

AI for Automatic Feedback on Assessment Portfolios in Secondary Education

KEVIN NIEUWENHUIS, University of Twente, The Netherlands

The purpose of this paper is to support a program designed to assist Dutch secondary education teachers in assessing student assignments. The teachers' work is graded by instructors based on a set of criteria, with a pass/fail rating and an optional comment. A locally-run LLM is used to generate feedback instead of the instructors. Running LLMs locally requires smaller models, which perform worse in Dutch. To find out the quality of using these smaller LLMs to generate feedback, we first analyzed the language performance in Dutch with various benchmarks for Gemma-3-27B-it. Furthermore, several approaches were explored to generate feedback with the model, including incorporating summaries of textbook chapters and an answer model. Feedback quality was manually assessed by an instructor, and automatically via pass/fail agreement with the reference feedback and using the LLM-as-judge framework, G-Eval, with Qwen3-32B. After analyzing the performance of Gemma-3-27B-it, we found that the model shows strong comprehension, but still struggles with understanding specific semantics of a sentence. Moreover, we found that supplying extra context of the textbook and the corresponding answer model in the prompt improves the quality of the feedback, according to one of the involved instructors. The scores from G-Eval with Qwen also returned the same conclusion. For future work, the system should have an appropriate interface for which the generated feedback can be further evaluated in practice. Our contributions with this paper are (1) showing how the Dutch language performance of LLMs can be analyzed, (2) how extra context can be used to improve the output of generated feedback, and (3) smaller LLMs can be used to evaluate content using LLM-as-judge.

Additional Key Words and Phrases: Automated feedback assessment, Automated feedback system, Feedback generation, G-Eval, Large language models, LLM benchmarking, LLM evaluation, LLM-as-judge, Prompt engineering, RAG

1 INTRODUCTION

A program has been developed to support Dutch secondary education teachers in developing their assessment skills. Each participating teacher has a portfolio consisting of different assignments focused on different assessment topics. For these assignments they have to use an accompanying textbook. Their portfolio undergoes a formative assessment where the teachers get feedback on their portfolios based on learning goals and success criteria, which they can use to improve their portfolio. At the end, the teachers submit the final version of the portfolio, which undergoes a final summative assessment.

The DMB research group at the University of Twente was asked to find a solution to automate the portfolio assessment using generative AI. The reason why this project was requested from the DBM research group is to save *time* and *money*. Hiring or recruiting teachers to grade the portfolios requires a significant amount of time and costs a considerable amount of money, for example, the average

hourly wage of a secondary education teacher in the Netherlands is between 25–35 euros [51].

An existing approach would be using LLMs (large language models) such as ChatGPT, DeepSeek, or Claude to generate feedback. LLMs are trained on vast amounts of text data and can perform various NLP (natural language processing) tasks with high accuracy such as generating human-like responses to questions [23]. They can give feedback on the assignments faster than teachers and are more cost-efficient. However, when using large LLMs, their commercial APIs have to be used. With this, there are privacy issues involved, such as that the data might be used to train the models, or a data leak might happen. A data leak could compromise the privacy of the participating teachers or their associated school. To mitigate this, smaller LLMs can be used, which can be run locally on the computers of the University. In addition, it costs less money to use the smaller models.

The aim of this thesis is to explore the capabilities and limitations of using smaller LLMs to assess the teacher's portfolios in Dutch.

2 PROBLEM STATEMENT

There is still a problem for small LLMs; their performance has not been thoroughly tested for the Dutch language. They do not always perform well in Dutch, even when models such as Qwen3-32B claim that they support the Dutch language¹. The university hosted the two models Qwen3-32B² and Gemma-3-27B-it³ for this paper. To obtain an initial impression of the performance of smaller models, we can compare these models with the output from ChatGPT using the prompt in Appendix A.1. The prompt asks for a detailed description of the Dutch breakfast, which also tests the cultural knowledge of the LLM. The output of Qwen, Gemma and ChatGPT is displayed in Appendix D.1, D.2, and D.3 respectively. Gemma's output appears to be grammatically correct, although it contains one spelling error, but the response from ChatGPT still sounds more natural. The response from Qwen, however, includes unnatural sentences, illogical words, English words, or simply cultural misinformation. The unnatural sentences and English words could be artifacts from English, since the models are mostly trained on English data. Non-existing words could occur because of hallucinations. In the context of LLMs, hallucinations refer to generated text that does not make sense or is not supported by its training data [20].

In this paper we try to determine how to provide feedback on assignments in Dutch using a small LLM. This problem statement can be divided into three research questions:

- **RQ1:** How do we measure the performance of the Dutch language of an LLM?
- **RQ2:** How can LLMs be used to generate feedback on assignments based on their learning goals and success criteria?

TSelT 43, July 4, 2025, Enschede, The Netherlands

© 2025 University of Twente, Faculty of Electrical Engineering, Mathematics and Computer Science.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

¹<https://qwenlm.github.io/blog/qwen3/>

²<https://huggingface.co/Qwen/Qwen3-32B>

³<https://huggingface.co/google/gemma-3-27b-it>

- **RQ3:** How should the generated feedback be evaluated?

We approach this problem by first benchmarking Gemma-3-27B-it. We then try several approaches to generate feedback, including adding context from the textbook, and supplying an answer model to the LLM. The context is provided as a summarized version of the relevant chapters of the textbook or the top three documents from the retriever component of the RAG mechanism. After this, we evaluate the feedback against the instructor's feedback and we use G-Eval with several metrics to measure the content of the output and the use of the provided textbook context. We compare these evaluations with the input we got from one of the instructors on the generated feedback. Our contributions are (1) showing how the Dutch language performance of LLMs can be analyzed, (2) how extra context can be used to improve the output of generated feedback, and that (3) smaller LLMs can be used to evaluate content using LLM-as-judge.

3 BACKGROUND

LLMs are defined by multiple key attributes. One of these is the parameter size [35]. The parameter size refers to the number of trainable values which are pre-trained on vast amounts of text and used by the LLM to generate output. Models like ChatGPT (GPT-3.5 or GPT-4) contain more than billions or trillions of parameters, the exact number being unknown. The smaller models used in this paper have around 32 billion parameters. Usually, the more parameters a model has, the more complex language patterns and calculations it can reproduce, but this also depends heavily on the training and data quality [25].

Another key attribute of an LLM is the context length, which is the maximum number of tokens it can process in a single input [45]. A larger context length is useful for handling longer texts and maintaining extended conversations [41]. In chat applications, the chat history is added before the current query. If the total input becomes too long, the oldest messages are removed [45].

To get the most out of LLMs, prompt engineering is often used. It refers to optimizing prompts to effectively use LLMs [14]. One common strategy is role prompting, where the LLM is asked to role play as a persona [50].

Before an LLM can process text, the text first undergoes tokenization, where the text is broken down into smaller units called tokens [15]. These tokens are converted into numerical representations for the model called embeddings [24]. When an LLM generates text, it considers multiple tokens to output, each with a different log probability. The model typically chooses the token with the highest probability [52]. This can be adjusted by changing the temperature of an LLM. The temperature controls the randomness inside the generated text. A lower temperature makes the output more deterministic, whereas a higher temperature increases the diversity and creativity by allowing less probable tokens to be selected [2].

The knowledge of LLMs is stored inside the parameters. After training, the knowledge of the model may no longer be up-to-date, but it is an expensive operation to add extra knowledge using extra training or fine-tuning. During fine-tuning, the parameters of a model are updated for a specific task or knowledge base [35]. Instead, RAG can be used to provide extra relevant information. This also

reduces hallucinations [29]. RAG stands for retrieval-augmented generation and was first introduced in the paper by Lewis et al. [29]. Using RAG, an extra knowledge base can be supplied to the LLM. The simplified mechanism of RAG is shown in Figure 1. RAG works by first passing the query of the user to the retriever. This retriever uses a dense vector search to find the top-K documents from the knowledge base, based on similarity with the input query. The knowledge base is turned into dense vectors using an embedding model, which transforms the text into embeddings. Dense vectors, unlike sparse vectors, contain mostly non-zero values. These documents and the original question are passed to the generator, which is the LLM we use.

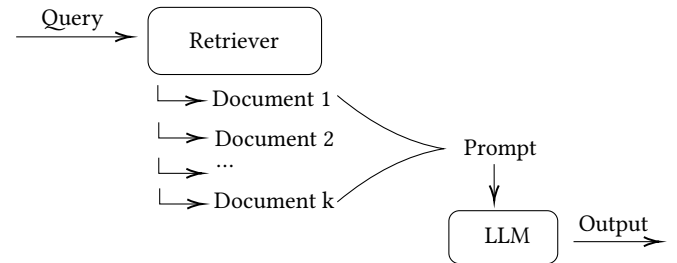


Fig. 1. Retrieval-Augmented Generation Pipeline

To evaluate the output generated by LLMs, specific benchmarks can be used. LLM benchmarks are evaluation frameworks that are used to test the capabilities of an LLM [57]. This mostly entails testing the size of their knowledge base, their ability to reason and to test their ability to understand natural language text, as the benchmarks in [9, 19, 26]. These benchmarks are large datasets with multiple choice questions or automated metrics that calculate the similarity between the reference output and the model output. Other benchmarks also check for possible biases in an LLM [17].

4 RELATED WORK

4.1 RQ1: Measuring language performance

Various benchmarks for LLMs have been translated or made for the Dutch language [48, 49]:

- ARC (AI2 Reasoning Challenge) [40] measures reasoning capabilities; translated by the University of Oregon [27] using ChatGPT.
- Belebele [3] is a multiple choice reading comprehension test translated from English.
- CONLL-nl (Conference on Computational Natural Language Learning) [46] is used to test entity recognition, which is the identification of proper nouns in a sentence [30].
- DBRD (Dutch Book Reviews Dataset) [6] tests sentiment analysis, to test whether a piece of text has a positive or negative connotation.
- Dutch CoLA (Corpus of Linguistic Acceptability) [16] is used for grammatical acceptability.
- The Global MMLU NL (Massive Multitask Language Understanding) [44] is based on the English MMLU [19], which tests knowledge and understanding.

- HellaSwag NL [1, 40] is a translated version of HellaSwag [53], to test common sense.
- ScaLA-NL, ScaLA was first published for Scandinavian languages only [39], and was later created for Dutch using the Dutch Universal Dependencies treebank [47]. This treebank was used to corrupt sentences to create grammatically incorrect sentences [12].
- SQuAD-NL v2.0 (Stanford Question Answering Dataset) [37] for reading comprehension and question answering.
- WikiLingua NL [26] tests summarization skills using WikiHow.
- XL-WiC-NL (Word-in-Context) [43] for word sense disambiguation, made using the Dutch WordNet [42].

The benchmarks can be split up into natural language understanding (NLU) and natural language generation (NLG). The only benchmark that tests NLG is WikiLingua, the rest tests NLU.

The translated benchmarks are the ARC, Belebele, SQuAD-NL v2.0, HellaSwag NL, and Global MMLU. Yet, translated benchmarks often fail to fully capture the expressions and cultural context of the translated language [22]. The Global version of MMLU tried to prevent this by focusing on minimizing cultural biases during its design [44]. Furthermore, in the ARC dataset we could infer several translation mistakes.

Most of these benchmarks (DBRD, CONLL NL, ScaLA-NL, SQuAD-NL, WikiLingua NL, Global MMLU NL, HellaSwag NL) have been tested on various models using the EuroEval [38] package in Python. The results of these have been made available on their website⁴. The benchmarks not included in EuroEval are the ARC, Dutch CoLA, and XL-WiC-NL.

One of the issues with the benchmarks previously mentioned is that they do not measure the generation of fluent and correct Dutch text, since the models are not requested to generate any text-style content [48, 4]. The only exception is WikiLingua, which checks the summary of the LLM with that of a human comparison metric, such as BLEU or ROUGE. Their scores, however, have been proven to have a low correlation with human judgment [32]. It is therefore not guaranteed that a summary is coherent or that it captures the key points that a human reader would expect.

Another issue is that the benchmarks do not test whether the output of a model aligns more with the preferences of humans [31, 56]. The actual performance of the LLM is thus not always reliably reflected. Furthermore, the datasets of these benchmarks are static and openly available, so the training data of the models could be contaminated with the test data of the benchmarks [55].

Chatbot Arena was made to overcome the limitations of benchmarks. It is an LLM evaluation framework introduced in the paper by Zheng et al. [56]. For Chatbot Arena, crowdsourcing is used to let humans judge the best LLM response between two LLMs, where the user can write their own input prompt. This reflects the quality of the output with human alignment more than traditional benchmarks. The results of the performance of the LLMs are however not filtered for the Dutch language.

4.2 RQ2: Using an LLM to give feedback

Dai et al. [10] explored the usability of ChatGPT (GPT-4, GPT-3.5) in giving feedback on assignments. The following prompt was used to generate feedback, which includes the grading criteria of the assignment:

Please give feedback on the following text in terms of a clear description of the goals of the project, appropriateness of the topic to data science, a clear description of the business benefits, novelty/creativity and overall clarity of the report. <INSERT THE TEXT OF A REPORT>

The feedback from ChatGPT was found to be easier to understand than from the teachers. GPT-4 seemed to be better at including most aspects of the feedback compared to GPT-3.5 and the human instructors, according to the framework by Hattie and Timperley [18]. While GPT-4's feedback was more comprehensive, it did not fully align with that of the teachers. The evaluation was limited to ChatGPT models and did not consider smaller LLMs, which are the focus of this study.

Building upon this, Matelsky et al. [34] proposed a different feedback generation approach that involves using open-ended questions, and grading criteria to prompt an LLM to generate feedback on student answers. The LLM first solved the question without the criteria, then evaluated its response using them. If it identified any gaps between its answer and the criteria, it modified the question to better align with the expected evaluation. The prompt they used can be seen in the `get_feedback` function in their GitHub repository [33]. In the prompt the model is first assigned the instructor role, after that, the assignment and 'secret' grading criteria are put in the prompt, to let the model give its own answer. The student's answer is given inside a new prompt and the model is asked to examine the differences between their answer and the student's answer. This difference is used to give constructive feedback to the student. However, this approach is not applicable to the feedback generation in this paper, as the answers to most assignments are highly dependent on the teacher's school and personal experience.

The second prompting technique diverges from the first in that it first asks the model to provide its own response and then compare it to the student's, rather than simply asking for feedback.

Another aspect that can be incorporated into prompts is in-context learning, which can be under divided into zero-shot, one-shot, and few-shot learning. For zero-shot learning no example is given in the prompt, for one-shot learning one example is given in the prompt, and for few-shot learning, multiple examples are given in the prompt. This has been shown to greatly improve performance on various tasks [5].

4.3 RQ3: Evaluating generated feedback

In the paper previously mentioned in Section 4.2 from Dai et al. [10], the generated feedback was evaluated against the instructor's feedback by human annotators. The generated feedback was rated on readability, effectiveness, and reliability. For this paper we unfortunately do not have the human resources to annotate the generated feedback.

⁴<https://euroeval.com/leaderboards/Monolingual/dutch/>

The other paper from Matelsky et al. [34] focused on the pipeline implementation of the generated feedback and does not discuss the evaluation of the feedback.

A different evaluation strategy is presented by another paper, which uses GPT-3.5 and GPT-4 to review research papers [58]. The generated reviews are evaluated using a large set of multiple-choice questions that assess the correctness of the generated reviews. However, for this paper, we do not have the resources to create such questions to evaluate the generated feedback.

To compare generated text with a reference text, embedding-based metrics can be used, which measure the semantic similarity between the two texts [32]. This is done by using word or sentence embeddings, where the word or sentence is converted into numerical vectors. One example of this is the BERTScore [54] which measures the similarity between two texts by calculating the cosine similarities using the contextual embeddings from BERT [11]. However, these methods show weak correlation with human judgments, especially for dialogues with open-ended or domain-specific tasks [8]. Moreover, the comments in the feedback from the instructor are very concise and often missing, which would not be a good comparison with the rather descriptive comments of the LLM.

LLMs can also be used to evaluate the generated output. This concept is called LLM-as-a-judge [7]. Using GPT-4 as judge to select the better of two responses, an 85% agreement with human judgment was achieved on MT-Bench, surpassing the 81% inter-human agreement [56]. MT-Bench is a benchmark of multi-turn, instruction-following questions used to evaluate dialogue quality.

Two examples of the LLM-as-judge framework are GPTScore and G-Eval. GPTScore leverages large GPT models, such as GPT-3, to assess how likely a given text was generated according to a specified rubric [13]. G-Eval [32] uses evaluation criteria to output a score for a given output. It requires as input the task and answer to be evaluated. In addition, the evaluation criteria are provided for the generation of evaluation steps using a chain-of-thought (CoT) approach, allowing the LLM to produce a reasoning process for evaluating the task and formulating an answer. To evaluate the output, G-Eval applies a scoring function that calls the target LLM with both the evaluation steps and the generated text, and then calculates a score weighted by the token-level probabilities. Compared to GPTScore, G-Eval uses a form-filling problem where the LLM gives a score, whereas GPTScore only uses the probabilities from the output. G-Eval outperforms GPTScore and embedding-based metrics based on the agreement with the scoring from humans. Nevertheless, it still does not achieve perfect agreement. It is important to note that G-Eval uses ChatGPT as LLM in their paper, and since the capability of the evaluation depends on the LLM, it could perform lower if we use smaller models.

One limitation of using LLMs for evaluation is that they can introduce a self-enhancement bias, where the LLM favors their own output compared to others [56]. This can simply be mitigated by not letting an LLM evaluate its own work.

An improvement from LLM-as-judge is cooperative multi-agent evaluation [7], where instead of only judging, the LLMs have a conversation to discuss each other's answers. As a result, biases such as self-enhancement and positional bias are minimized [7]. This approach is however more complex: it requires more computation

cost and the conversation between the LLMs could fall in a loop or become incoherent [7].

5 METHODOLOGY

5.1 RQ1: Measuring language performance

To measure the general performance of LLMs, we use several Dutch language benchmarks mentioned in Section 4.1. We take the average of three runs to improve the reliability of the scores.

We chose to test the model Gemma-3-27B-it, because, at first glance, it seemed to perform the best. We evaluated it using the following benchmarks: the ARC, Belebele, the Dutch CoLA, Global MMLU, and the XL-WiC-NL. The prompt used for each benchmark is listed in Appendix A.2. We chose these benchmarks because they are not included in EuroEval's library, except for the Global MMLU, which we used to see the different performance between the categories of the MMLU. We compared EuroEval's results of Gemma-3-27B-it with the results of the highest ranked model of ChatGPT.

We also checked for spelling mistakes, and grammar mistakes using the automated tool LanguageTool. The breakfast prompt from Appendix A.1 was used as input for the LLM and the outputted language was checked using a python wrapper of LanguageTool [28], to see how many errors would be listed.

5.2 RQ2: Using an LLM to give feedback

Before generating the feedback we first had to preprocess the dataset we received. This dataset included .docx documents of multiple assignments containing the answers of the students, along with the feedback from the instructor, and the feedback sheet used by the instructor. The identifiable personal information of the teachers was removed from the assignments.

For the grading process, there is a rubric for each assignment containing several learning goals and success criteria. For each criterion, the teachers receive a pass or a fail. They also receive an optional comment, mostly when a fail is obtained for that specific criterion. For the LLM it is however preferred that it gives a more descriptive comment for each criterion. An example of a rubric is listed in Appendix B.

To generate the feedback, we first assign the role of teacher to the LLM and give it the instruction to generate feedback for the student. For this paper, the students are in fact the teachers. A layout in Markdown is specified for the LLM to use. This layout includes a pass/fail for each criterion, one positive point and three points of improvement. We also supply the answer of the teacher and the criteria of the assignment to the LLM.

We use various approaches to generate the feedback with the model Gemma-3-27B-it. In RQ3 we look at what approach is the best by evaluating the different techniques.

The first two different techniques we used is generating the feedback per criterion, as described in Appendix A.3.1, where the model is called separately for each criterion. In contrast, for generating the feedback all at once, it is only called once as seen in Appendix A.3.2.

To incorporate information from the textbook, we cannot include the full textbook in the prompt, as it would exceed the context length. That is why we either included the top three documents

retrieved by the retriever component of the RAG mechanism, or we added the summarized chapters of the textbook as context. The summarization was performed by Gemma.

Thus, for providing context from the textbook we chose the following approaches:

- Providing no context at all; as control group.
- Creating a retriever for each assignment, using the chapters relevant for that assignment.
- Providing the relevant chapters, summarized, for each assignment.

To select an embedding model for the retriever component of RAG, we consulted the MTEB leaderboard [36] on Hugging Face⁵, which evaluates models across a diverse set of embedding benchmarks. The top-ranked model, gemini-embedding-001, was not open-source. The next highest entries were Qwen3-Embedding models (8B, 4B, and 0.6B). We selected one of these because it offers a 32K context window, which is suitable for long textbook passages, and because it is a multilingual model. It was possible to run the 0.6B parameters model on our own hardware (Windows 11 with an Intel Core i7 processor, 32 GB of RAM, and NVIDIA T600 Laptop GPU).

Besides the context, we could use the answer model for one of the assignments, which was used to assist the model in its feedback. We supply the answer model in the prompt, as detailed in Appendix A.3.3, with the instruction to base the feedback as much as possible on the answer model.

Since we are using two models, we checked whether the two models roughly give the same answer for each criterion. If they give the same answer, it increases the chance of the answer being correct. This was done by comparing the text embeddings using the BERTScore, which outputs a similarity score. We compared it for the feedback that was generated using summarized textbook chapters as context, with and without providing the answer model.

5.3 RQ3: Evaluating generated feedback

The first evaluation technique we employed is the most straightforward: the generated feedback is compared to the instructor’s feedback to determine whether it assigns the same pass/fail outcome for each criterion. Additionally, we verify that each criterion appears exactly once in the output. However, this method does not take the textual comments from the instructor’s feedback into account.

To evaluate the content of the generated feedback, we used the LLM-as-judge framework. We chose to use G-Eval for this, since G-Eval was proved to correlate more with human judgment compared to the other evaluation techniques. For G-Eval, Qwen3-32B was used to evaluate the feedback from Gemma, to prevent self-enhancement bias from Gemma. To apply G-Eval, it is necessary to define what metrics the LLM will use for evaluating. We defined the evaluation steps for each metric beforehand with the help of GPT-4. The metrics evaluate either the output or the provided context. The context is either the output of the retrieval component of the RAG, or it consists of the summarized chapters. The evaluation steps for each metric are listed in Appendix C. The metrics are:

- **Correctness:** evaluates how well the generated output aligns with the instructor’s feedback.
- **Relevance:** assesses if the feedback appropriately addresses the input and the given criteria.
- **Coherence:** determines whether the feedback is logically structured and easy to follow.
- **Tonality:** checks whether the feedback has an appropriate tone.
- **Relevance (for the context):** verifies that the context is relevant to the input.
- **Faithfulness (for the context):** examines whether the output accurately incorporates information from the retrieved context.
- **Correct book chapter (for the context):** confirms whether the context includes the correct chapter, especially when the assignment specifies one.

The template prompt used for G-Eval is inspired by the ones used in the deepeval library [21] and the G-Eval paper [32]. In the prompt the evaluation steps are given, and the LLM is asked to give a score and a reason why, based on the evaluation steps for a given metric. The full template prompt can be found in Appendix A.4.1.

For each metric, the LLM gives a score from 0 to 10. This score is normalized using the probabilities from the log probabilities of the top 5 tokens the LLM considered for its output, ignoring insignificant linear probabilities below 1%. We use a slightly modified formula compared to the formula specified in the paper by G-Eval [32], which was implemented in Python by deepeval [21]:

$$\text{score} = \frac{\sum_{i=1}^5 p(s_i) \times s_i}{\sum_{i=1}^5 p(s_i)} \quad (1)$$

In Equation 1, s_i is one of the scores inside the top 5 tokens. $p(s_i)$ is the probability of the token consisting of the score s_i . Only the top 5 tokens are used, and therefore the probabilities $p(s_i)$ do not sum to 1. Therefore, the scores have to be normalized by dividing it by the sum of the weights of $p(s_i)$. This is different from the original formula from G-Eval which iterates over all tokens. We do not use all tokens, because most tokens below the top 5 have a very insignificant probability and that would be more likely to introduce noise than to give a more meaningful score.

Finally, our test plan for evaluating the feedback consists of three stages:

- (1) Testing per criterion (only pass/fail).
- (2) Testing the context generation.
- (3) Testing the answer model (for assignment 2).

For each stage, we measured the pass/fail agreement between the LLM and the instructor’s feedback, as well as the G-Eval metrics. We only look at the pass/fail agreement for the first stage, because for evaluating the generation per criterion, G-Eval evaluates the output differently per criterion, because the input is different for each criterion.

6 RESULTS

6.1 RQ1: Measuring language performance

The results for the benchmarks not included by EuroEval, run on Gemma-3-27B-it, are listed in Table 1. The specific result for each

⁵<https://huggingface.co/spaces/mteb/leaderboard> (accessed on 09/06/2025)

Benchmark	Result
ARC	90.70%
BeleBele	88.37%
CoLA	70.33%
MMLU	70.17%
XL-WiC NL	58.14%

Table 1. Benchmark results for Gemma-3-27B-it, averaged over three runs.

Benchmark	Result	English equivalent result
CONLL NL	42.60 $\pm$ 1.19 / 24.65 $\pm$ 0.67	62.83 $\pm$ 0.53 / 43.13 $\pm$ 0.53
DBRD	82.32 $\pm$ 0.59 / 91.00 $\pm$ 0.30	-
ScaLA NL	43.98 $\pm$ 0.86 / 68.57 $\pm$ 0.57	-
SQuAD NL	70.73 $\pm$ 0.29 / 50.19 $\pm$ 0.37	77.59 $\pm$ 0.41 / 55.48 $\pm$ 0.66
WikiLingua NL	64.49 $\pm$ 0.04 / 14.58 $\pm$ 0.06	-
MMLU NL	64.78 $\pm$ 0.85 / 73.54 $\pm$ 0.64	69.71 $\pm$ 0.71 / 77.29 $\pm$ 0.54
HellaSwag NL	55.16 $\pm$ 0.98 / 64.89 $\pm$ 0.71	66.72 $\pm$ 0.77 / 74.49 $\pm$ 0.59

Table 2. Gemma-3-27B-it (zero-shot) EuroEval benchmark results with their English equivalent if available.

category of the MMLU, averaged over three runs, is listed in Appendix E.1. All the categories have a result of around 70%, whereas the humanities category is significantly lower with 60%.

In addition to the benchmark scores, we also evaluated the grammatical correctness using LanguageTool over three runs, listed in Table 4. One of the verbose results including suggestions, can be found in Appendix E.2. In total, out of the 21 errors, 3 were true positives. Most spelling errors were flagged due to the presence of foreign words or uncommon compound constructions that, while correct in Dutch, are not always listed inside dictionaries.

To support our findings, we reference publicly available benchmark results from EuroEval⁶. The results for Gemma-3-27B-it and the highest ranked model of ChatGPT (gpt-o3-2025-04-16) are shown in Table 2, 5.

6.2 RQ2: Using an LLM to give feedback

The generated feedback followed the exact layout specified, and the positive points and improvement points listed in the feedback appeared to be different inside the responses. Furthermore, the Dutch language used in the generated feedback was acceptable and appropriate, as confirmed by one of the instructors. Additionally, the

feedback seemed to reference the content from the provided textbook passages. However, the feedback was not perfect and contained several mistakes, but most of the points in the feedback were still of use for the instructor. The retrieved documents by the retriever component did not always manage to retrieve the most relevant documents when the criterion was referencing a specific paragraph number. An example of the feedback generated by Gemma can be found in Appendix F where the summarized textbook chapters were supplied as context and the feedback was generated in a single pass.

When comparing the feedback generated by Gemma and Qwen, the average F-score for without the answer model was 0.792. When the answer model was included, the average F-score was almost the same at 0.793. When comparing a few samples of the feedback with each other, it was clear that the same criterion was discussed, but slightly different feedback was given.

6.3 RQ3: Evaluating generated feedback

Discussions with one of the instructors responsible for assessing the assignments revealed that adding additional context into the prompts improved the performance of feedback generated by Gemma, and that providing the answer model further improved the quality of the generated feedback.

For the first testing stage, we tested generating the feedback per criterion and all at once, without any context. We found that when generating the feedback per criterion, often not all the criteria or more than the specified criteria were included. For generating all at once, it happened 4 times, whereas for generating it per criterion this happened 22 times.

For the second testing stage we chose to generate the criteria all at once and as context we provided the top 3 documents retrieved by the retriever component of RAG, or we provided the summarized chapters of the book. In Figure 2 the scores for each metric for the different types of context can be seen. The scores for the summarized chapters are higher for almost all metrics, except for the “Correctness” and “Relevance (for the context)” metric.

For the third stage we tested if the metric scores are improved when providing the answer model for assignment 2, with as context summarized chapters. We only evaluated the feedback of assignment 2 for this stage. In Figure 3 the detailed metric scores can be seen. The scores for the answer model are higher for all metrics, except for the “Pass/fail” and “Book chapters” metric, and again lower for the “Correctness” and “Relevance (for the context)” metrics.

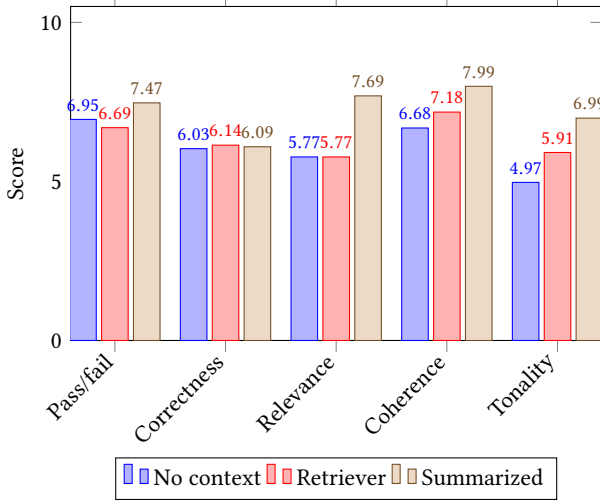
7 DISCUSSION

7.1 RQ 1: measuring LLMs

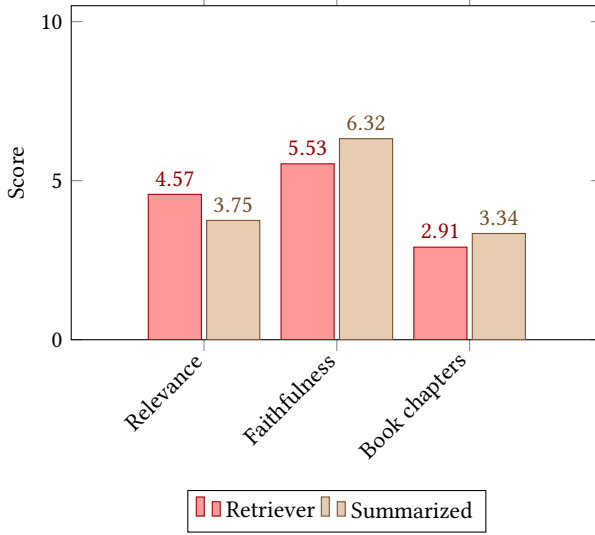
The LanguageTool results for Gemma indicate that the model makes only occasional and minimal spelling errors, demonstrating a strong proficiency in generating grammatically correct Dutch.

The benchmark results for Gemma in Table 1 show a high score for the ARC and BeleBele benchmarks, demonstrating high reasoning and reading capabilities. The average scores on CoLA and MMLU suggest a lesser known factual and grammar knowledge. The lower score for XL-WiC NL shows that the model still has difficulty distinguishing subtle semantic differences. From the benchmark results in Table 2 it can be seen that the model underperforms in

⁶<https://euroeval.com/leaderboards/Monolingual/dutch/> (accessed on 13/05/2025)



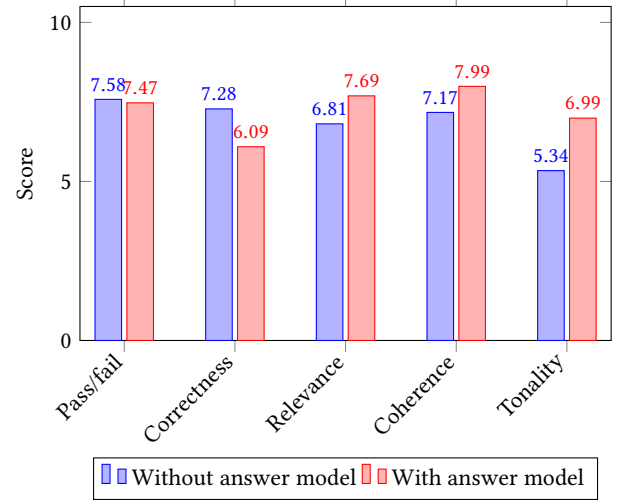
(a) Standard metrics



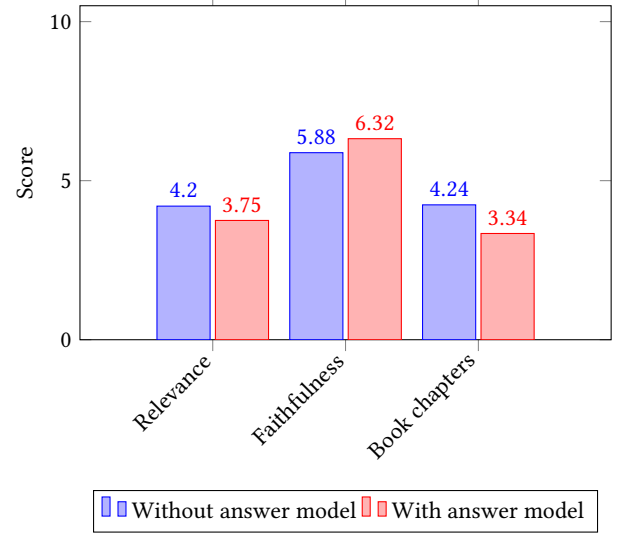
(b) Context-based metrics

Fig. 2. Comparison of metric scores for supplying no context, using a retriever per assignment, and inserting summarized chapters into the prompt.

Dutch for most benchmarks compared to its English equivalent, but for the SQuAD NL, MMLU NL, and HellaSwag NL the scores are not significantly lower, demonstrating a high reading understanding and factual knowledge in Dutch compared to English. From the high score of DBRD we can conclude that it performs well on sentiment analysis, further supporting the conclusion that the model demonstrates good reading comprehension. The low score for the benchmark CONLL NL also shows that the model still struggles with understanding specific semantics of a sentence. To conclude, Gemma-3-27B-it has strong reading comprehension in Dutch comparable to English and a good factual knowledge in Dutch almost comparable to English, but it still struggles to understand detailed and specific semantics of the language.



(a) Standard metrics



(b) Context-based metrics

Fig. 3. Comparison of metric scores for with and without the answer model in the prompt for assignment 2.

We compared Gemma's results with those of the highest-ranked ChatGPT model on EuroEval (o3-2025-04-16), as shown in Table 5. For all Dutch benchmarks, ChatGPT scores higher than Gemma. Since these benchmarks test factual knowledge and reading comprehension, it is expected that ChatGPT outperforms Gemma, because it has a larger number of parameters.

As mentioned in the related work section, benchmarks have several limitations, such as the fact that they do not measure the generation of fluent and correct Dutch text, and that they do not align well with human preferences. Furthermore, the training data of the model could have been contaminated with the data from the benchmark.

Since it is not fully reliable to measure the performance of the Dutch language using benchmarks, for further study it would be better to let humans show their preference between different LLMs, as done by Chatbot Arena [56]. This platform, however, has no filter for the Dutch language, so implementing this for Dutch could be something to consider.

7.2 RQ2: Using an LLM to give feedback

The results indicate that the LLMs are capable of generating feedback following predefined learning goals and success criteria. As the retriever did not always return the correct documents when a paragraph is mentioned, it is better to provide the summarized chapters, which is also supported by the results from research question 3.

Qwen and Gemma reach a fairly high F-score from the BERTScore of around 0.79, which suggests that their outputs are quite similar overall. This similarity is likely due to the fact that both models are responding to the same assignment and criteria, so they share much of the same content. When looking at a few samples, the content of the feedback is often different. These details are important when it comes to the feedback. Therefore, we can conclude that the feedback from the two models is not truly the same.

Even though using language models to give automatic feedback can be more convenient, there are some ethical implications. LLMs can introduce biases which could, for example, overlook marginalized groups. Furthermore, students should be made aware of the fact that LLMs are used for grading and their possible biases [34]. Therefore, the feedback should still be controlled and regulated by human instructors.

For future work, generation of the feedback could be tested with multiple temperature settings. It was left on the default setting (0.8) to ensure that the LLM uses some creativity for the output. Additionally, a multi-agent debate system could be used for generating the feedback, where multiple LLMs first discuss with each other before giving a final answer. Furthermore, few-shot prompting could be used where multiple examples of sufficient and insufficient feedback are given in the prompt.

Beyond improving the generation process itself, future development should also consider how instructors will interact with the system. For that, a user interface is required for the instructors. The interface would call a function from Python to generate the feedback using summarized chapters as context and providing the relevant answer model. Then the interface would let the instructor select which parts in the feedback they want to include.

7.3 RQ3: Evaluating generated feedback

When generating feedback per criterion, it is unclear why the model occasionally includes additional, unintended criteria. As this also happens for generating the feedback all at once, this suggests limitations in the model's ability to always follow instructions.

Figure 2 shows that supplying the summarized chapters receives an overall higher score compared to using the retriever or no context. This confirms both the evaluation received from the instructor about the feedback and our assumption that the retriever does not always retrieve the most relevant content. The retriever probably

does not retrieve the right documents when a paragraph number is mentioned, because the embedding model has an equal focus on the whole sentence and not only on the part where the paragraph number is mentioned. However, the "Correctness" and "Relevance (for the context)" metrics are unexpectedly lower when using summarized chapters.

Figure 3 indicates that including the answer model leads to improvements in most standard metrics. The answer model is not passed as context, therefore the scores for the context-based metrics stay roughly the same.

In conclusion, the use of summarized chapters and the answer model resulted in higher evaluation scores for feedback evaluated by Qwen. Therefore, it aligns with the evaluation from one of the instructors, which means that smaller LLMs could also be used for evaluating output with LLM-as-judge frameworks. For further study, the system should be used in its actual environment so that instructors can assess the quality of the generated feedback in practice.

8 CONCLUSION

In this paper, we analyzed the performance of Gemma-3-27B-it in generating feedback in the Dutch language. The model demonstrated solid factual knowledge and reading comprehension capabilities; however, it struggled with accurately interpreting specific semantic nuances within a sentence.

We employed Gemma-3-27B-it to produce feedback according to a predefined layout, which it consistently followed. Various prompt engineering strategies were explored, including adding context from the accompanying textbook and, for one assignment, inserting the corresponding answer model. For the system to be used in practice, a user interface needs to be built. The interface should allow the instructors to choose which parts of the feedback to include.

Based on input from one of the instructors involved in assessing the assignments, we found out that adding context from the textbook improved the quality of the generated feedback. Moreover, including the answer model further improved the quality. The feedback was also assessed using the G-Eval framework with Qwen3-32B. Interestingly, the model assigned higher scores to the feedback preferred by the instructor. These results suggest that smaller language models, while generally not as good as systems such as ChatGPT, can nonetheless be leveraged for the evaluation of generated content. Further manual evaluation is necessary to evaluate the system in practice.

ACKNOWLEDGMENTS

I would like to thank N. Bouali for his assistance with the project and for hosting the LLMs on the cluster of the university, as well as Dorien, one of the instructors assessing the assignments, for her valuable feedback on the generated output. In addition, ChatGPT was used to assist with editing and refining the text of this paper.

REFERENCES

- [1] [n. d.] Alexandrainst/m_hellaswag · Datasets at Hugging Face. Retrieved May 2, 2025 from https://huggingface.co/datasets/alexandrainst/m_hellaswag.
- [2] [n. d.] API Reference - OpenAI API. Retrieved June 25, 2025 from <https://platform.openai.com>.

- [3] Lucas Bandarkar et al. 2024. The Belebele Benchmark: a Parallel Reading Comprehension Dataset in 122 Language Variants. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. ACL 2024. Lun-Wei Ku, Andre Martins, and Vivek Srikumar, (Eds.) Association for Computational Linguistics, Bangkok, Thailand, (Aug. 2024), 749–775. doi: 10.18653/v1/2024.acl-long.44.
- [4] Sourav Banerjee, Ayushi Agarwal, and Eishkaran Singh. 2024. The Vulnerability of Language Model Benchmarks: Do They Accurately Reflect True LLM Performance? (Dec. 2, 2024). arXiv: 2412.03597 [cs]. Retrieved Apr. 29, 2025 from <http://arxiv.org/abs/2412.03597>. Pre-published.
- [5] Tom B. Brown et al. 2020. Language Models are Few-Shot Learners. (July 22, 2020). arXiv: 2005.14165 [cs]. Retrieved May 29, 2025 from <http://arxiv.org/abs/2005.14165>. Pre-published.
- [6] [SW] Benjamin van der Burgh, Benjaminvdb/DBRD Mar. 13, 2025. URL: <https://github.com/benjaminvdb/DBRD> Retrieved May 1, 2025 from.
- [7] Yixin Cao et al. 2025. Toward Generalizable Evaluation in the LLM Era: A Survey Beyond Benchmarks. Version 1. (Apr. 26, 2025). arXiv: 2504.18838 [cs]. Retrieved May 14, 2025 from <http://arxiv.org/abs/2504.18838>. Pre-published.
- [8] Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. 2023. ChatEval: Towards Better LLM-based Evaluators through Multi-Agent Debate. (Aug. 14, 2023). arXiv: 2308.07201 [cs]. Retrieved May 15, 2025 from <http://arxiv.org/abs/2308.07201>. Pre-published.
- [9] François Chollet. 2019. On the Measure of Intelligence. (Nov. 25, 2019). arXiv: 1911.01547 [cs]. Retrieved June 25, 2025 from <http://arxiv.org/abs/1911.01547>. Pre-published.
- [10] Wei Dai, Yi-Shan Tsai, Jionghao Lin, Ahmad Aldino, Hua Jin, Tongguang Li, Dragan Gašević, and Guanliang Chen. 2024. Assessing the proficiency of large language models in automatic feedback generation: An evaluation study. *Computers and Education: Artificial Intelligence*, 7, (Dec. 2024), 100299. doi: 10.1016/j.caeai.2024.100299.
- [11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. (May 24, 2019). arXiv: 1810.04805 [cs]. Retrieved June 28, 2025 from <http://arxiv.org/abs/1810.04805>. Pre-published.
- [12] [n. d.] Dutch - EuroEval - Datasets. Retrieved May 7, 2025 from <https://euroeval.com/datasets/dutch/>.
- [13] Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. 2023. GPTScore: Evaluate as You Desire. (Feb. 13, 2023). arXiv: 2302.04166 [cs]. Retrieved May 16, 2025 from <http://arxiv.org/abs/2302.04166>. Pre-published.
- [14] Louie Giray. 2023. Prompt Engineering with ChatGPT: A Guide for Academic Writers. *Annals of Biomedical Engineering*, 51, 12, (Dec. 1, 2023), 2629–2633. doi: 10.1007/s10439-023-03272-4.
- [15] Gregory Grefenstette. 1999. Tokenization. In *Syntactic Wordclass Tagging*. Hans van Halteren, (Ed.) Springer Netherlands, Dordrecht, 117–133. ISBN: 978-94-015-9273-4. doi: 10.1007/978-94-015-9273-4_9.
- [16] 2024. GroNLP/dutch-cola · Datasets at Hugging Face. (Apr. 25, 2024). Retrieved May 1, 2025 from <https://huggingface.co/datasets/GroNLP/dutch-cola>.
- [17] Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. 2022. ToxiGen: A Large-Scale Machine-Generated Dataset for Adversarial and Implicit Hate Speech Detection. (July 14, 2022). arXiv: 2203.09509 [cs]. Retrieved June 25, 2025 from <http://arxiv.org/abs/2203.09509>. Pre-published.
- [18] John Hattie and Helen Timperley. [n. d.] The Power of Feedback. Retrieved Apr. 23, 2025 from <https://journals.sagepub.com/doi/epdf/10.3102/003465430298487>.
- [19] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring Massive Multitask Language Understanding.
- [20] Lei Huang et al. 2025. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Trans. Inf. Syst.*, 43, 2, (Jan. 24, 2025), 42:1–42:55. doi: 10.1145/3703155.
- [21] [SW] Jeffrey Ip and Kritin Vongthongsri, Deepeval version 3.0.8, June 2025. URL: <https://github.com/confident-ai/deepeval>.
- [22] Shreyas Joshi, Muhammad Shah Nawaz Khan, Aditya Dafe, Kavita Singh, Vedant Zope, and Tanish Jhamtani. 2024. Fine Tuning LLMs for Low Resource Languages. In *2024 5th International Conference on Image Processing and Capsule Networks (ICIPCN)*. 2024 5th International Conference on Image Processing and Capsule Networks (ICIPCN). (July 2024), 511–519. doi: 10.1109/ICIPCN63822.2024.00090.
- [23] Enkelejda Kasneci et al. 2023. ChatGPT for good? On opportunities and challenges of large language models for education. *Learning and Individual Differences*, 103, (Apr. 1, 2023), 102274. doi: 10.1016/j.lindif.2023.102274.
- [24] Imed Keraghel, Stanislas Morbieu, and Mohamed Nadif. 2024. Beyond Words: A Comparative Analysis of LLM Embeddings for Effective Clustering. In *Advances in Intelligent Data Analysis XXII*. Ioanna Miliou, Nico Piatkowski, and Panagiotis Papapetrou, (Eds.) Springer Nature Switzerland, Cham, 205–216. ISBN: 978-3-031-58547-0. doi: 10.1007/978-3-031-58547-0_17.
- [25] Ben Koska and Mojmir Horváth. 2024. Towards Multi-Modal Mastery: A 4.5B Parameter Truly Multi-Modal Small Language Model. In *2024 2nd International Conference on Foundation and Large Language Models (FLLM)*. 2024 2nd International Conference on Foundation and Large Language Models (FLLM). (Nov. 2024), 587–592. doi: 10.1109/FLLM63129.2024.10852430.
- [26] Faisal Ladhak, Esin Durmus, Claire Cardie, and Kathleen McKeown. 2020. WikiLingua: A New Benchmark Dataset for Cross-Lingual Abstractive Summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*. Findings of the Association for Computational Linguistics: EMNLP 2020. Association for Computational Linguistics, Online. doi: 10.18653/v1/2020.findings-emnlp.360.
- [27] Viet Dac Lai, Chien Van Nguyen, Nghia Trung Ngo, Thuat Nguyen, Franck Dernoncourt, Ryan A. Rossi, and Thien Huu Nguyen. 2023. Okapi: Instruction-tuned Large Language Models in Multiple Languages with Reinforcement Learning from Human Feedback. (Aug. 2, 2023). arXiv: 2307.16039 [cs]. Retrieved May 7, 2025 from <http://arxiv.org/abs/2307.16039>. Pre-published.
- [28] [SW], Language-Tool-Python: Checks Grammar Using LanguageTool. Version 2.9.3.
- [29] Patrick Lewis et al. 2021. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. (Apr. 12, 2021). arXiv: 2005.11401 [cs]. Retrieved May 21, 2025 from <http://arxiv.org/abs/2005.11401>. Pre-published.
- [30] Jing Li, Aixin Sun, Jianglei Han, and Chenliang Li. 2022. A Survey on Deep Learning for Named Entity Recognition. *IEEE Transactions on Knowledge and Data Engineering*, 34, 1, (Jan. 2022), 50–70. doi: 10.1109/TKDE.2020.2981314.
- [31] Ruosen Li, Teerth Patel, and Xinya Du. 2024. PRD: Peer Rank and Discussion Improve Large Language Model based Evaluations. (Dec. 31, 2024). arXiv: 2307.02762 [cs]. Retrieved May 14, 2025 from <http://arxiv.org/abs/2307.02762>. Pre-published.
- [32] Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-Eval: NLG Evaluation using GPT-4 with Better Human Alignment. (May 23, 2023). arXiv: 2303.16634 [cs]. Retrieved May 16, 2025 from <http://arxiv.org/abs/2303.16634>. Pre-published.
- [33] [n. d.] Llm4teach-freetext-server/freetext/feedback_providers/OpenAIFeedbackProvider.py at ce892827e6d57aa2ef8c302366a924cc3c5473cc · KordingLab/llm4teach-freetext-server. GitHub. Retrieved Apr. 23, 2025 from https://github.com/KordingLab/llm4teach-freetext-server/blob/ce892827e6d57aa2ef8c302366a924cc3c5473cc/freetext/feedback_providers/OpenAIFeedbackProvider.py.
- [34] Jordan K. Matelsky, Felipe Parodi, Tony Liu, Richard D. Lange, and Konrad P. Kording. 2023. A large language model-assisted education tool to provide feedback on open-ended responses. (July 25, 2023). arXiv: 2308.02439 [cs]. Retrieved Apr. 23, 2025 from <http://arxiv.org/abs/2308.02439>. Pre-published.
- [35] Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2025. Large Language Models: A Survey. (Mar. 23, 2025). arXiv: 2402.06196 [cs]. Retrieved June 25, 2025 from <http://arxiv.org/abs/2402.06196>. Pre-published.
- [36] Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2023. MTEB: Massive Text Embedding Benchmark. (Mar. 19, 2023). arXiv: 2210.07316 [cs]. Retrieved June 16, 2025 from <http://arxiv.org/abs/2210.07316>. Pre-published.
- [37] [n. d.] NetherlandsForensicInstitute/squad-nl-v2.0 · Datasets at Hugging Face. Retrieved May 1, 2025 from <https://huggingface.co/datasets/NetherlandsForensicInstitute/squad-nl-v2.0>.
- [38] [SW] Dan Saatrup Nielsen, ScandEval: A Benchmark for Scandinavian Natural Language Processing May 2023. URL: <https://github.com/EuroEval/EuroEval> Retrieved May 17, 2025 from.
- [39] Dan Saatrup Nielsen. [n. d.] ScandEval: A Benchmark for Scandinavian Natural Language Processing.
- [40] [n. d.] Nlp-uoregon/mlmm-evaluation: Multilingual Large Language Models Evaluation Benchmark. Retrieved May 7, 2025 from <https://github.com/nlp-uoregon/mlmm-evaluation/tree/main#>.
- [41] Saurav Pawar, S. M. Towhidul Islam Tonmoy, S. M. Mehedi Zaman, Vinija Jain, Aman Chadha, and Amitava Das. 2024. The What, Why, and How of Context Length Extension Techniques in Large Language Models – A Detailed Survey. (Jan. 15, 2024). arXiv: 2401.07872 [cs]. Retrieved June 25, 2025 from <http://arxiv.org/abs/2401.07872>. Pre-published.
- [42] Marten Postma, Emiel van Miltenburg, Roxane Segers, Anneleen Schoen, and Piek Vossen. 2016. Open Dutch WordNet. In *Proceedings of the 8th Global WordNet Conference (GWC)*. GWC 2016. Christiane Fellbaum, Piek Vossen, Verginica Barbu Mititelu, and Corina Forascu, (Eds.) Global Wordnet Association, Bucharest, Romania, 302–310. Retrieved May 7, 2025 from <https://aclanology.org/2016.gwc-1.43/>.
- [43] Alessandro Raganato, Tommaso Pasini, Jose Camacho-Collados, and Mohammad Taher Pilehvar. 2020. XL-WiC: A Multilingual Benchmark for Evaluating

- Semantic Contextualization. (Oct. 13, 2020). arXiv: 2010.06478 [cs]. Retrieved May 1, 2025 from <http://arxiv.org/abs/2010.06478>. Pre-published.
- [44] Shivalika Singh et al. 2025. Global MMLU: Understanding and Addressing Cultural and Linguistic Biases in Multilingual Evaluation. (Feb. 19, 2025). arXiv: 2412.03304 [cs]. Retrieved Apr. 24, 2025 from <http://arxiv.org/abs/2412.03304>. Pre-published.
- [45] [n. d.] Text generation and prompting - OpenAI API. Retrieved June 25, 2025 from <https://platform.openai.com>.
- [46] Erik F. Tjong Kim Sang. 2002. Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*. CoNLL 2002. Retrieved June 14, 2025 from <https://aclanthology.org/W02-2024/>.
- [47] [SW], UniversalDependencies/UD_Dutch-Alpino Jan. 8, 2025. Universal Dependencies. URL: https://github.com/UniversalDependencies/UD_Dutch-Alpino Retrieved May 7, 2025 from.
- [48] Bram Vanroy. 2024. Fietje: An open, efficient LLM for Dutch. (Dec. 19, 2024). arXiv: 2412.15450 [cs]. Retrieved Apr. 24, 2025 from <http://arxiv.org/abs/2412.15450>. Pre-published.
- [49] Bram Vanroy. 2024. GEITje 7B Ultra: A Conversational Model for Dutch. (Dec. 5, 2024). arXiv: 2412.04092 [cs]. Retrieved Apr. 29, 2025 from <http://arxiv.org/abs/2412.04092>. Pre-published.
- [50] Weiqi Wu, Hongqiu Wu, Lai Jiang, Xingyuan Liu, Jiale Hong, Hai Zhao, and Min Zhang. 2024. From Role-Play to Drama-Interaction: An LLM Solution. (May 23, 2024). arXiv: 2405.14231 [cs]. Retrieved June 25, 2025 from <http://arxiv.org/abs/2405.14231>. Pre-published.
- [51] Ministerie van Algemene Zaken. [n. d.] Wat verdien ik als leraar in het voortgezet onderwijs? - rijksoverheid.nl. Retrieved Apr. 23, 2025 from <https://www.rijksoverheid.nl/onderwerpen/werken-in-het-onderwijs/vraag-en-antwoord/wat-verdien-ik-als-leraar-in-het-voortgezet-onderwijs>.
- [52] Krystian Zawistowski. 2024. Unused information in token probability distribution of generative LLM: improving LLM reading comprehension through calculation of expected values. (Sept. 26, 2024). arXiv: 2406.10267 [cs]. Retrieved June 25, 2025 from <http://arxiv.org/abs/2406.10267>. Pre-published.
- [53] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a Machine Really Finish Your Sentence? (May 19, 2019). arXiv: 1905.07830 [cs]. Retrieved May 2, 2025 from <http://arxiv.org/abs/1905.07830>. Pre-published.
- [54] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating Text Generation with BERT. (Feb. 24, 2020). arXiv: 1904.09675 [cs]. Retrieved Apr. 28, 2025 from <http://arxiv.org/abs/1904.09675>. Pre-published.
- [55] Ruochen Zhao, Wenxuan Zhang, Yew Ken Chia, Weiwen Xu, Deli Zhao, and Lidong Bing. 2024. Auto-Arena: Automating LLM Evaluations with Agent Peer Battles and Committee Discussions. (Oct. 7, 2024). arXiv: 2405.20267 [cs]. Retrieved May 19, 2025 from <http://arxiv.org/abs/2405.20267>. Pre-published.
- [56] Lianmin Zheng et al. 2023. Judging LLM-as-a-judge with MT-bench and Chatbot Arena. In *Proceedings of the 37th International Conference on Neural Information Processing Systems (NIPS '23)*. Curran Associates Inc., Red Hook, NY, USA, (Dec. 10, 2023), 46595–46623.
- [57] Kun Zhou, Yutao Zhu, Zhipeng Chen, Wentong Chen, Wayne Xin Zhao, Xu Chen, Yankai Lin, Ji-Rong Wen, and Jiawei Han. 2023. Don't Make Your LLM an Evaluation Benchmark Cheater. (Nov. 3, 2023). arXiv: 2311.01964 [cs]. Retrieved June 25, 2025 from <http://arxiv.org/abs/2311.01964>. Pre-published.
- [58] Ruiyang Zhou, Lu Chen, and Kai Yu. 2024. Is LLM a Reliable Reviewer? A Comprehensive Evaluation of LLM on Automatic Paper Reviewing Tasks. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*. LREC-COLING 2024. Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, (Eds.) ELRA and ICCL, Torino, Italia, (May 2024), 9340–9351. Retrieved June 27, 2025 from <https://aclanthology.org/2024.lrec-main.816/>.

A PROMPTS

A.1 Breakfast prompt

Schrijf een gedetailleerde beschrijving van een typisch Nederlands ontbijt, inclusief de gebruikte ingrediënten, de culturele betekenis ervan en hoe dit ontbijt verschilt van andere ontbijttradities in Europa. Gebruik daarnaast verschillende zinsconstructies en toon een geavanceerd begrip van grammatica en stijl. Zorg ervoor dat je zowel informele als formele elementen in je antwoord verwerkt en de tekst goed organiseert.

A.2 Benchmarking

A.2.1 Prompt ARC.

```
{instruction}
Antwoordopties:
A. {option_a}; B. {option_b}; C. {option_c}; D.
{option_d}
Antwoord met A, B, C of D. Geef GEEN uitleg."
```

A.2.2 Prompt Belebele.

```
{text_passage}
{question}
1. {mc_answer1}; 2. {mc_answer2}; 3. {mc_answer3};
4. {mc_answer4}
Antwoord met 1, 2, 3, of 4. Geef GEEN uitleg.
```

A.2.3 Prompt CoLA.

Is de volgende tekst grammaticaal (correct Nederlands) of ongrammaticaal (onjuist Nederlands)?
Tekst: {{ Sentence }} Antwoord met 'grammaticaal' of 'ongrammaticaal'. Geef GEEN uitleg.

A.2.4 Prompt Global MMLU.

```
{question}
Antwoordopties:
A: {option_a} B: {option_b} C: {option_c} D:
{option_d}",
Geef je antwoord als een enkele letter: A, B, C of D
Geef GEEN uitleg over hoe je op het antwoord bent
gekomen
```

A.2.5 Prompt XL-WiC-NL.

Is de betekenis van {target_word} in de volgende zinnen identiek of verschillend?
Zin 1: {example_1}
Zin 2: {example_2}
Antwoord met 'identiek' of 'verschillend'. Eindig je zin NIET met een punt. Geef GEEN uitleg.

A.3 Generating feedback

A.3.1 Prompt generating feedback per criterion.

Neem de rol van leraar aan, jouw taak is om feedback te geven op de opdracht van een student volgens gegeven criteria en daarbij ook aan te geven of de criteria voldaan is of niet.

```
--- BEGIN ANTWOORD STUDENT ---
```

```
{assignment}
```

```
--- EINDE ANTWOORD STUDENT ---
```

Dit waren je vorige beoordelingen:

```
{previous_response}
```

Geef voor de volgende criterium aan of deze gehaald is (voldoende of niet voldoende) en geef één positieve opmerking en geef drie verbeterpunten. Geef deze opmerkingen en verbeterpunten met bullet points. Geef geen andere informatie. De antwoorden van de student moeten gebaseerd zijn op de relevante context. Probeer ook zoveel mogelijk feedback te geven gebaseerd op het tekstboek in de relevante context. Zorg ervoor dat je opmerkingen niet te specifiek zijn op het antwoord van het student, maar algemener. Verwoord de positieve opmerkingen en verbeterpunten richting de student met gebruik van het voornaamwoord 'je/jouw'.

De volgende context uit het tekstboek kan relevant zijn voor je antwoord:

```
--- BEGIN EXTRA CONTEXT ---
```

```
{extra_context}
```

```
--- EINDE EXTRA CONTEXT ---
```

Volg de volgende layout voor je antwoord:

```
#### {criterion}
```

Criterium voldaan: voldaan of niet voldaan

Positieve opmerkingen:

- opmerking 1

Verbeterpunten:

- verbeterpunt 1

- verbeterpunt 2

- verbeterpunt 3

A.3.2 Prompt generating feedback all at once.

Neem de rol van leraar aan, jouw taak is om feedback te geven op de opdracht van een student volgens de volgende criteria. Geef daarbij ook aan of het criterium voldaan is of niet. Zorg er voor dat je feedback geeft voor ELK criterium. Voor de volgende opdracht zijn er de volgende criteria:

```
{criteria_list_bullet_points}
```

Hieronder volgt het antwoord van de student.

```
--- START ANTWOORD STUDENT ---
```

```
{self.assignment_answer}
```

```
--- EINDE ANTWOORD STUDENT ---
```

Geef voor elke criteria aan of deze gehaald is (voldoende of niet voldoende) en geef één positieve opmerking en geef drie verbeterpunten. Geef deze opmerkingen en verbeterpunten met bullet points. Geef geen andere informatie. De antwoorden van de student moeten gebaseerd zijn op het tekstboek in de relevante context. Probeer ook zoveel mogelijk feedback te geven gebaseerd op de relevante context. Zorg ervoor dat je opmerkingen niet te specifiek zijn op het antwoord van het student, maar meer algemener. Verwoord de positieve opmerkingen en verbeterpunten richting de student met gebruik van het voornaamwoord 'je/jouw'.

De volgende context uit het tekstboek kan relevant zijn voor je antwoord.

{relevant_context}

Volg de volgende layout voor je antwoord:

[criterium]

Criterium voldaan: voldaan of niet voldaan

Positieve opmerkingen:

- opmerking 1

Verbeterpunten:

- verbeterpunt 1

- verbeterpunt 2

- verbeterpunt 3

A.3.3 Prompt answer model.

Gebruik het volgende antwoordmodel om feedback te geven, en om te bepalen of een criterium gehaald is of niet. Controleer of het antwoord van de student in het antwoordmodel staat. Als het antwoord van de student er niet in staat moet je extra kritisch zijn op het antwoord, en als het antwoord niet klopt, gebruik dan de antwoorden uit het antwoordmodel voor je feedback.

--- START ANTWOORDMODEL ---

{answer_model}

--- EINDE ANTWOORDMODEL ---

A.4 Evaluating feedback

A.4.1 G-Eval Template Prompt.

You are not evaluating the student's work. Instead, you are evaluating the quality of the feedback that was given *on* the student's work.

Given the evaluation steps, return a JSON with two keys: 1) a 'score' key ranging from 0 to 10, with 10 being that it follows the criteria outlined in the steps and 0 being that it does not, and 2) a 'reason' key, a reason for the given score, but DO NOT QUOTE THE SCORE in your reason. Please mention specific information from {test_parameters} in your reason, but be very concise with it!

Evaluation Steps:

{evaluation_steps}

```
{input}
{output}
{optional:expected_output}
{optional:retrieval_context}
```

**

You are scoring the feedback itself, not the student. Use the Expected Output to judge whether the Actual Output made the correct evaluation. Refer to both in your explanation.

IMPORTANT: Please make sure to only return in JSON format, with the "score" and "reason" key. No words or explanation is needed.

Example JSON:

```
{
  "score": {score_range[0]},
  "reason": "The text does not follow the
  evaluation steps provided."
}
```

**

JSON:

B RUBRIC EXAMPLE

Opdracht 3: Kennis van verschillende toetsprocessen

Table 3. Leerdoelen en succescriteria bij competentie (3) Kennis van verschillende toetsprocessen.

Leerdoelen en succescriteria	nv/v	Opmerkingen
Ik kan het onderscheid tussen summatief en formatief gebruik van toetsen beschrijven.		
• Ik beschrijf tenminste drie verschillen en/of overeenkomsten tussen de modellen voor formatief evalueren.		
• Ik beschrijf tenminste drie verschillen en/of overeenkomsten tussen de modellen voor summatief toetsen.		
• Ik beschrijf ten minste twee overeenkomsten en twee verschillen tussen formatief evalueren en summatief toetsen.		
Ik kan beschrijven hoe men de balans tussen summatief en formatief gebruik van toetsen kan bewaken.		
• Ik beschrijf de huidige balans tussen formatief evalueren en summatief toetsen binnen de eigen school.		
• Ik illustreer de huidige balans met tenminste twee concrete praktijkvoorbeelden.		
• Ik beschrijf en onderbouw hoe ik idealiter de balans tussen formatief evalueren en summatief toetsen zou willen zien in mijn vakgebied.		
Ik kan collega's adviseren op het gebied van formatief evalueren en summatief toetsen.		
• Ik formuleer op basis van het verschil tussen de huidige en ideale situatie in school een advies voor collega's met betrekking tot de inzet van formatief evalueren en summatief toetsen.		
• Ik concretiseer het advies door minimaal twee concrete veranderingen te benoemen die ik als eerste zou doorvoeren om deze balans te realiseren.		
• Ik beschrijf hoe ik als change leader de collega's kan ondersteunen om deze veranderingen succesvol te implementeren.		

C METRICS

```
[MetricsCase(
    metric_name="Correctness", # Compares the actual
    output to the expected output
    evaluation_steps=[
        "Compare the actual output to the input and
        determine whether the feedback accurately
        reflects the content, intent, or issues present
        in the input. The feedback should not contain
        factual errors or misinterpretations.",
        "Check whether the pass and fail of the criteria
        in the expected output is the same as in the
        actual output. It is OK if the format from the
        actual output and expected output are
        different.",
        "Compare the comments from the expected output
        to the actual output and check if the same
        content from expected output is also in the
        actual output.",
        "The feedback in the actual output should
        clearly identify strengths or areas for
        improvement based on what's present in the
        input."
    ],
    params=[
        TestParams.INPUT,
        TestParams.GIVEN_OUTPUT,
        TestParams.EXPECTED_OUTPUT
    ]
),
MetricsCase(
    metric_name="Relevance", # Is the generated
    feedback relevant to the input and the criteria?
    evaluation_steps=[
        "Check if the criteria in the input is answered
        in the output.",
        "Check if the feedback directly addresses the
        points outlined in the criteria and does not
        include generic or unrelated comments.",
        "Verify that the feedback focuses specifically
        on the content and qualities of the input,
        rather than straying into irrelevant
        observations.",
        "Assess whether the feedback prioritizes the
        most important aspects of the input according to
        the criteria, rather than focusing on minor or
        tangential issues."
    ],
    params=[
        TestParams.INPUT,
        TestParams.GIVEN_OUTPUT
    ]
),
MetricsCase(
    metric_name="Coherence", # Is the feedback
    coherent and easy to understand?
```

```
evaluation_steps=[
    "Evaluate whether the response uses clear,
    direct language, and does not contain any
    grammar or spelling mistakes.",
    "Assess whether the feedback in the actual
    output is clear and actionable. It should be
    easy to understand and provide helpful
    suggestions.",
    "Evaluate whether the feedback is
    constructive, meaning it not only points out
    issues or areas for improvement but also
    offers specific, practical suggestions for how
    to address them.",
    "Identify any vague or confusing parts that
    reduce understanding."
],
    params=[
        TestParams.INPUT,
        TestParams.GIVEN_OUTPUT
    ]
),
MetricsCase(
    metric_name="Tonality", # Does the output have an
    appropriate tone for the task?
    evaluation_steps=[
        "Determine whether the response maintains a
        respectful and helpful tone throughout.",
        "Evaluate if the feedback offers constructive
        criticism without sounding dismissive or
        harsh.",
        "Evaluate if the feedback is not overly
        positive, or praising the student too much",
        "Check that the language is professional, avoids
        slang, and is appropriately formal for
        educational feedback."
    ],
    params=[
        TestParams.INPUT,
        TestParams.GIVEN_OUTPUT
    ]
),
MetricsCase(
    metric_name="Relevance (for the context)", # Does
    the answer from the retriever directly address the
    question?
    evaluation_steps=[
        "Determine whether the retrieved context
        contains information directly related to the
        user's input question.",
        "Check if the retrieved context provides
        sufficient detail to answer the question
        accurately and completely.",
        "Assess whether the information in the context
        aligns with the expected domain or topic of the
        question."
    ],
    params=[
        TestParams.INPUT,
        TestParams.GIVEN_OUTPUT
    ]
),
MetricsCase(
    metric_name="Contextual Relevance", # Does the
    feedback address the specific context of the
    question?
```

```

        "Evaluate whether any irrelevant or off-topic
        content in the context could distract or mislead
        the answer generation.",
    ],
    params=[
        TestParams.INPUT,
        TestParams.RETRIEVAL_CONTEXT
    ]
),
MetricsCase(
    metric_name="Faithfulness (for the context)", #
    Are the claims from the retrieved context used in
    the output?
    evaluation_steps=[
        "For each key claim or piece of information in
        the answer, check whether it does not contrast
        the information from the retrieval context.",
        "If the answer paraphrases or summarizes the
        retrieved content, confirm that it remains
        faithful to the meaning and does not distort or
        overextend the source material.",
        "Assess whether the answer omits any critical
        information from the retrieval context that
        would significantly alter the meaning or
        completeness of the response if left out."
    ],
    params=[
        TestParams.INPUT,
        TestParams.RETRIEVAL_CONTEXT,
        TestParams.GIVEN_OUTPUT
    ]
),
MetricsCase(
    metric_name="Correct book chapter (for the
    context)",
    evaluation_steps=[
        "Check whether the input explicitly mentions a
        paragraph number, section title, or specific
        chapter (e.g., 'see Chapter 3' or 'according to
        Paragraph 4').",
        "Verify that the retrieval context includes
        content specifically from the paragraph or
        chapter referenced in the input. It must
        directly correspond to the cited location.",
        "Confirm that the retrieved content is not only
        from the correct location but also meaningfully
        addresses the query or topic mentioned in the
        input.",
        "If the retrieval context comes from a different
        section than the one referenced in the input—or
        omits the correct section entirely—flag it as a
        misalignment.",
    ],
    params=[
        TestParams.INPUT,
        TestParams.RETRIEVAL_CONTEXT
    ]
)
]
)]

```

D DUTCH OUTPUTS EXAMPLES

D.1 Qwen3-32B

Een typisch Nederlands ontbijt is een symbool van eenvoud, praktischheid en een diepe band met de landbouwtraditie. Het maakt gebruik van lokaal geproduceerde ingrediënten en reflecteert de Nederlandse voorkeur voor gebalanceerde, niet te zware maaltijden. Tijdens het ontbijt, dat vaak op een zondag of bij speciale gelegenheden wordt genoten, vinden de Nederlanders troost in de combinatie van warme, koele en zachte smaken, terwijl de maaltijd tegelijkertijd ruimte biedt voor persoonlijke voorkeuren.

****Ingrediënten en bereiding****

Het hart van een Nederlands ontbijt is de *ontbijtkoek*, een donkerbruine roggebrood die vaak met boter of een laagvet melkproduct zoals *zilverbunt* wordt gesmeerd. Deze zachte, zwoele boter, gemaakt van melk en suiker, is een onmisbaar element. Daarnaast worden kaas en gekookte eieren (*gekookte eieren*), die in de koele lucht worden opgeborgen, vaak gepresenteerd. De kaas, zoals Gouda of Edam, symboliseert de rijke Nederlandse kaasindustrie, terwijl de gekookte eieren, met hun goudkleurige, roomachtige binnenkant, een contrast vormen met de zoute smaken. Voor dranken wordt koffie of thee verkozen, vaak gezeten bij een glas melk of sap. Soms verschijnt een appel of een broodje *ham en kaas* als extraatje, waardoor het ontbijt iets informeler wordt.

****Culturele betekenis en traditie****

Deze maaltijd is meer dan een louter culinair genot; het is een reflectie van de Nederlandse levensstijl. Het concept van *gezelligheid* – een gevoel van warme, gezamenlijke verbondenheid – hangt sterk samen met het ontbijt. De tafel wordt gezet met zorgvuldig uitgekozen stukken, en het delen van een kop koffie en een plakje brood met familie of vrienden is een moment van rust in een snelle wereld. De nadruk op lokaal geproduceerde producten, zoals roggebrood en kaas, spreekt bovendien de Nederlandse trots op hun agrarische geschiedenis aan. Historisch gezien was het ontbijt een manier om voedsel te gebruiken dat overbleef van de vorige dag, een praktijk die nog steeds invloed heeft op de keuken.

****Verschillen met andere Europese tradities****

In tegenstelling tot het Britse *Full English breakfast*, dat vaak bestaat uit worst, spek, eieren en gehaktballetjes, is het Nederlands ontbijt subtieler en lichter. Terwijl de Fransen hun ochtend beginnen met een croissant en koffie, waarbij het accent op de smaak van het brood ligt, kiest Nederland voor een rustiger evenwicht. Duitsland, met zijn zware broodsoorten en diverse vleeswaren, benadrukt de zwaarte van de maaltijd, terwijl de Nederlandse aanpak meer gericht is op het combineren van eenvoud en variatie. Ook in Scandinavië, waar granola en zure melk centraal staan, is de nadruk op gezondheid duidelijker dan in Nederland. De Nederlandse aanpak is dus uniek in haar vermogen om zowel comfort als kwaliteit te bieden zonder overdaad.

****Conclusie****

Samenvattend is het Nederlands ontbijt een geografisch en cultureel product dat zowel de geschiedenis van het land als de moderne levensstijl belichaamt. Het biedt een rustige start aan de dag, met ingrediënten die vertrouwd en vertrouwd zijn, en een aanpak die zich onderscheidt door haar evenwicht en eenvoud. Of je nu een plakje *ontbijtkoek* met zilverbunt of een eenvoudig broodje ham in handen hebt, het ontbijt blijft een moment van genieten – een kleine, maar krachtige ode aan de Nederlandse *gezelligheid*.

D.2 Gemma-3-27B-it

De Nederlandse Ochtendmaaltijd: Een Reflectie van Pragmatisme en Gemak

Het Nederlandse ontbijt, vaak aangeduid als de ochtendmaaltijd, is een fascinerend fenomeen dat in schril contrast staat met de uitgebreidere, vaak ceremoniële ontbijttradities die elders in Europa gehanteerd worden. Hoewel er regionale variaties bestaan en individuele voorkeuren uiteraard doorslaggevend zijn, is er een opvallende consistentie in de basiselementen die deze maaltijd definiëren, en de culturele waarden die eraan verbonden zijn.

****Ingrediënten en Bereiding: Een Simpele Affaire****

De kern van een typisch Nederlands ontbijt bestaat uit brood, belegd met een selectie van eenvoudige, maar essentiële ingrediënten. Volkorenbrood, bruin brood of wit brood – afhankelijk van de persoonlijke voorkeur – vormt de basis. Dit wordt vervolgens royaal besmeerd met boter, margarine, of, steeds vaker, plantaardige spreads. De belegkeuze is divers, maar enkele opties domineren:

* ****Vleeswaren:**** Leverworst, cervelaatworst, ham (vaak gekookte ham), en filé américain (fijngehakt rauw vlees) zijn populaire keuzes.

- * ****Kaas:**** Jonge kaas, oudere kaas, of Goudse kaas behoren tot de standaardopties.
- * ****Zoet beleg:**** Hagelslag (chocoladeschaafsel), vlokken (chocoladeschaafsel met melkchocolade), jam (aardbeien-, frambozen-, of abrikozenjam zijn gebruikelijk), en pindakaas (een ware nationale trots) completeren het aanbod.
- * ****Overig:**** Soms wordt er een gekookt ei toegevoegd, of een glas melk, yoghurt of sinaasappelsap.

Het bereidingsproces is opvallend ongedwongen. Er wordt zelden tijd besteed aan complexe bereidingen. Een snelle belegconstructie is voldoende om de dag te starten. Deze pragmatische benadering is kenmerkend voor de Nederlandse cultuur, die over het algemeen efficiëntie en functionaliteit waardeert.

****Culturele Betekenis: Functioneel en Familiaal****

Het Nederlandse ontbijt is historisch gezien een functionele maaltijd geweest, bedoeld om de nodige energie te leveren voor een dag hard werken. In het verleden, toen veel mensen in de landbouw of de industrie werkten, was een stevig, maar snel te nuttigen ontbijt cruciaal. Dit heeft geleid tot een voorkeur voor beleg dat langdurige energie aflevert, zoals brood met pindakaas of vleeswaren.

Tegenwoordig speelt het ontbijt ook een sociale rol. Vaak wordt de ochtendmaaltijd in familiekring genoten, waarbij de dag even wordt besproken. Het is een moment van rust en verbinding, al verloopt dit vaak in een ongecompliceerde, praktische sfeer. In tegenstelling tot bijvoorbeeld het Franse 'petit déjeuner' met zijn croissants en café au lait, of het Engelse 'full breakfast' met gebakken eieren, bacon en worst, staat het Nederlandse ontbijt bekend om zijn eenvoud en directheid.

****Vergelijking met Andere Europese Ontbijttradities****

De verschillen met andere Europese ontbijttradities zijn aanzienlijk.

- * ****Frankrijk:**** De Fransen hechten veel waarde aan een lichte, elegante ochtendmaaltijd, vaak bestaande uit viennoiseries (croissants, pain au chocolat) en koffie. Dit is een contrast met de meer substantiële Nederlandse optie.
- * ****Engeland:**** Het Engelse ontbijt is beroemd om zijn omvang en variëteit, met gebakken eieren, bacon, worst, bloedworst, tomaten, champignons en bonen. Dit is een veel meer uitgebreide maaltijd dan de gemiddelde Nederlandse ochtendmaaltijd.

- * ****Duitsland:**** Het Duitse ontbijt is divers, met een mix van brood, broodjes, vleeswaren, kaas, jam en soms ook eieren of fruit. De diversiteit is groter dan in Nederland, maar het blijft wel een relatief eenvoudige maaltijd.

- * ****Spanje/Italië:**** In deze landen is de ochtendmaaltijd vaak beperkt tot een kop koffie en een klein gebakje, aangezien de lunch de belangrijkste maaltijd van de dag is.

Het Nederlandse ontbijt onderscheidt zich dus door een combinatie van eenvoud, functionaliteit en een focus op beleg dat langdurige energie levert. Het is een weerspiegeling van de Nederlandse cultuur, die pragmatisch, efficiënt en down-to-earth is. Hoewel het misschien niet zo extravagant is als sommige andere Europese ontbijttradities, is het wel een belangrijk onderdeel van de dagelijkse routine voor miljoenen Nederlanders.

D.3 ChatGPT (gpt-4o-mini)

Een typisch Nederlands ontbijt is eenvoudig, voedzaam en gaat vaak gepaard met een rustige start van de dag. Het ontbijt in Nederland verschilt aanzienlijk van dat in veel andere Europese landen, zowel qua ingrediënten als qua eetgewoonten.

Ingrediënten en opbouw van een typisch Nederlands ontbijt

Een typisch Nederlands ontbijt bestaat meestal uit een combinatie van brood, beleg, zuivel en soms fruit. Brood is de basis van de maaltijd. Dit kan variëren van wit, bruin of volkorenbrood tot het populaire ****beschuit****, een luchtig, krokant broodje. Brood wordt vaak geserveerd met een breed scala aan beleg, zoals kaas, vleeswaren, pindakaas, hagelslag (een zoet beleg van chocolade), vruchtenjam of appelstroop. Daarnaast worden er vaak ****boerenyoghurt**** of ****kwark**** gegeten, vaak met een handvol muesli of vers fruit voor een frisse toets. ****Koffie**** en ****thee**** zijn de meest populaire dranken bij het ontbijt, maar ook ****melk**** wordt regelmatig gedronken, vooral door kinderen.

In sommige gevallen kan het ontbijt een meer ****zoete**** inslag hebben, vooral op zondagen of feestdagen. Dan worden er wel eens ****pannenkoeken**** gegeten of een broodje met suiker of kaneel. Echter, het gebruik van zoet beleg zoals hagelslag en chocoladepasta is alleraardigst in de Nederlandse ontbijtcultuur.

De culturele betekenis van het ontbijt

Het Nederlandse ontbijt heeft een onmiskenbare link met ****praktische eenvoud**** en ****efficiëntie****. Nederlanders hechten veel waarde aan een snelle en voedzame start van de dag, vooral gezien de werk- en schoolverplichtingen. Dit verklaart deels de voorkeur voor broodbeleg dat snel en gemakkelijk te gebruiken is. Bovendien is er een zekere ****gezinsgebondenheid**** in de manier waarop het ontbijt vaak samen wordt genuttigd. Het ontbijt is meestal een moment om met het gezin bij elkaar te komen, al gebeurt dit vaak zonder al te veel opsmuk of uitgebreide gesprekken.

In tegenstelling tot veel andere culturen, waar het ontbijt een meer uitgesproken ritueel is (zoals het uitgebreide Franse ontbijt met croissants of het Engelse ontbijt met eieren en bacon), is het Nederlandse ontbijt eerder functioneel en rationeel van aard. Het biedt voldoende energie, maar is tegelijkertijd eenvoudig in voorbereiding. Deze pragmatische benadering staat in schril contrast met de ****indulgentie**** die je in sommige andere Europese landen vindt, waar het ontbijt een uitgebreider en luxueuzer karakter heeft.

Vergelijking met andere Europese ontbijttradities

In andere Europese landen ziet men vaak veel rijkere of complexere ontbijtgewoonten. Bijvoorbeeld in ****Frankrijk**** is het ontbijt doorgaans licht maar verfrijend, met croissants, pain au chocolat en een café au lait. Het Franse ontbijt richt zich meer op smaakbeleving en is meestal zoet. Dit staat in contrast met het Nederlandse gebruik van hartiger beleg en het frequente gebruik van boter en kaas.

In ****Engeland**** is het ontbijt vaak een meer voedzame maaltijd met een brede variëteit aan gerechten, zoals gebakken eieren, worst, bacon, tomaten en bonen. Het Engelse ontbijt wordt vaak geassocieerd met een "full" en calorierijke maaltijd, wat een heel ander ritueel is dan het lichte Nederlandse ontbijt, dat meer gericht is op eenvoud en snelheid.

In ****Duitsland**** lijkt het ontbijt qua samenstelling dichter bij het Nederlandse ontbijt te liggen, maar de nadruk ligt meer op het gebruik van donker, zwaar brood en een verscheidenheid aan ****wurst**** en ****käse****. De Duitse traditie maakt ook meer gebruik van ****geroosterd brood****, terwijl in Nederland het ontbijt meestal uit vers brood bestaat.

Conclusie

Het typische Nederlandse ontbijt weerspiegelt de culturele waarden van eenvoud en efficiëntie. Het biedt een evenwichtige en snelle maaltijd, gericht op praktische, voedzame opties. In vergelijking met andere Europese landen, waar ontbijten vaak een uitgebreid ritueel is, blijft het Nederlandse ontbijt relatief bescheiden. De keuze voor brood en de verscheidenheid aan belegen is een weerspiegeling van de Nederlandse voorkeur voor ****praktische en voedzame gerechten**** die snel en gemakkelijk kunnen worden geconsumeerd.

E BENCHMARKING RESULTS GEMMA-3-27B-IT

E.1 MMLU

Total percentage correct: 70.17%
 Percentage correct in STEM: 70.30%
 Percentage correct in Business: 73.38%
 Percentage correct in Medical: 72.71%
 Percentage correct in Social Sciences: 79.13%
 Percentage correct in Humanities: 61.82%
 Percentage correct in Other: 81.73%

E.2 Language tool results

```

-----
Rule ID: MORFOLOGIK_RULE_NL_NL
Message: Er is een mogelijke spelfout gevonden.
Replacements: []
Offset: 2498
Error Length: 15
Context: ...opulariteit van bijvoorbeeld smoothies,
chiazaadpudding en andere 'gezondere' opties.
Desondank...
Offset in Context: 43
Sentence: Er is een groeiende populariteit van
bijvoorbeeld smoothies, chiazaadpudding en andere
'gezondere' opties.
Rule Issue Type: misspelling
Category: TYPOS
-----
Rule ID: MORFOLOGIK_RULE_NL_NL
Message: Er is een mogelijke spelfout gevonden.
Replacements: ['Le', 'LE', 'Lê', ...]
Offset: 3014
Error Length: 2
Context: ...eiden. Frankrijk: Het Franse ontbijt
('le petit déjeuner') is vaak minimalistisch...
Offset in Context: 43
Sentence: Frankrijk: Het Franse ontbijt ('le petit
déjeuner') is vaak minimalistisch, bestaande uit een
croissant of pain au chocolat met koffie.
Rule Issue Type: misspelling
Category: TYPOS
-----
Rule ID: MORFOLOGIK_RULE_NL_NL

```

Error	Error explanation	Run 1	Run 2	Run 3
MORFOLOGIK_RULE_NL_NL	Er is een mogelijke spelfout gevonden.	1 false positive; 8 false positive because of foreign words	1 true positive; 3 false positive because of foreign words; 1 false positive due to uncommon compound word	2 false positives due to uncommon compound word; 3 false positive because of foreign words
NL_SIMPLE_REPLACE____MAAR	Na een komma hoort geen hoofdletter.	0 errors	1 true positive	0 errors
KOMMA_MAAR	Voor 'maar' komt (als u een pauze hoort) een komma: "bescheiden, maar".	0 errors	0 errors	1 true positive

Table 4. LanguageTool errors for Gemma-3-27B-it

Message: Er is een mogelijke spelfout gevonden.
 Replacements: ['Petit', 'Petits', 'Petiet', ...]
 Offset: 3017
 Error Length: 5
 Context: ...en. Frankrijk: Het Franse ontbijt ('le petit déjeuner') is vaak minimalistisch, best...
 Offset in Context: 43
 Sentence: Frankrijk: Het Franse ontbijt ('le petit déjeuner') is vaak minimalistisch, bestaande uit een croissant of pain au chocolat met koffie.
 Rule Issue Type: misspelling
 Category: TYPOS

Rule ID: MORFOLOGIK_RULE_NL_NL
 Message: Er is een mogelijke spelfout gevonden.
 Replacements: ['pan', 'puin', 'Main', ...]
 Offset: 3089
 Error Length: 4
 Context: ...istisch, bestaande uit een croissant of pain au chocolat met koffie. Er is minder na...
 Offset in Context: 43
 Sentence: Frankrijk: Het Franse ontbijt ('le petit déjeuner') is vaak minimalistisch, bestaande uit een croissant of pain au chocolat met koffie.
 Rule Issue Type: misspelling
 Category: TYPOS

Rule ID: MORFOLOGIK_RULE_NL_NL
 Message: Er is een mogelijke spelfout gevonden.
 Replacements: ['Chocolat', 'chocola', 'chocola-', ...]
 Offset: 3097
 Error Length: 8
 Context: ... bestaande uit een croissant of pain au chocolat met koffie. Er is minder nadruk op eiwi...
 Offset in Context: 43

Sentence: Frankrijk: Het Franse ontbijt ('le petit déjeuner') is vaak minimalistisch, bestaande uit een croissant of pain au chocolat met koffie.
 Rule Issue Type: misspelling
 Category: TYPOS

Rule ID: MORFOLOGIK_RULE_NL_NL
 Message: Er is een mogelijke spelfout gevonden.
 Replacements: ['breukvast', 'breuklast', 'breaktest', ...]
 Offset: 3218
 Error Length: 9
 Context: ...and: Het Engelse ontbijt ('full English breakfast') is berucht om zijn omvang. Gebakken e...
 Offset in Context: 43
 Sentence: Engeland: Het Engelse ontbijt ('full English breakfast') is berucht om zijn omvang.
 Rule Issue Type: misspelling
 Category: TYPOS

Rule ID: MORFOLOGIK_RULE_NL_NL
 Message: Er is een mogelijke spelfout gevonden.
 Replacements: ['Drumstick', 'Rugstuk', 'Fruitstuk', ...]
 Offset: 3412
 Error Length: 9
 Context: ...erheid. Duitsland: Het Duitse ontbijt ('Frühstück') is vaak uitgebreider dan het Nederlan...
 Offset in Context: 43
 Sentence: Duitsland: Het Duitse ontbijt ('Frühstück') is vaak uitgebreider dan het Nederlandse, met een grote verscheidenheid aan broodsoorten, vleeswaren, kaas en zoetigheden.
 Rule Issue Type: misspelling
 Category: TYPOS

```
-----
Rule ID: MORFOLOGIK_RULE_NL_NL
Message: Er is een mogelijke spelfout gevonden.
Replacements: ['Delano', 'lesauto', 'Desain', ...]
Offset: 3634
Error Length: 8
Context: ...t groter. Spanje: Het Spaanse ontbijt ('desayuno') is vaak licht en zoet, bestaande uit ...
Offset in Context: 43
Sentence: Spanje: Het Spaanse ontbijt ('desayuno') is vaak licht en zoet, bestaande uit een 'tostada' (geroosterd brood) met olijfolie en tomaat, of een 'churros' met chocolade.
Rule Issue Type: misspelling
Category: TYPOS
-----

Rule ID: MORFOLOGIK_RULE_NL_NL
Message: Er is een mogelijke spelfout gevonden.
Replacements: ['Tostado']
Offset: 3687
Error Length: 7
Context: ... vaak licht en zoet, bestaande uit een 'tostada' (geroosterd brood) met olijfolie en to...
Offset in Context: 43
Sentence: Spanje: Het Spaanse ontbijt ('desayuno') is vaak licht en zoet, bestaande uit een 'tostada' (geroosterd brood) met olijfolie en tomaat, of een 'churros' met chocolade.
Rule Issue Type: misspelling
Category: TYPOS
-----

Total spelling errors: 9
```

E.3 ChatGPT (gpt-o3-2025-04-16 (zero-shot)) EuroEval results

Benchmark	Result	English equivalent result
CONLL NL	67.06 ± 3.19 / 56.12 ± 2.14	86.49 ± 0.97 / 78.67 ± 0.99
DBRD	93.56 ± 0.94 / 96.75 ± 0.48	-
ScaLA NL	68.86 ± 2.85 / 83.25 ± 1.59	51.94 ± 4.06 / 74.20 ± 2.27
SQuAD NL	72.99 ± 0.77 / 48.30 ± 1.43	81.43 ± 0.52 / 62.39 ± 1.41
WikiLingua NL	65.15 ± 0.08 / 13.31 ± 0.19	-
MMLU NL	84.11 ± 2.00 / 88.09 ± 1.48	95.25 ± 1.03 / 96.45 ± 0.76
HellaSwag NL	73.36 ± 2.14 / 80.12 ± 1.58	81.34 ± 2.26 / 85.82 ± 1.73

Table 5. Benchmark results from EuroEval on gpt-o3-2025-04-16 (zero-shot)

F GENERATED FEEDBACK EXAMPLE GEMMA-3-27B-IT

```
#### De deelnemer beschrijft tenminste drie verschillen en/of overeenkomsten tussen de modellen voor formatief toetsen uit paragraaf 3.2.
Criterium voldaan: voldaan
Positieve opmerkingen:
- Je hebt er twee duidelijke verschillen benoemd tussen de modellen van Black & Wiliam en Gulikers & Baartman, en een overeenkomst, wat aantoont dat je de modellen hebt bestudeerd.
Verbeterpunten:
- Je zou nog een extra verschil of overeenkomst kunnen benoemen om volledig aan het criterium te voldoen.
- De beschrijving van de overeenkomst zou specifiekere kunnen zijn; noem concrete elementen die beide modellen gemeen hebben.
- Probeer de verschillen en overeenkomsten meer te baseren op de theorie zoals beschreven in het tekstboek (bijv. de focus op strategieën vs. een cyclisch proces).

#### De deelnemer beschrijft tenminste drie verschillen en/of overeenkomsten tussen de modellen voor summatief toetsen uit paragraaf 3.3.
Criterium voldaan: voldaan
Positieve opmerkingen:
- Je hebt drie duidelijke verschillen benoemd tussen de modellen van de Hanzehogeschool en Moelands, wat aantoont dat je de modellen hebt begrepen.
Verbeterpunten:
- Je zou de verschillen kunnen relateren aan de functies van summatief toetsen, zoals de nadruk op normering en betrouwbaarheid.
```

- Beschrijf de verschillen iets gedetailleerder. Wat betekent het dat het model van Moelands de doel- en toetsspecificaties eerst uitwerkt?
- Vergelijk de modellen niet alleen op de stappen, maar ook op de onderliggende principes (bijvoorbeeld de mate van detail in de constructie).

De deelnemer beschrijft tenminste drie overeenkomsten en/of verschillen tussen modellen voor formatief toetsen en modellen voor summatief toetsen.

Criterium voldaan: niet voldaan

Positieve opmerkingen:

- Je hebt de essentie van het verschil (formatief evalueren vs. summatief toetsen) wel benoemd ("toetsen om te leren" vs. "toetsen van het leren").

Verbeterpunten:

- Je geeft geen drie concrete overeenkomsten of verschillen tussen de *modellen* zelf, maar concentreert je meer op het verschil tussen de toetsfuncties.
- Ga dieper in op de modellen: welke elementen van de formatieve modellen ontbreken in de summatieve modellen, en vice versa?
- Gebruik de informatie uit het tekstboek om dit criterium beter te beantwoorden (bijv. het belang van feedback in formatief evalueren).

De deelnemer onderbouwt welk formatief model hij/zij zou willen gebruiken of momenteel gebruikt, geeft hiervoor argumenten en integreert praktijkervaringen in de onderbouwing.

Criterium voldaan: voldaan

Positieve opmerkingen:

- Je geeft een duidelijke voorkeur voor het model van B&W en onderbouwt dit met argumenten zoals inzichtelijkheid en het betrekken van leerlingen.

Verbeterpunten:

- Je zou je argumentatie kunnen versterken door te linken aan de principes van formatief evalueren (zoals het bevorderen van zelfregulatie).
- Je praktijkervaringen zouden specifiekere kunnen zijn. Geef een concreet voorbeeld van hoe je het model van B&W in de klas hebt toegepast.
- Overweeg om in te gaan op de essentie van stap 5 uit het G&B model, zoals je aangeeft dat dit niet uit het oog verloren mag worden.

De deelnemer onderbouwt welk summatief model hij/zij zou willen gebruiken of momenteel gebruikt, geeft hiervoor duidelijke argumenten en integreert praktijkervaringen in de onderbouwing.

Criterium voldaan: voldaan

Positieve opmerkingen:

- Je geeft een duidelijke voorkeur voor het model van BKE/SKE en onderbouwt dit met een relevant argument (tijdsdruk).

Verbeterpunten:

- Je argumentatie is vooral gebaseerd op praktische overwegingen (tijd). Je zou ook de pedagogische aspecten kunnen betrekken.
- Je kunt je argumentatie versterken door concrete voorbeelden te geven van situaties waarin het model van Moelands niet haalbaar zou zijn.
- Ga in op de mogelijke nadelen van het model van BKE/SKE en hoe je die kunt ondervangen.

De deelnemer beschrijft de huidige verhouding tussen formatief en summatief toetsen binnen de eigen school en geeft concrete voorbeelden ter illustratie.

Criterium voldaan: voldaan

Positieve opmerkingen:

- Je geeft een eerlijk en realistisch beeld van de huidige situatie op je school: weinig structureel toetsbeleid en een focus op summatieve toetsen.

Verbeterpunten:

- Je kunt de constatering dat het 'zweten-weten-vergeten' een nadelig gevolg is, nog meer uitwerken.
- Geef concrete voorbeelden van formatief handelen dat docenten *wel* toepassen, ook al is het niet structureel.
- Beschrijf hoe de werkgroep 'Formatief Handelen' probeert om de verhouding te veranderen.

De deelnemer beschrijft hoe hij/zij idealiter de verhouding tussen formatief en summatief toetsen zou willen zien en geeft concrete voorbeelden van hoe deze ideale verhouding eruit zou kunnen zien binnen zijn/haar eigen vakgebied.

Criterium voldaan: voldaan

Positieve opmerkingen:

- Je erkent dat de ideale verhouding afhankelijk is van het vak, leerjaar en niveau, wat een realistisch standpunt is.

Verbeterpunten:

- Geef specifieke voorbeelden van hoe je de verhouding in *jouw* vakgebied idealiter zou zien. Welke concrete formatieve activiteiten zou je willen inzetten?
- Beschrijf hoe je de formatieve data zou gebruiken om het onderwijs aan te passen.
- Ga dieper in op de voordelen van een grotere nadruk op formatief toetsen voor je leerlingen.