Synergies between Case-based Reasoning and Deep Learning for Survival Analysis in Oncology

Isabelle Bichindaritz [1][0000-0003-1712-490X] and Guanghui Liu [1][0000-0002-1135-2939]

[1] Department of Computer Science, State University of New York at Oswego, New York, USA
ibichind@oswego.edu, guanghui.liu@oswego.edu

Abstract. Survival analysis is a field of statistics specialized in making predictions about the survival length of patients, even though it can be applied to the prediction of any future event. It is routinely used in medical research to stratify patients in groups based on risk, such as high-risk groups and low-risk groups, and has paramount important in patient stratification and treatment. Recently, deep neural networks (DNNs) have raised considerable attention for survival analysis because of their non-linear nature and their excellent ability to predict survival, in comparison to statistical methods. In this domain, case-based survival methods have started to by applied as well, with some success. It is therefore interesting to study how to synergistically combine the two for improved performance for several reasons. From the case-based reasoning standpoint, the deep neural network can detect deep similarity between cases with a time-to-event structure and from the DNN standpoint, case-based reasoning can provide the glass-box approach that remedies the "black box" label attached to them. In this study, we propose a synergy between case-based reasoning and Long Short-Term Memory (LSTM) model for survival prediction in oncology. In this deep survival model network, the total loss function combines four different factors and uses an adaptive weights approach to combine the four loss terms. The network learns a prototype layer during training which naturally comes with an explanation for each prediction. This study employs cross-validation and the concordance index for assessing the survival prediction performance and demonstrate on two cancer methylation data sets that the developed approach is effective.

Keywords: Survival Analysis, Deep Network, Case-based Reasoning, Objective Loss, Explainable Model.

1 Introduction

Cancer is the most common disease in the world. Because genetic factors have been associated with this disease with a preponderance of evidence, genomic data are key to understand the complex biological mechanisms of cancer patient survival. This approach could lead to the development of new treatments for patients and improved survival predictions. An easily measurable genomic factor is the DNA methylation process. DNA methylation levels exhibit differential expressions in a variety of tissues

[27]. One goal of cancer studies refers to gaining the ability to identify prediction-related elements to determine the survival length of a patient, thereby allowing clinical personnel to perform early treatment decision-making. Prediction-related disease signatures are critical to split cases between risk groups for personalized cancer management, which could avoid either overtreatment or under treatment. For instance, cases classified into the high-risk group may benefit from closer follow-up, more aggressive therapies, and advanced care planning [30]. Consequently, to explore the utility of DNA methylation data for cancer diagnosis, it is very useful to analyze DNA methylation of tumors from cases with cancers to identify potential cancer-specific survival risk.

In case-based reasoning (CBR), similarity assessment can be complex, in particular in domains involving temporal or sequential data. In bioinformatics in particular, most biological data are high dimensional and with low-sample size. To overcome the high-dimensional feature space and low-sample size problem in bioinformatics, dimensionality reduction techniques are often used to reduce the dimension of the input data. In particular, deep feature selection was developed to identify discriminative features in deep learning models [7]. This problem has been well studied in deep learning, where model overfitting often occurs because gradients tend to have high variance in back-propagation.

In domains involving temporal or sequential data, deep learning models can be advantageous to perform similarity assessment. As a matter of fact, deep Learning techniques can be used directly in survival analysis to learn the hazard function and create deep models [1]. However, if the input and output are understood, the processing that occurs in-between is obscure, so that a black- box effect in DNNs is alluded to. The large number of parameters and the typical non-linearity of the activation functions are the main reasons why this task is practically impossible. Nevertheless, interpretable approaches are necessary in medicine because users are ultimately responsible for their clinical decisions and therefore need to make informed decisions [19]. In survival analysis, the model interpretability is more of a concern than simply predicting patient survival with high accuracy. Therefore, the ability to provide explicit model interpretation in deep neural networks remains highly desirable in survival analysis.

In this study, we create a synergistic system between case-based reasoning and deep learning for survival analysis. The contribution of deep learning is two-fold. Firstly, an autoencoder reduces the dimensionality of the input space. Our purpose for using the encoder layers is to reduce the dimensionality of the original input features. Secondly, a Long-Short Term Memory (LSTM) learns the similarity between input cases and test cases. The survival prediction architecture is capable of explaining its own reasoning process. The learned model naturally provides explanations for each prediction, and these explanations are faithful to what the network is actually computing. An architecture is used to encode its own explanation in contrast to creating an explanation for a previously trained black-box model. We create a prototype layer, where each prototype corresponds to a case, to store the weight vector following the encoded input, and to receive the output from the encoder layers. The prototype layer, inspired by case-based reasoning, utilizes the strategy of the nearest distance retrieval in case-based reasoning (CBR) to provide a useful insight into the inner workings of the deep network. We can use this prototype layer to explain the input data features.

The contributions of this paper are the following:

1) The synergy between case-based reasoning and deep learning is explored in the context of survival analysis, which is a very different machine learning task from classification or prediction. Very few CBR systems have tackled this task.
2) The deep learning architecture used, LSTM, has not been used in synergy with CBR, to the best of our knowledge. LSTM belongs to the recurrent neural networks family and excels in sequence and temporal data analysis.
3) The application to methylation data in oncology has been rarely studied from a CBR standpoint. However, it is of growing importance in bioinformatics, in which this type of data has been shown to better classify and predict many diseases than gene expression alone.
4) Unlike many approaches to the synergy between CBR and deep learning, the proposed system adopts a balanced approach between the two, since the deep learning model mostly performs deep similarity learning, taking temporal data into account. CBR is not used in the system solely for explainability.

2 Research Background

DNA methylation has recently become more prevalent in genetic research in oncology. This paper proposes to apply these findings to the study of DNA methylation signatures for cancer prognostic survival analysis. Cancer cases can be divided into two categories i.e., censored cases and non-censored cases [2]. For censored cases, the death events were not observed during the follow-up period, and thus their genuine survival times are longer than the recorded data, while for non-censored cases their recorded survival times are the exact time from initial diagnosis to event – very often the event is death.

Several survival analysis approaches have been proposed in the literature. LASSO method [18][25] applies the lasso feature selection method for selecting the parts associated with cancer prediction. Random Survival Forests (RSF) [10] calculates a random forest with the log-rank test as the splitting standard. Though much progress has been made using above approaches, Yet the predicting performance of the previously proposed approaches remains far from satisfying, and room remains for subsequent advancement.

The deep learning models overcome many of the restrictions of Cox-based models like the proportionality assumption. DeepSurv [13] was developed with a cutting-edge deep neural network. It is based on the Cox proportional hazards method associated with a deep neural network to perform a prediction of time-to-event and facilitate risk stratification with the goal of enabling treatment efficacy by providing individual treatment suggestions [14]. However, DeepSurv lacked interpretability. It was urgent to propose interpretable nonlinear models for survival prediction.

3 Related Work

3.1 Case-based reasoning

Case-based reasoning (CBR) is a method of reasoning based on analogy. Its fundamental idea is to reuse similar previous experiences in order to solve new problems. The CBR methods have in common the following processes: retrieve, reuse, revise, and retain. The most used case retrieval strategy is the nearest neighbor strategy or *k*-nearest neighbors algorithm (*k*NN). *k*NN is one of the most explainable algorithms and belongs to instance-based learners, for which decisions are made by similarity between a new case and solved retrieved cases, which can serve as explanations for a system recommendation. CBR within the domain of microarray analysis is mostly unexplored, especially for epigenetic data. The primary foundation for CBR is its ability to consistently update from new cases, and to adapt prior solutions to a new problem. Within microarray analysis, however, problems exist that render updating and adaptation particularly difficult. The first problem is the high dimensionality with few samples. There are thousands of features for a small subset of samples (specifically tens of thousands for the standard chipset used in DNA methylation), and these samples are often imbalanced between cases and controls. Therefore, little work has been done so far in genetic survival analysis with case-based reasoning, We can cite Karmen et al., who calculate similarity based on survival functions [12]. Bartlett et al. (2021) consider clinical covariates when retrieving genetic cases for case-based survival prediction [2].

3.2 Synergies between Deep Learning and CBR

A number of approaches have been proposed to combine case-based reasoning and deep learning. Approaches range from resorting to deep learning in subtasks, to resorting to CBR to make deep learning more explainable. In the former approach, for example, several systems use deep learning for some tasks within a case-based reasoning architecture. Eisenstadt et al. classify design cases from labels to select most relevant cases during retrieval [8]. In the latter approach, deep learning systems mostly resort to CBR to provide explanations of their reasoning processes (XAI) [22, 23, 5]. Li et al. construct a prototype layer by adding an autoencoder to deep convolutional networks [17]. Their application processed image data, for which convolutional neural networks are particularly adapted. By contrast, our approach fits clinical and multi-omics data, using LSTM as the main deep learning method, and performs survival prediction tasks. Our approach can also tackle classical classification and regression tasks since LSTM can be adjusted for that purpose as well, even though they excel particularly on data having a serial form, such as time series and other forms of sequences. Several deep learning systems learn prototypes for grouping input cases and explaining deep learning results [15, 16, 11]. Our system uses deep learning methods to encode each case in a prototype, not for grouping several cases into a prototype. The prototype provides a representation of a case after encoding features. Although our system could learn prototypes for grouping, we prefer to keep each case separate in the current system.

3.3 Survival Analysis

Survival analysis is considered as a specific machine learning task predicting a time to event based on incomplete data, which refers to a mix of censored and uncensored data. Although it performs a prediction into the future, it is quite different from forecasting as well as from classification or prediction. Very specific machine learning models, mostly from statistics, have been applied to this task with the goal of evaluating the risk of patients into risk categories to reach an event in the future. Cox proportional hazard model [18] is one of the most popular survival prediction models. Recently, based on the Cox model, several regularization approaches have been proposed in the literature. The Least Absolute Shrinkage and Selection Operator COX model (LASSO-COX) [24, 25, 28] applies the lasso feature selection method for selecting parts associated with carcinoma prediction. Random survival forests (RSF) [10] calculates a random forest with the log-rank test as the splitting standard. It determines the cumulative hazards of the leaf nodes while averaging them over the totality of elements. Cox regression with neural networks by a one hidden layer multilayer perceptron (MLP) [29] was proposed to replace the linear predictor of the Cox model. Some novel networks were suggested to be capable of outperforming typical Cox models [1]. DeepSurv [13, 14] refers to a deep Cox proportional hazards neural network as well as a survival approach to model interacting processes of a case's covariates and treatment modalities for providing individual treatment suggestions. DeepSurv is developed upon Cox proportional assumption with a cutting-edge deep neural network. MTLSA [17] is a recently proposed model which regards survival analysis to be a multi-task learning issue. Following in this trend, Bichindaritz et al. [4] proposed an adaptive multi-task learning method, which combines the Cox loss task with the ordinal loss task, for survival prediction of breast cancer patients using multi-modal learning to integrate gene expression and methylation data instead of performing survival analysis on each feature data set. However, these models lacked interpretability.

4 Methods

In survival analysis, prediction of the time duration until a certain event occurs is the goal and the death of a cancer case is the event of interest in this study. We propose a synergy between CBR and deep learning to achieve this goal. The model learning process, highlighted in its architecture (see Fig. 1), comprises three stages: prototype learning to encode each case into a compact representation, similarity learning through LSTM model training, and survival prediction. The trained model then can be applied to new input cases, also referred to as test cases, for survival prediction (see Fig. 2).

4.1 Autoencoder

We use an autoencoder (an encoder and a decoder) with the leaky ReLU activation function in all autoencoder layers.

The autoencoder is used to reduce the dimensionality of the input and to learn useful features for prediction; then the encoded input is used to produce a deep Cox model through the prototype layer. The prototype layer receives the output from the encoder. Because the prototype layer output vectors live in the same space as the encoded inputs, we can feed these vectors into the decoder and visualize the learned custom network throughout the training process. In case-based reasoning, to determine the solution of new problems, the process of case retrieval uses a similarity function to find some similar problems and their solutions from the historical case base. Similarity functions are generally obtained by calculating their distances in the feature space. In this system, we minimize the distance between the output and input of the prototype layer during the model iteration training. The output of the prototype layer can then be used to interpret the input data features. This property can interpret how the network reaches its predictions and visualize the learning process.

In fact, the purpose of creating the prototype layer is to obtain a dimensionality reduction vector of the original input by autonomously training a model to represent and explain the original input. For each training sample, the linear expression of each feature was calculated and used to construct one prototypical case. Each prototype case would then represent typical DNA methylation patterns present in different samples. The Euclidean distance between each encoder from the case base and its respective prototype is used to determine how similar the prototype is to its own. This will appropriately determine how well the case fits a collection of unsolved cases during the prediction.

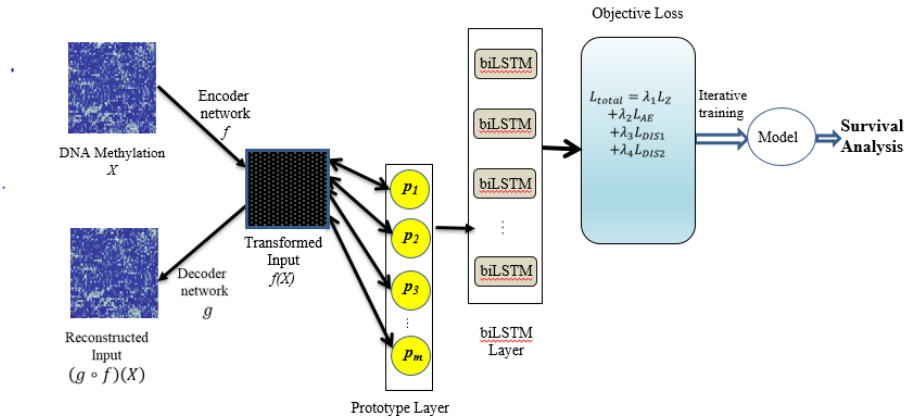


Fig. 1. Illustration of the proposed model training framework

4.2 biLSTM

A bidirectional Long Short-Term Memory (biLSTM) [9] is then trained as the last output layer.

In this deep survival model network, the total loss function consists of four terms: the negative log partial likelihood function of the Cox model, the autoencoder loss, and two required distances. These distances ensure that every feature vector in the original input looks like at least one of the prototype layer feature vectors and that every prototype layer feature vector looks like at least one of the feature vectors in the original input. We use an adaptive weights approach to combine the four loss terms. The network that the prototype layer learns during training naturally comes with an explanation for each prediction.

The negative log partial likelihood function of the Cox hazard model is defined as follows by Sy and Taylor [5]:

$$L_Z(\theta) = -\sum_{i=1}^{n} \delta_i \left(\theta^T x_i - log \sum_{j \in R(t_i)} exp(\theta^T x_j)\right) \tag{1}$$

where $x = (x_1, x_2, \cdots, x_n)$ corresponds to the covariate variable of dimensionality n, δ_i is a binary value indicating whether the event happened or not, and $R(t_i)$ denotes the set of all individuals at risk at time t_i, which represents the set of cases that are still at risk before time t_i. $\theta^T x_i$ is called the risk (or survival) function, in which θ can be estimated by minimizing its corresponding negative log partial likelihood function; n denotes the number of patients.

The autoencoder loss uses the squared L2 distance between the original and reconstructed input for penalizing the autoencoder's reconstruction error. We denote this loss as:

$$L_{AE} = \frac{1}{n}\sum_{i=1}^{n} \|(g \circ f)(x_i) - x_i\|_2^2 \tag{2}$$

where $(g \circ f)(x_i)$ is the decoder network reconstructed input.

The two required distances loss, which are two interpretability regularization terms, are formulated as follows:

$$L_{DIS1} = \frac{1}{m}\sum_{j=1}^{m} \min_{i \in [1,n]} \left\|p_j - f(x_i)\right\|_2^2 \tag{3}$$

$$L_{DIS2} = \frac{1}{n}\sum_{i=1}^{n} \min_{j \in [1,m]} \left\|f(x_i) - p_j\right\|_2^2 \tag{4}$$

where $f(x_i)$ is the encoded input vector, p_i is the vector learned from the prototype layer. The prototype layer p computes the squared L_2 distance between the encoded input $f(x_i)$ and each of the prototype layer vectors. Minimization of L_{DIS1} will make each prototype vector as close as possible to at least one training case. The minimization of L_{DIS2} will make each encoded training example as close as possible to some prototype vector. It is worth noting that the purpose of minimizing the distances L_{DIS2} is to find the most similar case to the test case in the prototype outputs of the training set. This implements the nearest neighbor strategy method from case-based reasoning in the deep network. We use the two terms L_{DIS1} and L_{DIS2} in our cost function to illustrate the interpretability. The network chooses prototypes that fully represent the input space, and some of the prototypes tend to be similar to each other. Intuitively, L_{DIS1} approximates each prototype to a potential training example, making the decoded prototype realistic, while L_{DIS2} forces each training example to find a close prototype in the latent space,

thus encouraging the prototypes to spread out over the entire latent space and to be distinct from each other. In the latent space, they are different from each other.

By combining the above 4 loss terms, the objective loss function can be formulated as follows:

$$L_{total} = \lambda_1 L_Z + \lambda_2 L_{AE} + \lambda_3 L_{DIS1} + \lambda_4 L_{DIS2} \tag{5}$$

where λ_i $(i = 1,2,3,4)$ is the regularization weight for the regularization terms respectively. Instead of fixing the weights, we use these weights as trainable parameters in the deep network for adaptive optimization [3].

This network architecture, unlike traditional case-based learning methods, automatically learns useful features. For methylated feature data, those methods (e.g., *k*-nearest neighbors) tend to perform poorly if we use the raw input space or use a hand-crafted feature space for predictions. We feed the sequence vectors into the decoder and train them throughout learned variables during the process. This approach will enable the system to explain how the network reaches its predictions and will show the learning process of the input variables without post-hoc analysis.

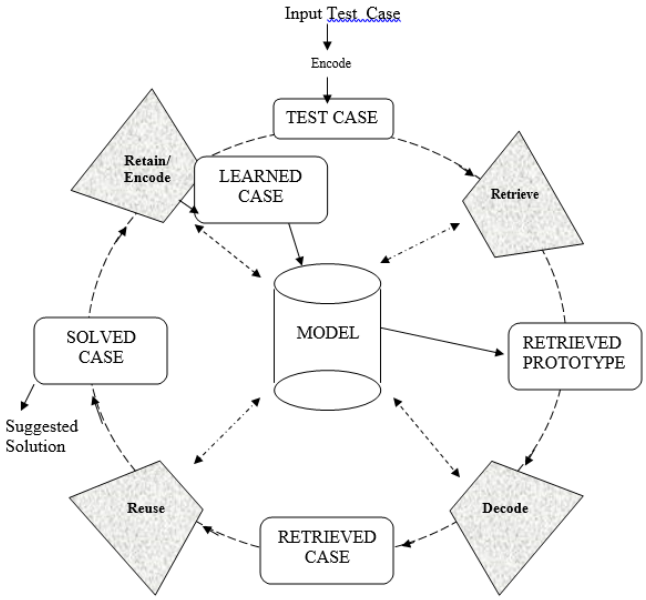


Fig. 2. Case-based reasoning for survival analysis

4.3 Survival Prediction

The survival prediction of the system can then be provided for any test case, associated with one input case corresponding to the nearest prototype activated by the biLSTM layer (see Fig. 2). Since the output of the prototype layer is generated during the iterative optimization of the model, it represents the characteristics of the training set. Minimizing the distance between the encoder output and prototype output means finding

the nearest neighbor to this test case in the training set. If the minimum prototype distance is found, it means that one prototype case can represent the encoder output of this test case. This method exactly utilizes the strategy of the nearest distance retrieval in case-based reasoning. . For each training sample, the linear expression of each feature was calculated and used to construct one prototypical case. Each prototype case would then represent typical DNA methylation patterns present in a sample. The Euclidean distance between each encoder from the case base and its respective prototype is used to determine how similar the prototype is to its own. This will appropriately determine how well the case fits a collection of unsolved cases during the prediction. The ability to trace the nearest neighbor, according to the deep similarity calculated by the model, adds to the transparency and explainability of the system.

5 Results and Discussions

5.1 Benchmark Datasets

In this section, we assess the performance of the proposed method and carry out experiments on two cancer DNA methylation datasets through ten-fold cross validation. We selected Glioma cohort (GBMLGG) cancer and Pan-kidney cohort (KIPAN) cancer, two datasets from Firehose [4]. The GBMLGG datasets include 1129 samples for clinical data and 20116 gene-level features for DNA methylation data. The KIPAN datasets contain 973 samples and 20533 DNA methylation features. In our case, we will also use the clinical data. Two clinical variables are used: survival status and survival time. In survival status, 'Deceased' represents the patient deceased, 'Living' means that he/she is living at the time of the last follow-up. The survival time represents the number of days between diagnosis and date of death or last follow-up. This study removes cases with survival days that were not recorded or negative. For these reasons, this study extracts 650 samples for GBMLGG data and 654 samples for KIPAN data that have both DNA methylation data and clinical data respectively after merging and filtering.

Table 1. Gene and clinical characteristics in two cancers.

Characteristics	GBMLGG	KIPAN
Patient no.	650	654
Gene no.		
DNA Methylation	20116	20533
Selected features	586	749
Survival status		
Living	434	500
Deceased	216	154
Follow up (days)	1-481	3-5925

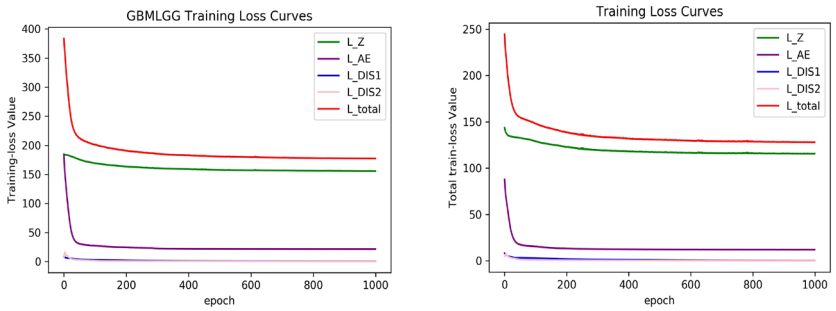
The high-dimension and low-sample size methylation data posed a challenge for obtaining sufficient statistical power. To accurately describe local features and all the levels (high & low) in feature representation of cancer samples, we use a multivariate Cox regression preprocess to extract the biomarkers. We calculate the log rank of each gene

feature and select the gene features whose p-value are less than 0.01. Thus, we can get the preliminarily reduced features. For GBMLGG data, by using this method, we extract 586 methylation features. Similarly, we can obtain 749 methylation features for KIPAN data. Table 1 shows the Gene and clinical characteristics for the selected cases.

As is classical in survival analysis, we use Concordance index (C-index) [21] to assess the performance of the developed approach and other comparable methods. C-index is the probability that the predicted survival time of a random pair of individuals is in the same order as their actual survival time. It is very useful for evaluating proportional hazard models.

5.2 Convergence Analysis

To investigate the convergence of the proposed method, we calculate the corresponding loss curves of Eq. 5 on two datasets. Fig. 3 shows the training loss curves of the five different loss functions we used concerning the GBMLGG and KIPAN datasets respectively. As shown in Fig. 3, the values of the training objective function loss decrease with respect to iterations on both datasets. The four loss terms (L_Z, L_{AE}, L_{DIS1}, and L_{DIS2}) and the total loss value (L_{total}) combined from them all converge to some stable values after a few iterations. Therefore, our proposed optimization algorithm is reliable and convergent.



(a) (b)

Fig. 3. The training loss curves using the proposed methods on two datasets. (a) the curves of training losses on GBMLGG dataset; (b) the curves of training losses on KIPAN dataset.

Let us investigate the autoencoder loss and the two interpretability prototype distance terms for a test case. We randomly select a test case from the GBMLGG and KIPAN datasets respectively for the experiments. Fig. 3 shows the curves of three distance terms (L_{AE}, L_{DIS1}, and L_{DIS2}) during the prediction iterations for one test case of each of the two datasets. In Fig. 3 (a), the values (L_{AE}, L_{DIS1}, and L_{DIS2}) are changed to (0.01671, 2.82664, and 0.02032) when 1000 epochs are completed. As also can be seen from Fig. 3 (b), the three values will converge to (0.005529, 2.0152, and 0.07098).

Obviously, by searching for the smallest distance L_{DIS2}, for GBMLGG, we can find the most similar case No. 356 in the training set to the test case No. 62. This means that we can use the characteristics of the known cases in the case base to explain the unsolved cases. Similarly, for KIPAN, we can find the most similar case No. 266 in the

training set to match the test case No. 319.

From Fig. 4, we can find that the curve of L_{AE} (purple) and the curve of L_{DIS2} (pink) both converge to almost the same value when the model converges after 1000 epochs. The results of the two different datasets are consistent. The autoencoder loss L_{AE} is the distance between the original and reconstructed input. We use the autoencoder to create a latent low-dimensional space. The smaller the distance between the decoder and the original input, the more the encoder output can represent the original input. The interpretability prototype distance L_{DIS2} means the minimum distance between the encoder output and prototype output. When the two distances (L_{AE} and L_{DIS2}) tend to be the same, the prototype features will explain the original input data.

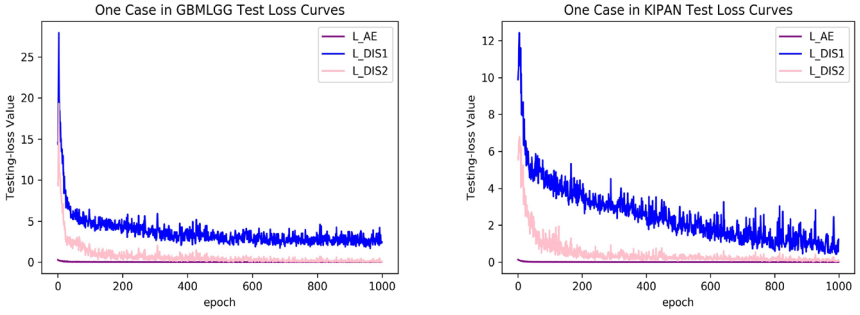


Fig. 4. Three curves of distance terms on two test cases from two datasets. (a) the curves of distance terms on one test case in GBMLGG dataset; (b) the curves of distance terms on one test case in KIPAN dataset

As also can be seen from Fig. 4, for the values of L_{DIS1} and L_{DIS2}, although their equations (Eq. 3 and Eq. 4) look similar, they are actually different. Actually, L_{DIS1} helps make the prototypes meaningful, and L_{DIS2} keeps the explanations faithful in forcing the network to use nearby prototypes.

Table 2. Performance comparison between two models by C-index (higher is better) on two datasets (with standard deviations)

Models	**GBMLGG**	**KIPAN**
Prototype	0.7132 (0.0166)	0.7246 (0.0149)
Without prototype	**0.7157 (0.0234)**	**0.7313 (0.0188)**

5.3 Survival prediction Performance and Interpretability

We compared our model to a network without the explainable parts, in which we removed the autoencoder layers and the prototype layer. We replaced the prototype output vectors with original input vectors (without prototype) as the input for the last output layer directly. Table 2 shows the performance comparison between these two mod-

els by the measurements of C-index on GBMLGG and KIPAN datasets. As demonstrated in Table 2, compared with no prototype model performance, the C-index of the prototype model is only 0.25% and 0.67% lower on GBMLGG and KIPAN datasets respectively. From Table 2, the result illustrates that we do not sacrifice much C-index when including the interpretability elements into the network.

5.4 Comparison with Different Survival Prediction Methods

To explore the effectiveness of the proposed method, we compare the developed method with three existing machine learning survival prediction approaches: LASSO, RSF, and DeepSurv. For the sake of fairness, this part of the study runs the same input feature set in all cross-validation tests. Table 4 presents the performance comparison between the proposed method and the three stated methods by the measurements of the C-index on GBMLGG and KIPAN datasets.

As shown in Table 3, it can be found that our proposed method outperforms all the other three methods. Compared with the approaches: LASSO, RSF, and DeepSurv, the C-index of the proposed method is improved by 10.97 percent, 12.96 percent, and 5.85 percent on GBMLGG data; and by 11%, 11.13%, and 3.88% on KIPAN data, respectively. As also can be seen from Table 3, the prognosis power of the deep cox model (i.e., DeepSurv) is superior to the other traditional regularized Cox model methods (i.e., LASSO and RSF). It is worth mentioning that DeepSurv method uses a linear network, but our method outperforms DeepSurv. So, it demonstrates the advantage in survival prediction and the efficacy of the proposed method.

Table 3. Performance comparison among a range of survival prediction approaches by C-index (higher is better) on two datasets (with standard deviations)

Methods	GBMLGG	KIPAN
LASSO	0.6035 (0.0141)	0.6146 (0.0246)
RSF	0.5836 (0.0238)	0.6133 (0.0233)
DeepSurv	0.6547 (0.0216)	0.6858 (0.0173)
Proposed Method	**0.7132 (0.0166)**	**0.7246 (0.0149)**

6 Discussion

In this study, we developed a synergistic machine learning combining CBR, an autoencoder, and a LSTM prediction model for survival analysis. In comparison with state-of-the-art survival analysis models, the proposed model performs better in predicting survival, while providing transparency and explainability through a prototype layer where each prototype can be traced back to a training case. This approach provides same transparency as case-based reasoning by tracing which training inputs have influenced the model behavior.

In a previous study, Bartlett et al. [2] used solely case-based reasoning, without the synergy with deep learning used in this paper for autoencoding and similarity assessment. The results of this study are not comparable with the current study because the

datasets were not the same: breast cancer in [2] and glioma and pan-kidney in the current study.

We plan in the future to compare the two systems on the same three datasets, both on the entire feature set and with same feature selection methods.

7 Conclusions

In this study, we developed a synergistic system between CBR, autoencoding, and LSTM for survival analysis in oncology. This system provides an explainable survival analysis framework of cancer patients, which uses an autoencoder network to reconstruct features of the training input and uses a prototype layer to store the weight vector following the encoded input. This deep survival prediction architecture can explain its own reasoning process and can provide explanations for each prediction, based on retrieved cases. We performed ten-fold cross-validation experiments on the DNA methylation data from two cancer types (GBMLGG and KIPAN). We have compared the performance of the proposed method with that of three other state-of-the-art existing methods (i.e., LASSO, RSF, and DeepSurv) through the performance measurement of C-index. The test results demonstrate that the survival prediction ability of the proposed method is better than that of the other three reported methods. We also investigate the convergence of the proposed method. The prototype layer can provide useful insight into the inner workings of the network. This method can partially trace the path of survival time prediction for a new observation to a previous case. This approach can partially trace the path of changes of the original input data in the deep network for survival prediction. Future plans include a comparison with CBR for survival analysis without LSTM, an evaluation of the interpretability of this model, and the addition of adaptation to the system's capability. The current approach has broader applications in the entire field of survival analysis as well as time series and sequence prediction in any domain.

References

1. Amiri, Z., Mohammad, K., Mahmoudi, M., Zeraati, H., and Fotouhi, A. Assessment of gastric cancer survival: using an artificial hierarchical neural network. Pak J Biol Sci 11, pp. 1076-1084 (2008).
2. Bichindaritz I, Bartlett C, Liu G. Predicting with Confidence: A Case-Based Reasoning Framework for Predicting Survival in Breast Cancer. In: The International FLAIRS Conference Proceedings, Apr 18 (Vol. 34) (2021)
3. Bichindaritz, I., Liu, G., and Bartlett, C. Survival Prediction of Breast Cancer Patient from Gene Methylation Data with Deep LSTM Network and Ordinal Cox Model. The Thirty-Third International Flairs Conference, Vol., pp. (2020).
4. Bichindaritz, I., Liu, G., and Bartlett, C. Integrative survival analysis of breast cancer with gene expression and DNA methylation data. Bioinformatics 37, pp. 2601-2608 (2021).
5. Caruana, R., Kangarloo, H., Dionisio, J.D., Sinha, U. and Johnson, D. Case-based explanationof non-case-based learning methods. In: Proceedings of the AMIA Symposium, p. 212.American Medical Informatics Association (1999).

6. Deng, M., Brägelmann, J., Kryukov, I., Saraiva-Agostinho, N., and Perner, S. FirebrowseR: an R client to the Broad Institute's Firehose Pipeline. Database 2017, (2017).
7. Farzindar, A. A., and Kashi, A. Multi-Task Survival Analysis of Liver Transplantation Using Deep Learning. The Thirty-Second International Flairs Conference, Vol., pp. (2019).
8. Eisenstadt, V., Langenhan, C., Althoff, K.D., Dengel, A. Improved and Visually Enhanced Case-Based Retrieval of Room Configurations for Assistance in Architectural Design Education. International Conference on Case-Based Reasoning. Springer, Cham (2020)
9. Hochreiter, S., and Schmidhuber, J.: Long short-term memory. Neural computation 9, pp. 1735-1780 (1997).
10. Ishwaran, H., Kogalur, U. B., Blackstone, E. H., and Lauer, M. S. Random survival forests. The annals of applied statistics 2, pp. 841-860 (2008).
11. Joshi, S., Koyejo, O., Vijitbenjaronk, W., Kim, B., & Ghosh, J. Towards realistic individual recourse and actionable explanations in black-box decision making systems. arXiv preprint arXiv:1907.09615. (2019).
12. Karmen, C., Gietzelt, M., Knaup-Gregori, P., and Ganzinger, M. Methods for a similarity measure for clinical attributes based on survival data analysis. BMC medical informatics and decision making 19, pp. 1-14 (2019).
13. Katzman, J. L., Shaham, U., Cloninger, A., Bates, J., Jiang, T., and Kluger, Y. Deep survival: A deep cox proportional hazards network. stat 1050, 2 (2016).
14. Katzman, J. L., Shaham, U., Cloninger, A., Bates, J., Jiang, T., and Kluger, Y. DeepSurv: personalized treatment recommender system using a Cox proportional hazards deep neural network. BMC medical research methodology 18, 24 (2018).
15. Kim B, Khanna R, Koyejo OO. Examples are not enough, learn to criticize! criticism for interpretability.In: Advances inNeural InformationProcessing Systems; 2016. p. 2280–2288
16. Kim, B., Rudin, C., & Shah, J. A. The bayesian case model: A generative approach for case-based reasoning andprototype classification. Advances in neural information processing systems, 27 (2014).
17. Li O, Liu H, Chen C, Rudin C. Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions. In Thirty-Second AAAI Conference on Artificial Intelligence **(**2018**).**
18. Lin, D. Y., Wei, L.-J., and Ying, Z. Checking the Cox model with cumulative sums of martingale-based residuals. Biometrika 80, pp. 557-572 (1993).
19. Lundberg, S. M., and Lee, S.-I. A unified approach to interpreting model predictions. Advances in neural information processing systems 30, (2017).
20. Lundberg, S. M., Nair, B., Vavilala, M. S., Horibe, M., Eisses, M. J., Adams, T. et al. Explainable machine-learning predictions for the prevention of hypoxaemia during surgery. Nature biomedical engineering 2, pp. 749-760 (2018).
21. Mayr, A., and Schmid, M. Boosting the concordance index for survival data. Ulmer Informatik-Berichte, 26 (2014).
22. Ramos, B., Pereira, T., Moranguinho, J., Morgado, J., Costa, J. L., and Oliveira, H. P. An Interpretable Approach for Lung Cancer Prediction and Subtype Classification using Gene Expression. 2021 43rd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Vol., pp. 1707-1710. IEEE (2021).
23. Ramon, Y., Martens, D., Provost, F., & Evgeniou, T. A comparison of instance-level counterfactual explanationalgorithms for behavioral and textual data: SEDC, LIME-C and SHAP-C. Advances in Data Analysis andClassification, 14(4), pp. 801-819 (2020).
24. Ryall, S., Tabori, U. and Hawkins, C. A comprehensive review of paediatric low-grade diffuse glioma: pathology, molecular genetics and treatment, Brain tumor pathology, 34, pp. 51-61 (2017).

25. Shao, W., Cheng, J., Sun, L., Han, Z., Feng, Q., Zhang, D. et al. Ordinal multi-modal feature selection for survival analysis of early-stage renal cancer. International Conference on Medical Image Computing and Computer-Assisted Intervention, Vol., pp. 648-656. Springer (2018).
26. Sy, J.P., and J.M.G. Taylor. Estimation in a Cox proportional hazards cure model.Biometrics 56.1 pp. 227-236 (2020).
27. Suzuki, H., Maruyama, R., Yamamoto, E., and Kai, M. DNA methylation and microRNA dysregulation in cancer. Molecular oncology 6, pp. 567-578 (2012).
28. Tibshirani, R. The lasso method for variable selection in the Cox model, Statistics in medicine, 16, pp. 385-395 (1997).
29. Xiang, A., et al. Comparison of the performance of neural network methods and Cox regression for censored survival data, Computational statistics & data analysis, 34, pp. 243-257 (2000).
30. Yu, K.-H., Zhang, C., Berry, G. J., Altman, R. B., Ré, C., Rubin, D. L. et al.: Predicting non-small cell lung cancer prognosis by fully automated microscopic pathology image features. Nature communications 7, 12474 (2016).