SPREADSHEETARENA: DECOMPOSING PREFERENCE IN LLM GENERATION OF SPREADSHEET WORKBOOKS

Srivatsa Kundurthy*
Longitude Labs, Cornell University

Clara Na*
Carnegie Mellon University

Michael Handley **Zach Kirshner**
Longitude Labs

Chen Bo Calvin Zhang **Manasi Sharma**
Scale AI

Emma Strubell
Carnegie Mellon University

John Ling
Longitude Labs

ABSTRACT

Large language models (LLMs) are increasingly tasked with producing and manipulating structured artifacts. We consider the task of end-to-end **spreadsheet generation**, where language models are prompted to produce spreadsheet artifacts to satisfy users' explicit and implicit constraints, specified in natural language. We introduce SPREADSHEETARENA, a platform for evaluating models' performance on the task via blind pairwise preference votes of LLM-generated spreadsheet workbooks. As with other complex, open-ended tasks, relevant evaluation criteria can vary substantially across use cases and prompts, often in ways that are difficult to formalize. Compared to general chat or text generation settings, spreadsheet generation presents unique challenges and opportunities: the task output structure is well-defined and multi-dimensional, and there are often complex considerations around interactivity and layout. Among other findings, we observe that stylistic, structural, and functional features of preferred spreadsheets vary substantially across use cases, and expert evaluations of spreadsheets for finance prompts suggests that even highly ranked arena models do not reliably produce spreadsheets aligned with domain-specific best practices. The pairwise preference data we collect is structurally identical to supervision used in preference-based post-training and RLHF, and our analyses surface data quality considerations, including feature importance and domain-dependent preference drivers, relevant to curating such data for structured generation tasks. We release a dataset of prompts, generated spreadsheets, and preference votes upon publication, which we hope will facilitate further study of tasks operating over spreadsheets as a challenging and interesting class of complex, open-ended tasks for LLMs. Our live arena is hosted at `https://spreadsheetarena.ai`.

1 INTRODUCTION

Tasks that involve the production or manipulation of structured artifacts are a natural fit for automation with large language models (LLMs), including code generation (Chen et al., 2021a; Rozière et al., 2024), table generation and representation (Zhang et al., 2024; Tang et al., 2024), text-to-SQL (Yu et al., 2018; Lei et al., 2025), and spreadsheet formula generation (Chen et al., 2021b; Zhao et al., 2024). In some cases, successful task completion can be evaluated through programmatic verification of the outputs. However, many tasks of significant practical value to human users are inherently more open-ended, admitting multiple valid solutions and involving objective and subjective evaluation criteria that may differ across use cases and users. While LLMs are often capable of performing these tasks, evaluation of their capabilities remains a challenge.

*Equal contribution. Corresponding author: `srivatsa@meridian.ai`

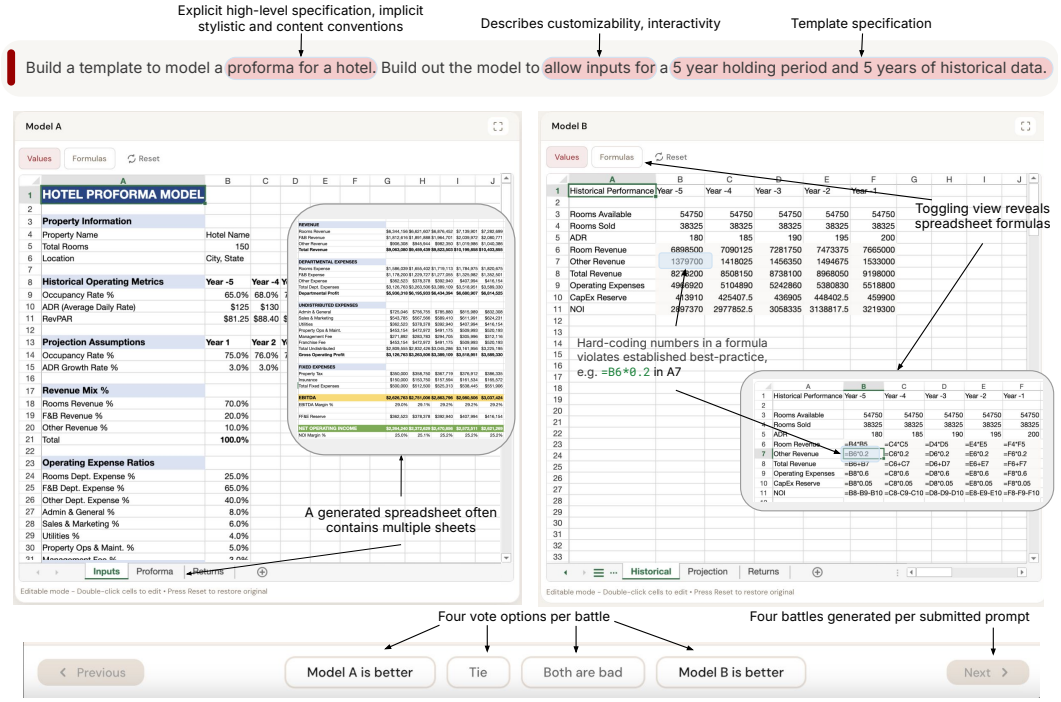


Figure 1: In SPREADSHEETARENA, users submit a prompt and are shown four pairwise battles between LLM-generated spreadsheet workbooks. Votes are blind, and users can indicate that one spreadsheet is preferred over the other, or that both are equally satisfactory or unsatisfactory. Workbooks can contain multiple sheets, and sheets often contain a mixture of text, values, and formulas, where cells may contain stylistic formatting (e.g. bold text or a fill color).

We consider *end-to-end* **spreadsheet generation** as a task for LLMs, where models are prompted to generate spreadsheet artifacts according to explicit and implicit natural language specifications. Use cases for spreadsheet generation span a variety of domains, such as professional finance (e.g., comparing risk across potential investments), academic research (e.g., setting up a statistical significance test given experimental results), and even creative uses (e.g., "Color in cells to look like Mario"). Criteria for a useful, high-quality spreadsheet workbook output can depend on explicit and implicit contextual factors. One prompt may call for strict adherence to instructions spanning both content and formatting, while another may call for only a template that can be updated by the user. Even given a prompt, evaluations may emphasize different criteria, such as correctness of formulas, adherence to domain-specific formatting conventions, or other readability or usability constraints.

Compared to both (1) open-ended dialogue benchmarks and (2) established tasks involving structured artifact generation, the evaluation of spreadsheet generation presents distinct challenges. Expected outputs are structured artifacts that encode dense, graph-structured dependencies across spreadsheet cells and formulas, exceeding the structural complexity typically seen in open-ended dialogue and even in other commonly studied artifacts such as JSON objects (Geng et al., 2025). Moreover, considerations around user interactivity in spreadsheet workbooks can render errors non-obvious (Panko & Aurigemma, 2010) and simple execution-based validation insufficient, whereas single-pass execution is common in evaluation of coding tasks (Chen et al., 2021a; Hendrycks et al., 2021).

We show that spreadsheet generation is a challenging task warranting further study; performant LLMs produce well-formed spreadsheet workbooks with valid formulas more often than not, but practical *functional* utility and adherence to stylistic guidelines, when applicable, are much less reliable. Since successful task completion in spreadsheet generation is inherently high-dimensional and context-dependent, human user preference evaluation is a critical component of capability assessment. Towards this, we introduce SPREADSHEETARENA, a platform for arena-style evaluations of LLM-produced spreadsheet workbooks. We collect user votes over 4357 pairwise battles between anonymized models' spreadsheet outputs, across both admin-curated and submitted prompts spanning

a variety of use cases and domains, with a particularly significant representation in the finance domain. We establish a stable ranking of 16 models across multiple model families.

Moreover, characteristics of winning spreadsheets vary across use cases, and controlling for measurable spreadsheet features, such as diversity in formatting, number of filled cells, number of sheets in a notebook, and number of formulas, influence model rankings.

In spreadsheets in the financial modeling domain, we contextualize our analyses of preference evaluations with established best practices such as color coding standards, the "one row, one formula" rule (FAST Standard Organization, 2015; Wall Street Prep, 2020), and expert evaluations of adherence to finance modeling conventions. We find that LLM-generated spreadsheets generally demonstrate poor alignment with financial modeling conventions, though models have distinct tendencies and some align more closely with conventions than others.

Our findings also connect to the study of preference data quality for post-training. Just as response length has been shown to influence text preference evaluations (Hu et al., 2025), we find that certain measurable spreadsheet features achieve varying significance in influencing model rankings, and that the features that achieve significance vary across domains. These findings carry implications for preference-based post-training methods applied to structured generation tasks, where models must simultaneously satisfy functional, structural, and domain-specific criteria that naive preference data does not uniformly reward.

We summarize our core contributions: (1) We introduce SPREADSHEETARENA, a platform for evaluating **end-to-end spreadsheet generation** via blind preference evaluations of spreadsheet workbooks produced by LLMs for user-submitted prompts. The arena is live at `https://spreadsheetarena.ai` and contains 4,357 user votes over pairwise battles.[1] (2) We establish a stable ranking of 16 LLMs across multiple model families, noting distinct tendencies and capabilities across model families. Moreover, we conduct extensive analyses of fine-grained spreadsheet characteristics in conjunction with preference rankings collected from our arena, finding, for example, that relevant evaluation criteria can differ dramatically between types of use cases.

We release a dataset of prompts, spreadsheets, and preference votes for analysis and further study. Code and data can be found at `https://github.com/Longitude-Labs/meridian-ssa-public`.

2 RELATED WORK

Human Preferences. Human preference is central to both post-training (Christiano et al., 2017; Ouyang et al., 2022; Rafailov et al., 2023) and evaluation of large language models. Preference-based "arenas" collect head-to-head comparisons and aggregate them into rankings via Bradley-Terry or Elo-style estimators (Bradley & Terry, 1952; Coulom, 2007). LMArena (Chiang et al., 2024) pioneered blind, community-driven head-to-head comparisons for text generation, while SEAL Showdown (Scale AI, 2025) highlights how preference signals might be confounded by factors such as verbosity or formatting, and motivates analyses that disentangle form from perceived quality (Cai et al., 2025). Arena-style evaluation is increasingly applied beyond conversation to agentic settings in works such as Remote Labor Index (RLI) (Mazeika et al., 2025). Our work evaluates *end-to-end spreadsheet workbook generation*, where user preference reflects functional correctness, formatting, and additional entangled factors, such as vertical-specific style, presented in §5.

Furthermore, human preference data can also be utilized for evaluations. Preference-based "arenas" collect head-to-head comparisons and aggregate them into model rankings, using Bradley-Terry or Elo-style estimators (Bradley & Terry, 1952; Coulom, 2007). Chiang et al. (2024) introduce LMArena, which uses blind, community-driven, head-to-head comparisons to produce model rankings. Related efforts, such as the SEAL Showdown (Scale AI, 2025), further emphasize that preference signals might be confounded by factors such as verbosity or formatting, and motivate analyses that disentangle form from perceived quality (Cai et al., 2025). While most arena-style evaluations have focused on conversational settings, this type of evaluation is increasingly being used in more agentic and tool-using tasks. For example, the Remote Labor Index (RLI) measures the performance of agents on real-world remote-work tasks (Mazeika et al., 2025).

[1] As of January 28th, 2026

Additionally, while rubrics have been found to be useful in conversational settings (Lin et al., 2024; Arora et al., 2025; Akyürek et al., 2025), it has been observed in agentic settings that granular per-project rubrics are often insufficient to capture project completion, and for artifacts with hard-to-specify aspects (e.g., design), a deliverable might technically satisfy rubric elements yet fail professional standards (Mazeika et al., 2025). Our work evaluates *end-to-end spreadsheet workbook generation*, where user preference reflects functional correctness, formatting, and additional entangled factors, such as vertical-specific style, presented in §5.

Finance task evaluation. A growing body of work develops language models trained on finance data (Wu et al., 2023; Yang et al., 2025) and benchmarks that evaluate them on tasks in the finance domain (Xie et al., 2023; 2024; Zhu et al., 2021; Chen et al., 2022). In parallel, research on spreadsheet-oriented tasks has addressed table detection and compression (Dong et al., 2025), formula prediction (Chen et al., 2021b; Zhao et al., 2024), and spreadsheet manipulation (Ma et al., 2024). Neither line of work engages with holistic, preference-based evaluation that captures the multi-dimensional quality considerations, such as functional correctness, structural organization, and adherence to domain-specific conventions (e.g. financial color coding standards (FAST Standard Organization, 2015)), that simultaneously inform real-world financial spreadsheet utility. Our work addresses this gap: SPREADSHEETARENA evaluates end-to-end workbook generation through arena-style preference votes complemented by feature decomposition and domain expert evaluation, revealing that general user preferences and expert judgments can substantially diverge.

Structured Artifact Generation. Many important tasks require the production or manipulation of structured artifacts. Code generation is the most well-studied such task, as it promises software and AI automation. LLMs are often specifically trained to generate and reason over code (Chen et al., 2021a; Rozière et al., 2024), LLM training corpora often feature carefully curated subsets of code repositories, and code generation benchmarks are popular for evaluating LLM capabilities (Hendrycks et al., 2021; Chen et al., 2021a; Jimenez et al., 2024; Deng et al., 2025). Tabular output and schema-constrained generation have also been studied. Zhang et al. (2024) proposes TableInstruct, a dataset of tables and tasks for instruction fine-tuning, and TableLlama, an open-source model fine-tuned on the former. Benchmarks such as StructBench (Gu et al., 2024) and JSONSchemaBench (Geng et al., 2025), push the evaluation of structured generation capabilities further.

Spreadsheets share some key properties with these tasks, but they introduce additional challenges for evaluation. First, spreadsheets encode graph-structured dependencies between cells and formulas. Second, spreadsheet quality is inherently multi-dimensional in many professional use cases: users care not only about numerical correctness, but also about the layout, the readability, and application-specific conventions. Existing spreadsheet benchmarks such as SpreadsheetBench (Ma et al., 2024), SheetCopilotBenchmark (Li et al., 2023), and SheetRM (Chen et al., 2025) assume spreadsheet manipulation tasks with gold answers that tend to be operationally defined, in terms of specific spreadsheet functions, objects (such as pivot tables), and even cell references. In contrast, SPREADSHEETARENA focuses on *end-to-end synthesis of full spreadsheet workbooks* (potentially with multiple sheets and formatting considerations) from tasks that are often *declaratively* defined in terms of end use case and content; accordingly, SPREADSHEETARENA uses arena-style preference evaluation to capture holistic utility to complement purely programmatic metrics.

3 BACKGROUND

As noted in Chiang et al. (2024), methods to compute rankings from pairwise comparisons are well-studied in the literature. We base our approach on the grounding provided by Chiang et al. (2024) and Scale AI (2025) and apply the Bradley-Terry model (Bradley & Terry, 1952) to estimate strength coefficients, from which we derive rankings. The Bradley-Terry model expresses the probability that model A beats model B in a match-up as $P(A \succ B) = \sigma\left(\theta_A - \theta_B\right)$, where $\sigma(\cdot)$ is the logistic function, $\sigma(x) = \frac{1}{1+e^{-x}}$. The coefficients θ comprise the Bradley-Terry (BT) strength coefficients, and are estimated via maximum likelihood estimation (MLE) to minimize the cross-entropy loss between estimated win probabilities and observed vote outcomes: $\hat{\theta} = \arg\min_\theta -\frac{1}{N}\sum_i \Big[y_i \log \sigma(\Delta_i) + (1-y_i)\log\left(1-\sigma(\Delta_i)\right) \Big]$, where $\Delta_i = \theta_{i_A} - \theta_{i_B}$ is the difference in BT coefficients between the two competing models i_A and i_B in battle i, and $y_i = 1$ if model i_A

won and 0 if i_B won. The resulting BT coefficients, when ordered, produce rankings that reflect relative average win probability.

We obtain Elo-like ratings from BT strength coefficients. Elo and the Bradley-Terry model parameterize pairwise win probabilities as log-odds that are equivalent up to a constant scale factor. For interpretability, we follow Scale AI (2025) and Coulom (2007) to convert the BT coefficients into Elo-like ratings. Without loss of generality, due to the under-specification of the Bradley-Terry model as noted in Cattelan (2012), we specify an anchor model m_0 for which we set $\theta_{m_0} = 1000$. We choose as our anchor GPT-4o, which emerged as the weakest closed model that consistently produces spec-adhering outputs for spreadsheet generation.

4 SPREADSHEETARENA

In this section, we introduce SPREADSHEETARENA for the human evaluation of LLM-produced spreadsheets, motivating the arena in the context of the task and describing our methodology.

4.1 TASK FORMULATION

In this paper, we study a problem we refer to as spreadsheet generation. In spreadsheet generation, a language model is provided a natural-language text prompt and must produce a spreadsheet artifact. The spreadsheet artifact must be syntactically valid, but beyond syntactic correctness, voting patterns may or may not align with established domain-specific best practices or conventions when applicable.

Spreadsheets occupy a unique position in the landscape of structured artifact generation. Estimates of the global software developer population range from 27 million (professional developers) to 47 million (including students and hobbyists), depending on methodology.[2][3] By contrast, Bloomberg estimates that in 2025, there were 500 million paying Excel users,[4] many of whom may not identify as programmers yet routinely build and maintain computationally intensive workbooks. The scale and heterogeneity of spreadsheet users presents distinct evaluation challenges: criteria for a useful, high-quality spreadsheet can depend heavily on explicit and implicit contextual factors that vary across domains, workflows, and user intent.

Although spreadsheet generation is a distinct problem with a bounded scope compared to open-domain chat settings where arena-style evaluations have previously been studied (Chiang et al., 2024; Scale AI, 2025), user satisfaction signals are similarly relevant for holistic evaluation of successful task completion. Criteria for a useful, high-quality spreadsheet workbook output can depend heavily on a variety of explicit and implicit contextual factors. Though comprehensive factorization of preference votes to profile the full cross-product of user, prompt, and model is beyond the scope of this study, we leverage preference votes in conjunction with spreadsheet and prompt features to conduct targeted analyses of model capabilities and user behaviors across prompt categories.

4.2 OUR APPROACH

Our task formulation and spreadsheet evaluation methods are agnostic to the spreadsheet synthesis method. In this paper, we explore a setting assuming a single end-to-end generation of a *serialized representation* of a spreadsheet workbook that is then rendered deterministically. Specifically, models are tasked with generating a JSON schema representation of a spreadsheet workbook according to the sheet specification format described in Appx. C. The schema calls for specification of cell content, sheet structure, and cell style, including, optionally, conditional formatting, over potentially multiple sheets in a workbook.

Alternative approaches to spreadsheet generation may be iterative or agentic; we leave these to future study. Our approach explicitly materializes portable representations of spreadsheet workbooks. These JSON representations are rendered deterministically in the client-side browser via SpreadJS. In most cases, we used a model's structured outputs API to enforce adherence to our schema, but for Anthropic models, the schema was appended to the system prompt, also shown in Appx. C.

[2] https://evansdata.com/press/viewRelease.php?pressID=365

[3] https://slashdata.co/post/global-developer-population-trends-2025-how-many-developers-are-there

[4] https://www.bloomberg.com/features/2025-microsoft-excel-ai-software/

4.3 ARENA METHODOLOGY

SPREADSHEETARENA is a platform for pairwise evaluation of LLM-produced spreadsheet workbooks via user vote. Users submit natural language descriptions of their use case or intent, and are shown eight anonymous generated spreadsheet artifacts for each submitted prompt. As we collect votes, we estimate Bradley-Terry ability parameters (Bradley & Terry, 1952) for our models. Elo scores (Coulom, 2007) are obtained by linearly rescaling the Bradley-Terry parameters, with GPT-4o anchored at 1000. We do not include new models in the leaderboard until they have at least 50 votes. To estimate the effects of these features on model performance, we adopt an augmented Bradley-Terry model to express win probability as:

$$P(A \succ B) = \sigma\left(\theta_A - \theta_B + \sum_{k=1}^{K} \beta_k (X_{Ak} - X_{Bk})\right) \tag{1}$$

where θ_i denotes the latent skill of model i, β_k is the coefficient for feature k, and X_{ik} is the mean value of feature k across outputs generated by model i.

We initialize SPREADSHEETARENA with 436 "seed" prompts authored and initially voted on by expert contributors, spanning 6 representative categories of spreadsheet generation: Academic & Research, Corporate Finance & Financial Planning and Analysis (FP&A), Creative & Generative, Operations & Supply Chain, Professional Finance, and Small/Medium-Sized Business (SMB) & Personal – see Appx. E for examples. This taxonomy captures variation in user context and workflow purpose rather than narrowing the scope to the subject domain alone. Given how workflow-dependent spreadsheet usage is in practice, categorization by prompt intent, prompt form, and context yields a more meaningful distribution across diverse spreadsheet applications.

To classify user-submitted prompts into these categories, we build a prompt categorization pipeline that executes upon prompt submission to auto-categorize prompts on-the-fly. The pipeline uses 1024-dimensional `Qwen3-Embedding-8B` embeddings of prompts, which are then labeled according to a k-nearest neighbors (k-NN) model fit on the 436 seed prompt embeddings. When a new prompt is submitted, the arena generates pairwise model match-ups dynamically using Algorithm 1, which prefers models so far seen in relatively fewer battles across the platform. Pairs where at least one model generates an invalid output are discarded and replaced following the same sampling strategy.

5 RESULTS AND ANALYSIS

We analyze spreadsheets generated by LLMs in SPREADSHEETARENA via arena votes, programmatically extracted spreadsheet features, and expert evaluations. We describe tendencies of different models, variation in use cases, variation in form and style of winning spreadsheets across domains, and comparisons of evaluations across domains.

5.1 GENERAL RESULTS

We collect a total of 4,357 blind preference votes over pairwise battles between 16 models in SPREADSHEETARENA. Most votes (87.5%) indicated a preference for one generated spreadsheet over the other. Among the remaining battles, 4.0% were ties (equally as good), and both candidate spreadsheets were judged as unsatisfactory in 8.5%. In general, prompts with more open-ended use cases (e.g., creative and generative prompts that request drawings or creation of spreadsheet-based puzzles) tend to be more commonly associated with "both are bad" votes. "Both are bad" votes are much more common in open-ended Creative & Generative use cases – on the other hand, "both are bad" votes are almost nonexistent in SMB & Personal use cases. However, for most of our analyses, we use only preference evaluations where a clear preference of one spreadsheet was indicated.

Spreadsheet Preferences vs. Code and Chat Settings In general chat settings, users prefer longer responses with richer formatting Scale AI (2025). Though there is no spreadsheet feature(s) that is a direct analog to this notion of verbosity or formatting, we do find that significant features corresponding to more text, larger spreadsheets, larger notebooks, more non-empty cells, or more formatting are positively associated with higher win probabilities Table 2. In comparison to code generation in particular, highly rated models in SPREADSHEETARENA are often also those that show

strong capabilities in coding benchmarks, but high coding benchmark scores are not fully explanatory of SPREADSHEETARENA rankings, nor should we assume that spreadsheet generation capability is simply a function of existing tasks.

Evaluation Taxonomies We use three complementary evaluation frameworks, each designed to illuminate a distinct layer of spreadsheet generation quality. (1) We extract a set of 29 **programmatic features** spanning formula quality, formatting, and structure directly from the spreadsheet artifacts (§5.2), and analyze their statistical associations with arena preferences. (2) We construct a **data-driven failure taxonomy** by clustering LLM-generated loss rationales (§5.3), revealing systematic breakdown patterns not easily captured by scalar features. (3) We apply an **expert-designed rubric** grounded in professional finance conventions (§5.4), introducing domain-specific normative standards that we find are not well-reflected in crowd preferences. Overall, we aim to capture the complexity of spreadsheet generation and its evaluation. Meaningful evaluation requires accounting for heterogeneous preference signals alongside the aggregate performance scores that our global arena rankings provide.

5.2 PREFERENCE AND PERFORMANCE DECOMPOSITION

We expand upon methodology from Scale AI (2025) and decompose model performance as determined by arena preference votes, by augmenting the vanilla Bradley-Terry model with explanatory feature variables. We extract 29 features programmatically from spreadsheet workbooks. The full-size features are described in detail in Tab. 4 and are distributed across 4 categories that broadly capture spreadsheet quality. **Formula Quality** features quantify computational correctness and sophistication, including error rates and the use of lookup, conditional, and financial functions; **Content** features capture the composition of cell types, including text, formulas, and numeric values; **Formatting** features characterize visual styling such as fills, borders, font treatments, and adherence to professional color-coding conventions; and **Structure** features describe spatial organization, including sheet dimensions, cell density, and table layouts.

5.2.1 GENERAL FEATURE EFFECTS.

We fit the Bradley-Terry model in Eq. 1, with 29 spreadsheet features as logistic regression covariates, to the pairwise votes. Tab. 1 reports model Elo ratings before and after feature controls, while Fig. 4 in Appx. D visualizes the corresponding shifts.

Model	Base	Feat.	ΔElo	ΔRnk
Claude Opus 4.5	1550	1333	−217	0
Gemini 3 Pro	1325	1268	−56	+2
Claude Opus 4.1	1406	1266	−140	0
Claude Sonnet 4.5	1427	1257	−170	−2
Gemini 2.5 Flash	1256	1225	−31	+2
Gemini 2.5 Pro	1279	1221	−58	0
GPT-5.2	1297	1175	−122	−2
GPT-5	1189	1159	−30	+1
Grok 4.1 Fast	1255	1139	−116	−1
Grok 4	1144	1132	−12	+1
GPT-5.1	1158	1125	−33	−1
Grok Code Fast 1	1089	1108	+19	0
Kimi K2 Instruct	977	1021	+44	+1
GPT-4o	1000	1000	0	−1
Qwen3 30B	692	849	+157	0
Llama 4 Maverick	632	783	+151	0

Table 1: New model rankings after feature control. See Fig. 2 for a visual

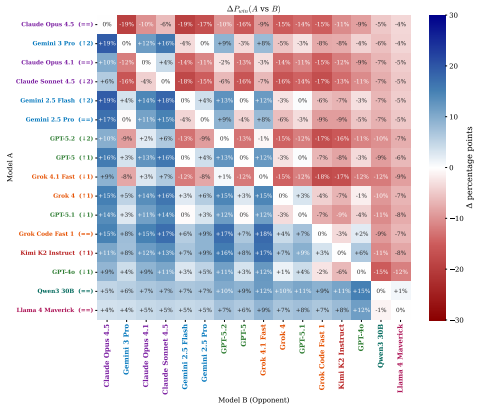


Figure 2: Pairwise win probability change (ΔP_{win}) after controlling for 29 spreadsheet features via Bradley-Terry regression.

Leaderboard Compression. The most immediate effect of feature controls is a compression of the rating distribution. Claude Opus 4.5 retains the top position but drops 217 Elo points (1550 → 1333). Models that underperform in raw rankings show substantial increases in Elo points after controls (Qwen3-30B: 157↑, Llama-4-Maverick: 151↑). The most notable ranking change is Gemini 3 Pro's

ascent from 4th to 2nd place, overtaking both Claude Sonnet 4.5 and Claude Opus 4.1. Critically, Gemini 3 Pro experiences only a 56-point Elo decrease, and other Gemini models undergo similarly small ratings shifts when controlling for features, suggesting that Gemini's baseline performance is less confounded by the features we measure; in particular, Claude models seem to have formatting tendencies that happen to align with preference votes. Fig. 2 presents the pairwise win probability changes. We find that feature controls redistribute competitive advantage. Claude Opus 4.5's average win probability against all opponents decreases by 11.2 percentage points on average.

Which features matter? Of the 29 features tested, 16 are statistically significant ($p < 0.05$); Tab 2 reports their coefficients. The strongest predictors are text density (`pct_text`, $+1.56$), background fills ($+1.15$), and numeric content ($+1.02$), features corresponding to explanatory features and formatting.Formula error rate (-1.34) is the strongest negative predictor. The importance of structure is nuanced, with wider layouts being preferred (`log_col_count`, $+0.72$) over fragmented structures such as parallel tables (-0.21) and tall aspect ratios (-0.81). On the other hand, formula sophistication features do *not* achieve significance: lookup functions ($p = 0.73$), conditionals ($p = 0.28$), and embedded constants ($p = 0.55$) show no reliable effect on win probability. Broadly, complex formula logic does not appear to be rewarded.

Table 2: Feature and ranking decomposition. **(a)** Statistically significant features across all prompts (see Tab. 5 for full results). **(b)** Significant features by domain; sign reversals indicate context-dependent preferences. **(c)** Model rankings after feature controls for Academic & Research prompts.

(a) Significant features (all prompts)

Feature	Coef.
pct_text	+1.562
compute_error_rate	−1.338
pct_fill	+1.150
compute_pct_numeric	+1.020
log_aspect_ratio	−0.814
log_col_count	+0.725
pct_number_format	+0.657
largest_table_pct	−0.563
has_border	+0.312
log_num_blank_rows	−0.248
has_parallel_tables	−0.214
log_distinct_functions	−0.211
log_total_text_tokens	+0.167
avg_tables_per_sheet	+0.104
log_table_size_variance	+0.050
num_single_cell_rows	−0.027

(b) Significant features by domain

Domain	Feature	Coef.	p
Academic	pct_number_format	−5.38	0.04
	pct_fill	+3.17	0.04
Finance	finance_color_convention	+1.63	0.02
	largest_table_pct	−1.00	0.03
	pct_number_format	+0.61	0.05
	has_border	+0.57	0.01

(c) Finance rankings after feature controls

Model	Elo	Ctrl Elo	ΔElo	ΔRank
Claude Opus 4.5	1678	1395	−283	0
Claude Opus 4.1	1586	1376	−209	0
Claude Sonnet 4.5	1580	1334	−247	0
Gemini 3 Pro	1502	1312	−190	0
Gemini 2.5 Flash	1448	1294	−154	+2
GPT-5.2	1493	1293	−200	−1
Gemini 2.5 Pro	1453	1256	−198	−1
GPT-5	1318	1229	−89	+1
GPT-5.1	1293	1172	−121	+1
Grok Code Fast 1	1208	1157	−51	+1
Grok 4.1 Fast	1392	1152	−240	−3
Kimi K2 Instruct	1089	1088	−1	0
GPT-4o	1000	1000	0	0

5.2.2 Domain Specific Feature Effects.

Arena-wide analyses potentially obscure domain-specific preferences. We re-estimate our model specifically at the Academic & Research Category, as well as combining Professional Finance and Corporate & FP&A into one category. Feature effects and rankings vary substantially across domains.

In Academic & Research prompts, we see the most dramatic ranking perturbation in our study (see Tab. 9 in Appx. I). Claude Opus 4.5 drops from 1st to 9th place (-236 Elo), while Grok 4, which already had an unusually high baseline, ascends to the top ($+149$ Elo) and GPT-5.1 gains 228 points. Only two features achieve significance in this domain, reported in Tab 2, but the large negative coefficient, -5.38 ($p = 0.04$), for `pct_number_format` is noteworthy – Claude's heavy use of formatting negatively affects perceived negatively in this domain.

In contrast, in the Finance domain, four features achieve significance (Tab. 2), three of which reflect professional financial modeling conventions. The strongest predictor is `finance_color_convention_score`, which is not statistically significant arena-wide ($p =$

0.09) but has a coefficient of $+1.63$ ($p = 0.02$) for the Finance domain. We note that, though alignment with color conventions is simple to check for programmatically, full evaluation of adherence to financial modeling conventions is more challenging; see §5.4 for an expert evaluation study. Tab. 2 contains ranking changes for models over both finance categories.

5.3 Characterizing Dispreferred Spreadsheets

To complement our analysis in §5.2 which uses a programmatic feature set, we construct a **data-driven** failure taxonomy by investigating failure modes of losing candidates. Following (Deng et al., 2025), we design a taxonomy of tags to support characterization of losing candidate outputs, and subsequently calibrate an LLM judge to apply it to all decisive arena battles. Unlike Deng et al. (2025)'s error taxonomy that assumes a single "primary" failure mode in candidate solutions, however, our categories are explicitly co-occurring diagnostic tags that assume a single losing spreadsheet may exhibit multiple failure modes.

To validate the LLM categorization judge, 5 expert spreadsheet annotators independently labeled a stratified sample of 50 dispreferred spreadsheets, identifying the single most significant failure bucket out of the given taxonomy. The LLM judge's tag set contained the expert-designated primary failure mode in 78% of cases, indicating strong human alignment with automated review.

See Appendix J for methodological details.

Bucket Definitions. Each losing spreadsheet is tagged with all categories that contributed to the loss. On average, each losing spreadsheet receives 3.49 tags, reflecting that spreadsheet failures are typically multi-factorial. *In practice, very few spreadsheets were deemed "Unjudgeable" and we merge the label into "Non-functional."

Unjudgeable*: Cannot be meaningfully evaluated. Empty/truncated or unrelated output.
Non-functional: Unusable. Pervasive formula errors block all interpretation of key results.
Spec Non-compliance: Missing core deliverables that the prompt requires. Missing sections, tabs, scenarios, time horizons, or required outputs.
Integrity Failure: The spreadsheet is structurally untrustworthy even if it looks plausible, due to hardcoded checks, drivers not linked to outputs, and models that do not respond to input changes. (Core spreadsheet-specific failure: not just wrong, but misleading)
Numerical Computation Failure: The spreadsheet is computationally integrated but produces incorrect results. The error is in correctness of the formulas themselves rather than broken linkage or misleading structure.
Interpretability Failure: The spreadsheet is hard to follow, teach from, or hand off due to assumptions, calculations, and outputs are not clearly separated.
Low User Value: Correct and readable, but provides no meaningful decision value.
Presentation Deficiency: inconsistent formatting/nonstandard conventions/missing visual hierarchy.

Results. Presentation Deficiency is the most pervasive tag, appearing in each model's losses between 57-96% of the time. Table 11 in Appendix J reports the rate at each model's failures are tagged with a given failure mode, demonstrating each model's characteristic failure signature. For example, in 77% of Qwen3 30B losses, Spec Noncompliance was identified as a contributing factor while 45% of losing battles were tagged as Non-functional. Similarly, Llama 4 Maverick has an 86% rate of Spec Non-Compliance.

Other models exhibit a qualitatively different signature. GPT-5 has fewer errors than the population average in Spec Non-Compliance, Integrity, and Numerical Computation categories, indicating its losses are less likely to stem from missing deliverables or computational errors. Instead, its residual failures are more often associated with presentation or interpretability.

Notably, the Claude family, while enjoying high SpreadsheetArena ratings, shows a distinctive failure profile. Claude Opus 4.5 losses are less often attributed to Spec Noncompliance and Presentation Deficiency relative to the other models (at 18% and 62% respectively), yet are relatively more often attributed to Integrity and Numerical Computation Errors, at 52% and 74% respectively. This suggests Claude's losses are least likely to stem from superficial polish or incomplete outputs. Instead, Claude models' losses are disproportionately related to auditability- and correctness-critical

failures that are harder for non-experts to detect but potentially more decisive under expert scrutiny – this result aligns with the baseline vs. feature-adjusted Elo scores seen in §5.2.

These profiles suggest that spreadsheet generation capability does not lie on a single continuum – different models have different tendencies toward apparent completeness and structural correctness.

5.4 Finance Domain Expert Evaluation Study

Dimension	Description
Color Coding, Formatting & Visual Restraint	Purposeful, consistent formatting that supports readability
Financial Modeling Conventions	Adherence to standard finance modeling norms
Purpose & Practical Utility	Degree to which the spreadsheet fulfills the prompt and supports decisions
Structure & Organization	Clear inputs-calculations-outputs flow and auditability
Errors & Accuracy	Formula correctness and absence of Excel errors
Formula Conventions	Use of best practices for inputs, calculations, and formula design

Table 3: Evaluation dimensions for expert annotation of finance-domain spreadsheets

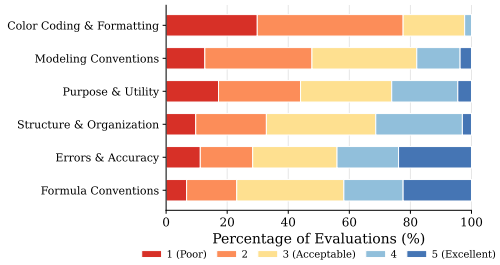


Figure 3: Distribution of expert ratings across six evaluation dimensions for finance-domain spreadsheets ($n = 134$ evaluations). Color Coding and Formatting stands out as the weakest dimension, with 77.6% of evaluations scoring 2 or below.

Professional finance spreadsheets follow established modeling conventions. While arena-style votes reflect generalized preferences, they do not directly measure adherence to industry standards. To contextualize SpreadsheetArena results for financial services settings, we conduct a blinded expert evaluation of arena-generated spreadsheets from finance-domain prompts.

Study design and protocol. We selected 52 battles with strict preference outcomes (excluding `Tie` and `Both are bad`), yielding 52 win-loss pairs (104 spreadsheets total). Battles were restricted to finance-domain prompts using manual labeling of seed prompts and k-NN classification for unlabeled submissions (§4.3). Prompts span canonical financial workflows, including DCFs, LBOs, and distribution waterfalls. Five evaluators with at least two years of Excel-based financial modeling experience (investment banking and private equity backgrounds) rated spreadsheets while blinded to model identity and arena outcome. Each spreadsheet was scored on six dimensions using a 5-point Likert scale (Table 3; full rubric in Appx. L). Fifteen spreadsheets were rated by three experts to assess agreement; the remaining 89 were rated once, yielding 134 total evaluations.

Overall performance. The mean overall rating was 2.87 (SD = 0.87), slightly below the midpoint (3 = acceptable). Only 23.1% of evaluations scored ≥ 4, while 32.1% scored ≤ 2. Performance was stronger on functional criteria: *Errors & Accuracy* ($M = 3.28$, 71.6% ≥ 3) and *Formula Conventions* ($M = 3.34$, 76.9% ≥ 3). Adherence was weaker for *Modeling Conventions* ($M = 2.61$, 47.8% ≤ 2) and *Purpose & Utility* ($M = 2.69$, 44.0% ≤ 2). The largest deficiency was *Color Coding and Formatting* ($M = 1.95$, SD = 0.77), with 77.6% scoring ≤ 2 and only 2.2% scoring ≥ 4. No model consistently followed established professional formatting standards (i.e., blue = hard-codes/assumptions, black = formulas/calculations, green = cross-sheet links).

Alignment with arena preferences and reliability. Across 52 battles, expert ratings agreed with arena outcomes in 42.3% of cases, disagreed in 32.7%, and tied in 25.0% (Fig. 12). Among decisive comparisons, agreement was 56.4%, only modestly above chance. For the 15 triply-rated spreadsheets, Krippendorff's α ranged from 0.27 to 0.51 across dimensions, indicating low inter-rater reliability. Limited alignment between expert evaluations and arena preferences suggests that generalized arena preferences may not fully capture finance domain-specific quality requirements. Despite variability in precise rankings, aggregate scores suggest only partial adherence to professional financial standards.

ETHICS STATEMENT

We obtain consent from users of SPREADSHEETARENA to use their anonymized submitted prompts and vote data for research purpose. Users are advised not to submit confidential or proprietary content. Generated spreadsheets may contain errors and do not necessarily adhere to professional standards; arena rankings reflect user preference rather than financial correctness or suitability for real-world decision-making. Our expert evaluation study is limited in scope and exhibits modest inter-rater agreement.

REFERENCES

Afra Feyza Akyürek, Advait Gosai, Chen Bo Calvin Zhang, Vipul Gupta, Jaehwan Jeong, Anisha Gunjal, Tahseen Rabbani, Maria Mazzone, David Randolph, Mohammad Mahmoudi Meymand, Gurshaan Chattha, Paula Rodriguez, Diego Mares, Pavit Singh, Michael Liu, Subodh Chawla, Pete Cline, Lucy Ogaz, Ernesto Hernandez, Zihao Wang, Pavi Bhatter, Marcos Ayestaran, Bing Liu, and Yunzhong He. Prbench: Large-scale expert rubrics for evaluating high-stakes professional reasoning, 2025. URL https://arxiv.org/abs/2511.11562.

Rahul K. Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñonero-Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, Johannes Heidecke, and Karan Singhal. Healthbench: Evaluating large language models towards improved human health. *arXiv preprint arXiv:2505.08775*, 2025. doi: 10.48550/arXiv.2505.08775. URL https://arxiv.org/abs/2505.08775.

Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952. ISSN 00063444, 14643510. URL http://www.jstor.org/stable/2334029.

Jianfeng Cai, Jinhua Zhu, Ruopei Sun, Yue Wang, Li Li, Wengang Zhou, and Houqiang Li. Disentangling length bias in preference learning via response-conditioned modeling, 2025. URL https://arxiv.org/abs/2502.00814.

Manuela Cattelan. Models for paired comparison data: A review with emphasis on dependent data. *Statistical Science*, 27(3):412–433, 2012. ISSN 08834237. URL http://www.jstor.org/stable/41714773.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021a. URL https://arxiv.org/abs/2107.03374.

Xinyun Chen, Petros Maniatis, Rishabh Singh, Charles Sutton, Hanjun Dai, Max Lin, and Denny Zhou. Spreadsheetcoder: Formula prediction from semi-structured context. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 1661–1672. PMLR, 18–24 Jul 2021b. URL https://proceedings.mlr.press/v139/chen21m.html.

Yibin Chen, Yifu Yuan, Zeyu Zhang, Yan Zheng, Jinyi Liu, Fei Ni, Jianye Hao, Hangyu Mao, and Fuzheng Zhang. Sheetagent: Towards a generalist agent for spreadsheet reasoning and manipulation via large language models. In *Proceedings of the ACM on Web Conference 2025*, WWW '25, pp. 158–177, New York, NY, USA, 2025. Association for Computing Machinery. ISBN 9798400712746. doi: 10.1145/3696410.3714962. URL https://doi.org/10.1145/3696410.3714962.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan Routledge, and William Yang Wang. Finqa: A dataset of numerical reasoning over financial data, 2022. URL https://arxiv.org/abs/2109.00122.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios N. Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael I. Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: an open platform for evaluating llms by human preference. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30, 2017.

Rémi Coulom. Computing "elo ratings"' of move patterns in the game of go1. *ICGA Journal*, 30(4): 198–208, 2007. doi: 10.3233/ICG-2007-30403. URL https://journals.sagepub.com/doi/abs/10.3233/ICG-2007-30403.

Xiang Deng, Jeff Da, Edwin Pan, Yannis Yiming He, Charles Ide, Kanak Garg, Niklas Lauffer, Andrew Park, Nitin Pasari, Chetan Rane, et al. Swe-bench pro: Can ai agents solve long-horizon software engineering tasks? *arXiv preprint arXiv:2509.16941*, 2025.

Haoyu Dong, Jianbo Zhao, Yuzhang Tian, Junyu Xiong, Shiyu Xia, Mengyu Zhou, Yun Lin, José Cambronero, Yeye He, Shi Han, and Dongmei Zhang. Spreadsheetllm: Encoding spreadsheets for large language models, 2025. URL https://arxiv.org/abs/2407.09025.

FAST Standard Organization. *FAST Modeling Best Practice Handbook*. FAST Standard Organization, London, 2015. Financial Modeling Standard.

Saibo Geng, Hudson Cooper, Michał Moskal, Samuel Jenkins, Julian Berman, Nathan Ranchin, Robert West, Eric Horvitz, and Harsha Nori. Jsonschemabench: A rigorous benchmark of structured outputs for language models, 2025. URL https://arxiv.org/abs/2501.10868.

Maarten Grootendorst. Bertopic: Neural topic modeling with a class-based tf-idf procedure, 2022. URL https://arxiv.org/abs/2203.05794.

Zhouhong Gu, Haoning Ye, Zeyang Zhou, Hongwei Feng, and Yanghua Xiao. Structbench: An autogenerated benchmark for evaluating large language model's ability in structure-rich text understanding. *arXiv preprint arXiv:2406.10621*, 2024. doi: 10.48550/arXiv.2406.10621. URL https://arxiv.org/abs/2406.10621.

Dan Hendrycks, Steven Basart, Saurav Kadavath, Mantas Mazeika, Akul Arora, Ethan Guo, Collin Burns, Samir Puranik, Horace He, Dawn Song, and Jacob Steinhardt. Measuring coding challenge competence with apps. In J. Vanschoren and S. Yeung (eds.), *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, volume 1, 2021. URL https://datasets-benchmarks-proceedings.neurips.cc/paper_files/paper/2021/file/c24cd76e1ce41366a4bbe8a49b02a028-Paper-round2.pdf.

Zhengyu Hu, Linxin Song, Jieyu Zhang, Zheyuan Xiao, Tianfu Wang, Zhengyu Chen, Nicholas Jing Yuan, Jianxun Lian, Kaize Ding, and Hui Xiong. Explaining length bias in llm-based preference evaluations, 2025. URL https://arxiv.org/abs/2407.01085.

Carlos E Jimenez, John Yang, Alexander Wettig, Shunyu Yao, Kexin Pei, Ofir Press, and Karthik Narasimhan. Swe-bench: Can language models resolve real-world github issues? In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 54107–54157, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/edac78c3e300629acfe6cbe9ca88fb84-Paper-Conference.pdf.

Fangyu Lei, Jixuan Chen, Yuxiao Ye, Ruisheng Cao, Dongchan Shin, Hongjin SU, Zhaoqing Suo, Hongcheng Gao, Wenjing Hu, Pengcheng Yin, Victor Zhong, Caiming Xiong, Ruoxi Sun, Qian Liu, Sida Wang, and Tao Yu. Spider 2.0: Evaluating language models on real-world enterprise text-to-sql workflows. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu

(eds.), *International Conference on Learning Representations*, volume 2025, pp. 28691–28735, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/46c10f6c8ea5aa6f267bcdabcb123f97-Paper-Conference.pdf.

Hongxin Li, Jingran Su, Yuntao Chen, Qing Li, and Zhaoxiang Zhang. Sheetcopilot: bringing software productivity to the next level through large language models. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Faeze Brahman, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. Wildbench: Benchmarking llms with challenging tasks from real users in the wild. *arXiv preprint arXiv:2406.04770*, 2024. doi: 10.48550/arXiv.2406.04770. URL https://arxiv.org/abs/2406.04770.

Zeyao Ma, Bohan Zhang, Jing Zhang, Jifan Yu, Xiaokang Zhang, Xiaohan Zhang, Sijia Luo, Xi Wang, and Jie Tang. Spreadsheetbench: Towards challenging real world spreadsheet manipulation. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 94871–94908. Curran Associates, Inc., 2024. doi: 10.52202/079017-3007. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/ac840df270ac537dd74530a15c332684-Paper-Datasets_and_Benchmarks_Track.pdf.

Mantas Mazeika, Alice Gatti, Cristina Menghini, Udari Madhushani Sehwag, Shivam Singhal, Yury Orlovskiy, Steven Basart, Manasi Sharma, Denis Peskoff, Elaine Lau, et al. Remote labor index: Measuring ai automation of remote work. *arXiv preprint arXiv:2510.26787*, 2025.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS) 2022*, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html. Also available as arXiv preprint arXiv:2203.02155.

Raymond R. Panko and Salvatore Aurigemma. Revising the panko–halverson taxonomy of spreadsheet errors. *Decision Support Systems*, 49(2):235–244, 2010. ISSN 0167-9236. doi: https://doi.org/10.1016/j.dss.2010.02.009. URL https://www.sciencedirect.com/science/article/pii/S0167923610000461.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741, 2023.

Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Romain Sauvestre, Tal Remez, Jérémy Rapin, Artyom Kozhevnikov, Ivan Evtimov, Joanna Bitton, Manish Bhatt, Cristian Canton Ferrer, Aaron Grattafiori, Wenhan Xiong, Alexandre Défossez, Jade Copet, Faisal Azhar, Hugo Touvron, Louis Martin, Nicolas Usunier, Thomas Scialom, and Gabriel Synnaeve. Code llama: Open foundation models for code, 2024. URL https://arxiv.org/abs/2308.12950.

Scale AI. Seal showdown: Technical report, September 2025. URL https://showdown.scale.com/assets/SEAL_Showdown_Tech_Report.pdf. Accessed: 2026-01-21.

Xiangru Tang, Yiming Zong, Jason Phang, Yilun Zhao, Wangchunshu Zhou, Arman Cohan, and Mark Gerstein. Struc-bench: Are large language models good at generating complex structured tabular data? In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pp. 12–34, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-short.2. URL https://aclanthology.org/2024.naacl-short.2/.

Wall Street Prep. Financial modeling best practices. https://www.wallstreetprep.com, 2020. Professional training materials used in investment banking.

Shijie Wu, Ozan Irsoy, Steven Lu, Vadim Dabravolski, Mark Dredze, Sebastian Gehrmann, Prabhanjan Kambadur, David Rosenberg, and Gideon Mann. Bloomberggpt: A large language model for finance, 2023. URL https://arxiv.org/abs/2303.17564.

Qianqian Xie, Weiguang Han, Xiao Zhang, Yanzhao Lai, Min Peng, Alejandro Lopez-Lira, and Jimin Huang. Pixiu: A large language model, instruction data and evaluation benchmark for finance, 2023. URL https://arxiv.org/abs/2306.05443.

Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, Yijing Xu, Haoqiang Kang, Ziyan Kuang, Chenhan Yuan, Kailai Yang, Zheheng Luo, Tianlin Zhang, Zhiwei Liu, Guojun Xiong, Zhiyang Deng, Yuechen Jiang, Zhiyuan Yao, Haohang Li, Yangyang Yu, Gang Hu, Jiajia Huang, Xiao-Yang Liu, Alejandro Lopez-Lira, Benyou Wang, Yanzhao Lai, Hao Wang, Min Peng, Sophia Ananiadou, and Jimin Huang. Finben: A holistic financial benchmark for large language models, 2024. URL https://arxiv.org/abs/2402.12659.

Hongyang Yang, Xiao-Yang Liu, and Christina Dan Wang. Fingpt: Open-source financial large language models, 2025. URL https://arxiv.org/abs/2306.06031.

Tao Yu, Rui Zhang, Kai Yang, Michihiro Yasunaga, Dongxu Wang, Zifan Li, James Ma, Irene Li, Qingning Yao, Shanelle Roman, Zilin Zhang, and Dragomir Radev. Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-SQL task. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 3911–3921, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1425. URL https://aclanthology.org/D18-1425/.

Tianshu Zhang, Xiang Yue, Yifei Li, and Huan Sun. TableLlama: Towards open large generalist models for tables. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 6024–6044, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.335. URL https://aclanthology.org/2024.naacl-long.335/.

Wei Zhao, Zhitao Hou, Siyuan Wu, Yan Gao, Haoyu Dong, Yao Wan, Hongyu Zhang, Yulei Sui, and Haidong Zhang. NL2Formula: Generating spreadsheet formulas from natural language queries. In Yvette Graham and Matthew Purver (eds.), *Findings of the Association for Computational Linguistics: EACL 2024*, pp. 2377–2388, St. Julian's, Malta, March 2024. Association for Computational Linguistics. URL https://aclanthology.org/2024.findings-eacl.158/.

Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat-Seng Chua. Tat-qa: A question answering benchmark on a hybrid of tabular and textual content in finance, 2021. URL https://arxiv.org/abs/2105.07624.

A Conclusion

Though many frontier models are often able to produce satisfactory spreadsheets according to user specifications, spreadsheet generation remains a challenging task for even the most performant LLMs in SpreadsheetArena. To our knowledge, spreadsheets are not a common benchmark domain compared to more well-studied domains such as coding; we argue that spreadsheets are a particularly interesting, understudied domain with potential for significant impact given the hundreds of millions of users of spreadsheet software. Our hope is that our work elucidates current gaps in spreadsheet generation capabilities and inspires further contributions in the space, including both strategies for improving LLM capabilities on the task and evaluations of other related tasks. For post-training in particular, our findings suggest that pairwise preference data over structured spreadsheet artifacts does not uniformly reward all dimensions. Notably, formatting features achieve significance while formula sophistication does not, significant features vary across domains, and crowd-sourced preferences

agree with expert judgments in finance only modestly. Useful future work may include upstream interventions for improving spreadsheet representation learning, data curation and post-training to improve task-specific generation ability, exploration of inference algorithms to explicitly compare distinct paradigms for generating spreadsheets, and scalable evaluations of spreadsheets that are simultaneously grounded in specific practical needs of users.

B Match Generation Algorithm Details

In this section, we present the full algorithm for match generation.

Algorithm 1 WeightedMatchEngine

Require: Valid models M, vote counts $V(\cdot)$
Ensure: Set of 4 valid model pairs
$P \leftarrow \{(m_i, m_j) \mid m_i, m_j \in M, i < j\}$
for all $(m_i, m_j) \in P$ **do**
  $w_{ij} \leftarrow (V(m_i), V(m_j))^{-1/2}$
end for
Sort P by decreasing w_{ij}
$W \leftarrow \emptyset$
$k \leftarrow 1$
while $W < 4$ and $k \leq P$ **do**
  $(m_i, m_j) \leftarrow P[k]$
  if both m_i and m_j produce valid outputs **then**
    $W \leftarrow W \cup \{(m_i, m_j)\}$
  end if
  $k \leftarrow k + 1$
end while

C SheetSpec Data Format Specification

We provide LLMs with a system prompt that calls for an output consisting of only a valid JSON schema representation of a spreadsheet workbook that fulfills the user's request specified in the prompt.

```
export const DEFAULT_SYSTEM_PROMPT = """You are a spreadsheet expert.

Return ONLY valid JSON conforming exactly to the provided JSON Schema
(SheetSpec@2).
Do not include any explanation, comments, or code fences  output a
single JSON object.

All formulas must:
- Use Excel-compatible A1 notation.
- Use commas (,) as argument separators.

Formatting and styling are optional but, if included, must comply
with the schema definitions.

Validate that:
- All sheet, column, and cell references used in formulas exist in
the output.
- The JSON is syntactically valid and can be parsed directly without
modification.""";
```

For Anthropic models, the `SheetSpec@2` spec is then appended to this system prompt. For all other models, the structured outputs API option is used to ensure valid schema JSON. A snippet of the full schema is shown below:

```
// ...
// ...
// SheetSpec JSON Schema
export const SheetSpecSchema = {
  type: 'object',
  required: ['version', 'sheets'],
  additionalProperties: false,
  properties: {
    version: { type: 'string', const: 'SheetSpec@2' },
    sheets: {
// ...
// ...
export type ConditionalFormatRule =
  | CellIsRule
  | CellIsBetweenRule
  | ExpressionRule
  | ContainsTextRule
  | ColorScaleRule
  | DataBarRule;

export type SheetSpec = {
  version: 'SheetSpec@2';
  sheets: Array<{
    name: string;
    cells: Array<Cell>;
    namedRanges?: Array<{
      name: string;
      ref: string;
    }>;
    conditionalFormats?: Array<ConditionalFormatRule>;
  }>;
  outputs?: Array<{
    name: string;
    sheet: string;
    ref: string;
    metric: 'value' | 'values';
  }>;
  rules?: {
    disallowVolatile?: boolean;
    allowedFunctions?: string[];
  };
};
```

Cell content can be strings, numerical values, or formulas. Cells can be styled with fills, fonts, borders, and number formatting. Named ranges for formula references are also supported. A substantial subset of Excel's conditional formatting functionality is supported, including value comparisons, custom formulas, color gradients, and data bars. Scale anchors support percentiles and auto-detected min/max values for data-relative formatting.

D SPREADSHEET FEATURES

Table 4 in Appx. D contains descriptions of all 29 spreadsheet features used as covariates in the Bradley-Terry model. Features are sorted into four categories spanning formula quality, content, formatting, and structure.

Table 5 contains feature effects on win probability for all prompts, for all 29 features.

Table 4: Spreadsheet features used as covariates in the Bradley-Terry model, grouped by category.

Category	Feature	Description
Formula Quality	`compute_error_rate`	Formula error rate
	`compute_pct_numeric`	Numeric cell ratio
	`log_distinct_functions`	Function variety
	`log_num_lookups`	Lookup function count
	`log_num_conditionals`	Conditional function count
	`pct_formulas_with_literals`	Embedded constants
Content	`pct_text`	Text cell ratio
	`pct_formula`	Formula cell ratio
	`log_total_text_tokens`	Text word count
Formatting	`pct_fill`	Background fill ratio
	`pct_bold`	Bold text ratio
	`has_border`	Border presence
	`pct_number_format`	Number formatting ratio
	`distinct_font_sizes`	Font size variety
	`pct_font_color`	Font color ratio
	`log_distinct_font_colors`	Font color variety
	`distinct_fills`	Fill color variety
	`finance_color_convention`	Color convention score
Structure	`log_row_count`	Row count
	`log_col_count`	Column count
	`log_aspect_ratio`	Sheet aspect ratio
	`cell_density`	Non-empty cell ratio
	`log_num_blank_rows`	Blank row count
	`num_single_cell_rows`	Single-cell rows
	`num_tables`	Table count
	`has_parallel_tables`	Side-by-side tables
	`avg_tables_per_sheet`	Tables per sheet
	`largest_table_pct`	Largest table share
	`log_table_size_variance`	Table size variance

Figure 4 shows the effects of controlling for all features on Elo ratings.

E Category Prompt Examples

Category	Task Description
Academic & Research	Create a spreadsheet to perform a difference-in-differences analysis for a policy intervention study. Set up two groups (treatment and control) with pre-intervention data for 2019–2020 and post-intervention data for 2021–2022. Include 8 observations per group with outcome variables showing baseline values around 50 for both groups, then treatment group increasing to around 65 post-intervention while control stays at 52. Calculate the difference-in-differences estimator, parallel trends assumption check, and standard errors. Include a simple visualization comparing the trends.
Corporate Finance & FP&A	Build a pricing and margin sensitivity model for a software business to help an entrepreneur understand how pricing changes impact profitability. Assume the business has 1,000 active customers, with monthly churn of 4% and 100 new customers added per month. Model three pricing scenarios: $20, $35, and $50 per month. Gross margin is 75% at $20, 80% at $35, and 85% at $50. Fixed operating costs are $40,000 per month. Show monthly revenue, gross profit, operating profit, and break-even point under each pricing scenario, and clearly compare outcomes side-by-side in a sensitivity table. Build with months across columns.

Continued on next page

Category	Task Description
Creative & Generative	Create a playable Checkers game in a spreadsheet. The 8×8 board should use shaded dark squares (playable) and locked light squares. Pieces use symbols: red = "r", black = "b", kings = "R"/"B". Implement click-based movement with alternating turns, legal diagonal moves only, mandatory jump captures with multi-jump enforcement, and automatic king promotion. Include illegal move prevention, turn indicator, captured piece counts, win/loss/draw detection, conditional formatting for valid moves and captures, and a "New Game" reset button.
Operations & Supply Chain	Create a centralized hiring tracker that logs incoming resumes and tracks candidates through each stage of the hiring process. Include applicant details, role applied for, screening status, interview stage, interview feedback, decision outcomes, and timelines. Add automatic status updates, time-to-hire metrics, funnel conversion rates, and visual summaries showing pipeline health and bottlenecks. Design as a reusable template with customizable stages, roles, and evaluation criteria.
Professional Finance	Build a fully integrated, institutional-quality leveraged buyout (LBO) model for a multi-segment operating company with three business segments: one cyclical, one subscription-based recurring revenue, and one capital-intensive legacy segment in decline. Finance the acquisition with a layered capital structure: revolver with cash sweep, Term Loan B with mandatory amortization, PIK toggle mezzanine tranche, seller notes with contingent interest, and rolled management equity with dilution mechanics. Project detailed operating assumptions per segment (revenue drivers, pricing vs. volume, gross margin bridges, SG&A leverage, maintenance vs. growth capex, working capital as function of revenue), consolidate into fully linked financial statements. Include transaction/financing fees, OID, deferred financing costs, goodwill/intangibles amortization, quarterly covenant testing (leverage, coverage) with breach triggers, excess cash flow sweeps, and PIK capitalization. Model scenario-based exits with sponsor IRR, MOIC, and cash-on-cash returns. Include sensitivity tables for leverage, entry/exit multiples, operating performance, and interest rates.
SMB & Personal	Create a weekly food tracker for calorie input from food and exercise output. Include an input area for current weight and target weight. Track calories in and calories out to facilitate weight loss monitoring.

F Category Spreadsheet Examples

F.1 Academic & Research

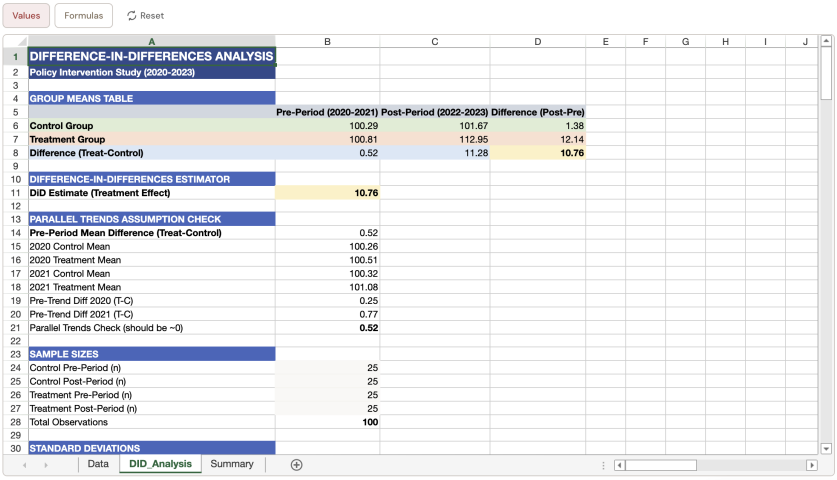


Figure 5: A model response to the "Academic & Research" prompt in Appx. E.

Table 5: Feature Effects on Win Probability (All Prompts). Asterisks denote statistical significance (*$p < 0.05$, **$p < 0.01$, ***$p < 0.001$).

Feature	**Coef.**	p**-value**
pct_text	+1.562	< 0.001***
compute_error_rate	−1.338	< 0.001***
pct_fill	+1.150	< 0.001***
compute_pct_numeric	+1.020	0.002**
log_aspect_ratio	−0.814	0.010**
pct_formula	+0.711	0.096
log_col_count	+0.725	< 0.001***
pct_number_format	+0.657	< 0.001***
pct_font_color	+0.592	0.152
finance_color_conv.	+0.558	0.094
largest_table_pct	−0.563	0.013*
has_border	+0.312	0.005**
cell_density	+0.303	0.200
log_row_count	+0.249	0.096
log_num_blank_rows	−0.248	0.002**
has_parallel_tables	−0.214	0.026*
log_distinct_functions	−0.211	0.026*
log_total_text_tokens	+0.167	0.014*
log_distinct_font_colors	+0.153	0.148
pct_formulas_w_literals	+0.114	0.547
avg_tables_per_sheet	+0.104	< 0.001***
distinct_font_sizes	+0.087	0.078
log_table_size_variance	+0.050	0.003**
log_num_conditionals	+0.037	0.283
num_single_cell_rows	−0.027	0.005**
log_num_lookups	+0.017	0.730
num_tables	−0.013	0.058
distinct_fills	+0.013	0.252
pct_bold	−0.040	0.880

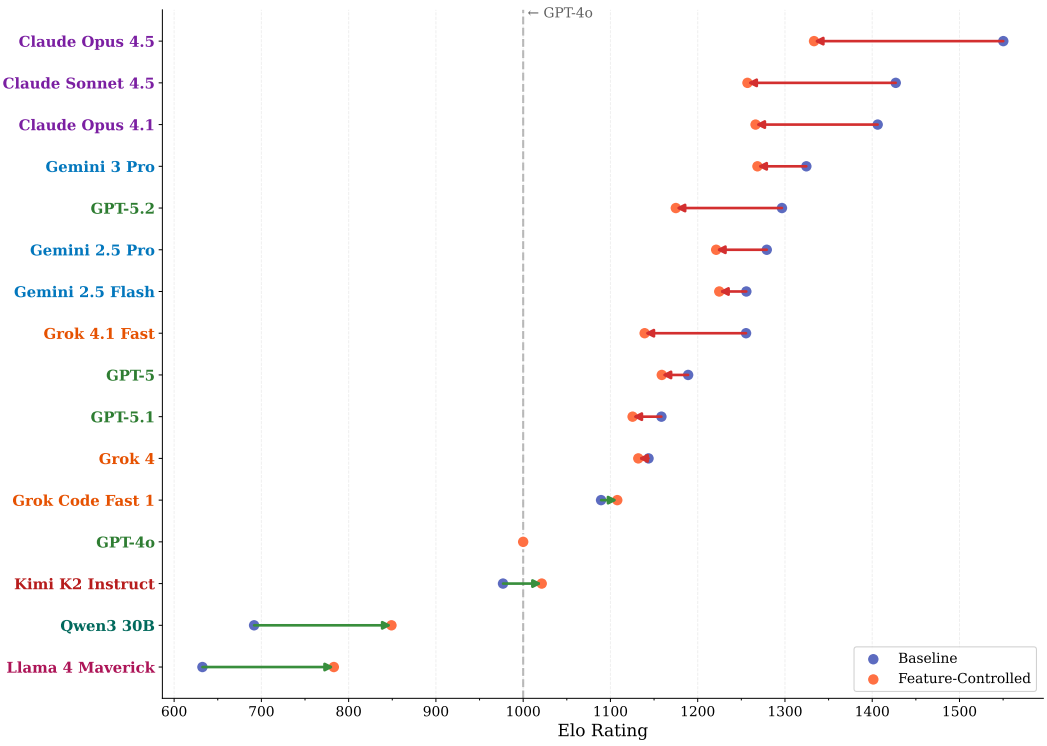
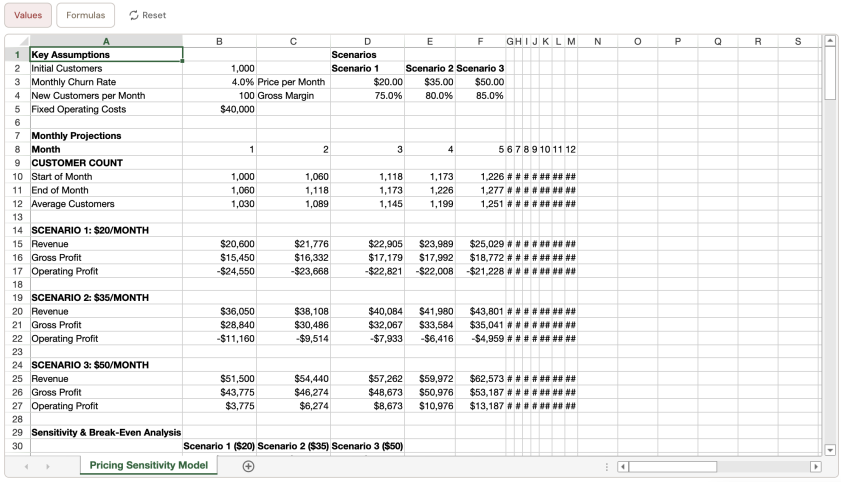


Figure 4: Elo ratings trend inwards after feature control.

F.2 Corporate Finance & FP&A



Values | Formulas | Reset

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	**Key Assumptions**			**Scenarios**															
2	Initial Customers	1,000		**Scenario 1**	**Scenario 2**	**Scenario 3**													
3	Monthly Churn Rate	4.0%	Price per Month	$20.00	$35.00	$50.00													
4	New Customers per Month	100	Gross Margin	75.0%	80.0%	85.0%													
5	Fixed Operating Costs	$40,000																	
6																			
7	**Monthly Projections**																		
8	**Month**	1	2	3	4	5	6	7	8	9	10	11	12						
9	**CUSTOMER COUNT**																		
10	Start of Month	1,000	1,060	1,118	1,173	1,226	#	#	#	#	#	#	#						
11	End of Month	1,060	1,118	1,173	1,226	1,277	#	#	#	#	#	#	#						
12	Average Customers	1,030	1,089	1,145	1,199	1,251	#	#	#	#	#	#	#						
13																			
14	**SCENARIO 1: $20/MONTH**																		
15	Revenue	$20,600	$21,776	$22,905	$23,989	$25,029	#	#	#	#	#	#	#						
16	Gross Profit	$15,450	$16,332	$17,179	$17,992	$18,772	#	#	#	#	#	#	#						
17	Operating Profit	-$24,550	-$23,668	-$22,821	-$22,008	-$21,228	#	#	#	#	#	#	#						
18																			
19	**SCENARIO 2: $35/MONTH**																		
20	Revenue	$36,050	$38,108	$40,084	$41,980	$43,801	#	#	#	#	#	#	#						
21	Gross Profit	$28,840	$30,486	$32,067	$33,584	$35,041	#	#	#	#	#	#	#						
22	Operating Profit	-$11,160	-$9,514	-$7,933	-$6,416	-$4,959	#	#	#	#	#	#	#						
23																			
24	**SCENARIO 3: $50/MONTH**																		
25	Revenue	$51,500	$54,440	$57,262	$59,972	$62,573	#	#	#	#	#	#	#						
26	Gross Profit	$43,775	$46,274	$48,673	$50,976	$53,187	#	#	#	#	#	#	#						
27	Operating Profit	$3,775	$6,274	$8,673	$10,976	$13,187	#	#	#	#	#	#	#						
28																			
29	**Sensitivity & Break-Even Analysis**																		
30		**Scenario 1 ($20)**	**Scenario 2 ($35)**	**Scenario 3 ($50)**															

Pricing Sensitivity Model

Figure 6: A model response to the “Corporate Finance & FP&A” prompt in Appx. E.

F.3 CREATIVE & GENERATIVE

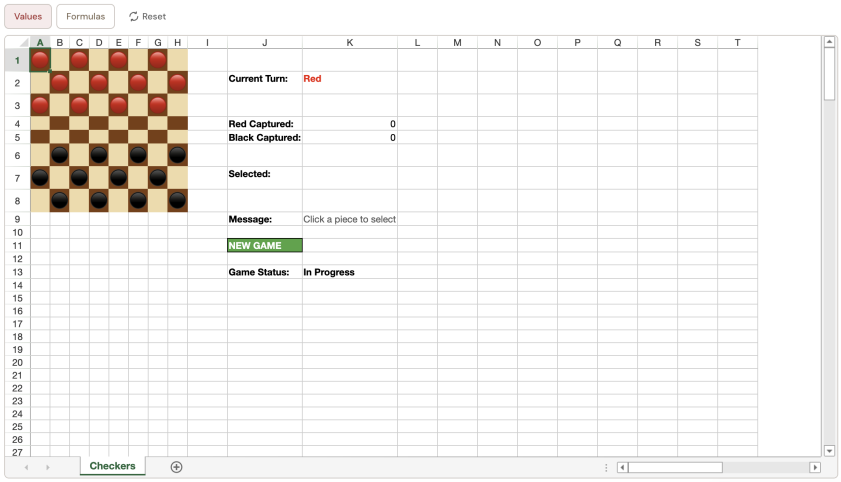


Figure 7: A model response to the "Creative & Generative" prompt in Appx. E.

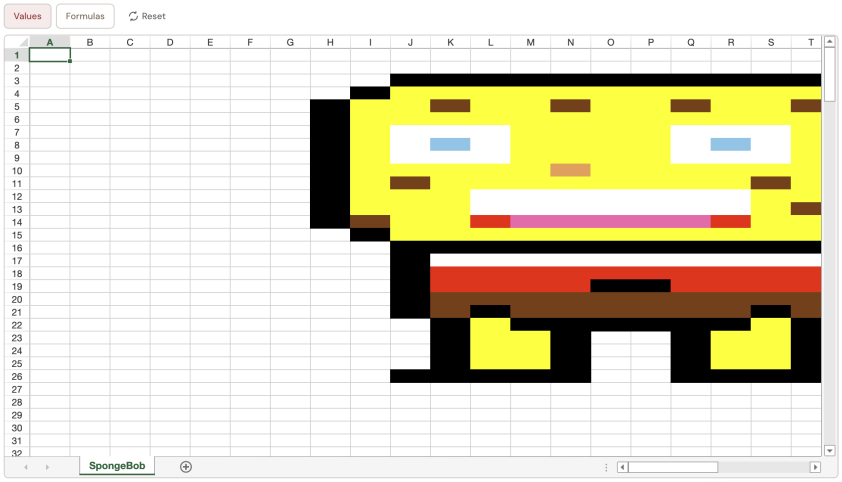
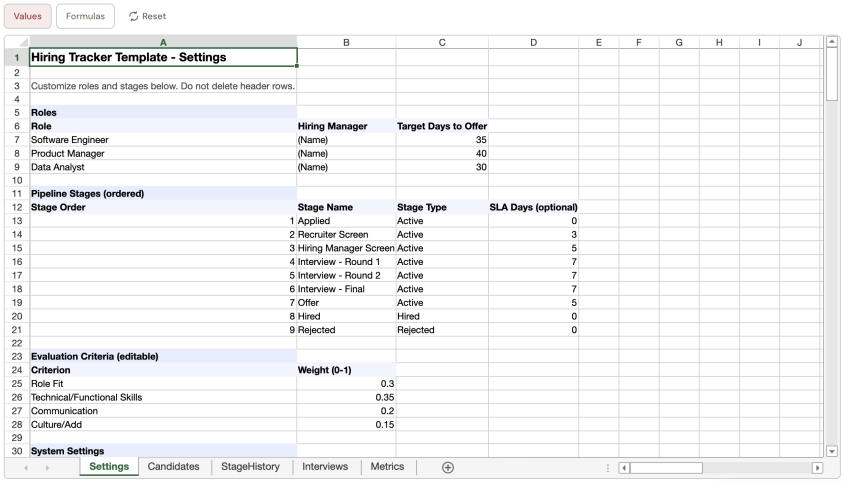


Figure 8: A model response to the prompt "draw a spongebob fully colored and not exceeding a 50 x 50 size".

F.4 OPERATIONS & SUPPLY CHAIN



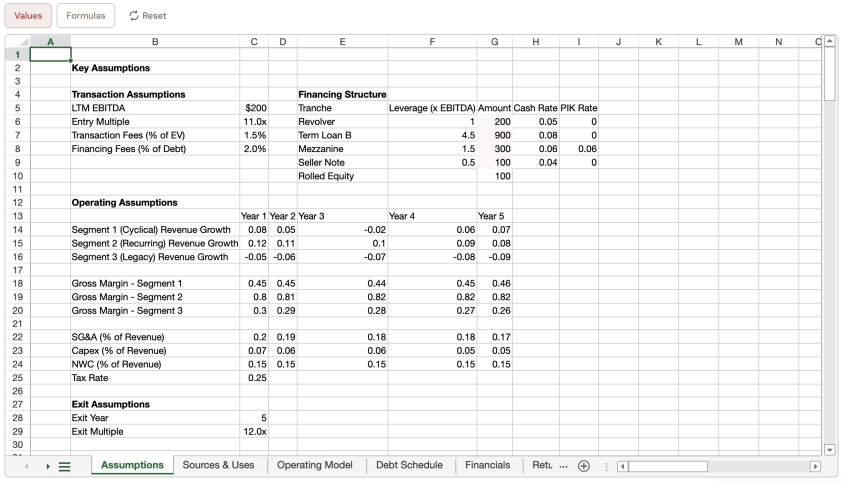
Values | Formulas | Reset

	A	B	C	D
1	**Hiring Tracker Template - Settings**			
2				
3	Customize roles and stages below. Do not delete header rows.			
4				
5	**Roles**			
6	**Role**	**Hiring Manager**	**Target Days to Offer**	
7	Software Engineer	(Name)	35	
8	Product Manager	(Name)	40	
9	Data Analyst	(Name)	30	
10				
11	**Pipeline Stages (ordered)**			
12	**Stage Order**	**Stage Name**	**Stage Type**	**SLA Days (optional)**
13	1	Applied	Active	0
14	2	Recruiter Screen	Active	3
15	3	Hiring Manager Screen	Active	5
16	4	Interview - Round 1	Active	7
17	5	Interview - Round 2	Active	7
18	6	Interview - Final	Active	7
19	7	Offer	Active	5
20	8	Hired	Hired	0
21	9	Rejected	Rejected	0
22				
23	**Evaluation Criteria (editable)**			
24	**Criterion**	**Weight (0-1)**		
25	Role Fit	0.3		
26	Technical/Functional Skills	0.35		
27	Communication	0.2		
28	Culture/Add	0.15		
29				
30	**System Settings**			

Settings | Candidates | StageHistory | Interviews | Metrics

Figure 9: A model response to the "Operations & Supply Chain" prompt in Appx. E.

F.5 PROFESSIONAL FINANCE



Values | Formulas | Reset

	A	B	C	D	E	F	G	H	I
1									
2		**Key Assumptions**							
3									
4		**Transaction Assumptions**			**Financing Structure**				
5		LTM EBITDA	$200		Tranche	Leverage (x EBITDA)	Amount	Cash Rate	PIK Rate
6		Entry Multiple	11.0x		Revolver	1	200	0.05	0
7		Transaction Fees (% of EV)	1.5%		Term Loan B	4.5	900	0.08	0
8		Financing Fees (% of Debt)	2.0%		Mezzanine	1.5	300	0.06	0.06
9					Seller Note	0.5	100	0.04	0
10					Rolled Equity		100		
11									
12		**Operating Assumptions**							
13			Year 1	Year 2	Year 3	Year 4	Year 5		
14		Segment 1 (Cyclical) Revenue Growth	0.08	0.05	-0.02	0.06	0.07		
15		Segment 2 (Recurring) Revenue Growth	0.12	0.11	0.1	0.09	0.08		
16		Segment 3 (Legacy) Revenue Growth	-0.05	-0.06	-0.07	-0.08	-0.09		
17									
18		Gross Margin - Segment 1	0.45	0.45	0.44	0.45	0.46		
19		Gross Margin - Segment 2	0.8	0.81	0.82	0.82	0.82		
20		Gross Margin - Segment 3	0.3	0.29	0.28	0.27	0.26		
21									
22		SG&A (% of Revenue)	0.2	0.19	0.18	0.18	0.17		
23		Capex (% of Revenue)	0.07	0.06	0.06	0.05	0.05		
24		NWC (% of Revenue)	0.15	0.15	0.15	0.15	0.15		
25		Tax Rate	0.25						
26									
27		**Exit Assumptions**							
28		Exit Year	5						
29		Exit Multiple	12.0x						
30									

Assumptions | Sources & Uses | Operating Model | Debt Schedule | Financials | Retu ...

Figure 10: A model response to the "Professional Finance" prompt in Appx. E.

F.6 SMB & PERSONAL

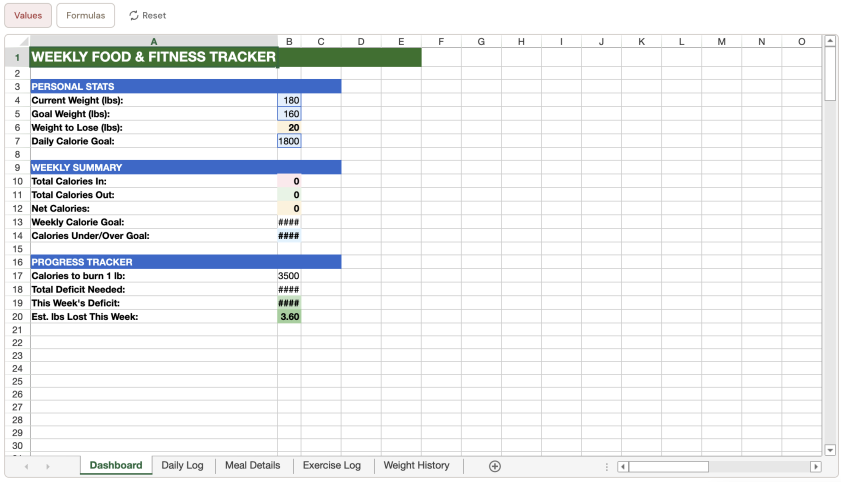


Figure 11: A model response to the "SMB & Personal" prompt in Appx. E.

G FEATURE COEFFICIENTS

Tab. 7 contains feature coefficients and p-values for our set of 29 features, across our six prompt categories. We use a single finance category for this analysis, merging professional finance and professional finance categories.

H MODEL CONFIGURATIONS

Table 8 contains model configurations used for our 16 models.

I ACADEMIC & RESEARCH CATEGORY MODEL RANKINGS CHANGE

Tab. 9 contains ranking changes for models over the Academic & Research category.

J FAILURE MODES ANALYSIS

See Table 11 for full results from the study in §5.3.

J.1 METHODOLOGICAL DETAILS

Category Discovery. We follow BERTopic (Grootendorst, 2022) to design a data-driven discovery pipeline to surface natural failure patterns from the arena corpus. We first generate open-ended failure rationales for a sample of 260 decisive battles (stratified across prompt category, losing model, and prompt complexity). For each battle, the `gpt-5-mini` judge receives JSON representations of both candidate spreadsheets along with the prompt text and winner designation, and produces a structured assessment of the losing spreadsheet's shortcomings.

We then embed these rationales using OpenAI's `text-embedding-3-small` model, reduce dimensionality with UMAP (5 components), and cluster via HDBSCAN with a minimum cluster size of 10. Central rationales from each cluster are fed to GPT-5 to generate descriptive category names and definitions. This pipeline yields 9 natural clusters, which we use as a starting point for the final hand-curated taxonomy of 7 buckets.

Table 7: Feature coefficients and p-values across prompt categories. Bold indicates statistical significance ($p < 0.05$). Coefficients represent the effect on the log-odds of winning.

	Creative & Generative		**Finance (Prof. + Corp.)**		**Academic & Research**		**SMB & Personal**		**Operations & Supply Chain**	
Feature	β	p	β	p	β	p	β	p	β	p
Formula Quality										
compute_error_rate	−0.90	.273	−1.36	.231	−1.01	.660	+0.66	.594	**−2.34**	**.042**
compute_pct_numeric	+1.02	.169	+1.44	.141	+3.65	.071	**+3.16**	**.002**	+0.18	.863
log_distinct_functions	+0.22	.601	−0.20	.264	−0.44	.256	−0.14	.585	−0.58	.061
log_num_lookups	**−0.45**	**.004**	+0.07	.464	−0.22	.215	+0.05	.704	+0.03	.849
log_num_conditionals	−0.02	.909	+0.02	.735	−0.11	.489	+0.04	.683	**+0.29**	**.010**
pct_formulas_with_literals	+0.29	.568	−0.19	.688	+0.04	.972	−0.61	.158	+0.78	.165
Content										
pct_text	**+2.35**	**.027**	+0.41	.774	+1.38	.593	**+3.52**	**.007**	**+3.41**	**.025**
pct_formula	+0.96	.404	−0.35	.730	+2.02	.328	**+2.68**	**.036**	+1.46	.258
log_total_text_tokens	+0.15	.212	+0.33	.149	+0.53	.256	+0.10	.663	+0.08	.824
Formatting										
pct_fill	**+1.17**	**.020**	+0.65	.530	**+3.17**	**.040**	+1.45	.059	−0.57	.451
pct_bold	−0.37	.478	+0.60	.397	+0.37	.858	**−1.97**	**.008**	−2.02	.051
has_border	**−0.87**	**.023**	**+0.57**	**.013**	−0.37	.569	+0.10	.698	**+0.71**	**.028**
pct_number_format	−1.13	.430	**+0.61**	**.046**	**−5.38**	**.041**	+0.99	.065	+1.63	.164
distinct_font_sizes	+0.16	.317	−0.01	.904	+0.02	.946	+0.14	.275	+0.03	.883
pct_font_color	+1.25	.119	+0.23	.898	+0.88	.819	+0.17	.902	−0.42	.762
log_distinct_font_colors	+0.15	.660	+0.30	.223	+0.53	.493	+0.06	.805	−0.08	.796
distinct_fills	+0.01	.670	−0.04	.299	+0.25	.055	+0.07	.217	−0.06	.238
finance_color_convention	+0.74	.390	**+1.63**	**.022**	−0.24	.889	−1.45	.152	−0.23	.836
Structure										
log_row_count	**+2.13**	**.044**	+0.41	.370	−0.71	.277	+0.94	.057	+0.12	.824
log_col_count	+0.05	.958	+0.15	.735	+1.19	.259	+0.73	.174	+1.14	.118
log_aspect_ratio	+0.20	.902	−0.49	.624	−1.62	.361	+0.40	.664	+0.19	.857
cell_density	**+2.22**	**.012**	−0.13	.811	+1.80	.083	−0.79	.186	+0.62	.464
log_num_blank_rows	−0.65	.108	−0.17	.316	+0.41	.358	−0.31	.160	−0.45	.091
num_single_cell_rows	−0.07	.153	−0.02	.402	+0.02	.776	−0.03	.160	+0.07	.335
num_tables	−0.09	.558	−0.00	.996	+0.02	.762	**−0.12**	**.002**	−0.07	.117
has_parallel_tables	+0.41	.335	−0.20	.308	−0.02	.955	**−0.57**	**.016**	−0.26	.411
avg_tables_per_sheet	**+0.50**	**.008**	+0.03	.617	+0.06	.718	**+0.26**	**.005**	**+0.34**	**.044**
largest_table_pct	+0.92	.369	**−1.00**	**.030**	−0.76	.521	+0.42	.468	**−2.17**	**.015**
log_table_size_variance	+0.00	.937	+0.07	.078	+0.05	.426	+0.02	.646	−0.03	.431

J.2 Sample Loss Categorization Judge Rationales

Table 10 contains sample LLM judge rationales for bucket categorizations.

Table 10: Sample LLM judge rationales for bucket categorizations.

Loss Bucket	**Judge Rationale**
Non-functional	Calculations contain pervasive formula errors caused by incorrect sheet references (e.g., `Calculations!B6..G6` and `B7..G7` use `'Assumptions.B6'` instead of `'Assumptions!B6'`), leaving key outputs non-functional.
Spec Non-compliance	The model fails the prompt requirement: the sensitivity table (`DCF!B43:F45`) produces enterprise-value outputs and is not converted to equity value per share (prompt requested equity value sensitivity).

Continued on next page

Loss Bucket	Judge Rationale
Integrity Failure	Input assumptions are not single-sourced or consistently linked (hardcoded step-up and amortization values are placed as year values rather than centralized blue input cells).
Numerical Computation Failure	There is incorrect math in the implied share price: `Bridge!B11` and `Bridge!B17` multiply price by 10 (`B7/B9*10`), which is an obvious unit/signature error that produces wrong implied prices.
Interpretability Failure	Labels contradict layout (A1 = "Quarter" while rows are product lines), assumptions and calculations aren't separated, making the model hard to audit.
Low User Value	It provides little user value—no translations, counts, or selection rationale so it's largely a wall of characters (shallow, low decision value).
Presentation Deficiency	Date cells are entered as plain text with formatting (`Assumptions!B4:B6`, `B11`) instead of true date types, and some number/date formatting is inconsistent with the requested conventions (e.g., days/years precision and long-date display), which lowers professional polish and increases risk of hidden errors.

Judging Method. After establishing our taxonomy, we apply our `gpt-5-mini` judge to each decisive arena battle, where one output was preferred over the other. The `gpt-5-mini` judge receives the original prompt and both full candidates as input. A system prompt (see Appendix J.3) provides all 8 category definitions with examples and instructs the judge to tag the losing spreadsheet with all relevant error categories, requiring clear evidence for each tag. The judge returns a structured JSON object containing the list of applicable category IDs and a 2-3 sentence rationale citing specific evidence, with example rationales in Appendix J.2. This multi-label design captures failure co-occurrence.

J.3 Loss Categorization Judge System Prompt

```
You are a senior spreadsheet professional analyzing why a spreadsheet
lost a head-to-head arena battle. You have deep expertise in
spreadsheet modeling, financial analysis, and the technical craft of
building production-quality workbooks.

A human reviewer compared two spreadsheet outputs built for the same
task and chose one as better. Your job is to think deeply and tag all
failure modes that genuinely contributed to the loss.

### SheetSpec Format

The spreadsheets are in SheetSpec@2 JSON:
- Each workbook has sheets, each with an array of cells
- Cell types: `text` (string), `number` (numeric), `formula` (Excel A1)
- Cells may have `style`: `fill`, `fontWeight`, `fontSize`,
`numberFormat`, `fontColor`, `border`
- Sheets may have `namedRanges` and `conditionalFormats`

### Failure Categories

**[0] Noise / Unjudgeable**
Definition: The spreadsheet can't be meaningfully judged against the
prompt. Use this tag ONLY in extreme cases.
Indicators:
  - File is empty or contains unrelated content
  - Prompt is incoherent or sheet content doesn't correspond to the
prompt
  - Generation is truncated in a way that makes evaluation impossible
```

Table 8: Model configurations grouped by model provider.

Model Name	Temp	Tokens
OpenAI (GPT)		
GPT-5	default	60,000
GPT-5.2	0.7	128,000
GPT-5.1	0.7	128,000
GPT-4o	default	16,384
Anthropic (Claude)		
Claude Opus 4.5	0.7	64,000
Claude Opus 4.1	0.7	32,000
Claude Sonnet 4.5	0.7	60,000
Google (Gemini)		
Gemini 3 Pro	0.7	64,000
Gemini 2.5 Pro	0.7	60,000
Gemini 2.5 Flash	0.7	60,000
xAI (Grok)		
Grok 4.1 Fast	default	2,000,000
Grok Code Fast 1	0.7	200,000
Grok 4	0.7	60,000
Meta (Llama)		
Llama 4 Maverick	0.7	1,000,000
Alibaba (Qwen)		
Qwen3 30B	0.7	128,000
Moonshot (Kimi)		
Kimi K2 Instruct	0.7	256,000

Table 9: Model Rankings After Feature Controls: Academic & Research

Model	Elo	Ctrl Elo	ΔElo	ΔRank
Grok 4	1481	1630	+149	+1
GPT-5.1	1298	1526	+228	+4
Gemini 3 Pro	1305	1457	+152	+2
GPT-5	1257	1449	+192	+5
Gemini 2.5 Flash	1297	1432	+135	+2
Claude Opus 4.1	1429	1414	−15	−2
Gemini 2.5 Pro	1283	1367	+84	+1
Claude Sonnet 4.5	1446	1360	−85	−5
Claude Opus 4.5	1527	1291	−236	−8
Grok Code Fast 1	1141	1246	+105	0
GPT-4o	1000	1000	0	0

```
**[1] Broken / Non-functional**
Definition: The spreadsheet is unusable - the equivalent of 'code
that doesn't compile.'
Indicators:
  - Pervasive #DIV/0!, #REF!, #NAME?, or #VALUE! errors across key
output areas
  - Circular references that clearly prevent meaningful outputs
  - Key results are blank or invalid due to broken references

**[2] Prompt Miss / Incomplete Build**
Definition: The spreadsheet doesn't include the core deliverables the
prompt requires. It might calculate 'something,' but not what was
asked.
Indicators:
```

Model	Win %	Non-Functional	Spec Non-compliance	Integrity	Numerical Computation	Interpret-ability	Shallow	Presentation
Claude Opus 4.5	83.5	19%	18%	74%	52%	52%	31%	62%
Claude Sonnet 4.5	72.4	9%	28%	66%	45%	48%	36%	57%
Claude Opus 4.1	69.2	9%	28%	72%	46%	60%	39%	81%
Gemini 3 Pro	58.3	8%	55%	46%	36%	70%	66%	85%
GPT-5.2	52.7	28%	32%	57%	40%	46%	51%	65%
Gemini 2.5 Pro	51.4	15%	40%	48%	33%	65%	49%	88%
Gemini 2.5 Flash	51.3	3%	39%	24%	22%	61%	59%	92%
Grok 4.1 Fast	49.5	19%	37%	53%	47%	57%	63%	64%
GPT-5	41.8	12%	24%	35%	20%	63%	46%	88%
GPT-5.1	35.2	27%	34%	57%	49%	60%	48%	80%
Grok 4	35.0	23%	44%	27%	11%	62%	52%	96%
Grok Code Fast 1	27.1	21%	48%	60%	51%	76%	60%	93%
Kimi K2 Instruct	23.7	44%	44%	63%	46%	64%	54%	76%
GPT-4o	20.1	22%	68%	65%	47%	55%	60%	70%
Qwen3 30B	9.6	45%	77%	73%	53%	83%	61%	75%
Llama 4 Maverick	6.7	20%	86%	53%	35%	77%	78%	87%

Table 11: Failure tag rate by model (% of each model's losses). Models show a high propensity towards presentation failures across the board. Weaker models struggle with prompt alignment and correctness.

```
  - Missing required sections, tabs, scenarios, or time horizons
  - Wrong dimensionality (e.g., annual vs. monthly when prompt
specifies otherwise)
  - Coverage too narrow - only a small subset of what was requested
  - Key required outputs (e.g., MOIC table, sensitivity analysis) are
absent

**[3] Integrity / Architecture Failure**
Definition: The spreadsheet is structurally untrustworthy even if it
looks plausible - not just wrong, but misleading or non-integrated.
Indicators:
  - Hardcoded 'checks' - status says PASS because the checker is fake
  - Key drivers are not linked to outputs - model doesn't respond to
input changes
  - Mis-referenced ranges, duplicated drivers, unintentional
circularity
  - 'Single source of truth' is violated - brittle and non-auditable

**[4] Incorrect Logic / Math**
Definition: The formulas are linked and the structure is real, but
the underlying logic or math is wrong.
Indicators:
  - Constraints violated in the output
  - Totals don't tie or reconcile
  - Off-by-one, double-counting, or sign-convention errors
  - Scenario outputs don't match inputs or stated assumptions

**[5] Unclear Structure / Interpretability Failure**
Definition: The spreadsheet is hard to follow, teach from, or hand
off to a collaborator.
Indicators:
  - Assumptions, calculations, and outputs are not clearly separated
  - Labels and numbers are misaligned or ambiguous
  - Not auditable by someone who didn't build it

**[6] Low User Value / Shallow**
Definition: The sheet may be correct and readable, but doesn't
provide meaningful decision value.
Indicators:
```

```
  - 'Wall of numbers' with no interpretive scaffolding
  - No sensitivity analysis, summaries, or 'so what' takeaways
  - Accurate but unactionable - letter of the prompt without serving
user intent

**[7] Presentation / Convention Deficiency**
Definition: The spreadsheet loses on polish and professional
conventions.
Indicators:
  - Messy or inconsistent formatting (number formats, alignment,
spacing)
  - Nonstandard accounting presentation
  - Missing visual hierarchy (no section headers, no input/calc color
coding)
  - Visually inferior in a head-to-head comparison

### Rules

- Tag ALL categories (1-7) that genuinely apply and significantly
contributed to the loss.
- Only tag a category if you see clear evidence for it.
- Do NOT tag everything - be honest and specific about categories
that truly contributed to the loss.
- Tag 0 (Noise) ONLY if the output truly cannot be evaluated - empty,
truncated, or completely unrelated to the prompt. If there is any
substantive content to judge, do NOT use tag 0.

### Input

You will receive:
- The original task prompt
- Spreadsheet A (SheetSpec@2 JSON)
- Spreadsheet B (SheetSpec@2 JSON)
- Any formula errors detected by the evaluation engine
- Which spreadsheet the human reviewer chose as better

Analyze the losing spreadsheet and explain why it lost.

### Output Format

Return a JSON object with exactly this schema (no other text):

{"tags": [1, 3], "rationale": "2-3 sentences citing specific
evidence."}

- `tags`: array of integer category IDs (0-7), sorted ascending
- `rationale`: brief explanation with cell references or structural
observations
```

K FINANCE EXPERT EVALUATIONS AUXILIARY RESULTS

Figure 12 displays overall ratings from the expert evaluation study discussed in §5.4 against arena results for spreadsheets in the finance study.

L FINANCE EXPERT EVALUATION SCORING RUBRIC AND INSTRUCTIONS

In this section, we include the full rater instructions and scoring anchors.

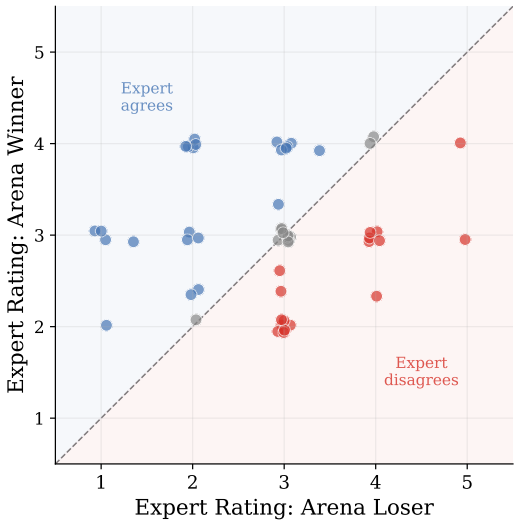


Figure 12: Expert ratings for arena winners vs. losers across 52 finance-domain battles. Points above the diagonal indicate expert agreement with arena outcomes; points below indicate disagreement.

L.1 Evaluator Instructions

For each assigned task, evaluators completed the following steps:

1. **Read the prompt.** Understand what the spreadsheet was supposed to accomplish.
2. **Download and open the Excel file.** Review it as you would any financial model—check formulas, structure, formatting.
3. **Rate on 6 criteria (1–5 scale).** Score each dimension using the detailed rubric below.
4. **Add notes (optional but helpful).** Brief explanations of scores help us understand the reasoning.
5. **Review and submit.** The overall rating is calculated automatically from the six dimension scores.

L.2 Rating Scale

All dimensions use a 5-point Likert scale with consistent anchors. The scale is described in Tab. 12.

Table 12: Likert scale description.

Score	General Definition
1	Poor: Significant issues; unacceptable in professional context
2	Below Average: Notable problems requiring substantial work to fix
3	Acceptable: Meets minimum requirements; functional but not polished
4	Good: Above average with only minor issues; professional quality
5	Excellent: Exceptional quality exemplifying best practices

L.3 Evaluation Dimensions and Scoring Anchors

L.4 Overall Rating

The overall rating is computed as the arithmetic mean of the six dimension scores, rounded to the nearest integer:

$$\text{Overall} = \text{round}\left(\frac{1}{6}\sum_{i=1}^{6} C_i\right) \tag{2}$$

Table 13: **Dimension 1: Errors & Accuracy.** *Focus*: Formula correctness and absence of Excel errors. This criterion evaluates whether the spreadsheet is free from formula errors, Excel error values (`#REF!`, `#DIV/0!`, `#NAME?`, `#VALUE!`, circular references), and calculation mistakes. A high-quality financial model should produce accurate results and be free of technical errors that would undermine trust in the outputs. *Evaluators assess*: Excel error values (`#REF!`, `#DIV/0!`, `#NAME?`, `#VALUE!`, `#N/A`), circular reference warnings, broken or invalid cell references, logical errors in formulas, calculation mistakes, and inconsistent formulas across similar rows/columns.

Score	Anchor
1	Multiple Excel errors present (`#REF!`, `#DIV/0!`, etc.), obvious calculation mistakes, circular references, or broken formulas that make the model unreliable
2	Several errors or inaccuracies that need fixing; model produces questionable results
3	Minor errors present but core calculations appear correct; needs cleanup but usable
4	Very few errors; calculations are accurate with only trivial issues
5	Error-free model; all formulas work correctly, calculations verified and accurate

Table 14: **Dimension 2: Formula Conventions.** *Focus*: Separation of inputs from calculations; no hardcoded values in formulas. This criterion assesses whether the model follows best practices for formula construction. Inputs (assumptions, raw data) should be clearly separated from calculations. Formulas should reference input cells rather than containing hardcoded "magic numbers." This makes models easier to audit, update, and understand. *Evaluators assess*: Hardcoded numbers embedded in formulas (e.g., `=A1*0.35` instead of `=A1*$B$5`), clear input/assumption sections separate from calculations, use of cell references instead of typed values, the "one row, one formula" rule, consistent formula patterns across rows/columns, and ability to change assumptions with automatic propagation.

Score	Anchor
1	Hardcoded values throughout; no separation between inputs and calculations
2	Many hardcoded values; inputs and calculations mixed together; difficult to audit
3	Some separation of inputs; occasional hardcoded values; functional but not ideal
4	Good separation of inputs from formulas; rare hardcoded values; easy to trace
5	Exemplary separation; all assumptions in dedicated area; fully dynamic model

where C_i denotes the score for dimension i.

Table 15: **Dimension 3: Color Coding & Visual Formatting.** *Focus*: Professional, purposeful use of color and formatting. This criterion evaluates the visual presentation of the spreadsheet. Professional financial models use color purposefully—typically blue for inputs, black for formulas, green for links to other sheets, and optionally red for external links or data provider pulls. Excessive or inconsistent coloring (the "rainbow effect") is distracting and unprofessional. Good formatting enhances readability without being garish. *Evaluators assess*: Consistent color scheme following finance conventions (blue for inputs/assumptions, black for formulas/calculations, green for cross-sheet links), absence of excessive "rainbow" formatting, professional font choices and sizes, consistent number formatting (decimals, percentages, currency), clear visual hierarchy, avoidance of merged cells, and clear distinction between headers/labels and data.

Score	Anchor
1	Garish "rainbow" formatting; colors obscure rather than clarify
2	Excessive or random coloring; distracting visual noise
3	Acceptable formatting; some color used but not consistently
4	Good visual presentation; mostly consistent; professional with minor issues
5	Clean, professional formatting; purposeful color coding; visually polished

Table 16: **Dimension 4: Structure & Organization.** *Focus*: Logical layout, clear sections, ease of audit. This criterion assesses how well the spreadsheet is organized for auditability. A well-structured model has a logical flow, clear sections, and is easy to navigate and audit. Information should be grouped sensibly, with inputs at the top or in a dedicated area, followed by calculations, and outputs clearly presented. *Evaluators assess*: Logical top-to-bottom or left-to-right flow, clear section headers and labels, distinct Inputs/Workings/Outputs sections, grouping of related items, easy-to-follow calculation flow, navigation aids for multi-sheet models, and absence of scattered calculations in random cells.

Score	Anchor
1	Disorganized; calculations scattered randomly; very difficult to audit
2	Poor organization; structure unclear; requires significant effort to follow
3	Functional structure; can follow logic but organization could improve
4	Well-organized; clear sections and flow; easy to navigate
5	Excellent organization; intuitive layout; professional structure

Table 17: **Dimension 5: Financial Modeling Conventions.** *Focus*: Adherence to standard financial modeling practices. This criterion evaluates whether the model follows established financial modeling conventions. This includes proper sign conventions, chronological time flow, integrity checks, and disciplined linking practices. A well-built model should be easy to audit without following complex reference chains. *Evaluators assess*: Consistent sign convention (expenses uniformly negative or positive), chronological left-to-right time flow, checks and integrity tests (balance checks, control totals, error flags), linking discipline (direct links to source, no daisy-chaining), standard financial statement formats, proper treatment of beginning vs. ending balances, and avoidance of unnecessary circularity.

Score	Anchor
1	Ignores conventions; inconsistent sign treatment; would not pass professional review
2	Multiple convention violations; difficult to reconcile with standard practices
3	Mostly follows conventions with some inconsistencies; acceptable for draft work
4	Good adherence to conventions; minor deviations; professional quality
5	Exemplary adherence to financial modeling best practices throughout

Table 18: **Dimension 6: Purpose & Practical Utility.** *Focus*: Does the model accomplish its stated purpose? This criterion evaluates whether the spreadsheet actually accomplishes what the prompt asked for and presents outputs in a decision-useful way. Note: this is distinct from Errors & Accuracy (which focuses on whether calculations are correct); here, focus on whether the model answers the prompt and is practically useful. *Evaluators assess*: Whether the model addresses all parts of the prompt, presence of requested outputs/calculations, usefulness for actual decision-making, appropriate scope (neither missing key elements nor over-engineered), suitability for sharing with clients or stakeholders, clarity of results presentation, and provision of actionable insights.

Score	Anchor
1	Fails to address the prompt; missing key requirements; not useful
2	Partially addresses prompt; significant gaps; limited practical utility
3	Meets basic requirements; answers core question but lacks polish
4	Good response to prompt; useful deliverable with minor gaps
5	Fully addresses all aspects; excellent utility; ready for professional use