission is using an idf-based term weighting to locate the most rare terms. Specifically, we used all train and test sentences combined as a reference corpus to calculate document (or rather sentence) frequencies, and use this to rank each term in the source sentence by increasing DF (or decreasing IDF).

For Task 2.3, we developed an approach to rank definitions or explanations for a given sentence and term pair. However the provided test data did provide only unmatched sets of scientific sentences and other sentences. Hence we submitted two runs only looking at the textual similarity of the large set of provided 'other' sentences.

Task 3 *This task asks to simplify scientific text.*

We submitted the twelve runs shown in Table 1. Our first set of experiments continues the earlier experiments with a GPT-2 model trained in an unsupervised way. First, we use the basic pretrained model on sentence level input. Second, we check all output against the source to avoid hallucination, and submit this checked version. Third, we merge the sentence level simplifications to create abstract level simplifications. Fourth, we run the model on long abstract level input, to create direct abstract level simplifications. All these four runs use the exact same GPT-2 text simplification model.

Our second set of experiments is with different BART trained models, either trained on Wiki-Auto or on aligned Lay Summaries from Cochrane (a home grown Cochrane-Auto). This leads to six runs, using either Wiki or Cochrane train data, and using either sentence level, paragraph level, or document

Table 2
Evaluation of SimpleText Task 1 (train data).

Run	MRR	Precision			NDCG			Bpref	MAP
		5	10	20	5	10	20		
UAms_Task1_Anserini_bm25	0.6503	0.4688	0.3906	0.2818	0.4468	0.3931	0.3405	0.4198	0.2439
UAms_Task1_Anserini_rm3	0.6043	0.4187	0.3609	0.2677	0.4003	0.3581	0.3220	0.4157	0.2297
UAms_Task1_CE100	0.6655	0.4813	0.4312	0.3214	0.4570	0.4206	0.3811	0.3275	0.2235
UAms_Task1_CE1K	0.6603	0.4531	0.4078	0.3089	0.4304	0.3998	0.3668	0.4299	0.2484
UAms_Task1_CE100_CAR	0.6709	0.4687	0.3937	0.2396	0.4530	0.3972	0.3163	0.3144	0.1922
UAms_Task1_CE1K_CAR	0.6403	0.4219	0.3672	0.2484	0.4032	0.3646	0.3092	0.3411	0.1904
GPL Base†	0.3301	0.1594	0.1719	0.1562	0.1560	0.1625	0.1708	0.3945	0.1062
GPL Domain Adapt†	0.4478	0.2719	0.2453	0.1958	0.2530	0.2380	0.2286	0.4012	0.1469
GPL Domain Adapt Remining†	0.5459	0.3125	0.2953	0.2141	0.3034	0.2874	0.2519	0.3978	0.1613

† Post-submission experiment.

(abstract) level input. Each of these six runs uses a different model, due to the different train input matching the output settings.

3. Experimental Results

In this section, we will present the results of our experiments, in three self-contained subsections following the CLEF 2024 SimpleText Track tasks.

3.1. Task 1: Content Selection

We discuss our results for Task 1, asking to retrieve passages to include in a simplified summary.

3.1.1. Retrieval effectiveness

Table 2 shows the performance of the Task 1 submissions on the train data. Let us first observe how different our runs are from the pooled runs, as those were based exclusively on the organizer's provided Elastic Search index and the particular keyword query. Due to the different tokenization and indexing choices in our Anserini index, the fraction of unjudged documents in the top 10s is high. First, the BM25 run has 36.6% and the BM25+RM3 run has 41.6% unjudged in the top 10. Second, the cross-encoder reranking has 27.5% (CE top 100) and 30.8% (CE top 1K) of unjudged, slightly lower due to similar neural reranker contributing to the pool in earlier years. Third, the complexity-aware filtered runs have 34.4% (CAR top 100) and 35.3% (CAR top 1K). Fourth, the domain adapted runs have no less than 50.9–72.2% unjudged in the top 10. In this light, the scores of the train adapted run on the train data are truly impressive.

We make a number of observations on the performance on the train set. First, the two Anserini baselines using BM25 with or without RM3 query expansion perform very reasonable with an NDCG@10 of 0.36-0.39 on the train data. The RM3 models underperforms the vanilla BM25 on all measures for train, but has a higher fraction of unjudged documents. The used Anserini index differs from the organizer's provided Elastic search index that dominates the pool of the train data. Second, the zero-shot reranking with an crossencoder lead to an improvement of retrieval effectiveness over the BM25 first stage ranker, with the top 100 reranking scoring 0.42 NDCG@10 on train. The bpref measure is less sensitive to pooling bias, and the highest bpref score of the top 1K reranking demonstrates the effectiveness of these runs. Third, we observe a favorable outcome for the domain adaptation of the models. The base scores are lower than GPL domain adaptation, and our novel remining strategy for continuous domain adaptation improves over GPL, the state-of-the-art for domain adaptation.

Table 3
Evaluation of SimpleText Task 1 (test data).

Run	MRR	Precision			NDCG			Bpref	MAP
		5	10	20	5	10	20		
UAms_Task1_Anserini	0.7187	0.5600	0.5500	0.4078	0.3867	0.3750	0.3507	0.3994	0.1973
UAms_Task1_Anserini_rm3	0.7878	0.5933	0.5700	0.3611	0.4039	0.3924	0.3282	0.4010	0.1824
UAms_Task1_CE100	0.6618	0.4800	0.5300	0.4044	0.3419	0.3654	0.3452	0.2657	0.1579
UAms_Task1_CE1K	0.5950	0.5133	0.5333	0.4033	0.3571	0.3672	0.3505	0.4031	0.1939
UAms_Task1_CE100_CAR	0.6420	0.5333	0.4700	0.3133	0.3435	0.3199	0.2741	0.2657	0.1321
UAms_Task1_CE1K_CAR	0.6611	0.5467	0.5133	0.2911	0.3800	0.3603	0.2778	0.2676	0.1348
GPL Base†	0.3752	0.2333	0.2100	0.1611	0.1823	0.1642	0.1465	0.3192	0.0654
GPL Domain Adapt†	0.5169	0.2733	0.2667	0.2233	0.2389	0.2240	0.2075	0.3600	0.0983
GPL Domain Adapt Remining†	0.5011	0.3133	0.3033	0.2467	0.2560	0.2412	0.2285	0.3732	0.1084

† Post-submission experiment.

Table 3 shows the performance of the Task 1 submissions on the train data. We submitted four runs focusing purely on standard retrieval effectiveness, and two runs addressing text complexity. On the test data, our submission were pooled, except for the combined score runs: we oberve 7.7% (CAR top 100) and 6.0% (CAR top 1K) of unjudged documents in the top 10 of each submission. Also the domain adapted runs have no less than 39.0–60.0% unjudged in the top 10, as they were not pooled.

We make a number of observations. First, we observer again that the two Anserini baselines using BM25 with or without RM3 query expansion perform very reasonable with an NDCG@10 of 0.38-0.39 on the test data. The RM3 models now outperforms the vanilla BM25 on all measures except MAP for test.

Second, the zero-shot reranking with an crossencoder does not lead to an improvement of retrieval effectiveness over the BM25 first stage ranker on the test data. Again, the bpref measure is less sensitive to pooling bias, and the highest bpref score of the top 1K reranking demonstrates the effectiveness of these runs.

Third, the complexity aware ranking runs filtering out the most complex abstract show competitive performance. Although these runs intentionally avoid complex, but topically relevant, results, they obtain higher precision scores and similar NCDG scores, and are almost on par with the runs retrieving complex results.

Fourth, recall that the domain adapted runs have not contributed to the pool and have high fractions of unjudged documents (no less than 39.0–60.0% unjudged in the top 10). In this light, again, the scores of the domain adapted runs are quite impressive. We observe again the relative score increase from base ranking, to standard GPL domain adaptation, and the GPL remining approach. We observe again that our novel remining strategy for continuous domain adaptation improves over GPL, the state-of-the-art for domain adaptation.

3.1.2. Analysis

This section analyzes various aspects of the submitted runs, where we pay particular attention to two aspects of core interest to the task and the overall use case of the track in which a lay user is accessing complex scientific text.

Credibility The first aspect of interest is the credibility of the retrieved information. Whilst one may assume that any scientific paper submitted after peer-review has passed a number of quality control steps during the peer-review process, and hence all retrieved abstract have high credibility. However, it is well-known that lay users have difficulty separating authoritative verses non-authoritative publications, as they are not able to discern the same cue as expert. For example, they are unaware of the reputation of the authors [9]. How authoritative are the results retrieved for our lay user?

Table 4
Analysis of SimpleText Task 1 output (over all 176 queries)

Run	Queries	Top	Year		Citations		Length		FKGL	
			Avg	Med	Avg	Med	Avg	Med	Avg	Med
UAms_Anserini_bm25	176	10	2012.9	2015	16.5	3.0	1355.9	1249.0	14.5	14.3
UAms_Anserini_rm3	176	10	2013.2	2015	16.8	3.0	1376.6	1272.5	14.5	14.4
UAms_CE100	176	10	2012.6	2015	20.5	3.0	1192.5	1115.0	14.5	14.4
UAms_CE100_CAR	176	10	2012.6	2015	18.0	3.0	1151.4	1081.0	12.5	12.8
UAms_CE1K	176	10	2012.5	2015	19.4	3.0	1147.0	1061.0	14.5	14.4
UAms_CE1K_CAR	176	10	2012.3	2015	18.5	3.0	1083.2	1009.0	12.4	12.7
GPL Base	176	10	2011.8	2014	13.1	2.0	910.5	970.5	14.3	14.3
GPL Domain Adapt	176	10	2011.9	2014	13.7	2.0	970.3	971.5	14.3	14.2
GPL Domain Adapt Remining	176	10	2011.7	2014	21.3	2.0	953.9	980.0	14.2	14.2

Table 4 shows the year of publication of the top 10 results retrieved for our lay user's popular science query. The systems retrieve publications with a median recency of 2015, ensuring that our lay user is consulting recent information not yet outdated or revised by more recent publications. This is an encouraging result as standard bibliometric literature ranking approaches have a strong bias for older publications given the fact that citations accumulate over time. But does this mean the results are not noteworthy and lack importance?

Table 4 also shows the number of citations of the top 10 results retrieved. We observe that our approach is retrieving results with significantly higher average numbers of citations, when compared to the baseline lexical rankers, with a gain from 17 to 21 citations on average. The GPL runs use a different baseline, but the difference between standard GPL, similar to the non-adapted baseline, and the novel remining approach is striking and obtains the highest average citation score. This higher average citation count is reassuring as it signals high levels of authoritativeness of the retrieved results. As citations are sparse and skewed, the median number of citations is only 2-3 throughout. This also signals that our approach is able to attract very highly cited publications into the top 10 results, leading to the significant average increase.

Readability The second aspect of interest is the readability of the retrieved information. We have seen above that the approaches are effective for retrieving relevant scientific papers. However, although topically relevant these paper may contain very advance scientific information that is not easy to understand and interpret by lay users. Recall that this was the motivation to use complexity-aware retrieval approaches [8]. Can complexity-aware search help retrieve relevant and accessible scientific text?

Table 4 shows the Flesch-Kincaid Grade Level (FKGL) readability score of the top 10 results retrieved for our lay user's popular science query. We observe that the lexical and neural rankers retrieve topically relevant information without taking the text complexity into account. Both lexical and neural rankers retrieve information with an FKGL of 14-15 corresponding to university level text complexity. The same holds for the domain adapted runs. This is not surprizing as we have an extensive scientific corpus with an average text complexity of 14-15 reflecting this.

Earlier we observed that our complexity-aware retrieval systems obtained almost almost the same effectiveness in terms of retrieval effectiveness. Hence this complexity aware approach was able to rank a similar number of topically relevant documents in the top 10 as standard lexical and neural ranking approaches. But is the complexity-aware approach able to rank more accessible content for our lay user issuing a popular science query?

Table 4 shows indeed favorable readability levels for the complexity aware search, with an FKGL of 12-13 corresponding to the exit level of compulsory education. Hence the complexity aware search approach is able to retrieve relevant and accessible content to our lay user. The retrieved source abstracts have a similar readability level as targeted by text simplification systems as discussed in Section 3.3.

Table 5
Evaluation of SimpleText Task 2 (test data).

Run	Recall		Terms "d"	
	Overall	Average	Recall	Precision
UAms_Task2-1_RareIDF	0.0854	0.0942	0.0259	0.0894

Table 6
Evaluation of SimpleText Task 2: submission UAms_Task2-1_RareIDF, only unique terms in the train (including validation) and test data.

Run	Precision					Recall					F1 Score				
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5
Train	0.16	0.14	0.13	0.13	0.12	0.04	0.07	0.10	0.13	0.15	0.06	0.09	0.11	0.11	0.12
Test	0.18	0.16	0.14	0.13	0.12	0.05	0.08	0.10	0.12	0.14	0.07	0.10	0.11	0.12	0.12

3.2. Task 2: Complexity Spotting

We continue with Task 2, asking to identify and explain difficult concepts.

3.2.1. Results

Task 2.1 Table 5 shows the performance of the Task 2 submission on the test data. At the time of writing, these score were released as (preliminary) scores without much further explanation.

The official results seem to focus entirely on recall aspects, or retrieving all terms annotated by the experts. Our simple approach is not expected to do well in terms of recall. We will conduct a more precision oriented evaluation below as additional analysis.

Task 2.3 There is no train data for Task 2.3 released, nor any test results made available at the time of writing. We hope and expect that these results will be released in time for the CLEF conferences in Grenoble.

3.2.2. Analysis

Table 6 shows the performance of the Task 2 submission on the train and test data. Due to the very limited data available, we treat spot here any terms. We included the complexity level as graded score, in order to filter the Boolean measures on minimal relevance score.[5] On the train and test data of earlier years, performance peaked around spotting 3 terms per sentence. Due to the many experts annotating the same set of sentences, we see that both recall and F1 increase over ranks and the highest scores are retrieved for spotting 5 rare terms per sentences. Overall, our simple approach achieves an MRR of 0.2542 (train) and 0.2741 (test) and, taking the difficulty level into account, an NDCG@5 of 0.1446 (train) and 0.1469 (test).

Table 7 shows an example sentence with references. In this example, our approach predicts 5 terms, that match one of the annotated references. The top ranked candidate matches one of the references annotated as difficult ("d"). There is a striking number of 16 references, with about 11 unique reference terms. Some references occur in variants (e.g., "simulated F1 car" is rated "d", whereas "F1 car" is rated "e"). Several references do not literally occur in the source sentence: we observe differences in case ("ResNet-18" vs. "resnet-18), plural/singular ("labels" vs. "label", "images" vs. "image"), and verb tense ("is fed" vs. "to be fed", "outputs" vs. "to output").

Table 8 shows the frequency of spotted terms on the train data. We observe a striking variation with 53 sentences having 1 complex terms, and 12 sentences having more than 15 complex terms. This

[5]Tables not shown as they exhibit the same qualitative pattern, but at the obvious lower score level.

Table 7
Example of SimpleText Task 2.1: source and references.

Sentence	G06.2_2810968146_2
Source	The model is a ResNet-18 variant, which is fed in images from the front of a simulated F1 car, and outputs optimal labels for steering, throttle, braking.
Reference	['ResNet-18 variant', 'braking', 'braking', 'f1 car', 'front', 'image', 'model', 'optimal label', 'resnet-18', 'simulated F1 car', 'steering', 'steering', 'throttle', 'throttle', 'to be fed', 'to output']
Difficulty	['d', 'e', 'e', 'e', 'e', 'e', 'e', 'e', 'd', 'd', 'e', 'e', 'e', 'e', 'e', 'm']
Source "d"	The model is a *ResNet-18* variant, which is fed in images from the front of a simulated F1 car, and outputs optimal labels for steering, throttle, braking.
Source "m"	The model is a ResNet-18 variant, which is fed in images from the front of a simulated F1 car, and *outputs* optimal labels for steering, throttle, braking.
Source "e"	The model is a ResNet-18 variant, which is *fed* in *images* from the front of a simulated F1 car, and outputs *optimal labels* for steering, throttle, braking.
Prediction	['resnet-18', 'throttle', 'braking', 'f1', 'fed']

Table 8
Example of SimpleText Task 2.1: Frequency of terms spotted.

Terms/Sentence	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**29**
Frequency (train)	53	99	90	100	44	55	23	22	16	20	3	5	4	4	1	7	2	2	1
Frequency (test)	18	31	61	65	45	32	26	16	10	3	2	4							

Table 9
Example of SimpleText Task 2.1: Spotted term or concept.

Source	**Number of Terms**	**Occurs in Sentence**	**Not in Sentence**
Train	2,579	2,098	481
Train (case folding)	2,579	2,334	245
Test	1,440	1,312	128
Test (case folding)	1,440	1,347	93

Table 10
CLEF 2024 SimpleText Task 2: Top 1 Semantic Match

Run	**Rouge**				**BERTScore**		
	1	**2**	**L**	**Lsum**	**P**	**R**	**F1**
Train	0.3729	0.0946	0.3723	0.3733	0.92	0.93	0.92
Test	0.3825	0.0957	0.3810	0.3825	0.93	0.93	0.92

variation is making the prediction of all terms neigh impossible, and makes averaging over terms an unreliable indicator of the per sentence performance. Evaluation over the sets of top retrieved terms, as we did in Table 6 shows indeed reasonable performance for our basic approach.

The recall of our approach is relatively low, as the baseline rarest term approach cannot find multi-word phrases. In addition, many of the ground truth terms do not literally appear in the sentence, and require case folding, morphologically normalization, or even more complex transformations to correctly align with the exact orthography of the scientific text.

Table 9 quantifies how often the spotted term or phrase is literally occurring in the sentences. We observe a fraction varying from 6.5% to 18.7%. While many cases concern morphological normalization that is useful to conflate similar concepts across different sentences (base form of verbs, singular for nouns etc). However, the evaluation measures will treat such cases as a failed match, and recall oriented measures should be treated with care.

Table 11
Results for CLEF 2024 SimpleText: Task 3.1 sentence-level (top) and Task 3.2 abstract-level (bottom) text simplification on the train set

run_id	count	FKGL	SARI	BLEU	Compression ratio	Sentence splits	Levenshtein similarity	Exact copies	Additions proportion	Deletions proportion	Lexical complexity score
Source	893	14,30	19,18	38,95	1,00	1,00	1,00	1,00	0,00	0,00	8,72
Reference References	893	11,70	100,00	100,00	0,84	1,07	0,72	0,04	0,21	0,37	8,63
UAms_GPT2_Check	714	11,87	35,21	27,35	1,02	1,22	0,87	0,11	0,17	0,14	8,59
UAms_GPT2	714	11,21	34,73	23,69	1,28	1,47	0,79	0,05	0,28	0,12	8,56
UAms_Wiki_BART_Snt	714	12,34	34,19	37,18	0,83	0,99	0,88	0,29	0,02	0,19	8,64
UAms_Cochrane_BART_Snt	714	13,74	26,70	36,69	0,94	0,99	0,95	0,56	0,03	0,08	8,67
Source	175	14,30	19,53	39,95	1,00	1,00	1,00	1,00	0,00	0,00	8,88
Reference	175	11,80	100,00	100,00	0,80	1,04	0,70	0,00	0,20	0,40	8,75
UAms_GPT2_Check_Abs	119	12,75	36,68	16,48	0,59	0,66	0,60	0,01	0,11	0,50	8,61
UAms_GPT2_Check_Snt	119	11,88	35,97	28,86	1,00	1,22	0,85	0,01	0,18	0,15	8,71
UAms_Cochrane_BART_Par	119	16,15	35,12	26,23	0,70	0,59	0,70	0,04	0,08	0,36	8,72
UAms_Wiki_BART_Doc	119	16,45	33,36	28,35	1,01	0,83	0,81	0,00	0,18	0,15	8,73
UAms_Cochrane_BART_Doc	119	14,78	33,23	9,55	0,40	0,40	0,52	0,03	0,01	0,61	8,76
UAms_Wiki_BART_Par	119	13,26	30,31	36,76	0,89	1,00	0,88	0,01	0,03	0,13	8,81

Table 10 evaluates the top 1 rarests term as returned by our baseline approach, and compares it to the entire list of reference terms. As our term is a unigram, we score well on Rouge-1 but not on Rouge-2 (we retained hyphened multiword terms, hence the score is not zero). With BERTScore we can see the semantic relatedness of our top 1 term and the reference terms, ignoring the exact orthography. The scores in terms are very encouraging in terms of over 90% precision, recall, and F1. Note that this evaluation is retricted to our first spotted term, and score 1.0 in case this term is part of any of the expert's reference terms.

3.3. Task 3: Text Simplification

We continue with Task 3, asking to simplify scientific text.

3.3.1. Evaluation

Table 11 shows the results on the train data, both in terms of text statistics and in terms of evaluation against the human reference simplifications.[6] We make a number of observations. First, looking at the GPT-2 models, we see that both sentence level and abstract level text simplification considerably brings down the FKGL measure, and obtain reasonable SARI and BLEU scores against the reference simplifications. The abstract level simplification leads to deletions of entire sentences, with 50% less tokens than the source, but still outperformaning the sentence level simplification retaining all sentences. Second, the BART model trained on Wiki-Auto and on Cochrane-Auto lay summaries significantly outperforms the GPT-2 model on BLEU with scores of 0.37, signaling high n-gram overlap with the

[6]Some of the differences in the number of sentences/abstracts are due to those sources not included in the test source file. This particularly concerns very short fragments from biomedical literature added as additional train data, but not part of the SimpleText corpus.

Table 12
Results for CLEF 2024 SimpleText: Task 3.1 sentence-level (top) and Task 3.2 abstract-level (bottom) text simplification on the test set

run_id	count	FKGL	SARI	BLEU	Compression ratio	Sentence splits	Levenshtein similarity	Exact copies	Additions proportion	Deletions proportion	Lexical complexity score
Source	578	13.65	12.02	19.76	1.00	1.00	1.00	1.00	0.00	0.00	8.80
Reference	578	8.86	100.00	100.00	0.70	1.06	0.60	0.01	0.27	0.54	8.51
UAms_GPT2_Check	578	11.47	29.91	15.10	1.02	1.23	0.87	0.14	0.17	0.14	8.68
UAms_GPT2	578	10.91	29.73	13.07	1.30	1.50	0.79	0.06	0.29	0.12	8.63
UAms_Wiki_BART_Snt	578	12.13	27.45	21.56	0.85	0.99	0.89	0.32	0.02	0.16	8.73
UAms_Cochrane_BART_Snt	578	13.22	18.45	19.21	0.95	0.99	0.96	0.59	0.02	0.07	8.77
Source	103	13.64	12.81	21.36	1.00	1.00	1.00	1.00	0.00	0.00	8.88
Reference	103	8.91	100.00	100.00	0.67	1.04	0.60	0.00	0.23	0.53	8.66
UAms_GPT2_Check_Abs	103	12.85	36.47	13.12	0.91	0.92	0.59	0.00	0.18	0.45	8.73
UAms_Cochrane_BART_Doc	103	14.46	33.51	9.39	0.65	0.58	0.54	0.04	0.06	0.53	8.80
UAms_Cochrane_BART_Par	103	16.53	31.58	15.40	1.08	0.80	0.67	0.04	0.15	0.32	8.81
UAms_GPT2_Check_Snt	103	11.57	30.71	15.24	1.54	1.70	0.78	0.00	0.27	0.13	8.77
UAms_Wiki_BART_Doc	103	15.68	26.50	15.11	1.51	1.14	0.76	0.01	0.25	0.11	8.79
UAms_Wiki_BART_Par	103	13.11	23.92	19.49	1.39	1.37	0.81	0.01	0.11	0.10	8.86

human reference simplifications. For abstract level simplification it is encouraging to see that the Cochrane model trained on scientific data is slightly outperforming the Wiki-Auto trained model. Third, the paragraph and document level models trained on Wiki-Auto and Cochrane do again not outperform the sentence level simplifications, under the conditions of the task's train data. The train data is derived from the sentence level scientific text simplification references from the earlier years of the track. Proper document level text simplification approaches lead to considerable deletions, and perform reasonable given their far more succinct output.

Table 12 shows the Task 3 results for both sentence-level (top) and abstract-level (bottom) scientific text simplifications. We make again a number of observations. First, looking at the GPT-2 models, we see again low FKGL scores indicating favorable readability, with reasonable SARI and BLEU scores. The abstract level simplification clearly outperforms the merged sentence level simplifications, despite a far more succinct output. Second, looking at the BART model trained on Wiki-Auto and on Cochrane-Auto lay summaries, we see that the Cochrane model trained on scientific data is clearly outperforming the Wiki-Auto trained model on SARI for document level text simplification. Third, the paragraph and document level models trained on Wiki-Auto and Cochrane do again not outperform the sentence level simplifications, under the conditions of the task's test data based on aggregated human reference sentence simplifications. These models take discourse structure into account, or may merge or reorder sentences, and are less focused on single sentence wordsmithing, or promoting sentence splits.

3.3.2. Analysis

In this section, we look analyze the output of our systems by realigning the simplified text predictions to the source sentences.

Table 13
Example of SimpleText Task 3 prediction versus source: ~~deletions~~, insertions, and whole sentence insertions

Topic	Document	Output
G07.1	2111507945	The growth of social media provides a convenient ~~communication scheme~~ way for people to communicate , but at the same time it becomes a hotbed of misinformation . |~~The~~ This wide spread of misinformation over social media is injurious to public interest . It is difficult to separate fact from fiction when talking about social media . |We design a framework , which ~~integrates~~ combines collective intelligence and machine intelligence , to help identify misinformation . |The basic idea is : (1) automatically index the expertise of users according to their microblog ~~contents~~ posts ; and (2) match ~~the~~ experts with the same information given to suspected misinformation . |By sending the suspected misinformation to appropriate experts , we can ~~collect~~ gather the ~~assessments of experts~~ relevant data to judge the credibility of the information , and help refute misinformation . |In this paper , we ~~focus on~~ look at expert finding for misinformation identification . We ask experts to identify the source of the misinformation , and how it is spread . |We propose a tag-based ~~method~~ approach to ~~index~~ indexing the expertise of microblog users ~~with social tags~~ . Our approach will allow us to identify which posts are most relevant and which are not . |Experiments on a real world dataset ~~demonstrate~~ show the effectiveness of our ~~method~~ approach for expert finding with respect to misinformation identification in microblogs .

Controlled Creativity Text simplification models are based on generative large language models. For example, one of the models we used is a GPT-2 model [10] called the Keep it Simple (KiS). The model is based on GPT-medium, using a straightforward unsupervised training task with an explicit loss in terms of fluency, saliency, and simplicity. Such models are used in generative mode, generating the output in fairly unconstrained mode in order to ensure none of the input is lost (in particular for longer input). As a result there is also a chance that the model continues to generate output after the source has been fully simplified. This can cause the model to overgenerate and produce spurious content.

Table 13 shows an example output simplification, combining the input sentences belonging to the abstract of documents 2111507945 retrieved for query G07.1. We show here deletions and insertions relative to the source input sentences (in this case 8 in total). Many simplifications are revisions of the input, but we also observe that sometimes an entire sentence is inserted (shown as xxx). Modern models such as ours generate the simplification, which may lead to additional output being generated at the end. Recall that the example as shown in Table 13 merges 8 separate input sentences in the train data (indicated by |), making this occur multiple times at the end of three of the inputs.

Spurious Content We analyze the frequency of spurious content in our runs. For human readers, detecting such sentences by simply inspecting the output is hard, as they are very reasonable completions generated with awareness of the preceding context. We experimented with unsupervised approaches to tackle the generation of spurious generation, by post-processing the output in relation to the original input. Similar to the edits as shown in the table, we process input and output, and remove any sentence that has been inserted without grounding in the input.

Table 14 quantifies how often such spurious generation occurs. We make a number of observations. First, the spurious generation is not infrequent. Some systems have a marginal number of cases, which may be a result of imperfect alignment due to short sentences or changing word orders. Other systems have many cases, up to 1,390 sentences or 29% (and 111 abstracts or 14%) of the input for the unconstrained GPT2 model.

Second, in the GPT2 sentence level case, we remove this additional content in a post-processing step, ensuring all the output is grounded on input sentences. This is effectively removing spurious content from the runs, and also leads to better performance in Table 12.

Third, while our post-processing already has a favorable effect on the evaluation measures, we feel that it has great benefits not reflected by these scores. Our post-processing is specifically, and only,

Table 14
Analysis of SimpleText Analysis: Spurious generation for sentence-level (top) and abstract-level (bottom) scientific text simplification

Run	# Input Sentences/Abstracts	Spurious Content	
		Number	Fraction
UAms-1_GPT2	4,797	1,390	0.29
UAms-1_GPT2_Check	4,797	3	0.00
UAms-1_Wiki_BART_Snt	4,797	14	0.00
UAms-1_Cochrane_BART_Snt	4,797	25	0.01
UAms-2_GPT2_Check_Snt	782	111	0.14
UAms-2_GPT2_Check_Abs	782	1	0.00
UAms-2_Wiki_BART_Par	782	46	0.06
UAms-2_Wiki_BART_Doc	782	74	0.09
UAms-2_Cochrane_BART_Par	782	28	0.04
UAms-2_Cochrane_BART_Doc	782	2	0.00

removing spurious generation (or "hallucination") of the output. These results highlight and quantify the severity of this problem in generative text simplification models such as our GPT2 model. At the same time, it offers a practical approach to tackle this undesirable aspect head-on.

4. Discussion and Conclusions

This paper detailed the University of Amsterdam's participation in the CLEF 2024 SimpleText track. We conducted a range of experiments, for each of the three tasks of the track.

For Task 1 on *Content Selection*, we observed a very solid performance for zero-shot neural reranking, as well as competitive effectivess for complexity-aware rankers that purposely avoid to retrieve results with a high text complexity.

For Task 2 on *Complexity Spotting*, we submitted preliminary approaches based on standard term weighting, and observed that naive approaches can help locate difficult terms.

For Task 3 on *Text Simplification*, we experimented with a range of models and approaches, and observed that sentence-level simplification approaches can be very effective to reduce the complexity of scientific text, and that paragraph and abstract level simplifications lead to far shorter output including whole sentence deletions.

Acknowledgments

This research was conducted as part of the final research projects of the Master in Artificial Intelligence at the University of Amsterdam. We thank the track and task organizers for their amazing service and effort in making realistic benchmarks for scientific text simplification available. Jaap Kamps is partly funded by the Netherlands Organization for Scientific Research (NWO CI # CISC.CC.016, NWO NWA # 1518.22.105), 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.

References

[1] L. Ermakova, et al., Overview of the CLEF 2024 SimpleText track: Improving access to scientific texts, in: L. Goeuriot, et al. (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Fifteenth International Conference of the CLEF Association (CLEF 2024), Lecture Notes in Computer Science, Springer, 2024.

[2] E. SanJuan, et al., Overview of the CLEF 2024 SimpleText task 1: Retrieve passages to include in a simplified summary, in: G. Faggioli, et al. (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), CEUR Workshop Proceedings, CEUR-WS.org, 2024.

[3] G. M. D. Nunzio, et al., Overview of the CLEF 2024 SimpleText task 2: Identify and explain difficult concepts, in: G. Faggioli, et al. (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), CEUR Workshop Proceedings, CEUR-WS.org, 2024.

[4] L. Ermakova, et al., Overview of the CLEF 2024 SimpleText task 3: Simplify scientific text, in: G. Faggioli, et al. (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), CEUR Workshop Proceedings, CEUR-WS.org, 2024.

[5] J. D'Souza, et al., Overview of the CLEF 2024 SimpleText task 4: Track the state-of-the-art in scholarly publications, in: G. Faggioli, et al. (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2024), CEUR Workshop Proceedings, CEUR-WS.org, 2024.

[6] L. Ermakova, T. Miller, A. Bosser, V. M. Palma-Preciado, G. Sidorov, A. Jatowt, Overview of JOKER - CLEF-2023 track on automatic wordplay analysis, in: A. Arampatzis, E. Kanoulas, T. Tsikrika, S. Vrochidis, A. Giachanou, D. Li, M. Aliannejadi, M. Vlachos, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction - 14th International Conference of the CLEF Association, CLEF 2023, Thessaloniki, Greece, September 18-21, 2023, Proceedings, volume 14163 of *Lecture Notes in Computer Science*, Springer, 2023, pp. 397–415. URL: https://doi.org/10.1007/978-3-031-42448-9_26. doi:10.1007/978-3-031-42448-9_26.

[7] J. Lin, X. Ma, S. Lin, J. Yang, R. Pradeep, R. F. Nogueira, Pyserini: A python toolkit for reproducible information retrieval research with sparse and dense representations, in: F. Diaz, C. Shah, T. Suel, P. Castells, R. Jones, T. Sakai (Eds.), SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021, ACM, 2021, pp. 2356–2362. URL: https://doi.org/10.1145/3404835.3463238. doi:10.1145/3404835.3463238.

[8] L. Ermakova, J. Kamps, Complexity-aware scientific literature search: Searching for relevant and accessible scientific text, in: G. M. D. Nunzio, F. Vezzani, L. Ermakova, H. Azarbonyad, J. Kamps (Eds.), Proceedings of the Workshop on DeTermIt! Evaluating Text Difficulty in a Multilingual Context @ LREC-COLING 2024, ELRA and ICCL, Torino, Italia, 2024, pp. 16–26. URL: https://aclanthology.org/2024.determit-1.2.

[9] J. Kamps, The impact of author ranking in a library catalogue, in: G. Kazai, C. Eickhoff, P. Brusilovsky (Eds.), Proceedings of the 4th ACM Workshop on Online books, complementary social media and crowdsourcing, BooksOnline 2011, Glasgow, United Kingdom, October 24, 2011, ACM, 2011, pp. 35–40. URL: https://doi.org/10.1145/2064058.2064067. doi:10.1145/2064058.2064067.

[10] P. Laban, T. Schnabel, P. N. Bennett, M. A. Hearst, Keep it simple: Unsupervised simplification of multi-paragraph text, in: ACL/IJCNLP'21: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Association for Computational Linguistics, 2021, pp. 6365–6378. URL: https://doi.org/10.18653/v1/2021.acl-long.498.