Multitask-Guided Self-Supervised Tabular Learning for Patient-Specific Survival Prediction

You Wu[†*] **Omid Bazgir**[‡] **Yongju Lee**[‡] **Tommaso Biancalani**[‡] **James Lu**[‡]

Ehsan Hajiramezanali[‡§]

† City University of New York, ‡ Genentech

Abstract

Survival prediction, central to the analysis of clinical trials, has the potential to be transformed by the availability of RNA-seq data as it reveals the underlying molecular and genetic mechanisms for disease and outcomes. However, the amount of RNA-seq samples available for understudied or rare diseases is often limited. To address this, leveraging data across different cancer types can be a viable solution, necessitating the application of self-supervised learning techniques. Yet, this wealth of data often comes in a tabular format without a known structure, hindering the development of a generally effective augmentation method for survival prediction.While traditional methods have been constrained by a *one cancer-one model* philosophy or have relied solely on a single modality, our approach, **Guided-STab**, on the contrary, offers a comprehensive approach through pretraining on all available RNA-seq data from various cancer types while guiding the representation by incorporating sparse clinical features as auxiliary tasks. With a multitask-guided self-supervised representation learning framework, we maximize the potential of vast unlabeled datasets from various cancer types, leading to genomic-driven survival predictions. These auxiliary clinical tasks then guide the learned representations to enhance critical survival factors. Extensive experiments reinforce the promise of our approach, as Guided-STab consistently outperforms established benchmarks on TCGA dataset.

1 Introduction

Survival analysis is commonly used in clinical research to quantify the likelihood distribution of the time-to-event of interest (e.g. death) and potential association with treatments [25]. Such analysis can provide scientific insights into clinical trials, whereby the efficacy of new therapeutics are evaluated. [4, 10, 32]. Recent technological advancements have resulted in the growing accessibility of RNA-seq data, which introduces an avenue for patient-specific survival prediction [20, 27, 31]. However, one faces the challenge that obtaining ample RNA-seq samples with labeled information for complex diseases is often constrained by biological, technical, and financial factors, especially for under-studied diseases[8, 30, 16, 33, 23, 3, 12, 15]. Contrastingly, the availability of a vast amount of unlabeled data presents an opportunity. Our goal is to develop a guided self-supervised representation learning in a tabular setting that harnesses this extensive yet mostly unlabeled, data across various cancer types, to enhance the precision of survival predictions in a way that is rooted in genetics.

[*]This work was done while interning at Genentech.

[§]Corresponding author: `hajiramezanali.ehsan@gene.com`

Table Representation Learning Workshop at NeurIPS 2023.

While cancers manifest themselves differently, many share underlying mechanisms, thereby we aim to train a single model with a number of cancer types that gives us a better capacity to extract meaningful mechanisms. Traditional methods have typically adhered to a *one disease-one model* paradigm, which inadvertently missed opportunities to use shared information between cancers and often struggled with data insufficiency for specific cancers [29, 7, 19]. Ideally, we seek to share information to tackle the heterogeneity of different cancer types. Apart from combining multiple cancer types, we also have other occasional data sources at our disposal. In the settings of our interests, clinical information, recognized for its potent prognostic indicators, stands out as a valuable source of information [20]. While clinical records are not consistently available for every patient, leveraging multitask (MT) learning across various clinical tasks can guide our use of RNA-seq data. This approach bridges the gap between genetic activity (as captured by RNA expression) and observable clinical outcomes, enabling more personalized treatment strategies.

Diving into the specifics of our methodology, we utilize RNA-seq data which inherently adopts a tabular format. Unlike image and language counterparts, RNA-seq tabular data lacks a known structure, making it difficult to design an effective augmentation method generically beneficial for survival prediction [38, 34]. This challenge is further exacerbated by the high dimensionality seen when features vastly outnumber samples for a specific target cancer [14]. Also, when trying to combine different types of data, there are often mismatches: for instance, some patients might have detailed clinical information but lack key survival data, or some subjects might have more clinical features than others.

To navigate these complexities, we have harnessed a multitask (MT)-guided self-supervised representation learning framework for tabular data, incorporating the benefits of multitask and self-supervised representation learning. Given the high dimensionality and sparsely labeled data, self-supervised learning proves essential for utilizing vast unlabeled datasets. By exploiting a novel augmentation technique tailored for tabular data [34], we divide RNA-seq input data into multiple subsets, to efficiently learn a latent representation capable of capturing crucial factors for survival prediction. Moreover, to accommodate the varying clinical information across patients, and to bridge the genetic to clinical outcome gap, multi-task guidance becomes pivotal, which also ensures seamless assimilation of labels from auxiliary tasks, further strengthening representation for primary tasks with data constraints. This paper unpacks this comprehensive approach, elucidating its potential to enhance survival prediction and hence advance personalized treatment.

2 Related Works

Survival predictions. Ching et al. [7] and Huang et al. [19] have evaluated multiple cancer types using the TCGA dataset. However, these methods train separate models for each disease, potentially overlooking shared patterns across cancer types. Furthermore, well-regarded techniques like Katzman et al. [22] and Ishwaran et al. [21], despite their promise, have mostly been tested on lower-dimensional datasets, constraining their predictive richness.

Self-supervised learning for tabular data. Representation learning on tabular data often grapples with challenges unique to its nature. A typical strategy has been to introduce noise, aiming for models to identify robust features amidst the corruption. For instance, [35] applied an autoencoder to map perturbed data instances to a latent space and then reconstructed them to their original form. However, it oversimplified the data by treating diverse context-specific features uniformly, thus potentially missing intricate relationships. A subsequent innovation was presented by Yoon et al. [38], which blended the denoising autoencoder paradigm with a classifier, which aimed to improve representations but encountered difficulties when using imbalanced binary masks on high-dimensional datasets. To address the aforementioned challenges, Ucar et al. [34] developed a novel strategy, presenting a multi-view representation concept where input features were divided into multiple subsets. More recent work [14, 2] also achieved impressive results on various scenarios. Yet, their effectiveness in survival analysis, particularly when handling the complexities of heterogeneous genomic data, is still undetermined.

3 Preliminary

Survival analysis, or time-to-event analysis, is a statistical method to examine the time until events of interest (like death or equipment failure) which is commonly used in biomedicine.

Central to survival data is the concept of censoring, which transpires when the event of interest for a subject remains unobserved— either because it has not yet been materialized (at the data cut) or due to reasons like study dropout.

Key concepts of survival analysis include **survival function** $\mathcal{S}(t)$, which represents the probability that the time-to-event exceeds some value t; and **hazard function** $h(t)$, that denotes the instantaneous occurrence rate of events, given no preceding event.

3.1 Cox Proportional Hazards Model

The Cox Proportional Hazards (CoxPH) model stands as one of the most extensively employed statistical methods for survival data analysis [9, 25, 11]. It adopts a semi-parametric approach, implying it imposes certain assumptions on the data but refrains from specifying a complete parametric form. The model can be delineated as:

$$h(t, \mathbf{X}) = h_0(t) \exp(\boldsymbol{\beta}^T \mathbf{X}), \tag{1}$$

where $h(t, \mathbf{X})$ denotes the hazard function at time t for a subject with covariates $\mathbf{X}$; $h_0(t)$ represents the baseline hazard, which corresponds to the hazard when all covariates equal zero; $\boldsymbol{\beta}$ is a coefficient vector that quantifies the covariates' impact on the hazard; $\mathbf{X}$ is a covariate or feature vector. The cornerstone of the CoxPH model is the proportional hazards postulate, which dictates that the hazard of an event for an individual remains proportional relative to another individual, in a constant manner irrespective of time.

3.2 CoxPH Loss Function

To discern the coefficients $\boldsymbol{\beta}$ of the CoxPH model, one needs to maximize the partial likelihood:

$$L(\boldsymbol{\beta}) = \prod_{i:\delta_i=1} \frac{\exp(\boldsymbol{\beta}^T \mathbf{x}_i)}{\sum_{j:t_j \geq t_i} \exp(\boldsymbol{\beta}^T \mathbf{x}_j)}, \tag{2}$$

where δ_i is the event indicator (i.e., equating to 1 if the event transpired and 0 otherwise); $\mathbf{x}_i$ and $\mathbf{x}_j$ are the covariate vectors for the i-th and j-th subjects, respectively; t_i and t_j are the observed event times for the i-th and j-th subjects, respectively. It is crucial to note that in the risk set of the denominator, that is the summation over j with $t_j \geq t_i$, both uncensored and censored subjects are included. Because even if a subject is censored at t_j, they were still "at risk" up to that time, therefore contributing to the risk set for all earlier times t_i.

Often, the negative log-partial likelihood is utilized as the loss function:

$$L(\boldsymbol{\beta}) = -\sum_{i:\delta_i=1} \left(\boldsymbol{\beta}^T \mathbf{x}_i - \log \left(\sum_{j:t_j \geq t_i} \exp(\boldsymbol{\beta}^T \mathbf{x}_j) \right) \right) \tag{3}$$

By optimizing this loss, one can uncover the values of $\boldsymbol{\beta}$ that best elucidate the relationship between the covariates and the event's hazard.

4 Methods

4.1 Problem formulating

Consider $\mathcal{C}$ as the set of various cancer types. Each $c_i \in \mathcal{C}$ is associated with a tabular input dataset $\mathbf{X}^{(c_i)} \in \mathbb{R}^{V \times J_{c_i}}$, representing the RNA-seq data. Here, V denotes the total number of genes, J_{c_i} is the number of samples specific to cancer type c_i, and $V \gg J_{c_i}$. Alongside RNA-seq input $\mathbf{x}_j^{(c_i)}$, we also have access to sparse auxiliary task labels, i.e. a sparse set of clinical features, denoted by $\{y_{j,t}^{(c_i)}\}_{t \in T_{\text{aux}}}$. Specifically, T_{aux} refers to set of available tasks that input $\mathbf{x}_j^{(c_i)}$ might have access.

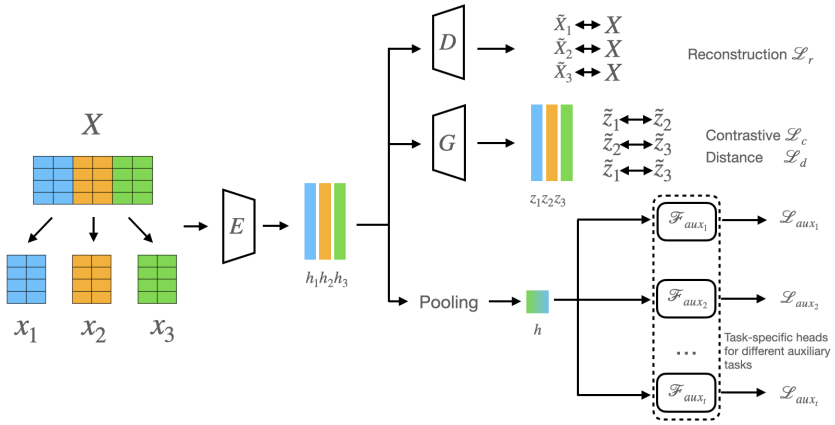

Figure 1: Guided-STab framework: Given the gene expression profile $\mathbf{X}$ where rows represent patient samples and columns denote genes, we divide the genes into K subsets, producing $\mathbf{x}_k$. These subsets are processed by encoder E to yield latent representations $\mathbf{h}_k$. A decoder D then reconstructs $\mathbf{X}$ from these subsets. For contrastive and distance loss, each $\mathbf{h}_k$ is projected to $\mathbf{z}_k$ using a projection network G. Auxiliary task predictions are made using multiple heads $\mathcal{F}_{\text{aux}_t}$ which utilize the pooled representation $\mathbf{h}$ as input.

We would like to learn informative latent representations from available RNA-seq datasets encompassing various cancer types. To enhance the predictive power for survival outcomes in rare target cancers, we intend to incorporate clinical features as guidance within a self-supervised Tabular learning framework (Guided-STab).

4.2 Guided self-supervised tabular learning

In self-supervised representation learning across the image and language domains, cropping stands out as the most effective among the commonly employed augmentation techniques [6]. Inspired by this, and exploiting a novel augmentation technique tailored for tabular data [34], we divide RNA-seq input data into multiple subsets. In the following, we describe the construction of the proposed framework, shown in 1, in which we have an encoder E, a decoder D, a projection G, and multiple auxiliary prediction heads $\mathcal{F}_{\text{aux}}$. Let's denote $\mathbf{h}_k$ and $\mathbf{z}_k$ as the latent representation and projection output of kth RNA-seq subset, respectively.

In the Guided-STab framework, every gene subset [3] is processed by the shared encoder to obtain its corresponding latent representation. A common decoder is used to reconstruct the entire RNA-seq data, effectively recovering all genes from the given subset of genes. The presence of multiple latent representations for each sample, stemming from the latent representations of each gene subset, empowers the framework to compute contrastive loss for every possible combination of pairs of projections. Pairs originating from the same samples are labeled as positive, while all others are negative. Furthermore, in order to minimize the gap between pairs of projections from the gene subsets, we incorporate a distance loss function, e.g. mean squared error (MSE). In addition to the previously mentioned self-supervised loss functions, the latent representations are further guided by predicting different auxiliary tasks, our hypothesis is that these embeddings, upon refinement, would align more closely with the primary objective of survival prediction.

The objective function of Guided-STab is:

$$\mathcal{L}_{tot} = \mathcal{L}_r + \mathcal{L}_c + \mathcal{L}_d + \sum_{t=1}^{T} \mathcal{L}_{\text{aux}_t}, \tag{4}$$

[3] The degree of overlap between neighboring gene subsets, if any, can be customized by adjusting a hyperparameter.

where $\mathcal{L}_{tot}$ is the total loss, and $\mathcal{L}_r$, $\mathcal{L}_c$, $\mathcal{L}_d$, $\mathcal{L}_{\text{aux}_t}$ are reconstruction, contrastive, distance, and task-specific losses, and T is total number of clinical tasks. The training schema can be found in Algorithm 1.

Reconstruction loss. Given a batch of RNA-seqs for a specific gene subset, denoted as $\tilde{\mathbf{X}}_k$, the shared decoder performs the reconstruction of the entire RNA-seq, represented as $\hat{\mathbf{X}}_k$. Subsequently, we can formulate the average reconstruction loss across all gene subsets as follows:

$$\mathcal{L}_r = \frac{1}{K}\sum_{k=1}^{K}\left(\frac{1}{J}\sum_{j=1}^{J}(\mathbf{x}_j - \hat{\mathbf{x}}_{j,k})^2\right) \tag{5}$$

where K represents the total number of subsets, J denotes the total number of samples within the batch, and $\hat{\mathbf{x}}_{j,k}$ denotes jth sample from $\hat{\mathbf{X}}_k$.

Contrastive loss. By combining RNA-seq data from various cancer types, we ensure the availability of a sufficient number of classes, allowing for the generation of negative samples essential for computing the contrastive loss. Specifically, we project $\mathbf{h}_{j,k}$ to $\mathbf{z}_{j,k}$ using a projection network denoted as (G). Subsequently, we consider the projections of distinct subsets of the same samples, denoted as $\{\mathbf{z}_{j,k}\}_{k=1}^{K}$, to serve as positive pairs $(\mathbf{z}_{j,k}, \mathbf{z}_{j,k'})$, while the projections of the remaining samples within the batch can be regarded as negative pairs, i.e $(\mathbf{z}_{j,k}, \mathbf{z}_{j',k'})$, where $j \neq j'$ and $k \neq k'$. The overall contrastive loss is:

$$\mathcal{L}_c = \frac{1}{\binom{K}{2}}\sum_{\substack{k,k'=1\\k\neq k'}}^{K}\left[\frac{1}{2J}\sum_{j=1}^{J}[l(\mathbf{z}_{j,k}, \mathbf{z}_{j,k'}) + l(\mathbf{z}_{j,k'}, \mathbf{z}_{j,k})]\right], \tag{6}$$

where l is the normalized temperature-scaled cross entropy (NT-Xent) loss [6], and it can be computed for a pair of projections as follows:

$$l(\mathbf{z}_{j,k}, \mathbf{z}_{j,k'}) = -\log\left(\frac{\exp(\text{sim}(\mathbf{z}_{j,k}, \mathbf{z}_{j,k'})/\tau)}{\sum_{\substack{j'=1\\j'\neq j}}^{J}\exp(\text{sim}(\mathbf{z}_{j,k}, \mathbf{z}_{j',k'})/\tau)}\right).$$

We used the dot product as the similarity metric (sim), with τ serving as a temperature parameter that scales the similarity scores. The summation in the denominator encompasses all negative samples, thus establishing a contrast between the positive and negative pairs.

Distance loss. We also utilize the Mean Squared Error (MSE) as a distance loss between pairs of projections, as the corresponding samples within the gene subsets are expected to be in close proximity to each other. The distance loss can be formulated as:

$$\mathcal{L}_d = \frac{1}{\binom{K}{2}}\sum_{\substack{k,k'=1\\k\neq k'}}^{K}\left[\frac{1}{J}\sum_{j=1}^{J}(\mathbf{z}_{j,k} - \mathbf{z}_{j,k'})^2\right], \tag{7}$$

Multitask-guidance loss. In our multitask guidance part, we initially compute the latent representation of each sample by aggregating the corresponding representations from its various subsets, denoted as $\mathbf{h}_j = \text{Aggregate}(\{\mathbf{h}_{j,k}\}_{k=1}^{K})$. We then utilize this aggregated representation with multiple Multi-Layer Perceptron (MLP) heads to generate predictions for various auxiliary clinical tasks associated with the given sample $\{y_{j,t}^{(c_i)}\}_{t\in T_{\text{aux}}}$.

For each clinical task $t \in T_{\text{aux}}$, we calculate the task-specific loss, denoted as $\mathcal{L}_{\text{aux}_t}$, using the cross-entropy loss for multi-class tasks defined as follows:

$$\mathcal{L}_{\text{aux}_t} = -\sum_{c=1}^{C_t}\mathbf{y}_{c,t}\log(p_{c,t}), \tag{8}$$

where C_t denotes the number of classes for task t. This approach enables simultaneous training on multiple clinical tasks, leveraging the shared information within the aggregated latent representations to enhance survival analysis.

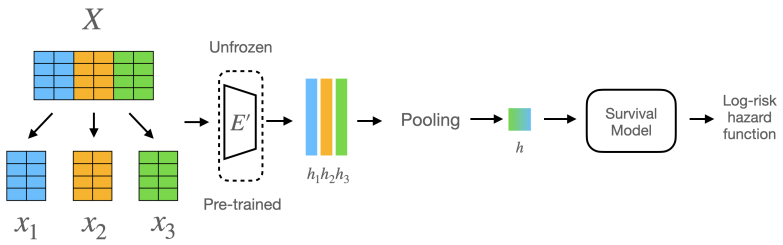

Figure 2: Survival prediction framework: A pooling layer aggregates the subsetted latent representations before passing to survival model, which essentially is an MLP layer that is specifically tailored to learn the parameters of the log-risk hazard function.

4.3 Survival prediction

After pretraining our encoder E through Guided-STab, we proceed to a fine-tuning stage tailored specifically for survival prediction. As illustrated in Figure 2, the RNA-seq data from the target cancer serves as our input. This input underwent a transformation by our encoder to generate embeddings. These embeddings, represented as $\mathbf{h}$ (an amalgamated latent representation), were then fed into an MLP. For survival prediction, we use the CoxPH loss function:

$$\mathcal{L}(\boldsymbol{\beta}) = -\sum_{i:\delta_i=1}\left[\boldsymbol{\beta}^T\mathbf{h}_i - \log\left(\sum_{j:t_j\geq t_i}\exp(\boldsymbol{\beta}^T\mathbf{h}_j)\right)\right] \tag{9}$$

5 Experiment and Results

Data. We sourced our data from the Cancer Genome Atlas (TCGA) [20]. This included nine major cancer types with a total of $4,888$ samples. To process the RNA-seq data, we applied the $\log(\text{RSEM}+1)$ transformation. By focusing on the top $1,000$ varied genes, we ensured that our analysis honed in on the most relevant genetic markers. For clinical task selection, we employed SHAP [28]. This resulted in a total of six out of 58 tasks: age, disease stage, tumor stage, metastasis stage, race, and radiation from the top-ranked importance score. It is worth noting that clinical features have missing values. For age classification, we normalized the data into four scales, determined by quartile cuts, which transformed all auxiliary clinical tasks into multi-class classification problems.

5.1 Experiment Setup

In the scope of this study, we rigorously benchmarked our proposed methodology against two main multiple baselines. Within the supervised algorithms, we selected RSF [21] and DeepSurv [22] for our comparisons. These models were primarily dependent on the RNA-seq data of the target cancer type for survival prediction. On the unsupervised methods, we incorporated the Variational autoencoder (VAE) [24] and AE. The encoder shown in Figure 1 was replaced by VAE and AE, ensuring all other components remained consistent to maintain the integrity of the comparison. Throughout this unsupervised learning phase, all the available RNA-seq data and auxiliary descriptors from other cancer types were used. Furthermore, the model was integrated with auxiliary tasks in the supervised models, resulting in configurations termed RSF + Guided-STab and Guided-STab. For clarity in subsequent discussions, *Guided-STab* will be referred to as our primary methodology.

Evaluation. Our main goal was to assess the performance of our method on target cancers, and we chose the Concordance index (C-index) as the evaluation metric [17]. The evaluation relied on a 5-fold cross-validation, which was also stratified based on events. To start, we divided the target cancer task into five train/test folds. When it came to pretraining, we employed all RNA-seq data, excluding a 20% sample set designated for testing. Similarly, we used all available clinical information, excluding the 20% test target samples. For survival prediction, we applied the trained encoder exclusively with the target cancer RNA-seq.

Table 1: Comparative study of survival analysis performance using C-index.

Models	**BRCA** (1079)	**HNSC** (515)	**LUAD** (502)	**LUSC** (478)	**BLCA** (406)
RSF	0.638 ± 0.038	0.589 ± 0.038	0.593 ± 0.028	0.525 ± 0.039	0.624 ± 0.030
DeepSurv	0.710 ± 0.041	0.607 ± 0.050	0.591 ± 0.049	0.556 ± 0.047	0.628 ± 0.019
AE	0.732 ± 0.052	0.610 ± 0.053	0.612 ± 0.014	0.579 ± 0.043	0.649 ± 0.024
VAE	0.717 ± 0.064	0.574 ± 0.045	0.609 ± 0.026	0.589 ± 0.061	0.622 ± 0.036
Guided-STab (Ours)	**0.740 ± 0.028**	**0.645 ± 0.047**	**0.631 ± 0.024**	**0.606 ± 0.021**	**0.669 ± 0.028**

5.2 Result

In our comprehensive evaluation across multiple cancer types, the superiority of our approach became evident. As shown in Table 1, our method proved its robustness and efficacy from the performance on notoriously challenging datasets. With the HNSC dataset, our strategy outperformed the second-best method by about 6.26%. In the case of LUAD, a disease for which many studies often report C-indices below 0.6, our method showcased an enhancement of roughly 3.10% over the competing method. For the BLCA dataset, our method reflected a performance gain of around 3.08% over the second-best approach. Equally impressive was our performance on the LUSC dataset, where our method marked an improvement of nearly 2.89%, further emphasizing its potential in dealing with complex diseases. Notably, with the BRCA dataset, which included 1079 data samples (nearly double that of other datasets), our method's improvement was a relatively conservative 1.09% over the subsequent best approach. This more modest improvement for BRCA, despite its larger dataset, suggests the challenges of extracting significant advancements in areas with abundant data and potentially higher baseline performances. Collectively, these results underscore the advanced capabilities of our method in survival prediction.

5.2.1 The Importance of Auxiliary Task Guidance

To investigate further the impact of incorporating an auxiliary task, we conducted an experiment where these tasks were omitted from the latent space. We performed the ablation studies on both our methods and RSF. The comparative outcomes are illustrated in Table 2 and Figure A1 in Appendix.

Table 2: Evaluating the effect of guidance in the proposed Guided-STab.

Models	BRCA	HNSC	LUAD	LUSC	BLCA
Ours w/o guidance[4]	0.732	0.628	0.625	0.599	0.646
Guided-Stab	**0.740**	**0.645**	**0.631**	**0.606**	**0.669**

The results highlight a consistent trend: the presence of auxiliary tasks ("w aux") consistently outperformed the absence of them ("w/o aux") in both scenarios. This observation underscores the important role that auxiliary tasks play in enhancing the predictive power of the embeddings. Their inclusion not only augments the information captured in the latent space but also bolsters the robustness of survival predictions. Hence, the findings emphasize the critical advantage of incorporating auxiliary tasks when generating representations tailored for survival analysis. Interestingly, the impact of auxiliary guidance was more pronounced in our method compared to RSF. While our approach fine-tuned the encoder, optimizing embeddings for survival prediction, RSF used fixed embeddings. This adaptability in our method, absent in RSF, likely accounts for the observed performance differences when integrating auxiliary tasks as guidance.

5.2.2 Ablation Study

Our ablation study examined three core aspects: the impact of varying the number of subsets, the consequences of removing specific loss functions, and the effects of excluding certain auxiliary tasks. The observations are as follows with results presented in Table 3 and 4:

Number of gene subsets. When looking at the effect of the number of subsets (Table 3), Guided-STab (4 subsets) consistently outperformed the 2-subset and 3-subset methods across all cancer types, highlighting its ability to capture broader and more intricate patterns in the data. While the performance for 2-subset and 3-subset methods varied across different cancers, indicating that some cancers might benefit more from an intermediate number of subsets, the 4-subset approach suggests its robustness in addressing the heterogeneity inherent in all the datasets for survival prediction.

[4]When we refer to *ours w/o guidance* in this paper, we are essentially utilizing *SubTab*[34] as the underlying framework.

Table 3: Effect of the number of gene subsets on Guided-STab framework measured in C-index.

Num. of Subsets	BRCA	HNSC	LUAD	LUSC	BLCA
1-subset	0.732 ± 0.052	0.610 ± 0.053	0.612 ± 0.014	0.579 ± 0.043	0.649 ± 0.024
2-subset	0.699 ± 0.041	0.618 ± 0.039	0.609 ± 0.016	0.587 ± 0.039	0.640 ± 0.024
3-subset	0.721 ± 0.040	0.613 ± 0.044	0.610 ± 0.024	0.582 ± 0.034	0.629 ± 0.022
4-subset (ours)	**0.740** ± 0.028	**0.645** ± 0.047	**0.631** ± 0.024	**0.606** ± 0.021	**0.669** ± 0.028

Table 4: Evaluating the effect of each individual loss on the performance of Guided-STab framework measured in C-index.

Losses	BRCA	HNSC	LUAD	LUSC	BLCA
w/o dist	0.729 ± 0.026	0.639 ± 0.052	0.631 ± 0.024	0.591 ± 0.006	0.664 ± 0.041
w/o contra	0.716 ± 0.024	0.634 ± 0.050	0.625 ± 0.033	0.601 ± 0.013	0.664 ± 0.032
w/o age	0.730 ± 0.034	0.640 ± 0.040	0.616 ± 0.028	0.597 ± 0.025	0.661 ± 0.029
w/o dis. stage	0.734 ± 0.041	0.644 ± 0.045	0.620 ± 0.023	0.599 ± 0.026	0.669 ± 0.028
w/o tumor stage	0.740 ± 0.028	0.638 ± 0.045	0.631 ± 0.024	0.603 ± 0.015	0.650 ± 0.036
w/o metastasis	0.734 ± 0.025	0.642 ± 0.041	0.617 ± 0.032	0.601 ± 0.018	0.667 ± 0.037
w/o race	0.725 ± 0.028	0.637 ± 0.046	0.623 ± 0.031	0.598 ± 0.020	0.663 ± 0.040
w/o radiation	0.722 ± 0.036	0.636 ± 0.046	0.618 ± 0.024	0.599 ± 0.025	0.653 ± 0.031
Guided-STab	**0.740** ± 0.028	**0.645** ± 0.047	**0.631** ± 0.024	**0.606** ± 0.021	**0.669** ± 0.028

Effect of projection. Removing the contrastive loss consistently resulted in a performance drop across all cancer types (Table 4). Contrastive learning is crucial for representation learning as it encourages the model to learn embeddings where similar instances are pulled closer together and dissimilar instances are pushed apart. This is important in heterogeneous datasets like those of different cancer types, where distinguishing between survival outcomes based on subtle differences in features is essential. Although the removal of distance loss was less impactful, it also led to a performance drop in the majority of the cases.

Effect of individual auxiliary tasks. In our ablation study on the removal of different auxiliary tasks shown in Table 4, the importance of the tumor stage was clear, especially for the BLCA dataset which saw a 2.84 % decline in performance without it. However, other datasets like LUAD showed only a minor drop, suggesting that universal tumor stage embeddings might not serve all cancers equally. For instance, in LUAD, other factors might be more pivotal for survival prediction than the tumor stage. Similarly, the LUSC disease, known for its more challenging nature and complexity, showed little change when the disease stage was removed.

Interestingly, BRCA performed better without the tumor stage, and BLCA peaked when the disease stage was excluded. Note that we updated the best score across all the studies. These findings underscore the idea that a one-size-fits-all approach might not be optimal. Instead, a more context-specific strategy, perhaps a hierarchical or adaptive one, should be considered. This approach would deploy some auxiliary tasks universally while customizing others based on the specific needs of individual datasets.

Upon evaluating the impact of various auxiliary tasks on our model's performance, it shows that the radiation task held significant importance. With an average performance decrease of 0.0126 upon its removal, radiation prediction consistently proved to be the most influential. This suggests that incorporating radiation information within Guided-STab universally offers a substantial enhancement to its efficacy on survival analysis across the datasets.

6 Future Work

There are several directions for further exploration: 1) Integration of prior knowledge such as tissue-specific gene-gene interactions, which can be modeled as graphs [26]. This will allow for a more comprehensive understanding and harnessing of underlying biological processes [13, 18]; 2) Incorporating relational multitask settings inspired by Cao et al. [5], Hajiramezanali et al. [15], by creating knowledge graphs from auxiliary tasks, we hope to not only better address the issues of data scarcity but also provide solution for disentangling genomic relationships; and 3) Adding interpretability can aid the scientific understanding of intricate genomics interactions, as well as informing clinical decisions in the future.

7 Acknowledgments

We sincerely thank the Roche Advanced Analytics Network (RAAN) for their support, which was vital to the development of this research.

References

[1] Behrooz Azarkhalili, Ali Saberi, Hamidreza Chitsaz, and Ali Sharifi-Zarchi. Deepathology: deep multi-task learning for inferring molecular pathology from cancer transcriptome. *Scientific reports*, 9(1):16526, 2019.

[2] Dara Bahri, Heinrich Jiang, Yi Tay, and Donald Metzler. Scarf: Self-supervised contrastive learning using random feature corruption. *arXiv preprint arXiv:2106.15147*, 2021.

[3] Omid Bazgir and James Lu. Refined-cnn framework for survival prediction with high-dimensional features. *Iscience*, 26(9), 2023.

[4] Lucie Biard, Anne Bergeron, Vincent Lévy, and Sylvie Chevret. Bayesian survival analysis for early detection of treatment effects in phase 3 clinical trials. *Contemporary Clinical Trials Communications*, 21:100709, 2021.

[5] Kaidi Cao, Jiaxuan You, and Jure Leskovec. Relational multi-task learning: Modeling relations between data and tasks. *arXiv preprint arXiv:2303.07666*, 2023.

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[7] Travers Ching, Xun Zhu, and Lana X Garmire. Cox-nnet: an artificial neural network method for prognosis prediction of high-throughput omics data. *PLoS computational biology*, 14(4): e1006076, 2018.

[8] Ana Conesa, Pedro Madrigal, Sonia Tarazona, David Gomez-Cabrero, Alejandra Cervera, Andrew McPherson, Michał Wojciech Szcześniak, Daniel J Gaffney, Laura L Elo, Xuegong Zhang, et al. A survey of best practices for rna-seq data analysis. *Genome biology*, 17(1):1–19, 2016.

[9] David R Cox. Regression models and life-tables. *Journal of the Royal Statistical Society: Series B (Methodological)*, 34(2):187–202, 1972.

[10] J Emmerson and JM Brown. Understanding survival analysis in clinical trials. *Clinical Oncology*, 33(1):12–14, 2021.

[11] Jiang Gui and Hongzhe Li. Penalized cox regression analysis in the high-dimensional and low-sample size settings, with applications to microarray gene expression data. *Bioinformatics*, 21(13):3001–3008, 2005.

[12] Ehsan Hajiramezanali, Siamak Zamani Dadaneh, Alireza Karbalayghareh, Mingyuan Zhou, and Xiaoning Qian. Bayesian multi-domain learning for cancer subtype discovery from next-generation sequencing count data. *Advances in Neural Information Processing Systems*, 31, 2018.

[13] Ehsan Hajiramezanali, Arman Hasanzadeh, Nick Duffield, Krishna Narayanan, and Xiaoning Qian. Bayrel: Bayesian relational learning for multi-omics data integration. In *NeurIPS*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/df5511886da327a5e2877c3cd733d9d7-Abstract.html.

[14] Ehsan Hajiramezanali, Nathaniel Lee Diamant, Gabriele Scalia, and Max W Shen. Stab: Self-supervised learning for tabular data. In *NeurIPS 2022 First Table Representation Workshop*, 2022.

[15] Ehsan Hajiramezanali, Talip Ucar, and Lindsay Edwards. Bayesian relational generative model for scalable multi-modal learning, 2022. URL https://openreview.net/forum?id=bVT5w39X0a.

[16] Ashraful Haque, Jessica Engel, Sarah A Teichmann, and Tapio Lönnberg. A practical guide to single-cell rna-sequencing for biomedical research and clinical applications. *Genome medicine*, 9(1):1–12, 2017.

[17] Frank E Harrell Jr, Kerry L Lee, Robert M Califf, David B Pryor, and Robert A Rosati. Regression modelling strategies for improved prognostic prediction. *Statistics in medicine*, 3(2): 143–152, 1984.

[18] Arman Hasanzadeh, Ehsan Hajiramezanali, Nick Duffield, and Xiaoning Qian. Morel: Multi-omics relational learning. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=DnG75_KyHjX`.

[19] Zhi Huang, Travis S Johnson, Zhi Han, Bryan Helm, Sha Cao, Chi Zhang, Paul Salama, Maher Rizkalla, Christina Y Yu, Jun Cheng, et al. Deep learning-based cancer survival prognosis from rna-seq data: approaches and evaluations. *BMC medical genomics*, 13:1–12, 2020.

[20] Carolyn Hutter and Jean Claude Zenklusen. The cancer genome atlas: creating lasting value beyond its data. *Cell*, 173(2):283–285, 2018.

[21] Hemant Ishwaran, Udaya B Kogalur, Eugene H Blackstone, and Michael S Lauer. Random survival forests. 2008.

[22] Jared L Katzman, Uri Shaham, Alexander Cloninger, Jonathan Bates, Tingting Jiang, and Yuval Kluger. Deepsurv: personalized treatment recommender system using a cox proportional hazards deep neural network. *BMC medical research methodology*, 18(1):1–12, 2018.

[23] Ishleen Kaur, MN Doja, and Tanvir Ahmad. Data mining and machine learning in cancer survival research: an overview and future recommendations. *Journal of Biomedical Informatics*, 128:104026, 2022.

[24] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[25] David G Kleinbaum and Mitchel Klein. *Survival analysis a self-learning text*. Springer, 1996.

[26] Cui-Xiang Lin, Hong-Dong Li, Chao Deng, Yuanfang Guan, and Jianxin Wang. Tissuenexus: a database of human tissue functional gene networks built with a large compendium of curated rna-seq data. *Nucleic acids research*, 50(D1):D710–D718, 2022.

[27] John Lonsdale, Jeffrey Thomas, Mike Salvatore, Rebecca Phillips, Edmund Lo, Saboor Shad, Richard Hasz, Gary Walters, Fernando Garcia, Nancy Young, et al. The genotype-tissue expression (gtex) project. *Nature genetics*, 45(6):580–585, 2013.

[28] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30, 2017.

[29] Pooya Mobadersany, Safoora Yousefi, Mohamed Amgad, David A Gutman, Jill S Barnholtz-Sloan, José E Velázquez Vega, Daniel J Brat, and Lee AD Cooper. Predicting cancer outcomes from histology and genomics using convolutional networks. *Proceedings of the National Academy of Sciences*, 115(13):E2970–E2979, 2018.

[30] Jennifer E Posey, Jill A Rosenfeld, Regis A James, Matthew Bainbridge, Zhiyv Niu, Xia Wang, Shweta Dhar, Wojciech Wiszniewski, Zeynep HC Akdemir, Tomasz Gambin, et al. Molecular diagnostic experience of whole-exome sequencing in adult patients. *Genetics in Medicine*, 18 (7):678–685, 2016.

[31] Aviv Regev, Sarah A Teichmann, Eric S Lander, Ido Amit, Christophe Benoist, Ewan Birney, Bernd Bodenmiller, Peter Campbell, Piero Carninci, Menna Clatworthy, et al. The human cell atlas. *elife*, 6:e27041, 2017.

[32] Paul G Richardson, Shaji K Kumar, Tamás Masszi, Norbert Grzasko, Nizar J Bahlis, Markus Hansson, Luděk Pour, Irwindeep Sandhu, Peter Ganly, Bartrum W Baker, et al. Final overall survival analysis of the tourmaline-mm1 phase iii trial of ixazomib, lenalidomide, and dexamethasone in patients with relapsed or refractory multiple myeloma. *Journal of Clinical Oncology*, 39(22):2430–2442, 2021.

[33] Rory Stark, Marta Grzelak, and James Hadfield. Rna sequencing: the teenage years. *Nature Reviews Genetics*, 20(11):631–656, 2019.

[34] Talip Ucar, Ehsan Hajiramezanali, and Lindsay Edwards. Subtab: Subsetting features of tabular data for self-supervised representation learning. *Advances in Neural Information Processing Systems*, 34:18853–18865, 2021.

[35] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103, 2008.

[36] You Wu, Qiao Liu, Yue Qiu, and Lei Xie. Deep learning prediction of chemical-induced dose-dependent and context-specific multiplex phenotype responses and its application to personalized alzheimer's disease drug repurposing. *PLOS Computational Biology*, 18(8): e1010367, 2022.

[37] You Wu, Qiao Liu, and Lei Xie. Hierarchical multi-omics data integration and modeling predict cell-specific chemical proteomics and drug responses. *Cell Reports Methods*, 3(4), 2023.

[38] Jinsung Yoon, Yao Zhang, James Jordon, and Mihaela van der Schaar. Vime: Extending the success of self-and semi-supervised learning to tabular domain. *Advances in Neural Information Processing Systems*, 33:11033–11043, 2020.

A Algorithm

Algorithm 1 Training Schema

Require: Input matrix $\mathbf{X}^{(C)} \in \mathbb{R}^{V \times \sum_{c_i \in C} J_{c_i}}$, where C is the set of all data sources stacked together.

for epoch = 1 **to** $n_{\text{warm\_up}}$ **do**
  for t = 1 **to** $\frac{N}{\sum_{c_i \in C} J_{c_i}}$ **do**
    Divide minibatch $\mathbf{X}^{(C)}$ to K subsets $\{\tilde{\mathbf{X}}_1, \tilde{\mathbf{X}}_2, \ldots, \tilde{\mathbf{X}}_k\}$
    Update E with $L_r + L_c + L_d$
  end for
end for
for epoch = $n_{\text{warm\_up}}$ **to** n_{total} **do**
  for t = 1 **to** $\frac{N}{\sum_{c_i \in C} J_{c_i}}$ **do**
    Divide minibatch $\mathbf{X}^{(C)}$ to K subsets $\{\tilde{\mathbf{X}}_1, \tilde{\mathbf{X}}_2, \ldots, \tilde{\mathbf{X}}_k\}$
    Update E with $L_r + L_c + L_d + \sum_{t=1}^{T} \mathcal{L}_{\text{aux}_i}$
  end for
end for

B Additional Related Works

Multitask learning. Multitask learning (MTL) has gained prominence as an effective strategy to simultaneously learn multiple related tasks, capitalizing on the shared information between them. Azarkhalili et al. [1] illustrated the enhanced accuracy and interpretability of drug prediction models by leveraging multitask learning across multiple drugs. Furthermore, Wu et al. [36, 37] demonstrated the utility of MTL in predicting multiple phenotypes from genomics data, underscoring its ability to utilize shared patterns across related phenotypic tasks like drug response and perturbation prediction. However, a noticeable gap remains, as neither study has explored the realm of survival prediction. More recently, Cao et al. [5] advanced conventional multitask learning by constructing a knowledge graph that linked data points and tasks, which maximized the label information from auxiliary tasks. However, its application was limited to graph data, which could not be directly adapted to tabular form.

C Experiment Setup

For our experiments, we utilized an NVIDIA DGX with CUDA version 11.4 for training the model. The implementation was carried out in Pytorch. The task-specific hyperparameters are shown in Table A1, and below is a compact representation of the hyperparameters and settings used globally:

Table A1: Task-specific hyperparameters in the proposed Guided-Stab.

Parameters	BRCA	HNSC	LUAD	LUSC	BLCA
Hidden dim	[256]	[128]	[256]	[256]	[256]
Warm-up/Train epochs	0/60	20/40	30/50	20/40	0/40

Table A2: Global Hyperparameters and Settings Overview

Parameter/Setting	**Value**
Platform	NVIDIA DGX, CUDA 11.4
Pretraining lr	0.001
Dropout	0.2
Batch Size	128
Number of Subsets (Overlap)	4 (0.75)
Aggregation	Concatenation
Noise (Masking, Level)	Gaussian (0.3, 0.1)
Survival MLP (lr, Hidden)	0.00008, [512,64]
Activation	ReLU

Table A3: Abbreviations

Abbreviation	**Description**
BRCA	Breast Invasive Carcinoma
HNSC	Head and Neck Squamous Cell Carcinoma
LUAD	Lung Adenocarcinoma
LUSC	Lung Squamous Cell Carcinoma
BLCA	Bladder Urothelial Carcinoma

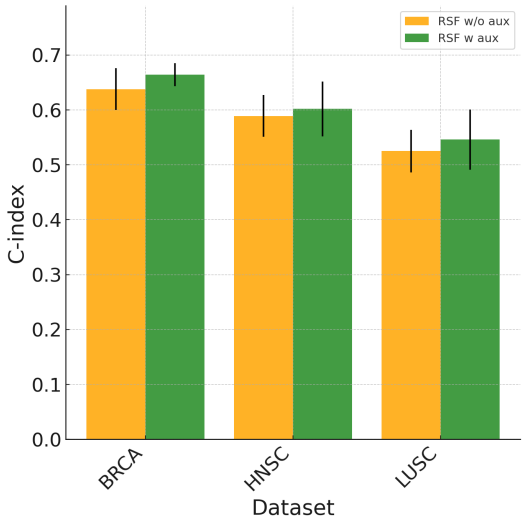

Figure A1: Comparison of RSF and RSF + Guided-STab