nature communications

Article

https://doi.org/10.1038/s41467-025-59822-y

Building a unified model for drug synergy analysis powered by large language models

Received: 18 November 2024

Accepted: 5 May 2025

Published online: 15 May 2025

Check for updates

Tianyu Liu[1,2], Tinyi Chu[2], Xiao Luo[3] & Hongyu Zhao[1,2]

Drug synergy prediction is a challenging and important task in the treatment of complex diseases including cancer. In this manuscript, we present a unified Model, known as BAITSAO, for tasks related to drug synergy prediction with a unified pipeline to handle different datasets. We construct the training datasets for BAITSAO based on the context-enriched embeddings from Large Language Models for the initial representation of drugs and cell lines. After demonstrating the relevance of these embeddings, we pre-train BAITSAO with a large-scale drug synergy database under a multi-task learning framework with rigorous selections of tasks. We demonstrate the superiority of the model architecture and the pre-trained strategies of BAITSAO over other methods through comprehensive benchmark analysis. Moreover, we investigate the sensitivity of BAITSAO and illustrate its promising functions including drug discoveries, drug combinations-gene interaction, and multi-drug synergy predictions.

Treating patients with a combination of drugs has become common for various diseases, including HIV[1] and cancers[2,3]. One key aspect of drug combinations is the synergistic effect, which means that the joint effect of multiple drugs is larger than the sum of individual drug effects[4]. Other definitions have also been used to define synergistic effects, such as ref. 5. Effective drug combination can reduce the drug resistance of monotherapy[6] with relatively lower doses of individual drugs[7]. Since drugs can change gene expressions when applied to different systems, e.g., cell lines, their effects can be studied through the genomics lens[8,9]. Currently, researchers use high-throughput combinatorial screening to identify drug combinations with synergistic effects for specific cell lines[10]. However, such experimental screening is laborious and time-consuming due to the very large number of potential drug combinations, and it is even more challenging to assess the synergistic effect of combinations with three or more drugs[11]. Therefore, it is important to develop computational methods based on extensive experimental datasets in the public domain as well as diverse types of prior biological knowledge to predict the presence and strength of synergistic effects for candidate drug combinations. Accurate prediction methods can facilitate drug discovery[12] and clinical development[13].

Given its importance, it is no surprise that many machine learning methods, especially deep learning methods, have been proposed to predict drug synergy. These methods differ in model architecture, training strategies, and datasets used to build the models. DeepSynergy[14] is among the earliest tools by building a neural network for both regression and classification, with follow-up work such as TreeComb[15,16] and MatchMarker[17]. Existing drug synergy prediction methods can be broadly classified into two groups. The first group of methods, such as MARSY[18], focuses on predicting specific synergy scores, whereas the second group of methods, such as DeepDDs[19], transfers the continuous synergy score into a binary one via thresholding to infer drug combination synergy. However, most existing methods do not incorporate the extensive synergy information from public databases[20] in their predictions.[21] utilized a transfer learning approach and pre-trained the model based on large-scale databases while incorporating different types of features (e.g., gene expression, molecular structure). However, it did not consider datasets[14] with only partial information and treated drugs with the same molecular formula but different names as distinct ones. Therefore, the generalization ability of this model is limited by its input data format. Moreover, since public databases are updated constantly, it is important to track the versions of training datasets.

[1]Interdepartmental Program in Computational Biology & Bioinformatics, Yale University, New Haven, CT, USA. [2]Department of Biostatistics, Yale University, New Haven, CT, USA. [3]Department of Computer Science, University of California, Los Angeles, Los Angeles, CA, USA. e-mail: hongyu.zhao@yale.edu

Large Language Models, as a type of Foundation Models (FMs)[22], have greatly improved the performance of deep learning on various tasks in Natural Language Processing (NLP)[23]. Such models have received broad attention from both industry and academia[24]. Researchers have proposed to use FMs to predict drug synergy via LLMs by transferring the drug synergy prediction problem into a Question-Answer problem[25,26]. By incorporating prior information of single drugs and single cell lines from LLMs, it has become possible to predict drug synergy of unknown drug combinations in unknown cell lines. Text information may be less noisy than the features (e.g., gene expression levels) that have been used in this task. However, such a QA setting limits the task to a classification problem, which introduces the potential bias of pre-defined thresholds. Moreover, these two LLMs are not open-source, so it is difficult for researchers to evaluate their performance. Open-source is important for the development of science[27]. Moreover, there is a lack of exploration on the utilization of the information in LLMs for more difficult drug synergy prediction problems, e.g., the effects of multiple drug combinations or model explainability.

Here we present a scalable unified model for drug synergy prediction called BAITSAO. BAITSAO utilizes the information from LLMs as input and was pre-trained based on large-scale known synergistic effect information of paired drug combinations and cell lines. The information on drug combinations and cell lines is necessary to predict synergy scores. We show that the embeddings of these features from LLMs can be effective input for drug synergy prediction, as well as the effects of drugs on gene expression. We further demonstrate the capability of building an effective predictor for synergy prediction under both the classification and regression settings through multi-task learning (MTL)[28]. Finally, we pre-train BAITSAO to predict synergistic effects for unseen drug combinations based on the zero-shot learning framework and the fine-tuning framework. The scalability of BAITSAO allows us to consider multiple drugs and incorporate extra meta information.

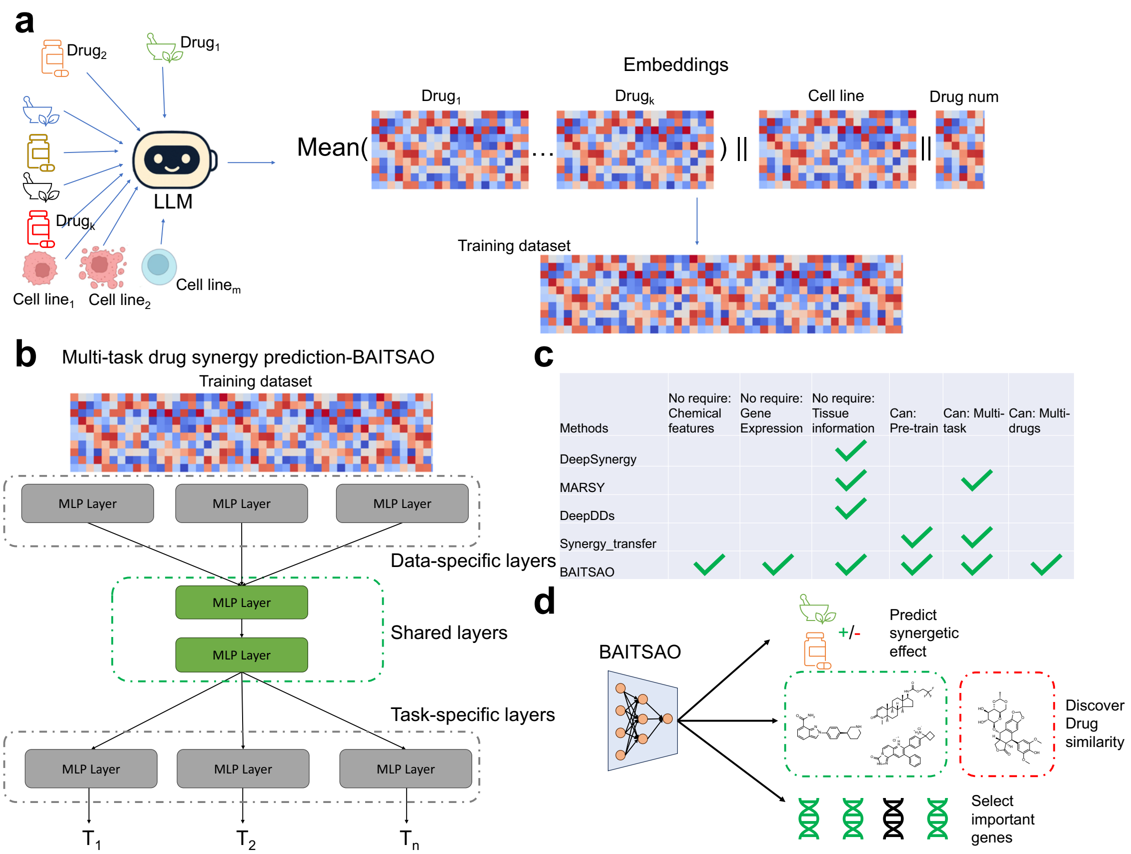
Results

Overview of BAITSAO

We highlight two major contributions of BAITSAO as a unified model. We first provide a unified pipeline for pre-processing the information from both drugs and cell lines for machine learning in a tabular format, and generate training datasets from these embeddings for multiple tasks. We show that these embeddings contain functional information for prediction. We then utilize the unified training datasets for different synergistic effect prediction tasks under the multi-task learning framework. We demonstrate the superiority of the model architecture and the contribution of pre-training through comprehensive experiments. BAITSAO can be easily transferred to perform practical downstream tasks related to drug synergy analysis. We illustrate the landscape of BAITSAO in Fig. 1a and b and summarize the differences between BAITSAO and other synergy prediction methods in Fig. 1c. The major functions of BAITSAO are shown in Fig. 1d.

Drug embeddings from LLMs reflect functional similarity and responses at the cell level

In this section, we discuss the information offered by drug embeddings and cell-line embeddings. We generate the descrip-



Methods	No require: Chemical features	No require: Gene Expression	No require: Tissue information	Can: Pre-train	Can: Multi-task	Can: Multi-drugs
DeepSynergy			✓			
MARSY			✓		✓	
DeepDDs			✓			
Synergy_transfer				✓	✓	
BAITSAO	✓	✓	✓	✓	✓	✓

Fig. 1 | An overview of BAITSAO as an FM under the pre-training and fine-tuning/zero-shot learning pipeline. a The pre-processing steps we used to transfer the meta information into embeddings to construct training datasets. **b** The model architecture of BAITSAO under a multi-task learning framework. **c** Comparisons of different methods for drug synergy analysis. **d** Different functions of BAITSAO. Logos of cell lines are created with BioRender.com.

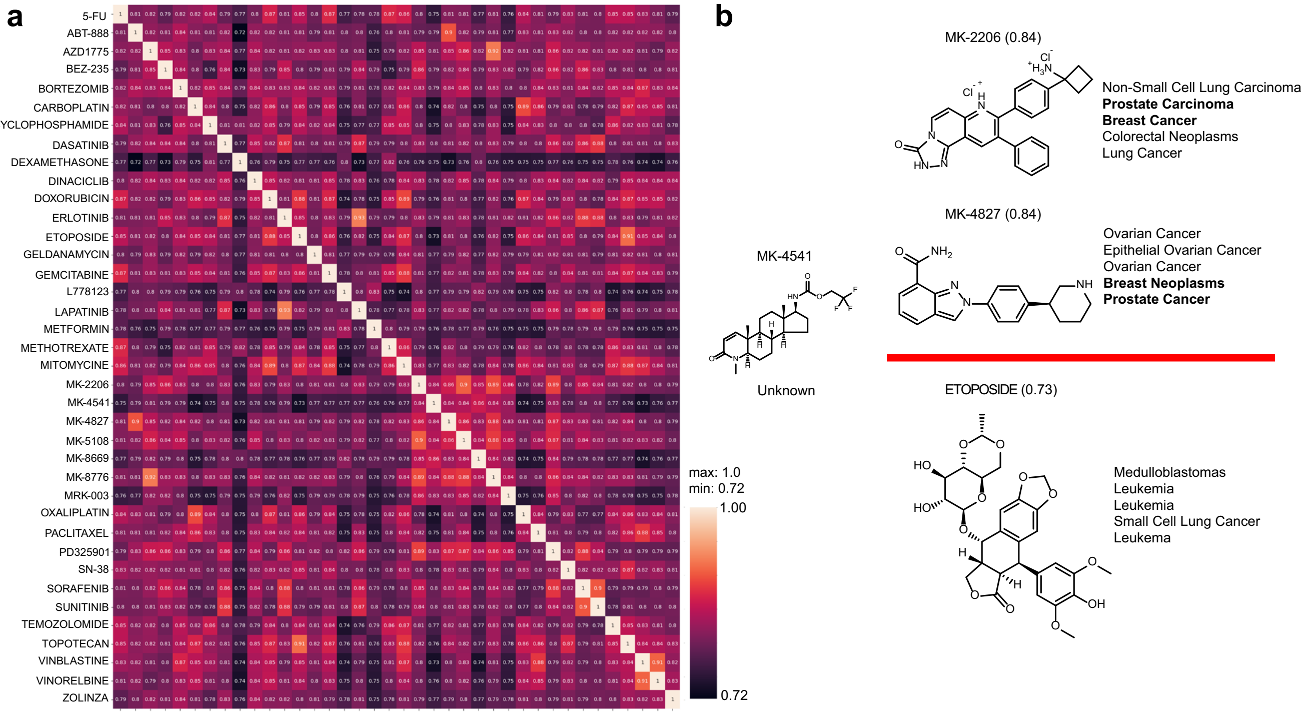


Fig. 2 | Investigation of drug embeddings. a The heatmap for the similarity of embeddings across all the drugs. **b** Exploration of drug similarity related to MK-4541. The drugs above the red line represent the two most similar drugs, while the drugs below the red line represent the most different drugs. We list five types of clinical trial information ranked by the phases. Source data are provided as a Source Data file.

tion for the drugs and cell lines from our training datasets based on designed prompts from LLMs, and then use the embedding module from GPT-3.5[29] to generate the embeddings of such descriptions, where the embeddings become the features of drugs or cell lines. We utilized GPT-3.5 rather than GPT 4[30] because the layer for generating embeddings is from the GPT-3[29] series, and the querying time from GPT 4 with similar quality required much more time[31], and efficiency is very important in LLM deployment[32,33]. Moreover, the performance difference between embeddings from GPT-3.5 and GPT 4 or from GPT-3.5 and Claude 3.5[34] is not significant based on our experiments, shown in Supplementary Fig. 1a (Wilcoxon rank-sum test, *p*-value = 0.86 for GPT-3.5 vs. GPT 4, and *p*-value = 0.44 for GPT-3.5 vs. Claude 3.5). In the same figure, we also found that embeddings from GPT-3.5 are better than embeddings from Gemini[35] (*p*-value = 0.0039), and thus our current selection is well-designed. We visualize the drug embeddings and the cell-line embeddings based on Uniform Manifold Approximation and Projection (UMAP)[36] shown in Supplementary Fig. 2a, b. We investigated the quality of the embeddings by considering both the quality of the description and the quality of the functions of the embeddings.

For the first aspect, we recorded the outputs as descriptions from GPT-3.5 based on our prompts and compared the content with information from DrugBank[37] and NCBI[38]. Here we used drugs and cell lines from DeepSynergy, which contains 39 drugs and 38 cell lines. The descriptions summarized the functional information of drugs and cell lines. Based on our experiments, only one drug (MK-8669) has a mismatched generated description, while 13 drugs cannot be matched with the indication information if we search them in DrugBank. All descriptions are included in Supplementary Data 1. We plot the Cosine Similarity (CS) for all drugs' embeddings in Fig. 2a. We also randomly selected 10 drugs from this dataset and plot the CS for the embeddings of the same drug under 10 different descriptions by running GPT-3.5 multiple times in Supplementary Fig. 3. These two figures show that the similarity from different drugs is generally lower than that from the same drug, suggesting that we can get informative embeddings from LLMs.

To perform a comprehensive analysis of our generated drug embeddings from LLMs, we downloaded the descriptions of drugs, including indication, summary, and background, from the DrugBank. We embedded these descriptions based on the same GPT-3.5 embeddings layer and computed the CS between embeddings from DrugBank descriptions and the LLM-generated descriptions. We found that embeddings from LLMs have a strong average similarity with all three descriptions from DrugBank (CS = 0.87 for indication, CS = 0.90 for summary, and CS = 0.90 for background), and thus, the generated drug embeddings preserved the important functional and chemical properties of the original drug. Furthermore, we visualize the CS based on the embeddings from drug indication (Supplementary Fig. 4a), drug summary (Supplementary Fig. 4b), and drug background (Supplementary Fig. 4c). We further computed the Pearson Correlation Coefficient (PCC) between the similarity matrix from DrugBank descriptions and LLM descriptions, which could be used to evaluate the ability of embeddings used by BAITSAO in preserving the drug-drug similarity. The PCCs are annotated under each figure, and all of the PCCs are high (PCC ≥ 0.76) and significant (*p*-value < 0.05). Therefore, we demonstrated the ability of LLMs to generate meaningful descriptions as well as embeddings by comparing the generated information with a known database, and further enhanced the reliability of the pipeline.

Furthermore, we performed a Mann−Whitney U test[39] to compare the PCCs among the drugs from the MK class and the PCCs between the drugs from the MK class and other classes, and the test statistics showed a significant difference (*p*-value = 9.9e-12). Therefore, in Fig. 2b, we used drug MK-4541 as one example, and there is no clinical information for this drug in the DrugBank, to infer its function based on our embeddings. By excluding the drug MK-8669 due to mismatched information, drugs MK-2206 and MK-4827 have the highest similarity with MK-4541. Since MK-2206 and MK-4827 have similar functions (e.g., treating Breast-cancer-related and Prostate-cancer-related diseases), we may infer that MK-4541 may have a similar effect. Among these drugs, EPTOPOSIDE has the lowest similarity, and it also has different clinical trial information, suggesting that correlation between embedding similarity and function similarity. Therefore, our drug embeddings may help the inference of clinical functions of drugs based on the embeddings' similarity.

To investigate whether the embeddings can be used to predict drug response for the cell-level task, we utilized CPA[9] and single-cell RNA sequencing (scRNA-seq)[40] datasets with different perturbations (defined by different drugs or drug combinations) to evaluate whether our drug embeddings can facilitate the gene expression prediction task. With drug embeddings, we can use CPA to predict gene expression response to unseen drugs. Cells with unknown perturbation results are also known as out-of-distribution (OOD) samples. The original implementation of CPA utilized the drug embeddings from Rdkit[41,42] to encode the molecular structure of the selected drug into the embedding space. However, such methods could not handle drugs not in the Simplified Molecular-Input Line-Entry System (SMILES)[43], which limits the generalization of CPA. Here, we considered replacing the original embeddings in CPA with the embeddings from GPT-3.5, enlarging the accessibility for drug embeddings. We compared three different embedding settings for two datasets (CPA example[9] and Openproblems[44]), which contain the gene expression profiles under the control case and drug-based perturbations. The results are shown in Supplementary Fig. 5a–d, where stacking the embeddings from SMILES and GPT-3.5 achieved the best performance under both datasets. For the CPA dataset, both using the embeddings from GPT-3.5 and the setting of embeddings stacking can enhance the prediction performance significantly, compared with the mode of only using SMILES (Wilcoxon rank-sum test, p-values < 0.05). For the Openproblems dataset, the contribution of such embeddings stacking for prediction is especially significant (p-values < 0.05). Therefore, the embeddings from LLMs can improve the gene expression prediction for perturbed scRNA-seq data.

Since our experiments demonstrate that drug embeddings and cell embeddings can summarize the functional information, and drug embeddings can also interact with cell-level gene expressions, we believe that these embeddings allow us to construct the training dataset to predict the drug synergy effect in different cell lines.

Demonstration of powerful embeddings and architecture by evaluation without pre-training

In this section, we show the strength of LLM embeddings and select the choice of network structure for model pre-training based on two different drug synergy prediction tasks: classification and regression. For each task, we selected two datasets and two metrics for evaluation. For regression, we included the Pearson Correlation Coefficient (PCC) and Mean Squared Error (MSE) for model evaluation based on datasets D1[14] and D2[18]. For classification, we included ROCAUC (ROCAUC) and Accuracy (ACC) for model evaluation based on datasets D1 and D3[19]. These metrics and datasets were widely used in the related work[14,18,19,45,46]. First, we tested if the model's performance would be affected by prompt engineering of LLMs, and we compared the raw embeddings with embeddings generated by drug descriptions from MetaPrompt[47] and Chain-of-Thought (COT)[48]. According to Supplementary Fig. 6b, the differences between the default mode and these two prompt engineering methods are not significant (p-value = 0.63 for raw mode vs. MetaPrompt, and p-value = 0.43 for raw mode vs. COT). Therefore, our embeddings have enough information as inputs for synergetic effects. Second, we validated the contribution of BAITSAO's architecture, shown in Supplementary Fig. 6c. We compared the performances between BAITSAO and DeepSynergy with LLM embeddings as inputs. The difference is significant, and thus, our optimization of model architecture also contributed to the prediction task (-value = 0.002). Finally, we selected seven other methods (DeepSynergy, MARSY, TreeComb, SVM[39,49], TabNet[50], BERT[51], and Lasso[39,52]) for benchmarking the regression task and seven methods (DeepSynergy, DeepDDs, TreeComb, SVC, TabNet, BERT, and Lasso) for benchmarking the classification task. We utilized the best hyperparameters of these methods for every dataset, with details of hyperparameter tuning summarized in the "Methods" section. Our results based on five-fold cross-validation[14] are summarized in Fig. 3. This figure shows that BAITSAO ranked the best in three out of four metrics. Moreover, BAITSAO was also the most stable among the top deep-learning-based methods (including MARSY, DeepSynergy, DeepDDs, and TabNet). The performance of BERT was worse than BAITSAO in three out of four metrics; thus, using embeddings as input is better than using the combination of description in general. For the evaluation based on MSE, BAITSAO performed well on the D1 dataset. Our experiments showed that embeddings from LLMs with a suitable model architecture can formalize a better training-testing framework compared with data from the classical feature space. The details of our dataset information, model construction, and training process are summarized in the "Methods" section.

Explainability of BAITSAO for drug-gene interaction and drug-cell line interaction with multi-modal learning

We interpret contributions of different features for the prediction task with the help of SHAP[53]. Here, we integrated known gene expression profiles of different cell lines in D3 into our input datasets and performed the same training process for the drug synergy prediction task. We then utilized SHAP to study the importance of different genes, and the results could be treated as the relevance between the gene expression levels (as another modality) and the possibility of producing a synergistic effect for drug combinations. We followed the default setting of SHAP to fix the number of genes for explainability at 20. We also performed statistical analysis based on the outputs of BAITSAO to discover the drug combination with the largest range of synergistic targets. The details of our approach are provided in the "Methods" section.

By collecting gene expression profiles of cell lines[54], we studied the explainability of BAITSAO for DEXAMETHASONE (drug)-DINACICLIB (drug) across different cell lines. In Fig. 3b, we visualize the importance of different genes. The gene VIM was top-ranked by the average importance, and VIM is known as important for various cancers from pan-cancer analysis[55]. Furthermore, we conducted three experiments to further investigate the contributions of the selected genes.

We first separated the samples into two groups based on the existence of the synergistic effect and performed DEG analysis using DESeq2[56,57] between the two groups of cell lines. We present the adjusted p-values using Benjamini–Hochberg for the selected genes in Fig. 3b. Genes SPON2, HMCN1, and BMP4 listed in this figure were significant DEGs. Our selected genes had significant overlap with DEGs (Fisher's exact test p-value = 0.0062), and the gene BMP4 is a validated target for each drug according to biology experiments[58,59]. Moreover, these genes had relatively lower expression levels with a synergistic effect, which matched the distribution of their SHAP values (enriched in the negative values).

We list the ranks of the selected genes based on their variances in Fig. 3b. The top five ranked genes had relatively greater variances, suggesting that the genes we selected characterize the heterogeneity from both cell lines and the drug synergistic effect. We further performed enrichment analysis based on Gene Ontology (GO)[60] for biological pathways and Molecular Signature Database (MsigDB)[61] for cancer-specific signals based on this set of genes, with results shown in Supplementary Fig. 7a, c. These enriched pathways represent important biological processes and cancer-specific signals. These results suggest that our method may uncover the heterogeneity in the drug

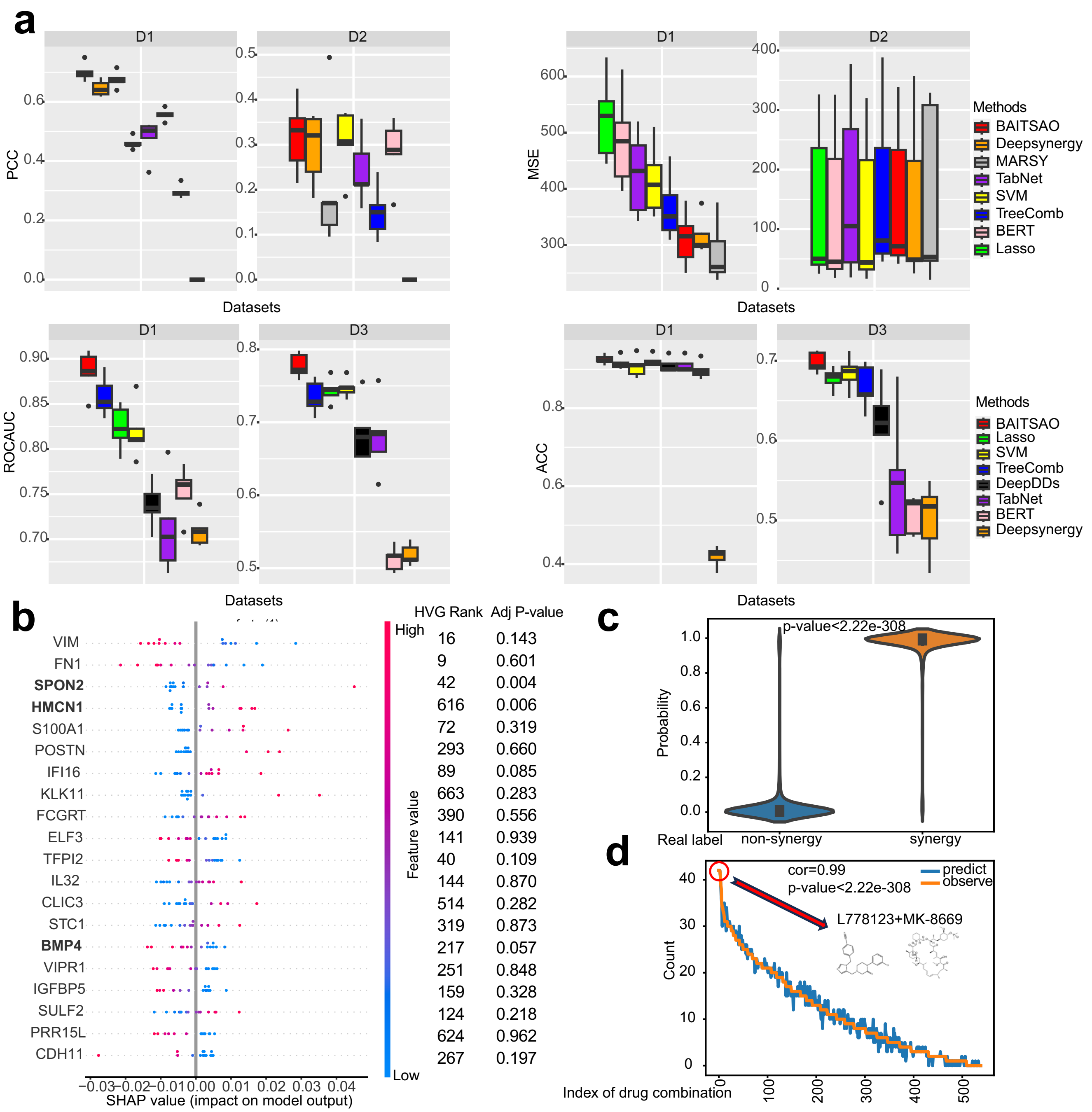


Fig. 3 | Results of evaluations for model structure, reliability, and explainability. a Evaluations for BAITSAO and other methods. Each panel represents one metric with two datasets. The ranks of methods are averaged across datasets. Data are presented as boxplots ($n = 5$ per group; center line, median; box limits, upper and lower quartiles; whiskers, up to 1.5× interquartile range; points, outliers). The explanations of datasets D1-D3 are summarized in the "Methods" section. **b** The explainability of BAITSAO for the combination DEXAMETHASONE (drug)-DINACICLIB (drug) for different cell lines. We also list the ranks based on variance (HVG rank) and the adjusted p-value based on DESeq2 analysis results for each gene. The genes with adjusted p-value for multiple comparisons smaller or close to 0.05 are boldfaced. **c** The violin plot ($n = 6299$ for non-synergy group; $n = 6116$ for synergy group; center point, median; box limits, upper and lower quartiles; whiskers, up to 1.5× interquartile range; points, outliers) for the outputs of BAITSAO (Probability) across the synergistic labels. We also present the two-sided p-value in this figure. This panel supports the reliability of selected features from SHAP. **d** The rank-based plot between the number of drug-cell line combinations with synergy (Count) and the index of drug combination (Index of drug combination). The index is ranked by the value of Count. We present the Pearson correlation (corr) and two-sided p-value in this figure. Source data are provided as a Source Data file.

synergy prediction process. We summarize our results for the single cell line with the same drug combination in Supplementary Note 1. The plots for important genes across different cell lines can be found in Supplementary Fig. 8. The test statistics used in this section are given in Supplementary Data 2.

We further investigated the drug combination that showed synergistic effects on the largest number of cell lines. We first plot the probabilities of all drug-cell line combinations to be classified as samples with synergistic effects in Fig. 3c. This figure shows that the distribution of such probabilities is different under different synergistic labels. We performed the Rank-sum test[62] for these two sets of probabilities, and their difference is significant (p-value < 2.22e-308 with two-sided mode and no adjustment is needed). Therefore, our model can uncover the relationship between input features and the synergistic effect. Moreover, we ranked the drug combinations based on the number of cell lines predicted to have synergistic effects in descending order. We computed the Pearson correlation coefficient[62] between the count value based on predicted labels and observed labels, summarized in Fig. 3d. Based on this figure, the count values based on the predicted labels had a strong positive correlation with those based on the real labels, thus, our model can also be used to identify the drug combinations with the most synergistic targets given a set of cell lines and cancer types used in our experiments. We also highlight the drug combination L778123 and MK-8669 that has the largest number of targeted cell lines with a synergistic effect in Fig. 3d. The p-value is computed with s two-sided mode, and no adjustment is needed. Therefore, BAITSAO can capture the variance of different drug combinations across cell lines, offering a promising option for selecting effective drug combinations.

Statistics of pre-training datasets

Here we summarize the statistics and properties of our pre-training datasets for BAITSAO. We collected information from DrugComb[20], which is known as the largest database containing synergistic effect information for drug pairs with different cell lines. We downloaded the updated version of DrugComb and removed the missing value or single-drug information. The major statistics of DrugComb are summarized in Fig. 4, whose Fig. 4a represents the total number of *drug-cell line* combinations by tissue types and Fig. 4b represents the total number of *cell lines* by the type of tissues from DrugComb. Most of the drugs presented here were analyzed using cells from skin, lymphoid, and/or lung. These tissues are important for maintaining normal physiological activity in the human body. In total, DrugComb collects more than 700,000 available combinations. As shown in Fig. 4c, the distribution of the synergy scores is not balanced, with a large number of combinations having low synergy scores. We further plot the Half Maximal Inhibitory Concentration (IC_50) for all drugs in Fig. 4d with a similar distribution to the synergy score. We illustrate the non-linear relationship between single-drug IC_50 and synergy score in Fig. 4e. Therefore, fitting non-linear models like neural networks may help the synergy prediction task. Finally, Fig. 4f shows the overlap of combinations by tissues, where most tissues have low overlap, and thus, the pre-training dataset has information from diverse tissues. We plot the embeddings for drugs and cell lines in the pre-training dataset, colored by clusters from Leiden[63] in Supplementary Fig. 9a, b. The items in the same Leiden cluster can be treated in a similar context of embeddings with functional information, so we can visualize the functional similarity of different drugs and cell lines through embeddings. Since our pre-training dataset was published in June 2021 and GPT-3.5 collected data for pre-training until Sep 2021, considering the time needed for pre-training an LLM, we believe that the data from DrugComb was precluded in GPT-3.5.

Pre-trained BAITSAO contributes to drug synergistic effect prediction under the multi-task condition

Here, we investigated and pre-trained BAITSAO based on the optimal model structure. Specifically, we extended the model structure with a multi-task learning framework. By pre-training BAITSAO with large-scale synergy datasets, BAITSAO is able to predict both single-drug inhibition and drug synergistic effect. For drug pairs, we expect to predict both drugs' inhibition, thus, we have a total of four tasks inspired by the pre-training datasets, including the regression task for synergy prediction, the classification task for synergy prediction, and regression tasks for single-drug inhibition of each drug in the drug pairs. For the regression task of synergy prediction, we only considered predicting the synergy score under the Loewe setting because we show that the synergy scores computed based on other methods are positively correlated with the Loewe score[64] in Supplementary Fig. 10, and literature[14,25] suggests using the threshold for generating a classification task from the Loewe score. For other synergy scores, including Zip score[65], HSA score[66], and Bliss score[67], we pre-trained specific models and restored the pre-training weights. Instead of using the simple average of loss functions from different tasks during the training process, we introduced the Uncertainty Weighting (UW) method[68] advocated by the performance evaluations of different multi-task learning strategies from ref. 69 and improved the numerical stability and the validation strategy of this method.

We first determined the tasks that can help each other in the multi-task learning framework by constructing the Help-Harm matrix. We sampled 1% of the pre-training dataset and trained task-specific models as well as multi-task models with paired tasks, and constructed the Help-Harm matrix shown in Fig. 5a. According to this figure, joint training always boosts the classification task, while joint training with the classification task can help predict the synergy scores as well as inhibition levels for a single drug. Moreover, the relative inhibition (RI) information from one of the drugs in drug pairs did not show a significant contribution to other tasks, and incorporating this information reduced performance for the classification task. Since we had RI levels for both drug pairs, we removed the information of RI_col in the training process, and collected three tasks in the pre-training stage. After finishing pre-training based on the sampled and full datasets, we plot the metrics for comparing the performance between BAITSAO under the STL framework and our final MTL framework in Fig. 5b. MTL can improve the performance of BAITSAO for solving all regression-based tasks. We show the outputs from the hidden layers of BAITSAO by ground truth synergistic labels and predicted synergistic labels in Supplementary Fig. 11a, b. According to these two figures, the learned drug embeddings for drugs with no synergistic effect tended to be co-embedded. Therefore, BAITSAO with the MTL framework is reasonable and superior in drug synergy analysis. Finally, we consider the generalization ability of BAITSAO with pre-trained weights. We conducted experiments based on three datasets we used in the subsection *Selection of the model structure by evaluation without pre-training*, and visualized the results in Fig. 5c. We report the metrics based on five-fold cross-validation results. According to this figure, BAITSAO with the pre-training design after fine-tuning (BAITSAO-FT) is comparable or better for the regression and classification tasks, compared with BAITSAO without pre-training (BAITSAO-ZS). When evaluating the ZS mode, we ensured that the combinations used in the pre-training stage were not used for testing. Moreover, our fine-tuning stage used fewer epochs, and we froze the shared layers during the fine-tuning process, thus, our fine-tuning approach was more efficient. We note the potential of BAITSAO under the zero-shot learning framework for solving this task. For example, BAITSAO-ZS showed a high ACC score in the evaluation based on D1. Moreover, for the metrics related to classification, BAITSAO-ZS had results higher than 0.5, and thus, BAITSAO under the zero-shot learning framework was better than random guessing. We also performed Rank-sum tests[62] between the pre-training dataset and fine-tuning datasets, and the results are shown in Supplementary Fig. 12, which demonstrates that samples in the fine-tuning datasets satisfied the OOD cases. Finally, we compared BAITSAO with other LLM-based models, discussed in Supplementary Note 2, which shows that BAITSAO also has advantages in modeling synergetic effects. We also pre-trained other deep-learning-based synergy predictors, such as DeepSynergy, DeepDDs, and MARSY, based on their designed tasks and compared the fine-tuned version of these models with BAITSAO (ft). According to

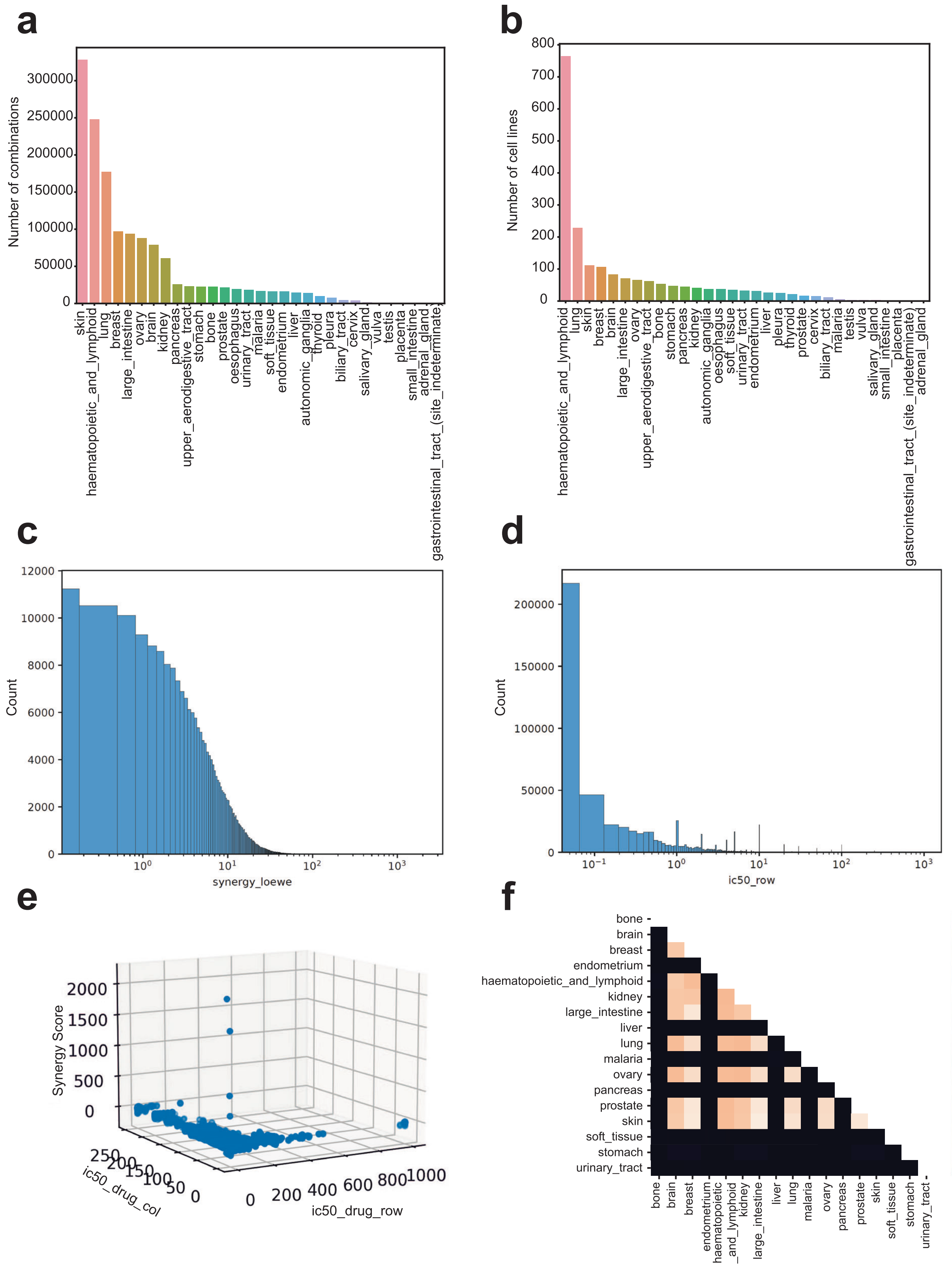


Fig. 4 | Statistics of the pre-training dataset from DrugComb. a The barplot for the number of *drug-cell line* combinations by different tissues. **b** The barplot for the number of *cell lines* by different tissues. **c** The histogram for the distribution of synergy score computed based on Loewe[64]. The x-axis is transferred into a log scale. **d** The histogram for the distribution of single-drug IC_50 levels. The x-axis is transferred to a log scale. **e** 3D plot for the relation between single-drug IC_50 levels and synergy score. **f** The heatmap for the overlap of combinations across different tissues. Source data are provided as a Source Data file.

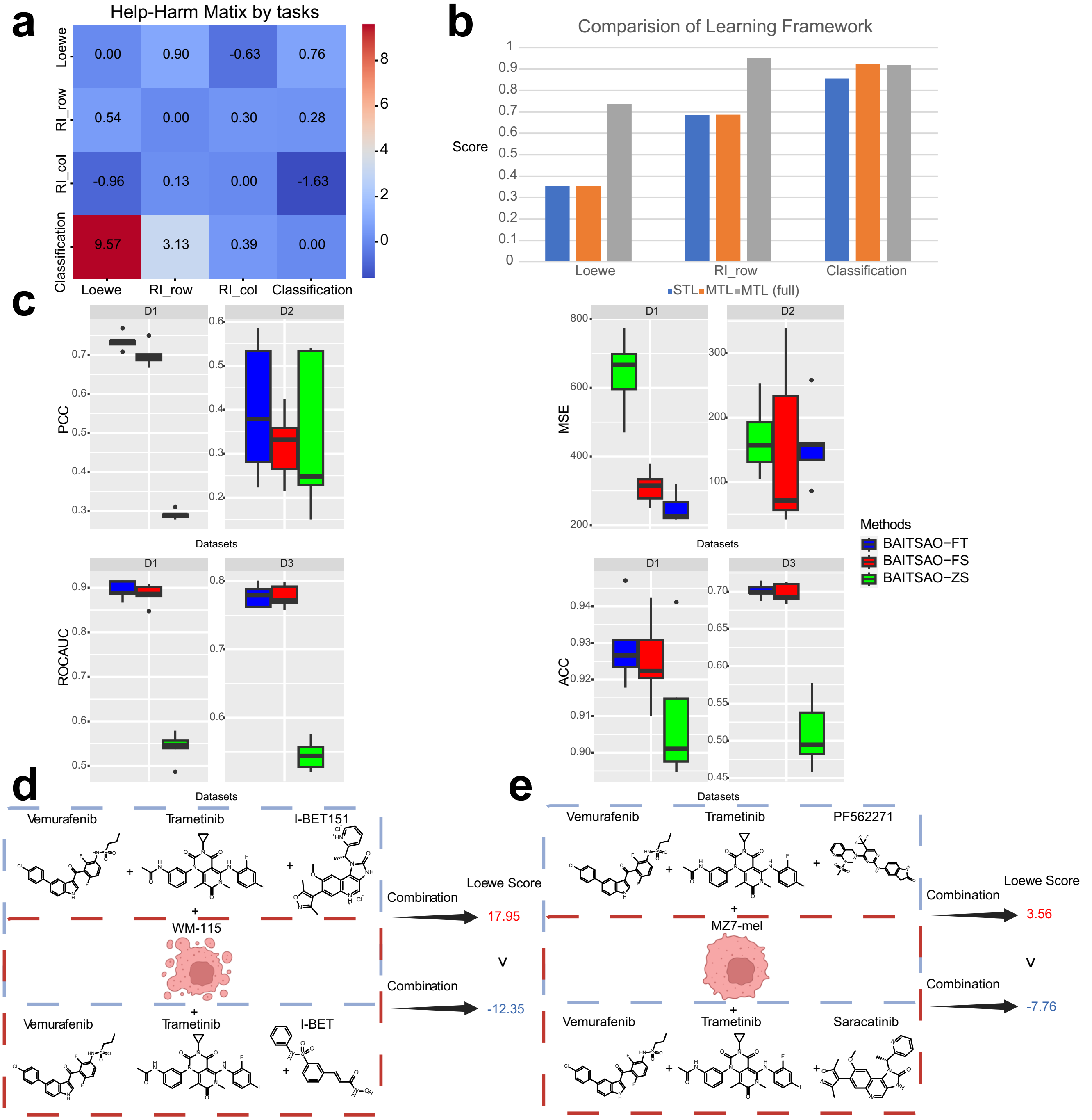


Fig. 5 | Results under the multi-task learning framework. a The Help-Harm matrix for different combinations of tasks. The values indicate the percentage (unit: %) of improvement using multi-task learning compared to single-task learning (STL) defined by the tasks in rows. The columns represent the paired tasks. We boldfaced blocks with increments larger than 0.5%, which is a threshold reported in ref. 96 as an acceptable improvement, and half of the natural threshold 1%. **b** Comparisons for the results under MTL and STL. The metric for regression tasks, including Loewe and RI_row, is PCC. The metric for the classification task, including Classification, is ROCAUC. **c** Comparisons for the results under different training settings. Data are presented in boxplots (*n* = 5 per group; center line, median; box limits, upper and lower quartiles; whiskers, up to 1.5× interquartile range; points, outliers). Here, *BAITSAO-FT* represents that we fine-tuned the pre-trained model, *BAITSAO-ZS* represents that we applied the pre-trained model for these tasks under a zero-shot learning framework, and *BAITSAO-FS* represents that we did not use the pre-trained weights for these tasks. Here, FT means fine-tuning, ZS means zero-shot learning, and FS means from scratch. We included four metrics across three datasets for comparisons. **d** The first example of tri-drug cases for drug synergy prediction with BAITSAO. **e** The second example of tri-drug cases for drug synergy prediction with BAITSAO. Logos of cell lines are created with BioRender.com. Source data are provided as a Source Data file.

Supplementary Fig. 13, BAITSAO still shows better performance than other baselines with either fine-tuning mode or from-scratch mode. Therefore, the multi-task pre-training strategy of BAITSAO is advanced and contributive, which leads to consistent improvement across different datasets. In summary, the combination of MTL and the pre-training process can improve the performance of BAITSAO on tasks related to drug synergy analysis.

We then predicted the synergistic effect for the combination of three drugs (tri-drugs) and one cell line, with two examples shown in Fig. 5d and e. The drug names and cell-line names were extracted from

DrugCombDB[70], which did not provide the observed synergistic information for the existing combinations. To enhance the reliability of our prediction results, we relied on Monte Carlo Dropout (MC Dropout)[71,72] and ran inference 100 times to generate the prediction interval of different drug combinations. According to ref. 73, MC Dropout was the only method considered in this benchmarking paper to estimate the mean and variance without extra hyperparameters. Our full prediction results are summarized in Supplementary Data 3. Here we compared the difference between the two combinations by changing the third drug. We found that the combination with I-BET151 was predicted to have a positive sign in the synergy score under Loewe, while the combination with I-BET was predicted to have a negative synergistic effect. As an explanation, although these two drugs can both combine with bromodomain and extra terminal domain (BET) with the same major targeted proteins[74,75], I-BET151 was reported as an optimized version with excellent BET target potency and selectivity[75]. Therefore, we expected I-BET151 to have better efficacy and thus a higher synergy score. Another example from Fig. 5e presents the difference between PF562271[76] and Saracatinib[77] as a third drug under the cell-line MZ7-mel. The combination with PF562271 had a higher predicted synergy score compared with Saracatinib, which was supported by the experimental results from ref. 78 as PF562271 generated higher growth inhibition. Therefore, the results from BAITSAO can help researchers to optimize drugs with higher synergistic effects and better clinical outcomes.

Sensitivity analysis

Here, we investigated the sensitivity of model training based on the statistics we collected. Figure 6a displays the ablation results by considering different types of embeddings as well as different types of combination rules for embeddings as model input. *BAITSAO* denotes our final choice for pre-training and fine-tuning. *BAITSAO-v3* denotes that we utilized the updated embeddings from OpenAI in 2024[79]. *Mean* denotes that we took the mean of drug embeddings and cell embeddings as input for training. *Sum* denotes that we took the sum of drug embeddings and cell embeddings as input for training. *SentStack* denotes that we stacked the descriptions of different drugs and used the modified description to generate drug embeddings, and then stacked such drug embeddings with cell-line embeddings. *Stack* represents that we stacked the drug embeddings and cell embeddings by rows. *Rdkit*[41,42] represents that we generated embeddings from Rdikt with SMILES and stacked the embeddings with cell embeddings from LLMs. This figure shows that averaging the drug embeddings and stacking them with cell embeddings by rows generated the best performance for all tasks. These results suggest the most effective way to incorporate embeddings from different sources to construct the datasets for training and testing. Moreover, our approach strikes a good balance between efficiency and performance. According to Fig. 6b, the running time of BAITSAO without pre-training was significantly lower than the classical methods, DeepSynergy and SVM, for drug synergistic effect prediction. Moreover, the pre-trained BAITSAO with the fine-tuning framework converges at a much faster rate, thus, pre-trained BAITSAO achieved an even faster running speed compared with MARSY and DeepDDs. Therefore, our training framework strikes a good balance between runtime and model performance. Both pre-training and fine-tuning stages can be finished with only one GPU, presenting no hardware barrier to deploying BAITSAO.

We performed ablation tests for the MTL strategy, shown in Supplementary Fig. 14 for ablation of methods and Supplementary Fig. 15 for ablation of task-specific layers. We compared the gradient matching-based approaches, including PCGrad[80], GradVac[81], CAGrad[82], Nash-MTL[83], and the linear MTL framework LinearMTL[84] with our revised UW approach and found that our choice generally had comparable or better results, especially for the classification task. Moreover, LinearMTL performed much worse than deep-learning based methods on the regression-type tasks. Therefore, we chose the revised UW as the method for the pre-training stage. Moreover, our final choice with one task-specific layer for each task had the best overall performance, and increasing the number of layers required more computing resources, thus, we chose our design shown in Fig. 1c.

We also analyzed the relation between the size of the training dataset and model performance. We adjust the proportion we used for model training and visualize the relation between proportion and metrics in Fig. 6c for regression and Fig. 6d for classification. From these figures, a larger proportion tended to increase the model performance, with its limit for proportion ≥0.9 for these two tasks. Moreover, using only 0.1% training dataset to train a model for a classification task can still generate relatively high ROCAUC, thus the classification task may not be difficult for BAITSAO.

In Fig. 6e and f, we examined the scaling law[85,86] of BAITSAO. We adjusted the layer width of our model and plotted the relation between the layer width in the hidden layer and model performance for the regression task and the classification task. These figures show that we can model the relation between model parameters and model performance to predict performance, where more parameters lead to better performance. Therefore, the performance improvement of our model with scaling can be explained by the scaling law, and our model has good scalability. Our findings can help us understand the model training process in a better approach and determine the optimized source allocation of a fixed compute budget. For example, for machines that cannot support the version of BAITSAO with a layer width of 10,240, the version of BAITSAO with a layer width of 4096 can also have acceptable performances and can be considered for deployment.

Discussion

Predicting drug synergistic effects is important for drug development and patient treatment. In the past, limited by available experimental data, information on drugs/cell lines, and pipelines to predict drug synergistic effect, there were few approaches to predicting drug synergistic effect for general use. With the help of large-scale drug synergy information databases, LLMs, and an MTL framework, we introduced BAITSAO as a unified model with a general pipeline for drug synergistic effect prediction as well as single-drug inhibition prediction. BAITSAO optimized the network architecture through comprehensive benchmarking analysis and was pre-trained based on the latest large-scale databases. It achieved top-tier performance in both regression tasks and classification tasks for drug synergistic effect prediction.

There are two major contributions of our work. Firstly, we presented a unified pipeline to construct datasets for synergistic effect analysis for both drugs and cell lines based on the embeddings from LLMs, thus, we mitigated the difference caused by aliases for drugs and cell lines of different datasets. We demonstrated that the embeddings contained functional information for drugs and cell lines. We proposed an advanced design to construct training datasets, thus, we only need to utilize the overlapped information across datasets for drug synergy analysis. Secondly, we pre-trained a unified model with an MTL framework for drug synergy analysis and single-drug inhibition analysis supported by rigorous task-selection steps. We demonstrated that BAITSAO benefited from the pre-training process and had good generalization ability with fine-tuning in fewer steps compared with the training process from scratch. Moreover, pre-trained BAITSAO showed its potential as a good zero-shot reasoner for drug synergy prediction under the classification settings. Therefore, we overcame the generalization issue in previous work based on transfer learning[21] and proposed a new avenue for the construction of BAITSAO for drug synergy analysis.

We conducted a sensitivity analysis to offer guidance for future model deployment. We showed that our current hyperparameter settings and data construction methods are the optimal choices by

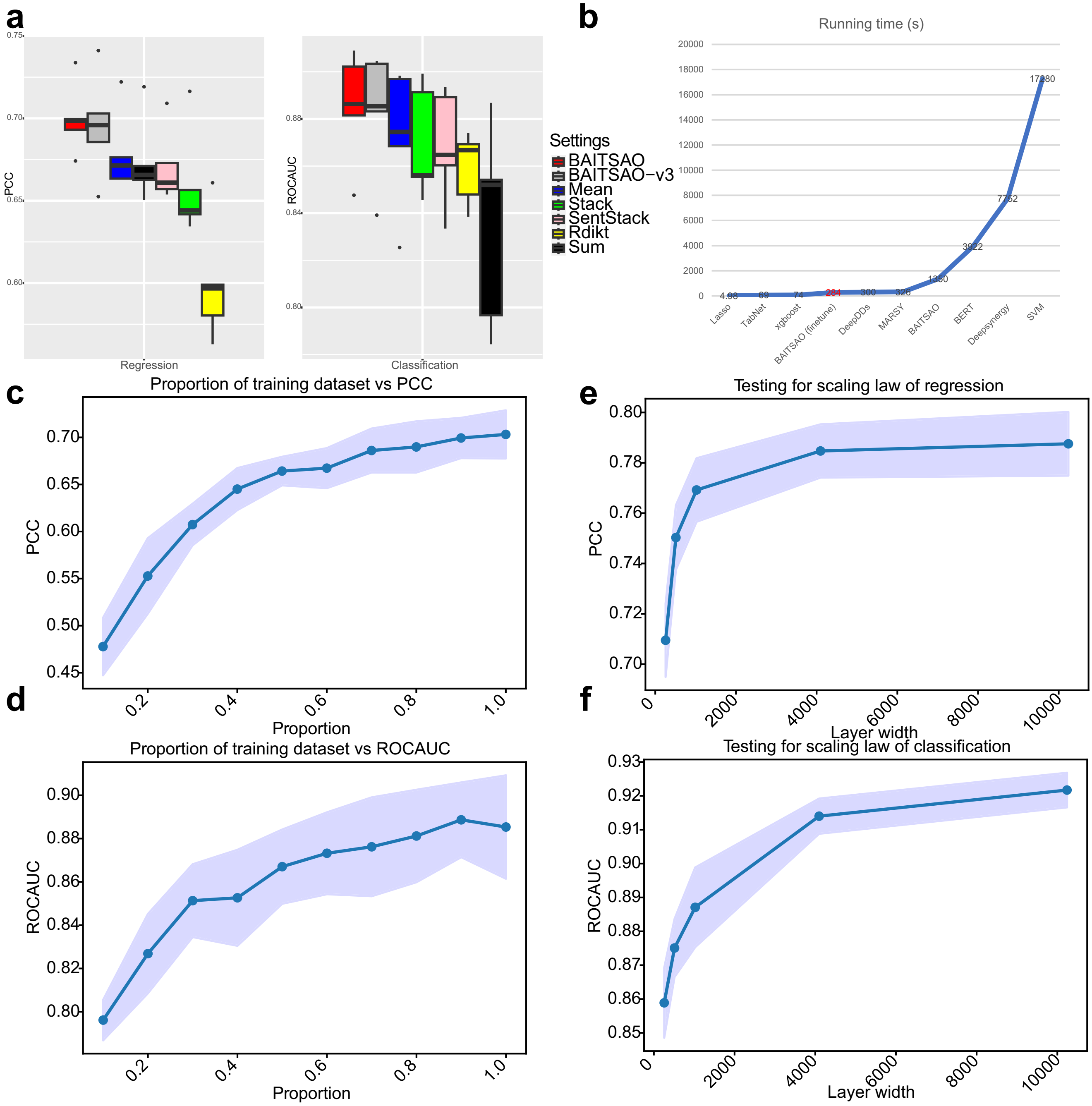


Fig. 6 | Statistics of model training. a Ablation test results for BAITSAO with different input formats. ($n = 5$ per group; center line, median; box limits, upper and lower quartiles; whiskers, up to 1.5× interquartile range; points, outliers). **b** The comparison of running time for different methods. We highlight the running time of BAITSAO and use the regression task as an example. **c** Plot for the proportion of the training dataset and PCC for BAITSAO under the regression task. We reported $(\mu - \sigma, \mu + \sigma)$ for each proportion, where μ represents the mean and σ represents the standard deviation. **d** Plot for the proportion of the training dataset and ROCAUC for BAITSAO under the classification task. We reported $(\mu - \sigma, \mu + \sigma)$ for each proportion. **e** Plot for layer width and PCC for BAITSAO under the regression task. We reported $(\mu - \sigma, \mu + \sigma)$ for each proportion. **f** Plot for layer width and ROCAUC for BAITSAO under the classification task. We reported $(\mu - \sigma, \mu + \sigma)$ for each proportion. Source data are provided as a Source Data file.

hyperparameter tuning and ablation tests. We also analyzed the relation between the proportions of data we used for training and model performance. While increasing training data proportions tended to improve prediction, BAITSAO performed well for the classification task for small data scales. Finally, we investigated the scaling law of BAITSAO and showed that the model performance is predictable, and we could increase the model performance by scaling up BAITSAO for drug synergy prediction.

In conclusion, we have developed BAITSAO, an explainable model for drug synergy prediction, and demonstrated the superiority of BAITSAO over other methods by comprehensive benchmarking analysis and rigorous sensitivity analysis. We hope that BAITSAO can help researchers to better understand the process of drug synergistic effect prediction and further help in optimizing drug structures for drug design and discovering more drug combinations with synergistic effects for clinical usage.

Furthermore, we also found that BAITSAO might not work well for drugs without a clear functional or chemical description in the early stage of drug development, which is a potential limitation of our application scenarios of all functional-based synergy predictors. In the future, we plan to incorporate more updated drug synergy databases to keep this model updated, and we also plan to combine this model with information from genomics, including single-cell data[87] and genome-wide association studies (GWAS)[88], especially for early-stage drugs.

Methods

Problem definition

In this manuscript, we intend to construct a dataset $\mathcal{D} = (X, Y)$ and pre-train a model known as $\mathcal{M}$ for the prediction of values in $Y^{n \times t}$, where n represents the number of combinations between drug pairs and the cell line, and t represents the number of tasks. Here, $X^{n \times p}$ represents the feature space with n samples and p features. We then split the dataset $\mathcal{D}$ into $\mathcal{D}_{train} = (X, Y)_{i=1}^{n_0}$ for training, and $\mathcal{D}_{val} = (X, Y)_{i=1}^{n_1}$ for validation. Our target is to train a model $\mathcal{M}^*$ based on $\mathcal{D}_{train}$ and then select the optimal model based on $\mathcal{D}_{val}$. That is,

$$\theta^* = argmin_{\theta} \mathcal{L}_m(\mathcal{M}(X_{val}, \theta), Y_{val}), \tag{1}$$

where $\mathcal{M}(, \theta)$ represents the pre-trained model with parameter θ, and θ^* represents the optimal model parameters. $\mathcal{L}_m$ represents multi-task learning loss. After obtaining the optimal model, we apply the model $\mathcal{M}^*(, \theta^*)$ for a new dataset containing out-of-distribution (OOD) data, known as $\mathcal{D}_{test}$.

Construction of pre-training datasets and testing datasets

One major contribution of our work is to unify the features we need to predict the drug-related information for both the synergistic effect and the inhibition effect. We at least need the names of drugs and cell lines. Considering we have a drug pair (d_1, d_2) and a cell line (c_1), our idea is to generate the description of both drugs as $W(d_1)$, $W(d_2)$ and the cell line as $W(c_1)$ based on LLMs such as GPT-3.5, and then utilize the embeddings tool of GPT-3.5 to transfer the text description into embeddings with e dimensions. Therefore, our final sample $x \in X$ is defined as:

$$x = AVG(emb(W(d_1)), emb(W(d_2)))||emb(W(c_1))||\#d, \tag{2}$$

where $AVG()$ represents the functions to compute the mean of the given variables, and $emb()$ is the function to obtain the embeddings of the input. $\#d$ represents the number of drugs we used, which can be encoded as embeddings[89]. We take the unbiased estimation of the drug combination in the feature levels by computing the average value of embeddings, and we show that this approach works better than other types of feature integration in the "Sensitivity analysis" section of the manuscript. Notably, this approach also scales for more drug combinations. Considering the case of k drugs with the cell line c_i, we define one sample $x \in X$ as:

$$x = AVG(emb(W(d_1)), \ldots, emb(W(d_k)))||emb(W(c_i))||k. \tag{3}$$

Therefore, for an arbitrary input dataset with feature space X containing drug information and cell-line information, we can transfer the samples in the given dataset from the text space to the numerical space, thus we unify the input data format for this task. Furthermore, to predict the drug synergistic effect, we consider both regression and classification. In the case of regression, we intend to predict the specific synergy score of samples. To compute the synergistic effect based on IC_50 information, under different rules, we can have different scores. Here we consider four methods to model the synergy scores, known as HSA, Bliss, Loewe, and ZIP. If we consider N drugs with multi-drug combination effect as $E_{A,B,\ldots,N}$ and we intend to compute the synergy scores S_{HSA}, S_{Bliss}, S_{Loewe}, and S_{ZIP}, according to ref. 5, we have:

$$S_{HSA} = E_{A,B,\ldots,N} - \max(E_A, E_B, \ldots, E_N). \tag{4}$$

$$S_{Bliss} = E_{A,B,\ldots,N} - (E_A + E_B + \ldots + E_N - E_A E_B \tag{5}$$

$$- E_A E_N - E_B E_N - \ldots - E_A E_B \ldots E_N). \tag{6}$$

$$S_{Loewe} = \frac{a}{E_A} + \frac{b}{E_B} + \ldots + \frac{n}{E_N}. \tag{7}$$

$$S_{ZIP} = E_{A,B,\ldots,N} - \left(\frac{\left(\frac{x_A}{m_A}\right)^{\lambda_A}}{1 + \left(\frac{x_A}{m_A}\right)^{\lambda_A}} + \frac{\left(\frac{x_B}{m_B}\right)^{\lambda_B}}{1 + \left(\frac{x_B}{m_B}\right)^{\lambda_B}} + \ldots \right. \tag{8}$$

$$+ \frac{\left(\frac{x_N}{m_N}\right)^{\lambda_N}}{1 + \left(\frac{x_N}{m_N}\right)^{\lambda_N}} - \frac{\left(\frac{x_A}{m_A}\right)^{\lambda_A}}{1 + \left(\frac{x_A}{m_A}\right)^{\lambda_A}} \frac{\left(\frac{x_B}{m_B}\right)^{\lambda_B}}{1 + \left(\frac{x_B}{m_B}\right)^{\lambda_B}} \tag{9}$$

$$- \frac{\left(\frac{x_A}{m_A}\right)^{\lambda_A}}{1 + \left(\frac{x_A}{m_A}\right)^{\lambda_A}} \frac{\left(\frac{x_N}{m_N}\right)^{\lambda_N}}{1 + \left(\frac{x_N}{m_N}\right)^{\lambda_N}} \tag{10}$$

$$- \frac{\left(\frac{x_B}{m_B}\right)^{\lambda_B}}{1 + \left(\frac{x_B}{m_B}\right)^{\lambda_B}} \frac{\left(\frac{x_N}{m_N}\right)^{\lambda_N}}{1 + \left(\frac{x_N}{m_N}\right)^{\lambda_N}} - \ldots \tag{11}$$

$$\left. - \frac{\left(\frac{x_A}{m_A}\right)^{\lambda_A}}{1 + \left(\frac{x_A}{m_A}\right)^{\lambda_A}} \frac{\left(\frac{x_B}{m_B}\right)^{\lambda_B}}{1 + \left(\frac{x_B}{m_B}\right)^{\lambda_B}} \ldots \frac{\left(\frac{x_N}{m_N}\right)^{\lambda_N}}{1 + \left(\frac{x_N}{m_N}\right)^{\lambda_N}} \right). \tag{12}$$

Here we have E_A, E_B, . . . , E_N as measured responses of different drugs, and $a, b, \ldots, n$ represent the doses of the single drugs we need to produce the combination effect. Moreover, to compute S_{ZIP}, we have x_N as the dose of drug N fitted with the four-parameter log-logistic model, and m_N represents the dose we need to produce the half-maximum effect (IC_50). We also have λ_N as the shape parameter to indicate the slope of the dose-response curve. In the MTL case, we consider S_{Loewe} for the targets of regression. We also pre-train models to predict the other three scores. All of the synergy scores are extracted from the database of DrugComb.

In order to characterize the inhibitory effects of individual drugs, we introduce the RI score in the prediction task. RI score is the normalized area under the log_{10} − transformed dose-response curves. RI scores of all drugs are also extracted from the database of DrugComb.

In the case of classification, we intend to predict whether the given drug pair has a synergistic effect under a specific cell line, which is a binary classification problem. To construct the dataset for this task, we set the threshold of S_{Loewe} to binarize the synergistic effect of different drug combinations. Since not all of the testing datasets in the real world contain data for both regression and classification, introducing a classification task is meaningful.

Investigation of embeddings

We set up different methods to ensure that embeddings from LLMs contain the necessary information to describe the properties of drugs and cell lines. We consider two prompt engineering approaches for description generation, including MetaPrompt[47] and Chain-of-Thought (COT)[47]. MetaPrompt introduces a system prompt for LLMs

Table 1 | Hyperparameter search space for each method ranked in alphabetical order

Methods	Searching space
BAITSAO	lr:[1e-5,1e-4]; Dropout:[0.1,0.3]
BERT	epochs:[5,10]; lr: [1e-5,1e-4]
DeepDDs	Optimal hyperparameters from ref. 19
DeepSynergy	Optimal hyperparameters from ref. 14
Lasso	alpha:[1,10]
MARSY	Optimal hyperparameters from ref. 18
SVM	C:[1,5]
TabNet	n_steps:[1,5]; n_a:[8,64]; n_d:[8,64]; gamma:[1.1,1.5]
TreeComb	max_depth:[10, 100]; n_estimators:[10,50]; min_child_-weight:[1, 3]

and generates outputs conditioned on the context. COT allows LLMs to obtain complex reasoning capabilities by forcing models to address the problem with intermediate steps. We also generate text descriptions and embeddings for drugs and cell lines from the dataset used by Deepsynergy (D1). We check the correctness of all descriptions and the similarity of 10 sampled embeddings across different drugs to evaluate the correctness. Moreover, we change the random seed to generate different descriptions as well as embeddings to check the variance of drug embeddings from the same drug. We also record the description of cell lines in Supplementary Data 1. Furthermore, we modify CPA to predict gene expression under different perturbations enhanced by drug embeddings from LLMs. In this step, we replace the original drug embeddings used in the CPA with our new embeddings. This approach allows us to check the correctness of embeddings from the application perspective. We use R^2 scores to evaluate the performance. To compute the R^2 score, we have the ground truth synergy score y and the predicted synergy score $\hat{y}$ and follow its definition:

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2}, \tag{13}$$

where $\bar{y}$ means that we compute the average value of the input variable y. R^2 represents the explanation of the independent variables for the dependent variable. Higher R^2 means better model performance.

Therefore, our assessment of the quality of embeddings takes into account meanings, variance, and applications.

Hyperparameter searching

We summarize the search space for hyperparameters of each method in Table 1. The best hyperparameter setting is determined by the performance of models based on the validation dataset.

Here lr means learning rate, Dropout means dropout rate (the ratio of neurons we intend to close during the training process), max_depth means the maximal depth for tree-based models, n_estimators means the number of estimators for tree-based models, min_child_weight means the minimal weights of child nodes in tree-based models, C means the regularization weight for SVM, n_steps means the number of decision steps in the model architecture, n_a means the width of the attention embedding for each masked choice, n_d means the width of the prediction layer, gamma means the coefficient for the feature re-usage in the masking process, epochs mean the number of epochs we used to train the model, alpha represents the regularization coefficient for Lasso. We present the results under different hyperparameters for BAITSAO in Supplementary Fig. 16a, b. We find that lr plays a more important role in the training process, while adjusting the dropout rate does not affect the model performance much.

Selection of model architecture

After setting up the pre-training dataset, we seek a suitable model architecture. Since deep neural networks (DNNs) related methods have shown impressive performance as a base model for large-scale models[90,91], we construct the pre-training architecture of BAITSAO based on DeepSynergy. To assess the strength of our model architecture, we remove the pre-training step and compare BAITSAO with other methods for both the regression task and classification task with three different datasets. We also determine the hyperparameters of model training in this stage. The superiority of BAITSAO is shown in the "Results" section, and we expect to see its similar performance at both the pre-training and fine-tuning stages.

In the model architecture selection stage, we utilize Adam[92] as the optimizer and ReduceLROnPlateau[89] as the learning rate scheduler. The starting learning rate for D1 and D3 is 1e-5, while it is 1e-4 for D2. The dropout rate is 0.2, and the patience for the scheduler is 10. Our patience for the early-stopping step is 100, and the maximum number of epochs is 1000.

Explainability

The design of BAITSAO allows us to characterize the relevance between the specific gene and drug combinations across different cell lines. To perform this analysis, the input format of one combination becomes:

$$x' = AVG(emb(W(d_1)), \ldots, emb(W(d_k)))||exp(c_i)||emb(W(c_i))||k, \tag{14}$$

where $exp(c_i)$ represents the gene expression profile for the cell line i. After the training process, we can extract the importance of different genes based on SHAP. For gene j, its importance for the synergistic effect of drug combinations $(d_1, \ldots, d_k)$ for cell line c_i can be calculated as:

$$I_j = ShapValue(\mathcal{M}, x'), \tag{15}$$

where I_j represents the importance and x' was defined above. *ShapValue*() is a function to compute the importance of model $\mathcal{M}$ and input x'. Here larger I_j represents more importance in the prediction process.

We select 1000 highly variable genes for the analysis of explainability. This number is determined by adjusting the number of genes to achieve the best model performance. Our tuning results are shown in Supplementary Fig. 17.

For the bulk RNA-seq datasets of different cell lines, we use DESeq2 to identify DEGs by comparing groups with and without predicted drug synergistic effects.

For the two scRNA-seq datasets used for validating our selected genes, we follow the pre-processing pipeline of Scanpy[93] and run the Wilcoxon rank-sum test to access the list of DEGs.

Pre-training BAITSAO under the multi-task learning framework

Here we explain our settings for the multi-task learning framework. After filtering tasks based on the constructed help-harm matrix, we consider three tasks: 1. Prediction of S_{Loewe} as a regression task. 2. Prediction of RI for one drug as a regression task. 3. Prediction of drug synergistic effect as a binary classification task. Therefore, we have two regression tasks and one classification task, and their loss functions are represented as $\mathcal{L}_1$, $\mathcal{L}_2$, and $\mathcal{L}_3$. Traditionally, we construct the final loss function as a linear combination:

$$\mathcal{L}_m = \sum_{i=1}^{n_t} w_i \mathcal{L}_i, \tag{16}$$

where w_i represents the pre-defined weights for the loss function $\mathcal{L}_i$ and $n_t = 3$. However, determining the values of the weights is difficult. Moreover, it is a strong assumption that the weights do not change during training is also a very strong assumption. Therefore, we introduce the uncertainty of the loss function in this process and make the weights learnable. Typically, by choosing mean squared error (MSE) as the loss function in the training process for the regression task, we have the equivalent maximum likelihood framework of a Gaussian distribution for prediction output y and model $\mathcal{M}$. Therefore, the log-likelihood of the regression task can be represented as:

$$log(p(y|\mathcal{M}(x,\theta))) \propto -\frac{1}{2\sigma^2}||y-\mathcal{M}(x,\theta)||^2 - log\sigma, \tag{17}$$

where the uncertainty is defined as σ, and x represents the model input. σ is a learnable parameter. Similarly, for a classification problem, we can represent the log-likelihood based on a Softmax function, that is:

$$log(p(y|\mathcal{M}(x,\theta))) = log(Softmax(\frac{1}{\sigma^2}\mathcal{M}(x,\theta))), \tag{18}$$

where $Softmax(x)_i = \frac{exp(x_i)}{\sum_j^p exp(x_j)}$, and p represents the length of x.

Therefore, the final loss function can be represented as maximizing the joint distribution of three tasks. In the validation stage, we minimize the maximal term in the loss function group rather than the original weighted loss function design from UW. We also add a constant term ϵ to ensure numerical stability, so our final loss function is:

$$\mathcal{L}_m = -log(p(y_1, y_2, y_3|M(x,\theta))) \tag{19}$$

$$\approx \frac{1}{2\sigma_1^2+\epsilon}\mathcal{L}_1(y_1, M(x,\theta)) + \frac{1}{2\sigma_2^2+\epsilon}\mathcal{L}_2(y_2, M(x,\theta)) \tag{20}$$

$$+\frac{1}{\sigma_3^2+\epsilon}\mathcal{L}_3(y_3, M(x,\theta)) + log(\sigma_1\sigma_2\sigma_3). \tag{21}$$

In the pre-training stage, we utilize Adam[92] as the optimizer and ReduceLROnPlateau[89] as the learning rate scheduler. The starting learning rate is 1e-4, the dropout rate is 0.2, and the patience for the scheduler is 100. Our patience for the early-stopping step is 500, and the maximum number of epochs is 1000. The number of combinations we used for pre-training is 739,652, including 4268 types of drugs and 288 types of cell lines.

After finishing the pre-training step, we test the model's performance on the testing datasets under both the zero-shot learning case and the fine-tuning case with a parameter-freezing design. We also extend the prediction of the synergistic effect to the case of $n (n \geq 3)$ drug combinations. Finally, we include a tutorial in our code repository for both the fine-tuning approach and the zero-shot inference approach.

We also pre-train other baselines, including DeepSynergy[14], DeepDDs[19], and MARSY[18], based on the same dataset. Details of model comparison are discussed in the "Results" section.

Zero-shot query and multi-drug prediction

Our model is capable of zero-shot synergy effect prediction. By transferring the knowledge and information of drugs and cell lines into embeddings through GPT-3.5 and the embedding layer, users can generate embeddings of arbitrary combinations as input for querying the synergy effects with a pre-trained BAITSAO.

For the combinations with three or more drugs, we directly generate the synergy score under the pre-trained model with the zero-shot learning framework. The three-drug case we used in the main text is from a known database, while it is possible to explore combinations with a larger number of drugs as long as the combinations are practical and meaningful. To access the determined predicted value, we do not use the dropout layers in the testing process. To access the predicted value with uncertainty, we keep the dropout layers in the testing process and repeat the prediction process 100 times to access the estimation of mean and standard deviation for each combination. Such an approach is known as MC Dropout.

We summarize the details of zero-shot query as a tutorial in our code repository.

Model evaluation

We consider four different metrics to evaluate the performance of different models for the drug synergistic effect prediction task, with two metrics for regression and two metrics for classification.

For the regression task, we consider two metrics: Pearson correlation coefficient (PCC) and Mean Squared Error (MSE).

1. PCC: Since we know the ground truth synergy score y and predicted synergy score $\hat{y}$, we can directly compute the PCC as:

$$PCC(y,\hat{y}) = \frac{COV(y,\hat{y})}{\sigma(y)\sigma(\hat{y})}, \tag{22}$$

where $COV()$ is the function to compute the covariance of two variables, and $\sigma()$ is the function to compute the standard deviation of the input variable. Higher PCC means better model performance.

2. MSE: To compute the mean squared error, we have the ground truth synergy score y and the predicted synergy score $\hat{y}$ and follow its definition:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2, \tag{23}$$

where i represents the index of samples, and lower MSE means better model performance.

For the classification task, we consider two metrics: Area under the ROC Curve (ROCAUC) and Accuracy (ACC).

1. ROCAUC: To compute this metric, we construct the relation between the true-positive rate and the false-positive rate under different probability thresholds. Such a relation can be reflected in the ROC curve. We then compute the area under the ROC curve, and this area represents ROCAUC. Higher ROCAUC means better model performance.

2. ACC: To compute this metric, we have the ground truth synergistic effect condition $y^{n\times 1}$ and the predicted binary value $\hat{y}$, we then compute the ACC as:

$$ACC = \frac{\sum_{i=1}^{n}\mathbb{1}_{y_i=\hat{y}_i}}{n}, \tag{24}$$

where $\mathbb{1}_{y_i=\hat{y}_i}$ is an indicator function and only takes 1 when $y_i = \hat{y}_i$. Higher ACC means better model performance.

We report the mean and standard deviation of these metrics by using five-fold cross-validation for each dataset.

Overview of other methods

In this section, we summarize the benchmarking methods used in our work. These methods (ranked in alphabetical order) include:

- BERT: BERT is a pre-trained bidirectional transformer for language understanding. For this model, we construct the training datasets and testing datasets directly from drug descriptions and cell-line descriptions. The problem is then formalized as a Question-answering case for both classification and regression tasks.
- DeepDDs[19]: DeepDDs is a Graph Neural Network (GNN)-based method for drug synergistic effect prediction. This method can only handle the classification task. The training dataset of DeepDDs is constructed based on features of drugs as graphs from chemical information and gene expression levels from cell lines.

- DeepSynergy[14]: DeepSynergy is a DNN-based method for drug synergistic effect prediction. This method can handle both the regression task and the classification task by changing the loss function and the activation function of the last network layer. The training dataset of DeepSynergy follows its default mode, including features of drugs from chemical information and cell-line features from gene expression levels.
- Lasso[39,52]: Lasso is a regularized regression method for drug synergistic effect prediction. This method can handle both the regression task and the classification task, by using the default mode or logistic regression mode with L1 penalty. The training dataset of Lasso is constructed based on the drug embeddings and cell-line embeddings from LLMs.
- MARSY[18]: MARSY is a DNN-based method with a multi-task learning framework for drug synergistic effect prediction. This method can only handle the regression task. The training dataset of MARSY is constructed based on features of drugs from chemical information, gene expression levels from cell lines, and tissue information.
- SVM[39,49]: SVM is a machine learning method based on constructing decision-making boundaries for drug synergistic effect prediction. This method can handle both the regression task and the classification task by using SVR or SVC. The training dataset of SVM is constructed based on the drug embeddings and cell-line embeddings from LLMs.
- TabNet[50]: TabNet is a DNN-based method with a transformer architecture for drug synergistic effect prediction. TabNet combines the ideas from both neural network design and tree-model design. This method can handle both the regression task and the classification task by changing the loss function and the activation function of the last network layer. The training dataset of TabNet is constructed based on the drug embeddings and cell-line embeddings from LLMs.
- TreeComb[15]: TreeComb is an explainable machine learning method based on XGBoost for drug synergistic effect prediction. This method can handle both the regression task and the classification task by using XGBREGRESSOR or XGBCLASSIFIER. The training dataset of TreeComb is constructed based on the drug embeddings and cell-line embeddings from LLMs.

Datasets preparation

We utilize public datasets from DrugComb v1.5 for pre-training. For the regression task, we have one dataset from DeepSynergy (as D1, which is processed in the original paper) using Loewe as the synergy score computation method. We also have one dataset from MARSY (as D2, which is processed in the original paper) using ZIP as the synergy score computation method. For the classification task, we have one dataset from DeepSynergy (as D1) using the Loewe synergy score with a threshold. We also have one dataset from DeepDDs with a known binary synergistic effect condition (as D3[94]). For multi-drug synergistic effect inference, we utilize one dataset from DrugCombDB. Every dataset at least contains the names of drugs and cell lines.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

We do not generate new data in this research, and all data used in this manuscript are publicly available without restricted access. The DrugComb data used in this study are available at [https://drugcomb.org/download/]. The training and testing data used in this study are available at [https://www.bioinf.jku.at/software/DeepSynergy/], [https://github.com/Emad-COMBINE-lab/MARSY], and [https://github.com/Sinwang404/DeepDDs/tree/master]. The scRNA-seq data used in this study are available under accession codes GSE215121 and SCP109. We collect the information on downloading training datasets as well as their statistics in Supplementary Data 4. Source data are provided with this paper.

Code availability

We used the resources from the Yale High Performance Center (Yale HPC) and UCLA Computing Servers to conduct all of the experiments. Our maximum running time for each dataset was 24 h, and the maximum RAM was 100 GB. The version of GPU we used is NVIDIA A5000 (24 GB) for fine-tuning and single-task learning, and NVIDIA A100 (40GB) for pre-training. The codes of BAITSAO can be found in https://github.com/HelloWorldLTY/BAITSAO and https://doi.org/10.5281/zenodo.15105815[95] with MIT license. The pre-trained weights can be found at https://huggingface.co/iLOVE2D/BAITSAO. The version of software used for data collection and model training is summarized in Supplementary Data 4.

References

1. Clercq, E. D. The design of drugs for hiv and hcv. *Nat. Rev. Drug Discov.* **6**, 1001–1018 (2007).
2. Mokhtari, R. B. et al. Combination therapy in combating cancer. *Oncotarget* **8**, 38022 (2017).
3. Al-Lazikani, B., Banerji, U. & Workman, P. Combinatorial drug therapy for cancer in the post-genomic era. *Nat. Biotechnol.* **30**, 679–692 (2012).
4. Holbeck, S. L. et al. The national cancer institute almanac: a comprehensive screening resource for the detection of anticancer drug pairs with enhanced therapeutic activity. *Cancer Res.* **77**, 3564–3576 (2017).
5. Ianevski, A., Giri, A. K. & Aittokallio, T. Synergyfinder 2.0: visual analytics of multi-drug combination synergies. *Nucleic Acids Res.* **48**, 488–493 (2020).
6. Law, M., Wald, N., Morris, J. & Jordan, R. Value of low dose combination treatment with blood pressure lowering drugs: analysis of 354 randomised trials. *BMJ* **326**, 1427 (2003).
7. Wood, K. B., Wood, K. C., Nishida, S. & Cluzel, P. Uncovering scaling laws to infer multidrug response of resistant microbes and cancer cells. *Cell Rep.* **6**, 1073–1084 (2014).
8. Hetzel, L. et al. Predicting cellular responses to novel drug perturbations at a single-cell resolution. *Adv. Neural Inf. Process. Syst.* **35**, 26711–26722 (2022).
9. Lotfollahi, M. et al. Predicting cellular responses to complex perturbations in high-throughput screens. *Mol. Syst. Biol.* **19**, e11517 (2023).
10. Wilding, J. L. & Bodmer, W. F. Cancer cell lines for drug discovery and development. *Cancer Res.* **74**, 2377–2384 (2014).
11. Ianevski, A., Giri, A. K. & Aittokallio, T. Synergyfinder 3.0: an interactive analysis and consensus interpretation of multi-drug synergies across multiple samples. *Nucleic Acids Res.* **50**, 739–743 (2022).
12. Roemer, T. & Boone, C. Systems-level antimicrobial drug and drug synergy discovery. *Nat. Chem. Biol.* **9**, 222–231 (2013).
13. Sun, W., Sanderson, P. E. & Zheng, W. Drug combination therapy increases successful drug repositioning. *Drug Discov. Today* **21**, 1189–1195 (2016).
14. Preuer, K. et al. Deepsynergy: predicting anti-cancer drug synergy with deep learning. *Bioinformatics* **34**, 1538–1546 (2018).
15. Janizek, J. D., Celik, S. & Lee, S.-I. Explainable machine learning prediction of synergistic drug combinations for precision cancer medicine. Preprint at *bioRxiv* https://doi.org/10.1101/331769 (2018).
16. Janizek, J. D. et al. Uncovering expression signatures of synergistic drug responses via ensembles of explainable machine-learning models. *Nat. Biomed. Eng.* **7**, 811–829 (2023).
17. Kuru, H. I., Tastan, O. & Cicek, A. E. Matchmaker: a deep learning framework for drug synergy prediction. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **19**, 2334–2344 (2021).

18. El Khili, M. R., Memon, S. A. & Emad, A. Marsy: a multitask deep-learning framework for prediction of drug combination synergy scores. *Bioinformatics* **39**, 177 (2023).
19. Wang, J. et al. DeepDDs: deep graph neural network with attention mechanism to predict synergistic drug combinations. *Brief. Bioinform.* **23**, 390 (2022).
20. Zheng, S. et al. Drugcomb update: a more comprehensive drug sensitivity data repository and analysis portal. *Nucleic Acids Res.* **49**, 174–184 (2021).
21. Kim, Y. et al. Anticancer drug synergy prediction in understudied tissues using transfer learning. *J. Am. Med. Inform. Assoc.* **28**, 42–51 (2021).
22. Bommasani, R. et al. On the opportunities and risks of foundation models. Preprint at https://arxiv.org/abs/2108.07258 (2021).
23. Min, B. et al. Recent advances in natural language processing via large pre-trained language models: a survey. *ACM Comput. Surv.* **56**, 1–40 (2023).
24. Zhao, W. X. et al. A survey of large language models. Preprint at https://arxiv.org/abs/2303.18223 (2023).
25. Edwards, C. N. et al. Synergpt: in-context learning for personalized drug synergy prediction and drug design. Preprint at *bioRxiv* https://doi.org/10.1101/2023.07.06.547759 (2023).
26. Li, T. et al. Cancergpt for few shot drug pair synergy prediction using large pretrained language models. *npj Digital Med.* **7**, 40 (2024).
27. Luecken, M. D. et al. Benchmarking atlas-level data integration in single-cell genomics. *Nat. Methods* **19**, 41–50 (2022).
28. Zhang, Y. & Yang, Q. A survey on multi-task learning. *IEEE Trans. Knowl. Data Eng.* **34**, 5586–5609 (2021).
29. Brown, T. et al. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **33**, 1877–1901 (2020).
30. Achiam, J. et al. Gpt-4 technical report. Preprint at https://arxiv.org/abs/2303.08774 (2023).
31. Liu, T., Chen, T., Zheng, W., Luo, X. & Zhao, H. scelmo: embeddings from language models are good learners for single-cell data analysis. Preprint at *bioRxiv* https://doi.org/10.1101/2023.12.07.569910 (2023).
32. Bai, G. et al. Beyond efficiency: a systematic survey of resource-efficient large language models. Preprint at *CoRR* https://arxiv.org/abs/2401.00625 (2024).
33. Yang, Z., Jin, Y. & Xu, X. Hades: hardware accelerated decoding for efficient speculation in large language models. Preprint at https://arxiv.org/abs/2401.00625 (2024).
34. Anthropic. The Claude 3 Model Family: Opus, Sonnet, Haiku. Preprint at https://api.semanticscholar.org/CorpusID:268232499 (2024).
35. Team, G. et al. Gemini: a family of highly capable multimodal models. Preprint at https://arxiv.org/abs/2312.11805 (2023).
36. McInnes, L., Healy, J. & Melville, J. Umap: uniform manifold approximation and projection for dimension reduction. Preprint at https://arxiv.org/abs/1802.03426 (2018).
37. Knox, C. et al. Drugbank 6.0: the drugbank knowledgebase for 2024. *Nucleic Acids Res.* **52**, 1265–1275 (2023).
38. Schoch, C. L. et al. Ncbi taxonomy: a comprehensive update on curation, resources and tools. *Database* **2020**, 062 (2020).
39. Pedregosa, F. et al. Scikit-learn: machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011).
40. Luecken, M. D. & Theis, F. J. Current best practices in single-cell RNA-seq analysis: a tutorial. *Mol. Syst. Biol.* **15**, 8746 (2019).
41. Landrum, G. Rdkit documentation. *Release* **1**, 4 (2013).
42. Landrum, G. et al. rdkit/rdkit: 2025_03_2 (Q1 2025) Release. (Release_2025_03_2). *Zenodo* https://doi.org/10.5281/zenodo.15286010 (2025).
43. Weininger, D. Smiles, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* **28**, 31–36 (1988).
44. Burkhardt, D. et al. Open problems - single-cell perturbations. Kaggle (2023).
45. Malyutina, A. et al. Drug combination sensitivity scoring facilitates the discovery of synergistic and efficacious drug combinations in cancer. *PLoS Comput. Biol.* **15**, 1006752 (2019).
46. Baptista, D., Ferreira, P. G. & Rocha, M. A systematic evaluation of deep learning methods for the prediction of drug synergy in cancer. *PLoS Comput. Biol.* **19**, 1010200 (2023).
47. Suzgun, M. & Kalai, A. T. Meta-prompting: Enhancing language models with task-agnostic scaffolding. Preprint at https://arxiv.org/abs/2401.12954 (2024).
48. Wei, J. et al. Chain-of-thought prompting elicits reasoning in large language models. *Adv. Neural Inf. Process. Syst.* **35**, 24824–24837 (2022).
49. Cortes, C. & Vapnik, V. Support-vector networks. *Mach. Learn.* **20**, 273–297 (1995).
50. Arik, S. Ö. & Pfister, T. Tabnet: attentive interpretable tabular learning. In *Proc. AAAI Conference on Artificial Intelligence*, Vol. 35 6679–6687 (the Association for the Advancement of Artificial Intelligence, 2021).
51. Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proc. 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)* (eds Burstein, J., Doran, C. & Solorio, T.) 4171–4186 (Association for Computational Linguistics, Minneapolis, MN, 2019).
52. Tibshirani, R. Regression shrinkage and selection via the lasso. *J. R. Stat. Soc. Ser. B Stat. Methodol.* **58**, 267–288 (1996).
53. Lundberg, S. M. & Lee, S.-I. A unified approach to interpreting model predictions. *Adv. Neural Inf. Process. Syst.* **30**, 4768–4777 (2017).
54. Barretina, J. et al. The cancer cell line encyclopedia enables predictive modelling of anticancer drug sensitivity. *Nature* **483**, 603–607 (2012).
55. Blatkiewicz, M., Białas, P., Taryma-Leśniak, O. & Hukowska-Szematowicz, B. Pan-cancer analysis of VIM expression in human cancer tissues. Preprint at *Research Square* https://doi.org/10.21203/rs.3.rs-646169/v1 (2021).
56. Muzellec, B., Teleńczuk, M., Cabeli, V. & Andreux, M. Pydeseq2: a Python package for bulk RNA-seq differential expression analysis. *Bioinformatics* **39**, 547 (2023).
57. Love, M. I., Huber, W. & Anders, S. Moderated estimation of fold change and dispersion for RNA-seq data with deseq2. *Genome Biol.* **15**, 1–21 (2014).
58. Lee, S. Y. et al. Bmp2 increases adipogenic differentiation in the presence of dexamethasone, which is inhibited by the treatment of tnf-α in human adipose tissue-derived stromal cells. *Cell. Physiol. Biochem.* **34**, 1339–1350 (2014).
59. Parveen, S., Ashfaq, H., Shahid, M., Kanwal, A. & Tayyeb, A. Emerging therapeutic role of cdk inhibitors in targeting cancer stem cells. *J. Biomed. Res.* **2766**, 2276 (2021).
60. Thomas, P. D. et al. Panther: making genome-scale phylogenetics accessible to all. *Protein Sci.* **31**, 8–22 (2022).
61. Liberzon, A. et al. The molecular signatures database hallmark gene set collection. *Cell Syst.* **1**, 417–425 (2015).
62. Virtanen, P. et al. Scipy 1.0: fundamental algorithms for scientific computing in Python. *Nat. methods* **17**, 261–272 (2020).
63. Traag, V. A., Waltman, L. & Van Eck, N. J. From Louvain to Leiden: guaranteeing well-connected communities. *Sci. Rep.* **9**, 5233 (2019).
64. Loewe, S. The problem of synergism and antagonism of combined drugs. *Arzneim. Forsch.* **3**, 285–290 (1953).
65. Yadav, B., Wennerberg, K., Aittokallio, T. & Tang, J. Searching for drug synergy in complex dose–response landscapes using an interaction potency model. *Comput. Struct. Biotechnol. J.* **13**, 504–513 (2015).

66. Me, B. What is synergy. *Pharmacol. Rev.* **41**, 93–141 (1989).
67. Bliss, C. I. The toxicity of poisons applied jointly 1. *Ann. Appl. Biol.* **26**, 585–615 (1939).
68. Cipolla, R., Gal, Y. & Kendall, A. Multi-task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 7482–7491) (IEEE Computer Society, 2018).
69. Lin, B. & Zhang, Y. Libmtl: a Python library for deep multi-task learning. *J. Mach. Learn. Res.* **24**, 18 (2023).
70. Liu, H. et al. Drugcombdb: a comprehensive database of drug combinations toward the discovery of combinatorial therapy. *Nucleic Acids Res.* **48**, 871–881 (2020).
71. Gal, Y. & Ghahramani, Z. Dropout as a Bayesian approximation: representing model uncertainty in deep learning. In *Proc. 33rd International Conference on Machine Learning. Proceedings of Machine Learning Research*, Vol. 48 (eds Balcan, M.F. & Weinberger, K.Q.) 1050–1059 (PMLR, New York, New York, USA, 2016).
72. Lemay, A. et al. Improving the repeatability of deep learning models with Monte Carlo dropout. *npj Dig. Med.* **5**, 174 (2022).
73. Michael, K. et al. Url: A representation learning benchmark for transferable uncertainty estimates. *Advances in Neural Information Processing Systems* **36**, 13956–13980 (2023).
74. Sigma-Aldrich's I-BET. https://www.emdmillipore.com/US/en/product/I-BET-CAS-1260907-17-2-Calbiochem,EMD_BIO-401010. Accessed: 2024-01-17.
75. Selleck's I-BET151 (GSK1210151A). https://www.selleckchem.com/products/i-bet151-gsk1210151a.html. Accessed: 2024-01-17.
76. Selleck's PF-562271. https://www.selleckchem.com/products/pf-562271.html. Accessed: 2024-01-29.
77. Selleck's Saracatinib. https://www.selleckchem.com/products/AZD0530.html. Accessed: 2024-01-29.
78. Saatci, O. et al. Targeting lysyl oxidase (LOX) overcomes chemotherapy resistance in triple negative breast cancer. *Nat. Commun.* **11**, 2416 (2020).
79. New embedding models and API updates. https://openai.com/blog/new-embedding-models-and-api-updates. Accessed: 2024-01-27.
80. Yu, T. et al. Gradient surgery for multi-task learning. In *Advances in Neural Information Processing Systems*, (eds Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F. & Lin, H.) Vol. 33, 5824–5836 (Neural Information Processing Systems Foundation, Inc., 2020).
81. Wang, Z., Tsvetkov, Y., Firat, O. & Cao, Y. Gradient vaccine: investigating and improving multi-task optimization in massively multilingual models. In *International Conference on Learning Representations* https://openreview.net/forum?id=F1vEjWK-lH_ (2021).
82. Liu, B., Liu, X., Jin, X., Stone, P. & Liu, Q. Conflict-averse gradient descent for multi-task learning. *Adv. Neural Inf. Process. Syst.* **34**, 18878–18890 (2021).
83. Navon, A. et al. Multi-Task Learning as a Bargaining Game. In *International Conference on Machine Learning*, pp. 16428–16446 (PMLR, 2022).
84. Kim, S. & Xing, E. P. Tree-guided group lasso for multi-response regression with structured sparsity, with an application to eQTL mapping. *Ann. Appl. Stat.* **6**, 1095–1117 (2012).
85. Frantar, E., Ruiz, C.R., Houlsby, N., Alistarh, D. & Evci, U. Scaling laws for sparsely-connected foundation models. In *The Twelfth International Conference on Learning Representations* https://openreview.net/forum?id=i9K2ZWkYIP (2024).
86. Kaplan, J. et al. Scaling laws for neural language models. Preprint at https://arxiv.org/abs/2001.08361 (2020).
87. Heumos, L. et al. Best practices for single-cell analysis across modalities. *Nat. Rev. Genet.* **24**, 550–572 (2023).
88. Sun, N. & Zhao, H. Statistical methods in genome-wide association studies. *Annu. Rev. Biomed. Data Sci.* **3**, 265–288 (2020).
89. Paszke, A. et al. PyTorch: an imperative style, high-performance deep learning library. *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Article 721, 8026–8037 (Curran Associates Inc., Red Hook, NY, USA, 2019).
90. Awais, M. et al. Foundational models defining a new era in vision: a survey and outlook. Preprint at https://arxiv.org/abs/2307.13721 (2023).
91. Campos Zabala, F. J. Neural networks, deep learning, foundational models. In *Grow Your Business with AI: A First Principles Approach for Scaling Artificial Intelligence in the Enterprise* 245–275 (Apress, Berkeley, CA, 2023).
92. Kingma, D. P. & Ba, J. Adam: a method for stochastic optimization. In 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings (eds Bengio, Y. & LeCun, Y.) http://arxiv.org/abs/1412.6980 (2015).
93. Wolf, F. A., Angerer, P. & Theis, F. J. Scanpy: large-scale single-cell gene expression data analysis. *Genome Biol.* **19**, 1–5 (2018).
94. O'Neil, J. et al. An unbiased oncology compound screen to identify novel combination strategies. *Mol. Cancer Ther.* **15**, 1155–1162 (2016).
95. Liu, T. Building a unified model for drug synergy analysis powered by large language models. HelloWorldLTY/BAITSAO. *Zenodo* https://doi.org/10.5281/zenodo.15105816 (2025).
96. Zhao, Y. & He, L. Deep learning in the EEG diagnosis of Alzheimer's disease. In *Computer Vision-ACCV 2014 Workshops: Singapore, Singapore, November 1-2, 2014, Revised Selected Papers, Part I 12* (pp. 340–353) (Springer International Publishing, Singapore, 2015).

Acknowledgements

We thank Yijia Xiao for the fruitful discussion about training the model. We thank Wangjie Zheng and Tianqi Chen for helpful discussions about designing drug embeddings. We thank Dr. Ning Sun, Xinyi Chen, Dr. Gefei Wang, Dr. Yingxin Lin, and Chenyu Wang for the suggestions about approaches to improve the experimental design. We also thank Jiawei He for helping us prepare the molecular file. This project was partly supported by NIH grants U24HG012108 and P50 CA196530 awarded to H.Z., and the OpenAI Researcher Access Program and Google Cloud Research Program awarded to T.L.

Author contributions

T.L. proposed this study. T.L., T.C., and X.L. designed the model. T.L. ran all the experiments. T.L., X.L., and H.Z. wrote the manuscript. H.Z. supervised this study.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41467-025-59822-y.

Correspondence and requests for materials should be addressed to Hongyu Zhao.

Peer review information *Nature Communications* thanks Pengtao Xie and the other anonymous reviewer(s) for their contribution to the peer review of this work. A peer review file is available.

Reprints and permissions information is available at http://www.nature.com/reprints