# University of Amsterdam at the CLEF 2023 SimpleText Track

Roos Hutter, Jop Sutmuller, Mary Adib, David Rau and Jaap Kamps

*University of Amsterdam, Amsterdam, The Netherlands*

Abstract
This paper reports on the University of Amsterdam's participation in the CLEF 2023 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 the retrieved information within the context of the track's setup. Our results suggest that text complexity is an essential aspect to consider for improving non-expert access to scientific information, and opens up new routes to develop effective scientific information access technology tailored to needs of the general public.

**Keywords**
Information Storage and Retrieval, Natural Language Processing, Scientific Information Access, Text Simplification

## 1. Introduction

The advent of the internet and social media has been revolutionary in changing every aspect of information creation and information consumption. Whereas this comes with unprecedented strengths and new opportunities, 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. However, in practice, few non-experts consult scientific sources and rely on shallow information distributed on the web and in social media. One of the main reasons for avoiding the scientific literature is its presumed complexity. The CLEF 2023 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 2023 LNCS proceedings [1], as well as the detailed task overviews in the CEUR proceedings [2, 3, 4].

We conduct an extensive analysis of the corpus of scientific abstracts and the three tasks of the track: Task 1 on content selection and *avoiding* complexity; Task 2 on complexity spotting in extracted sentences from scientific abstracts; and Task 3 on text simplification proper rewriting sentences from these abstracts.

*CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece*
kamps@uva.nl (J. Kamps)
0000-0002-6614-0087 (J. Kamps)

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 task. We end in Section 4 by discussing our results and outlining the lessons learned.

## 2. Experimental Setup

In this section, we will detail our approach for the three CLEF 2023 SimpleText track tasks.

For details of the exact task setup and results we refer the reader to the detailed overview of the track in Ermakova et al. [1]. The basic ingredients of the track are:

**Corpus** The CLEF 2023 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 114 requests with 1-4 queries per context article, 47 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.

**Train Data** For Task 1, there are relevance judgments for 29 requests (corresponding to 15 Guardian articles, G01–G15), with 23 queries having more than 10 relevant abstracts. For Task 2, there are 203 train sentences (with ground truth complex terms/concepts) and 2,234 (small), 4,797 (medium), and 152,072 (large) test sentences. For Task 3, there are 648 train sentences with human simplifications, and again 2,234 (small), 4,797 (medium), and 152,072 (large) test sentences.

**Assessments** For Task 1, there are new relevance assessments 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). For Task 2, evaluation is based on 592 distinct sentences, and 4,167 distinct sentence-term pairs (based on pooling) manually evaluated term limits (does the extracted term cover the entire concept) and difficulty (3 grades ranging from ‘no explanation needed’ to ‘explanation required’). For Task 3, in addition to the train data on 648 sentences, evaluation is based on the manual simplifications of 245 sentences.

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

**Task 1** *This task requires ranking scientific abstracts in response to a non-expert, general query prompted by a popular science article.*

We submitted ten runs in total, shown in Table 1. We first submitted three runs focusing on regular information retrieval effectiveness. One is a vanilla baseline run on the provided Elastic Search index, using normal keyword query rather than quoted phrase queries (as in the

[1] https://www.theguardian.com/science

[2] https://techxplore.com/

**Table 1**
CLEF 2013 SimpleText Track Submissions

| Task | Run | Description |
|---|---|---|
| 1 | UAms_Task_1_Elastic | Vanilla elastic run (queries without quotes) |
| 1 | UAms_Task_1_CE100 | Minilm12 full BERT based crossencoder reranker on top 100 |
| 1 | UAms_Task_1_CE1k | Minilm12 full BERT based crossencoder reranker on top 1k |
| 1 | UAms_Task_1_ElF_Read25 | Elastic filtered on Readability (rel) |
| 1 | UAms_Task_1_ElF_Cred53 | Elastic filtered on Credibility (rel) |
| 1 | UAms_Task_1_ElF_Cred44 | Elastic filtered on Credibility (rel) |
| 1 | UAms_Task_1_ElF_Cred53Read | Elastic filtered on Credibility and Readability (rel) |
| 1 | UAms_Task_1_ElF_Cred44Read | Elastic filtered on Credibility and Readability (rel) |
| 1 | UAms_Task_1_CE1k_Combine | Neural ranker combining relevance and readability (comb) |
| 1 | UAms_Task_1_CE1k_Filter | Neural ranker combining relevance and readability (comb) |
| 2 | UAms_Task_2_RareIDF | IDF baseline using single word terms only |
| 3 | UAms_Task_3_Large_KIS150_Clip | GPT-2 based text simplification |
| 3 | UAms_Task_3_Large_KIS150 | GPT-2 TS with post-processing removing hallucination |

provided examples). The other two are neural crossencoder rerankings of this run, based on zero-shot application of an MSMARCO trained ranker, reranking either the top 100, or the top 1k retrieved abstracts.[3]

We submitted seven runs aiming to take the readability and/or credibility of the results into account. The first run simply filters out the most complex abstract per request, using a standard readability measure. The run is aiming to remove about 25% of the results, with the remaining abstracts in the same relevance order as in the original Elastic Search run. The next two runs perform a similar filter based on credibility where we filter both on recency and the number of citations. One run selects abstracts since 2005 with at least 3 citations (removing about 5% of results), and the other abstracts since 2014 with at least 4 citations (removing about 25% of results). The next two runs combine the credibility and readability filters, removing about 30% of results for 2005 and 3 citations filter, and removing about 46% of results for the 2014 and 4 citations filter.

The final two runs combine the scores of the cross-encoder reranker with readability scores, which may lead to a different order of results in the file. Specifically, the neural crossencoder score is combined with a score based on (14 − FKGL), promoting easy (i.e., low FKGL) abstracts and demoting complex (i.e., high FKGL) abstracts. The second variant still removes those abstracts with complexity higher than FKGL 14, while reranking those with lower FKGL in the same way.

**Task 2** *What concept needs to be explained or rewritten in a given sentence, extracted from a scientific abstract.*

We submitted a single run, also shown in Table 1. Based on preliminary experiments, our submission is using an idf-based term weighting to locate the most rare terms. Specifically, we

[3] https://huggingface.co/cross-encoder/ms-marco-MiniLM-L-12-v2

**Table 2**
Evaluation of SimpleText Task 1 (Test data)

| Run | MRR | Precision | | | NDCG | | | Bpref | MAP |
|---|---|---|---|---|---|---|---|---|---|
| | | 5 | 10 | 20 | 5 | 10 | 20 | | |
| Elastic | 0.6424 | 0.4353 | 0.4059 | 0.2990 | 0.4165 | 0.3911 | 0.3315 | 0.2502 | 0.1895 |
| CE 100 | 0.7050 | 0.5118 | 0.4912 | 0.3657 | 0.5004 | 0.4782 | 0.4007 | 0.2616 | 0.2011 |
| CE 1k | 0.6329 | 0.4765 | 0.4735 | 0.3578 | 0.4502 | 0.4448 | 0.3816 | 0.2797 | 0.2051 |
| Read. filter | 0.6076 | 0.3824 | 0.3735 | 0.2833 | 0.3723 | 0.3539 | 0.3105 | 0.2194 | 0.1522 |
| Cred. filter (2005/3) | 0.6429 | 0.4235 | 0.4088 | 0.3010 | 0.4043 | 0.3883 | 0.3292 | 0.2454 | 0.1833 |
| Cred. filter (2014/4) | 0.6888 | 0.4294 | 0.4324 | 0.2951 | 0.4215 | 0.4103 | 0.3300 | 0.2395 | 0.1719 |
| C+R filter (2005/3) | 0.6088 | 0.3765 | 0.3676 | 0.2784 | 0.3623 | 0.3469 | 0.3042 | 0.2133 | 0.1456 |
| C+R filter (2014/4) | 0.6625 | 0.4118 | 0.3971 | 0.2775 | 0.3902 | 0.3723 | 0.3101 | 0.2123 | 0.1403 |
| Rel+Read | 0.5880 | 0.4412 | 0.4147 | 0.3098 | 0.3854 | 0.3706 | 0.3250 | 0.2700 | 0.1865 |
| Rel+Read filter | 0.6403 | 0.5000 | 0.4765 | 0.2941 | 0.4754 | 0.4533 | 0.3334 | 0.2727 | 0.1936 |

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).

**Task 3** *Rewrite a sentence from a scientific abstract.*

We submitted two runs shown in Table 1. We use a standard text simplification model, based on the GPT-2 based keep it simple (KiS) model of Laban et al. [5]. We run a pretrained version of this model available from HuggingFace,[4] in a zero-shot way on both the train and test corpus.

One of the main challenges of these models which generate the output is the risk of "hallucination," in which the model generates reasonable and credibly looking output that is not grounded on the input text. In preliminary experiments, we observed that was happening in particular on the end of the generation where additional content is generated, including entire extra sentences. We implemented a post-processing of the output that compares the input text to the generated output, and removes those sentences for which there is no direct overlap with the input.

## 3. Experimental Results

In this section, we will present the results of our experiments, in four self-contained subsections following the CLEF 2023 SimpleText Track corpus and tasks.

### 3.1. Task 1: Content Selection

We discuss our results for Task 1, asking to retrieve scientific articles in response to a query based on a popular science article.

[4] https://huggingface.co/philippelaban/keep_it_simple

**Table 3**
Analysis of SimpleText Task 1 output (over all 114 queries)

| Run | Queries | Top | Year | | Citations | | Length | | FKGL | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | **Avg** | **Med** | **Avg** | **Med** | **Avg** | **Med** | **Avg** | **Med** |
| Elastic | 114 | 10 | 2012.0 | 2014 | 13.1 | 3.0 | 1000.0 | 995.5 | 14.0 | 13.9 |
| CE 100 | 114 | 10 | 2011.7 | 2013 | **25.2** | 4.0 | 1102.3 | 1041.5 | 14.2 | 14.1 |
| CE 1k | 114 | 10 | 2011.8 | 2014 | 21.6 | 3.0 | 1142.3 | 1047.0 | 14.2 | 14.1 |
| Read. filter | 114 | 10 | 2011.6 | 2013 | 10.9 | 2.0 | 843.8 | 894.0 | 13.8 | 13.8 |
| Cred. filter (2005/3) | 114 | 10 | 2012.6 | 2014 | 13.6 | 3.0 | 1016.4 | 1010.0 | 14.0 | 14.0 |
| Cred. filter (2014/4) | 114 | 10 | 2013.0 | 2015 | 20.6 | 6.0 | 1052.0 | 1055.5 | 14.1 | 14.2 |
| C+R filter (2005/3) | 114 | 10 | 2012.2 | 2014 | 12.4 | 3.0 | 851.5 | 898.0 | 13.9 | 13.8 |
| C+R filter (2014/4) | 114 | 10 | 2012.7 | 2015 | 16.4 | 6.0 | 876.7 | 936.0 | 13.9 | 13.9 |
| Rel+Read | 114 | 10 | 2011.6 | 2014 | 16.9 | 3.0 | 992.9 | 909.0 | 11.2 | 11.2 |
| Rel+Read filter | 114 | 10 | 2011.5 | 2014 | 20.8 | 3.0 | 1056.8 | 982.0 | 12.2 | 12.4 |

### 3.1.1. Retrieval effectiveness

Table 2 shows the performance of the Task 1 submissions on the test data. First, comparing the elastic search and neural rerankers, we see that the crossencoders lead to considerable improvement of retrieval effectiveness, on all evaluation measures. In particular, NDCG@10 increases from 0.3911 up to 0.4782. Second, for the credibility filters on the Elastic baseline, we see that promoting recent and more cited papers lead to improvements of retrieval effectiveness. In particular, NDCG@10 improves from 0.3911 up to 0.4103. Third, for the readability filters on the Elastic baseline, we see that promoting more accessible papers lead to decrease of retrieval effectiveness. This is entirely expected as the relevance judgments did not consider the complexity of the abstracts: many relevant abstracts may have high text complexity. Fourth, the runs combining neural relevance and readability scores can lead to very similar retrieval effectiveness scores. In particular, the filter variant combining the neural crossencoder on the top 1k Elastic results, obtains an NDCG@10 of 0.4533.

Our general conclusion is that the approaches promoting credibility and readability are still effective and obtain a very reasonable performance. The main aim of these runs is not to improve retrieval effectiveness, but to improve the experience of our non-expert user by aiming to retrieve relevant and accessible abstracts in the ranking.

### 3.1.2. Analysis of retrieved papers

Some of the runs specifically target to retrieve easier to read abstracts, or are ranked on a combined score factoring in relevance and credibility or readability of the results. But to what extent do our approaches realize this?

Table 3 shows an analysis of the metadata and the text of the top retrieved articles (title+abstracts) over all topics in the train and test data.

Looking at *credibility*, we see that the baseline Elastic search already retrieves recent articles (mean 2012, median 2014) receiving reasonable numbers of citations (mean 13, median of 3).

**Table 4**
Results for the SimpleText Task 2: Selecting rare terms

| Run | Total | Evaluated | | Score | |
|---|---|---|---|---|---|
| | | | +Limits | | +Limits |
| UAms_Task_2_RareIDF | 675090 | 1293 | 1145 | 309 | 241 |

The credibility filters have a minor effect on recency (mean up to 2013, median up to 2015) and an increase in citations (mean up to 21, median up to 6). We also observe that the neural reranking also leads to a higher number of citations (mean up to 25, and median up to 4).

Looking at *readability*, we observe a fairly high level of text complexity for basic retrieval approaches, with average and median FKGL around 14 of the retrieved abstracts. The readability and credibility filters lead to limited reduction in text complexity over all 114 requests. The two runs combining the neural relevance scores with the readability scores are effective in significantly lowering the complexity of the retrieved abstracts, with a median FKGL of 11.2 and 12.4.

To put this into perspective: an FKGL of 11-12 corresponds to the reading level of an average user who finished compulsory education, whereas an FKGL of 14 corresponds to several years of university education. Hence, these approaches are able to rank easier to read results first, while still retrieving a very similar number of relevant results in terms of retrieval effectiveness.

### 3.2. Task 2: Complexity Spotting

We continue with Task 2, asking to locate the most difficult concepts in a sentence extracted from a potentially relevant abstract, retrieved in response to a general query prompted by a popular science article. We submitted a single run, using an IDF based approach to find the least common term in the sentence.

Table 4 shows the results of our official submission to Task 2. Our run retrieved a total of 675,090 single word terms for 135,508 unique sentences. A total of 1,295 terms in 592 sentences is evaluated, and a large fraction of highlighted terms (89%) has correct term limits.

Term difficulty is judged on a scale from 0 (no explanation required), 1 (explanation helps) to 2 (explanation necessary). A fair fraction of terms has a high level of difficulty (27% of the evaluated terms). Of these a high fraction has the correct term limits (78%).

Our results indicate that while the problem of identifying complex terms is a very hard problem in general, basic features such as IDF are already very useful as a first step and perform unexpectedly competitively. The main reason is the restricted choice of options given the small number of words in each sentence, making IDF a powerful initial filter for candidates.

### 3.3. Task 3: Text Simplification

We continue with Task 3, asking to perform text simplification proper, by rewriting a sentence extracted from a potentially relevant abstract, retrieved in response to a general query prompted by a popular science article.

**Table 5**
Example of SimpleText Task 3 output versus input: ~~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 . |

### 3.3.1. Approaches

Our experiments are based on the zero-shot application of an existing neural text simplification model from [5], called the Keep it Simple (KiS) model. The model is based on GPT-medium, using a straightforward unsupervised training task with an explicit loss in terms of fluency, saliency, and simplicity. We are interested in this model as it is fully trained in an unsupervised way, and could be retrained or fine-tuned for the corpus or other academic texts without the need for huge human training data.

Table 5 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 5 merges 8 separate input sentences in the train data (indicated by |), making this occur multiple times at the end of three of the inputs.

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 experiment 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 6**
Results for SimpleText Task 3: zero-shot GPT2 text simplification

| Run | #Snt | FKGL | SARI | BLEU | Comp. | Split | L.Sim. |
|---|---|---|---|---|---|---|---|
| UAms_Task_3_Large_KIS150 | 648 | 11.40 | 36.38 | 25.82 | 1.17 | 1.42 | 0.79 |
| UAms_Task_3_Large_KIS150_Clip | 648 | 11.93 | 36.66 | 28.68 | 0.99 | 1.23 | 0.85 |
| UAms_Task_3_Large_KIS150 | 245 | 10.51 | 33.02 | 14.60 | 1.27 | 1.48 | 0.76 |
| UAms_Task_3_Large_KIS150_Clip | 245 | 11.13 | 33.47 | 16.60 | 1.02 | 1.23 | 0.83 |

### 3.3.2. Results and Analysis

Table 6 shows the results of applying the KiS model zero-shot on the train (top) and test (bottom) data in terms of the generated output. We make a number of observations.

First, we observe reasonable SARI and BLEU scores for the scientific text, with a SARI of 0.36 for train and 0.33 for test sentences. To put this number into perspective, the original paper reports scores in the range of 0.26 to 0.43 on a Wikipedia corpus [6].

Second, we see that our model is able to reduce the text complexity to the 11-12 FKGL range corresponding to the exit level of compulsory education suitable for the average adult reader. While inspection of examples, such as shown in Table 5, show conservative edits it is encouraging that the readability measures are considerably lower than for the original scientific text.

Third, our post-processing to remove spurious generation has a positive effect throughout, leading to higher SARI and BLEU scores against the reference simplification. As this is affection only a fraction of the sentences, the effect on SARI and BLEU is modest. As we typically remove an entire sentence, the effect on compression rates is high, and it leads to considerable improvements of the Levenshtein similarity.

Table 7 quantifies how often such spurious generation occurs. Over the train data, consisting of 648 sentences, we remove additional sentences unwarranted by the original input in 126 cases. Over the large test data, consisting of 152,072 sentences, we remove additional sentences unwarranted by the original input in 40,449 cases. Over these samples, this affects between 19.4% (train) and 26.6% (test) of the cases. In all these cases we remove this additional content in a post-processing step, ensuring all the output is grounded on input sentences.

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, 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 2023 SimpleText track. We conducted a range of experiments, for each of the three tasks of the track.

**Table 7**
Results for SimpleText Task 3: Spurious generation

| Input | # Input Sentences | # Spurious Content | Fraction Spurious Content |
|---|---:|---:|---:|
| Train | 648 | 126 | 0.1944 |
| Test Large | 152,072 | 40,449 | 0.2660 |

For Task 1, we observed the effectiveness of zero-shot neural rankers for scientific text. We also found that specific credibility filters privileging recent or highly cited papers can even improve retrieval effectiveness. Readability filters can retain retrieval effectiveness on par with the best relevance rankers. This is an important and surprising finding as these approaches avoid complexity by retrieving only, or first, those abstracts at a readability level assumed to be suitable for a non-expert user. Hence the impact on the end-user in the track's use-case is even greater than indicated by the retrieval effectiveness evaluation.

For Task 2, we submitted preliminary approaches based on standard term weighting exploiting the corpus statistics or language model of a large scientific corpus. Our main finding was that although complex concept detection is a very hard task in general, it is a very viable and feasible task when the context is restricted to only the terms in a single sentence.

For Task 3, we experimented with a zero-shot pretrained GPT-2 based text simplification approach, Our main analysis was an extensive analysis of generative text simplification approaches, and to quantify the number and fraction of cases in which a generated output sentence is not warranted by any input sentence token. This is an actionable finding that can be immediately exploited to post-process the output in an unsupervised way, and to remove spuriously generated content. As this involves only a small fraction of the sentences, this leads to a small but consistent improvement of the evaluation scores. In fact, the standard text simplification evaluation measures are remarkably insensitive to hallucinated content, leading only to a minor penalty. However, the spurious content is very difficult spot by end-users, in particular non-experts, as it is a natural continuation of the previous text—yet at the same time completely unsupported by the original scientific abstract. Hence the impact on the end-user in the track's use-case is again far greater than indicated by the text simplification evaluation.

## Acknowledgments

This research was conducted as part of the final research projects of the Bachelor in Artificial Intelligence at the University of Amsterdam, This research is funded in part by the Netherlands Organization for Scientific Research (NWO CI # CISC.CC.016), and the Innovation Exchange Amsterdam (POC grant). Views expressed in this paper are not necessarily shared or endorsed by those funding the research.

## References

[1] L. Ermakova, E. SanJuan, S. Huet, H. Azarbonyad, O. Augereau, J. Kamps, Overview of the CLEF 2023 SimpleText Lab: Automatic simplification of scientific texts, in: A. Arampatzis, E. Kanoulas, T. Tsikrika, S. Vrochidis, A. Giachanou, D. Li, M. Aliannejadi, M. Vlachos,

G. Faggioli, N. Ferro (Eds.), CLEF'23: Proceedings of the Fourteenth International Conference of the CLEF Association, Lecture Notes in Computer Science, Springer, 2023.

[2] E. SanJuan, S. Huet, J. Kamps, L. Ermakova, Overview of the CLEF 2023 SimpleText Task 1: Passage selection for a simplified summary, in: [7], 2023.

[3] L. Ermakova, O. Augereau, H. Azarbonyad, Overview of the CLEF 2023 SimpleText Task 2: Identifying and explaining difficult concepts, in: [7], 2023.

[4] L. Ermakova, J. Kamps, Overview of the CLEF 2023 SimpleText Task 3: Scientific text simplification, in: [7], 2023.

[5] 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.

[6] W. Xu, C. Napoles, E. Pavlick, Q. Chen, C. Callison-Burch, Optimizing statistical machine translation for text simplification, Trans. Assoc. Comput. Linguistics 4 (2016) 401–415. URL: https://doi.org/10.1162/tacl_a_00107. doi:10.1162/tacl_a_00107.

[7] M. Aliannejadi, G. Faggioli, N. Ferro, M. Vlachos (Eds.), Working Notes of CLEF 2023: Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS.org, 2023.