Overview of the Sensemaking Task at the ELOQUENT 2025 Lab: LLMs as Teachers, Students and Evaluators

Pavel Šindelář[1,*], Ondřej Bojar[1,*]

[1] Charles University, Prague, Czech Republic

Abstract

ELOQUENT is a set of shared tasks that aims to create easily testable high-level criteria for evaluating generative language models. Sensemaking is one such shared task.

In Sensemaking, we try to assess how well generative models "make sense out of a given text" in three steps inspired by exams in a classroom setting: (1) Teacher systems should prepare a set of questions, (2) Student systems should answer these questions, and (3) Evaluator systems should score these answers, all adhering rather strictly to a given set of input materials.

We report on the 2025 edition of Sensemaking, where we had 7 sources of test materials (fact-checking analyses of statements, textbooks, transcribed recordings of a lecture, and educational videos) spanning English, German, Ukrainian, and Czech languages.

This year, 4 teams participated, providing us with 2 Teacher submissions, 2 Student submissions, and 2 Evaluator submissions. We added baselines for Teacher and Student using commercial large language model systems. We devised a fully automatic evaluation procedure, which we compare to a minimalistic manual evaluation.

We were able to make some interesting observations. For the first task, the creation of questions, better evaluation strategies will still have to be devised because it is difficult to discern the quality of the various candidate question sets. In the second task, question answering, the LLMs examined overall perform acceptably, but restricting their answers to the given input texts remains problematic. In the third task, evaluation of question answers, our adversarial tests reveal that systems using the LLM-as-a-Judge paradigm erroneously rate both garbled question-answer pairs and answers to mixed-up questions as acceptable.

Keywords

large language models, generative language models, evaluation, quality assessment, text comprehension, CLEF-WS

1. Introduction

Large language models (LLMs) have been shown to have a remarkable ability to summarize text, answer questions, and evaluate task performance. In the Sensemaking shared task run as part of the ELOQUENT Lab [1] [1] at CLEF 2025, we focus on a more constrained setting. We want to determine the capacity of an LLM-based system to "make sense" of a given material, adding yet another possible approach to machine reading or text comprehension. We assume, that the ability of a system to create questions that assess the understanding of arbitrary material is an indication that the system understands the material. We specify arbitrary here because many systems can create questions for material simply by reflecting on questions they have seen generated for similar material, which does not necessitate understanding the material, such generation is not possible for any arbitrary piece of material. The same reasoning as given above for question generation can be extended to the ability to answer arbitrary questions about materials and the ability to rate such answers given material. In general, our task could be attempted with a system using any generative language model. However, we correctly assumed that all teams would use LLMs in this first edition, so the evaluation is mostly focused on the capabilities and limitations of LLMs specifically.

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

[*] Corresponding author.
sindelar@ufal.mff.cuni.cz (P. Šindelář); bojar@ufal.mff.cuni.cz (O. Bojar)
0000-0002-0606-0050 (O. Bojar)

[1] https://eloquent-lab.github.io/

Our experiments come from two domains: fact-check analysis of potentially disinformative texts and a reasonably broad range of educational topics. The educational domain is of particular interest because our evaluation rubric (pose questions, answer them, evaluate various responses) is very similar to what educators are regularly doing. The use of generative language models in the domain of education is an area that has been studied extensively [2, 3, 4]. However, most of this research has focused on the ability of models to answer questions and finish assignments. Typically, models were not tested for the ability to exclusively rely on information from given material, but only for whether they were able to answer at all. We see such question answering as an insufficient method for evaluating the ability of a model to make sense of arbitrary material. For evaluation of abstractive question answering from context, there are not many procedures that do not rely on human-provided reference answers, and there is often no provision made to test whether the model is actually retrieving answers from the material rather than emitting its general knowledge. There is also a lack of evaluation procedures for the task of generating questions that comprehensively cover a given input text.

If we were creating an evaluation metric for educator tools or deploying a generative language model in the domain of education, we would need to cater to many requirements, such as adherence to curricula, ability to mimic the kind of questions and evaluations culturally expected, etc. For the sake of evaluating only basic material understanding as a starting point, we chose only very simple requirements on fluency, validity, adherence to context, and several other measures described in Section 4. In the end, we did not have to test for fluency because most models used almost universally produced fluent text.

The paper is structured as follows: In Section 2, we describe the organization of the shared task and baseline systems. Section 3 provides a summary of the underlying texts. Sections 4 to 6 provide the evaluation methodologies and results for the obtained **Teacher**, **Student**, and **Evaluator** systems, respectively. A brief discussion of our observations of the LLM-as-a-Judge usecase across these tasks is provided in Section 7. We conclude in Section 8 and list the limitations of this work in Section 9.

2. Experimental setup

The Sensemaking shared task in 2025 had three tracks, all voluntary:

1. **Teacher** systems were given input materials organized into sections and were expected to generate quizzes as lists of questions for each such section of input material. In addition to each question, they were asked to provide reference answers, if possible, to aid in the evaluation. The baselines for the **Teacher** track included questions generated, extracted from the original materials, or given by experts. Because such a baseline is not available for all materials, we also included baseline questions generated by a system using `GPT-4.1-nano-2025-04-14`.
2. **Student** systems were given input materials and quiz questions and were expected to provide answers to the questions (based on the input materials rather than general knowledge). The baselines for the **Student** task included golden answers to questions extracted from the original materials (where available) and answers generated by a system using `GPT-4.1-nano-2025-04-14`.
3. **Evaluator** systems worked with input materials, a question, and an answer, and were expected to score the answer on a scale of 0 to 100. We again used `GPT-4.1-nano-2025-04-14` as an **Evaluator** baseline and also measured the extent to how the **Evaluator** systems correlate with .

A total of four teams participated, each in various tasks. Three teams submitted valid submissions for two tasks, while one team focused only on **Evaluator** submission. Some submissions contained invalid output for too much of the dataset and were not included in the evaluation.

The data span 4 languages and 3 modalities. For the sake of the participants, all data were also provided in plaintext English form. Texts in languages other than English were machine-translated to English using the Google Cloud Translation - Basic (v2) system. We used the `whisper-turbo` model provided by the `openai-whisper 20240930` system for automatic speech recognition of speech input materials. All teams worked only with English data. Only one team chose to use the PDF modality in addition to plain text, using a system similar to [2].

A very small subset of data was released as a development set prior to the evaluation period to illustrate the task to the participants: only one section for three kinds of materials. This was considered satisfactory since the only goal was for the participants to get an understanding of the kind of input their systems will need to handle. Providing such a small development set allowed us to have a large, mostly uncorrelated test set. There were no recommended training sets. Although we did some small-scale experiments with a few datasets, such as SQuAD [5], we were unable to find any dataset that we were confident in recommending.

3. Data

The material and questions were sourced from freely-accessible websites, YouTube, a university lecture, a study, and an academic book. The answers were acquired from one study and one non-copyrighted book website's administrators. To maintain copyright restrictions, we do not release our processed form of the dataset publicly, but we can grant access to it upon request, as we did for registered task participants.

There were a total of 7 data sources (referred to as material kinds in the paper).

1. A database of public statements by Czech politicians that were fact-checked by the Demagog.cz project. An example of a datapoint constructed from this database can be seen in Figure 1. (demagog-statements-public)
2. A 19th-century book providing alternative logical and theological arguments about many natural phenomena (these views are not endorsed by the authors). (flat-earth-book)
3. An English neural machine translation publication [6]. (nmt-book)
4. Audio from video recordings of a university lecture given by Ondřej Bojar. (nmt-class)
5. A selection of various popular science material and European parliament speeches in German. The questions are in Czech and provided by [7]. (popular)
6. Ukrainian high-school textbooks on biology (ukr-biology). These materials had many questions, and the individual sections were very long. However, we were unable to get hold of reference answers. (ukr-biology)
7. English world history textbook from an open education resource (OER) website. For educators, reference questions and answers were also available. The textbook had many additional information subsections that were not related to the main body of information; these subsections had to be removed for the sake of brevity and text continuity. (world-history)

For the **Student** and **Evaluator** systems, we also constructed adversarial inputs from both the material and the other systems' outputs, as described in Figure 2.

Material section (truncated): The former government ... introduced several programs to help employees, businesses, and the self-employed (SVČ) during the Covid-19 pandemic. As for the current ... compensation programs, the Ministry of Labor and Social Affairs ... includes , for example, ... the Antivirus A and ... Antivirus B programs. The purpose of the first of these is ... to compensate employers whose employees were ordered to quarantine or isolate. Antivirus B was to ... compensate companies if they had to limit operations due to a significant number of employees in quarantine, or, for example, if demand for their services or products is limited due to the pandemic. For injured self-employed persons, but also for partners of small companies and people working on the DPČ and DPP, there is a ... Compensation Bonus program, which ... falls under the Ministry of Finance. For ... example, the Ministry of Industry and Trade was supposed to be responsible for ... the COVID 2021 subsidy program and the ... COVID Uncovered Costs program, ... intended for companies with a significant drop in sales. Although they were already prepared, the government ultimately decided ... We therefore assess Marian Jurečka... statement as true.
Question automatically generated from database (truncated): Determine whether the statement relevant to the given summary is TRUE or FALSE based on a summary and a number of statements. Only one of the following statements is relevant to the summary:
SECTION A: The mechanism (increase in pensions, note: Demagog.cz) is given by law (...), if since the last month taken for the ...
SECTION B: The proposal that I presented last year contained approximately 11, 11 measures in total. It also concerned widows, ...
SECTION C: So, the tools of assistance (compensation for entrepreneurs, note: Demagog.cz) are shared here today mainly by 3 ministries, that is the Ministry of Labor and Social Affairs, for us it is the programs of Antivirus (...), and then we also share the tools of assistance with the Ministry of Industry and Trade and the Ministry of Finance, where the other tools of assistance for entrepreneurs and tradesmen are.
Respond only with either TRUE or FALSE and the correct section name, examples: TRUE, SECTION A.
Reference answer: TRUE, SECTION C **Rating by Gemma-3-27b:** 100.00
Answer by baseline GPT-4.1-nano: TRUE, SECTION C **Rating by Gemma-3-27b:** 100.00
Answer by Llama-3-8b: <|assistant|> *newline* FALSE *newline* SECTION C. **Rating by Gemma-3-27b:** 100.00
Answer by Llama-2-7b plus DeepSeek-R1-Distill-Qwen-1.5b: FALSE, SECTION C **Rating by Gemma-3-27b:** 100.00

Figure 1: A sample test item (i.e., the input material and a question) derived from the Demagog database with answers and ratings by competing systems. We underlined several spans in the input material and in the question to highlight what the models should have ideally noticed in order to answer correctly. Only baseline GPT-4.1-nano managed this, so the automatic ratings by Gemma-3-27b are also mostly wrong.

Material section (truncated): Pirate Party National consists of an election committee and an election coordinator. The functioning of the election committee is regulated by of the Election Code. The election committee currently has 6 people and its chairman is Senator ... . He was unanimously elected to the position by the members of the election committee at an introductory held on July 19, 2019. Other members include ...(representative of the media department), ...(coordinator of the Liberec region and representative of the personnel department), ... , and .... The election coordinator is ... . The main ... of the election committee include approving the overall strategy and financing of the election campaign and coordinating the election staff. We therefore assess the statement as true.
Question extracted from material (adversarial, because it is related to a different material) How is human insulin produced using genetically modified bacteria?
Answer by baseline GPT-4.1-nano: The text does not provide information on how human insulin is produced using genetically modified bacteria. **Rating by: Gemma-3-27b-it** 100.00
Answer by llama-3-8b-Instruct finetuned: <|assistant|><*newline*>The production process involves introducing a gene that codes for the human insulin precursor into a bacterium, such as E. coli, and then inducing the bacteria with a substance such as IPTG to express the gene and produce the insulin precursor. The insulin precursor is then processed into mature insulin using enzymes present in the bacterium. Finally, the mature insulin is harvested from the bacteria and purified using standard biotechnology procedures. **Rating by: Gemma-3-27b-it** 95.00
Answer by Llama-2-7b plus DeepSeek-R1-Distill-Qwen-1.5b: The election committee approves the overall strategy and financing of the election campaign and coordinates the election staff. **Rating by: Gemma-3-27b-it** 0.00

Figure 2: A sample adversarial test item (i.e., the material, and a misleading, unrelated question) derived from the Demagog database with answers and ratings by competing systems. The baseline GPT-4.1-nano succeeded in handling the adversarial input and was also correctly rated as perfect by Gemma-3-27b-it. Other models produce unrelated outputs, and in the first case (llama-3-8b-Instruct finetuned), this also confuses the rating system, which assigns 95% to it.

4. Teacher system evaluation

There were 2 fully valid submissions to the **Teacher** task: `DeepSeek plus LLaMA` (`DeepSeek-R1--Distill-Qwen-14b`, and `LLaMA-3.2-3b`, by the team LLMinds [8]) and `Mistral-7b` (by the team BarFoo). In total, 3 submissions were received, but the last was received late and contained only the *<|assistant|>* token as the questions, so we disregarded it in the evaluation.

There were 2 baselines, the aforementioned expert-made questions and a system using `GPT-4.1--nano`. The `GPT-4.1-nano` baseline used a simple system, where we uniformly extracted spans of words from the section. We then prompted the LLM to generate one question for each span, answerable from that span; see Figure 3. In order to ensure that all the test set items received at least some questions, we also troubleshooted the baseline on the test set. This means it is artificially stronger when compared to the contestant systems. To ensure the output had the correct format, we used the OpenAI API's structured output functionality.

Evaluation of question generation is an underexplored and difficult task. Generally, one would need to acquire many reference questions [9] and compare them with the submitted questions with metrics such as ROUGE or BLEU, or use human evaluators. We decided to use two methods of evaluation: (1) a set of simple automatic methods, which should have a small variance, and (2) a method relying on an LLM to hopefully handle the semantics correctly. For the latter, we also explore a manually revised version of the LLM outputs on a small subset.

4.1. Simple automatic evaluation

The first approach to evaluation is designed to be as straightforward as possible, but nevertheless considers several aspects of the generated questions: their relevance, coverage of the source text, and mutual diversity of the questions. All these quantities are derived from a common pre-processing of the input material and the evaluated set of questions.

We start with a set of questions Q and a text. First, we segment the text into a set of overlapping windows W. Then we obtain embeddings of Q and W using `paraphrase-multilingual-mpnet-base-v2`, a small sBERT-based [10] model as provided by the `sentence-transformers 4.1.0` system.[2] Using these embeddings and a similarity function, we define a relevance metric $r : Q \times W \to \mathbb{R}$. We measure the cosine similarity between the embeddings of each of the questions $q \in Q$ and each of the windows $w \in W$. Then, for each window, we recalculate the relevance to focus only on the most relevant questions. Cosine similarity can be defined as

$$\forall_{x \in \mathbb{R}^n, y \in \mathbb{R}^n} s_{\text{cosine}}(x, y) = \frac{x^T y}{\sqrt{x^T x}\sqrt{y^T y}}. \tag{1}$$

Let us have c_x for each $x \in W$, a random variable that describes the similarities of embeddings of questions drawn from Q with the window x. Then we can express the relevance as

$$r(q, w) = \begin{cases} s_{\text{cosine}}(M(q), M(w)) + \text{quantile}_{0.5} c_x & s_{\text{cosine}}(M(q), M(w)) > \text{quantile}_{0.95} c_x \\ 0 & \text{otherwise} \end{cases}, \tag{2}$$

where $M(q)$ and $M(w)$ represent the embeddings of the question and window, respectively.

We further normalize this relevance to get a categorical distribution with probabilities p expressing the estimated likelihood that the question q is related to the window w of the text:

$$p : Q \times W \to \mathbb{R}, p(q, w) = \frac{r(q, w)}{\sum_{q', w' \in Q \times W} r(q', w')} \tag{3}$$

Using this metric and distribution we measure the following three quantities.

[2] https://sbert.net/

Relevance Relevance measures how much a given set of questions and text relate to each other. It is aggregated from the relevance of a question to a window over the elements of $Q \times W$.

$$\text{relevance}(Q, W) = \sum_{q,w \in Q \times W} r(q, w) \tag{4}$$

Coverage Coverage measures how uniformly the questions cover different parts of the text. Formally, we define it as the entropy of the distribution $m : W \to \mathtt{R}$, the marginal distribution on windows:

$$m(w) = \sum_{q \in Q} (p(q, w)) \tag{5}$$

Maximizing the $H(m)$ is equivalent to making the coverage of the text by the questions as uniform as possible.

Diversity Diversity estimates how uniquely different questions cover different parts of the text. We compute it from the conditional distribution $p_q : W \to \mathbb{R}$ for a fixed $q \in Q$:

$$p_q(w) = \frac{p(q, w)}{\sum_{w' \in W} p(q, w')} \tag{6}$$

Then the diversity is

$$\sum_{(q,q') \in Q \times Q} \text{KL-DIV}(p_q || p_{q'}) \tag{7}$$

High values of the Kullback-Leibler divergence indicate that the sequence of distributions $p_{q_1}, p_{q_2}, \ldots, p_{q_{|Q|}}$ is diverse.

Evaluation procedure We observed that the ranges of the relevance, coverage and diversity values vary from document to document by orders of magnitude. Therefore, to obtain a final result, it would be problematic to take the average values over all documents. Instead, we decided to rank the systems on each document and report the average rank (with 4 denoting the best rank). The results can be seen in Tables 1 to 3. In cases where there were no questions submitted by the team, we automatically rate them as the worst with ties broken by chance.

We can see the variance of ratings over the different material sections is quite high. This could mean that the variance of this method was still quite high or that the systems performed rather irregularly in their question quality. We can also see that all the methods were rated very similarly. This is very unlikely to be a good estimate, as some of the systems scored significantly worse when rated manually, as will be shown below in Section 4.2.

It is possible that if the evaluation method were better calibrated, e.g. by choosing a better way to convert the embedding pairs to a distribution, a better estimator could be created.

You are given a JSON dictionary labeled "Inputs" containing two fields:
- "Text": a passage of informational text - "Words": a list of sentences or phrases extracted from the text
Your task is to generate a list of question-answer pairs based on this information. Your output should be similar to a teacher's assistant preparing material for deep comprehension and critical thinking. Each item in the list must be a JSON object with the following fields: - "Question": a challenging, clear, and grammatically correct question - "Answers": a list of five distinct example answers
Guidelines: 1. Generate one question per item in the "Words" list.
2. Each question must be challenging, natural-sounding, and grammatically correct.
3. Ensure diversity in the questions, with each one exploring a different angle or idea related to the text.
4. The questions should prompt critical thinking and help deepen understanding of the topic.
5. Do not refer to the words themselves as "phrases" or "items"—form questions directly about the concepts they describe.
6. Each "Answers" list must contain five distinct, well-reasoned, responses derived from the content of the text. These should demonstrate a mix of reasoning, inference, and interpretation, not just direct extraction. If five completely unique responses cannot be produced, then at least ensure the responses are phrased differently.
7. Each answer should make sense on its own and be completely independent of the other answers.
8. Avoid pronouns such as "you", "he", "she", "they", and "we", and instead use neutral terms like "one", "the author", or proper nouns where applicable.
9. Ensure each question stands on its own, so the topic and intent are clear without needing to see the full text. 10. Avoid questions that ask about the wording or phrasing in the passage.
11. Avoid referring to the speaker, narrator, or specific experiments in the text.
12. The output should be only a JSON array of "Question": ..., "Answers": [...] objects.
13. Use only English.

Example Input:

```
{
  "Text": "You are attacked countless times every day, ...",
  "Words": [
    "Your immune system is a complex army of billions of cells: ...",
    "Hundreds of millions of people in the West have been vaccinated ..."
  ]
}
```

Example Output:

```
[
  {
    "Question": "What roles do different immune cells play in ...",
    "Answers": [
    "Some immune cells function as soldiers, attacking and destroying harmful ...",
      "Some cells produce weapons against the infection ...",
      "Some components of the immune system serve as production centers ...",
      "Some components of the immune system serve as production ...",
      "Each type of immune cell contributes uniquely to ..."
    ]
  },
  ...
]
```

Input:
...
(The input formatted as above)
...

Figure 3: The simple prompt used for the **Teacher** baseline using `GPT-4.1-nano`. The prompt is relatively long because we wanted to make sure it did not fail on any part of the dataset. Further prompt optimization was surely possible.

4.2. LLM evaluation

For comparison with the simple automatic measure, we evaluate the questions by ranking unique triplets of submissions for a given document and rating them by `GPT-4.1-`**`mini`**`-2025-04-14` in four categories.[3]

We designed a complex prompt that asks the LLM to decide the worst and the best output from three submissions in multiple categories. The third evaluated system (i.e. the one not mentioned as the best or worst by the LLM) is implicitly ranked as the second one of the three. The highest (best) possible value of the rank is thus 3 in this evaluation.

In order to score all 4 systems (two submissions and two baselines), we repeatedly sample triplets from this set of 4 and average the obtained ranks.

The exact prompt is provided in Figure 4. The model should consider the situation of a student learning for an exam. Provided with the learning material and the three possible sets of questions, the model should go over 8 statements and for each of them indicate which of the three sets of questions matches the statement best. The statements are always paired; the first in a pair asks about the most covering/complex/etc. set of questions, the second asks about the least covering/complex/etc. set of questions. We opted for this complex and structured request because we wanted to reuse the relatively long input material across the evaluated question sets. The comparative rating seemed to us to be easier for the model to process. Our limited experiments also indicated that this prompt led to more stable evaluations than other methods when comparing models.

The results are provided in Table 4 and we can see that the outputs of our baseline system are scored the highest (remember the artificial advantage the baseline had, as discussed at the beginning of Section 4), followed by `DeepSeek plus LLaMA` submission by the LLMinds team. This is true broadly for all categories.

The main reason why the system using `Mistral-7b` submitted by BarFoo is scored so low is that many of its quizzes contain questions related to a completely different part of the text. We suspect that the BarFoo team extended the context by concatenating the inputs but failed to filter the generated questions properly.

In contrast to automatic evaluation in Section 4.1, here we only score submissions on the texts where the participating systems provided some questions.

Manually revised LLM evaluation In the final type of evaluation, we manually review the scores provided on a selected small subset of outputs. Our plan was to fix any clearly wrong assessments when skimming the provided sets of questions and referring back to the source material as needed. However, the sets of questions were very diverse and therefore difficult to compare. In the end, we left most of the automatic scores intact. We provide the manually revised ranks in Table 5.

When reviewing the outputs, we observed some of the systems produced questions asking about the author or the title of the document, which is likely caused by a significant misunderstanding of the task.

4.3. Teacher evaluation overview

We rate `DeepSeek plus LLaMA` as the best of the 2 submissions. When looking at the manually revised ratings in Table 5 `DeepSeek plus LLaMA` has a range of rankings, 2.1 to 2.5, that is similar to the range of rankings of the expert-made (manual) baseline and the `GPT-4.1-nano` baseline. The other submission, `Mistral-7b`, has rankings in a significantly worse range of 1.4 to 1.5.

Overall, while the exact ratings of systems under these evaluation methods differ, we can see that a pattern emerges across all of the methods. The baseline, manually created by experts (the creators of the material), when available, typically scores among the better systems. It usually takes the second rank. In our experience, LLMs used as judges tended to rate other LLMs as better than human outputs, so we are not surprised to see this result. The fact that the expert-made questions from the creators

[3]Because the ranking model `GPT-4.1-`**`mini`** was similar to the baseline model `GPT-4.1-nano`, we decided to also double-check the rankings using `Gemini-2.5-flash`; we got almost exactly the same results; see Table 25 in the Appendix.

Table 1
Simple automatic evaluation of **Teacher** submissions: Mean rank with 4 being best overall.

questions by	average rank
DeepSeek plus LLaMA	2.50±1.11
Mistral-7b with BLIP	2.48±1.13
baseline from creators	2.48±1.14
baseline GPT-4.1-nano	2.47±1.13

Table 2
Simple automatic evaluation of **Teacher** submissions. Mean rank with 4 being best, grouped by evaluation measure: coverage, diversity and relevance.

questions by	type	average rank
DeepSeek plus LLaMA	coverage	2.51±1.11
	diversity	2.47±1.10
	relevance	2.51±1.12
Mistral-7b with BLIP	coverage	2.44±1.13
	diversity	2.51±1.13
	relevance	2.50±1.13
baseline from creators	coverage	2.53±1.14
	diversity	2.45±1.14
	relevance	2.46±1.15
baseline GPT-4.1-nano	coverage	2.45±1.13
	diversity	2.54±1.14
	relevance	2.43±1.12

Table 3
Simple automatic evaluation of **Teacher** submissions: Mean rank with 4 being best, grouped by material kind. Best result for each material kind in bold.

questions by / material kind	DeepSeek plus LLaMA	Mistral-7b with BLIP	baseline from creators	baseline GPT-4.1-nano
demagog-statements-public	**2.51±0.98**	2.41±1.06	no questions	2.02±0.92
flat-earth-book	2.64±1.19	2.55±0.99	no questions	**2.86±1.00**
nmt-book	**2.85±0.99**	2.70±1.07	no questions	2.41±0.93
nmt-class	**2.49±1.12**	2.48±1.13	no questions	2.47±1.14
popular	**2.82±1.09**	2.53±1.18	2.58±1.16	2.22±1.13
ukr-biology	2.40±1.12	**2.69±1.12**	2.58±1.12	2.57±1.09
world-history	2.46±1.11	2.31±1.33	**2.87±1.08**	2.76±1.06

score better in the LLM-based evaluation than **Teacher**s' outputs can also be seen as a small sanity check of our evaluation methods, even if it did not last after our small manual revision.

The `DeepSeek plus LLaMA` system seems to be only a little behind both baselines or even exceeds them at times, and does not fail on any material kind.

Based on our browsing of the data, but without providing any further details, we would like to mention that the systems that attempted to also provide reference answers generally provided very low-quality ones. We see this mismatch as surprising given the overall acceptability of the generated questions. Being able to derive questions from a given text and still be unable to answer them (based on the very same text) suggests that the models are probably using some shallow cues on question

You are required to fulfill an api request to a large language model, please respond with the requested filled in freeform text.
Simulate a randomly sampled university student required to prepare for an oral exam by studying the given material. The exam will require memorizing the material as well as learning to think about the topics in the material. The questions are not required to cover all the information in the material but must give at least some coverage of the main topics and pieces of information. We need you to fill the fields in our questionnaire about quiz questions so we can fill in our statistics.
You are a randomly sampled university student required to prepare for an oral exam by studying the given text, we need you to answer our questions about a quiz so we can fill in our statistics.
You are not sure what the exam will contain other than that it will be answerable using information from the text. You are given quiz questions purely to help you understand the text better. Quiz questions are very different than those asked by the teacher during the oral exam.
You should be able answer these questions from the text provided.
Our scientific questionnaire is a simple format string. Each QUESTIONSET_NUMBER: needs to be followed by the corresponding QUESTIONSET number.
...
(The text and numbered question sets in json format)
...
Please fill in the value QUESTIONSET_NUMBER: {} with the correct QUESTIONSET numbers
ENTRY1: The questions require thinking about least different parts of the material. is most true about "QUESTIONSET_NUMBER": {}
ENTRY2: The questions require thinking about most different parts of the material. is most true about "QUESTIONSET_NUMBER": {}
ENTRY3: The questions cover the material least. is most true about "QUESTIONSET_NUMBER": {}
ENTRY4: The questions cover the material most. is most true about "QUESTIONSET_NUMBER": {}
ENTRY5: The questions are least useful for learning to reason about the material for the test. is most true about "QUESTIONSET_NUMBER": {}
ENTRY6: The questions are most useful for learning to reason about the material for the test. is most true about "QUESTIONSET_NUMBER": {}
ENTRY7: The questions are least useful for learning the material for the test. is most true about "QUESTIONSET_NUMBER": {}
ENTRY8: The questions are most useful for learning the material for the test. is most true about "QUESTIONSET_NUMBER": {}

Figure 4: The prompt used in the **Teacher** ranking. Some parts are repeated to stress their importance. This prompt was constructed in a manual iterative updates so it is possible that there are prompts easier to read that achieve a similar or even better results.

formation rather than processing the underlying meaning of the material. In the next section, we move to testing the ability to answer questions.

Table 4
LLM-based evaluation of **Teacher** submissions: Ranking in multiple categories as determined by `GPT-4.1-mini-2025-04-14` with the prompt provided in Figure 4. We report the mean rank with 3 being best.

type of evaluation questions by	cover the material well	require thinking about different parts of the material	useful for learning the material	useful for learning to reason about the material
DeepSeek plus LLaMA	1.80±0.76	1.72±0.67	1.86±0.86	1.77±0.76
Mistral-7b with BLIP	1.52±0.67	1.36±0.58	1.43±0.70	1.83±0.72
baseline from creators	2.19±0.75	2.19±0.62	2.08±0.73	2.14±0.49
baseline GPT-4.1-nano	**2.62±0.60**	**2.83±0.43**	**2.54±0.61**	**2.48±0.78**

Table 5
LLM-based evaluation of **Teacher** submissions with manually revised ranks. We only revised the ordering for one document from each material kind. We changed about half of the individual rankings, most of them only by 1 point. We report the mean rank with 3 being the best.

type of evaluation questions by	cover the material well	require thinking about different parts of the material	useful for learning the material	useful for learning to reason about the material
DeepSeek plus LLaMA	2.27±0.70	2.25±0.87	**2.40±0.63**	2.27±0.70
Mistral-7b with BLIP	1.41±0.62	1.40±0.63	1.50±0.71	1.47±0.64
baseline from creators	2.00±0.87	1.62±0.74	2.00±0.87	1.78±0.83
baseline GPT-4.1-nano	**2.58±0.67**	**2.47±0.61**	2.36±0.81	**2.50±0.76**

5. Student system evaluation

For **Student**, only two submissions were formatted in a recoverable way: `Llama-3-8b` (by TeamUTK) and `LLaMA plus DeepSeek` (`Llama-2-7b` plus `DeepSeek-R1-Distill-Qwen-1.5b`, by LLMinds [8]). We again provided a simple baseline using `GPT-4.1-nano`, this time simply providing the entire section and a quiz, as can be seen in Figure 5. To ensure that the output had the correct format, we again used the OpenAI API's structured output functionality.

5.1. Automatic evaluation

We chose ROUGE-L Recall as the evaluation metric as advised by [3]. We preferred the Recall variant to ROUGE-L F1 because most of the answers are reasonably long, and we did not want to penalize longer answers. The results of these two metrics were very similar anyway. As can be seen in Tables 6 and 8, the results of this automatic evaluation of the systems on expert-made questions are much higher—up to almost twice as high for `LLaMA plus DeepSeek` (`Llama-2-7b` plus `DeepSeek-R1-Distill-Qwen-1.5b`, by LLMinds [8]), with 25 vs. 40. This indicates that the teacher-submitted questions were either harder to answer from the text provided, they were bad questions, or their reference answers were bad answers. Some submissions have relatively poor ratings, likely due to some responses having formatting issues or being based on questions/documents not matching the response indexation. Such issues should have been relatively easy to predict and debug on the devset.

The system `Llama-3-8b` (by TeamUTK) provided answers to only a smaller subset of the questions. The results of the systems on this subset can be seen in Table 7. They show that the TeamUTK model actually fares much better (up to 10 times) when we consider only this subset, but still underperforms the other models significantly.

You are a knowledgeable assistant. Your task is to answer a set of questions using ONLY the information explicitly provided in the following text. Make sure to follow the following rules to the letter.
1. Do NOT use any outside knowledge or make assumptions.
2. Do not explicitly reference the text.
3. Avoid pronouns such as "you", "he", "she", "they", and "we", and instead use neutral terms like "one", "the author", or proper nouns where applicable.
4. Use only English.
5. Avoid referring to the speaker, narrator, or specific experiments in the text.
6. For each question, return an object with the following fields:

- "answer": If the question is answerable, provide the exact direct answer in the form requested by the question, do not describe your answer unless the question asks you to. If the question is not exactly answerable from the text try to answer anyway to the best of your ability.
- "answerable": A boolean indicating whether the question can be directly answered using the text.
7. Return exactly as many answers as there are items in the question list.
8. Only say a question is not answerable if the text does not provide ANY information relevant to the question, otherwise answer the question.

Format your responses as a list of such objects, one per question.
Example response format:

```
[
  {{
    "answer": "The immune soldiers usually take care of attacks ...",
    "answerable": true
  }},
  {{
    "answer": "The monastery includes a baroque library, farm ...",
    "answerable": true
  }},
  {{
    "answer": "The text mentions that it is difficult to ...",
    "answerable": false
  }}
]
```

Reference text: ...
(The material section) ...
Questions: ...
(The list of questions) ...

Figure 5: The simple prompt used for the **Student** baseline using `GPT-4.1-nano`.

5.2. Correlations of ROUGE-L Recall scores between teams

To verify our choice of the ROUGE-L Recall evaluation method, we tested whether expert-made questions could be reliably separated into easy and difficult questions from scores across the **Student** systems. To this end, we calculate the Pearson correlation coefficients of ROUGE-L recall across all expert-made questions, for all pairs of system submissions. We see that `Llama-3-8b` and `DeepSeek plus LLaMA` are rated significantly more similarly than any other pair, that is, 0.24 vs. 0.1 or -0.21. This makes sense as they use similar LLMs. This shows that for similar LLMs, the expert-made questions are similarly easy or hard.

On the other hand, it is strange that the ROUGE-L Recall scores of our GPT baseline and `LLaMA plus DeepSeek` (`Llama-2-7b` plus `DeepSeek-R1-Distill-Qwen-1.5b`, by LLMinds [8]) are highly negatively correlated, that is, -0.31. One possible explanation is that each uses an approach to question answering that works for questions with different binary properties; for example, factual vs. abstractive.

Table 6
The average ROUGE-L Recall scores of **Student** submissions on all questions.

answered by	rating
Llama plus DeepSeek	25±33
baseline GPT-4.1-nano	**49±34**
Llama-3-8b	02±11

Table 7
The average ROUGE-L Recall scores of **Student** submissions on questions where all teams submitted.

answered by	rating
Llama plus DeepSeek	32±36
baseline GPT-4.1-nano	**57±38**
Llama-3-8b	19±25

Table 8
The average ROUGE-L Recall scores of **Student** submissions on expert-made questions.

answered by	rating
Llama plus DeepSeek	40±41
baseline GPT-4.1-nano	**57±41**
Llama-3-8b	04±15

Table 9
The average ROUGE-L Recall scores of **Student** submissions on expert-made questions grouped by kind.

material kind answered by	demagog-statements-public-determine-explanation	demagog-statements-public-determine-statement	demagog-statements-public-statement-w-explanation	popular
Llama plus DeepSeek	44±42	**71±28**	**69±39**	**25±12**
baseline GPT-4.1-nano	**51±39**	46±36	26±34	08±04
Llama-3-8b	11±23	00±00	00±00	no anwers

The different observed difficulty of expert-made questions would deserve further exploration.

5.3. Manual evaluation

The answers were also manually evaluated, randomly choosing one set of questions for each data source. We excluded world history because for its sections, the set of all questions was too large to be evaluated manually given our limited evaluation budget.

To maintain consistency in the evaluation procedure, we prepared simple instructions in the form of a questionnaire; see Figure 6. The questionnaire was manually filled in by the main task evaluator. The time was limited, so the examination per answer was very brief.

As we can see in Table 11, the results of the manual evaluation seem to generally agree with the automatic evaluation. This might be mostly due to the fact that many answers have technical difficulties, which makes them quite easy to distinguish as incorrect. A more detailed breakdown can be seen in Table 12.

Table 10
For each pair of submissions we report the Pearson correlation coefficient of ROUGE-L Recall scores on expert-made questions.

	Llama-2-7b plus DeepSeek-R1-Distill-Qwen-1.5b	baseline GPT-4.1-nano	Llama-3-8b
Llama plus DeepSeek	1.00	-0.31	0.24
baseline GPT-4.1-nano	-0.31	1.00	0.1
Llama-3-8b	0.24	0.1	1.00

Table 11
The manual Likert ratings 1-5, 5 is best.

rating kind → answered by ↓	answer is correct	answer is based solely on the material
Llama plus DeepSeek	2.78±1.31	2.97±1.45
baseline GPT-4.1-nano	**3.35±1.25**	**3.58±1.44**
Llama-3-8b	2.00±1.41	2.20±1.70

Table 12
The manual Likert ratings 1-5, 5 is best, by data category.

	rating kind	is correct	was created from the material
answered by	material kind		
Llama plus DeepSeek	demagog-statements-public	2.49±1.29	2.49±1.29
	flat-earth-book	1.60±0.85	1.80±1.13
	nmt-book	2.05±1.48	1.98±1.38
	nmt-class	3.80±1.13	**4.60±0.57**
	popular	**4.47±0.19**	4.67±0.47
	ukr-biology	2.29±0.75	2.29±0.75
baseline GPT-4.1-nano	demagog-statements-public	**3.24±0.06**	**3.24±0.06**
	flat-earth-book	**2.80±2.55**	**3.00±2.83**
	nmt-book	**3.22±1.46**	**3.64±1.93**
	nmt-class	**3.90±0.99**	4.20±1.13
	popular	4.38±0.31	**4.83±0.24**
	ukr-biology	**2.58±1.72**	**2.58±1.72**
llama-3-8b	flat-earth-book	2.00±1.41	2.20±1.70

5.4. Using question answering systems for fact-checking

For determining the system's capacity to answer questions about fact-checking, we derived a collection for question-answer pairs automatically from the Demagog database, obtained in May 2025.

The database contains manually selected and processed statements by Czech politians and can be browsed online.[4] Each item in the database is equipped with: (1) the original statement, i.e. a very short excerpt, typically in the form of an abridged sentence, (2) a long fact-check explanation, i.e. an elaborate analysis with many references to public sources, carefully discussing the verity of the statement, (3) a short, one-paragraph summary derived from the long explanation, (4) the final verdict on the verity: true, untrue, misleading, impossible to decide. Importantly, the verdict is typically explicitly stated at the end of the long summary in plain Czech.

[4]http://demagog.cz/

For each question-answer pair, address each of these points on a single line numbered correspondingly, giving 1-5 on opinion-based points. Whenever we are talking about using the provided texts, this also means using elementary school-level background knowledge. If you would answer any point as 1, continue to the next question-answer pair:
1. is correct.
2. was created using only the provided texts.

Figure 6: The manual evaluation questionnaire.

As input to the **Student** task, we used three kinds of question-material combinations:

1. The materials are the combination of a statement, a short fact-check explanation, and the corresponding long fact-check explanation. The questions are those generated by **Teacher** systems when given the material as input. (demagog-statements-public-statement-w-explanation)
2. The materials are a single long fact-check explanation. The questions are constructed so that each one contains a set of up to five possible statements uttered by the same politician and labelled A–E in the question. At the end, the questions ask the student to determine the statement related to the explanation provided in the material, and to report its truthfulness. As said, the ground-truth truthfulness is typically present at the end of the explanation. The possible answers were TRUE followed by SECTION (A-E), FALSE followed by SECTION (A-E), and if none of the statements matched the explanation, we demanded the answer UNKNOWABLE. (demagog-statements-public-determine-statement)
3. The materials are very short statements. The questions are created by concatenating a set of up to five possible long fact-check explanations for statements from the same person, and then ask the student to determine the relevant explanation and determine the statement's truthfulness. As above, the explanation typically contains the verdict spelled out in some plain language form. The possible answers were TRUE followed by SECTION (A-E), FALSE followed by SECTION (A-E), and if none of the statements matched the explanation, we demanded the answer UNKNOWABLE. (demagog-statements-public-determine-explanation)

Converting a genuine fact-checking task to a multiple-choice question might not have been ideal, and in further iterations of this task we are going to experiment further with formulating the questions in a way that is understandable to both humans and even small LLMs.

In Table 9, we can see that `LLaMA plus DeepSeek` (`Llama-2-7b` plus `DeepSeek-R1-Distill--Qwen-1.5b`, by LLMinds [8]) performed significantly better than their overall average on this task (44 vs. 71, and 69 vs. 40 in Table 8). This can be attributed in part to the fact that the verdict is present in the explanation and that the correspondence between the statement and the explanation can often be gathered from the names they both mentioned. Both of these effects were illustrated in Figure 1.

We attempted to make the questions more challenging by creating the wrong options using data from the same speaker, but this was not enough, as the statement and fact-check often contain other specific names of ministries or politicians, and the content-bearing words also easily identify the topic. In light of this, it is somewhat interesting that the systems did not achieve a ROUGE-L Recall closer to 100.

In order to provide some quantitative details on the submitted systems' ability to extract readily available information from a little obfuscated text, we look at two different divisions of the "demagog-statements-public-determine-statement" subset:

1. We calculate vocabulary similarity using the Jaccard index over word forms (number of word forms occurring in both texts divided by the number of word forms occurring in either text) between the correct statement and the explanation. We then separate items where the correct statement has a similarity at least 1.5 times higher than any of the confounding statements in the item. We expect that word-level overlap is going to help LLMs in identifying the relevant statement, and indeed, we get a very large difference in the accuracy of predicting the correct statement: 0.44 without the overlap vs 0.83 with the high vocabulary overlap. While this could be

caused by semantic differences in the groups, another possible explanation is that the models are using the similarity in terms and names and other shallow expressions instead of understanding the texts.

2. We look at all items in the "demagog-statements-public-determine-explanation" subset where the relevant explanation explicitly states the truthfulness with phrases "as true" or "as untrue" and compare them to those where these phrases are not stated. In our data, all statements containing these phrases are rated as true or as untrue according to them. We evaluated the accuracy of predicting truthfulness in each of these two groups separately. While the correct truthfulness could have been read directly from these phrases, so the former group of items should have obtained much higher accuracy than the latter one, we measured rather similar accuracies, 0.58 for the former and 0.41 for the latter. This suggests that the systems were unable to understand that these particular words in the conclusion of the explanation are a solid indicator.

5.5. Student evaluation overview

It seems that even for the questions where `Llama-3-8b` (by TeamUTK) submitted the answers, `LLaMA plus DeepSeek` (`Llama-2-7b` plus `DeepSeek-R1-Distill-Qwen-1.5b`, by LLMinds [8]) provides better results. Both sets of submissions appear to score worse than our `GPT-4.1-nano` baseline.

However, the reliability of LLMs on **Student**-like systems is very far from acceptable. Our small analysis on the fact-checking obfuscated multiple choice questions indicates that LLMs use shallow word overlap to identify relevant sections in the input, and are not able to understand which words from the input are directly providing the expected answer, even if they are present.

6. Evaluator system evaluation

There were 2 valid **Evaluator** submissions, `Gemma-3-27b-it` (by Outstanding Outsiders [11]) and `Mistral-7b`. We decided to limit our evaluation to responses in which both teams submitted an evaluation. This did not change the results as the system with more submitted ratings still leads by a large margin.

We again provided a baseline using `GPT-4.1-nano` to compare with the submissions of the contestants. In this baseline system, instead of using the full scale of 0–100 in the LLM prompt, we use a scale of 0–5 so that we can describe the meaning of each rating in words, as can be seen in Figure 7.

The **Evaluator** systems sometimes get confused and use a scale from 0-1. This is obvious from the fact that they never do such fine-grained ratings at any other part of the 0-100 scale and the fact that they sometimes provide real-valued ratings instead of the integer-valued ratings requested. We took the liberty of multiplying all ratings less than or equal to 1 by 100 to recover as much usable data as possible.

6.1. Adversarial answers

The most interesting direct way to check validity is simply to provide the system with a tailored adversarial input. One possible way to create such an input is to take an answer that is related to the material section as closely as possible, but not to the question itself. As we can see in Table 13, most **Evaluator** systems, as expected, rate these inputs as worse responses, but the decrease is not nearly as drastic as we would have liked. Since these answers are almost always completely unrelated to the questions, the average rating should fall close to 0.

Evaluator systems show similar ignorance when we shuffle words in the answer or swap the material section from which the question and answer originated. As can be seen in Table 14, we also see a relatively large decrease, but not values close to 0.

Please respond with the requested plain text.
You are given a list of question answer pairs. You are required to rate the quality of the answer on a scale from 0 to 5.
The scale is as follows: 0 means the answer is completely wrong, 1 means the answer is partially useful, 2 means the answer is useful, 3 means the answer is very useful, 4 means the answer is almost correct, and 5 means the answer is completely correct.
Fill in a format strings with the ratings integers as per this EXAMPLE:
"RATINGS:
rating of ANSWER 1: {}"
OUTPUT:
"rating of ANSWER 1: 0"
REFERENCE TEXT:
...
(The material section)
...

QUESTION-ANSWER:
...
(The question answer pairs stored as a JSON dictionary.)
...

Fill in the following format string with the ratings integers
"RATINGS:
rating of ANSWER 1: {}
..."
OUTPUT:

Figure 7: The simple prompt used for the **Evaluator** baseline using `GPT-4.1-nano`.

6.2. Adversarial questions using Student submissions

Another simple way to check the validity of an **Evaluator** system is to test it on adversarial questions. These were created by shuffling words, replacing them with random text, or swapping questions between different kinds of material. As we can see in Table 15, most systems rate the answers to these questions as worse, for example, the baseline reduced its ratings from 65 to 49. However, the decrease is not nearly as drastic as we might have hoped, given that such questions are almost always completely nonsensical. Some of this could be caused by some **Student** systems detecting that they are given unanswerable questions and adapting their response, as can be seen in Figure 1, with **Evaluator** systems scoring such responses high. However, that is unlikely to entirely account for a decrease with such a small magnitude.

6.3. Scoring human-made answers to human-made questions

Finally, we can also assess how **Evaluator** systems score golden answers to expert-made questions.

In Table 16, we see that on average the golden answer to a golden question receives a higher score than the **Student** answer to the same question, the expert-made questions get up to 94 percent average rating for `Gemma-3-27b-it`.

Table 18 below shows that one of the systems, `Gemma-3-27b-it`, actually gives a mean score of 100 to the golden answers for documents in the category demagog-determine-explanation, which means that it rated them all with the most fitting rating.

As we have pointed out in Table 16 earlier, the other system, Mistral, at least provides golden answers with higher scores than the student answers, even if the average rating increase from 62 for non-gold to 72 for gold gives only a 10-point difference — a significantly lower difference than the variance of its scores given to golden answers to different questions.

Interestingly enough, our **Evaluator** baseline using `GPT-4.1-nano` produced counter-intuitive

Table 13
Adversarial test of **Evaluator**s by swapping the question for a different one from the same document while keeping the answer intact. The left column contains the ratings for unaltered items, the right columns contains the ratings with questions that were replaced by random ones or ones from a different kind of material or with words shuffled. The scores for questions in the right column should have been substantially lower. The underlying set of materials is fixed.

rating by	rating non adversarial	rating adversarial
Gemma-3-27b-it	82.90±27.49	59.44±43.00
Mistral-7b	62.47±47.86	57.65±49.37
baseline GPT-4.1-nano	65.54±37.02	26.38±35.84

Table 14
Adversarial tests for **Evaluator**s by shuffling words in the answer or swapping in a material of a different kind. The left column contains the ratings for unaltered items, the right columns contains the ratings with answers where the scores should have been substantially lower. The underlying set of questions corresponding to the answers is fixed.

rating by	rating non adversarial	rating adversarial
Gemma-3-27b-it	82.90±27.49	32.86±40.27
Mistral-7b	62.47±47.86	55.37±49.60
baseline GPT-4.1-nano	65.54±37.02	13.33±27.05

Table 15
Adversarial tests for **Evaluator**s by replacing the answer with an answer to a different question from the same section if possible; if it is not possible we use an answer to a question from the same material kind.

rating by	rating non adversarial	rating adversarial
Gemma-3-27b-it	82.90±27.49	67.40±40.42
Mistral-7b	62.47±47.86	58.80±48.72
baseline GPT-4.1-nano	65.54±37.02	49.38±42.67

Table 16
The comparison of how **Evaluator**s rate the expert-made answers. The left column contains ratings for answers to expert-made questions generated by **Student** submissions, the right column contains ratings of expert-made answers to expert-made questions. Adversarial entries are not included.

rating by	rating non-gold	rating gold
Gemma-3-27b-it	81.50±28.26	94.69±15.68
Mistral-7b	62.15±47.92	72.73±46.71
baseline GPT-4.1-nano	63.71±36.82	80.94±35.35

outputs for one of the Demagog-derived question sets. It graded the golden answers in "demagog-statements-public-determine-explanation" significantly lower than the **Student** answers; 45.00 vs 77.78, which is almost a 23-point difference, as seen in the detailed Table 18.

6.4. Agreement between different Evaluator systems

One way of checking the overall stability of the automatic evaluation provided by **Evaluator** systems is to test how often they agree in their output scores for a given text-question-answer tuple. In general,

Table 17
Agreement in evaluations by different **Evaluator** systems. In this table, we report the mean of the indicator function $I_{f_1 f_2}(x) = 1 \iff c(f_1(x)) = c(f_2(x))$ for each pair of systems f_1, f_2 and the way in which the question and answers were created/modified. The class $c(y)$ of a rating y is determined via quantizing y into bins defined by ranges, class 1 corresponds to scores of [0-33), class 2 to scores of [33-66), and class 3 to scores of [66-100]. The upper part of the table uses genuine question-answer pairs, the lower part uses different adversaries.

adversarial category	Gemma-3-27b-it $\sim$ Mistral-7b	Gemma-3-27b-it $\sim$ baseline GPT-4.1-nano	Mistral-7b $\sim$ baseline GPT-4.1-nano
golden answer	0.76	0.90	0.76
answer by **Student** system	0.68	0.82	0.68
answers swapped	0.43	0.65	0.47
answer random text	0.55	0.57	0.57
question random text	0.56	0.32	0.58
words in answer shuffled	0.57	0.56	0.47
words in question shuffled	0.69	0.85	0.56

for all pairs of **Evaluator** systems, when both systems gave a valid rating, these ratings often agreed with each other. They agreed in up to 90% of the cases, as illustrated in Table 17. As expected, they most often agreed when rating the correct golden answers.

It is interesting to note that when dealing with adversarial input, the **Evaluator** systems disagreed more often than when rating non adversarial inputs. They got agreement percentages of around 60 on this adversarial input as opposed to percentages of around 70 on golden and **Student** inputs. We see this observation as a possible basis for improving the reliability of the systems: a decrease in agreement by two or more independent systems could be seen as an indicator that the item is problematic and its scoring should be decreased, or is completely inappropriate. However, this is still not an ideal technique for the evaluated **Evaluator** systems as there is at least one clear counterexample: The agreement of **Evaluator** systems when they are evaluating answers generated by **Student**s to questions with shuffled words is almost as high and sometimes even higher than when rating **Student** system answers to non-adversarial questions; see the last lines in the lower vs. upper part of Table 17.

6.5. Comparison with metrics using reference answers

A more indirect way is to compare the results of the **Evaluator** system with the metrics we used to rate **Student** systems in Section 5.

Instead of directly correlating ROUGE scores with **Evaluator** predictions, we are curious to assess **Evaluator**s' ability to predict the rough correctness in absolute terms. We thus divide ROUGE-L and **Evaluator** scores into three classes, equidistantly covering the full range of 0–100. Then we determine how accurately the **Evaluator** scores fall into the same class as the ROUGE-L score and average this across all question-answer pairs and all **Student** submissions.

In Table 19, we see that the overall accuracy for non-adversarial questions is around 0.33 for all systems. This means that it is about as good as randomly guessing because there are only 3 classes. This could be primarily attributed to the low precision, below 0.1, in predicting when ROUGE-L Recall scores an answer as half correct (denoted “class 2” in the tables). This makes sense as it is for these sorts of answer where any metrics on question answering quality disagree most.

As can be seen in Table 20, the models lack the ability to correctly score the answers with the lowest ROUGE-L Recall (denoted “class 1” in the tables) for expert-made questions. For most systems the accuracy of agreement with ROUGE-L Recall is higher in the case when dealing with expert-made questions, that is 0.19 increase for `Gemma-3-27b-it` and 0.13 increase for `GPT-4.1-nano` baseline. In Table 22, we can see that when dealing with **Student** answers to adversarial questions the systems seem to agree with ROUGE-L slightly more often than overall, even though it should be easy to match

Table 18
The mean ratings of **Evaluator** systems on questions automatically generated from the Demagog.cz database as described in Section 5.4.

material kind	rating by → answers made by ↓	Gemma-3-27b-it	Mistral-7b	baseline GPT-4.1-nano
demagog-statements-public-determine-explanation	experts	100.00±0.00	No ratings	45.00±48.70
	Student systems	66.56±36.30	89.63±31.06	77.78±36.87
demagog-statements-public-determine-statement	experts	93.72±17.89	57.14±53.45	84.26±32.55
	Student systems	77.35±32.37	75.80±42.69	64.44±43.61
demagog-statements-public-statement-w-explanation	experts	87.02±21.44	63.90±47.88	65.32±30.08
	Student systems	80.19±29.29	64.01±47.40	63.11±36.03

Table 19
The accuracy of predicting class of rating of ROUGE-L Recall scores on all non-adversarial questions. The class of a rating determined via quantizing ratings into bins defined by ranges, class 1 corresponds to scores of [0-33), class 2 to scores of [33-66), and class 3 to scores of [66-100]. The first column reports the accuracy over all the three classes, the subsequent columns report accuracies for individual classes from the lowest ROUGE class ("accuracy class 1") to the highest class ("accuracy class 3").

rating by	overall accuracy	accuracy class 1	accuracy class 2	accuracy class 3
Gemma-3-27b-it	0.27±0.21	0.28±0.38	0.06±0.16	0.19±0.22
Mistral-7b	0.36±0.22	0.52±0.38	0.04±0.12	0.18±0.24
baseline GPT-4.1-nano	0.33±0.21	0.47±0.41	0.07±0.14	0.17±0.23

Table 20
The accuracy for predicting class of rating of ROUGE-L Recall on expert-made questions. For details see the caption of Table 19.

rating by	overall accuracy	accuracy class 1	accuracy class 2	accuracy class 3
Gemma-3-27b-it	0.46±0.27	0.01±0.06	0.04±0.17	0.40±0.30
Mistral-7b	0.32±0.29	0.10±0.24	0.01±0.07	0.21±0.27
baseline GPT-4.1-nano	0.46±0.28	0.09±0.18	0.04±0.17	0.32±0.34

the usually low scores given to them. It should also be noted that some **Student** answers to adversarial questions do achieve a high rating purely because of the inadequacies of ROUGE-L Recall.

6.6. Using Evaluator systems for checking fact-check explanation validity

As we can see in Table 21, when predicting the rating of the **Student** answers which aim to identify the related statement and "read off" its verity (see Section 5.4 for the description of this data subset), `Gemma-3-27b-it` performs much better than Mistral, esp. in the high ROUGE-L range and performs on par with the baseline using `GPT-4.1-nano`.

Table 21
The accuracy for predicting class of rating of ROUGE-L Recall on questions created by the procedure described in Section 5.4 from the Demagog database. For details see the caption of Table 19.

rating by	overall accuracy	accuracy class 1	accuracy class 2	accuracy class 3
Gemma-3-27b-it	0.50±0.26	0.01±0.07	0.06±0.19	0.43±0.31
Mistral-7b	0.26±0.26	0.06±0.13	0.01±0.09	0.19±0.27
baseline GPT-4.1-nano	0.50±0.28	0.10±0.18	0.06±0.19	0.34±0.36

Table 22
The accuracy for predicting class of rating of ROUGE-L Recall. The questions include ones made adversarial by corrupting text or swapping material, the reference answers are kept unchanged. For details see the caption of Table 19.

rating by	overall accuracy	accuracy class 1	accuracy class 2	accuracy class 3
Gemma-3-27b-it	0.39±0.24	0.42±0.37	0.04±0.14	0.18±0.22
Mistral-7b	0.39±0.21	0.51±0.36	0.03±0.13	0.22±0.24
baseline GPT-4.1-nano	0.42±0.25	0.51±0.35	0.05±0.13	0.11±0.20

Table 23
We report two values for each participating team. The left column reports the mean rating by all participating **Evaluator**s of the given team's **Student** system answers (excluding adversarial items). The right column reports the mean rating by all participating **Evaluator**s of reference answers by the team's **Teacher** system to the questions this **Teacher** system generated. Bear in mind that all teams used different prompts or even models or approaches for their **Teacher** and **Student** submissions.

answers by	in the **Student** mode	in the **Teacher** mode
BarFoo	not applicable	70.97±36.74
reference answers extracted from material	not applicable	86.62±30.03
baseline GPT	73.04±37.18	64.43±31.96
TeamUTK	68.23±44.42	not applicable
LLMinds	68.04±39.56	70.09±37.87

6.7. Broader comparison of Evaluator outputs with our evaluation results

The left column of Table 23 presents the mean **Evaluator** scores assigned to the answers of the respective **Student** submissions. These scores suggest that the `LLaMA plus DeepSeek` (`Llama-2-7b` plus `DeepSeek-R1-Distill-Qwen-1.5b`, by LLMinds [8]) **Student** system delivers the best answers, followed by `Llama-3-8b` (by TeamUTK) and finally by the GPT baseline.

When we compare this ranking to the results presented in **Student** evaluation (Tables 6 and 11, discussed in Section 5.5), we see that the order of contestants given by the mean rating of the **Evaluator** systems is different from the order that ROUGE-L Recall and manual evaluation gave us. In particular, Section 5.5 concludes that the GPT baseline was better than both `LLaMA plus DeepSeek` (`Llama-2-7b` plus `DeepSeek-R1-Distill-Qwen-1.5b`, by LLMinds [8]) and `Llama-3-8b` (by TeamUTK).

This significant disagreement between the manual ratings and ROUGE-L Recall on the one hand and **Evaluator** average ratings on the other hand may indicate that the **Evaluator** system ratings are incorrect. This likely was not caused by any undue preference because of similar LLMs, because all **Evaluator** systems used models of different strains than those used in **Student** systems.

A more detailed analysis, inspecting the evaluations down to the level of domains and likely also individual questions and answers, would be necessary for a full explanation of this discrepancy. Our tentative conclusion is that **Evaluator** ratings are generally unreliable.

6.8. Teacher vs Student system ability to answer questions

Teacher task participants were asked to provide not just questions, but also reference answers, whenever possible. Not all participants did this, but when they did and when they also submitted a **Student** system, we can examine whether the answers made by the **Teacher**s are better than the answers made by the separate **Student** systems. We present this comparison in Table 23.

We can see that for the LLMinds team, the reference answers by their **Teacher** system are rated relatively closely behind the answers by their **Student** system. This is quite strange because for the reference answer, the LLMinds **Teacher** returned just an untreated small excerpt of the original material, which served as the basis for creating the question. Such excerpts are not phrased as an answer and do not point out the specific piece of information requested, even though they almost always contain all the relevant information. This suggests that **Evaluator**s' scores reflect some form of information overlap but are not very good at checking if the provided answer is actually in a form appropriate for answering the question.

For our GPT baseline, the **Evaluator** ratings of the answers constructed in the **Teacher** mode were, as expected, higher than the ratings of the answers provided in the **Student** mode. This could be explained by the fact that, in the **Teacher** mode, the model knew that the questions were answerable from the text because it generated them like that. It also probably helped that the baseline **Teacher** was prompted with the instruction that the answer to the generated question should be contained in the text. However, the fact that these reference answers are rated worse than the golden answers to expert-made questions again indicates some inadequacy on the part of the **Evaluator** systems or on the part of the GPT **Teacher** baseline.

6.9. Evaluator evaluation overview

Based on the accuracy in predicting the ROUGE-L Recall scores (Table 19) and considering the general match between the ROUGE-L Recall score and manual rating, as presented in Section 5.3, it seems that the **Evaluator** submission using `Gemma-3-27b-it` is the best. The other validation approaches and a limited manual qualitative review seem to corroborate this fact.

All **Evaluator** systems, however, seem to be very far from perfect. In particular, they are very easily fooled by adversarial inputs (Sections 6.1 and 6.2), such as nonsensical questions or answers. What is even harder to fix in the long run is their low reliability when the evaluated questions and answers are all on the same topic, but shuffled. In this setting, the tested **Evaluator** systems assign relatively high scores even to mismatching question-answer pairs.

Our further analyses also support our concern that evaluation systems that use currently available LLMs may be unreliable. For example, the discrepancy between **Evaluator** outputs and ROUGE-L Recall (which matches the manually assigned scores) in Section 6.7, or the high acceptance of direct citations of the original text snippets instead of proper answers to questions in Section 6.8. As another example, we give the occasional failures, even for the best performing **Evaluator**s, such as the counter-intuitive results of `GPT-4.1-nano` in Section 6.3.

7. Our observations for the LLM-as-a-Judge usecase

Across our experiments, we collected the first following observations, which will likely be relevant to anyone wishing to use LLMs for automatic scoring of various tasks, the so-called LLM-as-a-Judge setting. This strategy appeared twice in our shared task: (1) We used LLMs to evaluate our **Teacher** submissions, and (2) our shared task participants relied on LLMs in their **Evaluator** systems.

Looking at Section 6.1, we can see that **Evaluator** systems sometimes fail to distinguish between correct answers and obviously wrong ones. On the other hand, considering the more thorough evaluation of **Teacher** submissions in Appendix A, we can see that our rating of the **Teacher** systems is very stable under change of LLM and prompt. From these two observations, it seems that the evaluation of the **Teacher** system is significantly more stable than the evaluation of the **Student** outputs by the

Evaluator systems.[5] Here, we discuss the possible differences between the two settings and present them as recommendations for improving the LLM-as-a-Judge reliability. We nevertheless note that we have not yet conducted any experiments to verify these preliminary observations; we will likely do so in follow-up work.

There were many differences between our **Teacher** system evaluation and the **Evaluator** system outputs. In **Teacher** evaluation, we compare sets of questions instead of individual question-answer pairs rated in the **Evaluator** track. We also obviously only compare the text and the question when evaluating the **Teacher** systems as opposed to the text, the question, and the answer. The increased stability may be entirely due to the different distributions of rated items in both evaluations. However, manual review showed that the quality of the **Teacher** system outputs was, if anything, harder to distinguish. This should mean their rating should be more, not less, difficult. We identify the following factors as possible reasons why our **Teacher** evaluation seems to be more reliable than **Evaluator**s performance:

- **Range of output values must be carefully selected.** In the **Evaluator** track, to give maximum flexibility to our contestants, we allowed the ratings to be in the range 0–100. When experimenting with the creation of the **Evaluator** baseline and evaluating the **Evaluator** submissions, we noticed that when prompted with just the 0–100 range, the systems often output only the values 0 and 100. When we designed our **Evaluator** baseline system, we instead chose to use 0–5 and multiply the result by 20. In the **Teacher** systems, we only use the words "least" and "most" and only require the system to assign input question sets to statements to indicate rating, so the system does not have to interpret the meaning of numerical ratings.
- **Answer correctness is conflated with semantic similarity.** When we looked at how **Evaluator** systems rate nonsensical answers, we saw that when they were semantically similar, even almost nonsensical answers were often rated relatively highly. This is also corroborated by the inability to understand the specific requirements of a more complex question, as can be seen in Figure 1.
- **Judgements on text-level relations appear more stable.** We think that part of the increased stability of the evaluation of the **Teacher** system can be attributed to the fact that we only compare the text and the question as a whole. When evaluating the question answering from context, the LLM is required to consider both the meaning of the question in context and the correctness of the answer in context. In other words, evaluation of **Teacher**s needs less semantic inference and shallow topic overlap is sufficient, whereas evaluation of **Student**s needs to interpret both question and answer in the given context and assess their match.
- **Relative judgements appear more stable.** Another factor that could cause increased stability in **Teacher** evaluation is that our rating using `GPT-4.1-nano` was relative. This is not always possible, as we often want to rate a system in isolation and in absolute terms. We suspect that when processing the outputs of multiple systems at the same time, the LLM doing the rating can better understand its task, and thus become less prone to hallucination. We note that the relative rating could be emulated during the absolute evaluation using a few-shot method.

8. Conclusions

Systems that utilize LLMs can initially appear proficient in many tasks and seem to excel in text "understanding". In the Sensemaking task, we put this to a thorough, three-stage test. Participating systems and our baselines had to create questions to cover a given input material, answer them (based on the information in the material only) and evaluate such answers, again constrained by the input texts. This setting was inspired by the educational domain but applies to many other domains, including, e.g.,

[5] More tests of **Teacher** evaluation would be needed for full confidence, because for now we could not do the same kind of adversarial tests with our **Teacher** evaluation as we do with **Evaluator** systems.

fact-check analysis of potentially disinformative texts. As we have shown, most of the systems seemed to be similarly adept at handling all tested domains.

Our results indicate that, across the board, the models were sometimes able to generate and answer the questions. However, the results also show that they cannot be fully trusted to limit their knowledge to the materials provided.

The received **Teacher** submissions and our baselines show that LLMs are reasonably good at converting text to questions about its content. The coverage can be easily ensured by extracting questions from every portion of the input. At the same time, the reference answers, if provided at all, were not very good, so the question construction appears to be done in a way detached from the actual content.

The answers of the **Student** systems often did not make use of obvious clues about the answer in the materials. For example, in the fact-checking domain, the golden answer was spelled out explicitly in the text, even if a little hidden in a handful of unrelated statements, and yet the systems had difficulties extracting it.

The ratings produced by the **Evaluator** systems were unreliable and often were obviously wrong. This was striking, especially in our adversarial tests, where the **Evaluator** systems gave good scores to answers artificially damaged beyond comprehensibility, or to good answers connected to damaged questions. When the answers and questions were shuffled but generally coming from the same domain, the **Evaluator** systems were also fooled and gave relatively high scores. We see this result as an important caveat for the currently popular idea of using LLMs as judges.

Overall, our experiments with LLMs as judges seem to point to many imperfections and some promising potential paths. We hope to see more literature on the topic published and plan to do a thorough literature review in the future.

The main conclusion of Sensemaking 2025 is that it is very difficult to achieve consistent results and progress without good automatic measures for evaluating text understanding systems in the examined tracks. The most reliable methods we used here were those that built upon adversarial inputs.

In this sense, our quest for "sensemaking" and its evaluation has only just begun. If LLMs indeed have the ability to understand text or validate if an agent exhibits such understanding, this capacity is far from easy to reach reliably. Most contestants in this task have used relatively simple methods and small models with a small or no amount of inference-time thinking, so there is still much experimentation to do.

9. Limitations

Here we give a short summary of the limitations, including some already mentioned ones.

- Not counting our baselines, there were only two valid submissions for each track, barely enough to even have a comparison.
- Because the number of dataset items increases exponentially as the tracks build on each other, we needed to prune the datasets in order to keep the compute requirements manageable for our contestants; this decreased the number and therefore the variety of the material sections used.
- Submission items often have wildly different quality, so it is sometimes hard to compare some adversarial and non-adversarial entries when evaluating **Evaluator** systems.
- When evaluating, we and the **Evaluator** systems focus only on the material section provided, because it is difficult to take into account the entire material at once; some **Teacher** and **Student** systems managed this and thus had a better chance. In future editions, it will likely be useful to ensure that the tracks are defined in a way that lends itself to easier evaluation.
- In general, we ran Sensemaking 2025 in a rather exploratory way and the scope of this task thus did not allow us to focus precisely enough on the specific issues of each individual track. It might be beneficial to focus closely on only one of the tracks in the follow-up work and further editions of the task.

Acknowledgments

The authors acknowledge the support of the National Recovery Plan funded project MPO 60273/24/21300/21000 CEDMO 2.0 NPO and the funding from the Project OP JAK Mezisektorová spolupráce Nr. CZ.02.01.01/00/23_020/0008518 named "Jazykověda, umělá inteligence a jazykové a řečové technologie: od výzkumu k aplikacím."

Declaration on Generative AI

While LLMs (as a particular type of generative AI tool) are the central element in this study (they create questions, answer them, and evaluate the answers), they were not used in the production of this text. The only exception is the sentence polishing feature built into Overleaf, where we sparingly accepted the suggested improvements.

References

[1] J. Karlgren, K. Artemova, O. Bojar, M. I. Engels, V. Mikhailov, P. Šindelář, E. Velldal, L. Øvrelid, Overview of ELOQUENT 2025: shared tasks for evaluating generative language model quality, in: J. C. de Albornoz, J. Gonzalo, L. Plaza, A. G. S. de Herrera, J. Mothe, F. Piroi, P. Rosso, D. Spina, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Sixteenth International Conference of the CLEF Association (CLEF 2025), Springer Lecture Notes in Computer Science, 2025.

[2] Z. Wang, R. Baraniuk, MultiQG-TI: Towards question generation from multi-modal sources, in: E. Kochmar, J. Burstein, A. Horbach, R. Laarmann-Quante, N. Madnani, A. Tack, V. Yaneva, Z. Yuan, T. Zesch (Eds.), Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023), Association for Computational Linguistics, Toronto, Canada, 2023, pp. 682–691. URL: https://aclanthology.org/2023.bea-1.55/. doi:10.18653/v1/2023.bea-1.55.

[3] A. Farea, Z. Yang, K. Duong, N. Perera, F. Emmert-Streib, Evaluation of question answering systems: complexity of judging a natural language, arXiv preprint arXiv:2209.12617 (2022).

[4] E. Kamalloo, N. Dziri, C. L. A. Clarke, D. Rafiei, Evaluating open-domain question answering in the era of large language models, 2023. URL: https://arxiv.org/abs/2305.06984. arXiv:2305.06984.

[5] P. Rajpurkar, J. Zhang, K. Lopyrev, P. Liang, SQuAD: 100,000+ questions for machine comprehension of text, in: J. Su, K. Duh, X. Carreras (Eds.), Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, Austin, Texas, 2016, pp. 2383–2392. URL: https://aclanthology.org/D16-1264/. doi:10.18653/v1/D16-1264.

[6] P. Koehn, Neural machine translation, Cambridge University Press, 2020.

[7] D. Javorský, D. Macháček, O. Bojar, Continuous rating as reliable human evaluation of simultaneous speech translation, in: P. Koehn, L. Barrault, O. Bojar, F. Bougares, R. Chatterjee, M. R. Costa-jussà, C. Federmann, M. Fishel, A. Fraser, M. Freitag, Y. Graham, R. Grundkiewicz, P. Guzman, B. Haddow, M. Huck, A. Jimeno Yepes, T. Kocmi, A. Martins, M. Morishita, C. Monz, M. Nagata, T. Nakazawa, M. Negri, A. Névéol, M. Neves, M. Popel, M. Turchi, M. Zampieri (Eds.), Proceedings of the Seventh Conference on Machine Translation (WMT), Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Hybrid), 2022, pp. 154–164. URL: https://aclanthology.org/2022.wmt-1.9/.

[8] A. Sajdoková, M. Macek, O. Hlava, M. Štefanec, A. Kříž, J. Kučera, O. Bojar, Team LLMinds Submission for ELOQUENT Sensemaking Task, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 – Conference and Labs of the Evaluation Forum, CEUR-WS, 2025.

[9] S. Oh, H. Go, H. Moon, Y. Lee, M. Jeong, H. S. Lee, S. Choi, Evaluation of question generation needs more references, 2023. URL: https://arxiv.org/abs/2305.16626. arXiv:2305.16626.

[10] N. Reimers, I. Gurevych, Sentence-BERT: Sentence embeddings using Siamese BERT-networks, in: K. Inui, J. Jiang, V. Ng, X. Wan (Eds.), Proceedings of the 2019 Conference on Empirical Methods

in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3982–3992. URL: https://aclanthology.org/D19-1410/. doi:10.18653/v1/D19-1410.
[11] K. Lutsai, M. Thér, J. Venc, O. Bojar, ELOQUENT Sensemaking task - LLMs in the Evaluator role, in: G. Faggioli, N. Ferro, P. Rosso, D. Spina (Eds.), Working Notes of CLEF 2025 – Conference and Labs of the Evaluation Forum, CEUR-WS, 2025.

A. Examining the stability of the Teacher system evaluation

In Table 4, we can see that our **Teacher** baseline using `GPT-4.1-nano` dominates the evaluation. One reason for this dominance could be the similarity between `GPT-4.1-nano` doing the **Teacher** task and `GPT-4.1-`**`mini`** evaluating it. The question we ask in this section is thus: Does the **Teacher** evaluation change when we use different models or change the order of the requested entries in the prompt in Figure 4?

We chose to examine two alternative models for evaluation, `GPT-4.1-nano` and `Gemini-2.5--flash` as provided by the Google Gemini openAI compatibility API.[6] When we use the smaller one, i.e. `GPT-4.1-nano`, as seen in Table 24, we observe that the ratings of the `Mistral-7b` system increase dramatically (1.48 to 1.83, far away from the minimum rank), and the ratings of all other systems seem to decrease relatively uniformly. The questions the `Mistral-7b` system generated were often ranked worse in our manual review Table 5, which indicates that the `GPT-4.1-nano` system is a worse judge than `GPT-4.1-`**`mini`**. When we change the evaluation model to Gemini, we would expect that `GPT-4.1-nano` in the role of **Teacher** loses some of its dominance. However, this does not happen, so we conclude that `GPT-4.1-nano` is maybe not the best model for **Teacher** evaluation, but it was still the best **Teacher** in our experiments; see Table 25.

In our second branch of examination, we check the stability of outputs when varying the prompt. For simplicity, we do not modify the actual wording of the prompt (Figure 4) but only change the order in which the evaluated outputs are presented to the `GPT-4.1-`**`mini`** and `Gemini-2.5-flash` models.

Specifically, we experimented with swapping the pairs of “least” and “most” in the sequence, i.e. swapping the position and index of ENTRY1 with ENTRY2, ENTRY3 with ENTRY4, etc. The resulting ratings do not change much regardless if we stick to the `GPT-4.1-nano` model (Table 26) or use Gemini (Table 27).

Similarly, we do not see a large change even when we reverse the entire order of the entries in our prompt; see Tables 28 and 29.

[6] https://ai.google.dev/gemini-api/docs/openai

Table 24
Teacher submissions evaluated by `GPT-4.1-nano` (a different version than in Table 4). We report the average and standard deviation of the scores assigned to the provided sets of questions across all the tested materials. For an easy comparison of `GPT-4.1-`**`mini`** (used in Table 4) and `GPT-4.1-nano` (here), we also report the overall scores for each evaluated **Teacher** across the categories in the last two columns of this table.

type of evaluation	cover the material most	require thinking about different parts of the material	useful for learning the material	useful for learning to reason about the material	Overall	
questions by					4.1-mini	4.1-nano
DeepSeek plus LLaMA	2.18±0.82	1.53±0.77	1.76±0.74	1.66±0.66	1.77±0.76	1.78±0.78
Mistral-7b with BLIP	1.76±0.79	1.88±0.63	1.81±0.71	1.88±0.77	1.48±0.67	1.83±0.72
baseline from creators	1.71±0.77	2.11±0.72	1.86±0.80	2.14±0.72	2.15±0.65	1.95±0.77
baseline GPT-4.1-nano	**2.33±0.75**	**2.54±0.77**	**2.56±0.75**	**2.46±0.85**	**2.69±0.55**	**2.48±0.78**

Table 25
Teacher submissions evaluated by `Gemini-2.5-flash`. We report the average and standard deviation of the scores assigned to the provided sets of questions across all the tested materials. For an easy comparison of `GPT-4.1-`**`mini`** (in Table 4) and `Gemini-2.5-flash`, we also report the overall scores for each evaluated **Teacher** across the categories in the last two columns of this table.

type of evaluation	cover the material most	require thinking about different parts of the material	useful for learning the material	useful for learning to reason about the material	Overall	
questions by					4.1-mini	Gemini
DeepSeek plus LLaMA	1.96±0.73	1.82±0.69	2.04±0.81	1.88±0.75	1.77±0.76	1.93±0.74
Mistral-7b with BLIP	1.40±0.66	1.33±0.57	1.62±0.85	1.31±0.60	1.48±0.67	1.42±0.69
baseline from creators	2.19±0.79	2.14±0.76	**2.31±0.82**	2.06±0.58	2.15±0.65	2.17±0.74
baseline GPT-4.1-nano	**2.52±0.67**	**2.75±0.48**	2.19±0.66	**2.75±0.56**	**2.69±0.55**	**2.55±0.63**

Table 26
Results using `GPT-4.1-mini` with a similar methodology as in Table 4 but with order of entries in the prompt (Figure 4) swapped within each least–most pair. The last two columns again compare the overall scores across categories to `GPT-4.1-mini` with the original prompt.

type of evaluation / questions by	cover the material most	require thinking about different parts of the material	useful for learning the material	useful for learning to reason about the material	Overall	
					orig 4.1-mini	swapped 4.1-mini
DeepSeek plus LLaMA	1.94±0.74	1.70±0.71	**2.12±0.87**	1.82±0.77	1.77±0.76	1.90±0.79
Mistral-7b with BLIP	1.81±0.83	1.55±0.67	2.02±0.92	1.55±0.67	1.48±0.67	1.73±0.80
baseline from creators	2.17±0.97	2.17±0.74	2.11±0.95	2.00±0.63	2.15±0.65	2.11±0.83
baseline GPT-4.1-nano	**2.27±0.66**	**2.71±0.54**	1.92±0.48	**2.69±0.64**	**2.69±0.55**	**2.40±0.67**

Table 27
`Gemini-2.5-flash` with a similar methodology as in Table 25 but with order of entries in the prompt (Figure 4) swapped within each least–most pair. The last two columns again compare the overall scores across categories to `GPT-4.1-mini` with the original prompt.

type of evaluation / questions by	cover the material most	require thinking about different parts of the material	useful for learning the material	useful for learning to reason about the material	Overall	
					orig 4.1-mini	swapped Gemini
DeepSeek plus LLaMA	2.06±0.79	2.00±0.64	2.16±0.82	2.00±0.70	1.77±0.76	2.06±0.74
Mistral-7b with BLIP	1.38±0.62	1.29±0.55	1.55±0.86	1.19±0.45	1.48±0.67	1.35±0.65
baseline from creators	2.11±0.75	2.00±0.76	**2.36±0.87**	2.11±0.57	2.15±0.65	2.15±0.75
baseline GPT-4.1-nano	**2.52±0.64**	**2.73±0.56**	2.10±0.50	**2.73±0.60**	**2.69±0.55**	**2.52±0.63**

Table 28
Results using `GPT-4.1-**mini**` with a similar methodology as Table 4 but with order of entries in the prompt (Figure 4) reversed. The last two columns again compare the overall scores across categories to `GPT-4.1-**mini**` with the original prompt.

type of evaluation	cover the material most	require thinking about different parts of the material	useful for learning the material	useful for learning to reason about the material	Overall	
questions by					orig 4.1-mini	reversed 4.1-mini
DeepSeek plus LLaMA	2.04±0.78	2.04±0.86	1.98±0.80	2.04±0.78	1.77±0.76	2.02±0.80
Mistral-7b with BLIP	1.76±0.91	1.50±0.59	1.95±0.96	1.48±0.63	1.48±0.67	1.67±0.81
baseline from creators	2.14±0.87	1.83±0.65	**2.14±0.83**	1.81±0.67	2.15±0.65	1.98±0.77
baseline GPT-4.1-nano	**2.21±0.67**	**2.65±0.59**	2.10±0.69	**2.69±0.58**	**2.69±0.55**	**2.41±0.68**

Table 29
Results using `Gemini-2.5-flash` with a similar methodology as in Table 25 but with the order of entries in the prompt (Figure 4) reversed. The last two columns again compare the overall scores across categories to `GPT-4.1-**mini**` with the original prompt.

type of evaluation	cover the material most	require thinking about different parts of the material	useful for learning the material	useful for learning to reason about the material	Overall	
questions by					orig 4.1-mini	reversed Gemini
DeepSeek plus LLaMA	2.14±0.76	1.98±0.77	**2.24±0.69**	1.98±0.68	1.77±0.76	2.08±0.73
Mistral-7b with BLIP	1.50±0.74	1.48±0.67	1.95±0.94	1.26±0.54	1.48±0.67	1.55±0.77
baseline from creators	**2.33±0.86**	2.22±0.76	2.19±0.89	2.06±0.71	2.15±0.65	2.20±0.81
baseline GPT-4.1-nano	2.19±0.69	**2.44±0.73**	1.65±0.62	**2.73±0.53**	**2.69±0.55**	**2.25±0.75**