ScreenShot: A Foundation Model for Few-Shot Combination Drug Screening

Antoine de Mathelin, Christopher Tosh and Wesley Tansey
Computational Oncology
Memorial Sloan Kettering Cancer Center
New York, NY 10065
tanseyw@mskcc.org

Abstract

Treating patients with combinations of drugs reduces the risk of resistance to any individual drug. Finding effective combinations is difficult because the large search space makes combinatorial screens prohibitively expensive, time consuming, and often technically infeasible. Predictive models can fill this gap, yet existing methods typically require molecular profiling of each sample and per-cohort training, limiting their applicability when time and tissue are scarce. To address this challenge, we introduce ScreenShot, a hierarchical transformer pretrained on 40 drug screening datasets covering 3,700 drugs and 6,000 biological samples, whose architecture mirrors the nested structure of screening data. Given a few-shot context of observations from a new patient, ScreenShot predicts the response of the sample to combination therapies through in-context learning, operating directly on functional measurements with no fine-tuning and no molecular profiling. On four held-out datasets, ScreenShot outperforms all baselines in both prediction accuracy and identification of selectively effective treatments. ScreenShot's internal representations are directly useful for experimental design: we use them to drive a weighted k-means++ active learning strategy that selects which experiments to run, achieving the same hit detection as uniform screening with a third of the budget. Source code and interactive dashboard: `https://github.com/tansey-lab/screenshot`.

1 Introduction

Drug screens enable functional characterization of tumors in terms of vulnerabilities to candidate therapies. Screens in triple negative breast cancer [1], fibrolamellar carcinoma [2], glioblastoma [3, 4], pancreatic adenocarcinoma [5–7], acute myeloid leukemia [8, 9], and epithelial ovarian cancers [10–12] have revealed novel vulnerabilities and new potential therapies for patients. Multiple recent clinical trials [13, 14] have integrated drug screens into the clinical decision making process and demonstrated that screen-guided treatments can improve patient outcomes.

A key limitation of existing screens and trials is the inability to scale the number of experiments. In many clinical settings, only a small number of cells can be gathered, limiting the number of experiments possible. Further, all trials to-date have focused on single drug screens with combinations designed based on ad hoc assessments of putative drug targets and the perceived molecular susceptibility of a tumor to combination targeting. Similarly, the largest combination screens to-date [15–17] screened only a small subset of combinations, since exhaustively enumerating all combinations would have been cost and time prohibitive. There is a pressing need to develop methods for combination drug screens that maximize sample efficiency and can determine responses to a broad panel of drugs based on relatively few observations.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

To address this need, we developed ScreenShot, a transformer-based foundation model for few-shot prediction in combination drug screens. ScreenShot performs *in-context learning* at inference: given a small set of drug-dose-viability measurements from a new patient sample, it predicts the remaining (untested) responses, without fine-tuning and without requiring molecular profiling of the sample. This contrasts with prior methods that rely on genomic or transcriptomic features to encode biological context [18–20], which are often unavailable or inconsistent for patient-derived samples [21]. By operating directly on the functional measurements produced by the screen, ScreenShot can be applied to new samples with no feature engineering and without the need for time and cost-intensive molecular profiling.

Beyond prediction, we develop a multi-round active learning strategy to identify the most effective treatments with minimal experimental budget. The strategy selects experiments using k-means++ seeding [22] in ScreenShot's embedding space, weighting candidates by their predicted effectiveness to prioritize likely hits while ensuring diversity across rounds. In our benchmarks, our adaptive selection algorithm achieves the same hit recall as uniform selection with only a third of the budget. Our contributions are:

- **Foundation model for functional drug response.** We introduce ScreenShot, a hierarchical transformer pretrained on a pan-cancer collection of 40 drug screen datasets, which predicts dose-response behavior for new patient samples via in-context learning using only drug-dose-viability measurements (no omics features; no per-patient finetuning).
- **Embedding-driven experimental design.** We develop a practical screening policy that uses ScreenShot's pretrained drug-dose embeddings for cold-start batch selection and its context-conditioned embeddings for active learning, explicitly trading off testing promising treatments with broad coverage of the drug-dose design space.
- **Strong empirical performance.** We demonstrate state-of-the-art few-shot prediction and hit detection on held-out monotherapy and combination screens, outperforming XGBoost [23], TabPFN [24], and pretrained neural network baselines across budgets.

2 Method

2.1 Problem formulation

We consider a drug screen setting in which a biological sample s (e.g., a cell line or organoid) is treated with a drug combination $\mathbf{d} = (d_1, \ldots, d_q)$ at respective doses $\boldsymbol{\delta} = (\delta_1, \ldots, \delta_q)$, producing a viability measurement $y \in [0, 1]$. The combination size q varies across experiments (e.g., q=1 for monotherapy, q=2 for pairwise combinations, etc.). We assume a fixed drug library of size M, and encode each drug identity as an integer index $d \in \{1, \ldots, M\}$; doses are real-valued, $\delta \in \mathbb{R}$. Given a small context set of labeled observations for a new sample s, $\mathcal{D}_s = \{(\mathbf{d}_j, \boldsymbol{\delta}_j, y_j)\}_{j=1}^{n}$, our goal is to predict the viability for untested drug-dose queries $(\mathbf{d}', \boldsymbol{\delta}')$ on the same sample, outputting a prediction $\hat{y}$. To best match many precision medicine settings where only small biopsies and thus limited cells are available, we assume that no molecular profiling of s (e.g., gene expression or mutation data) is available: the model must condition only on the observed (drug, dose, viability) tuples in $\mathcal{D}_s$.

Formally, ScreenShot is a parametric predictor f_θ with parameters $\theta \in \mathbb{R}^W$ that maps a query perturbation and its context to a viability estimate,

$$f_\theta : \underbrace{\big(\{1, \ldots, M\}^q \times \mathbb{R}^q\big)}_{\text{query}} \times \underbrace{\big(\{1, \ldots, M\}^q \times \mathbb{R}^q \times [0, 1]\big)^n}_{\text{few-shot context}} \to [0, 1], \qquad \hat{y} = f_\theta\big(\underbrace{(\mathbf{d}', \boldsymbol{\delta}')}_{\text{query}}, \underbrace{\mathcal{D}_s}_{\text{context}}\big).$$

This differs from the classical transfer-learning approach, where one first trains a global predictor $h_\theta : \{1, \ldots, M\}^q \times \mathbb{R}^q \to [0, 1]$ and then fine-tunes it on $\mathcal{D}_s$ to obtain sample-specific parameters θ_s, yielding predictions $\hat{y} = h_{\theta_s}(\mathbf{d}', \boldsymbol{\delta}')$. In contrast, ScreenShot follows the paradigm of recent foundation models for tabular data such as TabPFN [24], and amortizes the adaptation step by conditioning directly on $\mathcal{D}_s$ via cross-attention, effectively replacing explicit fine-tuning with in-context inference. This design is practical and computationally efficient, and avoids the hyperparameter-sensitive fine-tuning trade-offs between fitting $\mathcal{D}_s$ and preserving prior knowledge (e.g., learning rate, number of epochs, and regularization) [25, 26].

2.2 Model architecture

Background and terminology. An *embedding* is a learned vector representation of a discrete item (e.g., a drug identity). A *transformer* [27] is a neural architecture that constructs context-dependent representations of sequence elements through *self-attention*. At each layer, every element updates its representation by computing similarity scores with other elements and aggregating their representations through a weighted sum. Stacking multiple layers enables the model to capture increasingly complex interactions among drug-dose observations. *Cross-attention* extends this mechanism to two distinct sets: each unobserved (query) drug-dose observation attends to the observed context, allowing its representation and resulting prediction to be conditioned on the available few-shot measurements.

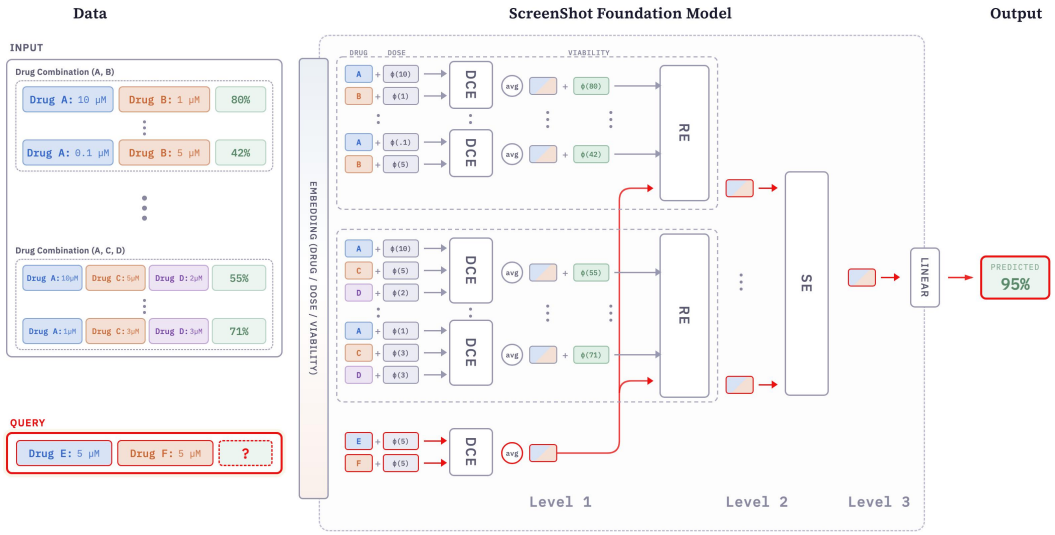

Figure 1: **Overview of the ScreenShot architecture.** ScreenShot processes drug screening data through a three-level hierarchy. Level 1: all drug + dose embeddings are combined via self-attention within each drug combination by the *Drug Combination Encoder* (DCE). Level 2: query combinations attend to context observations (augmented with observed viability) via cross-attention through the *Response Encoder* (RE). Level 3: perturbation-level embeddings are aggregated via self-attention across the sample by the *Sample Encoder* (SE), producing a final representation for viability prediction.

ScreenShot is a hierarchical transformer that leverages the compositional structure of drug screening data (Figure 1). The key observation is that measurements for a fixed drug combination across different dose levels naturally form a group (a dose-response curve) and that each combination is itself composed of individual drugs. Rather than representing the entire context as a single flat sequence of observations, ScreenShot organizes it into three nested levels: *Level 1*: drug-dose combination (perturbation), *Level 2*: response to the perturbation, *Level 3*: aggregation of all perturbations across the sample. An analogy in natural language processing is to first encode words into sentence representations and then attend across sentences in a document, instead of attending over all words in the document at once [28]. For a target query $(\mathbf{d}', \boldsymbol{\delta}')$, ScreenShot performs cross-attention between the query representation and the context at Level 2, enabling it to retrieve and aggregate evidence from dose-response curves of similar perturbations. All three encoders at the three levels use the same transformer configuration: 5 attention blocks, 8 heads, GeLU activations.

The hierarchical architecture is motivated by both modeling and practical considerations. First, the hierarchy mirrors the data generating structure and yields representations at multiple granularities (individual drug, combination, and sample-level) that can be reused in downstream tasks. Second, it reduces the computational cost of attention. A naïve flat-attention model over all n context observations would require $O(n^2)$ attention computations. In contrast, the hierarchical design factorizes this cost into: (i) cross-attention from the query to context observations at Level 2, $\mathcal{O}(n)$, and (ii) self-attention across perturbations, $O(p^2)$, where p is the number of unique perturbations in the context. For a typical screen with n=300 observations comprising p=50 unique perturbations at 6 dose points each, the dominant attention cost decreases from 300^2=90,000 to 50^2=2,500 attention

entries. This shares the same computational motivation as efficient attention methods that mitigate the quadratic cost via local or block-sparse patterns [29, 30]; however, in our case the partition is induced by the assay structure rather than an arbitrary chunking of the input sequence.

Drug, dose, and viability embeddings. Each drug identity d_i in the library is mapped to a learned embedding $\mathbf{e}_{d_i} \in \mathbb{R}^D$ via an embedding table. Crucially, these representations are learned purely from *function*, namely how drugs affect viability across diverse samples, doses, and combination partners, rather than from chemical structure. This makes ScreenShot agnostic to drug modality (e.g., small molecules, natural products, antibodies) and more flexible than approaches based on molecular descriptors, which may be incomplete or unavailable for certain compound classes. A drug absent from the pretraining corpus has no learned identity and is instead mapped to an UNK ("unknown") embedding, a placeholder vector reserved for unseen drugs. These embeddings are trained during pretraining by randomly replacing a fraction of the drug identities in each batch with UNK tokens (Section B), which teaches the model to predict the response to an unidentified drug from the surrounding context rather than from a drug-specific vector. We evaluate this mechanism on unseen drugs in Section C.

Doses and viabilities are encoded via learned Fourier features $\phi(\cdot)$ that map continuous scalars to $\mathbb{R}^D$ (details in Section A). Since dose is continuous, we want its embedding to vary smoothly, so that nearby doses receive nearby representations and the model interpolates along the dose axis rather than treating dose levels as unordered categories. The encoding follows the sinusoidal positional encodings of Vaswani et al. [27], except that the frequencies and their coefficients are trainable, so the network can allocate resolution where the dose-response signal is richest. We use $D{=}256$ throughout (a $D{=}16$ variant is markedly less accurate and barely improves with more context; Section K).

Drug combination encoder (Level 1). The drug embedding and dose encoding are summed, $\mathbf{h}_k = \mathbf{e}_{d_k} + \phi(\delta_k)$, and the resulting sequence $(\mathbf{h}_1, \ldots, \mathbf{h}_q)$ is processed by a transformer encoder. Self-attention allows each drug token to attend to the other drugs in the combination, capturing interaction effects. The outputs are aggregated via masked average pooling into a single combination embedding $\mathbf{c} \in \mathbb{R}^D$ (the drug token vectors are averaged, while a mask excludes padding entries, so that combinations of different sizes are summarized uniformly). Because the number of drugs q is handled through attention masking, the same architecture natively supports monotherapies, pairs, and higher-order combinations without any structural modification. This encoder is applied to each context perturbation $(\mathbf{d}_j, \boldsymbol{\delta}_j)$, yielding a sequence of embeddings $(\mathbf{c}_1, \ldots, \mathbf{c}_n)$, and is likewise applied to the query $(\mathbf{d}', \boldsymbol{\delta}')$ to produce a query embedding $\mathbf{c}'$.

Response encoder (Level 2). At this level, ScreenShot conditions the query embedding $\mathbf{c}'$ on the sample-specific context using multi-head cross-attention. Each context observation is represented by augmenting its combination embedding with the measured viability, $\tilde{\mathbf{c}}_j = \mathbf{c}_j + \phi_y(y_j)$. The context is then grouped by unique perturbation (i.e., all dose points measured for the same drug combination). Cross-attention is applied separately within each group: the query $\mathbf{c}'$ attends to the observations in perturbation group g, producing a group-conditioned query representation $\mathbf{r}'_g$. Repeating this over all groups produces $(\mathbf{r}'_1, \ldots, \mathbf{r}'_p)$, one representation per observed perturbation. When predicting multiple queries for the same sample, this procedure is applied independently to each query, so that predictions depend only on the shared context and the corresponding query.

Sample encoder (Level 3). ScreenShot then aggregates information across the perturbations observed for the same sample. The sequence $(\mathbf{r}'_1, \ldots, \mathbf{r}'_p)$ is processed by a transformer encoder with multi-head self-attention, allowing evidence to propagate across perturbations (e.g., response patterns for one drug informing predictions for related drugs). The resulting query representation is mapped to a viability estimate via a linear prediction head followed by a sigmoid, producing $\hat{y} \in [0, 1]$.

2.3 Pretraining

ScreenShot is pretrained on a pan-cancer drug screening atlas [21] spanning both monotherapy and combination assays across diverse screening technologies and biological systems. Overall, the corpus contains approximately 30M viability measurements across about 4k drugs and 6k biological samples, including cancer cell lines as well as patient-derived models such as ex vivo cultures and organoids.

Training follows an in-context learning objective that matches the downstream setting. Let $\mathcal{S}_{\text{train}}$ denote the set of biological samples available during pretraining, where each sample $s \in \mathcal{S}_{\text{train}}$ comes

with a set of labeled observations $\mathcal{D}_s$. At each training step, we sample a biological sample $s \sim \mathcal{S}_{\text{train}}$, then draw a context set $\mathcal{D} \subset \mathcal{D}_s$ and a query set $\mathcal{Q} \subset \mathcal{D}_s$ uniformly at random, with elements of the form $(\mathbf{d}, \boldsymbol{\delta}, y)$. We train ScreenShot to predict the query viabilities conditioned on the context, $\hat{y} = f_\theta\big((\mathbf{d}, \boldsymbol{\delta}),\, \mathcal{D}\big)$ for each $(\mathbf{d}, \boldsymbol{\delta}, y) \in \mathcal{Q}$.

To handle technical noise (e.g. failed experiments) in the data, we optimize θ by minimizing the mean absolute error over the query set,

$$\mathcal{L}(\theta) = \mathbb{E}_{s,\mathcal{D},\mathcal{Q}}\left[\frac{1}{|\mathcal{Q}|}\sum_{(\mathbf{d},\boldsymbol{\delta},y)\in\mathcal{Q}}\left|f_\theta\big((\mathbf{d},\boldsymbol{\delta}),\,\mathcal{D}\big) - y\right|\right],$$

which is more robust to outliers than the mean squared error. No molecular profiling features are used at any stage; the model is trained only from drug identity, dose, and observed viability in the context. Implementation details are provided in Sections A and B.

2.4 Few-shot inference

At inference time, ScreenShot is applied to a previously unseen sample $s \sim \mathcal{S}_{\text{test}}$, drawn from a held-out cohort. Given a small context set of labeled measurements $\mathcal{D}_s$, the model is queried on untested drug-dose conditions. ScreenShot conditions on $\mathcal{D}_s$ through cross-attention and produces predictions $\hat{y}$ for each query. No gradient updates or fine-tuning are performed: the model generalizes to new samples entirely through its pretrained attention mechanism, which learns to condition on the provided context. This is analogous to in-context learning in large language models, where a pretrained transformer adapts its predictions based on the input prompt without updating its weights [31, 24].

2.5 Active learning pipeline

ScreenShot enables few-shot screening: given a small set of (drug, dose, viability) observations from a new sample, it predicts the remaining responses through in-context inference. The practical bottleneck is therefore deciding *which* experiments to measure. Because ScreenShot's prediction accuracy depends on the context it receives, we seek a budgeted strategy for selecting context points that helps the model identify the most effective and selective treatments as quickly as possible. We use a multi-round design in which we select a batch of treatments to measure, append the new observations to the context, and re-run ScreenShot to update predictions and representations.

Cold-start batch. The first batch must be chosen before any viability is observed for the target sample, which is the cold-start setting [32]. At this stage, we rely on representations that do not require labels. We compute Level 1 drug-dose embeddings for all candidate treatments and select an initial batch using k-medoids clustering [33]. This targets coverage in the pretrained treatment space, providing a diverse initial context that is informative on average across samples.

Adaptive batches. After observing the initial measurements, ScreenShot produces context-conditioned representations for each candidate treatment. While these representations could be used with generic rules such as coverage (k-medoids) or uncertainty-based acquisition, we found it more effective to drive selection by the downstream objective: identifying selective hits. For each unobserved treatment t, we compute predicted differential viability $\Delta_t = \hat{y}_t^{(\text{sample})} - \hat{y}_t^{(\text{control})}$, where the control term is defined as in Section 4. Strongly negative Δ_t indicates predicted selectivity. To select the next batch, we run weighted k-means++ seeding [22] on the context-conditioned treatment representations: centers are chosen one at a time, each sampled with probability proportional to its weight times its squared distance to the nearest already-selected center, which spreads the selection apart. Candidates are weighted by $w_t = \max(-\Delta_t, 0)$ to prioritize predicted hits. Treatments already selected in previous rounds are included as fixed initial centers, so that new selections are pushed away from already-observed regions of the treatment space. We compare this objective-driven policy to coverage-only and uncertainty-based alternatives in Section L; we find it recovers top hits more efficiently under the same budget.

Algorithm 1 Multi-round active learning with ScreenShot

Require: Unlabeled pool $\mathcal{U}$, cold-start budget n_0, budget per round b, rounds T

Compute drug-dose embeddings $\mathbf{c}_t$ for all $t \in \mathcal{U}$ ▷ Level 1 (DCE)
$\mathcal{D}_s \leftarrow k$-medoids$(\{\mathbf{c}_t\}, n_0)$; and obtain labels for $\mathcal{D}_s$ ▷ Cold-start
for $r = 1, \ldots, T$ **do**
  Predict $\hat{y}_t$ for all $t \in \mathcal{U}$ using ScreenShot conditioned on $\mathcal{D}_s$
  $\Delta_t = \hat{y}_{\text{target},t} - \min_{c \in \mathcal{C}} \hat{y}_{c,t}$ for each $t \notin \mathcal{D}_s$
  $w_t = \max(-\Delta_t, 0)$ ▷ Prioritize predicted hits
  Extract embeddings $\mathbf{h}_t$ for all $t \in \mathcal{U}$ ▷ Level 3 (SE)
  $\mathcal{B} \leftarrow k$-means++$(\{\mathbf{h}_t\}_{t \notin \mathcal{D}_s},\ b,\ \{w_t\},\ \text{init}{=}\{\mathbf{h}_t\}_{t \in \mathcal{D}_s})$; and obtain labels for $\mathcal{B}$
  $\mathcal{D}_s \leftarrow \mathcal{D}_s \cup \mathcal{B}$
end for
return Final predictions from ScreenShot conditioned on $\mathcal{D}_s$

3 Related work

Drug response prediction. Machine learning for drug response prediction has been extensively studied, with deep learning methods proposed for monotherapy sensitivity [34, 35, 18–20, 36] and combination synergy [37–42]. A pervasive limitation is the reliance on genomic or transcriptomic features to represent biological samples, which restricts applicability to patient-derived samples where such profiling is often unavailable or inconsistent [21]. Transfer learning approaches [43–45] partially address the data scarcity problem by pretraining on cell line databases, but still require overlapping molecular features between source and target domains. ScreenShot removes this requirement entirely, operating only on drug-dose-viability measurements.

Foundation models and in-context learning. Foundation models pretrained on large corpora and adapted through in-context learning, without fine-tuning on the target task, have achieved strong few-shot performance in natural language processing [31] and, more recently, on tabular data [24, 46, 47]. This paradigm replaces the fragile fine-tuning step, which is sensitive to learning rate, regularization, and dataset size [25, 26], with a single forward pass conditioned on a context set. ScreenShot adopts this paradigm for drug screening: it is pretrained on a pan-cancer atlas and performs in-context inference on new patients. Several concurrent works explore foundation models or in-context learning for drug response prediction [48–50, 21]. However, these approaches present significant limitations. Some rely on knowledge encoded in large language models, requiring prior literature on the sample, which is unavailable for patient-derived cells [49, 48]. Others predict binary synergy labels rather than continuous dose-response, a strictly less flexible setting since dose-response models can derive synergy but not the reverse [49, 50]. All process context observations through flat attention, scaling quadratically in context size. Most do not release pretrained weights [49, 50, 21], and Pichotta et al. [21] is limited to monotherapy. ScreenShot addresses all of these limitations.

Active learning for drug screening. Active learning has shown promise for drug screening [51–56]. Existing methods for batch combination screening typically balance exploration and exploitation through uncertainty-based acquisition [57, 58] and retrain the predictive model after each batch. Cold-start selection of the initial batch using coverage-based methods has also been explored [32]. Building upon these works, we propose an adaptive selection strategy that performs the exploration-exploitation tradeoff through efficacy-weighted k-means++ seeding in ScreenShot's learned embedding space, avoiding the computational overhead of uncertainty estimation due to posterior sampling or ensemble construction [59, 60].

4 Experiments

4.1 Experimental setup

Datasets. We pretrain ScreenShot on a curated collection of 40 drug screening datasets comprising approximately 30M viability measurements across ∼3,700 drugs and ∼6,000 biological samples (full list in Section B). Pretraining uses the Adam optimizer with learning rate 10^{-5} for ∼400K iterations on a single NVIDIA H100 GPU; further details are in Sections A and B. For evaluation, we hold

out four datasets not seen during pretraining. Three are pairwise combination screens, which include observations for both drug combinations and monotherapies: **NCI-ALMANAC** [16] screens 104 drugs in ∼5,000 pairwise combinations across the NCI-60 cancer cell line panel; **GDSC-SQ** [17] screens 65 drugs in ∼1,300 combinations across 126 breast, colorectal, and pancreatic cancer cell lines; and **BATCHIE** [54] screens 207 drugs in ∼19,000 combinations across 16 cell lines (14 cancer plus 2 non-cancerous controls). For faster evaluation, we use 10 cell lines from NCI-ALMANAC and GDSC-SQ. The fourth, **PDO-Breast** [1], is a purely monotherapy ex vivo screen of 16 breast cancer patient-derived organoid models treated with 46 drugs (∼81K observations). Including PDO-Breast tests generalization to the monotherapy-only setting. For each sample in the evaluation datasets, we randomly select $n \in \{50, 100, 150, 200, 250, 300\}$ observations as few-shot context and predict viability for all remaining held-out observations.

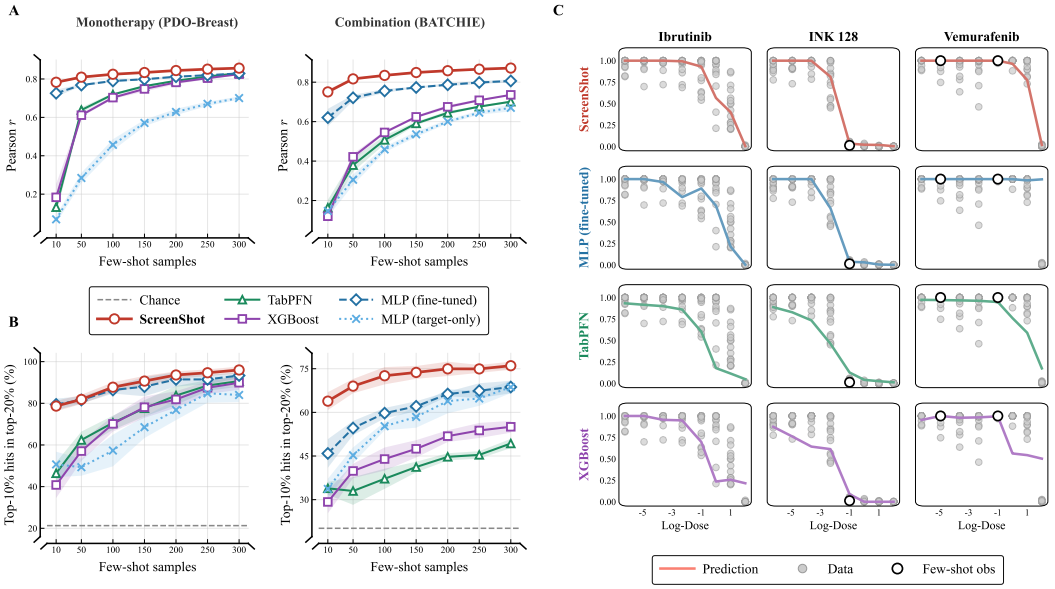

Figure 2: **Few-shot dose-response prediction on held-out datasets.** **(A)** Pearson correlation as a function of the number of few-shot observations on the PDO-Breast screen (left) and BATCHIE screen (right). **(B)** Top-10% hit recall in the top-20% predictions (PDO-Breast, left; BATCHIE, right). **(C)** Example dose-response curves for three drugs on sample HCI-003 (PDO-Breast) at n=100.

Metrics. We evaluate predictions along two complementary axes. For *prediction quality*, we report Pearson correlation r between predicted and true viability values. For *hit detection*, the primary practical goal in drug screening, we measure the ability to identify *selectively effective* treatments: drug combinations that reduce viability in the target cancer sample more than in non-cancerous controls. Following Tosh et al. [54], we quantify selectivity via the therapeutic index of a drug combination $\mathbf{d}$, defined as the minimum differential viability over its dose grid:

$$\mathrm{TI}(\mathbf{d}, s) \;=\; \min_{\boldsymbol{\delta}} \Big[y_{s,(\mathbf{d},\boldsymbol{\delta})} \;-\; \min_{c \in \mathcal{C}} \; y_{c,(\mathbf{d},\boldsymbol{\delta})} \Big], \tag{1}$$

where s is the target sample, $\mathcal{C}$ is the set of control samples, and the outer minimum selects the dose at which the target is most sensitive relative to controls. Strongly negative TI indicates a selectively effective treatment. For PDO-Breast, a monotherapy screen, we instead rank drugs by the mean differential viability across doses (i.e., the difference in area under the dose-response curve between target and control), which is a standard measure of single-drug efficacy.

For the BATCHIE dataset, $\mathcal{C}$ consists of two non-cancerous cell lines (RPE and BJ) included in the screen by design. For NCI-ALMANAC, GDSC-SQ, and PDO-Breast, which lack designated controls, we select the cell line with the highest average viability across all treatments as a proxy control, under the assumption that non-cancerous samples are on average more resistant to drug treatment, a pattern confirmed empirically in BATCHIE, where the two non-cancerous controls rank first and third in average viability.

We define ground-truth hits as the 10% of treatments with the most negative true TI per sample, and report *hit recall*: the fraction of true top-10% hits recovered among the top-20% of treatments ranked by predicted TI, averaged across non-control samples. When computing predicted TI for evaluation, we use ground-truth viability values for the control samples. This benefits all methods equally and reduces evaluation noise; it is also realistic, as controls are shared across targets and would be measured exhaustively in practice. When each method must instead predict the control response from the same few-shot budget, absolute hit recall decreases for every method but the ranking is preserved: ScreenShot is best in 18 of 20 dataset/budget cells (Section H). For the active learning acquisition strategy, we use model predictions for controls: using ground truth would selectively benefit adaptive strategies over random and cold-start, since only adaptive methods use the predicted delta for selection (details in Section D).

Baselines. We compare ScreenShot against four baselines. **XGBoost** and **TabPFN** [24] both use 512-dimensional Morgan fingerprints [61] concatenated with dose as input features; XGBoost hyperparameters are selected via 5-fold cross-validation, while TabPFN performs in-context learning. **MLP (target-only)** is a multi-layer perceptron with learned drug and dose embeddings, trained from scratch on each target sample. **MLP (fine-tuned)** uses the same architecture, pretrained on the same 40-dataset corpus as ScreenShot and fine-tuned on the target sample (see Section E.2 for more details).

These baselines span the main axes of methodological variation: no pretraining (XGBoost, MLP target-only), general-purpose pretraining (TabPFN), and domain-specific pretraining with fine-tuning (MLP fine-tuned, the closest comparator to ScreenShot). All baselines receive the same observations as ScreenShot. Among existing drug response models (Section E), most require molecular profiling of the sample, do not model dose, or are restricted to monotherapy. The two that handle drug combinations and doses without molecular profiling, ComboLTR [62] and PIICM [63], are trained independently on each sample and were not designed for the few-shot regime. We evaluate them in Section E.3, where both perform substantially below all other methods. Additionally, we compare our active learning strategy to BATCHIE-AL [54] in Section J.1. Code, pretrained models, and an interactive dashboard are available at `https://github.com/tansey-lab/screenshot`.

4.2 Few-shot prediction

We first evaluate all methods with randomly selected few-shot observations. Figure 2A and B show that ScreenShot outperforms all baselines on both the PDO-Breast monotherapy ex vivo screen and the BATCHIE combination screen, in both Pearson correlation and hit recall, across all budgets. The gap between ScreenShot and the second-best method (MLP fine-tuned) is larger on the combination screen, which is a more challenging setting due to the combinatorial treatment space.

Figure 2C shows dose-response predictions for one PDO-Breast sample (HCI-003) at n=100. The three drugs illustrate different few-shot regimes: Vemurafenib has two context observations, INK 128 has one, and Ibrutinib has none. ScreenShot fits the ground truth more closely in all three cases.

Table 1 extends the comparison to all four held-out datasets. ScreenShot achieves the highest Pearson correlation on every dataset and budget, and the highest hit recall on 16 out of 20 dataset/budget combinations. The advantage is largest at low budgets, where pretraining matters most: at n=10 on BATCHIE, ScreenShot reaches 63.9% hit recall compared to 45.9% for the next-best method, and on NCI-ALMANAC the gap over MLP fine-tuned exceeds 13 percentage points at n=300. ScreenShot is also 5–50$\times$ faster than all baselines.

The advantage of ScreenShot holds under stricter generalization settings on BATCHIE (Sections C and G): it is the most accurate method at every budget for queries with a drug absent from the context (leave-drug-out), with a drug pair never observed together in the context (leave-combination-out), or with one drug unseen during pretraining; only for combinations made entirely of unseen drugs does the target-only MLP overtake it at $n \geq 50$. Cluster-robust confidence intervals and additional regression and ranking metrics are reported in Section I.

Table 1: Few-shot prediction across four held-out drug screening datasets. Values: mean±std across 3 to 5 repetitions. **Bold**: best per column within each dataset. Time: average wall-clock seconds per experiment. Bottom: average score normalized by ScreenShot (1.00 = ScreenShot level).

		Pearson r					Top-hit recall (%)					Time
		10	50	100	200	300	10	50	100	200	300	(s)
PDO-Br.	XGBoost	.18±.07	.61±.01	.70±.01	.77±.01	.82±.00	40.8±6.3	57.1±4.8	70.1±3.7	81.9±4.0	89.9±3.3	192
	TabPFN	.13±.04	.64±.01	.72±.01	.79±.01	.82±.00	46.4±5.3	62.4±4.0	70.7±3.4	83.7±4.6	90.7±2.7	127
	MLP	.07±.02	.28±.03	.45±.03	.62±.02	.69±.01	50.7±6.1	49.3±4.1	57.3±7.6	76.8±4.9	84.0±3.4	1156
	MLP-FT	.73±.02	.77±.01	.79±.01	.81±.01	.82±.00	**79.5±2.8**	81.6±1.7	86.4±1.7	91.5±3.5	93.3±1.3	1180
	ScreenShot	**.78±.01**	**.81±.01**	**.82±.00**	**.84±.00**	**.85±.00**	78.7±2.5	**81.9±2.2**	**87.7±2.4**	**93.6±1.7**	**96.0±1.6**	**8**
NCI	XGBoost	.11±.03	.25±.02	.35±.02	.48±.00	.56±.01	24.0±1.5	28.5±1.7	33.0±4.2	42.6±1.5	50.0±0.5	118
	TabPFN	.16±.03	.26±.01	.32±.01	.43±.01	.50±.01	24.6±1.8	30.1±0.9	33.0±4.3	41.1±2.8	47.8±1.9	524
	MLP	.12±.01	.18±.01	.23±.01	.25±.05	.35±.02	24.8±2.9	30.0±1.7	32.2±1.7	36.4±1.8	40.6±1.9	710
	MLP-FT	.51±.01	.57±.01	.61±.01	.65±.01	.67±.01	34.6±0.8	38.2±0.8	40.6±1.1	48.5±1.5	49.5±2.8	726
	ScreenShot	**.58±.02**	**.65±.01**	**.68±.00**	**.72±.01**	**.74±.00**	**44.5±3.9**	**51.0±2.3**	**54.8±1.1**	**61.1±2.2**	**63.0±0.6**	**68**
GDSC-SQ	XGBoost	.19±.04	.47±.03	.59±.01	.70±.01	.75±.01	26.5±2.7	41.9±1.8	51.5±4.1	61.0±0.5	66.0±2.4	137
	TabPFN	.28±.04	.44±.01	.54±.01	.68±.01	.75±.01	30.3±2.9	43.0±4.4	49.3±2.2	60.1±0.4	**66.3±1.2**	655
	MLP	.25±.05	.39±.01	.48±.04	.58±.02	.66±.01	27.5±2.0	41.6±3.9	45.7±3.1	55.3±2.1	60.8±2.2	735
	MLP-FT	.51±.03	.63±.02	.68±.02	.72±.01	.76±.00	38.9±2.5	49.9±2.7	**54.8±2.2**	**62.5±0.5**	66.0±1.7	737
	ScreenShot	**.60±.02**	**.71±.00**	**.73±.01**	**.76±.00**	**.78±.01**	**39.8±2.1**	**51.7±1.0**	54.4±2.9	58.2±0.7	60.6±0.6	**64**
BATCHIE	XGBoost	.12±.04	.42±.02	.54±.01	.67±.01	.73±.00	29.2±3.9	39.9±2.5	44.0±4.0	51.8±2.0	55.0±2.0	196
	TabPFN	.16±.04	.38±.03	.50±.02	.64±.01	.69±.01	33.9±2.2	33.0±4.8	37.2±3.5	44.7±1.2	49.3±1.6	432
	MLP	.15±.03	.30±.03	.45±.01	.59±.01	.66±.02	33.7±3.5	45.2±2.7	55.3±1.4	63.9±4.2	68.7±2.1	1152
	MLP-FT	.62±.04	.72±.01	.75±.01	.78±.01	.80±.01	45.9±4.9	54.6±2.4	59.7±1.8	66.3±0.8	68.7±2.1	1142
	ScreenShot	**.75±.01**	**.82±.00**	**.83±.01**	**.86±.00**	**.87±.00**	**63.9±3.1**	**69.0±2.4**	**72.6±3.0**	**75.0±2.5**	**76.0±1.4**	**22**
Avg	XGBoost	0.23	0.58	0.71	0.82	0.88	0.54	0.66	0.74	0.83	0.89	9.48
	TabPFN	0.28	0.57	0.67	0.79	0.85	0.61	0.67	0.71	0.80	0.86	13.59
	MLP	0.23	0.39	0.52	0.64	0.72	0.60	0.66	0.71	0.80	0.86	56.23
	MLP-FT	0.87	0.90	0.92	0.94	0.95	0.87	0.88	0.89	0.93	0.94	56.98
	ScreenShot	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**	**1.00**

4.3 Active learning for hit detection

We now compare different strategies for selecting the few-shot observations, all using ScreenShot as the underlying predictor. We consider four strategies: *random* selection (the baseline used in Section 4.2); *cold-start* selection via k-medoids on drug-dose embeddings; and two *adaptive* strategies that allocate 33% of the budget to a cold-start batch and the remainder to delta-guided adaptive rounds (Algorithm 1), with either one round (batch size = 67% of budget) or two rounds (two batches of 33%).

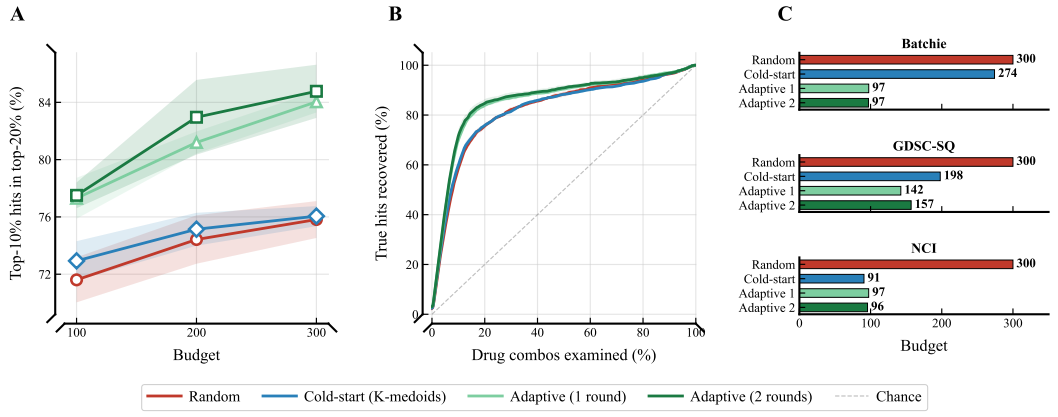

Figure 3: **Active learning for hit identification. (A)** Hit recall (top-10% hits in top-20% predictions) as a function of total budget on BATCHIE. **(B)** Cumulative hit recall curves at budget 300 on BATCHIE, showing that the ranking is robust to the choice of prediction set size. **(C)** Budget required to match random selection at budget 300, across three datasets.

Figure 3A shows that the adaptive approach yields a significant improvement on BATCHIE: at budget 300, adaptive selection reaches $\sim$85% hit recall compared to $\sim$76% for both random and cold-start selection, a 9 percentage-point gain. Cold-start selection alone does not significantly improve over random on this dataset. Figure 3B shows cumulative recall curves at budget 300: the adaptive variants recover a larger fraction of true hits at every prediction set size, confirming that the improvement is robust to the choice of top-k threshold.

Figure 3C extends the analysis to three datasets via a budget equivalence comparison: for each method, we report the budget needed to match the hit recall of random selection at budget 300 (obtained by linear interpolation between evaluated budgets). Adaptive strategies require 2–3$\times$ less budget than random to reach the same recall. Cold-start selection helps on some datasets (notably NCI). The number of adaptive rounds has a modest effect; the dominant factors are the total budget and whether adaptive selection is used at all. This is encouraging for clinical applications, where sample availability naturally limits the number of experimental rounds.

5 Discussion and limitations

Drug coverage. ScreenShot is designed to assist in large-scale pharmacological screens for compounds that have experimental data in the literature. The pretrained library covers $\sim$3,700 unique drugs resolved through PubChem, providing at least 89% coverage on all four evaluation datasets (Section C). For novel compounds, the model uses pretrained UNK tokens that allow it to leverage few-shot context observations to infer the drug's behavior; we evaluate the effect of UNK tokens compared to alternatives in Section C and show that they are beneficial. ScreenShot remains the most accurate method on combinations containing one unseen drug, but is overtaken by the target-only MLP on combinations made entirely of unseen drugs, a regime that can be addressed by extending the pretraining corpus with studies that include these drugs. Compared to methods relying on Morgan fingerprints as drug representation in our benchmark, ScreenShot consistently performs better. As the corpus of public drug screening data grows, the pretrained library will naturally expand. Extending ScreenShot to genetic perturbation screens [64], where dose has no natural analog, is a promising direction for future work.

Architecture. An ablation study (Section K) confirms that each level of the hierarchy improves prediction quality and that the hierarchical design is faster than flat attention. Removing the Level 3 self-attention reduces inference time with only a modest drop in hit recall, suggesting that future work could explore lighter alternatives to the sample-level self-attention while preserving accuracy.

Accuracy vs. hit detection. The active learning ablation (Section L) reveals that global prediction accuracy and hit detection ability are not directly linked. Using MC dropout uncertainty as selection weights instead of predicted delta yields the best MAE and Pearson correlation but 6 percentage points lower hit recall on BATCHIE (78.6% vs. 84.8% at budget 300). This suggests that uncertainty-guided selection targets regions of model ignorance rather than regions of differential drug sensitivity. When the goal is to identify top hits, weighting by the objective of interest (therapeutic index) is more effective than reducing global uncertainty.

Acknowledgments and Disclosure of Funding

WT is supported by the NIH/NCI (R37 CA271186, U54 CA274492, P30 CA008748), Break Through Cancer, the Fund for Innovation in Cancer Informatics, the Cancer AI Alliance, the Tow Center for Developmental Oncology, and the Maurice Campbell Initiative at Memorial Sloan Kettering Cancer Center.

References

[1] Katrin P Guillen, Maihi Fujita, Andrew J Butterfield, Sandra D Scherer, Matthew H Bailey, Zhengtao Chu, Yoko S DeRose, Ling Zhao, Emilio Cortes-Sanchez, Chieh-Hsiang Yang, et al. A human breast cancer-derived xenograft and organoid platform for drug discovery and precision oncology. *Nature cancer*, 3(2):232–250, 2022.

[2] Gadi Lalazar, David Requena, Lavoisier Ramos-Espiritu, Denise Ng, Patrick D Bhola, Ype P De Jong, Ruisi Wang, Nicole JC Narayan, Bassem Shebl, Solomon Levin, et al. Identification of novel therapeutic targets for fibrolamellar carcinoma using patient-derived xenografts and direct-from-patient screening. *Cancer Discovery*, 11(10):2544–2563, 2021.

[3] Jin-Ku Lee, Zhaoqi Liu, Jason K. Sa, Sang Shin, Jiguang Wang, Mykola Bordyuh, Hee Jin Cho, Oliver Elliott, Timothy Chu, Seung Won Choi, et al. Pharmacogenomic landscape of patient-derived tumor cells informs precision oncology therapy. *Nature Genetics*, 50(10):1399–1411, 2018.

[4] Patrik Johansson, Cecilia Krona, Soumi Kundu, Milena Doroszko, Sathishkumar Baskaran, Linnéa Schmidt, Claire Vinel, Elin Almstedt, Ramy Elgendy, Ludmila Elfineh, et al. A patient-derived cell atlas informs precision targeting of glioblastoma. *Cell Reports*, 32(2), 2020.

[5] Hervé Tiriac, Pascal Belleau, Dannielle D. Engle, Dennis Plenker, Astrid Deschênes, Tim D. D. Somerville, Fieke E. M. Froeling, Richard A. Burkhart, Robert E. Denroche, Gun-Ho Jang, et al. Organoid profiling identifies common responders to chemotherapy in pancreatic cancer. *Cancer Discovery*, 8(9):1112–1129, 2018.

[6] Else Driehuis, Arne van Hoeck, Kat Moore, Sigrid Kolders, Hayley E. Francies, M. Can Gulersonmez, Edwin C. A. Stigter, Boudewijn Burgering, Veerle Geurts, Ana Gracanin, et al. Pancreatic cancer organoids recapitulate disease and allow personalized drug screening. *Proceedings of the National Academy of Sciences*, 116(52):26580–26590, 2019.

[7] Christian K. Hirt, Tijmen H. Booij, Linda Grob, Patrik Simmler, Nora C. Toussaint, David Keller, Doreen Taube, Vanessa Ludwig, Alexander Goryachkin, Chantal Pauli, et al. Drug screening and genome editing in human pancreatic cancer organoids identifies drug-gene interactions and candidates for off-label therapy. *Cell Genomics*, 2(2), 2022.

[8] Shawn HR Lee, Wenjian Yang, Yoshihiro Gocho, August John, Lauren Rowland, Brandon Smart, Hannah Williams, Dylan Maxwell, Jeremy Hunt, Wentao Yang, et al. Pharmacotypes across the genomic landscape of pediatric acute lymphoblastic leukemia and impact on treatment response. *Nature Medicine*, 29(1):170–179, 2023.

[9] Amy Burd, Ross L Levine, Amy S Ruppert, Alice S Mims, Uma Borate, Eytan M Stein, Prapti Patel, Maria R Baer, Wendy Stock, Michael Deininger, et al. Precision medicine treatment in acute myeloid leukemia using prospective genomic profiling: feasibility and preliminary efficacy of the beat aml master trial. *Nature Medicine*, 26(12):1852–1858, 2020.

[10] Jason K. Sa, Jae Ryoung Hwang, Young-Jae Cho, Ji-Yoon Ryu, Jung-Joo Choi, Soo Young Jeong, Jihye Kim, Myeong Seon Kim, E. Sun Paik, Yoo-Young Lee, et al. Pharmacogenomic analysis of patient-derived tumor cells in gynecologic cancers. *Genome Biology*, 20(1), 2019. ISSN 1474-760X. doi: 10.1186/s13059-019-1848-3.

[11] Filipe Correia Martins, Dominique-Laurent Couturier, Ines de Santiago, Carolin Margarethe Sauer, Maria Vias, Mihaela Angelova, Deborah Sanders, Anna Piskorz, James Hall, Karen Hosking, et al. Clonal somatic copy number altered driver events inform drug sensitivity in high-grade serous ovarian cancer. *Nature Communications*, 13(1), 2022.

[12] Astrid Murumägi, Daniela Ungureanu, Suleiman Khan, Mariliina Arjama, Katja Välimäki, Aleksandr Ianevski, Philipp Ianevski, Rebecka Bergström, Alice Dini, Anna Kanerva, et al. Drug response profiles in patient-derived cancer cells across histological subtypes of ovarian cancer: real-time therapy tailoring for a patient with low-grade serous carcinoma. *British Journal of Cancer*, 128(4):678–690, 2022.

[13] Heike Peterziel, Nora Jamaladdin, Dina ElHarouni, Xenia F. Gerloff, Sonja Herter, Petra Fiesel, Yannick Berker, Mirjam Blattner-Johnson, Kathrin Schramm, Barbara C. Jones, et al. Drug sensitivity profiling of 3D tumor tissue cultures in the pediatric precision oncology program INFORM. *NPJ Precision Oncology*, 6(1), 2022.

[14] Loretta M S Lau, Chelsea Mayoh, Jinhan Xie, Paulette Barahona, Karen L MacKenzie, Marie Wong, Alvin Kamili, Maria Tsoli, Tim W Failes, Amit Kumar, et al. In vitro and in vivo drug screens of tumor cells identify novel therapies for high-risk child cancer. *EMBO Molecular Medicine*, 14(4), 2021.

[15] Jennifer O'Neil, Yair Benita, Igor Feldman, Melissa Chenard, Brian Roberts, Yaping Liu, Jing Li, Astrid Kral, Serguei Lejnine, Andrey Loboda, et al. An unbiased oncology compound screen to identify novel combination strategies. *Molecular Cancer Therapeutics*, 15(6):1155–1162, 2016. ISSN 1538-8514. doi: 10.1158/1535-7163.mct-15-0843.

[16] Susan L. Holbeck, Richard Camalier, James A. Crowell, Jeevan Prasaad Govindharajulu, Melinda Hollingshead, Lawrence W. Anderson, Eric Polley, Larry Rubinstein, Apurva Srivastava, Deborah Wilsker, Jerry M. Collins, and James H. Doroshow. The National Cancer Institute ALMANAC: A comprehensive screening resource for the detection of anticancer drug pairs with enhanced therapeutic activity. *Cancer Research*, 77(13):3564–3576, 2017. ISSN 0008-5472.

[17] Patricia Jaaks, Elizabeth A Coker, Daniel J Vis, Olivia Edwards, Emma F Carpenter, Simonetta M Leto, Lisa Dwane, Francesco Sassi, Howard Lightfoot, Syd Barthorpe, et al. Effective drug combinations in breast, colon and pancreatic cancer cells. *Nature*, 603(7899):166–173, 2022.

[18] Brent M Kuenzi, Jisoo Park, Samson H Fong, Kyle S Sanchez, John Lee, Jason F Kreisberg, Jianzhu Ma, and Trey Ideker. Predicting drug response and synergy using a deep learning model of human cancer cells. *Cancer cell*, 38(5):672–684, 2020.

[19] Qiao Liu, Zhiqiang Hu, Rui Jiang, and Mu Zhou. Deepcdr: a hybrid graph convolutional network for predicting cancer drug response. *Bioinformatics*, 36(Supplement_2):i911–i918, 2020.

[20] Ingoo Jin and Hojung Nam. Hidra: Hierarchical network for drug response prediction with attention. *Journal of Chemical Information and Modeling*, 61(8):3858–3867, 2021.

[21] Karl Pichotta, Jessica B White, Jeffrey F Quinn, Anneliese Markus, Christopher Tosh, Antoine De Mathelin, Erin Coyne, Feiyang Huang, and Wesley Tansey. A pan-cancer ex vivo drug screen atlas for functional precision oncology. *bioRxiv*, pages 2026–02, 2026.

[22] David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 1027–1035, 2007.

[23] Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016.

[24] Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326, 2025.

[25] Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, 2018.

[26] Marius Mosbach, Maksym Andriushchenko, and Dietrich Klakow. On the stability of fine-tuning bert: Misconceptions, explanations, and strong baselines. In *9th International Conference on Learning Representations*, 2021.

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[28] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 1480–1489, 2016.

[29] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

[30] Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33: 17283–17297, 2020.

[31] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901, 2020.

[32] Antoine de Mathelin, Christopher Tosh, and Wesley Tansey. Addressing the cold-start problem for personalized combination drug screening. In *Machine Learning in Computational Biology*, pages 230–239. PMLR, 2025.

[33] Leonard Kaufman and Peter J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons, 1990.

[34] Theodore Sakellaropoulos, Konstantinos Vougas, Sonali Narang, Filippos Koinis, Athanassios Kotsinas, Alexander Polyzos, Tyler J Moss, Sarina Piha-Paul, Hua Zhou, Eleni Kardala, et al. A deep learning framework for predicting response to therapy in cancer. *Cell reports*, 29(11): 3367–3373, 2019.

[35] Pengfei Liu, Hongjian Li, Shuai Li, and Kwong-Sak Leung. Improving prediction of phenotypic drug response on cancer cell lines using deep convolutional network. *BMC Bioinformatics*, 20 (1):408, 2019.

[36] Tuan Nguyen, Giang TT Nguyen, Thin Nguyen, and Duc-Hau Le. Graph convolutional networks for drug response prediction. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 19(1):146–154, 2022.

[37] Kristina Preuer, Richard PI Lewis, Sepp Hochreiter, Andreas Bender, Krishna C Bulusu, and Günter Klambauer. Deepsynergy: predicting anti-cancer drug synergy with deep learning. *Bioinformatics*, 34(9):1538–1546, 2018.

[38] Jinxian Wang, Xuejun Liu, Siyuan Shen, Lei Deng, and Hui Liu. Deepdds: deep graph neural network with attention mechanism to predict synergistic drug combinations. *Briefings in Bioinformatics*, 23(1):bbab390, 2022.

[39] Fatemeh Rafiei, Hojjat Zeraati, Karim Abbasi, Jahan B Ghasemi, Mahboubeh Parsaeian, and Ali Masoudi-Nejad. Deeptrasynergy: drug combinations using multimodal deep learning with transformers. *Bioinformatics*, 39(8):btad438, 2023.

[40] Halil Ibrahim Kuru, Oznur Tastan, and A Ercument Cicek. Matchmaker: A deep learning framework for drug synergy prediction. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 19(4):2334–2344, 2022.

[41] Mohamed Reda El Khili, Safyan Aman Memon, and Amin Emad. Marsy: a multitask deep-learning framework for prediction of drug combination synergy scores. *Bioinformatics*, 39(4): btad177, 2023.

[42] Heli Julkunen, Anna Cichońska, Prson Gautam, Sandor Szedmak, Jane Douat, Tapio Pahikkala, Tero Aittokallio, and Juho Rousu. Leveraging multi-way interactions for systematic prediction of pre-clinical drug combination effects. *Nature Communications*, 11(1):6136, 2020.

[43] Jianzhu Ma, Samson H Fong, Yunan Luo, Christopher J Bakkenist, John Paul Shen, Soufiane Mourragui, Lodewyk FA Wessels, Marc Hafner, Roded Sharan, Jian Peng, et al. Few-shot learning creates predictive models of drug response that translate from high-throughput screens to individual patients. *Nature Cancer*, 2(2):233–244, 2021.

[44] Casey Sederman, Tonya Di Sera, Yi Qiao, Xiaomeng Huang, Bryan E Welm, Alana L Welm, and Gabor Marth. Screendl: A transfer learning framework integrating tumor omics and functional drug screening for personalized clinical drug response prediction. *Cancer Research*, 84(6_Supplement):907–907, 2024.

[45] Soufiane MC Mourragui, Marco Loog, Daniel J Vis, Kat Moore, Anna G Manjon, Mark A van de Wiel, Marcel JT Reinders, and Lodewyk FA Wessels. Predicting patient response with models trained on cell lines and patient-derived xenografts by nonlinear transfer learning. *Proceedings of the National Academy of Sciences*, 118(49):e2106682118, 2021.

[46] Myung Jun Kim, Léo Grinsztajn, and Gaël Varoquaux. CARTE: Pretraining and transfer for tabular learning. In *International Conference on Machine Learning*, 2024.

[47] Junwei Ma, Valentin Thomas, Rasa Hosseinzadeh, Alex Labach, Jesse C Cresswell, Keyvan Golestan, Guangwei Yu, Anthony L Caterini, and Maksims Volkovs. Tabdpt: Scaling tabular foundation models on real data. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[48] Tianyu Liu, Tinyi Chu, Xiao Luo, and Hongyu Zhao. Building a unified model for drug synergy analysis powered by large language models. *Nature Communications*, 16(1):4537, 2025.

[49] Tianhao Li, Sandesh Shetty, Advaith Kamath, Ajay Jaiswal, Xiaoqian Jiang, Ying Ding, and Yejin Kim. Cancergpt for few shot drug pair synergy prediction using large pretrained language models. *NPJ Digital Medicine*, 7(1):40, 2024.

[50] Carl Edwards, Aakanksha Naik, Tushar Khot, Martin D Burke, Heng Ji, and Tom Hope. Synergpt: In-context learning for personalized drug synergy prediction and drug design. In *First Conference on Language Modeling*, 2024.

[51] Daniel Reker and Gisbert Schneider. Active-learning strategies in computer-assisted drug discovery. *Drug discovery today*, 20(4):458–465, 2015.

[52] David E Graff, Eugene I Shakhnovich, and Connor W Coley. Accelerating high-throughput virtual screening through molecular pool-based active learning. *Chemical science*, 12(22): 7866–7881, 2021.

[53] Ying Yang, Kun Yao, Matthew P Repasky, Karl Leswing, Robert Abel, Brian K Shoichet, and Steven V Jerome. Efficient exploration of chemical space with docking and deep learning. *Journal of Chemical Theory and Computation*, 17(11):7106–7119, 2021.

[54] Christopher Tosh, Mauricio Tec, Jessica B White, Jeffrey F Quinn, Glorymar Ibanez Sanchez, Paul Calder, Andrew L Kung, Filemon S Dela Cruz, and Wesley Tansey. A bayesian active learning platform for scalable combination drug screens. *Nature Communications*, 16(1):156, 2025.

[55] Paul Bertin, Jarrid Rector-Brooks, Deepak Sharma, Thomas Gaudelet, Andrew Anighoro, Torsten Gross, Francisco Martínez-Peña, Eileen L Tang, MS Suraj, Cristian Regep, et al. RECOVER identifies synergistic drug combinations in vitro through sequential model optimization. *Cell Reports Methods*, 3(10), 2023.

[56] Shuhui Wang, Alexandre Allauzen, Philippe Nghe, and Vaitea Opuu. A guide for active learning in synergistic drug discovery. *Scientific Reports*, 15(1):3484, 2025.

[57] David D Lewis. A sequential algorithm for training text classifiers: Corrigendum and additional data. In *Acm Sigir Forum*, volume 29, pages 13–19. ACM New York, NY, USA, 1995.

[58] Burr Settles. Active learning literature survey. 2009.

[59] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, pages 1050–1059. PMLR, 2016.

[60] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and scalable predictive uncertainty estimation using deep ensembles. *Advances in neural information processing systems*, 30, 2017.

[61] David Rogers and Mathew Hahn. Extended-connectivity fingerprints. *Journal of chemical information and modeling*, 50(5):742–754, 2010.

[62] Tianduanyi Wang, Sandor Szedmak, Haishan Wang, Tero Aittokallio, Tapio Pahikkala, Anna Cichonska, and Juho Rousu. Modeling drug combination effects via latent tensor reconstruction. *Bioinformatics*, 37(Supplement_1):i93–i101, 2021.

[63] Leiv Rønneberg and Vidhi Lalchand. Scalable permutation invariant multi-output gaussian processes for cancer drug response. In *NeurIPS 2024 Workshop on Bayesian Decision-making and Uncertainty*, 2024.

[64] Thomas M Norman, Max A Horlbeck, Joseph M Replogle, Alex Y Ge, Albert Xu, Marco Jost, Luke A Gilbert, and Jonathan S Weissman. Exploring genetic interaction manifolds constructed from rich single-cell phenotypes. *Science*, 365(6455):786–793, 2019.

[65] Daniel Bottomly, Nicola Long, Anna Reister Schultz, Stephen E. Kurtz, Cristina E. Tognon, Kara Johnson, Melissa Abel, Anupriya Agarwal, Sammantha Avaylon, Erik Benton, et al. Integrative analysis of drug response and clinical outcome in acute myeloid leukemia. *Cancer Cell*, 40(8):850–864, 2022.

[66] Nishanth Ulhas Nair, Patricia Greninger, Xiaohu Zhang, Adam A. Friedman, Arnaud Amzallag, Eliane Cortez, Avinash Das Sahu, Joo Sang Lee, Anahita Dastur, Regina K. Egan, et al. A landscape of response to drug combinations in non-small cell lung cancer. *Nature Communications*, 14(1), 2023.

[67] Johannes Betge, Niklas Rindtorff, Jan Sauer, Benedikt Rauscher, Clara Dingert, Haristi Gaitantzi, Frank Herweck, Kauthar Srour-Mhanna, Thilo Miersch, Erica Valentini, et al. The drug-induced phenotypic landscape of colorectal cancer organoids. *Nature Communications*, 13(1), 2022.

[68] Alejandra Bruna, Oscar M. Rueda, Wendy Greenwood, Ankita Sati Batra, Maurizio Callari, Rajbir Nath Batra, Katherine Pogrebniak, Jose Sandoval, John W. Cassidy, Ana Tufegdzic-Vidakovic, et al. A biobank of breast cancer explants with preserved intra-tumor heterogeneity to screen anticancer compounds. *Cell*, 167(1):260–274, 2016.

[69] Jordi Barretina, Giordano Caponigro, Nicolas Stransky, Kavitha Venkatesan, Adam A. Margolin, Sungjoon Kim, Christopher J. Wilson, Joseph Lehár, Gregory V. Kryukov, Dmitriy Sonkin, et al. The Cancer Cell Line Encyclopedia enables predictive modelling of anticancer drug sensitivity. *Nature*, 483(7391):603–607, 2012.

[70] Amrita Basu, Nicole E. Bodycombe, Jaime H. Cheah, Edmund V. Price, Ke Liu, Giannina I. Schaefer, Richard Y. Ebright, Michelle L. Stewart, Daisuke Ito, Stephanie Wang, et al. An interactive resource to identify cancer genetic and lineage dependencies targeted by small molecules. *Cell*, 154(5):1151–1161, 2013.

[71] Brinton Seashore-Ludlow, Matthew G. Rees, Jaime H. Cheah, Murat Cokol, Edmund V. Price, Matthew E. Coletti, Victor Jones, Nicole E. Bodycombe, Christian K. Soule, Joshua Gould, et al. Harnessing connectivity in a large-scale small-molecule sensitivity dataset. *Cancer Discovery*, 5(11):1210–1223, 2015.

[72] Chelsea Mayoh, Jie Mao, Jinhan Xie, Gábor Tax, Shu-Oi Chow, Roxanne Cadiz, Karina Pazaky, Paulette Barahona, Pamela Ajuyah, Peter Trebilcock, Angela Malquori, Kate Gunther, Anica Avila, Doo-Young Yun, Stephanie Alfred, Anjana Gopalakrishnan, Alvin Kamili, Marie Wong, Mark J. Cowley, Sophie Jessop, Loretta M S Lau, Toby N. Trahair, David S. Ziegler, Jamie I.

Fletcher, Andrew J. Gifford, Maria Tsoli, Glenn M. Marshall, Michelle Haber, Vanessa Tyrrell, Tim W. Failes, Greg M Arndt, Richard B. Lock, Paul G. Ekert, and M Emmy M Dolman. High-throughput drug screening of primary tumor cells identifies therapeutic strategies for treating children with high-risk cancer. *Cancer Research*, 83:2716–2732, 2023.

[73] Adam A. Friedman, Arnaud Amzallag, Iulian Pruteanu-Malinici, Subash Baniya, Zachary A. Cooper, Adriano Piris, Leeza Hargreaves, Vivien Igras, Dennie T. Frederick, Donald P. Lawrence, et al. Landscape of targeted anti-cancer drug synergies in melanoma identifies a novel BRAF-VEGFR/PDGFR combination treatment. *PLOS ONE*, 10(10):e0140310, 2015.

[74] Adam A. Friedman, Yun Xia, Lorenzo Trippa, Long Phi Le, Vivien Igras, Dennie T. Frederick, Jennifer A. Wargo, Kenneth K. Tanabe, Donald P. Lawrence, Donna S. Neuberg, et al. Feasibility of ultra-high-throughput functional screening of melanoma biopsies for discovery of novel cancer drug combinations. *Clinical Cancer Research*, 23(16):4680–4692, 2017.

[75] Wanjuan Yang, Jorge Soares, Patricia Greninger, Elena J. Edelman, Howard Lightfoot, Simon Forbes, Nidhi Bindal, Dave Beare, James A. Smith, I. Richard Thompson, et al. Genomics of Drug Sensitivity in Cancer (GDSC): A resource for therapeutic biomarker discovery in cancer cells. *Nucleic Acids Research*, 41(D1):D955–D961, 2013.

[76] Helen H.N. Yan, Hoi Cheong Siu, Simon Law, Siu Lun Ho, Sarah S.K. Yue, Wai Yin Tsui, Dessy Chan, April S. Chan, Stephanie Ma, Ka On Lam, et al. A comprehensive human gastric cancer organoid biobank captures tumor subtype heterogeneity and enables therapeutic screening. *Cell Stem Cell*, 23(6):882–897, 2018.

[77] Nora Liebers, Peter-Martin Bruch, Tobias Terzer, Miguel Hernandez-Hernandez, Nagarajan Paramasivam, Donnacha Fitzgerald, Heidi Altmann, Tobias Roider, Carolin Kolb, Mareike Knoll, et al. Ex vivo drug response profiling for response and outcome prediction in hematologic malignancies: the prospective non-interventional SMARTrial. *Nature Cancer*, 4:1648–1659, 2023.

[78] Ryan J. Ice, Michelle Chen, Max Sidorov, Tam Le Ho, Rinette W. L. Woo, Aida Rodriguez-Brotons, Tri Luu, Damon Jian, Kevin B. Kim, Stanley P. Leong, et al. Drug responses are conserved across patient-derived xenograft models of melanoma leading to identification of novel drug combination therapies. *British Journal of Cancer*, 122(5):648–657, 2019.

[79] Gustave Ronteix, Jean Bouteiller, Anna-Rose Gryspeert, Alice Boileve, Cartry Jerome, Jacques RR Mathieu, Sabrina Bedja, Michel Pierre Ducreux, and Fanny Jaulin. Cohort-level patient-derived organoid assays and prediction of clinical trial outcomes in pdac, 2025.

[80] Reid T. Powell, Abena Redwood, Xuan Liu, Lei Guo, Shirong Cai, Xinhui Zhou, Yizheng Tu, Xiaomei Zhang, Yuan Qi, Yan Jiang, et al. Pharmacologic profiling of patient-derived xenograft models of primary treatment-naïve triple-negative breast cancer. *Scientific Reports*, 10(1), 2020.

[81] Disha Malani, Ashwini Kumar, Oscar Brück, Mika Kontro, Bhagwan Yadav, Monica Hellesøy, Heikki Kuusanmäki, Olli Dufva, Matti Kankainen, Samuli Eldfors, et al. Implementing a functional precision medicine tumor board for acute myeloid leukemia. *Cancer Discovery*, 12 (2):388–401, 2021.

[82] Kohta Toshimitsu, Ai Takano, Masayuki Fujii, Kazuhiro Togasaki, Mami Matano, Sirirat Takahashi, Takanori Kanai, and Toshiro Sato. Organoid screening reveals epigenetic vulnerabilities in human colorectal cancer. *Nature Chemical Biology*, 18(6):605–614, 2022.

[83] Ling Li, Hildur Knutsdottir, Ken Hui, Matthew J. Weiss, Jin He, Benjamin Philosophe, Andrew M. Cameron, Christopher L. Wolfgang, Timothy M. Pawlik, Gabriel Ghiaur, et al. Human primary liver cancer organoids reveal intratumor and interpatient drug response heterogeneity. *JCI Insight*, 4(2), 2019.

[84] Suk Hyung Lee, Wenhuo Hu, Justin T. Matulay, Mark V. Silva, Tomasz B. Owczarek, Kwanghee Kim, Chee Wai Chua, LaMont J. Barlow, Cyriac Kandoth, Alanna B. Williams, et al. Tumor evolution and drug response in patient-derived organoid models of bladder cancer. *Cell*, 173(2): 515–528, 2018.

[85] Ahmad Al Shihabi, Peyton J. Tebon, Huyen Thi Lam Nguyen, Jomjit Chantharasamee, Sara Sartini, Ardalan Davarifar, Alexandra Y Jensen, Miranda Diaz-Infante, Hannah Cox, Alfredo Gonzalez, et al. The landscape of drug sensitivity and resistance in sarcoma. *Cell stem cell*, 31 (10):1524–1542, 2024.

[86] Ziyue Gu, Yanli Yao, Guizhu Yang, Guopei Zhu, Zhen Tian, Rui Wang, Qi Wu, Yujue Wang, Yaping Wu, Lan Chen, et al. Pharmacogenomic landscape of head and neck squamous cell carcinoma informs precision oncology therapy. *Science Translational Medicine*, 14(661): eabo5987, 2022.

[87] Eric Polley, Mark Kunkel, David Evans, Thomas Silvers, Rene Delosh, Julie Laudeman, Chad Ogle, Russell Reinhart, Michael Selby, John Connelly, et al. Small cell lung cancer screen of oncology drugs, investigational agents, and gene and microRNA expression. *Journal of the National Cancer Institute*, 108(10), 2016.

[88] Tea Pemovska, Mika Kontro, Bhagwan Yadav, Henrik Edgren, Samuli Eldfors, Agnieszka Szwajda, Henrikki Almusa, Maxim M. Bespalov, Pekka Ellonen, Erkki Elonen, et al. Individualized systems medicine strategy to tailor treatments for patients with chemorefractory acute myeloid leukemia. *Cancer Discovery*, 3(12):1416–1429, 2013.

[89] Antoine de Mathelin, Nicolas Enrique Cecchi, François Deheeger, Mathilde Mougeot, and Nicolas Vayatis. OneBatchPAM: A Fast and Frugal K-Medoids Algorithm. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(15):16172–16180, April 2025.

[90] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[91] Conghao Wang, Gaurav Asok Kumar, and Jagath C Rajapakse. Drug discovery and mechanism prediction with explainable graph neural networks. *Scientific Reports*, 15(1):179, 2025.

[92] Peng Zhang and Shikui Tu. Mgae-dc: Predicting the synergistic effects of drug combinations through multi-channel graph autoencoders. *PLoS computational biology*, 19(3):e1010951, 2023.

[93] Shiyu Yan and Ding Zheng. A deep neural network for predicting synergistic drug combinations on cancer. *Interdisciplinary Sciences: Computational Life Sciences*, 16(1):218–230, 2024.

[94] CA Hafsath and AS Jereesh. Perturbsynx: Deep learning framework for predicting drug combination synergy scores using drug induced gene perturbation data. *Computational Biology and Chemistry*, page 108736, 2025.

[95] A Colin Cameron and Douglas L Miller. A practitioner's guide to cluster-robust inference. *Journal of Human Resources*, 50(2):317–372, 2015.

[96] Antoine de Mathelin, François Deheeger, Mathilde Mougeot, and Nicolas Vayatis. Discrepancy-based active learning for domain adaptation. In *International Conference on Learning Representations*, 2022.

A Model architecture details

Figure 4 provides a layer-by-layer view of the ScreenShot architecture (∼7M parameters). All three encoders (Drug Combination Encoder, DCE; Response Encoder, RE; Sample Encoder, SE) share the same transformer block configuration: 5 layers, 8 attention heads (head dim 32), hidden dim 256, GeLU activations, residual connections and layer normalization. Given n context observations for a sample, we group them by perturbation: observations from the same drug combination are placed together. This yields p unique perturbations, each with up to m dose-level observations, where each perturbation involves up to q drugs. The context is stored as a padded tensor of shape (p, m, q); p, m, and q are the maximum values across the batch, and padding masks are applied at every level. In particular, q is padded because datasets often contain both monotherapy (q=1) and combination (q=2 or 3) observations. A learnable CLS token is appended to the perturbation sequence before the Sample Encoder; its output after self-attention is used as the pooled sample representation.

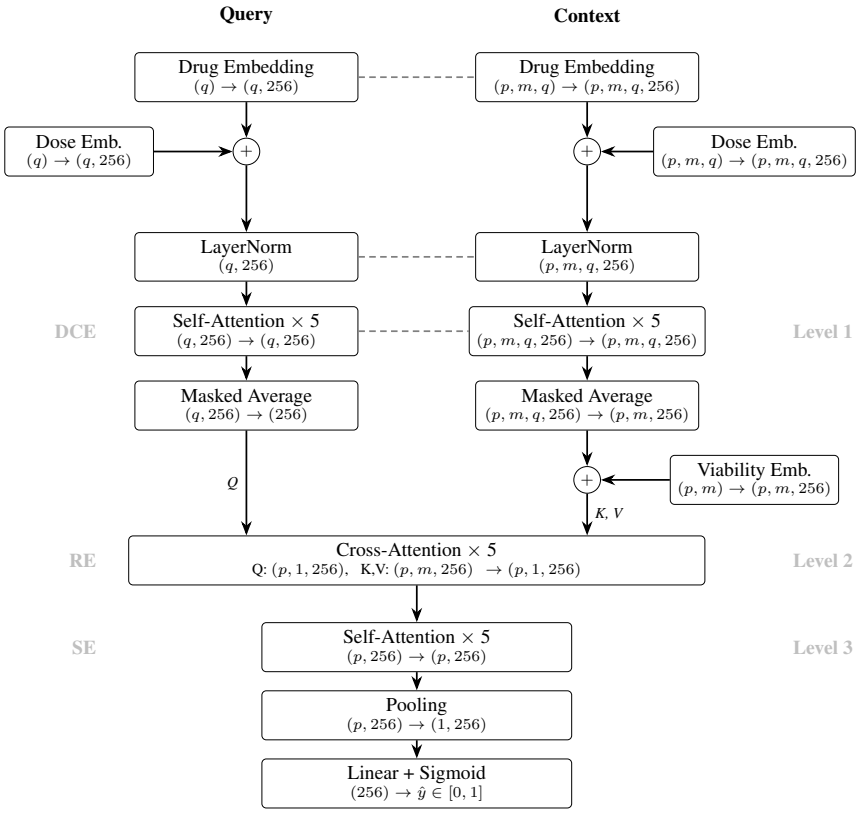

Figure 4: Layer-by-layer architecture of ScreenShot (7.0M parameters). Query and context follow identical pipelines through Level 1 (dashed lines = shared weights). DCE = Drug Combination Encoder, RE = Response Encoder, SE = Sample Encoder. q: drugs per combination, p: unique perturbations, m: observations per perturbation. Dose and viability use trainable Fourier features. Drug embedding vocabulary: 3,779 entries (3,679 drugs + 100 UNK).

Fourier encoding. Continuous scalars (doses and viabilities) are encoded using learned Fourier features:

$$\phi(x) = W\big[\sin(x \cdot \boldsymbol{\omega}),\ \cos(x \cdot \boldsymbol{\omega})\big] + \mathbf{b}, \tag{2}$$

where $\boldsymbol{\omega} \in \mathbb{R}^F$, $W \in \mathbb{R}^{D \times 2F}$, and $\mathbf{b} \in \mathbb{R}^D$ are learnable parameters (with independent parameters for doses and for viabilities). We use F=64 Fourier frequencies.

B Pretraining

Pretraining corpus. Table 2 lists all datasets in the pretraining and evaluation corpus. The ex vivo datasets were obtained on request from Pichotta et al. [21], and we added publicly available cell line datasets for a total of 40 studies spanning both monotherapy and combination screens across diverse cancer types.

Table 2: Pretraining corpus: 40 drug screening datasets. †Held out for evaluation. CL = cell line, EV = ex vivo (patient-derived). *Number of unique consensus drug IDs after PubChem resolution.

Dataset	Reference	Samples	Drugs	Obs.	Model	Type
Beat AML	Bottomly et al. [65]	631	166	491K	EV	Mono
Nair et al.	Nair et al. [66]	81	244	3.6M	CL	Combi
Betge et al.	Betge et al. [67]	14	520	26K	EV	Mono
Martins et al.	Martins et al. [11]	25	12	6K	EV	Mono
Bruna et al.	Bruna et al. [68]	21	105	42K	EV	Combi
CCLE	Barretina et al. [69]	504	24	93K	CL	Mono
Driehuis et al.	Driehuis et al. [6]	24	78	35K	EV	Combi
CTRP1	Basu et al. [70]	242	203	257K	CL	Mono
CTRP2	Seashore-Ludlow et al. [71]	887	545	12.4M	CL	Mono
Mayoh et al.	Mayoh et al. [72]	151	144	35K	EV	Mono
Friedman et al.'15	Friedman et al. [73]	37	108	419K	EV	Combi
Friedman et al.'17	Friedman et al. [74]	20	76	105K	EV	Combi
GDSC1	Yang et al. [75]	988	304	2.6M	CL	Mono
GDSC2	Yang et al. [75]	810	169	1.4M	CL	Mono
Murumägi et al.	Murumägi et al. [12]	13	641	45K	EV	Mono
Sa et al.	Sa et al. [10]	45	97	42K	EV	Mono
Yan et al.	Yan et al. [76]	9	37	7K	EV	Mono
SMarTrial	Liebers et al. [77]	180	108	66K	EV	Combi
Lau et al.	Lau et al. [14]	17	194	21K	EV	Mono
Ice et al.	Ice et al. [78]	8	38	5K	EV	Mono
Merck	O'Neil et al. [15]	39	38	1.5M	CL	Combi
Lee et al.'18	Lee et al. [3]	284	67	234K	EV	Mono
Johansson et al.	Johansson et al. [4]	86	265	202K	EV	Mono
Ronteix et al.	Ronteix et al. [79]	158	29	35K	EV	Mono
Powell et al.	Powell et al. [80]	16	559	118K	EV	Mono
Malani et al.	Malani et al. [81]	181	539	369K	EV	Mono
Toshimitsu et al.	Toshimitsu et al. [82]	26	56	31K	EV	Mono
Hirt et al.	Hirt et al. [7]	18	25	3K	EV	Mono
Li et al.	Li et al. [83]	27	129	3K	EV	Mono
Lee et al.'18b	Lee et al. [84]	11	21	5K	EV	Mono
Al Shihabi et al.	Shihabi et al. [85]	109	530	21K	EV	Combi
Gu et al.	Gu et al. [86]	13	2,245	34K	EV	Mono
Polley et al.	Polley et al. [87]	70	523	389K	CL	Mono
Tiriac et al.	Tiriac et al. [5]	68	5	14K	EV	Mono
Peterziel et al.	Peterziel et al. [13]	65	79	21K	EV	Mono
Pemovska et al.	Pemovska et al. [88]	35	222	31K	EV	Mono
BATCHIE†	Tosh et al. [54]	16	207	148K	CL	Combi
GDSC-SQ†	Jaaks et al. [17]	126	65	3.7M	CL	Combi
NCI-ALMANAC†	Holbeck et al. [16]	60	102	3.4M	CL	Combi
PDO-Breast†	Guillen et al. [1]	16	46	81K	EV	Mono
Total		**6,135**	**3,679***	**32.1M**		

Data preprocessing. All viability values are clipped to $[0, 1]$. Dose concentrations are expressed in micromolar (μM) and converted to $\log_{10}(\mu\text{M})$ scale, clipped to $[-6, 4]$ (corresponding to 10^{-6}–10^{4} μM), and linearly rescaled to $[0, 1]$. Drug names across datasets are resolved to consensus identifiers via PubChem lookup (matching by compound ID, substance ID, parent compound, or canonical SMILES). To remove noisy observations, we fit a monotone decreasing isotonic regression to each (sample, drug combination) dose-response curve and discard curves with MAE above 0.2, which removes approximately 5% of the data.

Leave-one-out evaluation. For each evaluation dataset, we train a separate foundation model on all remaining datasets. The BATCHIE and PDO-Breast evaluations use the same pretrained model (trained on all datasets excluding BATCHIE and PDO-Breast). For GDSC-SQ and NCI-ALMANAC, we train on all datasets except the evaluation set itself. All models use identical hyperparameters (Section E.2).

Data sampling. At each training step, one dataset is sampled uniformly, and a batch is constructed by randomly selecting 1–16 biological samples, up to 100 unique drug combinations per sample, and up to 10 observations per combination. A query set of 32 observations per sample is drawn separately for supervision. Each dataset contributes 100 batches per epoch, yielding ∼4,000 batches per epoch and ∼400K iterations over 100 epochs.

UNK token training. To prepare the model for unknown drugs at inference time, a random fraction $\alpha \sim \text{Uniform}[0, 0.2]$ of unique drug identities per batch are replaced with one of the 100 trainable UNK tokens during pretraining. This teaches the model to produce reasonable predictions even when a drug has no specific embedding, by relying on the few-shot context observations to infer the drug's behavior.

Training curve. Figure 5 shows the training curve for the BATCHIE foundation model. Both train and held-out MAE decrease steadily throughout training with no sign of overfitting.

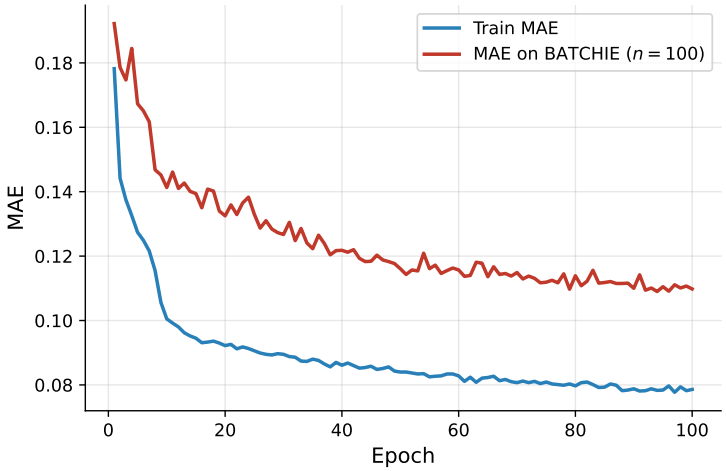

Figure 5: Foundation model training curve (BATCHIE). Train MAE and held-out MAE on BATCHIE over 100 epochs, where the held-out evaluation uses n=100 few-shot context observations per sample, uniformly sampled.

C Drug coverage

Table 3 reports the overlap between each pretrained drug library and the drugs in the corresponding evaluation dataset. Each model's library contains ∼3,650–3,680 unique consensus drug IDs, resolved through PubChem to handle name aliases across studies. Drug names are mapped to PubChem compound IDs (CIDs) and substance IDs (SIDs); drugs sharing the same CID, SID, parent compound, or canonical SMILES are assigned a single consensus ID. Coverage ranges from 89.4% (BATCHIE) to 100% (GDSC-SQ), with most evaluation drugs already present in the pretrained vocabulary. The uncovered drugs are investigational compounds that appear in few studies (e.g., CYC065, LY3295668) and are therefore absent from the pretraining corpus. These drugs are assigned dedicated UNK embeddings.

Table 4 evaluates generalization to drugs unseen during pretraining on BATCHIE, the evaluation dataset with the most such drugs (22 drugs, ∼18% of observations). We stratify every held-out combination by how much of its content is unseen: *known* (no unseen drug), *partially unknown* (exactly one unseen drug), and *fully unknown* (all constituent drugs unseen), and aggregate the error within each group. This is a demanding test for ScreenShot: for unseen drugs it has no drug-specific embedding (only the UNK token), whereas XGBoost and TabPFN still receive the full Morgan fingerprint, and the target-only MLP learns its drug embeddings directly from the few-shot context. ScreenShot is the most accurate method on known and partially unknown combinations at every budget. On fully unknown combinations (∼1% of observations), it leads at n=10 but is overtaken by the target-only MLP at larger budgets, while remaining second best, ahead of MLP-FT, XGBoost, and

Table 3: **Drug coverage.** For each evaluation dataset, the number of unique drugs, the pretrained library size (consensus drug IDs), and the number of drugs resolved to a known embedding vs. assigned a UNK token.

Dataset	Library size	Eval drugs	Known (%)	UNK
BATCHIE	3,654	207	185 (89%)	22
GDSC-SQ	3,679	65	65 (100%)	0
NCI-ALMANAC	3,675	102	98 (96%)	4
PDO-Breast	3,654	46	44 (96%)	2

TabPFN. This regime, in which every drug of a combination is novel, is not the setting ScreenShot is designed for, and can be addressed by extending the pretraining corpus with studies that include the missing drugs.

Table 4: **Generalization to drugs unseen during pretraining (BATCHIE).** Held-out MAE (mean±std over 5 seeds), with every held-out combination stratified by how many of its drugs are absent from the pretraining corpus. **Bold**: best per column within each group.

Method	n=10	n=50	n=100	n=200	n=300
Known: no unseen drug (∼94K observations)					
XGBoost	0.326±0.004	0.258±0.003	0.226±0.002	0.188±0.001	0.168±0.001
TabPFN	0.312±0.002	0.279±0.004	0.255±0.005	0.218±0.003	0.198±0.001
MLP	0.313±0.002	0.278±0.005	0.240±0.008	0.200±0.003	0.176±0.006
MLP-FT	0.202±0.017	0.167±0.006	0.153±0.004	0.139±0.002	0.131±0.003
ScreenShot	**0.146±0.003**	**0.117±0.002**	**0.108±0.002**	**0.098±0.001**	**0.093±0.001**
Partially unknown: exactly one unseen drug (∼19K observations)					
XGBoost	0.306±0.011	0.256±0.002	0.227±0.006	0.194±0.003	0.178±0.002
TabPFN	0.289±0.005	0.265±0.002	0.249±0.005	0.219±0.004	0.204±0.004
MLP	0.299±0.006	0.257±0.008	0.223±0.007	0.183±0.004	0.162±0.004
MLP-FT	0.221±0.006	0.195±0.004	0.184±0.002	0.170±0.001	0.162±0.002
ScreenShot	**0.200±0.003**	**0.173±0.005**	**0.164±0.004**	**0.154±0.002**	**0.149±0.001**
Fully unknown: all drugs unseen (∼1.6K observations)					
XGBoost	0.344±0.020	0.306±0.005	0.282±0.011	0.259±0.012	0.248±0.010
TabPFN	0.336±0.011	0.313±0.008	0.297±0.008	0.269±0.005	0.256±0.008
MLP	0.320±0.007	**0.266±0.024**	**0.236±0.013**	**0.188±0.014**	**0.173±0.013**
MLP-FT	0.302±0.024	0.287±0.009	0.281±0.011	0.255±0.010	0.242±0.017
ScreenShot	**0.278±0.007**	0.267±0.007	0.258±0.006	0.244±0.007	0.233±0.006

To assess whether using a pretrained UNK token is useful, we compare it against assigning a random known drug token to each unknown drug. For each budget (100, 200, 300), we run ScreenShot inference on BATCHIE with randomly selected few-shot context and compute the absolute error on all remaining observations. We repeat with 5 random seeds, each with a different random UNK-to-known-drug assignment, and average the error across seeds. Random tokens perform worse than pretrained UNK embeddings (+0.017 MAE at budget 300), confirming that an incorrect drug identity is more harmful than a non-specific placeholder, because a wrong identity may actively mislead the model's dose-response expectations.

D Experimental setup details

D.1 Samples

Table 5 lists the biological samples used for evaluation in each dataset. BATCHIE and PDO-Breast use all available samples. For NCI-ALMANAC and GDSC-SQ, we randomly select 10 cell lines for faster evaluation.

Table 5: Evaluation samples and control samples per dataset.

Dataset	N	Control(s)	Sample IDs
BATCHIE	16	RPE, BJ	A673, Kelly, MDA-MB-231, MG-63, MSKEWS-33838, MSKEWS-66647, MSKOST-11890, MSKRMS-12808, SAOS-2, SJSA-1, SKNEP, TC-71, U2OS, Wit49
PDO-Breast	16	HCI-011	HCI-001, HCI-002, HCI-003, HCI-005, HCI-008, HCI-010, HCI-012, HCI-015, HCI-016, HCI-017, HCI-019, HCI-023, HCI-024, HCI-025, HCI-027
NCI-ALMANAC	10	UACC-257	786-0, CAKI-1, HCT-15, HL-60(TB), NCI-H522, OVCAR-4, SK-MEL-2, SK-OV-3, T-47D
GDSC-SQ	10	MDA-MB-157	CAL-148, DU-4475, HCC1395, HCC1599, HCC2218, HCC70, HDQ-P1, MCF7, T47D

Control samples. For BATCHIE, two non-cancerous cell lines (RPE and BJ) serve as controls; the control baseline is $\min(\text{RPE}, \text{BJ})$ per treatment. For NCI-ALMANAC, GDSC-SQ, and PDO-Breast, which lack designated controls, we use the cell line with the highest average viability across all treatments as a proxy control: UACC-257, MDA-MB-157, and HCI-011 respectively. This selection is based on the assumption that less drug-sensitive samples approximate non-cancerous controls.

D.2 Few-shot protocol

For each sample, we randomly select $n \in \{10, 50, 100, 150, 200, 250, 300\}$ observations as context and predict viability for all remaining observations. Few-shot experiments use 5 random seeds on BATCHIE and PDO-Breast, and 3 seeds on GDSC-SQ and NCI-ALMANAC.

D.3 Active learning protocol

The adaptive strategy (Algorithm 1) proceeds in multiple rounds. Active learning experiments use 5 seeds on all three datasets. Observations are selected uniformly at random without replacement within each sample. A fraction of the total budget (33% in the main experiments) is allocated to a cold-start batch; the remainder is split equally across 1 or 2 adaptive rounds.

Cold-start selection. The cold-start batch is selected by k-medoids clustering on the Level 1 drug-dose embeddings (output of the DCE, before any viability is observed). k-medoids selects k representatives from the candidate set by solving

$$\min_{S \subset \{1,\ldots,N\},\ |S|=k} \sum_{i=1}^{N} w_i \min_{j \in S} \|\mathbf{h}_i - \mathbf{h}_j\|^2, \tag{3}$$

where $\mathbf{h}_i$ is the embedding of treatment i and w_i is a per-treatment weight (uniform for cold start). We use the OneBatchPAM implementation [89]. For control samples, we always use k-medoids with uniform weights regardless of the strategy used for target samples, since the goal for controls is to maximize coverage rather than target specific hits.

Adaptive selection. In adaptive rounds, we use k-means++ seeding on the Level 3 context-conditioned embeddings (output of the SE after observing the current context). k-means++ sequentially samples centers with probability proportional to the weighted squared distance to the nearest existing center:

$$P(i) \propto w_i \min_{j \in S} \|\mathbf{h}_i - \mathbf{h}_j\|^2, \tag{4}$$

where S is the set of already-selected centers. We use a custom implementation built on top of scikit-learn [90], extended to support frozen centers and per-point weights. Treatments selected in previous rounds are included as fixed (frozen) initial centers, so new selections are pushed away from already-observed regions.

The key difference between k-medoids and k-means++ is in their objective. k-medoids minimizes the total distance between every data point and its closest medoid, which promotes both diversity and

representativity of the selection. k-means++ only considers distances between selected points, which emphasizes diversity over representativity: the selected set tends to be maximally spread out. When the goal is to find top hits rather than uniformly cover the treatment space, diversity is preferable because it avoids selecting redundant treatments in dense but uninteresting regions.

Weighting strategies. We consider three weighting schemes for adaptive rounds: (i) *Delta weighting*: $w_i = \max(-\Delta_i, 0)$, where $\Delta_i = \hat{y}_i^{(\text{target})} - \hat{y}_i^{(\text{control})}$ is the predicted differential viability, so treatments predicted to be selectively effective receive higher weight. (ii) *Uncertainty weighting*: $w_i = \sigma_i$, where σ_i is the MC dropout uncertainty [59] computed by enabling dropout (rate 0.1) on the Level 3 Sample Encoder and running 10 stochastic forward passes; the prediction is the mean and σ_i is the standard deviation across passes. (iii) *Top-k*: instead of weighted clustering, we greedily select the k treatments with the most negative Δ_i (top-k delta) or highest σ_i (top-k uncertainty), without any diversity mechanism.

E Baselines and hyperparameters

E.1 Comparison with related methods

Table 6 provides a systematic comparison of drug response prediction methods from the literature. Most existing methods require genomic or transcriptomic features to represent biological samples, which limits their applicability to patient-derived ex vivo models where such is often unavailable [21]. Monotherapy methods do not model drug interactions, and combination synergy methods typically predict a scalar synergy score rather than full dose-response surfaces. Transfer learning approaches partially bridge the domain gap between cell lines and patient samples, but still require overlapping molecular features. TabPFN [24], XGBoost, and MLPs all meet the requirements for our setting (no genomics, dose-aware, combination-capable) when paired with a general drug representation such as Morgan fingerprints or learned embeddings; we include them as baselines. TabPFN additionally shares the in-context learning paradigm with ScreenShot. ComboLTR and PIICM produce full dose-response predictions for combinations without genomic features, but they lack a pretraining mechanism and learn drug representations from scratch on each sample. This makes them effective when sufficient data is available per sample, but poorly suited to the few-shot regime where only a small number of observations are provided, which we confirm empirically in Section E.3.

E.2 Baseline hyperparameters

Input features for XGBoost and TabPFN. Both XGBoost and TabPFN use 512-dimensional Morgan fingerprints [61] (radius 2) as drug representations. For each drug in the combination, the fingerprint is concatenated with the corresponding dose value, yielding a $(512 + 1)$-dimensional vector per drug slot. For combinations of q drugs, the per-drug feature vectors are concatenated, producing a $513q$-dimensional input. Empty drug slots (monotherapy observations in a combination dataset) are filled with zero vectors.

XGBoost. We use XGBoost with 100 estimators and squared error loss. Hyperparameters are selected independently for each sample via 5-fold cross-validation with grid search over: max depth $\in \{2, 4, 6\}$, learning rate $\in \{0.05, 0.1, 0.3\}$, subsample ratio $\in \{0.7, 0.9, 1.0\}$, min child weight $\in \{1, 5, 10\}$, and L2 regularization $\lambda \in \{0.1, 1.0, 10.0\}$ (243 combinations).

TabPFN. We use TabPFN [24] with default settings and the same Morgan fingerprint input features as XGBoost. TabPFN performs in-context learning: it conditions on the few-shot observations and predicts the held-out viability in a single forward pass, with no task-specific training.

MLP architecture. Both MLP baselines share the same architecture, selected through extensive manual tuning on pretraining validation loss. The model uses learned embedding tables for drugs (vocabulary size matching the drug library) and samples, with embedding dimension D=64. Doses are encoded via a linear projection from $\mathbb{R}^1$ to $\mathbb{R}^{64}$. For each observation, the drug embedding and dose embedding are multiplied element-wise for each drug in the combination, then averaged across the q drug slots to produce a single drug-dose representation in $\mathbb{R}^{64}$. This is concatenated with the sample embedding to form a 128-dimensional input, which is passed through 3 fully connected layers

Table 6: **Comparison of drug response prediction methods.** We list methods cited in related work and indicate their requirements and capabilities. ✓ indicates a desirable property. Methods that are not genomics- or transcriptomics-free require molecular profiling, which limits applicability to patient-derived ex vivo models. Methods without pretraining must be trained from scratch on each new dataset. *Pretrained*: the method is pretrained on a large corpus before being applied to a new dataset. *Public weights*: pretrained weights are publicly available (– when the method has no pretraining).

Method	Combinations	Dose info	Pretrained	Public weights	Genomics-free	Transcriptomics-free
Monotherapy response prediction						
DNN [34]	✗	✗	✗	–	✓	✗
tCNNs [35]	✗	✗	✗	–	✗	✓
DrugCell [18]	✗	✗	✗	–	✗	✓
DeepCDR [19]	✗	✗	✗	–	✗	✗
HiDRA [20]	✗	✗	✗	–	✓	✗
GraphDRP [36]	✗	✗	✗	–	✗	✓
XGDP [91]	✗	✗	✓	✓	✓	✗
Combination synergy prediction						
DeepSynergy [37]	✓	✗	✗	–	✓	✗
comboFM / ComboLTR [42, 62]	✓	✓	✗	–	✓	✓
PIICM [63]	✓	✓	✗	–	✓	✓
MatchMaker [40]	✓	✗	✓	✓	✓	✗
DeepDDS [38]	✓	✗	✗	–	✓	✗
MARSY [41]	✓	✗	✗	–	✓	✗
DeepTraSynergy [39]	✓	✗	✗	–	✓	✗
MGAE-DC [92]	✓	✗	✓	✓	✓	✗
DconnC [93]	✓	✗	✗	–	✓	✗
PerturbSynX [94]	✓	✓	✓	✓	✓	✗
Transfer learning						
TRANSACT [45]	✗	✗	✓	✓	✓	✗
TCRP [43]	✗	✗	✓	✓	✗	✗
ScreenDL [44]	✗	✗	✓	✓	✓	✗
Foundation models						
PPC-FM [21]	✗	✓	✓	✗	✓	✓
BAITSAO [48]	✓	✗	✓	✓	✓	✓
CancerGPT [49]	✓	✗	✓	✗	✓	✓
SynerGPT [50]	✓	✗	✓	✗	✓	✓
TabPFN [24]	✓	✓	✓	✓	✓	✓
ScreenShot (ours)	✓	✓	✓	✓	✓	✓

(hidden dimension 128, ReLU activations, dropout 0.2) followed by a linear output layer and sigmoid activation.

MLP (target-only). Trained from scratch on each target sample using Adam (lr $= 10^{-3}$, no weight decay), gradient clipping at 1.0, batch size 32, for up to 200 epochs. Each epoch consists of 100 iterations (sampling with replacement from the few-shot context). We use a 20% validation split for early stopping and select the checkpoint with the lowest validation loss.

MLP (fine-tuned). Same architecture, pretrained on the same 40-dataset corpus as ScreenShot (excluding the held-out evaluation dataset) using Adam (lr $= 10^{-3}$, no weight decay), batch size 128, gradient clipping at 1.0, for 50 epochs over the full corpus ($\sim$234K iterations per epoch, $\sim$11.7M total training steps). A 5% held-out split of (sample, drug) pairs is used for validation, and the checkpoint with the lowest validation loss is selected. The pretrained model uses the same drug library as ScreenShot for drug identity encoding. At evaluation time, a new sample embedding is initialized for the target sample, and the full model is fine-tuned on the few-shot context with a learning rate of 10^{-4} for up to 200 epochs with early stopping on a 20% validation split.

E.3 Domain-specific baselines

We evaluate two domain-specific methods under the same few-shot protocol as the other baselines: ComboLTR [62], a representative latent tensor factorization method (from the same family as comboFM [42]), and PIICM [63], a permutation-invariant multi-output Gaussian process. For each sample, both methods are fitted using the n randomly selected context observations and evaluated on the remaining observations, with results averaged over five random seeds.

For ComboLTR, we use the same Morgan fingerprints as for XGBoost and TabPFN. The method exhibits substantial overfitting to the few-shot context. We additionally evaluate a range of hyperparameter configurations, varying tensor degree, rank, regularization, and number of training epochs, but find no configuration that improves upon the default settings.

For PIICM, we center the viability measurements by subtracting the context mean before fitting and add this mean back to the predictions. This centering is necessary to avoid predictions reverting toward zero. We use a noise parameter of 0.1, a rank of $\min(50, n-1)$, and 200 optimization iterations.

As reported in Table 7, both methods substantially underperform ScreenShot and also fall below the generic baselines (Tables 10 and 11). These results should be interpreted in light of the mismatch between the evaluation setting and the regimes for which these methods were originally developed. In their original evaluations, both methods are trained using five-fold cross-validation within a single densely measured study, with 80% of the observations available for training. In contrast, our few-shot setting restricts training to a small context set, often comprising only a few hundred observations. This difference in data availability poses a substantial challenge for methods that rely on fitting complex interaction models from the available measurements.

PIICM also faces a computational limitation on NCI-ALMANAC. Although it is trained on only n context observations, the model jointly represents the full dose-response surface of every drug combination in the sample, so its computational cost scales with the size of the screen rather than with the number of observations. For NCI-ALMANAC, where each sample comprises approximately 5,000 drug combinations, the fitting procedure does not converge and fails on most samples.

Table 7: **Domain-specific few-shot baselines.** Hit recall (%) / MAE at budget n=300, mean over 5 seeds. PIICM fails numerically on most NCI-ALMANAC samples (n/a). **Bold**: best per column.

Method	BATCHIE	GDSC-SQ	NCI-ALMANAC
ComboLTR [62]	21.2 / 0.339	41.1 / 0.254	24.4 / 0.339
PIICM [63]	26.1 / 0.300	32.9 / 0.205	n/a
ScreenShot	**76.0 / 0.101**	**60.6 / 0.121**	**63.0 / 0.126**

F Few-shot additional results

Figure 6 shows the trade-off between top-hit recall and computation time for each method at a budget of 100 shots. ScreenShot achieves the highest accuracy on every dataset while being 5–50$\times$ faster than all baselines. MLP and MLP-FT require training on the few-shot context for each experiment, resulting in computation times of 10–20 minutes per experiment. XGBoost runs on CPU (8 cores) and takes 2–3 minutes. TabPFN is slower than ScreenShot despite also being inference-only, due to its larger model size and its use of self-attention across both context observations and input features. ScreenShot requires only a single forward pass through the pretrained model, completing in 8–68 seconds depending on dataset size.

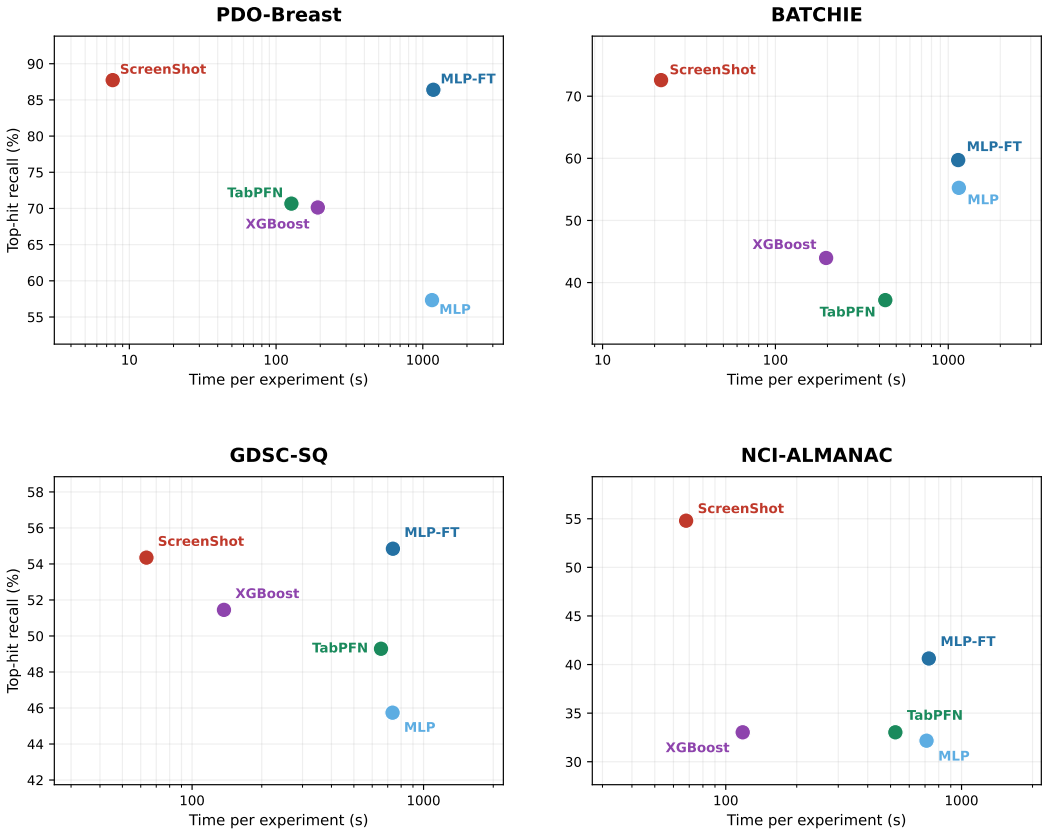

Figure 6: Top-hit recall vs. computation time per experiment at 100-shot budget. ScreenShot (red) is consistently the fastest and most accurate method across all four datasets. All GPU methods ran on NVIDIA A100 80GB GPUs; XGBoost ran on 8 CPU cores.

G Generalization to unseen drugs and combinations

The few-shot evaluation in Section 4.2 uses randomly sampled context observations. This protocol also allows us to assess generalization under more restrictive holdout settings, according to the overlap between the drugs in each query and those represented in the context. We distinguish two cases: *leave-drug-out*, where at least one drug in the query is absent from the context, and *leave-combination-out*, where all query drugs are represented in the context, but the queried combination is not. Table 8 reports the MAE for each setting on BATCHIE. ScreenShot achieves the lowest MAE in both settings across all context budgets. Generalization to drugs unseen during pretraining is evaluated separately in Section C (Table 4).

Table 8: **Leave-drug-out and leave-combination-out generalization (BATCHIE).** Held-out MAE (mean±std over 5 seeds) on queries for which one of the drugs never appears in the few-shot context (leave-drug-out), or all drugs appear but never together in this combination (leave-combination-out). **Bold**: best per column within each setting.

Method	n=10	n=50	n=100	n=200	n=300
Leave-drug-out					
XGBoost	0.342±0.006	0.316±0.004	0.301±0.005	0.277±0.003	0.267±0.005
TabPFN	0.325±0.004	0.313±0.003	0.302±0.005	0.279±0.002	0.264±0.005
MLP	0.329±0.002	0.319±0.002	0.302±0.009	0.280±0.012	0.261±0.009
MLP-FT	0.215±0.020	0.187±0.005	0.179±0.002	0.172±0.004	0.167±0.004
ScreenShot	**0.161±0.002**	**0.143±0.002**	**0.139±0.003**	**0.132±0.004**	**0.132±0.003**
Leave-combination-out					
XGBoost	0.259±0.015	0.225±0.007	0.229±0.007	0.224±0.003	0.215±0.002
TabPFN	0.252±0.016	0.262±0.009	0.259±0.008	0.242±0.004	0.223±0.001
MLP	0.243±0.020	0.252±0.016	0.232±0.006	0.216±0.002	0.201±0.004
MLP-FT	0.176±0.011	0.168±0.007	0.162±0.007	0.157±0.001	0.152±0.004
ScreenShot	**0.142±0.011**	**0.117±0.003**	**0.118±0.003**	**0.117±0.002**	**0.114±0.001**

H Hit recall with predicted control viabilities

Our primary hit recall evaluation uses ground-truth control viabilities (Section 4.1). To assess the sensitivity of our results to this choice, we repeat the evaluation using predicted control viabilities. Specifically, each method predicts both target and control responses under the same few-shot budget, and treatments are ranked according to the resulting predicted therapeutic index; the ground-truth hit labels remain unchanged. As shown in Table 9, using predicted controls reduces absolute hit recall across all methods and increases variability, particularly at the smallest context budget. Nevertheless, ScreenShot achieves the highest hit recall in 18 of the 20 dataset–budget combinations, compared with 16 of 20 when using ground-truth controls (Table 1).

We next consider a setting in which control responses are estimated from a larger, fixed budget of 300 observations, while the target sample remains few-shot. This setting reflects the practical availability of larger reference datasets for well-characterized control cell lines, in contrast to the typically more limited measurements available for patient-derived samples. Compared with the fully few-shot control setting (Table 9), this additional control data improves absolute hit recall, with gains concentrated at smaller target budgets. The average improvements are $+6.8$, $+5.0$, $+3.2$, $+1.4$, and $+0.0$ percentage points at target budgets of $n \in \{10, 50, 100, 200, 300\}$, respectively. As expected, there is no difference at $n = 300$, where the target and control budgets are identical. The largest improvements are observed on PDO-Breast, the most data-limited dataset, with gains of up to 39 percentage points at $n = 10$. Despite these changes in absolute performance, the relative ranking of methods remains stable: ScreenShot achieves the highest hit recall in 18 of the 20 dataset–budget combinations.

Table 9: **Hit recall (%) with predicted control viabilities.** Each method predicts both the target and the control response from the same number n of few-shot observations, and hits are ranked by the resulting predicted therapeutic index. Mean±std over seeds. **Bold**: best per column within each dataset.

Method	n=10	n=50	n=100	n=200	n=300
BATCHIE					
XGBoost	27.4±10.1	35.4±4.0	41.6±7.4	50.0±3.8	52.0±2.6
TabPFN	30.4±6.8	32.3±5.4	43.2±5.7	48.4±4.5	50.3±3.8
MLP	36.3±7.2	41.2±2.4	45.4±1.9	51.9±4.6	53.0±4.5
MLP-FT	38.9±7.3	46.2±6.9	50.8±4.1	57.6±0.8	60.1±3.6
ScreenShot	**51.6±5.4**	**59.0±2.8**	**61.1±1.5**	**66.8±2.0**	**67.6±1.5**
GDSC-SQ					
XGBoost	18.8±1.6	33.6±3.1	42.9±5.4	52.3±2.4	58.9±4.1
TabPFN	22.1±2.7	36.3±3.5	42.3±3.2	**54.5±3.8**	**61.5±3.7**
MLP	23.7±1.0	33.5±4.2	37.8±7.0	49.0±2.6	52.6±2.1
MLP-FT	21.7±0.7	30.7±4.9	39.4±2.6	48.5±1.9	55.5±3.2
ScreenShot	**27.8±0.6**	**40.6±0.1**	**45.0±0.9**	52.2±2.1	54.7±0.8
NCI-ALMANAC					
XGBoost	21.8±1.6	28.7±2.2	33.4±5.8	42.2±4.5	48.6±2.1
TabPFN	22.8±2.4	31.1±1.9	33.3±4.0	41.3±4.0	48.6±2.6
MLP	21.6±3.3	28.9±4.9	30.8±2.8	34.3±1.9	38.2±1.5
MLP-FT	29.0±2.6	31.5±2.5	32.6±4.5	39.3±2.3	39.5±2.9
ScreenShot	**33.7±9.8**	**40.7±10.3**	**45.4±4.8**	**51.8±2.5**	**55.1±5.2**
PDO-Breast					
XGBoost	29.3±8.8	42.4±5.9	58.9±10.9	73.3±10.8	84.8±6.9
TabPFN	31.2±8.2	45.9±6.1	55.2±9.0	74.4±9.9	84.3±7.4
MLP	30.7±11.5	36.0±3.7	51.5±13.6	69.6±9.7	75.2±9.3
MLP-FT	29.3±22.3	40.3±16.5	62.4±13.4	73.9±5.8	82.4±3.8
ScreenShot	**35.7±24.4**	**69.6±11.8**	**76.8±4.8**	**84.3±8.9**	**89.9±6.8**

I Confidence intervals and additional metrics

With only 3–5 random seeds, standard deviations provide a limited characterization of uncertainty, particularly for hit recall, which is a bounded proportion. We therefore report 95% confidence intervals for hit recall in Table 10. A naive Clopper–Pearson interval computed from counts pooled across seeds would treat repeated evaluations of the same true hits as independent observations, resulting in underestimated uncertainty. Instead, we use a cluster-robust estimator, with true hits as the clustering units. This approach retains all observations when estimating hit recall while accounting for dependence across repeated evaluations of the same hit in the standard error [95].

The resulting confidence intervals provide further evidence of ScreenShot's advantage. On BATCHIE and NCI-ALMANAC, its interval does not overlap that of the next-best method. On PDO-Breast, its interval overlaps only with that of MLP-FT, except for a small overlap with XGBoost and TabPFN at a budget of 300. On GDSC-SQ, consistent with Table 1, MLP-FT, XGBoost, and TabPFN outperform ScreenShot at budgets of 200 and 300.

Finally, Table 11 reports complementary regression (MAE, MSE, Pearson, and Spearman) and ranking (AUROC and NDCG) metrics on BATCHIE at n=300. ScreenShot achieves the best performance across all reported metrics.

Table 10: **Hit recall (%) with 95% cluster-robust confidence intervals.** Intervals cluster on the true hit, since the same hits recur across seeds; the point estimate pools all seeds and samples, so it can differ slightly from the seed-averaged values of Table 1. PIICM fails numerically on most NCI-ALMANAC samples and is omitted there; ComboLTR and PIICM were not run on PDO-Breast, a monotherapy screen. **Bold**: best per column within each dataset.

Method	n=100	n=200	n=300
BATCHIE			
ComboLTR	20.6 [18.8, 22.5]	21.9 [20.0, 23.7]	22.7 [20.8, 24.7]
PIICM	27.7 [24.6, 30.7]	25.4 [22.6, 28.3]	27.4 [24.5, 30.2]
XGBoost	41.3 [38.7, 44.0]	48.8 [46.0, 51.6]	52.4 [49.5, 55.3]
TabPFN	35.5 [32.8, 38.2]	43.0 [40.1, 45.9]	47.8 [44.9, 50.7]
MLP	53.0 [50.4, 55.7]	61.7 [59.2, 64.3]	67.6 [65.2, 70.0]
MLP-FT	60.8 [57.6, 64.0]	66.0 [62.9, 69.1]	68.1 [65.1, 71.1]
ScreenShot	**72.0 [68.9, 75.1]**	**74.7 [71.6, 77.7]**	**75.7 [72.7, 78.8]**
GDSC-SQ			
ComboLTR	30.6 [29.3, 31.9]	37.7 [36.3, 39.1]	41.0 [39.5, 42.4]
PIICM	29.3 [27.8, 30.8]	30.5 [29.0, 32.0]	32.9 [31.4, 34.4]
XGBoost	51.5 [49.4, 53.5]	61.0 [59.0, 63.1]	66.0 [63.9, 68.1]
TabPFN	49.3 [47.2, 51.4]	60.1 [57.8, 62.3]	**66.3 [64.1, 68.6]**
MLP	45.7 [43.6, 47.8]	55.3 [53.1, 57.5]	60.8 [58.7, 62.9]
MLP-FT	**54.8 [52.4, 57.3]**	**62.5 [60.1, 64.8]**	66.0 [63.7, 68.3]
ScreenShot	54.4 [51.9, 56.8]	58.2 [55.7, 60.7]	60.6 [58.1, 63.1]
NCI-ALMANAC			
ComboLTR	21.5 [20.9, 22.1]	24.0 [23.4, 24.7]	24.5 [23.9, 25.1]
XGBoost	33.0 [31.9, 34.1]	42.6 [41.4, 43.8]	50.0 [48.7, 51.2]
TabPFN	33.0 [31.9, 34.1]	41.1 [39.9, 42.4]	47.8 [46.5, 49.1]
MLP	32.2 [31.1, 33.2]	36.4 [35.2, 37.6]	40.6 [39.4, 41.8]
MLP-FT	40.6 [39.1, 42.2]	48.5 [47.0, 49.9]	49.5 [48.1, 51.0]
ScreenShot	**54.8 [53.2, 56.4]**	**61.0 [59.4, 62.6]**	**63.0 [61.4, 64.6]**
PDO-Breast			
XGBoost	70.1 [64.4, 75.9]	81.9 [76.6, 87.2]	89.9 [85.6, 94.1]
TabPFN	70.7 [63.1, 78.3]	83.7 [77.9, 89.6]	90.7 [86.7, 94.6]
MLP	57.3 [51.4, 63.3]	76.8 [71.2, 82.4]	84.0 [79.5, 88.5]
MLP-FT	86.4 [80.0, 92.8]	91.5 [86.7, 96.2]	93.3 [89.2, 97.5]
ScreenShot	**87.7 [81.9, 93.5]**	**93.6 [89.5, 97.7]**	**96.0 [92.3, 99.7]**

Table 11: **Additional regression and ranking metrics (BATCHIE, n=300).** Mean over 5 seeds. AUROC and NDCG treat observations with true viability below 0.5 as effective. **Bold**: best per column.

Method	MAE ↓	MSE ↓	Pearson ↑	Spearman ↑	AUROC ↑	NDCG ↑
ComboLTR	0.339	0.178	0.27	0.27	0.647	0.531
PIICM	0.308	0.127	0.21	0.20	0.610	0.428
XGBoost	0.171	0.060	0.72	0.71	0.887	0.919
TabPFN	0.199	0.067	0.69	0.67	0.864	0.908
MLP	0.174	0.082	0.66	0.62	0.816	0.921
MLP-FT	0.138	0.047	0.80	0.78	0.907	0.955
ScreenShot	**0.104**	**0.032**	**0.87**	**0.84**	**0.942**	**0.974**

J Active learning additional results

J.1 Hit recall across datasets

Table 12 reports top-hit recall for the four experimental design strategies across three datasets: random selection, k-medoids cold start (100% budget), and two adaptive strategies with 33% cold start followed by 1 or 2 delta-weighted k-means++ rounds. Adaptive methods consistently outperform random selection and cold-start baselines. On BATCHIE, adaptive selection with two rounds reaches 84.8% at budget 300, compared to 75.8% for random, a gain of 9 percentage points. On GDSC-SQ and NCI-ALMANAC, adaptive methods improve by 4–7 points over random.

Comparison with BATCHIE-AL. We compare against the Bayesian active learning platform of Tosh et al. [54], which we refer to as BATCHIE-AL to distinguish it from the BATCHIE dataset. To match our protocol, one BATCHIE-AL model is fit per target sample with the same total budget, including the initial batch: the budget is split into three equal plates, a randomly chosen initial plate followed by two adaptive rounds selected by the BATCHIE-AL acquisition function, after which the model predicts the full treatment grid (5 seeds). Hit recall is computed with the same metric and ground-truth controls as for ScreenShot. The adaptive strategy of ScreenShot outperforms BATCHIE-AL at every budget on all three datasets (Table 12), with the largest gap on NCI-ALMANAC, where ScreenShot at budget 100 (58.2%) already exceeds BATCHIE-AL at budget 300 (53.5%). MAE shows the same ordering (0.094, 0.124, and 0.127 for ScreenShot vs. 0.133, 0.147, and 0.155 for BATCHIE-AL on BATCHIE, GDSC-SQ, and NCI-ALMANAC at budget 300).

Table 12: **Active learning: hit recall (%).** Mean±std across 5 seeds. **Bold**: best per column. BATCHIE-AL: Bayesian active learning method of Tosh et al. [54], run per sample with the same total budget.

Method	BATCHIE			GDSC-SQ			NCI-ALMANAC		
	100	200	300	100	200	300	100	200	300
BATCHIE-AL	44.7±3.3	61.9±0.6	71.1±1.3	47.1±2.8	55.1±1.0	62.9±2.2	29.1±2.0	45.8±2.7	53.5±4.0
Random	71.6±1.6	74.4±1.7	75.8±1.3	56.3±2.0	60.6±1.8	63.0±1.2	49.4±2.0	54.0±0.6	56.6±0.6
Cold-start (k-medoids)	72.9±1.4	75.1±1.1	76.1±0.7	60.4±1.7	63.0±1.4	65.2±1.1	**60.3±0.8**	60.8±1.4	61.2±1.5
Adaptive (1 round)	77.3±1.4	81.2±0.8	84.0±0.7	**61.1±0.6**	**65.5±0.8**	67.0±1.6	57.6±2.9	**62.3±0.9**	62.9±1.0
Adaptive (2 rounds)	**77.5±0.9**	**83.0±2.6**	**84.8±1.9**	59.9±1.4	65.3±0.7	**67.4±1.0**	58.2±2.0	61.5±0.9	**63.6±1.1**

J.2 Effect of cold-start proportion and number of rounds

Table 13 examines how the split between cold-start and adaptive budget affects performance on the BATCHIE combination screen. The 33% cold-start split achieves the best hit recall (84.8% at budget 300), while allocating more to cold start (50%) gives the best MAE and Pearson correlation. This suggests a balance between initial coverage and adaptive exploitation. Two rounds consistently outperform one round, though the marginal gain is modest ($\sim$0.5–1 percentage points). All adaptive strategies substantially outperform pure k-medoids cold start (76.1%) in hit recall. However, pure k-medoids achieves the best MAE, which is consistent with the known effectiveness of k-medoids for reducing global prediction error through its diversity-representativity tradeoff [96], but demonstrates that minimizing global error does not directly translate to better top-hit detection.

Table 13: **Effect of cold-start proportion and number of adaptive rounds** on the BATCHIE combination screen. All adaptive methods use the proposed strategy (k-medoids cold start, k-means++ with delta weighting on final embeddings). Bold indicates best per column.

Method	**Hit Recall (%)**			**MAE**		
	100	200	300	100	200	300
k-medoids (100% cold start)	72.9±1.4	75.1±1.1	76.1±0.7	**0.107±0.001**	**0.097±0.001**	**0.093±0.001**
Adaptive (50% CS, 1 round)	75.3±1.4	78.9±1.6	82.2±1.3	0.109±0.001	0.099±0.001	**0.093±0.001**
Adaptive (50% CS, 2 rounds)	75.8±0.8	80.5±1.4	82.6±1.2	0.109±0.001	0.098±0.001	**0.093±0.001**
Adaptive (33% CS, 1 round)	77.3±1.4	81.2±0.8	84.0±0.7	0.110±0.002	0.100±0.001	0.095±0.001
Adaptive (33% CS, 2 rounds)	**77.5±0.9**	**83.0±2.6**	**84.8±1.9**	0.110±0.001	0.100±0.001	0.094±0.001
Adaptive (20% CS, 1 round)	74.7±0.9	79.1±1.0	83.3±1.7	0.112±0.001	0.104±0.001	0.099±0.001
Adaptive (20% CS, 2 rounds)	75.5±1.9	80.0±1.4	83.8±2.0	0.112±0.001	0.104±0.001	0.098±0.001

K Architecture ablation

Table 14 compares the full ScreenShot architecture against three ablated variants on the BATCHIE and PDO-Breast datasets. *No Level 3* removes the sample-level self-attention that aggregates across perturbations. *Flat attention* replaces the hierarchical encoder with a single self-attention layer over all flattened context observations followed by cross-attention. *No Level 1 & 3* removes both the drug-combination encoder and the sample-level aggregation; drug and dose embeddings are combined via element-wise product and averaged across drugs in the combination (as in the MLP baseline), producing a single vector per observation that is passed directly to the cross-attention response encoder.

The full architecture achieves the best hit recall on 9 out of 10 dataset/budget combinations. *No Level 1 & 3*, which reduces the model to the cross-attention level, drops significantly on BATCHIE (54.7% vs. 76.0% at budget 300) and saturates on PDO-Breast ($\sim$78% regardless of budget), indicating that a single encoder level lacks the capacity to capture the structure of screening data. *Flat attention* and *No Level 3* perform closer to the full model but consistently trail on BATCHIE, where the combinatorial treatment space is larger. Overall, each level contributes to prediction quality at the cost of a reasonable increase in computation (Figure 7).

Table 14: Architecture ablation: top-hit recall (%) across few-shot budgets on BATCHIE and PDO-Breast. Values: mean±std across 5 seeds. **Bold**: best per column.

	BATCHIE					PDO-BREAST				
	10	50	100	200	300	10	50	100	200	300
ScreenShot	**63.9±3.1**	**69.0±2.4**	**72.6±3.0**	**75.0±2.5**	**76.0±1.4**	**78.7±2.5**	81.9±2.2	**87.7±2.4**	**93.6±1.7**	**96.0±1.6**
Flat attention	60.3±2.2	65.6±1.6	68.6±0.6	70.9±0.9	72.3±1.3	76.3±6.5	82.9±3.0	86.9±3.2	91.5±2.6	94.7±1.6
No Level 3	59.5±2.0	66.9±1.2	70.9±1.3	74.5±1.1	75.6±1.0	75.5±3.8	**83.5±1.8**	87.7±3.2	92.0±2.1	95.2±3.1
No Level 1 & 3	51.0±3.4	53.2±1.5	55.0±1.5	54.4±0.4	54.7±0.9	77.1±1.7	78.4±2.0	77.9±1.2	78.1±1.5	78.4±1.1

Figure 7 shows inference time as a function of context size on both PDO-Breast and BATCHIE. On PDO-Breast, ScreenShot scales from 3.9 s at 100 context observations to 5.0 s at 3,000, because the dominant self-attention at Level 3 operates over perturbations ($p \ll n$) rather than individual observations. The flat attention variant exhibits quadratic scaling, reaching 28.6 s at 3,000 observations (5.7$\times$ slower). On BATCHIE, flat attention reaches 155 s at 500 context observations (2.7$\times$ slower than ScreenShot at the same budget). This confirms the computational motivation for the hierarchical design discussed in Section 2.2.

Embedding dimension. We use an embedding dimension of D=256, which provides a balance between model capacity and inference efficiency. To assess the sensitivity of performance to this choice, we retrain ScreenShot with D=16, keeping all other settings unchanged, and evaluate both models under the same BATCHIE few-shot protocol (Table 15).

Reducing the embedding dimension substantially degrades performance and limits the model's ability to benefit from additional context. In particular, with D=16, performance quickly plateaus as the context budget increases: MAE improves only marginally, from 0.173 to 0.163, while hit recall

remains approximately 57%. In contrast, the D=256 model continues to improve with increasing context budget. These results suggest that sufficient embedding capacity is necessary for ScreenShot to effectively incorporate additional few-shot observations.

Table 15: **Effect of the embedding dimension D (BATCHIE).** MAE and hit recall (%) of ScreenShot retrained with D=16 instead of D=256, all else identical; mean over 5 seeds.

	MAE ↓		Hit recall (%) ↑	
Budget	D=256	D=16	D=256	D=16
10	**0.153**	0.173	**63.9**	55.3
100	**0.117**	0.162	**72.6**	56.8
300	**0.101**	0.163	**76.0**	57.1

Combination size. BATCHIE includes single agents, drug pairs, and triplets, allowing us to evaluate ScreenShot on combinations of up to three drugs within the same experimental setting. To isolate the effect of combination size q on inference cost, we measure runtime while keeping the number of samples, context budget, and query count fixed, and varying only q (Table 16).

The self-attention operation over the q drug tokens at Level 1 has quadratic complexity, $O(q^2)$. However, its computational cost remains limited for the small combination sizes considered here ($q \in \{1, 2, 3\}$). Moreover, the cross-attention operation over the observed context does not introduce an additional dependence on q in its context dimension. Empirically, increasing the combination size from one to three drugs increases per-query inference time by approximately a factor of three, indicating near-linear scaling in this range.

Table 16: **Inference time vs. combination size.** Per-query CPU inference time with sample count, context budget, and number of queries fixed, varying only the number of drugs q per combination (mean±std over 5 repeats).

Combination size	Time per query (ms)	Relative to 1 drug
1 drug	0.626±0.005	1.00
2 drugs	1.587±0.009	2.54±0.02
3 drugs	1.959±0.013	3.13±0.03

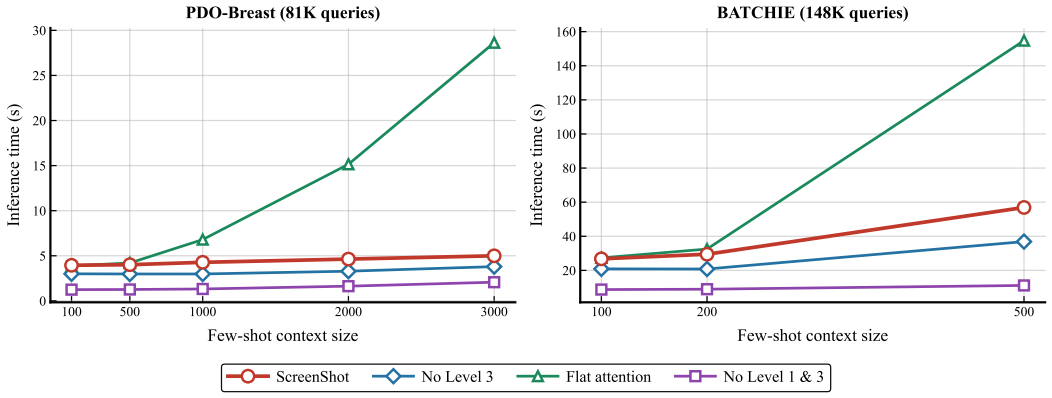

Figure 7: Inference time vs. context size on PDO-Breast (left) and BATCHIE (right), measured on NVIDIA H100 GPU.

L Active learning ablation

We ablate each component of the adaptive experimental design strategy described in Section 4. The *proposed* method uses k-medoids on drug-dose embeddings for cold start (33% of budget),

followed by k-means++ on final model embeddings with delta weighting adaptive rounds. Each ablation modifies exactly one component. We evaluate on three datasets (BATCHIE, GDSC2, NCI-ALMANAC) with budgets of 100, 200, and 300 treatments per sample, averaged over 5 seeds. The "Uncertainty weighting" ablation uses MC dropout uncertainty instead of predicted delta as selection weights. Uncertainty is computed by enabling dropout (applied to the Level 3 Sample Encoder only) and running 10 stochastic forward passes. The prediction is the mean across passes; uncertainty is the standard deviation.

Table 17 reports hit recall for each variant. Table 18 reports MAE and Pearson correlation, which measure overall prediction quality rather than hit detection.

Table 17: **Active learning ablation: Hit Recall@20%.** The first row reports absolute hit recall (%); all other rows report the difference in percentage points relative to the proposed method. Each row removes or replaces one component. All methods use 33% cold start budget. Blue : significantly better than proposed; red : significantly worse (Wilcoxon signed-rank test). Shading intensity: light $p < 0.05$, medium $p < 0.01$, dark $p < 0.001$.

Method	BATCHIE			GDSC2			NCI-ALMANAC		
	100	200	300	100	200	300	100	200	300
1 adaptive round									
Proposed	77.3±1.4	81.2±0.8	84.0±0.7	61.1±0.6	65.5±0.8	67.0±1.6	57.6±2.9	62.3±0.9	62.9±1.0
No delta weighting	-4.3±1.0	-4.9±0.9	-5.8±1.3	-4.0±1.0	-4.3±0.6	-3.0±0.6	-0.7±0.8	-2.0±0.8	-1.8±0.5
No k-means++	-3.2±0.6	-3.1±0.6	-4.3±0.8	+0.5±0.6	-0.6±0.5	-0.8±0.5	+0.8±0.7	-0.3±0.5	+0.1±0.5
No final embeddings	-2.8±0.7	-0.4±0.5	-1.3±0.6	+0.1±0.9	-0.8±0.7	+0.2±0.6	+1.3±0.7	-0.9±0.5	-0.1±0.4
k-means++ cold start	-1.7±0.9	+0.1±1.0	+0.2±0.7	-0.3±0.9	-0.6±0.8	+0.2±0.7	-3.1±1.0	-1.6±0.7	+0.0±0.6
Random cold start	-0.4±0.9	-0.0±0.9	+0.0±1.0	-1.4±0.8	-1.2±0.9	+0.0±0.7	-4.1±1.0	-4.7±1.1	-1.7±0.7
Uncertainty weighting	-4.3±0.9	-4.3±0.9	-5.4±0.9	-1.8±0.6	-2.5±0.7	-2.0±0.6	+0.8±1.0	-0.8±0.7	-0.5±0.5
Top-k delta	-5.8±0.8	-3.4±0.5	-3.4±0.6	-10.0±0.9	-8.1±0.8	-7.0±0.5	-7.4±1.1	-7.7±1.4	-11.8±1.4
Top-k uncertainty	-8.1±1.3	-5.6±0.9	-6.0±0.9	-6.6±1.0	-6.4±0.8	-5.7±0.8	-6.4±1.0	-7.4±1.6	-9.6±1.7
2 adaptive rounds									
Proposed	77.5±0.9	83.0±2.6	84.8±1.9	59.9±1.4	65.3±0.7	67.4±1.0	58.2±2.0	61.5±0.9	63.6±1.1
No delta weighting	-5.2±1.0	-7.0±1.2	-5.7±1.2	-2.9±0.8	-4.4±0.6	-3.2±0.7	-3.0±0.8	-1.1±0.7	-2.4±0.5
No k-means++	-2.6±0.9	-3.9±0.8	-4.0±0.7	+0.9±0.7	-0.7±0.8	-0.2±0.6	-0.0±0.7	-0.2±0.5	-0.2±0.4
No final embeddings	-3.9±1.0	-3.1±0.9	-1.8±0.7	+2.3±0.9	+0.9±0.7	+0.7±0.6	+0.4±0.6	+0.1±0.6	-0.1±0.5
k-means++ cold start	-0.1±0.9	-0.9±1.1	-0.2±0.9	-0.1±1.0	-0.1±0.8	-0.4±0.6	-3.8±1.0	-0.7±0.8	-0.8±0.6
Random cold start	-0.4±0.7	-1.2±0.9	+0.4±1.0	-1.5±1.2	-0.8±0.8	-0.7±0.4	-3.8±1.0	-1.7±0.8	-1.5±0.6
Uncertainty weighting	-3.1±0.9	-5.6±1.0	-6.2±0.9	-1.9±0.8	-3.8±0.7	-2.4±0.8	-0.5±0.8	-0.7±0.5	-0.9±0.5
Top-k delta	-6.1±1.2	-5.6±0.8	-4.6±0.8	-6.1±1.1	-6.7±0.8	-6.0±0.8	-6.6±0.8	-7.3±1.6	-10.9±1.9
Top-k uncertainty	-8.5±1.4	-7.8±1.0	-8.0±1.1	-4.3±0.9	-5.4±0.8	-5.7±0.8	-5.0±0.7	-4.2±0.8	-4.3±0.8

Delta weighting is the most important component. Removing delta weighting ("No delta weighting") consistently degrades hit recall across all datasets and budgets, with losses of 2–6 percentage points on BATCHIE and 2–3 on GDSC2. This confirms that prioritizing treatments with large predicted differential sensitivity is essential for hit detection.

Diversity is essential. Pure exploitation strategies (Top-k delta, Top-k uncertainty) that select the most promising treatments without diversity constraints perform substantially worse than their diversity-aware counterparts across all metrics. On NCI-ALMANAC, Top-k delta achieves only 51.1% hit recall at budget 300, compared to 62.9% for the proposed method, a loss of nearly 12 points. This demonstrates that greedy selection leads to redundant observations that fail to improve the model's global predictive accuracy.

k-means++ outperforms k-medoids for adaptive selection. Replacing k-means++ with k-medoids in the adaptive rounds ("No k-means++") reduces hit recall by up to 4 points on BATCHIE, though the difference is smaller on GDSC2 and NCI-ALMANAC.

Uncertainty weighting improves prediction but not hit detection. Weighting by MC dropout uncertainty instead of predicted delta ("Uncertainty weighting") achieves the best MAE and Pearson correlation on all three datasets, indicating that uncertainty-guided selection improves overall predictive accuracy. However, it underperforms delta weighting for hit recall by 4–6 points on BATCHIE, consistent with the observation that uncertainty targets regions of model ignorance rather than regions of differential drug sensitivity.

Table 18: **Active learning ablation: MAE and Pearson correlation.** Same ablation as Table 17. Bold indicates best per column (lowest MAE, highest Pearson).

Method	BATCHIE			GDSC2			NCI-ALMANAC		
	100	200	300	100	200	300	100	200	300
MAE — 1 adaptive round									
Proposed	0.110±0.002	0.100±0.001	0.095±0.001	0.132±0.001	0.126±0.001	0.125±0.001	0.144±0.003	0.134±0.001	0.131±0.001
No delta weighting	0.111±0.001	0.100±0.001	0.095±0.001	0.132±0.001	0.126±0.001	0.123±0.001	0.147±0.003	0.137±0.001	0.132±0.001
No k-means++	0.112±0.001	0.102±0.001	0.096±0.000	**0.131±0.001**	0.127±0.000	0.126±0.001	0.144±0.002	0.136±0.002	0.132±0.001
No final embeddings	0.113±0.001	0.102±0.001	0.097±0.000	0.133±0.001	0.130±0.001	0.128±0.001	**0.142±0.002**	0.136±0.001	0.132±0.001
k-means++ cold start	0.113±0.001	0.102±0.001	0.097±0.000	0.133±0.002	0.128±0.001	0.125±0.001	0.145±0.002	0.137±0.001	0.132±0.001
Random cold start	0.113±0.002	0.105±0.001	0.100±0.001	0.135±0.001	0.130±0.001	0.126±0.001	0.146±0.001	0.138±0.001	0.133±0.001
Uncertainty weighting	**0.108±0.000**	**0.097±0.000**	**0.092±0.001**	0.131±0.001	**0.125±0.001**	**0.122±0.001**	0.142±0.001	**0.132±0.001**	**0.128±0.001**
Top-k delta	0.124±0.001	0.111±0.001	0.104±0.001	0.144±0.003	0.138±0.001	0.137±0.000	0.165±0.005	0.152±0.004	0.156±0.003
Top-k uncertainty	0.121±0.001	0.108±0.001	0.102±0.000	0.141±0.003	0.133±0.002	0.129±0.001	0.157±0.002	0.148±0.003	0.146±0.002
MAE — 2 adaptive rounds									
Proposed	0.110±0.001	0.100±0.001	0.094±0.001	0.132±0.002	0.126±0.001	0.124±0.001	0.143±0.002	0.135±0.001	0.130±0.002
No delta weighting	0.112±0.000	0.100±0.001	0.095±0.001	0.132±0.001	0.127±0.001	0.123±0.001	0.149±0.001	0.138±0.001	0.133±0.001
No k-means++	0.111±0.001	0.101±0.001	0.096±0.001	**0.130±0.001**	0.126±0.001	0.124±0.000	0.144±0.002	0.136±0.001	0.132±0.002
No final embeddings	0.113±0.001	0.101±0.001	0.096±0.000	0.132±0.001	0.127±0.001	0.127±0.001	**0.142±0.001**	0.136±0.001	0.131±0.001
k-means++ cold start	0.112±0.001	0.102±0.001	0.096±0.001	0.133±0.002	0.128±0.001	0.126±0.001	0.146±0.002	0.137±0.002	0.132±0.001
Random cold start	0.112±0.001	0.104±0.001	0.099±0.001	0.136±0.002	0.130±0.001	0.126±0.002	0.145±0.002	0.137±0.001	0.133±0.001
Uncertainty weighting	**0.108±0.000**	**0.097±0.000**	**0.091±0.000**	0.131±0.000	**0.125±0.001**	**0.122±0.000**	0.143±0.001	**0.133±0.001**	**0.128±0.001**
Top-k delta	0.122±0.001	0.110±0.001	0.103±0.001	0.141±0.002	0.136±0.002	0.136±0.000	0.164±0.005	0.153±0.005	0.155±0.006
Top-k uncertainty	0.121±0.002	0.107±0.001	0.100±0.001	0.136±0.001	0.129±0.001	0.126±0.001	0.153±0.002	0.145±0.001	0.139±0.002
Pearson — 1 adaptive round									
Proposed	0.844±0.004	0.866±0.003	0.875±0.002	0.700±0.003	0.723±0.004	0.730±0.002	0.673±0.015	0.714±0.005	0.727±0.006
No delta weighting	0.844±0.003	0.867±0.002	0.878±0.002	0.686±0.006	0.712±0.005	0.725±0.003	0.662±0.010	0.703±0.004	0.721±0.003
No k-means++	0.839±0.003	0.861±0.003	0.872±0.002	**0.701±0.002**	0.722±0.003	0.727±0.003	0.676±0.014	0.710±0.006	0.725±0.006
No final embeddings	0.838±0.004	0.861±0.003	0.872±0.001	0.696±0.004	0.715±0.003	0.725±0.002	0.676±0.011	0.711±0.006	0.725±0.006
k-means++ cold start	0.842±0.003	0.861±0.003	0.872±0.001	0.696±0.008	0.717±0.006	0.728±0.002	0.669±0.011	0.706±0.004	0.721±0.006
Random cold start	0.842±0.004	0.858±0.003	0.866±0.003	0.682±0.008	0.709±0.004	0.721±0.004	0.665±0.010	0.695±0.010	0.718±0.009
Uncertainty weighting	**0.850±0.001**	**0.873±0.001**	**0.884±0.001**	0.698±0.003	**0.723±0.002**	**0.734±0.001**	**0.681±0.004**	**0.720±0.002**	**0.734±0.004**
Top-k delta	0.816±0.002	0.844±0.001	0.858±0.001	0.654±0.005	0.689±0.004	0.698±0.002	0.598±0.013	0.654±0.003	0.666±0.006
Top-k uncertainty	0.822±0.003	0.852±0.002	0.867±0.002	0.667±0.006	0.700±0.005	0.713±0.004	0.628±0.006	0.661±0.012	0.675±0.012
Pearson — 2 adaptive rounds									
Proposed	0.848±0.004	0.867±0.003	0.877±0.002	0.701±0.006	0.724±0.003	0.732±0.001	0.679±0.012	0.714±0.004	0.731±0.009
No delta weighting	0.843±0.002	0.866±0.002	0.878±0.003	0.691±0.006	0.710±0.002	0.727±0.003	0.653±0.008	0.698±0.007	0.720±0.005
No k-means++	0.841±0.004	0.864±0.003	0.873±0.002	**0.705±0.002**	**0.724±0.001**	0.731±0.002	0.677±0.008	0.715±0.004	0.729±0.006
No final embeddings	0.839±0.005	0.863±0.003	0.872±0.002	0.701±0.004	0.722±0.003	0.729±0.003	**0.683±0.012**	0.711±0.005	0.729±0.007
k-means++ cold start	0.844±0.005	0.862±0.003	0.874±0.003	0.697±0.005	0.717±0.005	0.728±0.004	0.668±0.011	0.707±0.006	0.725±0.005
Random cold start	0.845±0.004	0.859±0.004	0.867±0.003	0.682±0.009	0.708±0.004	0.723±0.004	0.670±0.011	0.702±0.009	0.720±0.008
Uncertainty weighting	**0.851±0.001**	**0.873±0.001**	**0.887±0.001**	0.697±0.003	0.721±0.005	**0.733±0.003**	0.679±0.005	**0.716±0.005**	**0.737±0.004**
Top-k delta	0.821±0.003	0.848±0.003	0.860±0.002	0.666±0.003	0.695±0.004	0.701±0.002	0.606±0.016	0.657±0.003	0.672±0.012
Top-k uncertainty	0.823±0.004	0.854±0.002	0.870±0.001	0.678±0.008	0.709±0.005	0.718±0.002	0.643±0.008	0.674±0.006	0.699±0.007

Cold-start method matters less when followed by adaptive rounds. When only 33% of the budget is allocated to the initial batch and adaptive rounds follow, the cold-start method (k-medoids, k-means++, or random) has limited impact on the final hit recall, as the adaptive rounds compensate for suboptimal initial coverage. However, when the full budget is allocated to the initial batch (i.e., no adaptive rounds), k-medoids on drug-dose embeddings substantially outperforms random selection (Table 12).

Embedding choice for adaptive rounds. Using drug-dose embeddings instead of final model embeddings for adaptive selection ("No final embeddings") yields mixed results: slightly worse on BATCHIE but comparable or better on GDSC-SQ and NCI-ALMANAC, suggesting that the benefit of response-conditioned embeddings is dataset-dependent.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract claims that ScreenShot outperforms baselines in prediction accuracy and hit detection (supported by Table 1 and Figure 2) and that the active learning strategy reduces the budget by a factor of three (supported by Figure 3C). All claims are backed by experiments on four held-out datasets.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 5 discusses drug coverage and UNK token handling, the reliance on control samples for hit detection, and the trade-off between prediction accuracy and hit detection in active learning. Section C provides quantitative analysis of performance on unknown drugs.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: This paper is an empirical contribution and does not include theoretical results.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The model architecture is described in Section 2.2 with a detailed diagram in Section A. The pretraining procedure, data sampling, and UNK token training are described in Section 2.3 and Section B. All hyperparameters for baselines are in Section E.2. The detailed evaluation protocol, sample selection, and control definitions are in Section D. The active learning strategy is detailed in Section 2.5 and Section D.3. Source code is provided as supplemental material.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

Justification: A code repository is provided at `https://github.com/tansey-lab/screenshot`, containing the ScreenShot source code, pretrained weights, preprocessed evaluation datasets, an interactive dashboard, and scripts to reproduce all experimental results. The cell line pretraining datasets (CCLE, GDSC, CTRP) are publicly available; the ex vivo pretraining datasets can be obtained upon request to the authors of Pichotta et al. [21].

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The evaluation datasets, sample selection, and control samples are listed in Section 4.1 and Section D. The optimizer and pretraining details are in Section 2.3 and Section B. All baseline hyperparameters are in Section E.2. The leave-one-out pretraining setup is described in Section B.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: All few-shot experiments report mean $\pm$ standard deviation. The active learning ablation tables include Wilcoxon signed-rank tests with color-coded significance levels ($p < 0.05$, 0.01, 0.001). The source of variability (random selection of few-shot context) is stated in Section D. Hit recall is additionally reported with 95% cluster-robust confidence intervals in Section I.

   Guidelines:

   - The answer NA means that the paper does not include experiments.

- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Pretraining was performed on a single NVIDIA H100 GPU for ∼400K iterations (Section 4.1). Few-shot and active learning experiments ran on NVIDIA A100 80GB GPUs; XGBoost ran on 8 CPU cores. Per-experiment wall-clock times are reported in Table 1 and Section F.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This work does not involve human subjects. All data are de-identified viability measurements with no personally identifiable information.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The positive impact is enabling more efficient personalized cancer treatment by reducing the number of experiments needed. The model is specific to drug screening viability prediction and we do not foresee negative societal impacts from this application.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The model predicts drug screening viability values and does not pose risks for misuse.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All 40 pretraining datasets are cited with their original publications in Section B. The ex vivo datasets were obtained from Pichotta et al. [21] with permission. Baseline implementations (XGBoost, TabPFN, scikit-learn) are cited.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The ScreenShot model, pretrained weights, experiment code, and an interactive dashboard are released at `https://github.com/tansey-lab/screenshot`. The architecture is documented in Section A, training in Section B, and usage instructions are provided in the repository README.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This work does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: This work does not involve research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are not used as a component of the core methodology.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.