AIIR Lab Systems for CLEF 2024 SimpleText: Large Language Models for Text Simplification

Notebook for the SimpleText Lab at CLEF 2024

Nicholas Largey, Reihaneh Maarefdoust, Shea Durgin and Behrooz Mansouri*

University of Southern Maine, Portland, Maine, USA

Abstract

This paper presents the participation of the Artificial Intelligence and Information Retrieval (AIIR) Lab from the University of Southern Maine in the CLEF 2024 SimpleText Lab. SimpleText has three main Tasks. Five systems are proposed for the first Task, which involves retrieving passages to include in a simplified summary. These systems select candidates using TF-IDF with expanded queries via LLaMA3. The re-ranking is performed using a bi-encoder, a cross-encoder, and LLaMA3. In Task 2, which involves identifying and explaining difficult concepts, three models utilizing LLaMA3 and Mistral are employed. Finally, for Task 3, which focuses on simplifying scientific text, four systems are introduced. Similar to Task 2, LLaMA3 and Mistral are used with different prompting and fine-tuning approaches. The experimental results show the proposed systems in Task 1 are the most effective, and for Tasks 2 and 3 are comparable with other systems proposed in the SimpleText lab.

Keywords

Scientific Text Simplification, Definition Extraction, Large Language Models.

1. Introduction

The CLEF 2024 SimpleText lab [1] is dedicated to enhancing accessibility to scientific information for all users, encompassing both information retrieval and natural language processing aspects. Unlike traditional text simplification methods, which often target lower literacy levels by making general texts more accessible to younger readers, Scientific Text Simplification has a distinct focus.

The Artificial Intelligence and Information Retrieval (AIIR) lab from the University of Southern Maine (USA) participated in three Tasks of the CLEF 2024 SimpleText lab. With advances in large language models (LLMs), our team considered using them for different Tasks, mainly focusing on two models: LLaMA3[1] and Mistral[2].

SimpleText Task 1 [3], **Retrieving Passages to Include in a Simplified Summary**, aims to retrieve passages from a vast collection of academic abstracts and bibliographic metadata that can aid in understanding this article. These relevant passages should pertain to any of the topics covered in the article. For Task 1, we submitted five runs using different techniques, ranging from cross and bi-encoders to large language models (LLMs) for query expansion and re-ranking.

In Task 2 [4], **Identifying and Explaining Difficult Concepts**, the goal is to identify which concepts in scientific abstracts need explanation and contextualization to assist readers in understanding the scientific text. Our team participated in two Subtasks: 2.1) retrieve up to 5 difficult terms in a given passage from a scientific abstract, and 2.2) provide an explanation of these difficult terms. For these Subtasks, in addition to LLaMA3 and Mistral, we used a fine-tuned LLaMA3 model.

Finally, Task 3 [5], **Simplify Scientific Text**, tackles the problem of creating simplified versions of sentences taken from scientific abstracts. The input for the systems is popular science articles, queries,

CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024, Grenoble, France

*Corresponding author.

nicholas.largey@maine.edu (N. Largey); reihaneh.maarefdoust@maine.edu (R. Maarefdoust); shea.durgin@maine.edu (S. Durgin); behrooz.mansouri@maine.edu (B. Mansouri)

https://cs.usm.maine.edu/~behrooz.mansouri/ (B. Mansouri)

0009-0008-4004-2244 (N. Largey); 0009-0007-4152-8409 (R. Maarefdoust); 0009-0005-8925-5329 (S. Durgin); 0000-0002-0400-9761 (B. Mansouri)

[1] https://github.com/meta-llama/llama3

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

and corresponding scientific paper abstracts, all divided into individual sentences. For this Task, we also used a fine-tuned LLaMA3 model, and Mistral as our proposed approaches.

The reported results show our proposed systems for all three Tasks have high effectiveness. For Task 1, and Subtask 3.2 our proposed models were the most effective ones, while for Task 2, and Subtask 3.1, they are comparable to the leading systems. In the next sections, we will describe our systems for each Task, followed by evaluation results and analysis.

2. Task 1: Retrieving Passages to Include in a Simplified Summary

This section first describes the data for Task 1 [3]. Then we describe our five proposed systems. Finally, we will provide the results and analysis.

2.1. Topic and Collection

As described by the organizers, the topics for this Task are from two resources: 1) the tech section of The Guardian[2] newspaper (topics G01 to G20), and 2) Tech Xplore[3] website (topics T01 to T20). Each topic represents a query selected from one of these resources. For instance, for the topic 'G13.1', the query is "digital marketing", with its context being an article titled "Baffled by digital marketing? Find your way out of the maze", from The Guardian. Participants have access to the whole article, its title, and the query.

The main corpus consists of a large set of scientific abstracts and associated metadata in the field of computer science and engineering. The 12th version of the Citation Network Dataset [6], released in 2020, provides this data extracted from DBLP, ACM, MAG (Microsoft Academic Graph), and other sources. It contains 4,894,083 bibliographic references published before 2020, 4,232,520 English abstracts, 3,058,315 authors with affiliations, and 45,565,790 ACM citations.

2.2. Proposed Models

AIIR Lab submitted five runs, of which three participated in the pooling and assessment process. Here, we explain each of our proposed approaches:

- **Query Expansion with LLaMA3, Search with Bi-Encoder / Cross-Encoder. (LLaMA Bi-Encoder/CrossEncoder)**: For Task 1, input queries are short keyword terms (e.g., "drones", "advertising", "gene editing") selected from technical articles. To contextualize and potentially expand these queries, we consider their related articles and leverage LLaMA3[4] for query reformulation/expansion. Following the approach proposed by Anand et al. [7], we provide the query and the article to the model, and use the system prompt for query rewriting/expansion shown in Table 9 in the Appendix.
  Using our system prompt, we then pass the query, the related article title, and context to LLaMA3 and expand the initial query. Table 1 shows examples of expanded queries. After this step, we use TF-IDF from PyTerrier [8] with default parameters to get the top-5000 results for each expanded query.
  We then re-rank the candidates using two architectures of SentenceBERT [9]: bi- and cross-encoder. For the bi-encoder, we use 'all-mpnet-base-v2' model due to its demonstrated effectiveness in capturing semantic similarity between queries and documents across various information retrieval Tasks. This model is used without further fine-tuning. The input query for the bi-encoder combines the initial query, related article title, and LLaMA-expanded query. We consider the title and abstract of each passage as the document for comparison with the query.
  For our second run, based on observations from previous lab participation [10], we fine-tune a cross-encoder model, 'ms-marco-MiniLM-L-6-v2'. For fine-tuning, we use the data from previous

[2] https://www.theguardian.com/uk/technology
[3] https://techxplore.com/
[4] We used Meta-LLaMA3-8B-Instruct model from HuggingFace

Table 1
Query rewriting/expansion using LLaMA3.

Initial Query	Expanded Query
drones	UK military drones Nagorno-Karabakh conflict Azerbaijan Armenia
advertising	advertising digital marketing channels
gene editing	Gene editing Crispr therapy diseases treatment prospects

years of the SimpleText lab, splitting into 90% training and 10% validation sets. We fine-tune the model for 25 epochs, choosing the hyperparameters with the highest MRR@10 (Mean Reciprocal Rank) on the validation set. The input queries were fed to this model as

Initial Query + [TOP] + Article's Title + [CON] + Expanded Query

where the initial query is the query specified by the organizers, the Article's Title corresponds to the topic text, and the Expanded Query is the context generated by LLaMA3. For example, the input query for topic G11.1 would be:

drones + [TOP] + UK wants new drones in wake of Azerbaijan military success + [CON] + UK military drones Nagorno-Karabakh conflict Azerbaijan Armenia.

Documents in the collection are represented as 'title + [ABS] + abstract'. In our fine-tuning process, three special tokens {TOP, CON, ABS} are included to separate different text types. After fine-tuning the cross-encoder model, we re-rank the top-100 results retrieved by the bi-encoder model.

- **Re-ranking with LLaMA (LLaMA Re-Ranker)**: While we used LLaMA3 for query expansion for our two first runs, for our next two runs, we used it as a pairwise re-ranker. Following the approach proposed by Qin et al. [11], we used a system prompt for pair-wise re-ranking shown in Table 9 (Appendix).
  Two variations of this architecture were implemented, differing in the user message provided to LLaMA3. In one version, the user message included the query, related article title, and context generated from the previous runs (i.e., the expanded query from the LLaMA3). The other version omitted the context.
  Essentially, LLaMA3 was Tasked with determining which of the two documents was more relevant to the query based on the provided information. We re-ranked the top-100 candidates retrieved by the bi-encoder model. Since LLaMA3's outputs in this context might not be suitable for direct confidence scores, we assigned a simple ranking based on enumeration. The highest-ranked document received a score of 100, with scores decreasing by 1 for lower ranks.
- **Fine-Tuned Cross-Encoder combined with ElasticSearch (CERRF)**: Our last run leverages ElasticSearch, provided by the organizers. We first retrieve the top-100 results for each topic using a combination of the query and topic text. Subsequently, we re-rank these results using a fine-tuned cross-encoder 'ms-marco-MiniLM-L-6-v2'. For fine-tuning, the training data from previous labs was used. We represented each input query as "<query> [QSP] <topic text>", while the papers were represented as "<title> [TSP] <abstract>". Here, [QSP] and [TSP] are special tokens separating the query text from the topic text and the paper title from its abstract, respectively. To select optimal hyperparameters, topics G10 and G11 were chosen for validation. The 2023 test set was used for the final evaluation. After hyperparameter selection, the model was fine-tuned on all available training topics.
  In addition to the cross-encoder approach, we also perform a separate retrieval using Elasticsearch with only the query (without the topic text). The results from both methods are then combined using the modified Reciprocal Rank Fusion (MRRF) technique [12] as EQ.1, where d is the document, s_m and r_m are the model's similarity score and the rank, respectively. The underlying principle of MRRF is that documents ranked highly by both retrieval methods are likely more

Table 2
AIIRLab systems results for CLEF 2024 SimpleText Task 1 on the test Qrels (G01.C1-G10.C1 and T06-T11).

Model	MRR	P@10	P@20	NDCG@10	NDCG@20	Bpref	MAP
LLaMABiEncoder	**0.9444**	**0.8167**	0.5517	**0.6170**	**0.5166**	**0.3559**	**0.2304**
LLaMAReranker2	0.9300	0.7933	0.5417	0.5943	0.5004	0.3495	0.2177
LLaMAReranker	0.8944	0.7967	**0.5583**	0.5889	0.5011	0.3541	0.2200
LLaMACrossEncoder	0.7975	0.6933	0.5100	0.4745	0.4240	0.3404	0.1970
CERRF	0.7264	0.5033	0.4000	0.3584	0.3239	0.2204	0.1309

Table 3
Evaluation of AIIRLab systems for complexity and credibility in Task 1 (over all 176 queries) .

Model	Avg. #Refs	Avg. Sentence Length	Avg. Syllabus per Word
LLaMABiEncoder	9.5	31.0	1.865
LLaMAReranker2	8.6	20.9	1.707
LLaMAReranker	8.8	22.1	1.772
LLaMACrossEncoder	10.0	30.6	1.890
CERRF	10.6	22.0	1.895

relevant than those ranked highly by only one method.

$$RRFscore(d \in D) = \sum_{m \in M} \frac{s_m(d)}{60 + r_m(d)} \tag{1}$$

2.3. Experimental Results and Analysis

Table 2 shows the effectiveness of our proposed models, reported by the organizers. Except for P@20, the *LLaMABiEncoder* archives the highest effectiveness across all measures. Another aspect evaluated in Task 1, is credibility and text complexity, for which the results from our systems are shown in Table 3.

Looking at the *LLaMABiEncoder* results, for only 10% of topics, the MRR value is not 1. The lowest MRR is for the topic ‘G02.C1’, at 0.33 (P@10 of 0.7). For this topic, the query text by the organizers is defined as “concerns related to the handling of sensitive information by voice assistants”. With LLaMA3, the expanded query is “voice assistants handling sensitive information concerns Apple Siri recordings”, does not seem to add any new useful terms to the original query. The top retrieved results for this topic is an article titled, “Poster: A First Look at the Privacy Risks of Voice Assistant Apps.”, assessed as non-relevant. For topics like ‘T11.1’ the original query “character relationship” is expanded to “character relationship network map The Witcher”, helping find more relevant results, leading to MRR and P@10 of 1.

Comparing our LLaMA3 re-ranking approach system, *LLaMAReranker2* against *LLaMABiEncoder*, there is no significant difference between the two systems, using Two-sided Paired Student’s t-Test (p-value=0.05). Interestingly, both models have the same topics for which they did not achieve MMR of 1. For topic ‘G02.C1’, the MMR drops to 0.2 with *LLaMAReranker2* (P@10 of 0.3). Investigating the results for this topic, LLaMA3 gave higher ranks to articles that have only titles (abstract missing) such as the article titled “Examining the Use of Voice Assistants: A Value-Focused Thinking Approach”. With the article’s abstract missing, these articles are assessed as non-relevant. Overall, using LLaMA3 for either re-ranking or query expansion showed similar effectiveness, while re-ranking with a bi-Encoder proved more efficient.

3. Task 2: Identifying and Explaining Difficult Concepts

This section describes the data for Task 2 [4], our proposed models, and evaluation results. We rely on LLaMA3 and Mistral [2] language models and propose three systems for Subtasks 1 and 2.

3.1. Training and Test Data:

For Task 2, 576 sentences from 115 documents are provided for training. For these sentences, 2590 annotated difficult terms are available. Subtask 2.2 leverages a dataset of 501 sentences across 55 documents, containing 2,006 explanations and 1,521 definitions. These documents are selected from high-ranked abstracts to the requests of Task 1. Participants are asked to detect difficult terms, along with the difficulty level for Subtask 2.1, and provide definitions and explanations of detected difficult terms for Subtask 2.2 [1].

3.2. Proposed Models

Our team participated in Task 2, with three proposed systems, based on LLaMA3 and Mistral. Here we describe our models:

- **LLaMA:** Our first model uses LLaMA3-8B-Instruct, using a system prompt to instruct the model to act as a knowledgeable high school student (details in Table 10). This prompt achieved the best performance among those studied on the training data. We process each sentence from the test set using the following user message:

  > For the sentence: SENTENCE, what are difficult terms (one to five consecutive terms)? What is the difficulty level? Your output is term or terms: difficulty level (e, m, or d). Do not provide explanation, just give the answer.

  where SENTENCE represents the actual sentence. We specify the output format, as LLaMA can add unnecessary information. After identifying difficult terms, we again utilize LLaMA to generate definitions and explanations. As shown in Table 10, we instruct LLaMA to act as a technician with knowledge of technical terms and request definitions and explanations. The following user message is used for this step:

  > You have identified term "TERM" in the sentence: "SENTENCE" as an unclear term. Provide its definition and explain what it is. The output should be like:
  >
  > Definition: Give definition here, Explanation: Give explanation here

  where TERM represents the term identified earlier and SENTENCE is the sentence it originated from.
- **LLaMA Fine-tuned (LLaMAFT):** Our second approach is based on prompt engineering and reinforcement learning with human feedback to improve the quality of outputs generated by the LLaMA model. We designed several models to enhance the feedback loop, ultimately aiming for better results. Our exploration resulted in three distinct models, shown in Figure 1. Our models differ in how the *user* and *system* messages are sent to LLaMA. Table 11, shows the order of prompts used for each model. Each model mainly follows a two-step process:
  - *Step 1:* After instructing model based on the prompts in Table 11, the user message is based on the sentence and in some cases, by incorporating human-annotated data (output) from training data. This output represents the desired outcome for the Task, including identified difficult terms and their corresponding definitions and explanations.
  - *Step 2:* Using the generated result by LLaMA from Step 1, and a new *user* prompt, a second round of results is produced.

  Each model was studied with different combinations of training data and prompts. Through our experiments, Model M3 outperformed other approaches and was used as our second run.

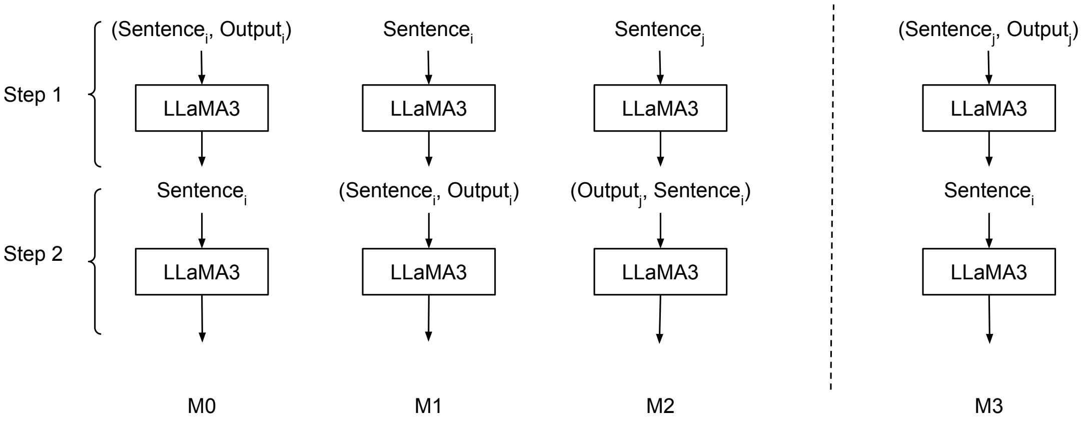
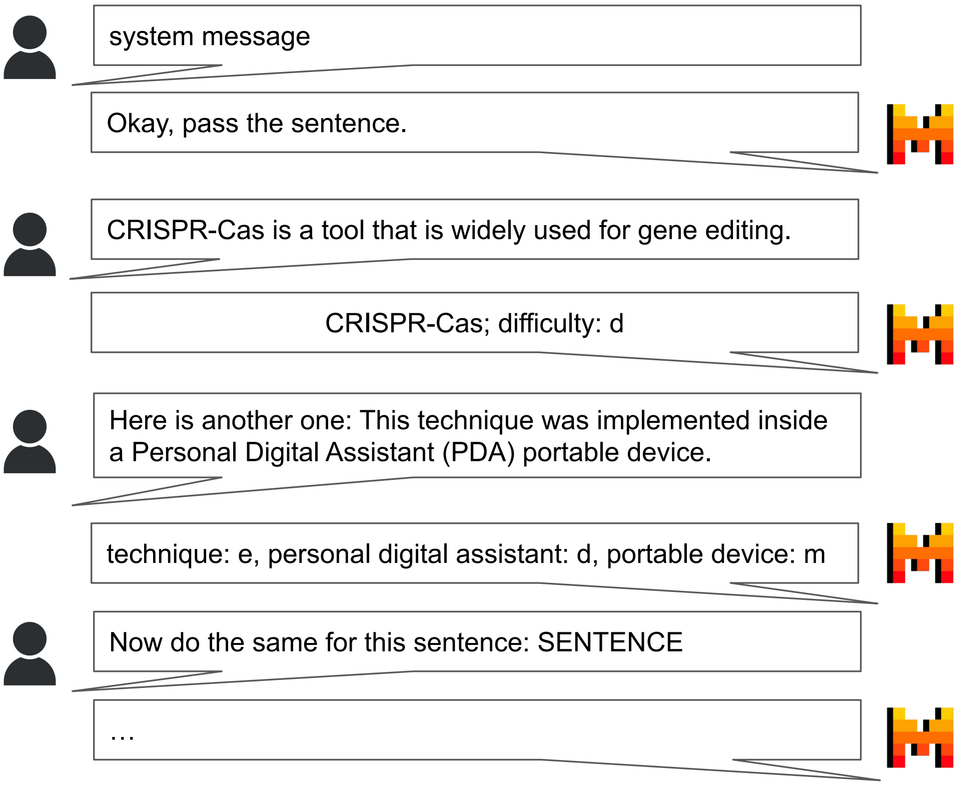


Figure 1: Our studied approaches for using LLaMA3 for Task 2. Sentence represent the input sentences for which difficult terms should be extracted and defined. The output shows human-annotated data, including the extracted terms, their definitions, and explanations. Four different approaches were studied; Model M3's consistent high-quality performance on the training data makes it the preferred choice for further evaluation and testing phases.

- **Mistral:** Similar to our LLaMA3-based model, our approach with Mistral-7B leverages a system prompt (details in Table 10). This prompt instructs Mistral to identify difficult terms. We process training examples through a series of prompts and responses with Mistral to achieve this. Figure 2 illustrates the process in which we represent several ground truths to the model. The examples used in the figure come from the training data, and the SENTENCE represents the test sentence being analyzed. After detecting the difficult terms, we use the similar system prompt (shown in Table 10) as our first model to generate definitions and explanations of difficult terms with Mistral.



system message

Okay, pass the sentence.

CRISPR-Cas is a tool that is widely used for gene editing.

CRISPR-Cas; difficulty: d

Here is another one: This technique was implemented inside a Personal Digital Assistant (PDA) portable device.

technique: e, personal digital assistant: d, portable device: m

Now do the same for this sentence: SENTENCE

…

Figure 2: Prompts used for Subtask 2.1 to extract difficult terms. The SENTENCE represents the test sentence passed to Mistral, and the final response from Mistral provides the identified difficult terms.

Table 4
AIIRLab systems results for CLEF 2024 SimpleText Task 2.

Model	Recall	Precision	Rec_Difficult	Prec_Difficult	Blue
Mistral	0.41	0.69	0.19	0.49	0.13
LLaMA	0.28	0.65	0.26	0.67	0.15
LLaMAFT	0.01	0.99	0.00	1.00	0.12

Table 5
Extracted difficult terms with their difficulty levels for sentence ID ‘G08.1_2972302621_1’ from SimpleText 2024. Letters ’d’, ’m’, and ’e’ show difficult, medium, and easy terms, respectively.

Ground-truth		Mistral		LLaMA	
Term	Difficulty	Term	Difficulty	Term	Difficulty
cryptocurrency	m	cryptocurrency	d	cryptocurrency	d
digital currency	m	digital currency	m	digital currency	m
capital management	m	capital management	m	derivatives	m
nonmonetary applications	d	nonmonetary applications	m		
financial transactions	e	financial transactions	e		

3.3. Experimental Results and Analysis

Our proposed systems results on the test set are summarized in Table 4. For each run, the organizers reported:

- Recall of all the terms, independently from the level of difficulty
- Precision of all the terms, independently from the level of difficulty
- Recall of the difficult terms
- Precision of the difficult terms
- BLEU score computed for bigrams

Our proposed Mistral approach provided better results compared to LLaMA3. Providing an example, for the sentence “Cryptocurrency was built initially as a possible implementation of digital currency, then various derivatives were created in a variety of fields such as financial transactions, capital management, and even nonmonetary applications.” (sentence ID: G08.1_2972302621_1), Table 5 shows the ground-truth, and the results generated by Mistral and LLaMA, for Subtask 2.1. As can be seen, LLaMA tends to extract fewer terms for each sentence, leading to lower recall; however, the precision for correctly identifying difficulty level is more precise.

Another interesting aspect of Task 2 is duplicate sentences. The organizers have provided repeated sentences to study whether LLMs provide the same results. Our results show while Mistral mostly produces the same responses, LLaMA3 responses seem to differ each time. For a short sentence, “This is especially true for self-driving vehicles deployed in public transport services.”, LLaMA3 once extracts the terms ‘self-driving’, ‘vehicles’, ‘public transport’ and the next time extracts ‘self-driving’, ‘deployed’. Mistral extracted terms, however, remained the same.

Note on LLaMAFT Run: We have identified a mistake while submitting this run. Our studies for different models (M0 to M3) used a two-stage process of first identifying the difficult terms and then generating the definitions. In our submitted model for the test data, we mistakenly used a single prompt for all the Subtasks. Upon correction, including previous related documents and human answers improved the results (Precision: 0.28, Recall: 0.41).

4. Task 3: Simplify Scientific Text

This section describes the data, proposed models, and evaluation results for Task 3 [5]. LLaMA3-8B-Instruct and Mistral were utilized for both Subtasks 3.1 and 3.2.

Table 6
Task 3.1 training data example. The source shows the input sentence, and the target is its simplified version.

snt_id	G06.2_855132903_1
source	In this paper we present queuing-theoretical methods for the modeling, analysis, and control of autonomous mobility-on-demand MOD systems wherein robotic, self-driving vehicles transport customers within an urban environment and rebalance themselves to ensure acceptable quality of service throughout the network.
target	Queuing models are used for autonomous mobility-on-demand MOD systems. A queuing model is constructed so that queue lengths and waiting time can be predicted. In MOD systems, robotic, self-driving vehicles transport customers within an urban environment and rebalance themselves to ensure quality of service.

4.1. Topic and Collection

The training data consists of a collection of parallel text passages (source and simplified versions). These simplified sentences are directly created from original scientific abstracts in the DBLP Citation Network Dataset for Computer Science, Google Scholar, and PubMed articles on Health and Medicine (all from 2023). The dataset includes 648 sentences for training and 245 sentences for testing. The simplification process involved either master's students in Technical Writing and Translation or a team of a computer scientist and a professional translator (native English speaker). An example of this source (original) and target (simplified) sentence pair is provided in Table 6.

4.2. Proposed Models

AIIR Lab submitted a total of four runs for both Subtasks 3.1 (sentence-level) and 3.2 (abstract-level). Three of the runs utilized a fine-tuned LLaMA3-8B model and one used Mistral, with prompt engineering. Our proposed approaches are as follows:

- **Prompt Engineering with Instruction-tuned LLaMA3-8B:** Our first three runs for this Task utilized LLaMA3-8B which was instruction-tuned with the provided training data for both the sentence and abstract levels. We used a split of 90:10 for training and validation. For instruction tuning with LLaMA, we used Quantized Low-Rank Adaptation (QLoRA). QLoRA, as shown in Figure 3, is a method used for fine-tuning processes to reduce the amount of memory required and computational cost [13]. The model's weights are first converted from 16-bit floating point numbers to 4-bit NormalFloat. These reduced-size weight matrices are then approximated to low-rank matrices by reducing the number of parameters, speeding up computation time, and reducing the data footprint. These 4-bit embeddings then utilize NVIDIA's *unified memory* feature, which allows for automatic paging optimization before updating the weights. This paging optimization allows for the CPU RAM to be accessed by the GPU directly for page-to-page transferring, preventing the possibility of running out of GPU memory space as long as sufficient system memory is available.
  During the training process, the data was first run through QLoRA for the token embeddings to be resized. The hyperparameters are set as follows: an *alpha* of 32, a *dropout* of 0.1, a *Task type* of "CASUAL_LM" and an *R-value* of 8. The output data was then fed to LLaMA3-8B with the hyperparameters of a *learning rate* of e-4, a paged_adam_32 *optimization function*, 20 *epochs* and a *batch size* of 8.
  As shown in Table 6 each entry for the data is paired into *source* and *target* values. We passed training data for LLaMA3 instruction-tuning as:

  "Instruction:" + [P] + "Input: " + [S] + "Response: " + [T]

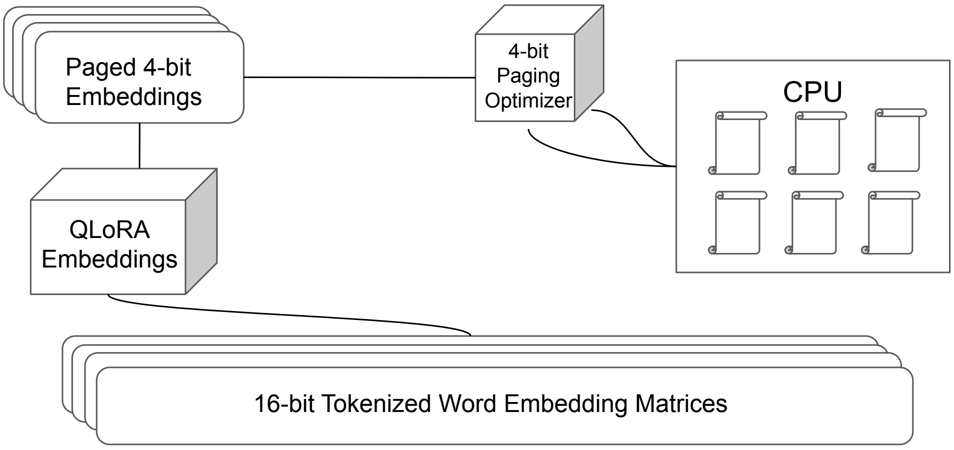


Figure 3: QLoRA embedding and paging pipeline.

where prompt (P), for all training samples, was the one used for Run 1 (Table 12). The source (S) and target (T) values would be the output token embeddings from QLoRA. We believe this would give LLaMA3 a better understanding of the linguistic styles in the desired target simplifications. For prompt engineering, we focused on the average FKGL (Flesch-Kincaid Grade Level) score for the provided test sentences and abstracts. The data was passed into our instruction-tuned model and an FKGL score was averaged at the end of each run.

- **Mistral (RUN 4):** Using Mistral 7B, we used the system prompt as shown in Table 12. We then used three sample sentences from training data, along with their simplified versions, to provide examples for Mistral. As our final user message, we passed the test sentence/abstract to mistral with the prompt:

  > Now do the same for this text, simplify by explaining technical terms or replacing them with easier words without removing context: TEXT

  where TEXT is the input sentence/abstract.
  Note: While submitting this run, we only evaluated the model on the training data by mistake. Therefore, this run was excluded from the evaluation.

4.3. Experimental Results and Analysis

Task 3 results are evaluated based on several metrics, with SARI [14] score against the human reference simplifications as the main measure. Table 7 shows our results for both Subtasks 3.1 (sentence-level) and 3.2 (abstract-level). While for Subtask 3.1, our team runs are ranked second in terms of SARI score, we achieved the highest SARI score for Subtask 3.2 between the participating teams. For the level of text complexity, the FKGL readability measure is used. Compared to the references, our models have high compression ratios and sentence splits, as LLaMA's outputs are lengthier. An example of this is shown in Table 8, where our simplified version of the original input text is compared against the ground-truth and for Subtask 3.1.

For Subtask 3.1, all LLaMA3's Sari scores fell within a ± 0.82 difference from one another. The Sari scores for Subtask 3.2 were similar to Subtask 3.1, in that, they varied by a relatively narrow margin of ± 1.25. The original sentences have an FKGL of 13-14 corresponding to a university-level text, with the reference scores being 8.86 for Subtask 3.1 and 8.91 for Subtask 3.2. Our FKGL results for all runs in both Tasks fell within the 8.39 to 10.33 FKGL range, with our run 1 scores being 0.47 points below for Task 3.1 and 0.16 points above for Task 3.2 compared to the reference FKGL score.

Table 7
AIIRLab systems results for CLEF 2024 SimpleText Task 3.

	Subtask 3.1			Subtask 3.2		
Model	FKGL	SARI	BLEU	FKGL	SARI	BLEU
LLaMA3-8B Run1	**8.39**	**40.58**	**7.53**	**9.07**	**43.44**	**11.73**
LLaMA3-8B Run3	9.47	40.36	6.26	10.17	43.21	11.03
LLaMA3-8B Run2	10.33	39.76	5.46	10.22	42.19	7.99

Table 8
Our results for sentence-level simplification for sentence ID ‘G01.1_1552637960_1’ in Subtask 3.1.

Original text	The goal of the MOST project is to develop a novel, inexpensive, easy-to-use digital talking device for blind and visually impaired users based on off-the-shelf handheld computers (Personal Digital Assistant).
Simplification system	**Simplified Result**
Ground-truth	The goal of the MOST project is to create a new talking device for blind people.
LLaMA3-8B Run 1	The MOST project aims to create a simple, affordable, and easy-to-use digital talking device for blind and visually impaired people using ordinary handheld computers.
LLaMA3-8B Run 2	The goal of the MOST project is to create a simple, affordable, and easy-to-use digital device that can talk to blind and visually impaired people using handheld computers.
LLaMA3-8B Run 3	The MOST project aims to create a simple, affordable, and user-friendly digital talking device for blind and visually impaired people using common handheld computers.

5. Conclusion

AIIR lab participated in SimpleText CLEF 2024 lab Tasks 1 to 3, relying on large language models, namely LLaMA3 and Mistral. In Task 1, we submitted five runs leveraging LLaMA for query expansion, TF-IDF for candidate selection, and both bi-encoder and fine-tuned cross-encoder models for re-ranking. We also explored LLaMA for re-ranking within this Task. Our bi-encoder model and LLaMA re-ranker models were the most effective systems among the participating teams. For Task 2, we had three runs, using LLaMA and Mistral. Our Mistral-based model provided better effectiveness compared to LLaMA, providing higher recall and precision in detecting difficult terms. However, LLaMA model was better at detecting difficulty levels. Finally, for Task 3, we participated in both Subtasks, submitting four runs that employed LLaMA and Mistral. Our LLaMA models had high SARI scores for Subtasks 3.1 and 3.2. For future work, we aim to explore large language models further for these Tasks, incorporating techniques such as chain-of-thoughts to study the effectiveness of these models for the related Tasks.

References

[1] L. Ermakova, et al., Overview of CLEF 2024 SimpleText track on 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] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. d. l. Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, et al., Mistral 7B, arXiv preprint arXiv:2310.06825 (2023).

[3] 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.

[4] G. M. D. Nunzio, et al., Overview of the CLEF 2024 SimpleText task 2: Identify and explain difficult

Table 9
System prompts used for query expansion and re-ranking with LLaMA3 for Task 1.

Task	Prompt
Query Expansion	Being a ranking model your first Task is to do query expansion. For an information need, you will add more context to it. Contextualize the query as best as you can in one or two short sentences, for a given information need and context.
Re-ranking	You are a ranking model for information retrieval. Given a query and two documents, you will say which one is more relevant. If Document 1 is more relevant say yes, otherwise say no.

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.

[5] 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.

[6] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, Z. Su, ArnetMiner: Extraction and Mining of Academic Social Networks, in: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, 2008.

[7] A. Anand, V. Setty, A. Anand, et al., Context Aware Query Rewriting for Text Rankers using LLM, arXiv preprint arXiv:2308.16753 (2023).

[8] C. Macdonald, N. Tonellotto, Declarative Experimentation in Information Retrieval using PyTerrier, in: Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval, 2020.

[9] N. Reimers, I. Gurevych, Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, 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), 2019.

[10] B. Mansouri, S. Durgin, S. Franklin, S. Fletcher, R. Campos, AIIR and LIAAD Labs Systems for CLEF 2023 SimpleText., in: CLEF (Working Notes), 2023.

[11] Z. Qin, R. Jagerman, K. Hui, H. Zhuang, J. Wu, L. Yan, J. Shen, T. Liu, J. Liu, D. Metzler, et al., Large Language Models are Effective Text Rankers with Pairwise Ranking Prompting, in: Findings of the Association for Computational Linguistics: NAACL 2024, 2024.

[12] B. Mansouri, D. W. Oard, R. Zanibbi, DPRL Systems in the CLEF 2022 ARQMath Lab: Introducing MathAMR for Math-Aware Search, Proc. CLEF 2022 (CEUR Working Notes) (2022).

[13] T. Dettmers, A. Pagnoni, A. Holtzman, L. Zettlemoyer, QLoRA: Efficient Finetuning of Quantized LLMs, Advances in Neural Information Processing Systems 36 (2024).

[14] W. Xu, C. Napoles, E. Pavlick, Q. Chen, C. Callison-Burch, Optimizing Statistical Machine Translation for Text Simplification, volume 4, 2016.

A. Prompts

This section shows the prompts used for SimpleText lab for the Tasks we participated in. For query rewriting/expansion and re-ranking, we used the system prompts shown in Table 9 with LLaMA3. For Task 2, Table 10 shows the system prompts that we used for Subtasks 1 and 2. Table 11 shows our prompts for fine-tuning LLaMA for Task 2. Finally, Table 12 shows our prompts for Task 3.

Table 10
System prompts used for detecting difficult terms and generating definitions and explanations with LLaMA3 and Mistral for Task 2.

Model	Prompt
Task 2.1 (Detecting Difficult Terms)	
LLaMA3	You are a high school student with good general knowledge. Given a sentence, you want to determine which terms are not clear . You choose the terms that should be defined in order to understand the sentence. This includes technical terms and abbreviations. You can choose one to five consecutive terms. You will also decide the difficulty level for each identified term, with labels easy (label e), medium (label m), hard (label d). Here is an example For sentence: CRISPR-Cas is a tool that is widely used for gene editing; you identified "CRISPR-Cas" with difficulty: d
Mistral	You are a helpful assistant. Given a sentence, you will just output the unclear technical term or terms (up to 5 terms). You choose the terms that should be defined in order to understand the sentence. Each sentence can have up to 5 phrases. You will decide the difficulty of unclear terms with scales easy (e), medium (m), hard (d). Note that easy does not include terms such as shown, pronouns, or numbers
Task 2.2 (Definition and Explanation)	
LLaMA3 & Mistral	You are a technician with knowledge of technical terms. Given a term, in a sentence provide definition of it. Then provide an explanation of that term. Your goal is to make sure other non technicians understand the sentence. Definition and explanation should be separate from each other

Table 11
System prompts used for LLaMA3-8B for Task 2 (second run). In our prompts, Human refers to human-annotated data from the training set.

Model	Prompts
M0	*Instruction:* Extract complex words from sentence, generate only one definition for each complex word. **System:** Human answer to find complex term is $\{Human\_answer\}_i$, Human definition $\{Human\_def\}_i$, Human positive definition $\{Human\_pos\}_i$, Human negative definition $\{Human\_neg\}_i$. **User:** $\{Sentence_i\}$ {instruction} **User:** $\{Sentence_i\}$ {instruction}
M1	*Instruction:* Extract complex words from sentence, Do not generate long text. **User:** $\{Sentence_i\}$ **System:** Human answer to find complex term is $\{Human\_answer\}_i$, Human definition $\{Human\_def\}_i$. **User:** $\{Sentence_i\}$ {instruction}
M2	*Instruction:* Extract difficult words from sentence, Do not generate long text. **User:** $\{Sentence_j\}$ **System:** Human answer to find complex term is $\{Human\_answer\}_j$ with difficulty $\{diff\}_j$. **User:** $\{Sentence_i\}$ {instruction}

M3	*Instruction:* Extract complex words from sentence, Do not generate long text.
	System: Human answer to find complex term is $\{Human\_answer\}_j$.
	User: $\{Sentence\}_j$ {instruction}
	User: $\{Sentence\}_i$ {instruction}

M3-Test	*Instruction:* Extract complex words from sentence, and label difficulty of word with one of 'e' means easy, 'm' means medium, 'd' means difficult, and then generate a definition for each complex word based on sentence, generate an explanation for each complex word.
	System: Human answer to find complex term is $\{Human\_answer\}_j$, and difficulty $\{difficulty\_list\}_j$, Human definition $\{Human\_def\}_j$, Human good definition $\{Human\_pos\}_j$, Human wrong definition $\{Human\_neg\}_j$.
	User: $\{Sentence\}_j$ {instruction}
	User: {Test Sentence} {instruction}

Table 12
System prompts used for Task 3.

Model	Prompt
LLaMA3-8B Run 1	Simplify this text for English speaking science students in college. Maximize the use of simple words and short sentences, but include keywords from the original text. Optimize the output ROUGE, SARI, and BLEU scores
LLaMA3-8B Run 2	You are a skilled editor, known for your ability to simplify complex text while preserving its meaning. You have a strong understanding of readability principles and how to apply them to improve text comprehension.
LLaMA3-8B Run 3	Simplify the following scientific text for an average American citizen. Keep, but define, any keywords and subjects with less complex words and phrases.
Mistral	You are a skilled editor, known for your ability to simplify complex text while preserving it. You explain the technical terms, defining what they are (e.g., terms like Blockchain, Cryptojacking, all abbreviations), without removing sentences or summarizing them.