When LLMs Judge LLMs

Cross-Model Evaluation Bias in LLM-as-Judge Frameworks

A Controlled Experiment Across Five Frontier Models

Tamas Almasi, February 2026

Abstract

As large language models (LLMs) become embedded in professional workflows, a critical yet underexplored question arises: does the choice of model for text generation affect how a different LLM evaluates that output? This paper presents a controlled experiment in which five frontier models — Gemini, ChatGPT, DeepSeek, Claude, and Grok — each generated a structured analytical essay on World War II belligerents, after which the same five models evaluated every response using a standardised scoring rubric. Across 29 full generation-and-evaluation cycles and more than 500 individual scored assessments, clear and reproducible cross-model scoring patterns emerged. ChatGPT and Gemini consistently achieved the highest peer reputations, while Grok exhibited the largest self-assessment gap, scoring its own output 13 percentage points above the peer mean. The findings suggest that model selection for content generation is not a matter of stylistic preference alone — it carries measurable strategic implications when output will be assessed by another LLM downstream.

1. Introduction

Most professionals who rely on large language models for writing — cover letters, executive summaries, technical reports, LinkedIn posts, client emails, academic essays — may already have settled on a preferred model and stopped questioning that choice. The output looks good; it passes the human eye test. Usage is accelerating across every profession.

But here is the question that is rarely asked: **who is reading your output?**

In modern digital pipelines, a human is often not the first—or most consequential—reader. Increasingly, an LLM is the one summarizing or grading the text. If different models evaluate the same text with reproducible, systematic variance, choosing a writing model becomes a strategic decision rather than a stylistic one.`

This paper reports an experiment designed to answer that question.

The short answer is ***yes, it matters — considerably.***

Across 29 evaluation cycles and more than 500 individual scored assessments, clear and reproducible patterns emerged. Certain models consistently earned high scores from their peers, while others were rated measurably lower—not as isolated instances, but as a structural trend observed across every evaluator and run. We found that some models rate their own output significantly higher than their peers do; in one case, the self-assessment gap reached 13 percentage points above the peer mean. These results suggest that such variances are not random fluctuations, but rather stable characteristics of how each model positions itself and others within the evaluation space. These findings carry direct implications for any workflow where the output of one model is assessed by another.

2. Project Overview

This paper describes a project comparing evaluation results produced by different LLMs on texts generated by other LLMs — an LLM-as-Judge framework applied systematically across multiple frontier models. Three experimental rounds were conducted on tasks of varying specificity, from general to highly specialised. What follows is the first and simplest experiment.

Disclaimer: This is a specific experiment with a defined scope — particular models, versions, topic, and prompts. It was conducted to the best of the author's knowledge and abilities, and all methods, prompts, and configurations are fully transparent and available throughout this paper.

☝☝☝ *Practical Recommendation: When working on anything consequential, consider using multiple models in parallel — if the opportunity exists to do so.* ☝☝☝

2.1 Task Design

All selected LLMs were presented with the same historical question. The same group of LLMs then evaluated each other's answers using a standardised evaluation framework. For this initial experiment, the topic of World War II was chosen for methodological reasons: the war is sufficiently distant in time and extensively researched, which implies broad scholarly consensus on major facts, actors, motivations, and consequences.

To simulate realistic professional use, all models were queried via their respective APIs. To maintain parity with standard user experiences (GUI), default model configurations were utilized. No system prompt was utilized during the answer generation phase — only a single user prompt was dispatched to all five models. The full text of the generation prompt and the evaluation prompts are provided in Appendix B and Appendix C respectively.

2.2 Model Selection

The following models were selected and queried directly through their respective vendor APIs:

Model Label	Model String	API Provider
Gemini	gemini-3-flash-preview	Google Gemini API
OpenAI	gpt-5.2	OpenAI API
DeepSeek	deepseek-chat (v3.2)	DeepSeek API
Claude	claude-sonnet-4-20250514	Anthropic API
Grok	grok-4-1-fast-reasoning	xAI API

Technical note on API routing: All five models were called directly through their respective vendor APIs, not through any intermediary or OpenAI-compatible proxy endpoint. Initial routing through OpenRouter was abandoned because it distributes requests across model versions in ways that are not always transparent — for example, a GPT-5.2 call could route to an AWS-hosted instance rather than OpenAI's own infrastructure (verifiable via the system_fingerprint field in the response). Direct API calls ensure reproducibility and vendor consistency.

3. Generation Phase: Observations on the WWII Historical Analyses

3.1 Runtime and Output Size

Even before analysing the content, the variance in response latency and output length was significant. The observed latency reflects API-driven generation under the 1,500-word constraint; notably, these same models typically deliver responses within five seconds when accessed via their standard web interfaces (GUI).

Model Label	Latency via API calls	Output Size	Observations
Gemini	~ 20 sec	~ 1,300 words	Fast and within the requested length constraint.
OpenAI	~ 60 sec	~ 1,700 words	The slowest and longest, exceeding the 1,500-word limit.
DeepSeek	~ 40 sec	~ 1,200 words	DeepSeek and Claude: comparable output profiles.

Claude	~ 40 sec	~ 1,200 words	
Grok	~ 25 sec	~ 970 words	Fast but the most concise response by a significant margin.

3.2 Coverage and Structural Approach

Content-wise, the outputs clustered into two main approaches:

Approach 1 — Five-Core-Powers Framing

Gemini, Grok, DeepSeek, and Claude converged on the canonical five countries — Germany, Japan, the USSR, the United States, and the United Kingdom — delivering a clean, country-by-country structure with explicit treatment of overt versus covert motivations and both short- and long-term outcomes.

Approach 2 — Broader Country Set

ChatGPT expanded beyond the five core powers to explicitly include China, France, Italy, and a grouped section covering Canada, Australia, and India. This increased historical breadth and factual completeness but introduced two practical trade-offs:

- word-limit non-compliance (which occurred here), and
- a greater surface area for evaluator disagreement, given that no other model shared this framing.

However, *ChatGPT's* comprehensive approach appears to have been viewed positively by the other models in their role as evaluators, as *ChatGPT* emerged as the most respected model across the entire peer group.

3.3 Structural and Presentation Choices

All five outputs were clearly structured, with coherent narratives and an explicit attempt to contrast wartime intentions against actual outcomes. Several model-specific presentation choices stood out as likely to influence evaluator perception:

- *Gemini* added a comparative summary table ("Ambition vs. Reality"), mapping goals to outcomes per country — a device that forces structured consistency and is highly legible to automated evaluators.
- *Grok* appended a self-declared word count ("Word count: 1,248"), functioning as a compliance signal — though the reported figure did not match a programmatic token count.

- *DeepSeek* produced a tight, readable structure with a strong concluding synthesis on the "chasm between aims and outcomes."
- *ChatGPT* delivered the broadest coverage and the most textbook-like completeness, but at the cost of exceeding the word constraint.
- *Claude*: Prioritized stylistic sophistication and intellectual depth over mechanical checklists.

Reading these outputs side by side generated the key insight of the experiment: the differences between responses are not simply "good vs. bad." They reflect a kind of model personality — *Grok's* functional brevity, *ChatGPT*'s encyclopaedic coverage, *Gemini's* structured elegance, *DeepSeek's* analytical clarity and *Claude's* intellectual depth.

Almost every response feels high quality in isolation; it is only in comparison that consistent model-specific preferences in scope management, structural choices, and compliance behaviour become visible.

4. The Evaluation Framework

4.1 Setup

Once the five generated texts were collected, the same five models were used as evaluators — simulating a recursive "LLM-as-Judge" scenario. To prevent self-recognition bias, the five answers were anonymised as Answer A through Answer E before being passed to the evaluators. The evaluation prompts (system and user) are provided in Appendix C.

The JSON output format was requested to make further analysis straightforward and to enforce structured, comparable output across all evaluator models.

4.2 Technical Challenges

Implementing this pipeline required resolving several model-specific technical constraints.

JSON Output Reliability

Although evaluator models were explicitly instructed to return strictly valid JSON, outputs occasionally included Markdown code fences, extraneous text, or minor syntax errors, rendering them incompatible with Python's standard json.loads() parser. Despite extensive prompt engineering efforts to enforce clean output, *DeepSeek* proved particularly resistant to this constraint. To resolve this, a dedicated JSON sanitization and syntax-repair function was implemented to pre-process raw API returns before analysis.

Gemini Configuration Parameter Handling

Unlike the other APIs in this study, the *Gemini* Python SDK utilizing a dedicated configuration object (types.GenerateContentConfig) for passing system prompts and generation parameters, rather than including them directly in the request call or message structure. Setting max_tokens and temperature via this configuration object often led to runtime instability, characterized by significant latency and occasional request stalls. After several attempts, the temperature and token limit parameters were removed from *Gemini* evaluator calls, retaining only the system prompt. This produced stable and consistently reliable execution.

Temperature Parameter Consistency

A temperature setting of 0.0 was used for all evaluator calls to maximise reproducibility, with the exception of *Gemini*, due to the configuration instability described above. Whether frontier models truly enforce a strict temperature of 0.0 is an open question; however, running the full evaluation pipeline 29 times produced stable average results, providing a reliable empirical basis for comparison.

Token Limit Parameter Handling

Token limit handling differed across model APIs. *Claude* is the only model for which max_tokens is a required parameter. *Grok* frequently returned empty outputs with no error when this parameter was absent. In contrast, including max_tokens caused instability in *Gemini*, so it was excluded from *Gemini* calls. A value of max_tokens=16,384 was used for *Grok* and *Claude* only. This was necessary to accommodate the large evaluation prompt - which included all five model responses - totalling approximately *10,000–12,000 tokens*.

Evaluation Runtime

Evaluating all five answers in a single evaluator call took approximately 18–25 seconds across most models, with *Gemini* taking around 45 seconds. Notably, several models completed the evaluation pass faster than their initial answer generation, despite receiving a substantially larger input.

Important Implementation Note: It is critical to emphasize that this observation applies specifically to API-based execution. This suggests the possibility in case of API calls that discriminative tasks - such as evaluating existing text against a rubric - may place different computational demands on a model than generative tasks, which require the synthesis of a novel, knowledge-grounded response from scratch. Unlike customer-facing GUIs, which may use streaming or background optimizations, ***these raw API runtimes may offer a more direct insight into the models' internal processing efficiency for different task types****.*

5. Evaluation Phase – Results

5.1 Cross-Model Scoring Matrix

The heatmap below shows the full cross-model scoring matrix: how each model scored every other model, including itself, aggregated across all 29 evaluation runs.

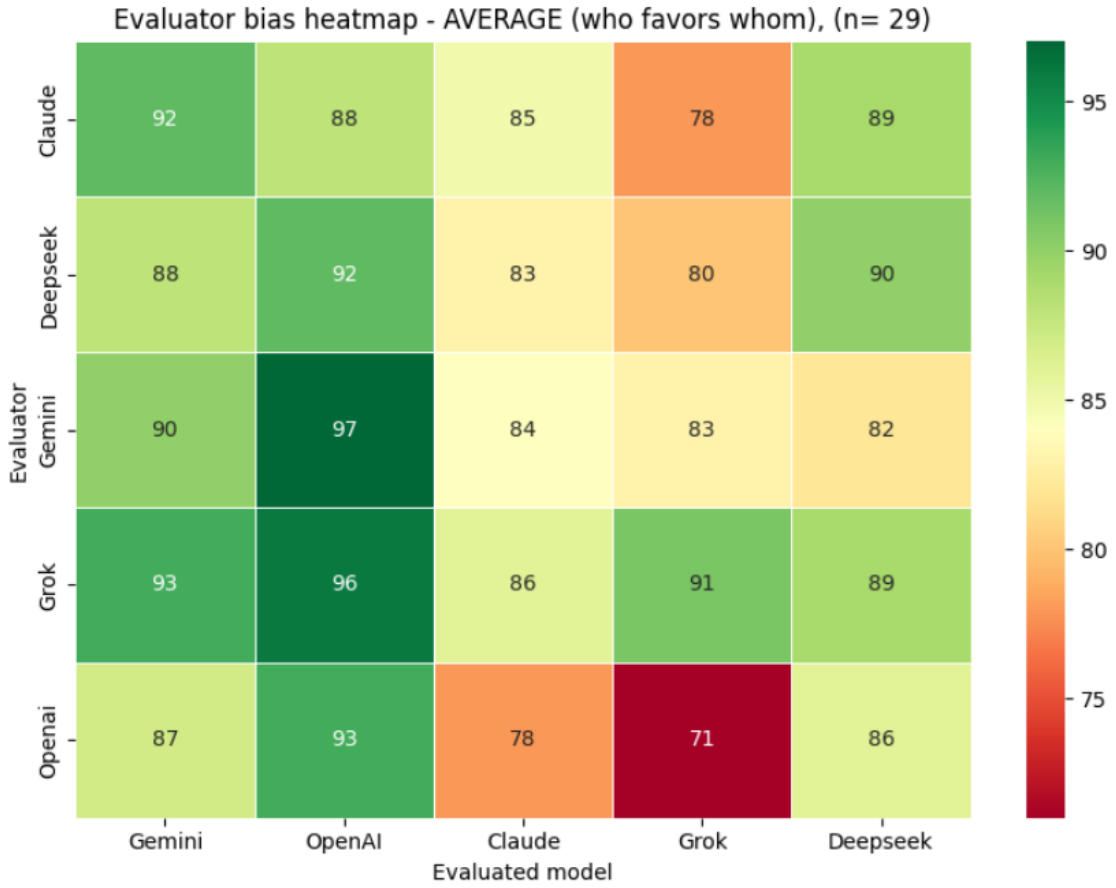

Figure 1: Cross-model scoring matrix — which model favours which model.

Across all 29 cycles, aggregate scores are high overall, with *ChatGPT* and *Gemini* emerging as the two consistently top-rated models by their peers.

Authors using *Grok* or *Claude* for text generation should be aware that peer models score their output measurably lower on average — a structural pattern, not a statistical artefact.

5.2 Evaluator Strictness

The next question is whether the models differ systematically in how strict or lenient they are as evaluators.

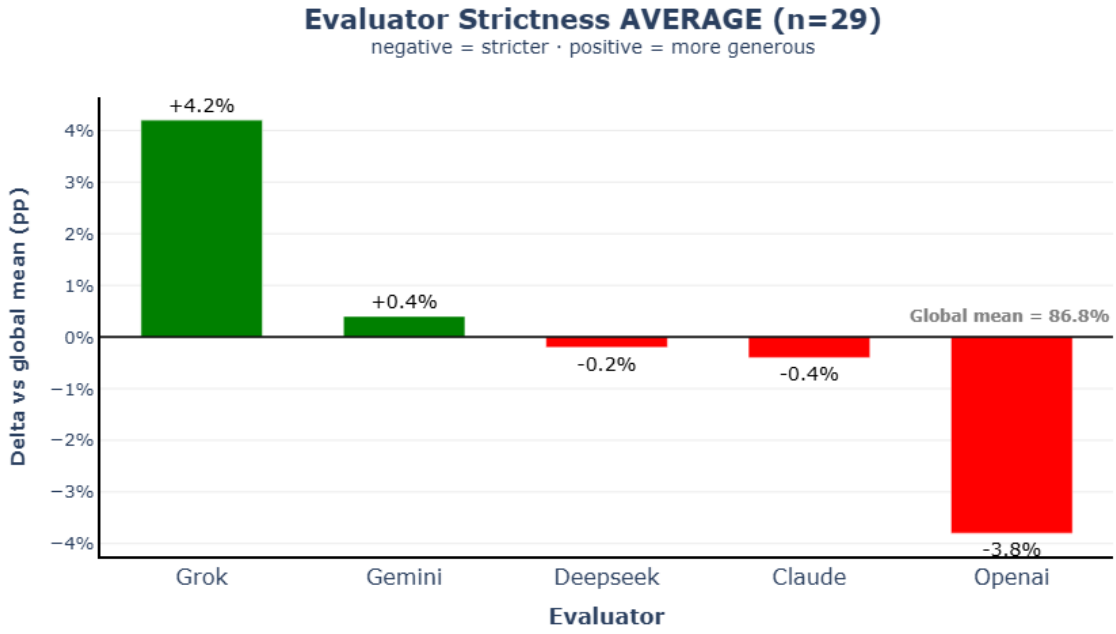

Figure 2: Evaluator generosity — how strict or lenient each model was when scoring the others.

Figure 2 illustrates evaluator generosity relative to the global mean of 86.8% across all 29 runs. *Grok* sits slightly above this mean as the most generous evaluator, while *ChatGPT* is the most demanding — consistently applying the strictest scoring standards across all five answers. The spread between the most and least generous evaluator is approximately 8 percentage points, which is large enough to meaningfully affect comparative rankings.

5.3 Model Reputation

The following figure addresses the question most relevant to practitioners: *which model should be used if the evaluator model is unknown?*

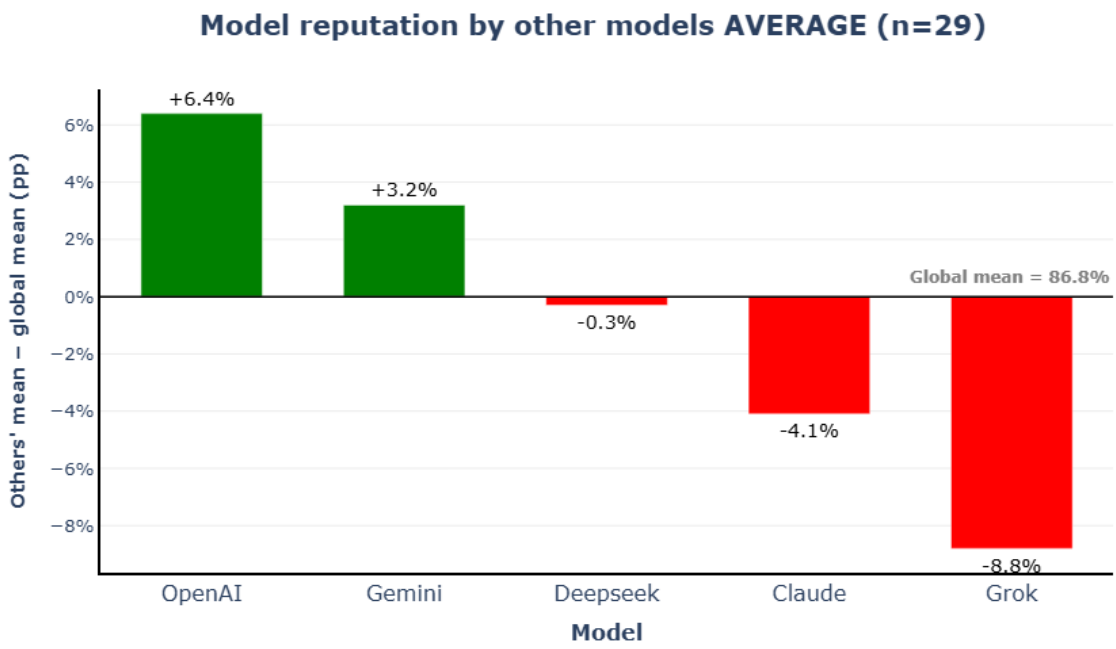

Figure 3: Model reputation — how each model is perceived on average by its peers (excluding self-assessment).

Figure 3 shows each model's mean peer score — what might be called its evaluation reputation. *ChatGPT* scores highest, with *Gemini* close behind. *Claude* and *Grok* sit at the lower end, with a roughly 15 percentage point spread between the top and bottom of the distribution. This gap is large enough to be practically relevant for anyone selecting a model for text generation, especially when the output is expected to be assessed by another LLM.

5.4 Self-Assessment vs. Peer Assessment

A final dimension of this study is the gap between how each model rates its own output and how its peers rate the same output.

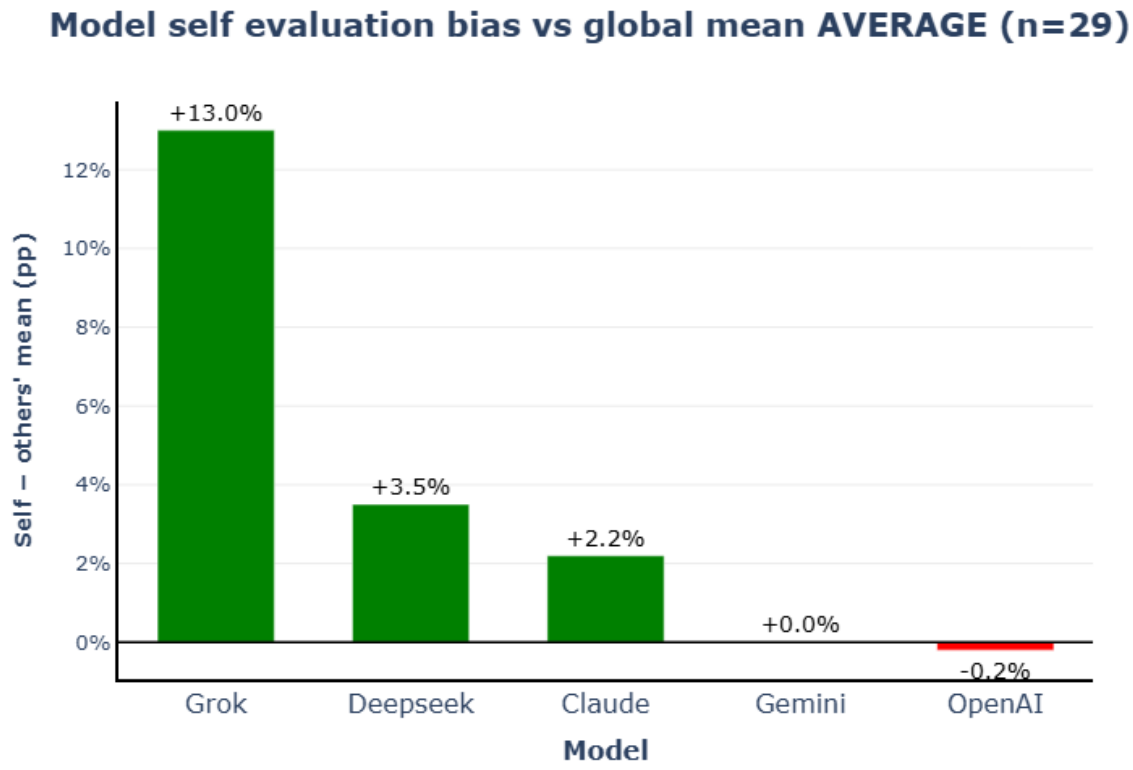

Figure 4: Model self-confidence versus peer assessment — self-score minus peer mean.

Figure 4 contrasts each model's self-assessment against the mean score assigned by the other four models.

- *Grok* shows the largest positive gap — a pattern consistent with systematic overconfidence. Again, anyone using *Grok* for text generation should be mindful of its high and ungrounded self-confidence, which may lead the model to overestimate the objective quality or accuracy of its own output.
- *Gemini* and *ChatGPT* show well-calibrated self-assessments, closely aligned with peer opinion.
- *Claude* and *DeepSeek*, meanwhile, score themselves with a relatively healthy self-confidence.

5.5 Consolidated View

Figure 5 consolidates the preceding analyses into a single comprehensive view: per-model evaluation bands, peer mean scores, and self-evaluation scores presented side by side.

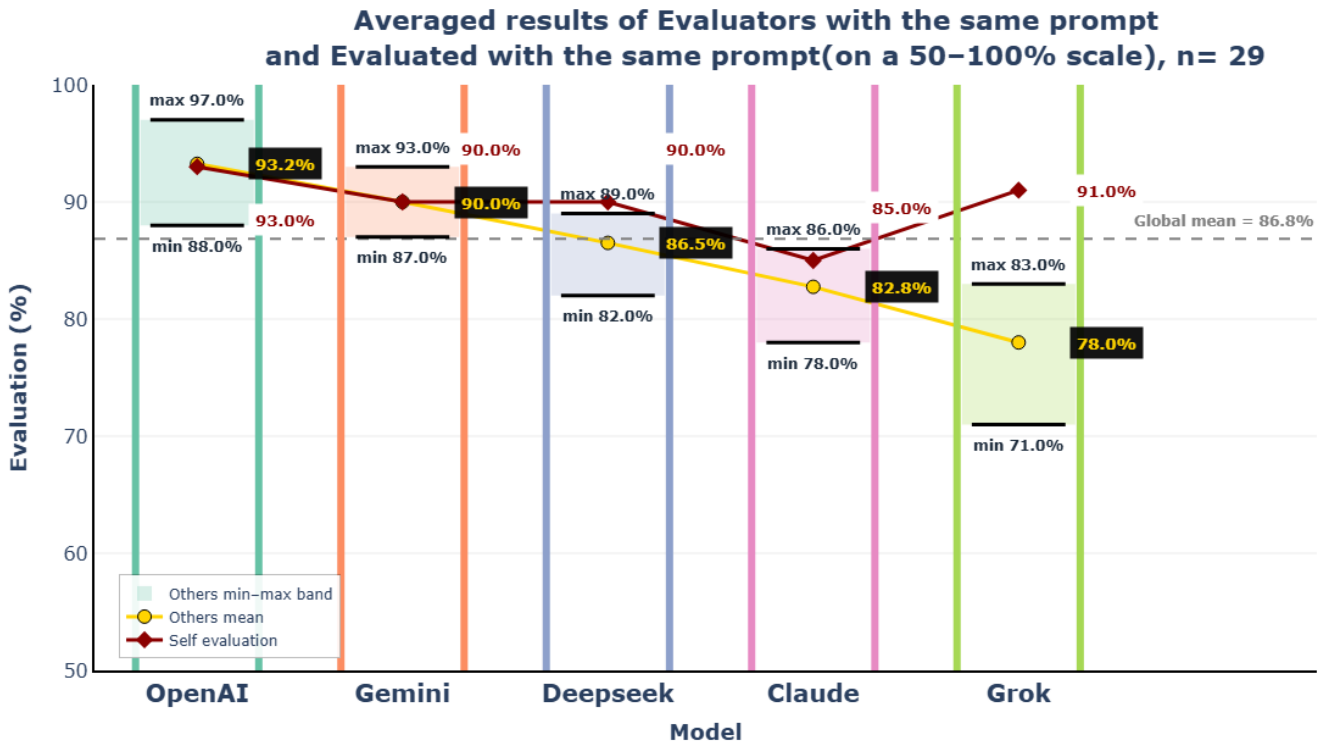

Figure 5: Consolidated view — per-model evaluation bands, peer mean scores, and self-evaluation scores.

Figure 5 illustrates that the combination of high peer reputation and well-calibrated self-assessment characterising *ChatGPT* and *Gemini* makes their output the most likely to be well-received by LLM evaluators across the widest range of possible models.

6. Conclusions

The central finding of this experiment is straightforward: the choice of model for text generation is ***not neutral*** when the output will be assessed by another LLM. The *15-percentage point spread in peer reputation* between the highest- and lowest-rated models is large enough to have *practical consequences* — particularly in automated pipelines where LLM-generated content is screened, ranked, or evaluated without direct human review.

Among the models tested, ***ChatGPT*** achieved the ***highest*** and most consistent peer reputation, combined with a well-calibrated self-assessment. ***Gemini*** is a ***close second***, and the combination of high reputation and accurate self-knowledge makes both models the most strategically sound choice for generation tasks in LLM-as-Judge contexts. ***Grok's*** systematic ***overconfidence*** and Claude's systematic tendency toward self-underrating are both noteworthy calibration patterns, regardless of their absolute quality levels.

Important caveat: This experiment used a specific set of models, model versions, a single topic domain, and a specific prompt design. Results may not generalise directly to other task types, subject areas, or future model versions. The findings should be interpreted as evidence

of a real and reproducible phenomenon — cross-model evaluation bias — rather than a definitive ranking of model quality in the general sense.

Future Research

Future experiments in this series will apply the same framework to more specialized, expert-level knowledge and technically demanding prompts. It will be interesting to observe whether the gap between models narrows or widens when additional expert knowledge is required, and how these varying depths of knowledge ultimately influence evaluation outcomes.

Appendix A — Qualitative Evaluation Analysis

This appendix provides a detailed summary of the qualitative feedback patterns observed across all evaluators and all 29 runs.

A.1 Common Evaluative Strengths

Coverage and Comprehensiveness

Coverage was the single most rewarded quality across all evaluators. Answers that covered not just the canonical five belligerents (Germany, Japan, USSR, USA, UK) but also Italy, China, and France were praised almost universally. OpenAI (Answer B) was singled out most frequently, with phrases such as "most comprehensive coverage, including all required belligerents" appearing in nearly every evaluator's feedback across all 29 runs. DeepSeek was a close second in completeness scores.

Overt vs. Hidden Motivation Distinction

The analytical distinction between official war aims and implicit underlying drivers was consistently the most differentiating quality factor. Every evaluator, regardless of model, rewarded answers that went beyond publicly stated objectives to identify covert motivations: the US goal of dismantling British imperial trade preferences and embedding American economic norms in the Bretton Woods system; the Nazi economy's structural dependence on territorial plunder to avoid domestic inflation; Soviet territorial expansionism framed as a security requirement rather than ideological ambition; and the UK's balance-of-power logic rather than straightforward empire preservation.

Specific Historical Terminology

The use of precise historical terminology was rewarded consistently across all evaluators. Frequently praised terms included: Lebensraum, Wirtschaftswunder, Vergangenheitsbewältigung, Generalplan Ost, Yoshida Doctrine, Stunde Null, autarky, Pax Americana, Dodge Line, Oder-Neisse line, and the Molotov-Ribbentrop Pact. Answers employing these terms were consistently described as "scholarly," "academically rigorous," or possessing "appropriate depth."

Synthesis and Irony Analysis

The observation that defeated Axis powers — Germany and Japan in particular — ultimately achieved greater long-term prosperity and geopolitical influence through peaceful means than their wartime aims could ever have delivered was highlighted as a mark of analytical sophistication. Gemini (Answer A) and DeepSeek (Answer E) were most often credited for this thematic depth.

Contrast Sections

Dedicated goal-versus-outcome paragraphs per country were praised especially by OpenAI and DeepSeek evaluators, who treated their structural presence as a quality signal in itself.

A.2 Common Evaluative Weaknesses

Italy Omission

Italy's omission was the single most universal weakness flagged across all models, all evaluators, and all 29 runs. It functioned as an automatic score deduction. Gemini, Claude, and DeepSeek were most frequently penalised for this. The feedback was remarkably consistent: "omits Italy despite it being a major belligerent."

China Omission

China's omission was the second most common complaint, with evaluators noting that China's role in tying down large Japanese forces represented a strategic contribution that could not be ignored in a comprehensive treatment. Claude was most commonly identified as omitting both Italy and China simultaneously.

Imprecise or Unsupported Quantitative Claims

Imprecise quantitative assertions were flagged across all answers, but most intensively for Grok (Answer D). Recurring examples included:

- The USSR being credited with "80% of German casualties" — cited as debatable or unqualified.
- Japan becoming "the world's second-largest economy by the 1960s" — flagged as potentially inaccurate or overstated.
- Germany losing "25% of pre-war territory" — noted as imprecise depending on the territorial baseline used.
- Germany's "industrial capacity reduced by 75%" — flagged as unsubstantiated.
- Post-1945 Germany described as experiencing "hyperinflation" — a factual error, as Germany's hyperinflation occurred in 1923, not after World War II.

Hidden Motivations Underdeveloped

Claude (Answer C) and DeepSeek (Answer E) were both specifically critiqued for providing thinner treatment of implicit and covert motivations relative to the quality of their coverage of overt aims.

Word Limit Violations

Word limit violations were noted repeatedly for GPT (Answer B), which was frequently described as "significantly exceeding the 1,500-word limit" while paradoxically still receiving strong structural scores — suggesting that evaluator models do not automatically penalise length violations as heavily as the scoring rubric implies.

Appendix B — Generation Prompt

The following single user prompt was sent to all five models during the generation phase. No system prompt was used.

```
Provide a comprehensive and well-structured analysis of the most significant countries
worldwide that actively participated in World War II. For each major nation, please address the following points:

Role and Contribution: Describe the country's role in the war (as an aggressor, defender, or ally) and its specific contributions to the conflict.

Strategic Objectives and Motivations: Outline the official strategic and political goals. Critically examine both the overt (publicly stated) and the hidden or less explicit motivations (e.g., territorial expansion, ideological dominance, economic interests, domestic power consolidation, or geopolitical shifts).

Short-term Outcomes (1945-1950): Analyze what the country gained or lost in the immediate post-war period, including territorial changes, political control, and immediate economic conditions.

Long-term Consequences: Evaluate the lasting impact in the subsequent decades, focusing on geopolitical influence, long-term economic development, and societal changes.

Requirements:
- Focus: Highlight the contrast between the initial war goals (explicit and implicit) and the actual outcomes.
- Quality: Provide a balanced, fact-based, and high-quality academic analysis that avoids oversimplification.
- Length: Maximum 1500 words.
- Language: High-quality English with a logical flow and clear structure.
```

Appendix C — Evaluation Prompts

C.1 System Prompt (Evaluator Role)

The following system prompt was sent to all five models in their evaluator role:

```
You are an independent, neutral, and objective expert evaluator with deep academic
knowledge of 20th-century history, especially World War II, geopolitics, and
international relations.

TASK:
Evaluate how well each provided answer executes the specific task, using your expert
knowledge of history. Compare each answer against established historical facts and
scholarly consensus.

IMPORTANT RULES:
- Evaluate ONLY what is explicitly written in each answer.
- Do NOT add missing facts, context, or explanations (do not mentally fill in gaps).
- Do NOT correct, improve, or rewrite any answer.
- Do NOT speculate about what the author might have intended.
- Identify factual errors, omissions, inaccuracies, or unsupported claims.

EVALUATION CRITERIA:
1. Historical Accuracy: Are facts, dates, events, and causality correct?
2. Completeness: Does it cover all major parts (roles, aggressor status, objectives,
   hidden motivations, short-term and long-term outcomes)? Does it include major
   belligerents?
3. Quality & Depth: Is the explanation coherent, logically consistent, and
   well-structured? Does it meaningfully distinguish between stated (overt) goals
   and hidden (covert) agendas?
4. Clarity: Is the text understandable and written in high-quality, clear English?
5. Word Limit Compliance: Does it stay within the 1500-word limit while maintaining
   high quality?

SCORING:
- Assign a percentage score (0-100) to each answer.
- 100% = fully accurate, coherent, complete, covers all overt/covert motivations,
  and follows all requirements.
- 0% = largely incorrect, misleading, incoherent, or severely incomplete.

OUTPUT FORMAT (STRICT):
Return ONLY a valid JSON object. No text before or after the JSON.

JSON RULES:
```

```
- Use double quotes (") for strings.
- Percent values must be numeric (no % symbol).
- Ensure valid, parseable JSON.
- Do NOT use quotation marks inside strings.
- Rephrase instead.

{
  "results": [
    {
      "answer_id": "A",
      "score_percent": 85,
      "strengths": ["string1", "string2"],
      "weaknesses": ["string1", "string2"]
    }
  ]
}
```

C.2 User Prompt (Evaluator Role)

The following user prompt was sent to each evaluator model. The placeholder text {answer_X} was replaced with the actual anonymised model responses (A = Gemini, B = OpenAI, C = Claude, D = Grok, E = DeepSeek):

```
Evaluate the following answers provided in response to the original task question.
Analyze each answer independently according to the evaluation criteria and scoring
rules defined in the system instructions.

ORIGINAL TASK QUESTION:
"Provide a comprehensive analysis of the most significant countries that actively
participated in World War II. For each major nation, explain:
1) Their role and contribution to the war (as aggressor, defender, or ally),
2) Their strategic objectives and what they aimed to achieve - including both
   publicly stated goals and hidden/underlying agendas (territorial expansion,
   ideological dominance, economic interests, geopolitical power shifts),
3) What they gained or lost in the short term (immediate post-war period, 1945-
1950),
4) What they gained or lost in the long term (subsequent decades, geopolitical
   influence, economic impact). Focus on the primary belligerent nations and
organize
   the analysis by country, highlighting the contrast between their initial war
goals
   and actual outcomes. Requirements: Maximum 1500 words, high-quality analysis."

ANSWERS TO EVALUATE:
<<<
Answer A: {gemini_response}
```

```
Answer B: {openai_response}
Answer C: {claude_response}
Answer D: {grok_response}
Answer E: {deepseek_response}
>>>

Return your evaluation in the STRICT JSON format specified in the system prompt,
listing numeric scores, strengths, and weaknesses for each answer.
```