Doing More with Less: Strategies to Reduce the cost of Generative AI in HEOR

Oliver Pople,[1] Sarah Cudworth,[1]

[1] Estima Scientific, London, United Kingdom

MSR174

Introduction

- Advances in the reasoning capabilities of Large Language Models (LLMs) are expanding opportunities for their use within Health Economics and Outcomes Research (HEOR), including the automation of complex and resource-intensive workflows.[1, 2] However, advanced models can incur substantial costs due to higher per-token pricing (see **Table 1**) and increased token use during reasoning-intensive tasks.
- As adoption of generative AI within HEOR grows, strategies that optimize model utilization and reduce computational costs will be important to enable scalable and cost-efficient implementation.

Aim

- To evaluate the impact of optimisation strategies to reduce token usage and cost within an AI-enabled HEOR workflow, and consequently, to assess their potential to improve computational efficiency and reduce expenditure.

Methods

- Four optimization strategies (**Figure 1**), were evaluated within an LLM-assisted HEOR data-extraction workflow.
- One hundred open-source cost-effectiveness studies were identified and processed, extracting relevant study characteristics, economic outcomes and key findings, followed by synthesis of the extracted evidence.
- The search, data extraction and qualitative summary of the findings were performed by an LLM, with manual checkpoints at each stage.
- Each strategy was compared with a non-optimized baseline, assessing estimated cost, token utilization, and processing time.

Figure 1. Four strategies to optimize cost, token use and time.

01 Prompt caching

Caching repeated extraction instructions allows subsequent requests to reuse the same input, reducing processing costs.

02 Smart model routing

Simple tasks use a lower-cost model while complex tasks use an advanced reasoning model to balance cost and extraction accuracy.

03 Batch processing

Batch processing submits extraction requests grouped together to an externally hosted LLM to lower costs when longer processing times are acceptable.

04 Result caching

Stored study-level extractions support multiple evidence-synthesis tasks, avoiding repeat extraction to reduce token use and cumulative cost.

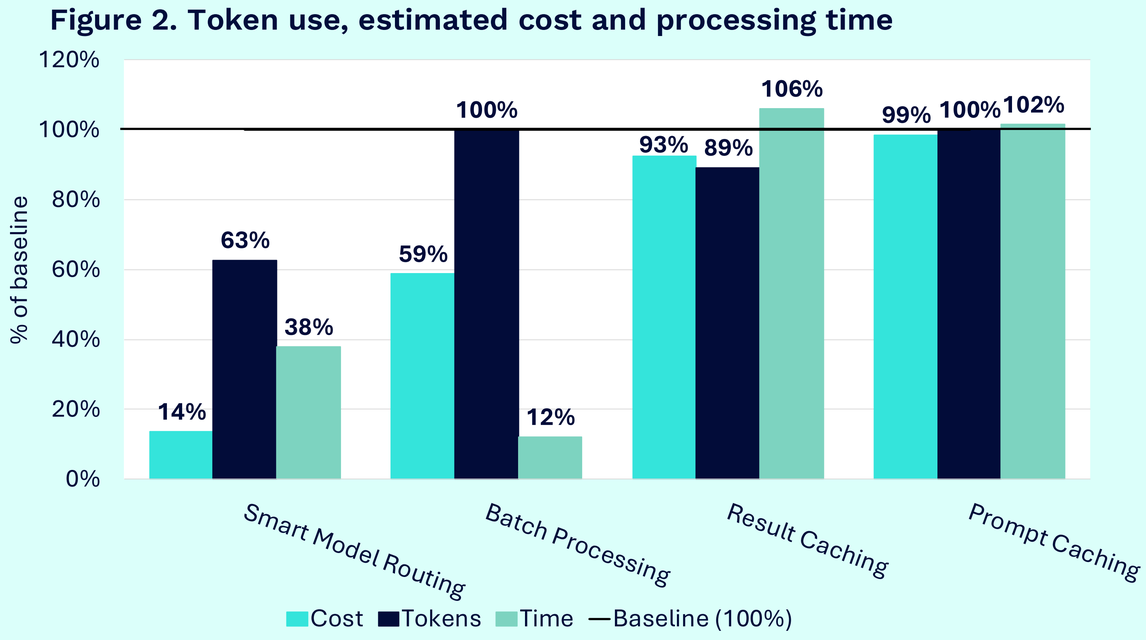
Table 1. LLM usage costs (USD)[3]

LLM Model	Cost per million tokens					
	Base input	Base output	Cache write 1 hour	Cache read	Batch input	Batch output
Claude Opus 5	$5.00	$25.00	$10.00	$1.00	$2.50	$12.50
Claude Haiku 4.5	$1.00	$5.00	$2.00	$0.20	$0.50	$2.50

Results

- The baseline workflow, using Claude Opus 5 without optimization, consumed 3.15 million tokens, with an estimated cost of $17.93 and processing time of 66 minutes.
- All four optimization strategies reduced estimated LLM costs relative to baseline.
- Smart model routing achieved the greatest cost reduction (86.4%), alongside a 37.3% reduction in token utilization and a 62.1% reduction in processing time. However, the suitability of lower-cost models and therefore the potential for cost, token and time reduction depends on task complexity, and potential trade-offs in reasoning capability and output quality.
- Batch processing reduced costs by 41.3%, with negligible changes in token utilization, demonstrating that cost savings can be achieved independently of token reduction. It also recorded the shortest processing time (8 minutes), although this should be interpreted in the context of the human-time investment of batch execution method and not the end-to-end turnaround time.
- Result caching reduced costs by 7.5% and token utilization by 10.8%, while prompt caching achieved a modest 1.5% cost reduction with negligible change in token utilization. These limited savings may reflect the document-intensive nature of the workflow, where processing 100 publications accounted for a substantial proportion of token usage, leaving fewer opportunities to benefit from repeated prompts or previously generated outputs.
- Overall, these findings demonstrate the potential for LLM optimization strategies to improve cost-efficiency within automated HEOR workflows, with model selection and batch processing delivering the greatest savings in this evaluation.
- However, a combination of these methods would likely yield even greater cost and token savings, with the design of the workflow potentially changing the ideal approach to cost-saving.
- All cost, token usage and time savings could vary based on chosen LLM model, size of batch processing and caching methods which are applied.

Figure 2. Token use, estimated cost and processing time



Method	Model	Total tokens	Baseline tokens	Estimated LLM cost ($)	Time (mins)
Baseline	**Claude Opus 5**	**3,150,123**	**3,150,123**	**$17.93**	**66**
Smart Model Routing	Claude Haiku 4.5	1,976,235	3,150,123	**$2.44**	25
Batch Processing	Claude Opus 5	3,150,924	3,150,123	**$10.53**	8
Result Caching	Claude Opus 5	2,809,635	3,150,123	**$16.59**	70
Prompt Caching	Claude Opus 5	3,151,903	3,150,123	**$17.67**	67

Estimated LLM costs use configured token rates.

Limitations

This study represents a simplified, exploratory assessment of LLM optimization strategies within selected HEOR workflows. Several limitations should be considered:

- Generalizability: Findings are specific to the workflows, models and tasks evaluated. Cost-saving potential may differ across other HEOR applications and should be validated in more complex, real-world workflows.
- Scale: The benchmarking task may not fully reflect the variability, complexity or computational demands of larger HEOR projects.
- Model and pricing variability: Cost savings are dependent on model selection, provider pricing and technical configurations, which may change over time.
- Combined effects: Strategies were assessed independently, and their potential cost savings may not be additive when implemented together.

Conclusions

Optimization strategies can substantially improve the cost-efficiency of generative AI within HEOR workflows. As the use of generative AI expands across HEOR, proactive management of costs will become increasingly important to support the long-term scalability of automation.

Further research for each workflow should evaluate these approaches across a broader range of HEOR workflows to establish their scalability, reproducibility and impact on cost-efficiency.

References

1. Reason T, Klijn S, Rawlinson W, Benbow E, Langham J, Teitsson S, Johannesen K, Malcolm B. Using Generative Artificial Intelligence in Health Economics and Outcomes Research: A Primer on Techniques and Breakthroughs: T. Reason et al. PharmacoEconomics-open. 2025 Jul;9(4):501-17.
2. Rawlinson W, Teitsson S, Reason T, Malcolm B, Gimblett A, Klijn SL. Generative artificial intelligence to automate the adaptation of Excel health economic models and word technical reports. Value in Health. 2025 Jun 14.
3. Anthropic. Pricing - Batch processing [Internet]. Claude Platform Docs. Available from: https://platform.claude.com/docs/en/about-claude/pricing#batch-processing [cited 2026 Oct 13].

Acknowledgements

- The authors would like to express their sincere appreciation to William Rawlinson (Estima Scientific) for his valuable insights, expert guidance and constructive feedback throughout this research. We also extend our thanks to Mohit Goswami (Estima Scientific) for his time, technical expertise and support in the implementation of the study methodology. Finally, the authors gratefully acknowledge Estima Scientific for providing the time, resources and support necessary to undertake this research.