npj | precision oncology

Article

Published in partnership with The Hormel Institute, University of Minnesota

https://doi.org/10.1038/s41698-024-00700-z

Exploiting common patterns in diverse cancer types via multi-task learning

Check for updates

Bo-Run Wu[1], Sofia Ormazabal Arriagada[1,2,3], Te-Cheng Hsu[4], Tsung-Wei Lin[1] & Che Lin[1,5,6,7,8,9,10]

Cancer prognosis requires precision to identify high-risk patients and improve survival outcomes. Conventional methods struggle with the complexity of genetic biomarkers and diverse medical data. Our study uses deep learning to distil high-dimensional medical data into low-dimensional feature vectors exploring shared patterns across cancer types. We developed a multi-task bimodal neural network integrating RNA Sequencing and clinical data from three The Cancer Genome Atlas project datasets: Breast Invasive Carcinoma, Lung Adenocarcinoma, and Colon Adenocarcinoma. Our approach significantly improved prognosis prediction, especially for Colon Adenocarcinoma, with up to 26% increase in concordance index and 41% in the area under the precision-recall curve. External validation with Small Cell Lung Cancer achieved comparable metrics, indicating that supplementing small datasets with data from other cancers can improve performance. This work represents initial strides in using multi-task learning for prognosis prediction across cancer types, potentially revealing shared mechanisms among cancers and contributing to future applications in precision medicine.

Cancer is a leading cause of death globally, underlining the importance of early detection for improved survival rates[1,2]. Accurate prognosis predictions, aided by data science and deep learning, can assist in treating high-risk patients. However, the high dimensionality of omics data presents challenges such as overfitting, especially when dealing with high-dimensional data with insufficient samples, a situation known as the "curse of dimensionality"[3].

Additionally, gathering enough patient medical data on the same cancer type remains challenging when training robust Deep neural networks (DNNs). DNNs have proven to be an effective tool for precisely diagnosing diseases using medical data[4–6]. However, DNNs trained on limited samples can suffer from overfitting, so manually labeling sufficient training samples is not a practical solution. To address this, previous studies[7,8] have integrated multiple datasets of the same cancer type into a single cohort dataset to augment the volume of labeled data for training and testing. Studies typically focus on a single cancer type due to different cancers' unique genotypes and phenotypes. Yet, many medical experts maintain that shared underlying mechanisms exist among various cancers[9–11]. Despite different cancer types sharing certain commonalities, they each possess distinct characteristics and are typically treated separately in the medical field. Yet, a naive data combination strategy may negatively affect prediction performance. Thus, models are usually developed for each specific cancer type.

This study addresses the challenge of effectively integrating data from various cancers by using multi-task learning (MTL), mitigating problems caused by high dimensionality and small sample sizes. This learning paradigm mitigates data sparsity issues in scenarios where each task has limited labeled data[12]. MTL uses shared structural knowledge across multiple tasks through inductive bias and has shown promise in various fields, including natural language processing[13–15], computer vision[16–19], and bioinformatics[20–22]. In this study, we consider each cancer type as a separate task. Using MTL offers two main benefits: first, it reduces the number of parameters needed across tasks through parameter sharing; second, it enhances data efficiency by incorporating more diverse medical data, which is challenging to label and procure for a single cancer type.

We applied dimension reduction techniques to tackle the challenge of excessive dimensionality in data generated via high-throughput technologies[23]. We used a hybrid ensemble systems biology feature selector (SBFS) to extract representative features with biological insights[7,24]. This selector combines data and function perturbation, considering the biological

[1]Graduate Institute of Communication Engineering, National Taiwan University (NTU), Taipei, Taiwan. [2]Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan. [3]Taiwan International Graduate Program in Artificial Intelligence of Things, NTU, Taipei, Taiwan. [4]Institute of Communications Engineering, National Tsing Hua University, Hsinchu, Taiwan. [5]Department of Electrical Engineering, NTU, Taipei, Taiwan. [6]Center for Advanced Computing and Imaging in Biomedicine, NTU, Taipei, Taiwan. [7]Smart Medicine and Health Informatics Program, NTU, Taipei, Taiwan. [8]School of Medicine, NTU, Taipei, Taiwan. [9]Center for Biotechnology, NTU, Taipei, Taiwan. [10]Computer and Information Networking Center of Electrical Engineering, NTU, Taipei, Taiwan. e-mail: soa2100@gmail.com; chelin@ntu.edu.tw

THE HORMEL INSTITUTE
UNIVERSITY OF MINNESOTA

relevance and interactions between genes to select the most salient features (Fig. 1). In other words, it is an unsupervised feature selector that uses the interaction networks between genes to rank them according to their importance.

We hypothesize that shared, universal representations of information across cancer types can improve the performance of cancer prognosis prediction models. As a case study, we used data from three primary datasets from the TCGA project[25], focusing on breast invasive carcinoma (BRCA), lung adenocarcinoma (LUAD), and colorectal adenocarcinoma (COAD), due to their prevalence in the USA and Taiwan.

Consequently, we used a multi-task bimodal deep learning model, capable of learning across multiple related cancer type features by integrating two different data types: genomic data from RNA-Seq and patient clinical data. To manage the high dimensionality of these data, we applied an SBFS[24], which maps high-dimensional data into a lower-dimensional space, distilling the most meaningful features for enhanced prognosis predictions.

Cancer prognosis prediction was formulated as a binary classification problem with a five-year outcome window starting from the diagnosis date, a time frame often used to stratify patients in clinical settings. This period was chosen for its clear interpretability and because it is sufficiently long to accurately monitor patients' statuses, yet not so extended as to result in an excessive number of censored cases. Patients who died within the outcome window were labeled as having poor cancer prognoses (1), while others were considered to have good cancer prognoses (0) (Fig. 2). Due to the TCGA project's focus, specific survival times for living patients are unavailable[26], so all alive patients were treated as good cancer prognostic labels. In summary, we collected 1093, 510, and 454 patients for BRCA, LUAD, and COAD, respectively, with label imbalance rates (representing poor prognosis labels) of 9.241%, 33.333%, and 20.485%. The TNM stage distribution for our cancer datasets, shown in Supplementary Section 1, leans towards early-stage cancers, often indicative of better prognoses.

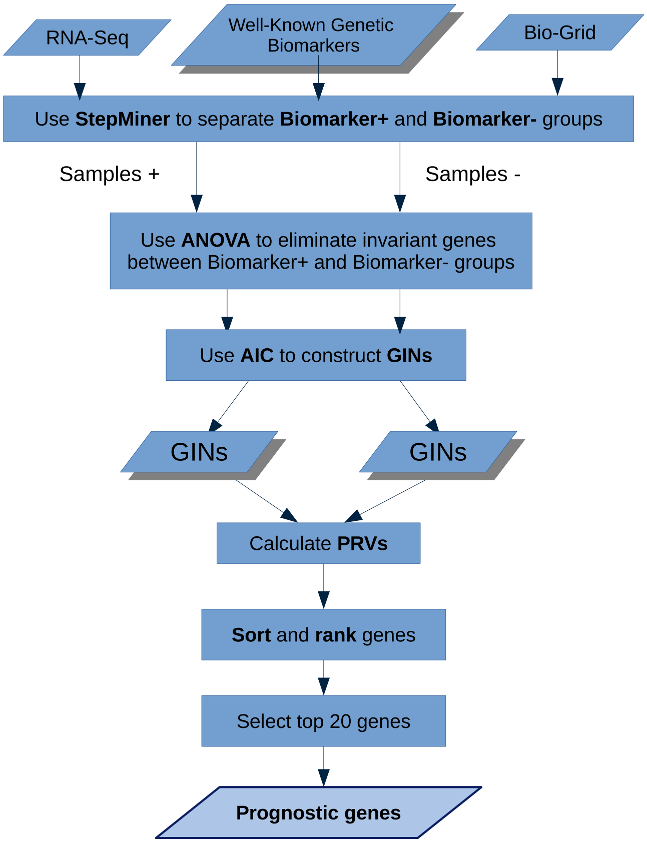
RNA-Seq
Well-Known Genetic Biomarkers
Bio-Grid
Use **StepMiner** to separate **Biomarker+** and **Biomarker-** groups
Samples +
Samples -
Use **ANOVA** to eliminate invariant genes between Biomarker+ and Biomarker- groups
Use **AIC** to construct **GINs**
GINs
GINs
Calculate **PRVs**
Sort and **rank** genes
Select top 20 genes
Prognostic genes

Fig. 1 | Workflow of the systems biology feature selector for prognostic gene identification. Systems biology feature selector pipeline used to choose the relevant prognostic genes that will be later used as input for the RNA-Seq feature extractor. The well-known biomarkers are shown in Table 6.

For external validation, we used the University of Cologne's 2015 Small Cell Lung Cancer (SCLC) study[27] obtained from cBioPortal[28–30]. The imbalance rate for this set is 81.481% after filtering out samples with missing data and patients who did not follow up with the check-ups. The demographic information for all datasets can be found in Table 1.

Previous models addressing patient stratification based on survival outcomes include DeepProg[31], DeepSurv[32], and various Cox-proportional hazard-derived models. Huang et al.'s study[33] compared Cox-based models like Cox-nnet[34], DeepSurv, and AECOX[33] across twelve cancer types. They found that Cox-nnet, the simplest model, performed better regarding the concordance index and log-rank test p value. However, it was noted that these models are sensitive to the variability in genomic and clinical profiles across different cancers. In contrast, DeepProg is an ensemble framework that integrates deep learning and more traditional machine learning approaches to predict patient survival groups using a pan-cancer strategy. However, its reliance on a boosting strategy detracts from the model's interpretability, making it harder to discern which biological features influence the predictions the most.

Due to the difficulties in obtaining well-annotated and integrating different genomic data types, along with understanding the contribution of each modality to our predictions, our study focused only on RNA-Seq data. We adopted an MTL approach to address the heterogeneous data issue, enabling our model to take advantage of these variations and effectively distinguish between various cancer types and risk groups. Although we aim to integrate data from different cancers and modalities, we avoided ensemble models for greater explainability. Our primary goal is to develop a model capable of providing reliable predictions even with limited sample sizes. Furthermore, our focus is to surpass the limitations presented by data scarcity and offer a reliable way of discriminating between high and low-risk patients.

In this study, we make two key contributions. First, we used MTL to exploit data from diverse cancer types for cancer prognosis prediction. This led to significant improvements in several evaluation metrics compared to single-task learning (STL), primarily on the smaller datasets. Second, we show our model architecture's ability to handle multiple tasks and modalities effectively. Our model can generalize representations across all cancer types in a model-independent manner and overcome common challenges associated with MTL.

Results

Overview

We conducted multiple experiments to evaluate the effectiveness of MTL by comparing MTL with STL using a bimodal neural network. The data was preprocessed using the same pipeline for both learning paradigms. Four models were used in the STL experiments: logistic regression (LR), random forest (RF), support vector machine (SVM), and an STL bimodal neural

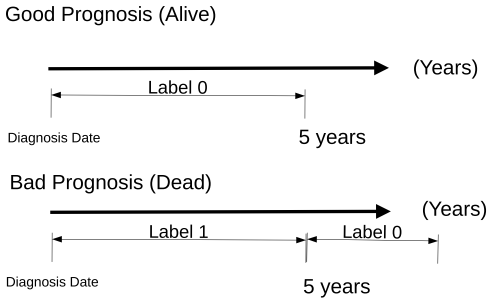

Fig. 2 | Visual representation of patient prognosis based on 5-year survival. The top timeline represents patients with a good prognosis, indicated by a longer survival span, while the bottom represents patients with a bad prognosis, marked by a shorter survival duration and a terminal event at year 5. Patients who survive this 5-year mark are labeled with "0", while patients who die before five years are labeled with "1".

Table 1 | Summary of the patients' demographic attributes and distribution in training and test sets for all cancers, including the external validation set SCLC (test only)

Attributes		BRCA		LUAD		COAD		SCLC
		Train	Test	Train	Test	Train	Test	Test
Samples		874	219	408	102	363	91	81
Median age (years)		59.24	58.18	66.87	66.12	69.29	65.88	65
Median birth year		1950	1952	1942	1943	1939	1941	1950
Median diagnosis year		2009	2009	2010	2010	2009	2009	2015
Gender	Female	862	219	214	60	164	50	25
	Male	12	0	194	42	199	41	56
Race	White	605	148	311	79	168	44	34
	Black or African American	145	38	40	12	45	14	0
	Asian	49	12	6	2	9	2	8
	American Indian or Alaska Native	1	0	0	1	0	1	0
	Not reported	74	21	51	8	141	30	39
Ethnicity	Not Hispanic or Latino	705	175	304	81	210	59	42
	Hispanic or Latino	31	8	5	2	4	0	0
	Not reported	138	36	99	19	149	32	39
Label	Good cancer prognosis	793	199	272	68	289	72	10
	Poor cancer prognosis	81	20	136	34	74	19	71

The validation set corresponded to a stratified 25% of the training set.

Table 2 | Summary of the single-task learning model results using all TCGA cancer datasets

Single-task learning				
Datasets	Models	AUPRC	AUROC	C-index
BRCA	LR	0.292 ± 0.090	0.709 ± 0.066	0.699 ± 0.064
	RF	0.353 ± 0.101	0.741 ± 0.065	0.730 ± 0.061
	SVM	0.356 ± 0.100	0.712 ± 0.075	0.698 ± 0.072
	BNN	**0.380** ± **0.097**	**0.796** ± **0.050**	**0.783** ± **0.050**
LUAD	LR	0.502 ± 0.080	0.646 ± 0.058	0.595 ± 0.050
	RF	0.526 ± 0.083	**0.686** ± **0.056**	**0.638** ± **0.050**
	SVM	**0.532** ± **0.081**	0.617 ± 0.061	0.573 ± 0.051
	BNN	0.498 ± 0.083	0.629 ± 0.062	0.574 ± 0.051
COAD	LR	0.408 ± 0.103	0.641 ± 0.077	0.632 ± 0.078
	RF	**0.432** ± **0.104**	**0.650** ± **0.078**	**0.641** ± **0.074**
	SVM	0.421 ± 0.106	0.589 ± 0.084	0.588 ± 0.079
	BNN	0.353 ± 0.097	0.554 ± 0.079	0.554 ± 0.078

network (BNN). Three evaluation metrics were used: Area Under the Receiver Operating Curve (AUROC), Area Under the Precision-Recall Curve (AUPRC), and C-index. Due to the clinical annotations in the TCGA project, the C-index was used only for pairs of deceased patients with an exact date of death (to calculate an accurate survival time). Additionally, we conducted an external validation for the STL and MTL models using the SCLC dataset. This set was used to test models already trained on TCGA data.

Single-task learning

Table 2 shows the results of STL, with early fusion on two modalities for LR, RF, and SVM and intermediate fusion for the STL BNN. The STL BNN performed the best in BRCA, whereas RF performed the best in LUAD and COAD, except for AUPRC in LUAD. No single model consistently outperformed all other models for all three cancer types under STL. This experiment served as a benchmark for MTL.

Table 3 | Results of using STL and MTL on bimodal neural networks using all cancer datasets

Single-task learning vs multi-task learning on bimodal neural network				
Datasets	Learning paradigms	AUPRC	AUROC	C-index
BRCA	STL	**0.380** ± **0.097**	0.796 ± 0.050	0.783 ± 0.050
	MTL	0.348 ± 0.090	**0.839** ± **0.044**	**0.823** ± **0.043**
LUAD	STL	0.498 ± 0.083	0.629 ± 0.062	0.574 ± 0.051
	MTL	**0.509** ± **0.082**	**0.645** ± **0.060**	**0.587** ± **0.049**
COAD	STL	0.353 ± 0.097	0.554 ± 0.079	0.554 ± 0.078
	MTL	**0.498** ± **0.102**	**0.712** ± **0.073**	**0.696** ± **0.067**

BRCA, LUAD, and COAD correspond to the TCGA sets.

MTL on bimodal neural network

Table 3 and Fig. 3 compare single-task and MTL in the bimodal neural network. Significant performance improvements were observed in COAD, with AUROC, AUPRC, and C-index increases of 29%, 41%, and 26%. BRCA and LUAD improved slightly, with the AUROC and C-indexes increasing 5% in BRCA and 2% in LUAD. Despite a drop of 8% in AUPRC for BRCA, MTL outperformed single-task for the AUROC and C-index in all cancer types, especially for COAD, with fewer available patients.

External validation

We anticipated that the external validation set's performance would lag behind the TCGA datasets, primarily due to its shorter survival time distribution and a higher proportion of later-stage patients than the TCGA datasets (Fig. 4). Given SCLC's small sample size (81 samples), we trained SCLC on only 64 samples and later tested it on the remaining 17. STL-BRCA, STL-LUAD, and STL-COAD, trained exclusively on one of the TCGA datasets (BRCA, LUAD, or COAD) and subsequently tested on the SCLC set, demonstrated limited generalizability. STL-SCLC, an STL model trained on 80% of the SCLC data and tested on the remaining 20%, showed the worst AUPRC among the STL models, probably due to the limited

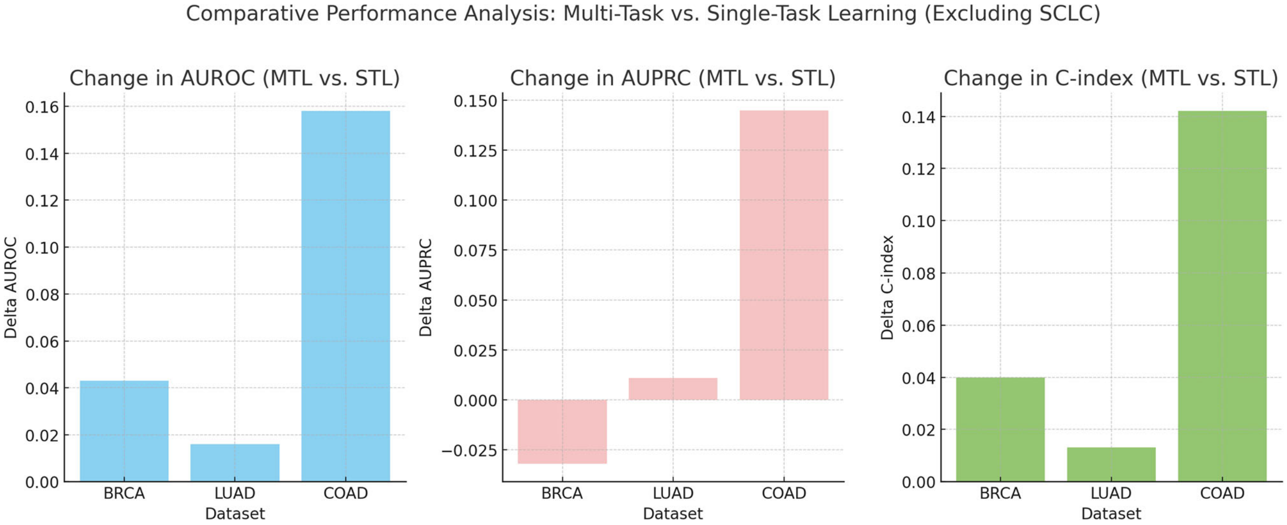

Fig. 3 | Comparative performance analysis: multi-task vs. single-task learning. Each subplot represents the performance changes when switching from STL to MTL for each cancer dataset. The delta for each metric is defined as the MTL metric minus the STL metric.

samples used for training. The high AUPRC across all models is due to the large number of patients with poor prognoses at the 5-year mark (Table 4).

Ablation studies

We conducted ablation studies to verify the effectiveness of techniques used in MTL. These four studies focused on the order of RNA-Seq data used for the feature extractor, RNA-Seq feature extractor, task descriptor, and weighted random data sampler, and are summarized in Table 5.

1. **Without ordered RNA-Seq data:** We explored the impact of input sequence order on prediction accuracy on the classifier (Fig. 5) and RNA-Seq feature extractor (Fig. 6) using RNA-Seq data arranged in different order. We used alphabetical order for the unordered data, sorting the gene's names in ascending order. Yet, using ordered (according to PRVs) and unordered RNA-Seq data as input for the feature extractor showed no significant differences, implying the order of RNA-Seq data might be less effective than initially thought.
2. **Unique RNA-Seq feature extractor without parameter sharing:** Comparing distinct parameter sharing showed that using a unique RNA-Seq feature extractor caused decreases of between 11% and 14% for all COAD metrics. In addition to drops in performance of around 3% for AUROC and C-index in BRCA and 1% for AUPRC in LUAD, as shown in Table 5. We only shared parameters in the classifier and used a unique RNA-Seq feature extractor per cancer type.
3. **Without the task descriptor:** Removing the classifier's task descriptor resulted in drops between 5% to 7% in BRCA and 5% to 12% in COAD for all metrics (Table 5). For LUAD, AUROC and C-index decreased by 1% to 3%, and AUROC increased by 2%. By deleting the task descriptor, the classifier had the same structure as the one used for the single-task bimodal network.
4. **Without a weighted random data sampler:** Using a naive random data sampler instead of a weighted random one resulted in drops of 1% to 10% in BRCA and 11% to 15% in COAD for all metrics (Table 5). For LUAD, all metrics rose between 2% to 6%. Unlike the naive sampler, which samples training data equally across all tasks, the weighted random sampler balanced the training data, using a variety of patients with different cancers throughout the training process.

Further analyses

We used **SHAP (SHapley Additive exPlanations) values**[35] to examine the strength of a particular genomic feature's effects over our predictions. We calculated these values using the Captum[36] framework. Captum approximates values by computing gradient expectations, which is achieved by random sampling from a distribution of baselines. These baselines serve as non-informative inputs and typically lack predictive significance.

The magnitude of SHAP values indicates the strength of a gene's impact on risk prediction, with the direction (positive or negative) representing a risk-increasing or protective effect, respectively. Our main observation from comparing SHAP values for STL and MTL is that STL relies more heavily on a reduced pool of specific genes. At the same time, MTL bases its predictions on a more uniform and widespread gene selection for LUAD. This is patent in Fig. 7b. Additionally, the variation in values and range between STL and MTL frameworks is more pronounced for LUAD than BRCA or COAD. For a deeper dive into the variation and correlation of feature importance across folds, please consult Supplementary Section 6.

We calculated the Kaplan–Meier (KM) survival curves[37], applying a separate Kaplan–Meier fitter for each bootstrap in the test set (Fig. 8). We computed the average survival probabilities and log-rank p-values per time point. In almost all cases, there was a significant distinction between poor and good prognosis groups using the threshold $\alpha = 0.05$. The exception was STL on COAD, though its p value was close to the threshold ($p = 0.05659$).

Discussion

Our findings indicate that the MTL model benefits significantly from training on larger, more diverse datasets, even when the data are sourced from different cancers. This is evidenced by improved performance across most metrics when training on only 60% of the total data, compared to 80%. This improvement is especially prevalent when contrasting the performance in the cross-validation (Supplementary Section 2) and the bootstrap test (Table 3.) This is particularly crucial for cancers with fewer available patients, such as COAD, where MTL showed marked improvements in all metrics.

The SHAP value analysis suggests that MTL captures a broader range of genomic features than STL. Moreover, gene influences on prognosis are more substantial and variable in LUAD, with greater variance between STL and MTL results than in the other two cancers. The KM survival curves reveal that, while the STL model successfully distinguished between good and bad prognosis groups for BRCA and LUAD, it failed to do so for COAD. This might be attributed to the fact that the STL BNN significantly underperformed in COAD, indicating potential overfitting. In contrast, all classes predicted by the MTL model were significantly different.

The ablation studies reinforce the importance of sharing RNA-Seq feature extractors to benefit performance, suggesting common RNA-Seq

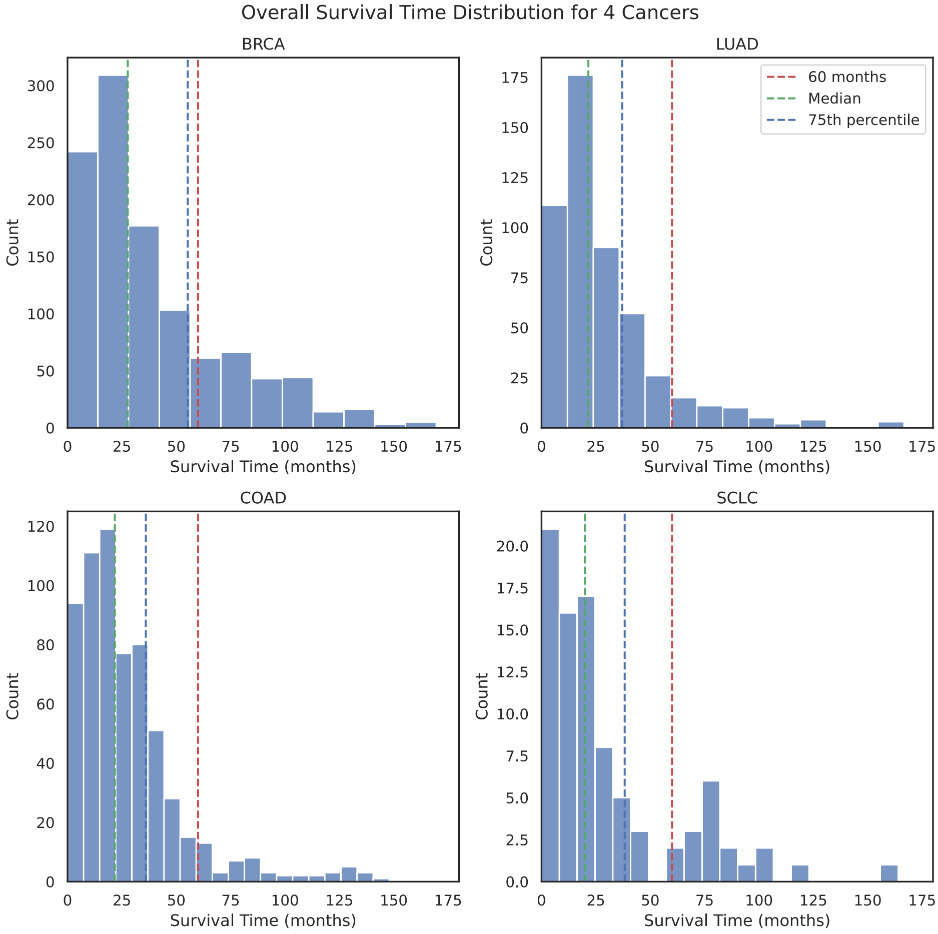

Fig. 4 | Overall survival time distribution for 4 cancer types (BRCA, LUAD, COAD, SCLC). The histograms show the count of patients across different survival times (in months) for each cancer type: BRCA, LUAD, COAD, and SCLC. The vertical dashed lines represent important survival time statistics: green for the median, blue for the 75th percentile, and red for the cutoff at 60 months.

Table 4 | Performance of STL and MTL models using SCLC as the testing set

Performance of STL and MTL learning paradigms on different datasets					
Training dataset	**Testing set**	**Learning paradigm**	**AUPRC**	**AUROC**	**C-index**
BRCA	SCLC	STL	0.84790 ± 0.05356	0.54930 ± 0.07627	0.57398 ± 0.03459
LUAD	SCLC	STL	0.86949 ± 0.04360	**0.57273 ± 0.07461**	0.48825 ± 0.04495
COAD	SCLC	STL	0.86785 ± 0.04009	0.54391 ± 0.07167	0.49265 ± 0.03709
SCLC	SCLC	STL	0.67977 ± 0.13818	0.50554 ± 0.17836	0.54630 ± 0.12084
LUAD, SCLC	SCLC	MTL (2 cancers)	0.88008 ± 0.09716	0.56364 ± 0.27581	0.55417 ± 0.12947
BRCA, LUAD, COAD	SCLC	MTL (3 cancers)	0.83564 ± 0.05517	0.5000 ± 0.08374	0.49333 ± 0.03977
BRCA, LUAD, COAD, SCLC	SCLC	MTL (4 cancers)	**0.89121 ± 0.08435**	0.56962 ± 0.26270	**0.58991 ± 0.10259**

STL-BRCA, STL-LUAD, and STL-COAD refer to single-task learning models trained exclusively on one of the TCGA datasets (BRCA, LUAD, or COAD) with their respective selected genes and subsequently tested on the SCLC set. For instance, STL-BRCA indicates a model trained on BRCA data and later tested on SCLC data. STL-SCLC corresponds to an STL model trained on 80% of the SCLC data and tested on the remaining 20%. "MTL (2 cancers)" refers to a multi-task learning model trained on LUAD and SCLC data. "MTL (3 cancers)" represents a model trained on TCGA data and tested on the entire SCLC dataset. "MTL (4 cancers)" denotes a model trained on BRCA, LUAD, COAD, and SCLC simultaneously. For the STL-SCLC and MTL models, the SCLC dataset was split in the same proportion as the TCGA datasets for training and testing, and these models were evaluated on an unseen testing set. Models that did not use SCLC for training were tested on the whole SCLC dataset.

expression patterns among the three cancer types. Additionally, providing task information to the classifier improved overall model performance, as shown by the decrease in AUROC when removing the task descriptor. Furthermore, the results from using an unweighted random sampler imply that the ratio of data from different tasks significantly influences performance. This suggests that further exploration of data sampling techniques is needed. Nonetheless, the effect on LUAD was minimal due to the relatively similar number of training samples with and without the weighted random sampler.

The external validation results indicate that MTL models trained on larger, more diverse cancer datasets offer better generalization and more stable performance in SCLC. However, the MTL model trained solely on TCGA data did not perform satisfactorily when directly applied to the SCLC dataset without further retraining. Thus, updating and refining the model with new data, as seen with the MTL (4 cancers) model, is crucial for improving its generalizability and applicability to specific cancer types. Additionally, focused research on the unique characteristics of SCLC, such as its aggressiveness and

Table 5 | Summary of the ablation studies conducted on the MTL bimodal neural network model

Ablation studies on multi-task bimodal neural network				
Datasets	**Modifications**	**AUPRC**	**AUROC**	**C-index**
BRCA	Original	0.348 ± 0.090	**0.839 ± 0.044**	**0.823 ± 0.043**
	Without ordered RNA-seq data	0.353 ± 0.092	0.831 ± 0.045	0.813 ± 0.045
	Unique RNA-seq feature extractor	**0.465 ± 0.111**	0.811 ± 0.052	0.793 ± 0.051
	Without task descriptor	0.277 ± 0.075	0.790 ± 0.051	0.773 ± 0.050
	Without weighted random sampler	0.333 ± 0.099	0.736 ± 0.065	0.721 ± 0.064
LUAD	Original	0.509 ± 0.082	0.645 ± 0.060	0.587 ± 0.049
	Without ordered RNA-seq data	0.532 ± 0.081	0.650 ± 0.059	0.596 ± 0.048
	Unique RNA-seq feature extractor	0.499 ± 0.079	0.643 ± 0.062	0.595 ± 0.053
	Without task descriptor	0.537 ± 0.083	0.614 ± 0.066	0.570 ± 0.055
	Without weighted random sampler	**0.566 ± 0.084**	**0.677 ± 0.062**	**0.614 ± 0.053**
COAD	Original	**0.498 ± 0.102**	**0.712 ± 0.073**	**0.696 ± 0.067**
	Without ordered RNA-seq data	0.447 ± 0.101	0.679 ± 0.071	0.663 ± 0.065
	Unique RNA-seq feature extractor	0.381 ± 0.103	0.580 ± 0.088	0.576 ± 0.084
	Without task descriptor	0.376 ± 0.092	0.655 ± 0.065	0.638 ± 0.060
	Without weighted random sampler	0.351 ± 0.092	0.573 ± 0.080	0.577 ± 0.076

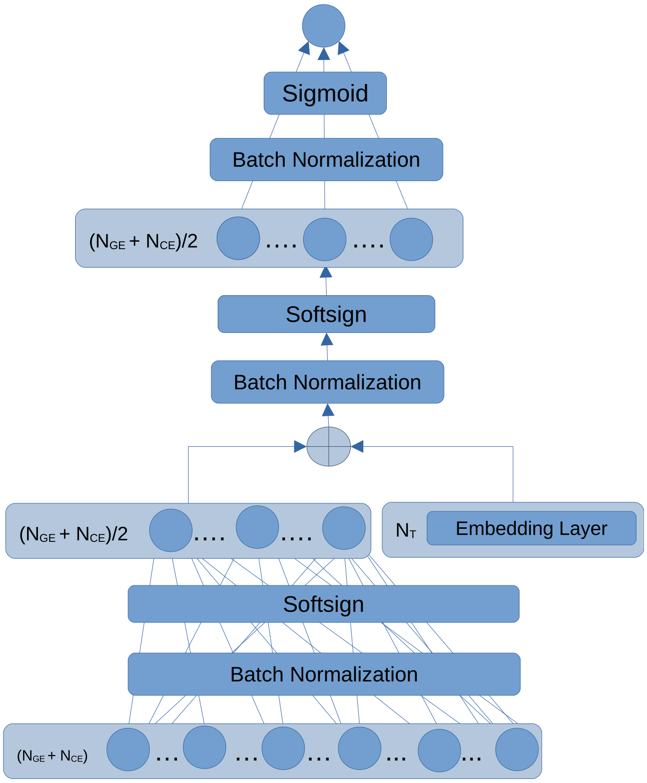

Fig. 5 | Classifier architecture. N_{GE} refers to the RNA-seq feature embedding, N_{CE} corresponds to the dimension of the clinical feature embedding, and N_T represents the number of tasks.

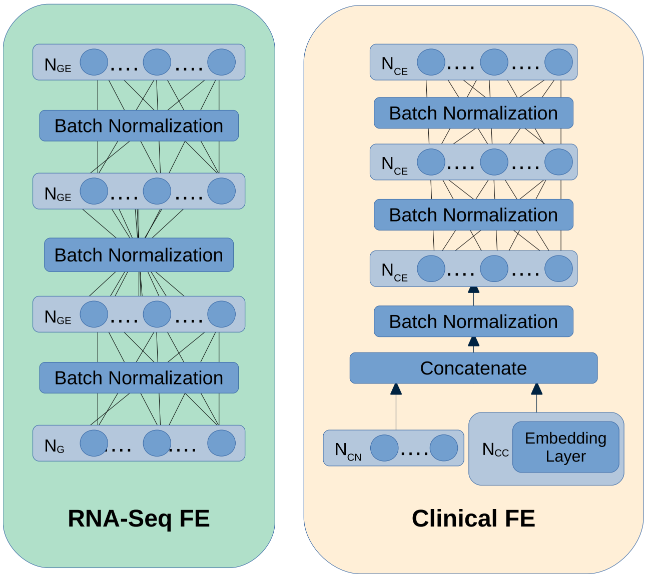

Fig. 6 | Feature extractors for RNA-seq and clinical data. RNA-seq (left) and clinical (right) feature extractors. N_G stands for the number of genes, N_{GE} and N_{CE} corresponds to the RNA-Seq and clinical feature embedding dimensions. N_{CN} and N_{CC} correspond to the number of numerical and categorical clinical features.

staging distribution, may further enhance long-term prediction accuracy.

This study examined the potential of MTL to overcome the limitations of small datasets in cancer prognosis prediction. We used MTL to improve data efficiency by treating each cancer type as a different task. MTL showed significant improvements in AUROC, AUPRC, and C-index of 29%, 41%, and 26% in COAD, respectively, and minor improvements of 2% to 5% for all metrics in BRCA and LUAD. Furthermore, an external validation using SCLC data yielded performance comparable to that of the LUAD dataset. This result suggests extending our MTL approach to other types of cancers may be a promising direction for future research. Our model took advantage of shared information among the embeddings of different cancer types via MTL and bimodal neural networks to overcome the limitations of small datasets. Ablation studies justified the model design and the effectiveness of combining multiple cancer data using a unified prediction model. Consequently, we plan to extend our work to include rarer cancer types, addressing the challenge posed by small sample sizes. Ultimately, we aspire to contribute to significant advancements in precision oncology.

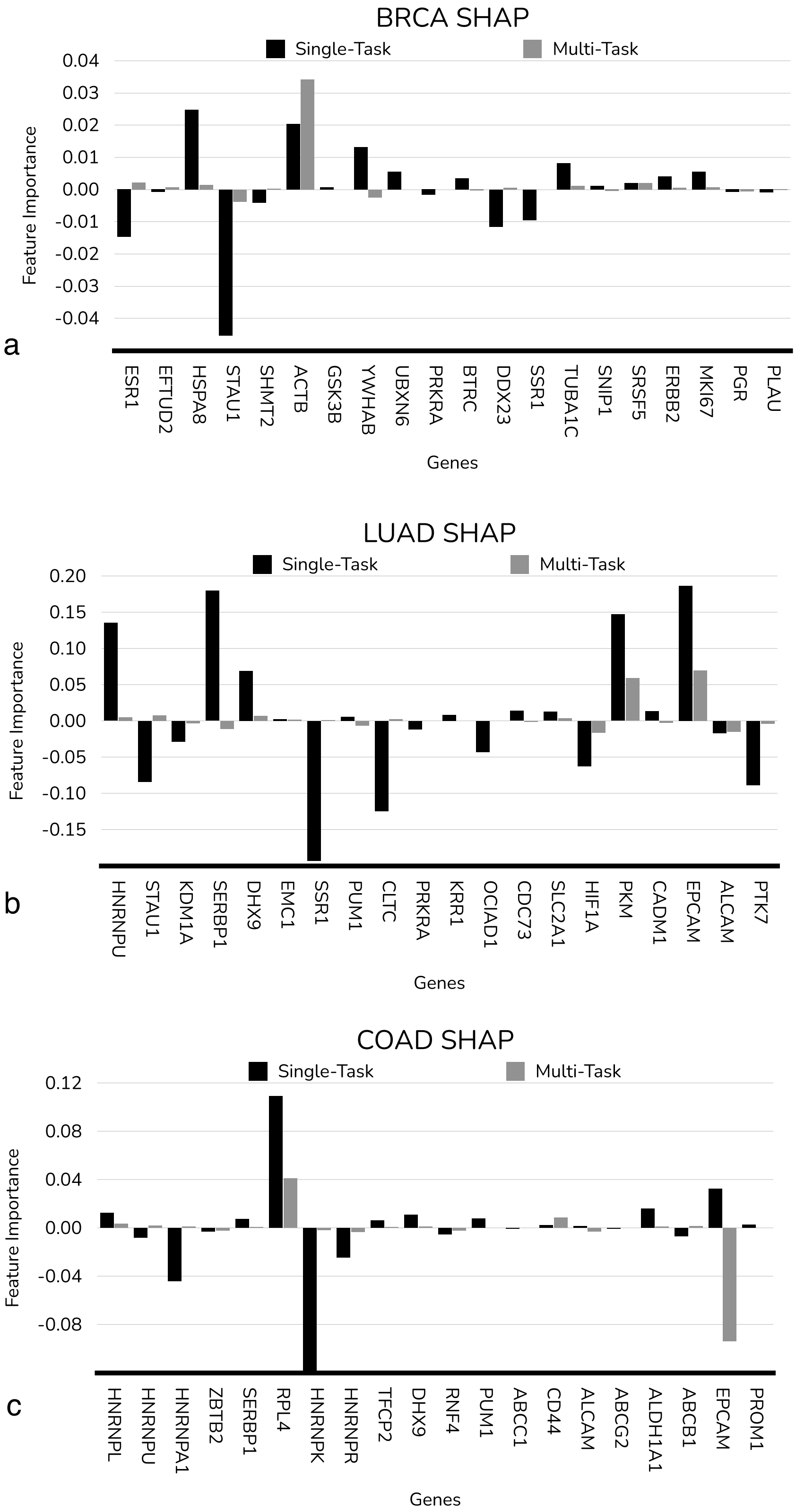

Fig. 7 | Evaluating gene contribution to prognosis with SHAP values in different learning models. The subfigures represent BRCA (**a**), LUAD (**b**), and COAD (**c**) genomic features. The bar chart contrasts the contribution of individual genes to model predictions between STL (black bars) and MTL (gray bars) frameworks. Positive SHAP values indicate a gene's expression level contributes to a higher risk prediction, whereas negative values suggest a protective or lessened effect. Each bar's magnitude reflects the corresponding gene's average impact across every dataset.

Methods

Problem formulation

Consider three prediction tasks $\{\mathcal{T}_i\}_{i=1}^{3}$, all prognosis prediction tasks for different cancer types. Each task $\mathcal{T}_i$ comes with a training dataset $\mathcal{D}_i^{\text{train}}$ containing N_i samples, i.e., $\mathcal{D}_i^{\text{train}} = \left\{\left(\boldsymbol{x}_n^i, y_n^i\right)\right\}_{n=1}^{N_i}$, where $\boldsymbol{x}_n^i \in \mathbb{R}^{d_i}$ is the feature vector with dimension d_i and $y_n^i \in \{0, 1\}$ is the label of the task $\mathcal{T}_i$ for the nth training sample. We consider a model $f_i\left(\boldsymbol{x}^i; \boldsymbol{\theta}^{\text{share}}, \boldsymbol{\theta}^i\right)$ for each task $\mathcal{T}_i$ such that $\boldsymbol{\theta}^{\text{share}}$ are the parameters shared between tasks and $\boldsymbol{\theta}^i$ are the parameters related to the specific task $\mathcal{T}_i$. The binary cross-entropy is adopted for each task $\mathcal{T}_i$

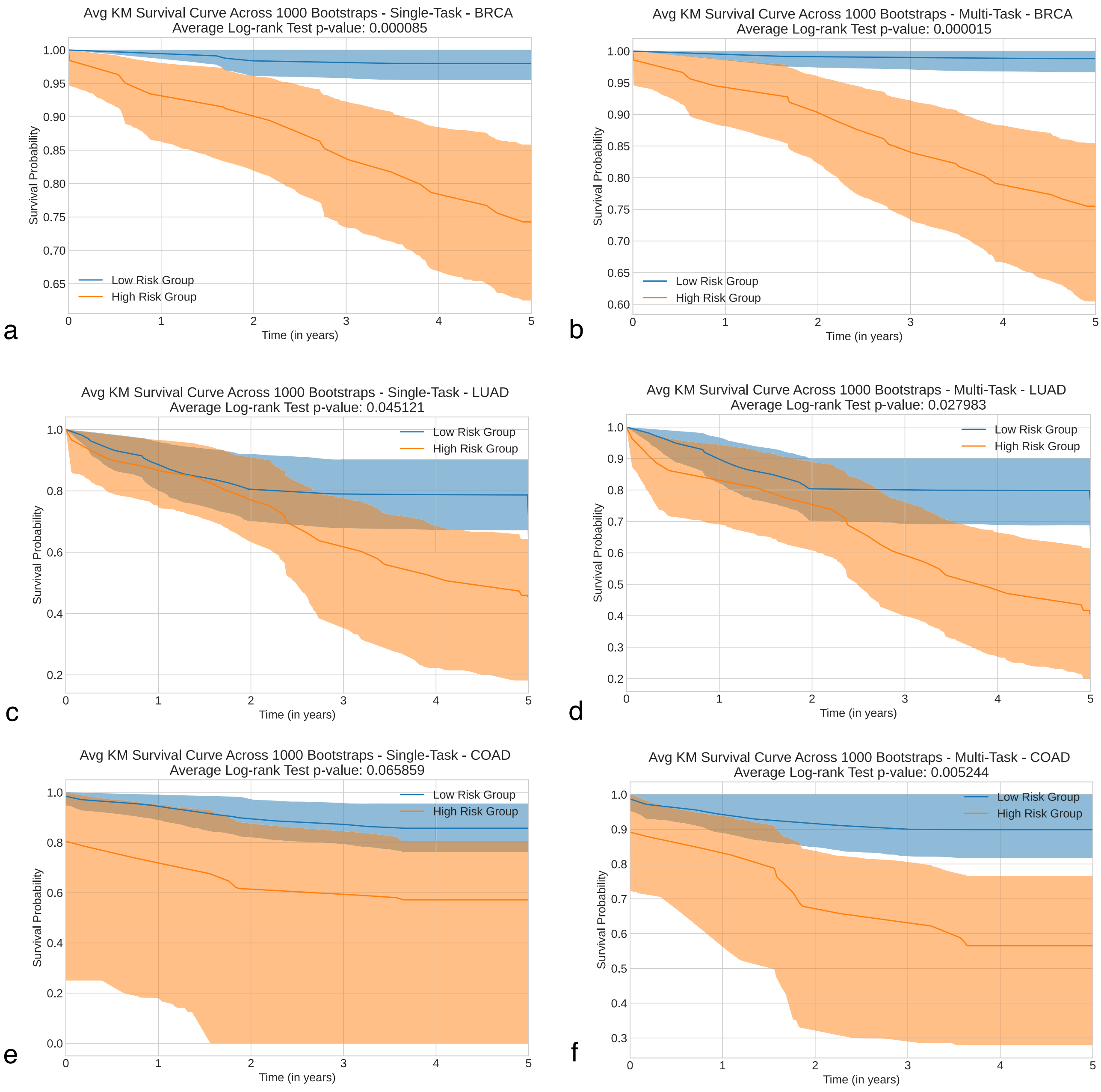

Fig. 8 | Kaplan–Meier survival curves for BRCA, LUAD, and COAD using STL and MTL. Subfigures (**a**), (**c**), and (**e**) show the STL KM plots for BRCA, LUAD, and COAD, respectively. Subfigures (**b**), (**d**), and (**f**) present their MTL counterparts. The shaded area represents the confidence intervals for each risk group. Blue refers to the low-risk group, and orange refers to the high-risk group. The *p* value for the log-rank test is shown along the title for each plot.

as the loss function and can be written as:

$$\mathcal{L}_i\left(\left\{\boldsymbol{\theta}^{\text{share}}, \boldsymbol{\theta}^i\right\}, \mathcal{D}_i^{\text{train}}\right) = -\frac{1}{N_i}\sum_{n=1}^{N_i}\left(y_n^i \log\left(f_i\left(\boldsymbol{x}_n^i; \boldsymbol{\theta}^{\text{share}}, \boldsymbol{\theta}^i\right)\right) + \left(1-y_n^i\right)\log\left(1-f_i\left(\boldsymbol{x}_n^i; \boldsymbol{\theta}^{\text{share}}, \boldsymbol{\theta}^i\right)\right)\right). \quad (1)$$

We obtained optimal neural network parameters by minimizing the loss function across all tasks, as shown in Eq. (2).

$$\min_{\boldsymbol{\theta}^{\text{share}}, \left\{\boldsymbol{\theta}^i\right\}_{i=1}^3} \sum_{i=1}^{3} \mathcal{L}_i\left(\left\{\boldsymbol{\theta}^{\text{share}}, \boldsymbol{\theta}^i\right\}, \mathcal{D}_i^{\text{train}}\right). \quad (2)$$

Evaluation metrics

AUROC. The Area Under the Curve of the Receiver Operating Characteristic curve for a binary classification model calculates the proportion of all positive-negative pairs that are correctly classified[38]. The ROC curve plots the True Positive Rate (TPR) against the False Positive Rate (FPR) at various thresholds. These rates are calculated as follows:

$$TPR = \frac{TP}{TP + FN},$$

where *TP* is the number of true positives, and *FN* is the number of false negatives.

$$FPR = \frac{FP}{FP + TN},$$

where *FP* is the number of false positives, and *TN* is the number of true negatives.

The AUROC can be interpreted as the probability that the classifier will rank a randomly chosen positive instance higher than a randomly chosen negative instance. A higher AUROC value indicates better model performance, with 1 being a perfect classifier and 0.5 indicating a performance no better than random chance. We calculate the AUROC using the following formula:

$$AUROC(f) = \frac{\sum_{t_0\in\mathcal{D}^0}\sum_{t_1\in\mathcal{D}^1}\mathbf{I}[f(t_0)<f(t_1)]}{|\mathcal{D}^0|\cdot|\mathcal{D}^1|},$$

here $\mathcal{D}^0$ and $\mathcal{D}^1$ represent the negative and positive classes and $|D^0|$ and $|D^1|$ the number of samples in each class. $\mathcal{I}$ represents the indicator function

AUPRC. The precision-recall curve plots the recall on the x axis and Precision on the y-axis for different threshold values[39]. It measures how well a classifier identifies positive cases. As a result, the baseline for this metric is not fixed and depends on the proportion of positive instances in the dataset. The Precision (P) corresponds to the ratio of true positive predictions versus the total number of positive predictions (both true positives and false positives), and it is given by the following formula:

$$P = \frac{TP}{TP + FP}.$$

In contrast, the Recall (R) corresponds to the ratio of true positive predictions to the total number of actual positives in the data, calculated using:

$$R = \frac{TP}{TP + FN},$$

here *TP* is the number of true positives, *FP* is the number of false positives, and *FN* is the number of false negatives.

Finally, since we are in a discrete setting, the AUPRC is calculated by the following formula:

$$AUPRC = \sum_{n}(R_n - R_{n-1})P_n,$$

where P_n and R_n are the Precision and Recall for a threshold with index n.

C-index. The C-index quantifies the correlation between observed survival times and predicted risk scores[40]. A pair is defined as concordant if the patient with the shorter survival time also has a higher risk score. In this study, the C-index is calculated using:

$$C = \frac{\sum_{i,j}\mathcal{I}\left(T_i>T_j\right)\cdot\mathcal{I}\left(\eta_i<\eta_j\right)}{\sum_{i,j}\mathcal{I}\left(T_i>T_j\right)},$$

where *i* and *j* denote distinct patients, T_i and T_j represent their survival times, η_i and η_j are their corresponding model-predicted risk scores, and $\mathcal{I}$ the indicator function.

Datasets

The study includes three primary datasets from the TCGA project: BRCA, LUAD, and COAD. The TCGA project focuses on generating, managing, analyzing, and interpreting molecular profiles at various levels for many human tumors with different types and subtypes[41]. We filtered outpatients lacking survival status, time, and incomplete RNA-Seq or clinical data. We downloaded the data of qualified patients using GDC API version 33.1, released on May 31, 2022[25]. An external dataset used to test our model's generalizability was obtained from cBioPortal[28–30]. This validation dataset was collected for the University of Cologne's 2015 SCLC study[27]. We followed the same filtering procedure as with TCGA datasets. Since not all genes in LUAD are also present in the SCLC data, we used BRCA's selected genes for the SCLC-MTL prediction, as all genes selected for BRCA were present among the SCLC genomic data. Surprisingly, the performance on the SCLC set using BRCA genes is comparable to that of LUAD and COAD. We attribute this to the relative abundance of BRCA data compared to the other cancers. This reinforces our idea that the MTL model can transfer insight to less-represented cancers by drawing upon a larger dataset, thereby improving its performance in smaller datasets.

RNA-seq and clinical data

The data comprised RNA-Seq and clinical data. The RNA-Seq data had up to 60,000 probes per patient, with multiple probes possibly matching a gene name using the GENCODE v36 version. Due to the limitations of the SBFS, we selected protein-encoding genes as our gene candidates. We used transcripts per million (TPM) as our count transformation for the RNA-Seq data. We selected six categories of clinical attributes for the clinical data. Age, birth year, and diagnosis year were considered numerical data, and gender, race, and ethnicity were used as categorical data. The numerical data were standardized using the training set's mean and standard deviation. In contrast, categorical data were integrated and encoded through an embedding layer for the bimodal neural network and one-hot encoded for other models. RNA-Seq data were not further standardized, as the count transformation had already been normalized.

Systems biology feature selector

To address the curse of dimensionality issue in high-dimensional omics data, we apply a robust SBFS method (Fig. 1) for selecting a small gene subset with biological insights[24]. We used well-known genetic biomarkers from previous studies for three cancer types to select prognostic genes[7,8,24]. Based on the literature, we selected five, seven, and eight well-known genetic biomarkers for BRCA, LUAD, and COAD (summarized in Table 6). We divided the SBFS into two primary parts, detailed below.

StepMiner and ANOVA

We used StepMiner[42] to categorize patients in the training set into two separate genetic biomarker+ and genetic biomarker- groups based on the gene-level expression for each well-known genetic biomarker. Then, we conducted an analysis of variance (ANOVA) to exclude invariant genes between the two groups, resulting in two distinct genetic biomarker groups and a list of relevant gene candidates.

GIN and ranking

We built two Gene interaction network (GINs) based on interactions documented in BioGrid[43] for these two biomarker groups. Figure 9 shows an example of a Gene Interaction Network. Following this, we calculated the Akaike information criterion to select a proper model for each GIN. Then, we computed multiple prognostic relevance values (PRVs) for all candidate genes and ranked them by PRV scores. We calculated the PRVs by summing the differences in model weights for neighboring genes between both GINs. PRVs allow us to condense how a gene will interact with its neighbors in different prognosis scenarios.

Gene candidates without a significant difference in expression between biomarker+ and biomarker- groups were cut off from the list. We then

Table 6 | Each row shows well-known genetic biomarkers for each cancer type (BRCA, LUAD, and COAD) that the SBFS later used

Datasets	Well-known genetic biomarkers
BRCA	ESR1, PGR, ERBB2, MKI67, PLAU
LUAD	EPCAM, HIF1A, PKM, PTK7, ALCAM, CADM1, SLC2A1
COAD	EPCAM, CD44, ALCAM, PROM1, ABCB1, ABCC1, ABCG2, ALDH1A1

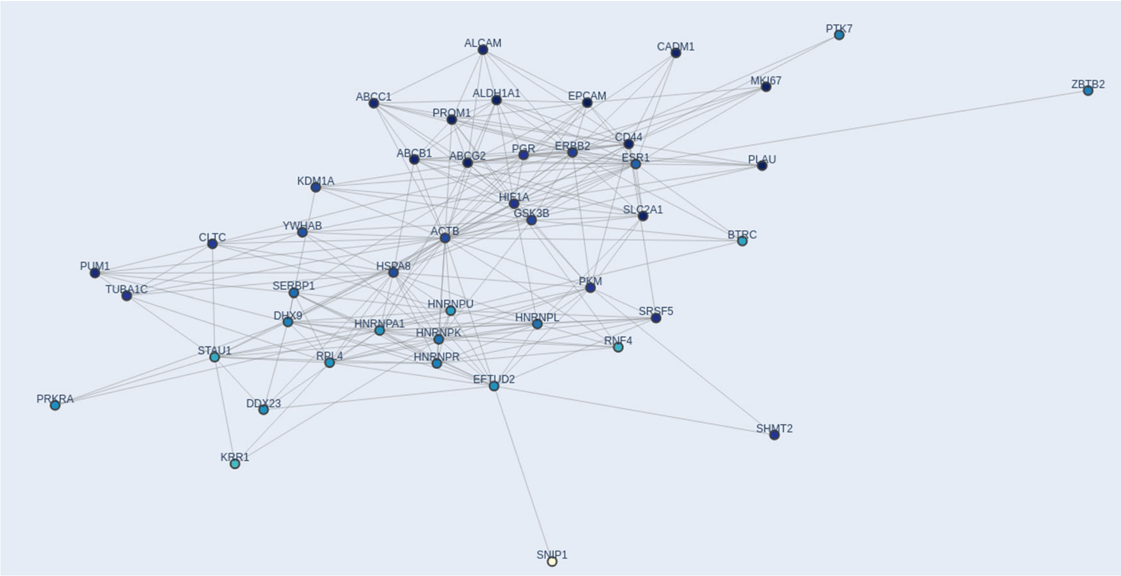

Fig. 9 | Gene Interaction Network illustrating the interactions between relevant biomarkers for all three cancers.

Table 7 | This table showcases the top 20 prognostic genes selected by the SBFS

Datasets	Twenty top-ranking prognostic genes
BRCA	**ESR1**, EFTUD2, HSPA8, STAU1, SHMT2, ACTB, GSK3B, YWHAB, UBXN6, PRKRA, BTRC, DDX23, SSR1, TUBA1C, SNIP1, SRSF5, **ERBB2, MKI67, PGR, PLAU**
LUAD	HNRNPU, STAU1, KDM1A, SERBP1, DHX9, EMC1, SSR1, PUM1, CLTC, PRKRA, KRR1, OCIAD1, CDC73, **SLC2A1, HIF1A, PKM, CADM1, EPCAM, ALCAM, PTK7**
COAD	HNRNPL, HNRNPU, HNRNPA1, ZBTB2, SERBP1, RPL4, HNRNPK, HNRNPR, TFCP2, DHX9, RNF4, PUM1, **ABCC1, CD44, ALCAM, ABCG2, ALDH1A1, ABCB1, EPCAM, PROM1**

The genes are listed in the same order as they were ranked by the feature selection process. Well-known genetic biomarkers (previously mentioned in Table 6) are emphasized in boldface.

ranked the remaining gene candidates according to their PRV scores. Subsequently, we obtained a global gene list by summing the rankings of all gene candidates. Finally, we selected the twenty top-ranking prognostic genes from this comprehensive list. The final list is presented in Table 7.

Model architecture

We introduce a bimodal neural network, a unique model architecture designed for two modalities[44]. It integrates RNA-Seq and clinical data to obtain accurate prognostic predictions. Each modality uses a distinct feature extractor to generate high-level embedded features. These embedding features are subsequently concatenated into a single embedded feature, fed through a classifier for prognosis prediction, a process referred to as intermediate fusion. As opposed to early fusion, where different modalities are combined before feature extraction, intermediate fusion integrates distinct modalities after feature extraction but before classification. Our bimodal neural network is divided into two main components: a feature extractor and a classifier. The architecture of the proposed model is depicted in Fig. 10.

The feature extractor (Fig. 6) comprises an RNA-Seq feature extractor, a clinical feature extractor, and a concatenating layer. These two neural networks are designed for the different modalities, transforming each into high-level embeddings. As RNA-Seq and clinical data have similar raw data formats, their feature extractors share some principles in their model architecture design (shown in Table 8).

The RNA-Seq feature extractor has four layers, using count transformations, such as TPM[45], for RNA-Seq data. We added batch normalization layers across layers to prevent gradient explosion[46]. The RNA-Seq data were sorted in descending order based on the PRV calculated by the SBFS to ensure consistency across tasks and embed implicit information.

The clinical feature extractor also has four layers with batch normalization. We added an embedding layer to transform clinical categorical data into a fixed-dimension embedding. This prevents sparsity issues and allows the clinical feature extractor to map clinical categorical data, as in word embeddings in natural language processing[47]. The embeddings of the clinical categorical data are averaged over a single patient and concatenated with clinical numerical data embeddings.

Finally, we concatenated the high-level embeddings generated from the RNA-Seq and clinical feature extractors, forming a single embedding, which the classifier (Fig. 5) later uses as an input, along with the task descriptor.

The classifier (Fig. 5) takes the high-level feature embedding generated from the feature extractor and predicts cancer prognosis. It has four layers with batch normalization[46] and Softsign[48] to improve the nonlinear transform and stable training. We applied Softsign to improve the model's stability across computer devices and its capacity to handle high-level embeddings from multiple tasks.

MTL techniques

Humans typically benefit from learning multiple tasks simultaneously[49], but in machine learning, MTL may underperform STL due to negative transfer[50]. Some challenges affecting performance negatively are dealing with infrequent and highly specific tasks, tuning shared parameters, and transferring features[12,51]. To address these challenges, we apply several techniques to enhance MTL's performance.

We use hard parameter sharing to reduce shared parameter portions and prevent overfitting on specific tasks[52], in contrast to parameter-sharing state-of-the-art models[53–55]. All parameters, including the feature extractor

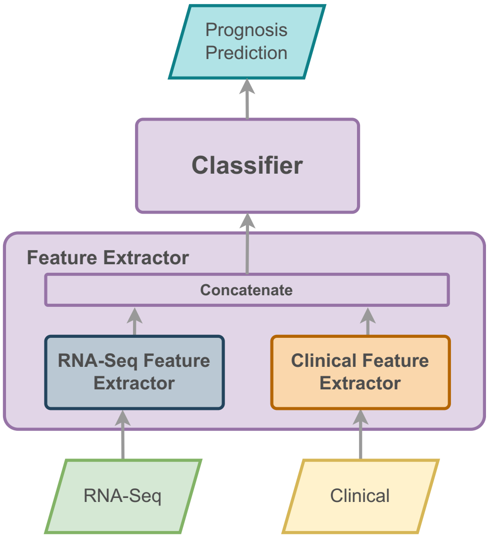

Fig. 10 | Bimodal model architecture overview. The diagram illustrates the overall architecture of our model, combining the two feature extractors and the classifier.

and classifier, are directly shared across tasks, resulting in identical parameters for all tasks.

We added a task descriptor to the model architecture, containing information representing each task and distinguishing it from others. In this study, the task descriptor indicates the cancer type. An embedding layer is added to the classifier, transforming this information into a task descriptor embedding. This approach allows the feature extractor to capture the general relationship between tasks without confusion from different task features[56].

Two optimization techniques are applied for proper MTL. A weighted random data sampler balances the number of samples between tasks in each batch, using weights (provided as probabilities) determined by the inverse square of each task's sample size in the training set. The sampler then uses these probabilities to draw data according to the multinomial probability distribution across all tasks, with replacement. This approach avoids manipulating losses across all tasks[57–59]. We used a weighted random data sampler to prevent tasks with more extensive data samples from becoming dominant and ensure that all gradients contributed nearly equally to the learning. Additionally, two different optimizers were used to optimize the feature extractor and classifier separately, updating the feature extractor first and then the classifier to stabilize gradient updates during training.

Experiment settings

In the STL experiment, we applied default parameter settings from Scikit-learn[60] for LR, RF, and SVM with minor changes. We adopted the balanced class weights mode, as datasets were imbalanced. SVM required input data normalization. We used the PyTorch framework[61] to build bimodal neural networks and stochastic gradient descent (SGD) with momentum as optimizer[62]. The embedding dimensions for the bimodal neural networks (RNA-Seq feature extractor, clinical feature extractor, and classifier) were equal to eight. We adopted SGD hyperparameters from the literature[62], with a learning rate of 0.01 and momentum of 0.9. We trained bimodal neural networks for 50 epochs and used the last epoch's model checkpoint for testing. Experiments were performed using Ryzen 72700 (CPU), DDR4 64 GB (RAM), and Nvidia GeForce RTX 3060 (GPU).

Patients were divided into training and test sets for each cancer type in a 4:1 ratio, with stratified splits based on cancer prognostic labels using a random seed number "1126". The test set was treated like a hold-out set to avoid data leakage. Different data samplers were adopted for different learning paradigms: random sampler without weighting for STL and random sampler with weighting for MTL. Before the testing stage, we used a fourfold cross-validation approach, where the training data was divided into four equal parts. In each fold, 75% of the training data was used for training and 25% for validation, ensuring each data segment was used once as the validation set.

Table 8 | MTL BNN hyperparameter settings

Model	Hyperparameter settings
RNA-seq feature extractor	RNA-Seq dimensions: 20
	RNA-Seq embedding dimensions: 8
Clinical feature extractor	Clinical numerical dimensions: 3
	Clinical categorical dimensions: 11
	Clinical embedding dim: 8
	Task descriptor dimensions: 3
Classifier	RNA-Seq embedding dimensions: 8
	Clinical embedding dimensions: 8
	Output dimensions: 1

We trained on the whole training set (including training and validation sets used in the cross-validation) for testing. Then, we sampled patients from the original unseen test set using 1000 bootstrapped test sets with replacement, using the size of the original test set for each bootstrapped test set, estimating variations for all evaluation metrics[63]. We followed the same methodology for the external SCLC dataset, assigning it the same task descriptor as LUAD for the MTL model. The 20 genes selected for the feature extractor were present both in the TCGA genes used for training and in the SCLC RNA-Seq data.

For the SCLC external validation, we trained the MTL model similarly to the TCGA evaluation. We then evaluated its performance on the complete SCLC set using this trained model. Various configurations combining selected genes from BRCA, LUAD, and COAD, along with different task descriptors, were tested to analyze the model's behavior under diverse assumptions. Given the similarities of SCLC with LUAD compared to other cancers, we decided to use LUAD's gene selection for the MTL tasks. Since not all LUAD genes were present in SCLC, we used 18 LUAD genes and supplemented them with two genes from BRCA that were also present in SCLC, completing the 20-gene set.

Data availability

The datasets analyzed during the current study are publicly available in the Genomic Data Commons (GDC) Data Portal. The three primary datasets focusing on BRCA, LUAD, and COAD can be accessed through the TCGA project, available at https://portal.gdc.cancer.gov. The external validation dataset is available at https://www.cbioportal.org/study/summary?id=sclc_ucologne_2015.

Code availability

The source code for reproducing the main results is available in this repository, along with their documentation: https://github.com/idssplab/Multi-Cancer. A static version can be found at Zenodo with https://doi.org/10.5281/zenodo.12203877.

Received: 8 September 2023; Accepted: 30 August 2024;
Published online: 29 October 2024

References

1. Siegel, R. L., Miller, K. D., Fuchs, H. E. & Jemal, A. Cancer statistics, 2022. *CA Cancer J. Clin*. **72**, 7–33 (2022).
2. Barry, M. J. Prostate-specific–antigen testing for early diagnosis of prostate cancer. *N. Engl. J. Med.* **344**, 1373–1377 (2001).
3. Indyk, P. & Motwani, R. Approximate nearest neighbors: towards removing the curse of dimensionality. In: *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, 604–613 (1998).

4. Gulshan, V. et al. Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *JAMA* **316**, 2402–2410 (2016).
5. Bejnordi, B. et al. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *JAMA* **318**, 2199–2210 (2017).
6. Esteva, A. et al. Dermatologist-level classification of skin cancer with deep neural networks. *Nature* **542**, 115–118 (2017).
7. Lai, Y.H., Chen, W.N., Hsu, T.C., Lin, C., Tsao Y. & Wu S. Overall survival prediction of non-small cell lung cancer by integrating microarray and clinical data with deep learning. *Sci. Rep*. **13**, 4679 (2020).
8. Hsu, T.-C. & Lin, C. Training with small medical data: robust bayesian neural networks for colon cancer overall survival prediction. In: *2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), 2030–2033* (IEEE, 2021).
9. Reya, T., Morrison, S., Clarke, M. & Weissman, I. Stem cells, cancer, and cancer stem cells. *Nature* **414**, 105–111 (2001).
10. Bogenrieder, T. & Herlyn, M. Axis of evil: molecular mechanisms of cancer metastasis. *Oncogene* **22**, 6524–6536 (2003).
11. Gaire, R. Discovery and analysis of consistent active sub-networks in cancers. *BMC Bioinform.* **14**, S7 – S7 (2013).
12. Zhang, Y. & Yang, Q. A survey on multi-task learning. In: *IEEE Transactions on knowledge and data engineering* (2022).
13. Ando, R., Zhang, T., Bartlett, P. & Barry, M. A framework for learning predictive structures from multiple tasks and unlabeled data. *J. Mach. Learn. Res.* **6**, 1373–1377 (2005).
14. Collobert, R. & Weston, J. A unified architecture for natural language processing: deep neural networks with multitask learning. In: *Proceedings of the 25th international conference on Machine learning*, 160–167 (2008).
15. Collobert, R. et al. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.* **12**, 2493–2537 (2011).
16. Zhang, T., Ghanem, B., Liu, S. & Ahuja, N. Robust visual tracking via structured multi-task sparse learning. In: *International journal of computer vision* **101**, 367–383 (2013).
17. Donahue, J. et al. Decaf: a deep convolutional activation feature for generic visual recognition. In: *International conference on machine learning*, 647–655 (PMLR, 2014).
18. Zhang, Z., Luo, P., Loy, C. & Tang, X. Facial landmark detection by deep multi-task learning. In: *European conference on computer vision*, 94–108 (Springer, 2014).
19. Girshick, R. Fast r-cnn. In: *Proceedings of the IEEE international conference on computer vision*, 1440–1448 (2015).
20. Zhou, J., Yuan, L., Liu, J. & Ye, J. A multi-task learning formulation for predicting disease progression. In: *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, 814–822 (2011).
21. Mordelet, F. & Vert, J.-P. Prodige: prioritization of disease genes with multitask machine learning from positive and unlabeled examples. *BMC Bioinform.* **12**, 1–15 (2011).
22. Ramsundar, B. et al. Massively multitask networks for drug discovery. *ArXiv* https://arxiv.org/abs/1502.02072 (2015).
23. Wu, C. et al. A selective review of multi-level omics data integration using variable selection. *High Throughput* **8**, 4 (2019).
24. Cheng, L.-H., Hsu, T.-C. & Lin, C. Integrating ensemble systems biology feature selection and bimodal deep neural network for breast cancer prognosis prediction. *Sci. Rep.* **11**, 1–10 (2021).
25. Grossman, R. et al. Toward a shared vision for cancer genomic data. *N. Engl. J. Med.* **375**, 1109–1112 (2016).
26. Liu, J. et al. An integrated TCGA pan-cancer clinical data resource to drive high-quality survival outcome analytics. *Cell* **173**, 400–416 (2018).
27. George, J. et al. Comprehensive genomic profiles of small cell lung cancer. *Nature* **524**, 47–53 (2015).
28. Cerami, E. et al. The cBio cancer genomics portal: an open platform for exploring multidimensional cancer genomics data. *Cancer Discov.* **2**, 401–404 (2012).
29. Gao, J. et al. Integrative analysis of complex cancer genomics and clinical profiles using the cBioPortal. *Sci. Signal.* **6**, pl1–pl1 (2013).
30. de Bruijn, I. et al. Analysis and visualization of longitudinal genomic and clinical data from the AACR project genie biopharma collaborative in cBioPortal. *Cancer Res.* **83**, 3861–3867 (2023).
31. Poirion, O. B., Jing, Z., Chaudhary, K., Huang, S. & Garmire, L. X. Deepprog: an ensemble of deep-learning and machine-learning models for prognosis prediction using multi-omics data. *Genome Med.* **13**, 1–15 (2021).
32. Katzman, J. L. et al. Deepsurv: personalized treatment recommender system using a Cox proportional hazards deep neural network. *BMC Med. Res. Methodol.* **18**, 1–12 (2018).
33. Huang, Z. et al. Deep learning-based cancer survival prognosis from RNA-seq data: approaches and evaluations. *BMC Med. Genom.* **13**, 1–12 (2020).
34. Ching, T., Zhu, X. & Garmire, L. X. Cox-nnet: an artificial neural network method for prognosis prediction of high-throughput omics data. *PLoS Comput. Biol.* **14**, e1006076 (2018).
35. Lundberg, S. M. & Lee, S.-I. A unified approach to interpreting model predictions. In: *Advances in neural information processing systems* 30 (2017).
36. Kokhlikyan, N. et al. Captum: a unified and generic model interpretability library for pytorch. *arXiv* https://arxiv.org/abs/2009.07896 (2020).
37. Kaplan, E. L. & Meier, P. Nonparametric estimation from incomplete observations. *J. Am. Stat. Assoc.* **53**, 457–481 (1958).
38. Calders, T. & Jaroszewicz, S. Efficient AUC optimization for classification. In: *European conference on principles of data mining and knowledge discovery*, 42–53 (Springer, 2007).
39. Sofaer, H. R., Hoeting, J. A. & Jarnevich, C. S. The area under the precision-recall curve as a performance metric for rare binary events. *Methods Ecol. Evol.* **10**, 565–577 (2019).
40. Harrell, F. E., Califf, R. M., Pryor, D. B., Lee, K. L. & Rosati, R. A. Evaluating the yield of medical tests. *JAMA* **247**, 2543–2546 (1982).
41. Weinstein, J. et al. The cancer genome atlas pan-cancer analysis project. *Nat. Genet.* **45**, 1113–1120 (2013).
42. Sahoo, D., Dill, D., Tibshirani, R. & Plevritis, S. Extracting binary signals from microarray timecourse data. *Nucleic Acids Res.* **35**, 3705–3712 (2007).
43. Stark, C. et al. Biogrid: a general repository for interaction datasets. *Nucleic Acids Res.* **34**, 535–539 (2006).
44. Ngiam, J. et al. Multimodal deep learning. In: *ICML* https://people.csail.mit.edu/khosla/papers/icml2011_ngiam.pdf (2011).
45. Li, B., Ruotti, V., Stewart, R., Thomson, J. & Dewey, C. Rna-seq gene expression estimation with read mapping uncertainty. *Bioinformatics* **26**, 493–500 (2010).
46. Ioffe, S. & Szegedy, C. Batch normalization: accelerating deep network training by reducing internal covariate shift. In: *International conference on machine learning, 448456* (PMLR, 2015).
47. Mikolov, T., Chen, K., Corrado, G. & Dean, J. Efficient estimation of word representations in vector space. *ArXiv* https://arxiv.org/abs/1301.3781 (2013).
48. Turian, J., Bergstra, J. & Bengio, Y. Quadratic features and deep architectures for chunking. In: *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, 245–248 (2009).
49. Caruana, R. Multitask learning. *Machine learning* **28**, 41–75 (1997).
50. Pan, S. & Yang, Q. A survey on transfer learning. In: *IEEE transactions on knowledge and data engineering.* **22**, 1345–1359 (2010).
51. Yosinski, J., Clune, J., Bengio, Y. & Lipson, H. How transferable are features in deep neural networks? In: *Advances in neural information processing systems* 27 (2014).

52. Ruder, S. An overview of multi-task learning in deep neural networks. *ArXiv* https://arxiv.org/abs/1706.05098 (2017).
53. Misra, I., Shrivastava, A., Gupta, A. & Hebert, M. Cross-stitch networks for multi-task learning. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3994–4003 https://arxiv.org/abs/1604.03539 (2016).
54. Long, M., Cao, Z., Wang, J. & Yu, P. Learning multiple tasks with multilinear relationship networks. In: *Advances in neural information processing systems* 30 (2017).
55. Liu, S., Johns, E. & Davison, A. End-toend multi-task learning with attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 1871–1880 https://arxiv.org/abs/1803.10704 (2019).
56. Dumoulin, V. et al. Feature-wise transformations. *Distill* **3**, 11 (2018).
57. Chen, Z., Badrinarayanan, V., Lee, C.-Y. & Rabinovich, A. Gradnorm: gradient normalization for adaptive loss balancing in deep multitask networks. In: *International Conference on Machine Learning*, 794–803 (PMLR, 2018).
58. Kendall, A., Gal, Y. & Cipolla, R. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7482–7491 (2018).
59. Sener, O. & Koltun, V. Multi-task learning as multiobjective optimization. In: *Advances in neural information processing systems*, 31 (2018).
60. Pedregosa, F. et al. Scikit-learn: machine learning in python. *J. Mach. Learn. Res*. **12**, 2825–2830 (2011).
61. Paszke, A. et al. Pytorch: an imperative style, high-performance deep learning library. In: *Advances in neural information processing systems*, 32 (2019).
62. Sutskever, I., Martens, J., Dahl, G. & Hinton, G. On the importance of initialization and momentum in deep learning. In: *International conference on machine learning*, PMLR, **28**, 1139–1147 (2013).
63. Efron, B. & Tibshirani, R. An introduction to the bootstrap, 1st edn, 456 (CRC press, 1994).

Acknowledgements

This study was supported by the National Science and Technology Council (NSTC), Taiwan [grant number MOST 110-2221-E-002-112-MY3]. This work was also supported by the Ministry of Health and Welfare (MOHW), Taiwan [grant number MOHW111-TDU-B-221-114003]. The Systems Biology Feature Extractor development was funded by the Ministry of Science and Technology, Taiwan [grant number 109-2218-E-00T2-016].

Author contributions

B.-R.W. and S.O.A. drafted the manuscript. B.-R.W., T.-C.H., and C.L. designed the model and overall pipeline. T.C.H. contributed to the conception and revised the manuscript. B.-R.W., S.O.A., and T.-W.L. worked on the code and data preprocessing. B.-R.W. created the reusable code to download the data. S.O.A. and C.L. worked on the editing and rebuttals.

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/s41698-024-00700-z.

Correspondence and requests for materials should be addressed to Sofia Ormazabal Arriagada or Che Lin.

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

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.