MONKEY at SimpleText 2026: Static Few-Shot Prompting for Cochrane Review Scientific Text Simplification

SimpleText 2026 Track

Feifan Chen[1], Leilei Kong[1,*] and Yong Lu[2]

[1]School of Computer Science and Artificial Intelligence, Foshan University, China
[2]School of Artificial Intelligence Engineering, Guangzhou College of Technology and Business, China

Abstract

The Cochrane review scientific text simplification task asks systems to convert sentences extracted from biomedical abstracts in Cochrane systematic reviews into simplified text for non-expert readers. We address Cochrane review text simplification with a large language model prompting approach based on static few-shot prompting. A fixed set of examples is selected from the training data according to predefined strategies and inserted into the prompt. For comparison, we also evaluate zero-shot prompting and dynamic few-shot prompting. In addition, we explore supervised routing and generation pipelines. Our results show that static few-shot prompting performs best in our test configuration, suggesting that carefully selected examples and explicit simplification instructions are effective for sentence-level scientific text simplification.

Keywords

Text Simplification, Scientific Text, Large Language Models, Few-Shot Prompting

1. Introduction

The SimpleText 2026 track studies methods for simplifying scientific text while preserving factual content. Task 1 focuses on simplifying Cochrane systematic reviews into plain language for non-expert readers [1, 2]. The track also includes hallucination detection and research area classification tasks [3, 4]. In this paper, we report the MONKEY submission to Task 1.1, the sentence-level scientific text simplification subtask.

Scientific and biomedical texts are difficult for general readers because they contain technical terminology, statistical expressions, and domain-specific writing conventions [5]. Text simplification can be defined as the automatic process of transforming advanced or complex input text into a simpler and more readable form [5]. It is usually not limited to summarization or paraphrasing, but instead focuses on three related types of simplification: lexical simplification, which replaces rare or technical words with more common alternatives; syntactic simplification, which breaks long or complex sentence structures into simpler statements or more direct word order; and informational or conceptual simplification, which removes redundant modifiers or explains and condenses abstract concepts [5, 6]. Text simplification has been widely studied as a natural language processing task [6, 5]. In this work, we focus on sentence-level simplification, where complex Cochrane review sentences are rewritten into simpler forms while preserving their meaning [7]. In the biomedical domain, this task is especially sensitive because systems should reduce linguistic complexity without changing medical findings, treatment names, outcomes, or evidence statements [8].

Recent work has explored scientific text simplification from several directions, including fine-tuned sequence-to-sequence models, large language model prompting, complex term rewriting, and planning-based simplification [9, 10, 11, 12]. These studies reveal two practical difficulties. First, sentence-level simplification provides limited context, making it difficult to decide whether a sentence should be deleted [11]. Second, GPT-based experiments have reported mixed results for fine-tuned systems

CLEF 2026 Working Notes, 21 – 24 September 2026, Jena, Germany
*Corresponding author.
cff20000605@gmail.com (F. Chen); kongleilei@fosu.edu.cn (L. Kong); luy_2019@126.com (Y. Lu)
0009-0001-9344-3411 (F. Chen)

compared with prompted systems [10], and prior work suggests that model-based simplification does not always improve after fine-tuning [8].

Our work focuses on large language model prompting for sentence-level simplification. Cochrane reviews contain highly specialized biomedical language, including statistical expressions, clinical terminology, and evidence-certainty statements, which makes simplification particularly challenging. Although previous SimpleText systems have shown the potential of fine-tuned models and prompting methods, systematic comparisons of different prompt strategies for Cochrane-specific text simplification remain limited. We address this gap by comparing zero-shot prompting, dynamic few-shot prompting, and a single static few-shot prompt. Although dynamic retrieval can provide input-specific examples, our evaluation shows that static few-shot prompting performs best in our test configuration. We therefore use the static prompt as our main submission setting. We also conduct supplementary experiments with supervised routing and fine-tuned generators, but these systems do not outperform the prompting baseline.

The rest of this paper is organized as follows. Section 2 describes the task setting, zero-shot baseline, static few-shot prompt, dynamic few-shot prompt, and supplementary supervised pipelines. Section 3 presents the experimental setup. Section 4 reports the results and analysis. Section 5 concludes the paper.

2. Method

2.1. Task setting

We participated in SimpleText 2026 Task 1.1 [2]. We formulate the task as text simplification: given a complex sentence from a Cochrane systematic review, the system must generate a simpler version that is easier for non-expert readers to understand while preserving factual content. Unlike paraphrasing, which mainly restates a sentence at a similar complexity level, or style transfer, which changes register and tone, text simplification specifically aims to reduce linguistic complexity by replacing technical terminology with plain language and removing unnecessary statistical details. The input consists of sentence-level scientific text from Cochrane systematic reviews. Each input sentence is associated with a simplified reference sentence, or in some cases no simplified output.

Our system is built around prompt-based sentence simplification. We compare zero-shot prompting, static few-shot prompting, and dynamic few-shot prompting. We also explore supervised routing and generation pipelines as supplementary methods, but they are not used as the main submission setting.

2.2. Zero-shot baseline

Our baseline is a zero-shot prompt. The model receives only the task instruction and the input sentence. The instruction asks the model to rewrite the sentence in plain language for general readers while preserving medical findings, treatments, outcomes, and evidence statements.

The zero-shot setting measures the model's ability to follow simplification instructions without examples. However, without examples, model behavior can be inconsistent: it may over-rewrite the input, retain too many statistical details, or delete useful information.

2.3. Static few-shot prompting

Our main method is based on static few-shot prompting. The central idea is to guide the large language model with a carefully designed instruction and a fixed set of examples embedded in the prompt. The full prompt consists of six parts, each with a specific motivation.

Role definition. The first sentence defines the model as "You are a Cochrane medical sentence simplifier". This anchors the model in the setting of Cochrane medical text simplification and reduces ambiguity between generic rewriting and domain-specific medical simplification.

Output format constraint. The output format section states that the model may output only the simplified sentence or the [DELETE] marker, with no explanatory text. This constraint is intended to remove common redundant model outputs, such as prefixes like "Here is the simplified version:", and to ensure that outputs can be used directly in automatic evaluation.

Main goal. This section states the core simplification principle: preserve useful findings, remove technical statistical clutter, and treat most sentences as candidates for rewriting rather than deletion. This design addresses the tendency of zero-shot prompts to either delete too aggressively or retain too much detail.

Delete policy. This section defines the boundary for deletion decisions from both positive and negative directions. The model should output [DELETE] only when the sentence is purely about internal review methods, such as risk-of-bias tools, search, screening, or data extraction procedures, statistical heterogeneity methods, GRADE assessment procedures, or blinding and allocation concealment methods. Conversely, the prompt lists six types of content that should never be deleted: treatments, interventions, comparisons, diseases, patient groups, or outcomes; directions of effect; evidence certainty; numbers of studies, trials, participants, or follow-up time; recommendations that more research is needed; and conclusion or summary statements. These constraints provide explicit decision boundaries and reduce uncertainty in deletion decisions.

Rephrase rules. Five rules guide the model's rewriting behavior: (1) remove statistical details in parentheses, such as OR, RR, CI, I2, and p-values; (2) preserve the main conclusion, direction of effect, outcome, and treatment comparison; (3) replace technical terms with plain language, such as replacing RCT with study and myocardial infarction with heart attack; (4) keep the sentence almost unchanged if it is already simple; and (5) preserve important numbers, such as "16 studies". The term replacement list was manually constructed from common usage patterns in Cochrane plain language summaries.

Examples. We select a fixed set of examples from the training data. Each example contains a complex sentence and its simplified output. The examples cover three operation types: rephrasing, with nine examples of varying difficulty; deletion, with six examples that are purely methodological sentences; and keeping unchanged, with three examples where the original sentence is already simple. These examples provide concrete input-output mappings so that the model can learn the style and granularity of the reference simplifications.

Critical constraints. The final part of the prompt restates the core constraints in concise form: output only the simplified text or [DELETE], never add information not present in the original sentence, and prefer rephrasing over deletion when uncertain. This part reinforces instruction following.

The full prompt is shown in Appendix A.

2.4. Dynamic few-shot prompting

We also test a dynamic few-shot setting. In this setting, BM25 retrieves training examples that are lexically similar to the input sentence [13]. The retrieved examples are inserted into the prompt as demonstrations.

The motivation for dynamic retrieval is that Cochrane sentences often share repeated patterns, such as evidence-certainty statements, statistical result descriptions, and treatment comparison structures. However, dynamic retrieval can also introduce noise when retrieved examples are lexically similar but correspond to different simplification operations. We therefore evaluate dynamic few-shot prompting as a comparison method rather than using it as the final submission setting.

2.5. Supervised pipelines

We additionally explore supervised two-stage pipelines as supplementary experiments. In one setting, a DeBERTa-based classifier predicts whether each sentence should be deleted or simplified, and a fine-tuned T5-small model generates simplified outputs for non-deleted sentences. In another setting, a BM25-based routing classifier assigns each sentence to an operation type, such as statistical result, evidence quality, or study count, and then applies rule-based or model-based rewriting.

These supervised systems are used to analyze whether explicit operation prediction can improve sentence-level simplification. Since they do not outperform static few-shot prompting in evaluation, they are not used as the main system.

3. Experimental Setup

We evaluate our systems on the sentence-level setting of SimpleText 2026 Task 1.1. The evaluation set used in our experiments contains 500 sentence-level examples. We use standard automatic metrics, including SARI, BLEU, Flesch-Kincaid Grade Level (FKGL), and compression ratio. SARI is used as the primary simplification metric [7]. BLEU is used as an additional surface-overlap metric. FKGL estimates readability, where lower values indicate easier reading. Compression ratio measures how much the output is shortened relative to the source sentence.

The main prompt-based experiments use DeepSeek V4 Flash. Decoding uses temperature 0.1 and a maximum of 200 new tokens for sentence-level simplification. We do not use top-p sampling.

For dynamic retrieval, BM25 uses a simple regular-expression tokenizer that splits text into alphabetic words and numeric patterns: `[a-z]+|\d+(?:\.\d+)?%?`. The default number of retrieved examples is $k = 3$. During development, we also test $k = 1, 2, 3, 4, 5$.

For classifier-based experiments, we test DeBERTa-v3-small and DeBERTa-v3-base. The DeBERTa-v3-small classifier uses maximum input length 512, batch size 32, learning rate 2×10^{-5}, and 10 epochs. The DeBERTa-v3-base classifier uses maximum input length 512, batch size 16, learning rate 1×10^{-5}, and 15 epochs. Generation in the supervised pipeline uses T5-small with maximum source length 160 and maximum target length 160. All hyperparameters are selected on the validation set.

We apply lightweight post-processing to clean model outputs. Post-processing removes leading generation prefixes, such as "Output:" or "Simplified:", extra whitespace, and empty parentheses. If the model outputs `[DELETE]`, it is converted to an empty string. We do not use post-processing to add new content to the output.

We compare the following settings:

- **Zero-shot prompting**: task instructions without examples.
- **Static few-shot prompting**: a fixed demonstration prompt used for all input sentences.
- **Dynamic few-shot prompting**: BM25 retrieves examples for each input sentence.
- **Classifier + generator**: a supervised two-stage pipeline with a DeBERTa router and a fine-tuned T5-small generator.
- **Retrieval router + generator**: a retrieval-based router decides whether to delete or simplify each sentence, and a generator produces outputs for non-deleted sentences.

4. Results and Discussion

4.1. Main results

Table 1 reports the Task 1.1 evaluation results. Static few-shot prompting obtains the best SARI score in our test configuration.

The results show that static few-shot prompting achieves the highest SARI score. Zero-shot prompting is close to static few-shot prompting, while dynamic few-shot prompting performs worse in SARI even though it obtains a higher BLEU score. This suggests that lexical similarity in retrieved examples does not necessarily correspond to the correct simplification operation for the input sentence. BLEU is computed at corpus level using sacrebleu [14].

The classifier-based and retrieval-routing pipelines do not outperform the pure prompting systems in SARI. The classifier + generator pipeline achieves the lowest FKGL, indicating shorter or easier-to-read outputs, but its compression ratio is also much lower. This suggests that it may simplify through excessive shortening and may delete useful content.

Table 1
Evaluation results for Task 1.1 sentence-level scientific text simplification on 500 Cochrane-auto aligned sentence pairs. BLEU is computed at corpus level using sacrebleu. Lower FKGL indicates better readability.

Method	SARI	BLEU	FKGL	CR
Zero-shot prompting	40.11	16.05	10.83	0.82
Static few-shot prompting	**42.57**	16.58	10.92	0.79
Dynamic few-shot prompting	38.66	**18.70**	11.48	0.81
Classifier + generator	38.41	10.76	**7.91**	0.44
Retrieval router + generator	39.79	23.07	11.71	0.68

4.2. Effect of prompting strategy

The comparison among zero-shot, static few-shot, and dynamic few-shot prompting shows that static few-shot prompting provides the most stable guidance for the model. Zero-shot prompting relies only on the model's instruction-following ability and is competitive but less guided. Dynamic few-shot prompting introduces input-specific examples, but the retrieved examples may not match the required operation. For example, retrieved sentences may be lexically similar because they contain statistical expressions, while their references may correspond to deletion, light rewriting, or keeping the sentence unchanged.

Static few-shot prompting avoids this variability by using a fixed demonstration set that covers common operation types, making model behavior more consistent across examples.

4.3. Effect of the number of retrieved examples

To study the effect of the number of retrieved examples in dynamic few-shot prompting, we vary k from 1 to 5. Table 2 reports the results.

Table 2
Effect of the number of retrieved examples k in dynamic few-shot prompting with BM25.

k	SARI	BLEU	FKGL	CR
1	39.35	18.46	11.67	0.84
2	39.37	19.06	11.48	0.84
3	39.33	19.70	11.69	0.83
4	40.86	22.07	11.97	0.83
5	**42.38**	**24.37**	11.98	0.85

The results show that $k = 5$ obtains the highest SARI and BLEU scores. However, we use $k = 3$ as the default setting in the dynamic few-shot experiment because the improvement from $k = 3$ to $k = 5$ is limited and, given the inherent non-determinism of large language model outputs, may not generalize across runs. The overall trend suggests that a small number of retrieved examples, around three to five, is sufficient, while too few examples, such as $k = 1$, provide insufficient guidance.

4.4. Analysis of supervised pipelines

We further explore supervised two-stage pipelines as supplementary methods. In the classifier + generator system, a DeBERTa-based classifier first predicts whether each sentence should be deleted or simplified, and a fine-tuned T5-small model generates a simplified output for non-deleted sentences. We also test a BM25-based routing classifier that assigns sentences to operation types.

These supervised systems do not outperform static few-shot prompting. The best supervised pipeline obtains a SARI score of 38.41, and the retrieval router + generator obtains 39.79. We attribute this result to two factors. First, the DELETE and KEEP classes have highly overlapping surface features. Some sentences labeled DELETE are well-formed biomedical statements that include treatment or outcome

information, but they are omitted from the reference simplification. As a result, the DELETE versus KEEP classifier reaches only about 64% accuracy, only slightly above the majority-class baseline of about 63%. Second, the fine-tuned T5-small model often either copies the input too closely or oversimplifies it, failing to capture the more nuanced rewriting patterns in the reference data.

This analysis supports our main finding: in our setting, a carefully designed static few-shot prompt is more effective than the supervised fine-tuning pipelines we tested.

4.5. DELETE accuracy analysis

We analyze the ability of prompt-based methods to identify sentences that should be deleted. We map reference labels to a binary setting: sentences labeled `delete` are treated as DELETE, and all other labels are treated as KEEP. We compare this with model outputs: empty outputs after post-processing are treated as DELETE, and non-empty outputs are treated as KEEP.

Table 3
DELETE classification performance of three prompting methods against reference annotations (177 DELETE and 323 KEEP examples).

Method	DELETE (%)			Accuracy	Pred.	Ref.
	Precision	Recall	F1	(%)	DEL	DEL
Zero-shot	73.1	10.7	18.7	67.0	26	177
Static few-shot	79.0	8.5	15.3	66.8	19	177
Dynamic few-shot	71.4	11.3	19.5	67.0	28	177

Table 3 shows that all three prompting methods are highly conservative when predicting DELETE. When they predict DELETE, they are usually correct, with precision between 71% and 79%, but they miss most sentences that should be deleted, with recall between 8% and 11%. This explains the relatively low SARI deletion subscore and suggests that the DELETE policy in the prompt is too strict. Static few-shot prompting has the highest DELETE precision but the lowest recall, while dynamic few-shot prompting achieves the best DELETE F1 among the three methods.

We manually inspect the outputs of static few-shot prompting and supervised pipelines. The most common errors are over-deletion, incomplete removal of statistical details, and rewrites that change the strength of the original evidence statement. For supervised routing systems, false deletion is especially harmful because factual content is removed before generation. For prompt-based systems, the most common issue is inconsistency: the model sometimes follows the deletion policy correctly, but in other cases keeps methodological or statistical details.

Another common issue is the trade-off between readability and content preservation. Systems with lower FKGL may produce shorter outputs, but a low compression ratio may indicate that important information has been deleted. This is visible in the classifier + generator system, which has the lowest FKGL but also the lowest compression ratio and a lower SARI score.

5. Conclusion

This paper describes the MONKEY participation in SimpleText 2026 Task 1.1. We compare several LLM prompting strategies for sentence-level scientific text simplification, including zero-shot prompting, dynamic few-shot prompting, and static few-shot prompting. The best result is obtained by static few-shot prompting with DeepSeek V4 Flash.

We also explore supervised two-stage pipelines as supplementary methods. These systems combine a DeBERTa-based deletion classifier, a fine-tuned T5-small generator, and a MiniLM-based operation router. However, the supervised systems do not outperform the static prompting baseline. We find that deletion decisions are difficult because DELETE and KEEP sentences often have similar surface features. These results suggest that, under limited training data, carefully selected static demonstrations

can provide more reliable guidance than the supervised routing and generation pipelines tested in this work.

Generative AI Statement

During the preparation of this work, the authors used ChatGPT and Gemini for grammar and spelling checks. After using these tools or services, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No.62276064) and Guangdong Basic and Applied Basic Research Foundation (No.2024A1515140093).

References

[1] L. Ermakova, et al., Overview of CLEF 2026 SimpleText track: Simplify scientific texts (and nothing more), in: M. Hagen, et al. (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Seventeenth International Conference of the CLEF Association (CLEF 2026), LNCS, Springer-Verlag, 2026.

[2] J. Bakker, et al., Overview of the CLEF 2026 SimpleText task 1: Simplify scientific text, in: E. Sánchez Salido, et al. (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2026), CEUR Workshop Proceedings, CEUR-WS.org, 2026.

[3] B. Chakar, et al., Overview of the CLEF 2026 SimpleText task 2: Identify and avoid hallucination, in: E. Sánchez Salido, et al. (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2026), CEUR Workshop Proceedings, CEUR-WS.org, 2026.

[4] G. K. Shahi, et al., Overview of the CLEF 2026 SimpleText task 3: Research area classification, in: E. Sánchez Salido, et al. (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2026), CEUR Workshop Proceedings, CEUR-WS.org, 2026.

[5] S. S. Al-Thanyyan, A. M. Azmi, Automated text simplification: A survey, ACM Computing Surveys 54 (2021) 1–36.

[6] A. Siddharthan, A survey of research on text simplification, International Journal of Applied Linguistics 165 (2014) 259–298.

[7] W. Xu, C. Napoles, E. Pavlick, Q. Chen, C. Callison-Burch, Optimizing statistical machine translation for text simplification, in: Transactions of the Association for Computational Linguistics, volume 4, 2016, pp. 401–415.

[8] A. Devaraj, I. Marshall, B. C. Wallace, J. J. Li, Paragraph-level simplification of medical texts, in: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2021), 2021, pp. 4972–4984.

[9] T. Papandreou, J. Bakker, J. Kamps, University of amsterdam at the CLEF 2025 SimpleText track, in: Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2025), volume 4038 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2025, pp. 4356–4362. URL: https://ceur-ws.org/Vol-4038/paper_359.pdf.

[10] P. Kocbek, G. Stiglic, UM_FHS at the CLEF 2025 SimpleText track: Comparing no-context and fine-tune approaches for GPT-4.1 models in sentence and document-level text simplification, 2025. arXiv:2512.16541.

[11] N. Hofmann, J. Dauenhauer, N. O. Dietzler, I. D. Idahor, C. K. Kreutz, THM@SimpleText 2025 – task 1.1: Revisiting text simplification based on complex terms for non-experts, 2025. arXiv:2507.04414.

[12] K. C. Marturi, H. H. Elwazzan, LLM-guided planning and summary-based scientific text simplification: DS@GT at CLEF 2025 SimpleText, 2025. arXiv:2508.11816.

[13] S. Robertson, H. Zaragoza, The probabilistic relevance framework: BM25 and beyond, Foundations and Trends in Information Retrieval 3 (2009) 333–389.
[14] M. Post, A call for clarity in reporting BLEU scores, in: Proceedings of the Third Conference on Machine Translation: Research Papers, Association for Computational Linguistics, 2018, pp. 186–191.

A. Static Few-Shot Prompt

The static few-shot prompt used by our main system is shown below.

```
You are a Cochrane medical sentence simplifier.
Rewrite one complex medical sentence into plain language for general readers.

## OUTPUT FORMAT:
- Output ONLY the simplified sentence, or [DELETE] if the sentence should be removed.
- Do not explain your choice.
- One clear sentence preferred. Two short sentences allowed only if needed.

## MAIN GOAL:
Preserve the useful finding. Remove technical statistical clutter.
Most sentences should be REPHRASED, not deleted.

## DELETE POLICY:
Output [DELETE] only when the sentence is purely about internal review methods with no
    useful finding:
- Risk of bias assessment tools and methods
- Search/screening/data-extraction procedures
- Statistical heterogeneity methods (I2, sensitivity analysis)
- GRADE quality assessment procedures
- Blinding/allocation concealment methods

NEVER DELETE if the sentence contains:
- A treatment, intervention, comparison, disease, patient group, or outcome
- A result or direction of effect (improved, reduced, increased, similar, little
    difference, uncertain, no evidence)
- Evidence certainty or quality (low-certainty, high-certainty, very low certainty)
- Number of studies, trials, people, participants, or follow-up time
- A recommendation that more research is needed
- A conclusion or summary statement

## REPHRASE RULES:
1. REMOVE statistical details in parentheses: OR, RR, CI, I2, MD, SMD, p-values,
    confidence intervals
2. KEEP: main conclusion, direction of effect, outcome, treatment/comparison
3. REPLACE technical terms with plain language:
   - RCT/randomised controlled trial -> study
   - participants/people enrolled -> people
   - adverse events -> side effects
   - mortality -> deaths
   - efficacy -> how well it works
   - myocardial infarction -> heart attack
   - cognition -> thinking
   - prophylaxis -> prevention
   - pharmacological -> drug
   - neonatal -> newborn
   - tolerability -> how well people tolerate it
4. If already simple, keep it almost unchanged.
```

5. Preserve important numbers (e.g., "16 studies", "2232 couples")

EXAMPLES:

Rephrase:
Input: "We included 16 RCTs (2232 couples analysed)."
Output: We included 16 studies with 2232 couples.

Input: "We are uncertain whether glucocorticoids improved live birth rates (odds ratio (OR) 1.37, 95% confidence interval (CI) 0.69 to 2.71; 2 RCTs, n = 366; I2 = 7%; very low-certainty evidence)."
Output: We are uncertain whether glucocorticoids improved live birth rates.

Input: "Modafinil significantly improved self-reported sleepiness on the Epworth Sleepiness Scale by 5.08 points more than placebo (95% confidence interval (CI) 3.01 to 7.16; 2 studies, 101 participants; high-certainty evidence)."
Output: Modafinil improved self-reported sleepiness more than placebo.

Input: "However, short-term myocardial infarction was similar between both groups (RR 0.91, 95% CI 0.81 to 1.02; 19,430 participants; 11 studies; high quality evidence)."
Output: However, the risk of heart attack was similar in both groups.

Input: "It seemed that levetiracetam could improve cognition and lamotrigine could relieve depression, while phenobarbital and lamotrigine could worsen cognition."
Output: It seemed that levetiracetam could improve thinking and lamotrigine could help with depression, while phenobarbital and lamotrigine could make thinking worse.

Input: "Nasal interfaces were found to offer comparable efficacy to face masks (low- to very low-certainty evidence), supporting their use as an alternative to face masks for respiratory support in newborn infants."
Output: Nasal interfaces worked about as well as face masks for helping newborn babies breathe.

Input: "Additional metoclopramide significantly reduced nausea (P < 0.00006) and vomiting (P = 0.002) compared with aspirin alone."
Output: Adding metoclopramide reduced nausea and vomiting compared with aspirin alone.

Input: "The authors concluded that prenatal education for congenital toxoplasmoses has a significant effect on improving women’s knowledge."
Output: Prenatal education improved women’s knowledge about congenital toxoplasmosis.

Input: "The METAVIR activity score also improved (36/55 (65%) versus 20/46 (43.5%); RR 1.49, 95% CI 1.02 to 2.18; 2 trials)."
Output: Liver inflammation scores also improved with treatment.

Delete (pure methodology only):
Input: "We assessed the risk of bias of included studies using Cochrane’s ’Risk of Bias’ tool."
Output: [DELETE]

Input: "Potential sources of bias were a lack of blinding to treatment allocation of the caregivers and investigators in all trials."
Output: [DELETE]

Input: "We searched MEDLINE, EMBASE, and Cochrane Library from inception to December 2023."
Output: [DELETE]

```
Input: "The risk of bias relating to allocation, blinding and selective reporting was
    unclear."
Output: [DELETE]

Input: "Data extraction was performed independently by two review authors."
Output: [DELETE]

Input: "The I2 of 53% may represent moderate statistical heterogeneity and results have
    to be interpreted with caution."
Output: [DELETE]

### Keep unchanged:
Input: "The studies were conducted in North America, Europe, and Africa."
Output: The studies were conducted in North America, Europe, and Africa.

Input: "Planned follow-up periods ranged from six months to five years."
Output: Planned follow-up periods ranged from six months to five years.

Input: "We did not find any trial that compared different types of palliative care with
    each other."
Output: We did not find any trial that compared different types of palliative care with
    each other.

## CRITICAL:
- Output ONLY the simplified text or [DELETE].
- NEVER add information not in the original.
- When in doubt, REPHRASE rather than DELETE.
- Always simplify: replace technical terms, remove parenthetical statistics.
```