LLMs as Virtual Instruments for Drug Formulation

Michael Craig[a], Gary Tom[a], Pauric Bannigan[a], Christine Allen[a,b,c,d], Riley Hickman[a]

[a] Intrepid Labs, MaRS Centre, 661 University Ave, Toronto, ON M5G 0B7, Canada
[b] Leslie Dan Faculty of Pharmacy, University of Toronto, Toronto, ON M5S 3M2, Canada
[c] Acceleration Consortium, University of Toronto, Toronto, ON M5S 3E5, Canada
[d] Department of Chemical Engineering & Applied Chemistry, University of Toronto, Toronto, ON M5S 3E5, Canada

Abstract

Pharmaceutical formulation design is a long-tail problem in which most drug candidates are supported only by small heterogeneous datasets. Although each case is distinct, its resolution is critical to the clinical and commercial success of drug products. While large language models (LLMs) have been increasingly applied to accelerate scientific discovery, formulation science has been relatively understudied, largely due to a scarcity of suitable public data. We evaluated whether commercial large language models (LLMs) can act as virtual scientific instruments by encoding self-emulsifying drug delivery systems (SEDDS) into compact "formulation cards" that serialize composition, physicochemical descriptors, and design metadata into a standardized prompt. Using three regression targets, droplet size, active pharmaceutical ingredient (API) loading mass fraction, and polydispersity index (PDI), we benchmark these models in two deployment regimes: within-API generalization (bootstrapping new formulations from previous batches) and between-API generalization (cold start prediction for held out APIs). We systematically vary inference-time reasoning effort and few-shot context size $K \in \{0, 5, 10, 20\}$, and find that increasing K results in increased predictive accuracy, while increasing reasoning effort results in low to modest and target-dependent improvements. Larger models also generally outperform smaller ones, suggesting that inference time scaling and in-context learning provide practical knobs to improve predictive accuracy without retraining. Together, these results establish the first ML baselines on the SEDDS corpus and position frontier LLMs as flexible, data-efficient instruments for accelerating drug formulation design in resource-constrained discovery pipelines. They also offer a reference point for evaluating how customized open-source models, via fine-tuning, retrieval, or hybrid approaches, compare under consistent conditions.

1 Introduction

Developing advanced drug delivery systems remains one of the most expensive and uncertain stages of pharmaceutical R&D. Beyond identifying potent and safe active pharmaceutical ingredients (APIs), formulation scientists must design carriers that maximize solubility, stability, and patient acceptability [Gao et al., 2021a, Gormley, 2024a]. This involves navigating a high-dimensional formulation design space of excipient combinations and process parameters, where exhaustive wet-lab exploration is infeasible.

The NeurIPS 2025 Workshop on AI Virtual Cells and Instruments: A New Era in Drug Discovery and Development (AI4D3 2025), San Diego, California, USA, 2025.

Machine learning (ML) has shown promise for accelerating formulation optimization by predicting structure–property–performance relationships from limited experimental data. Classical models such as random forests, gradient boost and support vector machines have successfully guided the design of lipid-based and polymeric systems [Han et al., 2018a, Eugster et al., 2024a, Aghajanpour et al., 2025]. More recently, foundation models such as TabPFN have demonstrated accurate predictions on small heterogeneous datasets, highlighting the potential of pretrained models to address data scarcity in scientific applications [Hollmann et al., 2024]. Building on this progress, recent advances in LLMs open a new direction. LLMs can integrate structured experimental data with unstructured domain knowledge and adapt in real time to new tasks using in-context learning [Brown et al., 2020]. Moreover, inference-time scaling has been shown to increase precision in biomedical and scientific domains [Wei et al., 2022, Wang et al., 2025a]. These characteristics suggest that LLMs could act as *virtual instruments*: flexible tools that can be tuned at run-time to balance accuracy and resource constraints.

Long-tail problems are tasks that occur infrequently, often with sparse and heterogeneous data, but carry high practical or scientific value; pharmaceutical formulation exemplifies this because each drug requires unique solutions from limited datasets, making predictive modeling especially challenging. [Han et al., 2018a, Eugster et al., 2024a]. Recent work by Wang et al. [2025a] shows that GPT-5 improves multimodal clinical reasoning, integrating structured indicators, patient narratives, and imaging data without domain-specific fine-tuning. This illustrates how LLMs can adapt to low-frequency but high-value problems by leveraging pretrained generalist reasoning capabilities.

In this work, we benchmark GPT and Claude on the open SEDDS dataset [Zaslavsky and Allen, 2023] for three outcomes, droplet size, API mass fraction, and PDI, formatting each record as a standardized *formulation card*. We evaluate two regimes: (i) within-API (bootstrapping on a known active) and (ii) between-API (cold-start on a held-out active). Increasing few-shot context consistently improves performance, with larger models narrowing the gap to domain-optimized baselines. These results position LLMs as data-efficient tools for guiding formulation and provide a reproducible benchmark and foundation for hybrid systems that integrate optimization, automation, and experimental feedback.

2 Related Work

ML for SEDDS and other lipid-based systems. Early work showed that classical ML can meaningfully accelerate SEDDS design. Gao et al. [2021b] integrated random forests with experimental validation to predict self-emulsifying regions in pseudo-ternary phase diagrams (4,495 compositions), achieving strong accuracy before down-selecting candidates for meloxicam SEDDS. FormulationAI built a platform to (e.g., Dong et al. [2023]) aggregate tasks across formulation types to provide ready-to-use predictors for critical quality attributes. Domain reviews likewise survey how ML is being adopted for formulation science and controlled release, with increasing emphasis on reproducibility and standardized evaluation [Gormley, 2024b].

ML for other formulation classes. Recent studies apply supervised learning to predict critical attributes and process settings in liposomal systems [Eugster et al., 2024b]. For polymeric drug-delivery systems, several modeling approaches have been used to predict release profiles and other properties, complementing models with data-driven targets [Aghajanpour et al., 2025, Abdalla et al., 2024]. Work on oral solid dosage forms demonstrates that neural networks can learn formulation–performance mappings from small heterogeneous datasets; e.g., Han et al. [2018b] predicted orally disintegrating tablets (ODT) disintegration from $n = 145$ formulations. In the context of lipid nanoparticles, the AI-Guided Ionizable Lipid Engineering (AGILE) platform exemplifies a synergistic deep learning–combinatorial chemistry strategy, where neural networks screen vast in silico lipid libraries to rapidly identify ionizable lipids tailored for cell-specific mRNA delivery [Xu et al., 2024].

LLMs as generalist property predictors. Recent generalist, drug discovery, and therapeutics-focused LLMs (e.g., TxGemma) implement a unified prompting schema for

prediction tasks spanning classification, regression, and generation on the Therapeutics Data Commons (TDC) [Huang et al., 2021]. The prediction variants format each example as *Instruction / Context / Question* and return a numeric prediction (for regression, via binned outputs). They evaluate performance using the TDC splits using few-shot mixed with zero-shot prompts [Wang et al., 2025b]. In this paper, we use an analogous regression setup for our LLM baselines: we serialize SEDDS "formulation cards" into the same narrow prompt template, while varying few-shot exemplars and inference-time reasoning effort. This alignment makes our results directly comparable to prior LLM-as-predictor studies.

Formulation science has intrinsically long-tail problems in that each API–excipient–process combination is infrequently observed. Contemporary evaluations of frontier models in other long-tail biomedical domains (e.g., multimodal medical QA/VQA) show that scale and inference-time reasoning can close gaps despite sparse, heterogeneous supervision. A recent study demonstrated that GPT-5 significantly improves clinical reasoning over GPT-4 on MedQA, USMLE samples, and MedXpertQA, using a chain-of-thought prompting approach [Wang et al., 2025a]. Our within-API and held-out-API regimes instantiate the same long-tail pattern in formulation, motivating the use of few-shot prompting and increased reasoning compute as practical levers when labeled data are scarce.

3 Dataset

We evaluate on the public dataset of formulation compositions introduced by Zaslavsky and Allen [2023], which aggregates SEDDS from the literature. After screening 307 articles, the authors retained 152 sources, yielding 668 unique formulations spanning 20 poorly water-soluble APIs. Each record enumerates the drug and all excipients (oils, surfactants, cosolvents, and other additives) with mass fractions standardized to sum to 100% w/w, plus selected formulation properties. A companion data deposit provides the cleaned tables and code used to construct the resource (OSF project page; [Zaslavsky and Allen, 2023]). See S1 for more details.

The dataset includes (i) formulation-level measurements/descriptors—droplet size (`size`, nm), polydispersity (`PDI`), counts and total fractions of excipient classes (`oil_total`, `surfactant_total`, `cosolvent_total`), LFCS features including weight-averaged surfactant HLB (`s_HLB`), and a min–max normalized complexity score; (ii) API physicochemical properties from public sources (e.g., molecular weight, logP, melting point, aqueous solubility, polar surface area, rotatable bonds, HBD/HBA); and (iii) excipient summaries (oil chain length-/saturation; for cosolvents, weight-averaged molecular weight, melting/boiling point, density, viscosity). Trade names were harmonized to chemical names. Reporting is incomplete: `size` is available for 506/668 (75.7%) formulations and `PDI` for 289/668 (43.3%) [Zaslavsky and Allen, 2023]. We analyze three continuous targets: `size`, API mass fraction (`Total Content`), and `PDI`.

Regarding limitations to the dataset, because entries are literature-extracted, the corpus inherits sparsity and reporting biases from source studies (e.g., missing `size`/`PDI` for some formulations) and variability in experimental protocols. These characteristics motivate robust evaluation and the inclusion of rank-based metrics in our analyses, which are less sensitive to absolute value discrepancies and better reflect practical decision-making in formulation screening. A further caveat is potential data exposure during pretraining: while some assay results may have been encountered by models, they would appear in highly unstructured form and often require nontrivial inference or calculation to be usable. We therefore view leakage risk as limited in practice, though it remains an important consideration.

4 Methods

4.1 Evaluation Settings: Two Deployment Scenarios

We use two complementary evaluation settings that are inspired by realistic deployment conditions in formulation design. **Experiment 1: within-API generalization** approximates a scenario in which an initial batch of experiments on a specific API is available to bootstrap a predictive model that will be used to prioritize downstream experiments for

the *same* API. **Experiment 2: between-API generalization** approximates a *cold-start* setting in which no experiments exist for a target API and the model must leverage prior data from other APIs to make initial predictions. All experiments used a fixed random seed to ensure reproducibility.

4.1.1 Within-API generalization.

We take a subset for each API and split into train/test folds. When feasible, we stratify the data to preserve outcome balance across folds; otherwise we perform an unstratified shuffle split. Few-shot exemplars for the LLM are sampled *exclusively* from the training fold; test rows are never used as shots. For regression targets (droplet size, API content, PDI), we fit decoding scalers (per target) on the training fold of API a and use them to inverse-transform predictions at report time.

4.1.2 Between-API generalization.

For each held-out API, few-shot exemplars are drawn from the pooled training set comprising all non-held out API. Regression scalers are fit on the pooled training data and used to decode predictions for the held-out API.

4.2 Prompting and Output Schema

4.2.1 Formulation cards.

Each example is rendered as a compact *formulation card* that serializes the record into `key:value` pairs. Cards concatenate identification fields (article, DOI, index, source), API physicochemical descriptors (e.g., molecular weight, $\log P$, melting point, water solubility, polar surface area, rotatable bonds, H-bond donors/acceptors), component identities (oil/surfactant/cosolvent/other) with their fractions, and summary totals/counts (e.g., `oil_total`, `surfactant_total`, `c_num`, `cplx_minmax_norm`). Observed outcomes (`size`, `Total Content`, `PDI`) and any identifiable source information (i.e. reference title, DOI) are *never* shown on any card.

Formulation Card

```
Formulation Card | API_id: cannabidiol

API_SMILES: CCCCCC1=CC(=C(C(=C1)O)[C@@H]2C=C(CC[C@H]2C(=C)C)C)O API_mol_wt: 314.47 logp_chemaxon: 6.33
API_melt_temp:  67  API_water_sol:  0.01  API_polar_sa:  40.46  API_rot_bond:  6  API_H_bond_donor:  2
API_H_bond_accept: 2

oil_id1: Glyceryl monolinoleate surfactant_id1: PEG-40 hydrogenated castor oil cosolvent_id1: PEG 400
other_id1: Sweetener other_id2: Flavoring

oil_prop1: 56.56 surfactant_prop1: 30 cosolvent_prop1: 8 other_prop1: 0.01 other_prop2: 0.01

oil_total: 56.56 surfactant_total: 30 cosolvent_total: 8 other_total: 0.02

o_num: 1, s_num: 1, c_num: 1, other_num: 2 cplx_minmax_norm: 0.38

Content: {"size_code": 265, "Total Content_code": 211, "pdi_code": 336}
```

4.2.2 Few-shot prompting.

Inference uses a chat-style sequence with K training cards (each followed by an assistant answer) and one unlabeled test card. Training answers contain integer *codes* for each target (see Y-scaling below). Test cards contain only features (no labels or outcomes). The final user instruction enforces a strict, single-line JSON output with fixed keys; parsed outputs are schema-validated and any out-of-range codes are clipped to the allowed domain.

4.3 Regression Setup

Following the setup in [Wang et al., 2025b], regression predicts integer code indices `size_code`, `api_prop_code`, `pdi_code` $\in [0, 999]$; real-valued targets are recovered by inverse-

transform. For each target y and training fold, we fit a min–max scaler over $[y_{\min}, y_{\max}]$. *Encoding:* continuous labels are linearly quantized to integer codes in $[0, 999]$, with clipping outside the range. *Decoding:* predicted codes are mapped back to real units using the same scaler (fit on the training fold for within-API or the pooled non-held-out training data for between-API).

4.4 Inference-Time Scaling and K-Shot Scaling

We evaluate $K \in \{0, 5, 10, 20\}$. K=0 measures zero-shot transfer from the card schema and system instructions alone; K>0 measures in-context learning from matched-format exemplars. Shots are sampled without replacement from the applicable training pool; if fewer than K rows exist, we use all available rows. For models that expose a `reasoning_effort` (or equivalent) control, we sweep {`low`, `medium`, `high`} to modulate test-time compute. Inference-time scaling runs were only performed with K=0 and K=10.

4.5 Models

We benchmarked a recent frontier model (`GPT-5`) along with two smaller variants (`GPT-5-mini` and `GPT-5-nano`), under identical schemas and decoding settings. When supported, we used the model's native `reasoning_effort` control; otherwise we applied the directive-based mimic described above. In addition, we included `Claude-4-Sonnet` for comparison. For all models, we report Spearman's ρ, Pearson's r, mean absolute error (MAE), and root mean squared error (RMSE). Due to time constraints, we limited our evaluation to this set of models; however, extending the analysis to `Claude-4-Opus` and other frontier releases remains a promising direction for future work.

5 Results

5.1 Within-API Experiments

In the within-API setting, models showed improved performance from few-shot conditioning. As shown in Figure 1, Pearson and Spearman correlations improved consistently when moving from zero-shot ($K = 0$) to few-shot prompting ($K = 10$). For example, droplet size prediction improved from $r = 0.17$ at $K = 0$ to $r = 0.48$ at $K = 10$, with a corresponding decrease in RMSE from 262 nm to 237 nm. API mass fraction also showed strong gains, with Pearson r rising from near zero (0.07) at $K = 0$ to 0.58 at $K = 10$. Polydispersity index (PDI) remained the most challenging target but still showed moderate improvements in correlation (from -0.05 to 0.40). Overall, these results indicate that within-API few-shot prompting allows LLMs to adapt effectively to the local formulation space, producing meaningful accuracy improvements. Among all evaluated models, claude-sonnet-4 consistently achieved the best performance across most settings. For complete results see Table 3. Example calibration curves for cannabidiol, cyclosporine and paclitaxel can be seen in Figure S3. Comparing to classical baselines (Table 5), ridge regression performs well for Total Content and linear SVR for Droplet Size than the best LLM (claude-sonnet-4), indicating that this target's signal is well captured by low-capacity tabular models under the 20-shot regime (Table 5).

5.2 Between-API Experiments

In the between-API setting, where predictions were made for unseen formulations, performance was lower overall. Correlations for API mass fraction and droplet size were near zero in the zero-shot case, and although some modest improvements were observed with few-shot prompts, the gains were much smaller than in the within-API regime (See Figure S2 for results). For instance, droplet size correlation rose only from $r = 0.14$ to $r = 0.24$ when moving from $K = 0$ to $K = 10$, despite a large reduction in RMSE (from 1952 nm to 929 nm). Similarly, PDI correlations remained low, ranging between 0.07 and 0.35 across conditions. These findings suggest that cross-API transfer remains difficult, and most of the predictive benefits arise from in-domain adaptation rather than broad generalization. Notably, claude-sonnet-4 performed particularly well for total content prediction, standing out relative to other models in this more challenging setting. For complete results, see Table 4.

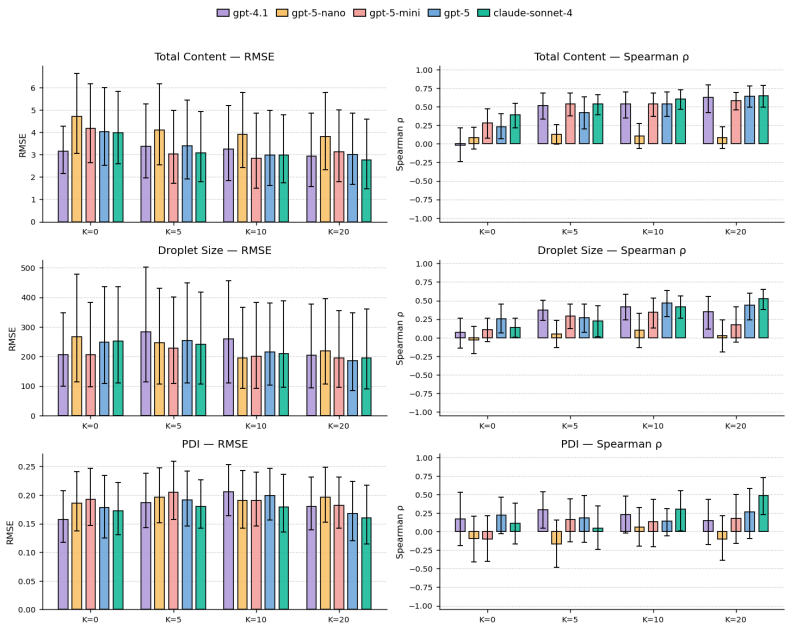

Figure 1: **Within-API performance across few-shot context (K).** The left column reports RMSE (lower is better) and the right column reports Spearman's ρ (higher is better) for total content, droplet size, and polydispersity index (PDI) across $K \in \{0, 5, 10, 20\}$ and model sizes. Models include GPT-4.1, GPT-5-nano, GPT-5-mini, GPT-5, and claude-sonnet-4. Increasing few-shot context generally reduces RMSE and increases rank correlation, indicating effective rank-ordering of formulations within a known API along with improved absolute calibration. Gains are target-dependent but broadly similar across the two metrics, suggesting LLMs provide useful signals for both prioritization and approximate prediction.

By contrast, in the between-API setting LLMs—especially claude-sonnet-4—substantially outperform classical baselines on Total Content, suggesting better cross-API transfer when no same-API exemplars are available (Table 6).

5.3 Reasoning Effort

Reasoning Traces

API ID: cannabidiol k = 0

CBD (logP 6.3) with Tween 80/Span 80 (HLB 11.4) and high oil (43%) but no cosolvent results in mid-sized droplets; strong solubilization in nonionic surfactants results in moderate-high API %; mixed surfactants improve homogeneity but high oil/low S:O ratio yields moderate PDI.

API ID: cannabidiol k = 10

Coconut oil with Tween 80/Span 80, no cosolvent; lowest surfactant (37.5%) in this series → slightly larger droplets than index 5/8 but likely below the worst (index 7). High oil fraction with very lipophilic CBD → near-top API loading. Reduced surfactant drives broader distribution → higher PDI. Size Prediction: 233.43, Total Content Prediction: 17.35, PDI Prediction: 0.31

We also evaluated the effect of varying inference-time reasoning effort (low, medium, high). As summarized in Table 1 and Table 2, higher reasoning effort generally yielded modest but improved correlations and reduced error, particularly in the within-API case. Between-API results also showed occasional benefits from higher reasoning effort, though improvements were smaller and less consistent. Interestingly, qualitative inspection of the reasoning traces revealed a complementary difference across retrieval depth: at $K = 0$, the model relied exclusively on general formulation knowledge (e.g., HLB balance, lipophilicity, surfactant-to-oil ratio) to justify its predictions, whereas at $K = 10$ it combined such mechanistic

reasoning with comparative context from other experimental indices, producing both relative judgments and quantitative predictions. This highlights how increased reasoning effort not only improves accuracy but also shifts how the model reasons, from principle-based heuristics toward more context-aware and data-grounded inference.

Table 1: Within-API Experiments

Reasoning	Target	k=0				k=10			
		MAE	RMSE	Pearson r	Spearman ρ	MAE	RMSE	Pearson r	Spearman ρ
Low	Total Content	3.035	4.105	0.067	0.123	1.865	3.095	0.582	0.601
	PDI	0.156	0.181	-0.054	-0.119	0.155	0.186	0.404	0.195
	Size	183.161	262.119	0.175	0.138	123.317	237.194	0.479	0.522
Medium	Total Content	2.987	4.040	0.216	0.231	1.743	2.991	0.566	0.538
	PDI	0.157	0.178	0.272	0.222	0.164	0.199	0.391	0.140
	Size	164.612	248.405	0.233	0.257	113.486	215.071	0.433	0.466
High	Total Content	2.873	4.057	0.246	0.297	1.646	2.902	0.582	0.541
	PDI	0.150	0.174	0.142	0.148	0.172	0.203	0.359	0.297
	Size	168.780	254.247	0.158	0.190	110.843	204.226	0.438	0.385

Table 2: Between-API Experiments

Reasoning	Target	k=0				k=10			
		MAE	RMSE	Pearson r	Spearman ρ	MAE	RMSE	Pearson r	Spearman ρ
Low	Total Content	8.439	9.594	0.116	0.134	4.659	5.349	0.030	0.107
	PDI	0.143	0.177	0.211	0.220	0.166	0.207	0.070	0.069
	Size	1862.207	1952.349	0.141	0.169	682.247	928.515	0.178	0.243
Medium	Total Content	10.391	12.869	0.097	0.125	6.110	8.919	0.072	-0.001
	PDI	0.150	0.187	0.173	0.124	0.116	0.147	0.346	0.396
	Size	1535.725	1701.486	0.159	0.116	1340.459	1578.218	0.085	0.174
High	Total Content	9.875	11.629	0.204	0.121	5.083	6.171	0.094	0.071
	PDI	0.140	0.182	0.196	0.193	0.155	0.197	0.009	0.014
	Size	1522.665	1689.329	0.152	0.180	396.811	581.397	-0.028	-0.022

6 Discussion

Our results demonstrate that LLMs can provide useful predictive power in formulation science when framed as *virtual instruments*. The within-API experiments show that few-shot prompting substantially improves correlations with experimental outcomes. In particular, droplet size and API mass fraction predictions benefited most from in-context examples, indicating that local adaptation to a single API's formulation space is feasible with relatively little data. This suggests that in early-stage formulation campaigns, LLMs can function as rapid bootstrapping tools to prioritize experiments before more specialized models are trained.

Within-API, simpler models like ridge and random forest marginally match or exceed LLMs on Total Content, consistent with an easier, largely additive structure in this endpoint. However, for between-API generalization LLMs recover more signal, likely because the card prompt lets them leverage pretrained physico-chemical priors across APIs. Of note, claude-sonnet-4 outperformed GPT-5 in $K = 0$ setting for both experimental conditions. Furthermore, claude-sonnet-4 outperformed all other models across K settings for the Total Content task, though had mixed performance for other tasks.

Although RMSE values improved with more exemplars and higher reasoning effort, correlation gains were modest. This aligns with observations from other biomedical domains, where we are beginning to see improvements in long-tail, heterogeneous problems for generalist models [Wang et al., 2025a]. These results suggest that hybrid strategies like fine-tuning open-source models, integrating retrieval mechanisms, or combining LLMs with mechanistic simulations may be necessary to achieve robust generalization across APIs.

Increased inference-time reasoning effort led to some but limited improvement in predictive accuracy in some tasks, however the results were inconsistent. Some effect was seen in the within-API case, where higher reasoning effort reduced error and slightly increased some correlation for PDI. The $K = 0$ setting showed a greater increase in performance as a

function of inference-time compute, suggesting that the few-shot condition may have resulted in the models overindexing on information in the context window instead of attempting to reason over their internal knowledge of drug formulation. These patterns indicate that inference-time scaling may serve as a practical lever for boosting predictive accuracy in some contexts; however, further work is needed to explore how impactful this method is. Further finetuning with domain-specific data could provide a better prior for models to reason over. Interestingly, our qualitative inspection of reasoning traces suggested a complementary shift across retrieval depths. At $K = 0$, predictions were justified primarily through general formulation heuristics (e.g., HLB balance, lipophilicity, surfactant-to-oil ratio), whereas at $K = 10$ the model integrated such mechanistic reasoning with comparative context from experimental indices, yielding more data-grounded and context-aware inferences. This indicates that increased reasoning effort not only enhances predictive accuracy but also alters the character of the reasoning process itself, moving from principle-based heuristics toward a synthesis of mechanistic and empirical evidence.

A notable point is that simpler baselines, such as ridge regression and random forest, sometimes matched or outperformed LLMs in the within-API setting, particularly for Total Content. This highlights that the added complexity and cost of frontier LLMs are not always warranted when low-capacity models suffice. We view LLMs instead as complementary tools: classical methods capture additive structure efficiently, while LLMs contribute adaptability, few-shot transfer, and integration of heterogeneous prior knowledge. Of course, post-training efforts and use of modern reinforcement learning approaches could yield significantly better results.

Limitations. A methodological limitation of our evaluation lies in the choice of performance metrics. We reported both correlation-based measures (Spearman's ρ, Pearson's r) and error-based measures (RMSE, R^2). While RMSE provides an interpretable estimate of absolute deviation in the physical units of each assay, R^2 quantifies variance explained relative to a mean-predictor baseline and can be negative in noisy, heterogeneous settings. This means that models showing useful rank-order signal (high ρ) may nonetheless have poor or negative R^2, potentially understating their utility for prioritization in early-stage formulation campaigns. R^2 values are reported in the supplementary information, and many of them are negative, suggesting more work is needed to move this type of approach from being useful in ranking formulations to directly predicting the outcomes of experiments.

Another limitation lies in our evaluation of context length. We restricted retrieval depths and reasoning effort to relatively small scales. Recent work on many-shot in-context learning [Agarwal et al., 2024] demonstrates that models can achieve significant gains when provided with hundreds or thousands of in-context examples, sometimes approaching fine-tuning performance. Exploring this many-shot regime could further improve scientific predictive tasks like those considered here. However, such evaluations require large volumes of curated demonstrations, which we leave for future work.

7 Conclusion

This study establishes baselines on an open source SEDDS dataset, targeting droplet size, API mass fraction, and polydispersity index. By representing each formulation as a structured *formulation card* and benchmarking across multiple LLM families, we show that frontier models can adapt effectively to within-API prediction tasks through few-shot prompting and inference-time scaling. Between-API generalization remains limited, underscoring the need for hybrid approaches that combine pretrained reasoning with domain-specific adaptation.

These findings position frontier LLMs as *data-efficient, tunable instruments* for guiding drug formulation experiments in resource-constrained discovery pipelines. The baselines also provide a reference framework against which customized open-source models—leveraging fine-tuning, retrieval augmentation, or integration with experimental feedback—can be evaluated under consistent conditions. Looking forward, coupling foundation models with domain-specific automation points toward general-purpose engines for drug formulation under data scarcity.

References

Haoshi Gao, Haoyue Jia, Jie Dong, Xinggang Yang, Haifeng Li, and Defang Ouyang. Integrated in silico formulation design of self-emulsifying drug delivery systems. *Acta Pharmaceutica Sinica B*, 11(10):3585–3594, 2021a. doi: 10.1016/j.apsb.2021.04.017. URL https://doi.org/10.1016/j.apsb.2021.04.017.

Adam J. Gormley. Machine learning in drug delivery. *Journal of Controlled Release*, 373:23–30, 2024a. doi: 10.1016/j.jconrel.2024.06.045. URL https://doi.org/10.1016/j.jconrel.2024.06.045.

Run Han, Yilong Yang, Xiaoshan Li, and Defang Ouyang. Predicting oral disintegrating tablet formulations by neural network techniques. *Asian Journal of Pharmaceutical Sciences*, 13(4):336–342, 2018a. doi: 10.1016/j.ajps.2018.01.003. URL https://doi.org/10.1016/j.ajps.2018.01.003.

Remo Eugster, Markus Orsi, Giorgio Buttitta, Nicola Serafini, Mattia Tiboni, Luca Casettari, Jean-Louis Reymond, Simone Aleandri, and Paola Luciani. Leveraging machine learning to streamline the development of liposomal drug delivery systems. *Journal of Controlled Release*, 2024a. doi: 10.1016/j.jconrel.2024.10.065. URL https://doi.org/10.1016/j.jconrel.2024.10.065. Published online.

Sareh Aghajanpour, Hamid Amiriara, Mehdi Esfandyari-Manesh, Pedram Ebrahimnejad, Haziq Jeelani, Andreas Henschel, Hemant Singh, Rassoul Dinarvand, and Shabir Hassan. Utilizing machine learning for predicting drug release from polymeric drug delivery systems. *Computers in Biology and Medicine*, 188:109756, April 2025. doi: 10.1016/j.compbiomed.2025.109756. URL https://doi.org/10.1016/j.compbiomed.2025.109756.

Nils M. Hollmann, Samuel Mühlbauer, Robert Schmier, Sebastian U. Stich, Sergei V. Ivanov, Anian Ruoss, Arne Nix, Maximilian Schleich, Andreas Krause, and José Miguel Hernández-Lobato. Accurate predictions on small data with a tabular foundation model. *Nature*, 628 (8006):764–771, 2024. doi: 10.1038/s41586-024-08328-6.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, 2020. URL https://arxiv.org/abs/2005.14165. NeurIPS 2020.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Le, Maarten Bosma, Brian Ichter, Fei Xia, Dale Zhou, Ed Li, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint*, arXiv:2201.11903, 2022. URL https://doi.org/10.48550/arXiv.2201.11903.

Shansong Wang, Mingzhe Hu, Qiang Li, Mojtaba Safari, and Xiaofeng Yang. Capabilities of gpt-5 on multimodal medical reasoning. *arXiv preprint arXiv:2508.08224*, 2025a. URL https://arxiv.org/abs/2508.08224.

Jonathan Zaslavsky and Christine Allen. A dataset of formulation compositions for self-emulsifying drug delivery systems. *Scientific Data*, 10(1):914, Dec 2023. doi: 10.1038/s41597-023-02812-w. URL https://doi.org/10.1038/s41597-023-02812-w.

Haoshi Gao, Haoyue Jia, Jie Dong, Xinggang Yang, Haifeng Li, and Defang Ouyang. Integrated in silico formulation design of self-emulsifying drug delivery systems. *Acta Pharmaceutica Sinica B*, 11(11):3585–3594, 2021b. doi: 10.1016/j.apsb.2021.05.014.

Jie Dong, Zheng Wu, Huanle Xu, and Defang Ouyang. Formulationai: a novel web-based platform for drug formulation design driven by artificial intelligence. *Briefings in Bioinformatics*, 25(1):bbad419, 2023. doi: 10.1093/bib/bbad419.

Adam J. Gormley. Machine learning in drug delivery. *Journal of Controlled Release*, 373: 23–30, 2024b. doi: 10.1016/j.jconrel.2023.12.007.

Remo Eugster, Markus Orsi, Giorgio Buttitta, Nicola Serafini, Mattia Tiboni, Luca Casettari, Jean-Louis Reymond, Simone Aleandri, and Paola Luciani. Leveraging machine learning to streamline the development of liposomal drug delivery systems. *Journal of Controlled Release*, 2024b. doi: https://doi.org/10.1016/j.jconrel.2024.10.065.

Youssef Abdalla, Laura E. McCoubrey, Fabiana Ferraro, Lisa Maria Sonnleitner, et al. Machine learning of raman spectra predicts drug release from polysaccharide coatings for targeted colonic delivery. *Journal of Controlled Release*, 2024. doi: https://doi.org/10.1016/j.jconrel.2024.08.010.

Run Han, Yilong Yang, Xiaoshan Li, and Defang Ouyang. Predicting oral disintegrating tablet formulations by neural network techniques. *Asian Journal of Pharmaceutical Sciences*, 13(6):575–582, 2018b. doi: https://doi.org/10.1016/j.ajps.2018.01.003.

Yue Xu, Shihao Ma, Haotian Cui, Jingan Chen, Shufen Xu, Fanglin Gong, Alex Golubovic, Muye Zhou, Kevin Chang Wang, Andrew Varley, Rick Xing Ze Lu, Bo Wang, and Bowen Li. AGILE platform: a deep learning powered approach to accelerate LNP development for mRNA delivery. *Nature Communications*, 15:6305, jul 2024. doi: 10.1038/s41467-024-50619-z. URL https://doi.org/10.1038/s41467-024-50619-z. Published 26 July 2024.

Kexin Huang, Tianfan Fu, Wenhao Gao, Yue Zhao, Yusuf Roohani, Jure Leskovec, Connor W. Coley, Cao Xiao, Jimeng Sun, and Marinka Zitnik. Therapeutics data commons: Machine learning datasets and tasks for drug discovery and development. *arXiv preprint arXiv:2102.09548*, 2021. doi: 10.48550/arXiv.2102.09548. URL https://doi.org/10.48550/arXiv.2102.09548. Published at NeurIPS 2021 Datasets and Benchmarks Track.

Eric Wang, Samuel Schmidgall, Paul F. Jaeger, Fan Zhang, Rory Pilgrim, Yossi Matias, Joelle Barral, David Fleet, and Shekoofeh Azizi. Txgemma: Efficient and agentic llms for therapeutics. *arXiv preprint arXiv:2504.06196*, 2025b. URL https://arxiv.org/abs/2504.06196.

Rishabh Agarwal, Avi Singh, Lei M. Zhang, Bernd Bohnet, Luis Rosias, Stephanie Chan, Biao Zhang, Ankesh Anand, Zaheer Abbas, Azade Nova, John D. Co-Reyes, Eric Chu, Feryal Behbahani, Aleksandra Faust, and Hugo Larochelle. Many-shot in-context learning, 2024. URL https://arxiv.org/abs/2404.11018.

8 Supplementary Material

8.1 Benchmarking Dataset Description

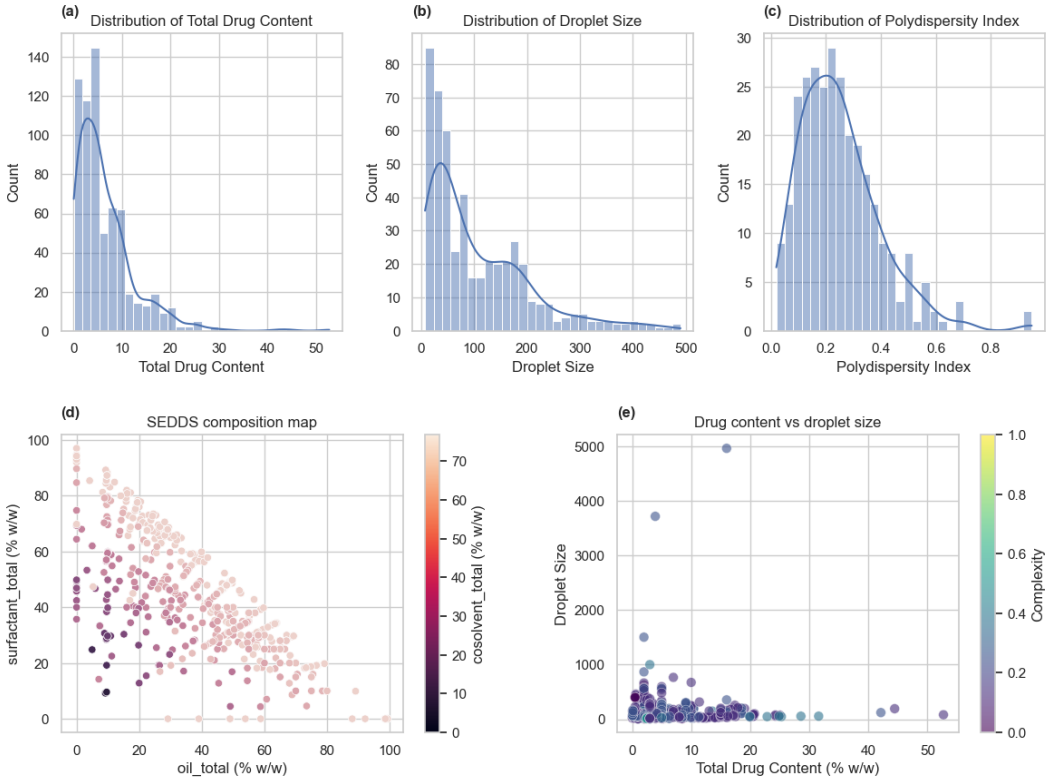

Figure S1: **SEDDS dataset characteristics.** (**a**) Distribution of API mass fraction (% w/w). (**b**) Distribution of droplet size (nm). (**c**) Distribution of PDI. (**d**) Composition map showing surfactant vs. oil totals, colored by cosolvent fraction. (**e**) Scatter of API mass fraction vs. droplet size, colored by PDI. Distributions are heterogeneous and heavy-tailed, motivating rank-based evaluation alongside absolute error.

8.2 System Prompt

System Prompt

"api": "cannabidiol"
"row_index": 1 "role": "system"
"content": "Reasoning effort: medium."
"content": "You are an automated virtual formulation instrument. Your task is NUMERIC REGRESSION for SEDDS data.

Given a 'Formulation Card', predict three continuous outcomes:

1) size_nm (droplet size, in nm)

2) Total Content_pct (API %w/w in formulation)

3) PDI (polydispersity index, unitless)

Output via code indices: emit integers in [0, 999] which will be mapped back to the data scale.

- size_code ∈ [0,999] maps to size_nm via per-fold min/max scaling

- Total Content_code ∈ [0,999] maps to Total Content (%w/w) via per-fold min/max scaling

- pdi_code ∈ [0,999] maps to PDI via per-fold min/max scaling

Output RULES (must follow):

- Respond with a SINGLE LINE of STRICT JSON, no prose, no backticks.n - Exactly these keys: {"size_code", "Total Content_code", "pdi_code", "reasoning"}.

- Values for codes must be integers between 0 and 999 (inclusive). If uncertain, choose your best single estimate.

- Reasoning trace should be a brief justification for each prediction."

8.3 Between-API Results

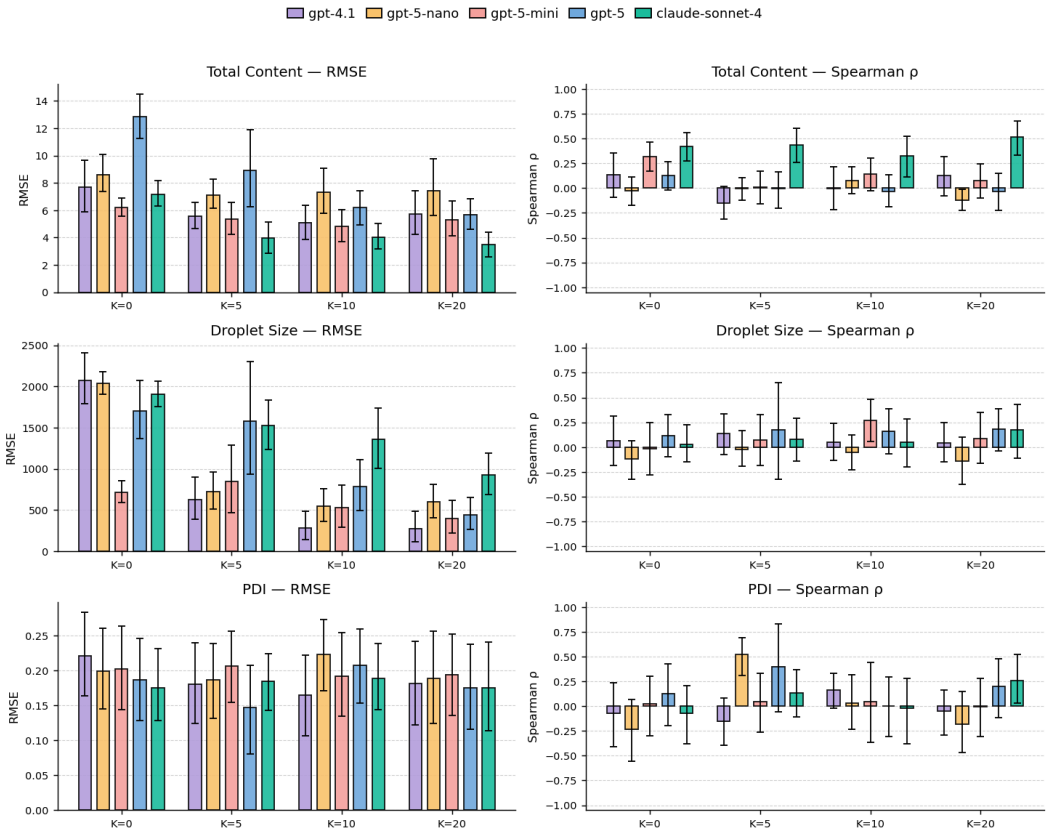

Figure S2: **Comparison of model performance across prediction targets**. Left column shows RMSE, right column shows Spearman ρ. Bars indicate mean with error bars for standard deviation. Among the evaluated models (`gpt-4.1`, `gpt-5-nano`, `gpt-5-mini`, `gpt-5`, and `claude-sonnet-4`), performance is broadly similar across droplet size and PDI. However, for the Total Content task, `claude-sonnet-4` achieves consistently lower RMSE and higher rank correlations, significantly outperforming all GPT-based models.

8.4 Within-API calibration curves for cannabidiol, cyclosporine and paclitaxel

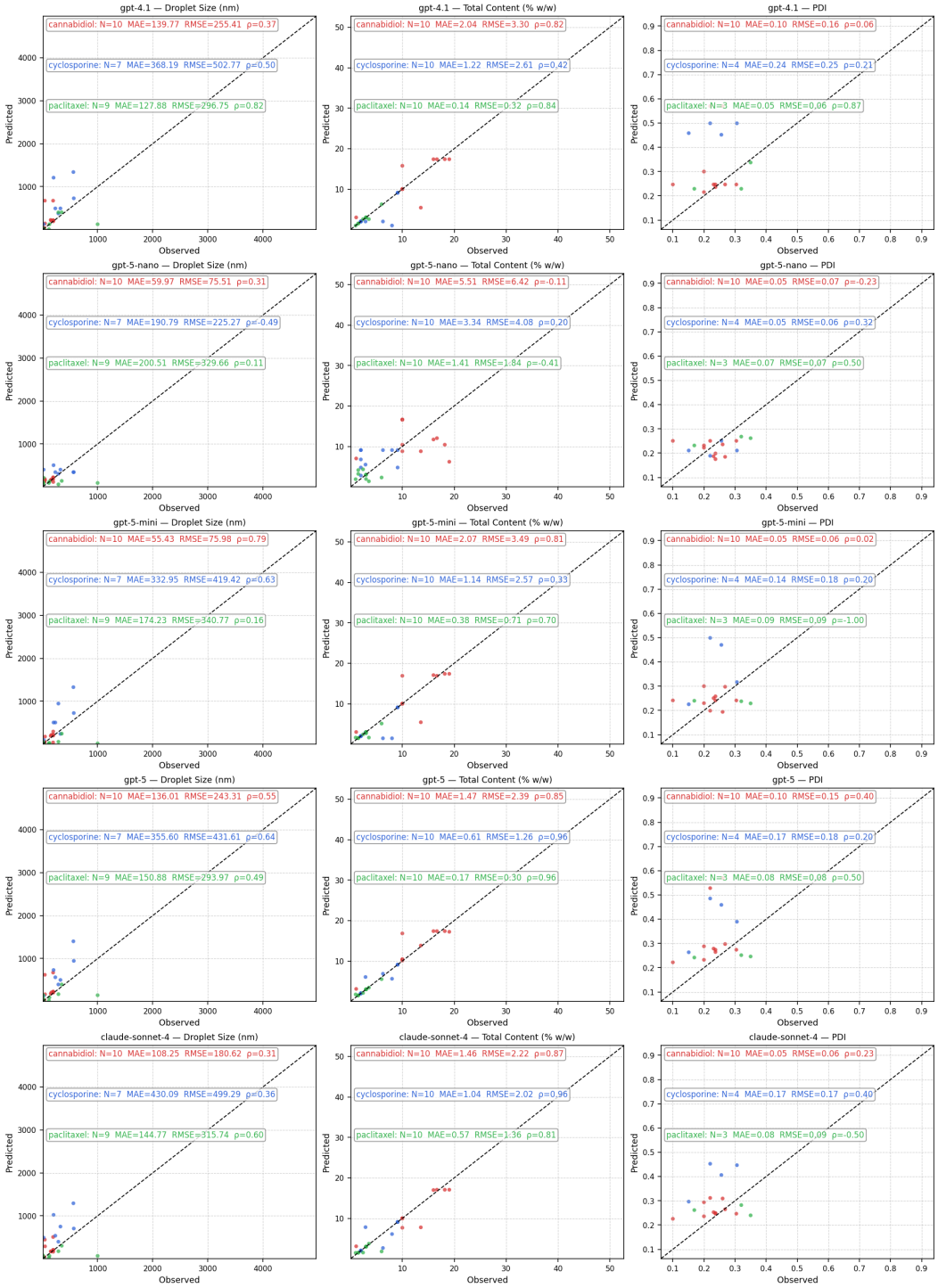

Figure S3: **Observed vs. predicted parity plots across models.** Columns show droplet size (nm), Total Content (% w/w), and PDI; rows show GPT-4.1, GPT-5-nano, GPT-5-mini, GPT-5, and claude-sonnet-4. Each point is a formulation; colors denote API (cannabidiol, cyclosporine, paclitaxel). The dashed line is $y = x$. Inset boxes report per-API N, MAE, RMSE, and Spearman's ρ.

Table 3: Within-API Performance Metrics

Task	Model	k	MAE	RMSE	R^2	Pearson r	Spearman ρ
Total Content	claude-sonnet-4-20250514	0	2.8194	3.9788	-0.8745	0.3974	0.3905
Total Content	claude-sonnet-4-20250514	5	1.8895	3.0795	-0.0259	0.5747	0.5385
Total Content	claude-sonnet-4-20250514	10	1.7272	2.9916	-0.1060	0.5996	0.6003
Total Content	claude-sonnet-4-20250514	20	1.5688	2.7629	0.2060	0.6782	0.6503
Total Content	gpt-4.1	0	2.5880	3.1634	-1.3558	0.0149	-0.0180
Total Content	gpt-4.1	5	1.9985	3.3697	-0.1844	0.5202	0.5123
Total Content	gpt-4.1	10	1.8907	3.2617	-0.0488	0.5265	0.5351
Total Content	gpt-4.1	20	1.6687	2.9477	0.2436	0.6029	0.6261
Total Content	gpt-5	0	2.9869	4.0404	-0.7043	0.2164	0.2313
Total Content	gpt-5	5	2.1274	3.3933	0.0674	0.5042	0.4195
Total Content	gpt-5	10	1.7432	2.9911	0.1056	0.5658	0.5383
Total Content	gpt-5	20	1.7400	3.0218	0.2169	0.6734	0.6399
Total Content	gpt-5-mini	0	3.0515	4.1900	-1.1873	0.2072	0.2837
Total Content	gpt-5-mini	5	1.7008	3.0372	-0.1048	0.4748	0.5360
Total Content	gpt-5-mini	10	1.5072	2.8376	0.3204	0.5671	0.5367
Total Content	gpt-5-mini	20	1.6388	3.1394	0.1022	0.5470	0.5782
Total Content	gpt-5-nano	0	3.4912	4.7083	-1.6225	0.0653	0.0830
Total Content	gpt-5-nano	5	2.7691	4.0997	-0.4280	0.0512	0.1305
Total Content	gpt-5-nano	10	2.6542	3.9085	-0.3954	0.1064	0.1039
Total Content	gpt-5-nano	20	2.5556	3.8023	-0.3335	0.0565	0.0851
size	claude-sonnet-4-20250514	0	174.26	251.22	-3.4409	0.1273	0.1338
size	claude-sonnet-4-20250514	5	145.86	241.02	-2.5385	0.2861	0.2289
size	claude-sonnet-4-20250514	10	115.66	210.33	-0.9356	0.3527	0.4147
size	claude-sonnet-4-20250514	20	99.42	194.53	-0.6874	0.4386	0.5226
size	gpt-4.1	0	182.03	206.17	-8.1448	0.0725	0.0730
size	gpt-4.1	5	149.15	283.81	-7.2640	0.3075	0.3703
size	gpt-4.1	10	125.64	259.16	-4.7020	0.4077	0.4158
size	gpt-4.1	20	109.03	204.84	-1.4975	0.4070	0.3513
size	gpt-5	0	164.61	248.41	-2.8328	0.2332	0.2565
size	gpt-5	5	147.06	254.60	-3.3662	0.2191	0.2713
size	gpt-5	10	113.49	215.07	-1.2761	0.4328	0.4656
size	gpt-5	20	97.23	185.89	-0.5496	0.4245	0.4396
size	gpt-5-mini	0	115.94	206.68	-0.7059	0.1537	0.1048
size	gpt-5-mini	5	120.88	227.33	-1.6974	0.2612	0.2931
size	gpt-5-mini	10	106.63	199.93	-0.5009	0.3746	0.3447
size	gpt-5-mini	20	98.21	194.50	-0.8683	0.2273	0.1774
size	gpt-5-nano	0	169.34	267.06	-5.7573	0.0543	-0.0339
size	gpt-5-nano	5	152.10	246.60	-3.2865	0.1606	0.0472
size	gpt-5-nano	10	110.59	195.26	-0.4235	0.1468	0.0986
size	gpt-5-nano	20	124.42	218.88	-1.3678	-0.0111	0.0255
pdi	claude-sonnet-4-20250514	0	0.1501	0.1728	-2.7315	0.0586	0.1132
pdi	claude-sonnet-4-20250514	5	0.1521	0.1796	-3.6866	0.2142	0.0427
pdi	claude-sonnet-4-20250514	10	0.1498	0.1788	-3.9382	0.4724	0.3009
pdi	claude-sonnet-4-20250514	20	0.1318	0.1604	-7.0408	0.4869	0.4825
pdi	gpt-4.1	0	0.1379	0.1575	-7.4036	0.1646	0.1716
pdi	gpt-4.1	5	0.1544	0.1863	-9.6625	0.4229	0.2963
pdi	gpt-4.1	10	0.1673	0.2055	-10.2109	0.3150	0.2258
pdi	gpt-4.1	20	0.1480	0.1801	-8.5751	0.3232	0.1494
pdi	gpt-5	0	0.1565	0.1781	-2.3419	0.2717	0.2218
pdi	gpt-5	5	0.1614	0.1910	-11.1632	0.3103	0.1862
pdi	gpt-5	10	0.1642	0.1987	-8.5178	0.3908	0.1396
pdi	gpt-5	20	0.1384	0.1677	-4.8623	0.3077	0.2663
pdi	gpt-5-mini	0	0.1662	0.1924	-3.6453	-0.0897	-0.0982
pdi	gpt-5-mini	5	0.1669	0.2052	-9.7246	0.1880	0.1648
pdi	gpt-5-mini	10	0.1543	0.1904	-8.7816	0.1900	0.1302
pdi	gpt-5-mini	20	0.1519	0.1822	-5.4854	0.1193	0.1798
pdi	gpt-5-nano	0	0.1596	0.1858	-3.1441	-0.1302	-0.0960
pdi	gpt-5-nano	5	0.1680	0.1963	-3.6057	-0.0517	-0.1643
pdi	gpt-5-nano	10	0.1631	0.1901	-3.6277	0.0935	0.0598
pdi	gpt-5-nano	20	0.1673	0.1966	-2.8833	-0.1604	-0.1029

Table 4: Between-API Performance Metrics

Task	Model	k	MAE	RMSE	R^2	Pearson r	Spearman r
Total Content	claude-sonnet-4-20250514	0	5.8873	7.1753	-705.6458	0.3799	0.4237
Total Content	claude-sonnet-4-20250514	5	2.9593	3.9377	-264.2340	0.4640	0.4375
Total Content	claude-sonnet-4-20250514	10	2.7517	4.0411	-39.0974	0.2927	0.3264
Total Content	claude-sonnet-4-20250514	20	2.1430	3.4591	-99.0823	0.5429	0.5169
Total Content	gpt-4.1	0	6.2667	7.6927	-370.5290	0.1290	0.1329
Total Content	gpt-4.1	5	4.9673	5.5744	-309.7863	-0.2285	-0.1519
Total Content	gpt-4.1	10	4.5917	5.0867	-154.9700	-0.0498	-0.0025
Total Content	gpt-4.1	20	5.1158	5.7415	-88.2042	0.0941	0.1256
Total Content	gpt-5	0	10.3913	12.8693	-666.7677	0.0968	0.1246
Total Content	gpt-5	5	6.1101	8.9195	-114.0675	0.0721	-0.0013
Total Content	gpt-5	10	5.0628	6.2006	-357.4742	-0.0880	-0.0319
Total Content	gpt-5	20	4.7689	5.6824	-210.4958	-0.0540	-0.0314
Total Content	gpt-5-mini	0	4.4467	6.2230	-258.3669	0.3153	0.3208
Total Content	gpt-5-mini	5	4.1409	5.3680	-570.3347	0.1104	0.0099
Total Content	gpt-5-mini	10	3.7973	4.7900	-353.2790	0.1642	0.1387
Total Content	gpt-5-mini	20	4.2742	5.3075	-177.0229	0.1094	0.0782
Total Content	gpt-5-nano	0	6.5966	8.6166	-606.4734	-0.0461	-0.0271
Total Content	gpt-5-nano	5	6.0797	7.1032	-405.8569	-0.0151	-0.0075
Total Content	gpt-5-nano	10	6.3206	7.3266	-931.3219	0.0379	0.0793
Total Content	gpt-5-nano	20	6.4894	7.4225	-1099.7783	-0.0685	-0.1192
size	claude-sonnet-4-20250514	0	1842.43	1901.98	-268734.2675	-0.0053	0.0268
size	claude-sonnet-4-20250514	5	1370.18	1527.47	-293300.7698	0.0742	0.0786
size	claude-sonnet-4-20250514	10	1181.50	1354.46	-334621.5333	0.0270	0.0478
size	claude-sonnet-4-20250514	20	700.64	922.60	-31026.9049	0.1620	0.1706
size	gpt-4.1	0	2022.59	2076.73	-37214.5034	0.0364	0.0659
size	gpt-4.1	5	466.82	629.55	-6312.7787	0.1898	0.1338
size	gpt-4.1	10	173.73	285.83	-1063.3810	0.0441	0.0480
size	gpt-4.1	20	154.26	269.00	-1023.7675	0.0387	0.0430
size	gpt-5	0	1535.72	1701.49	-236705.5523	0.1589	0.1158
size	gpt-5	5	1340.46	1578.22	-688338.1982	0.0855	0.1738
size	gpt-5	10	526.30	781.26	-223614.6468	0.1536	0.1589
size	gpt-5	20	262.02	440.84	-26047.5048	0.1877	0.1782
size	gpt-5-mini	0	578.58	713.16	-20598.5659	-0.0149	-0.0130
size	gpt-5-mini	5	647.84	849.61	-100651.2091	0.1513	0.0744
size	gpt-5-mini	10	335.84	531.15	-25648.7516	0.2690	0.2720
size	gpt-5-mini	20	244.03	399.66	-16029.8568	0.1063	0.0884
size	gpt-5-nano	0	1850.90	2041.41	-165755.0890	-0.1749	-0.1189
size	gpt-5-nano	5	531.64	723.87	-15584.7386	0.0374	-0.0235
size	gpt-5-nano	10	351.81	547.52	-7602.1847	-0.0052	-0.0548
size	gpt-5-nano	20	412.90	600.41	-86279.9676	-0.0377	-0.1413
pdi	claude-sonnet-4-20250514	0	0.1481	0.1757	-1.7956	-0.1022	-0.0699
pdi	claude-sonnet-4-20250514	5	0.1571	0.1853	-2.3897	0.1695	0.1329
pdi	claude-sonnet-4-20250514	10	0.1516	0.1892	-1.9392	-0.1134	-0.0212
pdi	claude-sonnet-4-20250514	20	0.1366	0.1752	-0.8490	0.1871	0.2602
pdi	gpt-4.1	0	0.1921	0.2216	-4.8834	-0.0841	-0.0720
pdi	gpt-4.1	5	0.1542	0.1808	-2.0241	-0.1452	-0.1536
pdi	gpt-4.1	10	0.1376	0.1648	-1.3014	0.1054	0.1645
pdi	gpt-4.1	20	0.1509	0.1821	-0.9984	-0.0535	-0.0485
pdi	gpt-5	0	0.1503	0.1865	-1.3481	0.1733	0.1244
pdi	gpt-5	5	0.1160	0.1472	-7.4827	0.3460	0.3957
pdi	gpt-5	10	0.1722	0.2080	-2.4687	0.0053	0.0014
pdi	gpt-5	20	0.1337	0.1750	-0.5165	0.1012	0.1988
pdi	gpt-5-mini	0	0.1572	0.2022	-1.8488	0.0691	0.0198
pdi	gpt-5-mini	5	0.1677	0.2064	-2.0086	0.0460	0.0427
pdi	gpt-5-mini	10	0.1505	0.1925	-1.2788	0.0581	0.0471
pdi	gpt-5-mini	20	0.1544	0.1940	-1.1867	-0.0609	-0.0047
pdi	gpt-5-nano	0	0.1591	0.1997	-1.7104	-0.1685	-0.2358
pdi	gpt-5-nano	5	0.1508	0.1865	-1.4144	0.5069	0.5188
pdi	gpt-5-nano	10	0.1742	0.2229	-2.5038	0.1496	0.0324
pdi	gpt-5-nano	20	0.1459	0.1894	-2.3025	-0.1755	-0.1854

8.5 Baseline Models

To simulate the LLM few-shot context, all scikit-learn models are trained on the same K=20 subset of the training partition (without replacement, seeded), and evaluated on the full test partition.

8.5.1 Models and Hyperparameter Tuning

We benchmark scikit-learn models and tune hyperparameters via GridSearchCV with 3-fold CV. Regression uses Spearman ρ as the primary score. When data constraints would make the requested CV infeasible (e.g., too few samples per class within the 20-shot subset), the effective number of folds is reduced accordingly. We drop rows with missing targets before fitting and mask the corresponding feature rows.

8.5.2 Baseline Models

- Random Forest Regressor: n_estimators $\in \{200, 500\}$; max_depth $\in \{\text{None}, 8, 16\}$; min_samples_split $\in \{2, 4\}$.
- Ridge Regression: $\alpha \in \{0.1, 1.0, 10.0\}$; pipeline includes StandardScaler.
- Linear SVR: $C \in \{0.5, 1, 2\}$; $\epsilon \in \{0.0, 0.1\}$; pipeline includes StandardScaler.

8.5.3 Evaluation and Aggregation

For classification we report accuracy and macro-F1; for regression we report MAE, RMSE, R^2, Pearson r, and Spearman ρ. We report overall (collapsed across APIs) by averaging metrics across APIs for each target and model.

Table 5: Within-API Performance Metrics (Baseline Models)

Task	Model	k	MAE	RMSE	R^2	Pearson r	Spearman ρ
Total Content	Linear SVR	—	2.900	4.064	-0.813	0.625	0.532
Total Content	Random Forest	—	1.815	2.946	0.346	0.655	0.671
Total Content	Ridge	—	1.573	2.645	0.396	0.765	0.705
PDI	Linear SVR	—	0.117	0.142	-4.973	0.398	0.287
PDI	Random Forest	—	0.115	0.138	-4.834	0.391	0.167
PDI	Ridge	—	0.146	0.194	-16.844	0.290	0.160
Droplet Size	Linear SVR	—	115.417	200.449	-1.058	0.564	0.528
Droplet Size	Random Forest	—	99.599	189.780	-0.525	0.469	0.460
Droplet Size	Ridge	—	112.032	203.603	-1.439	0.527	0.506

Table 6: Between-API Performance Metrics (Baseline Models)

Task	Model	k	MAE	RMSE	R^2	Pearson r	Spearman ρ
Total Content	Linear SVR	—	6.099	6.737	-466.172	0.116	0.100
Total Content	Random Forest	—	4.664	5.164	-285.985	0.276	0.160
Total Content	Ridge	—	5.842	6.451	-449.773	0.133	0.154
PDI	Linear SVR	—	0.118	0.155	-0.603	-0.055	-0.044
PDI	Random Forest	—	0.121	0.152	-0.525	0.167	0.117
PDI	Ridge	—	0.121	0.151	-0.416	0.011	-0.038
Droplet Size	Linear SVR	—	154.525	268.616	-69.487	0.063	0.052
Droplet Size	Random Forest	—	146.988	262.896	-307.077	0.021	-0.009
Droplet Size	Ridge	—	298.953	420.859	-1093.459	-0.083	-0.093