BERTSurv: BERT-Based Survival Models for Predicting Outcomes of Trauma Patients

Yun Zhao[1], Qinghang Hong[1], Xinlu Zhang[1], Yu Deng[2], Yuqing Wang[1], and Linda Petzold[1]

[1] Department of Computer Science, University of California, Santa Barbara
[2] Feinberg School of Medicine, Northwestern University
yunzhao@cs.ucsb.edu

Abstract. Survival analysis is a technique to predict the times of specific outcomes, and is widely used in predicting the outcomes for intensive care unit (ICU) trauma patients. Recently, deep learning models have drawn increasing attention in healthcare. However, there is a lack of deep learning methods that can model the relationship between measurements, clinical notes and mortality outcomes. In this paper we introduce BERTSurv, a deep learning survival framework which applies Bidirectional Encoder Representations from Transformers (BERT) as a language representation model on unstructured clinical notes, for mortality prediction and survival analysis. We also incorporate clinical measurements in BERTSurv. With binary cross-entropy (BCE) loss, BERTSurv can predict mortality as a binary outcome (mortality prediction). With partial log-likelihood (PLL) loss, BERTSurv predicts the probability of mortality as a time-to-event outcome (survival analysis). We apply BERTSurv on Medical Information Mart for Intensive Care III (MIMIC III) trauma patient data. For mortality prediction, BERTSurv obtained an area under the curve of receiver operating characteristic curve (AUC-ROC) of 0.86, which is an improvement of 3.6% over baseline of multilayer perceptron (MLP) without notes. For survival analysis, BERTSurv achieved a concordance index (C-index) of 0.7. In addition, visualizations of BERT's attention heads help to extract patterns in clinical notes and improve model interpretability by showing how the model assigns weights to different inputs.

Keywords: Deep learning · BERT · Survival analysis · Mortality prediction.

1 Introduction

Trauma is the leading cause of death from age 1 to 44. More than 180,000 deaths from trauma occur each year in the United States [1]. Most trauma patients die or are discharged quickly after being admitted to the ICU. Care in the first few

hours after admission is critical to patient outcome, yet this time period is more prone to medical decision errors in ICUs [2] than later periods. Therefore, early and accurate prediction for trauma patient outcomes is essential for ICU decision making.

Medical practitioners use survival models to predict the outcomes for trauma patients [3]. Survival analysis is a technique to model the distribution of the outcome time. The Cox model [4] is one of the most widely used survival models with linear proportional hazards. Faraggi-Simon's network [5] is an extension of the Cox model to nonlinear proportional hazards using a neural network. DeepSurv [6] models interactions between a patient's covariates and treatment effectiveness with a Cox proportional hazards deep neural network. However, these existing models deal only with well-structured measurements and do not incorporate information from unstructured clinical notes, which can offer significant insight into patients' conditions.

The transformer architecture has taken over sequence transduction tasks in natural language processing (NLP). Transformer is a sequence model that adopts a fully attention-based approach instead of traditional recurrent architectures. Based on Transformer, BERT [8] was proposed for language representation and achieved state-of-the-art performance on many NLP tasks. There has also been increasing interest in applying deep learning to end-to-end e-health data analysis [9]. Biobert [10] extends BERT to model biomedical language representation. Med-BERT [11] modifies BERT by leveraging domain specific hierarchical code embedding and layer representation to generate sequential relationships in the clinical domain. G-BERT [12] combines Graph Neural Networks (GNNs) and BERT for medical code representation and medication recommendation. Clinical BERT [13, 14] explores and pre-trains BERT using clinical notes. Clearly, there is an unmet need to include unstructured text information in deep learning survival models for patient outcome predictions.

In this paper we propose BERTSurv, a deep learning survival framework for trauma patients which incorporates clinical notes and measurements for outcome prediction. BERTSurv allows for both mortality prediction and survival analysis by using BCE and PLL loss, respectively. Our experimental results indicate that BERTSurv can achieve an AUC-ROC of 0.86, which is an improvement of 3.6% over the baseline of MLP without notes on mortality prediction.

The key contributions of this paper are:

1. We propose BERTSurv: a BERT-based deep learning framework to predict the risk of death for trauma patients. To the best of our knowledge, this is the first paper applying BERT on unstructured text data combined with measurements for survival analysis.

2. We evaluate BERTSurv on the trauma patients in MIMIC III. For mortality prediction, BERTSurv achieves an AUC-ROC of 0.86, which outperforms baseline of MLP without notes by 3.6%. For survival analysis, BERTSurv achieved a C-index of 0.7 on trauma patients, which outperforms a Cox model with a C-index of 0.68.

3. We extract patterns in clinical notes by performing attention mechanism visualization, which improves model interpretability by showing how the model assigns weights to different clinical input texts with respect to survival outcomes.

This paper is organized as follows: Section 2 describes how we processed the MIMIC trauma dataset. We present BERTSurv in Section 3.1 and describe the background of BERT and survival analysis in Section 3.2 and Section 3.3. Evaluation and discussion are given in Sections 4 and 5, respectively.

2   Dataset

BERTSurv is applied to the data from trauma patients selected using the ICD-9 code from the publicly available MIMIC III dataset [16], which provides extensive electronic medical records for ICU admissions at the Beth Israel Deaconess Medical Center between 2001 and 2012. The measurements, clinical notes, expire flag (0 for discharge and 1 for death), and death/discharge time for each patient were used to train and test BERTSurv. The patient data were aggregated over the first 4 hours to obtain the initial state of each individual admission. We took the average for each of the measurements taken during this time period, and concatenated all of the clinical notes together. Considering the missing value issue and redundancy in MIMIC III, we selected 21 common features as our representative set: blood pressure, temperature, respiratory rate, arterial PaO2, hematocrit, WBC, creatinine, chloride, lactic acid, BUN, sodium (Na), glucose, PaCO2, pH, GCS, heart rate, FiO2, potassium, calcium, PTT and INR. Our feature set overlaps 65% of the measurements required by APACHE III [15]. We also extracted 4 demographic predictors: weight, gender, ethnicity and age.

As is common in medical data, MIMIC III contains many missing values in the measurements, and the notes are not well-formatted. Thus, data preprocessing is very important to predict outcomes. To deal with the missing data issue, we first removed patients who have a missing value proportion greater than 0.4 and then applied MICE [17] data imputation for the remainder of the missing values. For the clinical notes, we removed formatting, punctuation, non-punctuation symbols and stop words. In addition to the most commonly used English stop words, our stop word dictionary includes a few specific clinical and trauma related stop words: *doctor, nurse and measurement*, etc. Following this preprocessing, our trauma dataset includes 1860 ICU patients, with 21 endogenous measurements, 4 exogenous measurements and notes. The sample class ratio between class 0 (discharge) and class 1 (death) is 1206 : 654.

3   Methods

In this section we first describe the framework of BERTSurv. Then we introduce some basics of BERT and survival analysis, which are the key components of BERTSurv.

3.1 BERTSurv

Our model architecture, shown in Fig 1, consists of BERT embedding of clinical notes concatenated with measurements followed by feed forward layers. The output for BERTSurv is a single node $h_\theta(x_i)$ parameterized by the weights of the neural network θ, which estimates either the probability of mortality or the hazard risk. For mortality prediction, we apply BCE loss to predict outcomes of death or discharge:

$$\text{BCELoss} := \sum_{i=1}^{n} p(y_i) \log(h_\theta(x_i)), \tag{1}$$

where x_i and y_i represent inputs and outcomes for the ith patient, respectively.

To estimate θ in survival analysis, similar to the Faraggi-Simon network [5,6], we minimize the PLL loss function, which is the average negative log partial likelihood:

$$\text{PLLLoss} := -\frac{1}{N_{D=1}} \sum_{i:D_i=1} \left(h_\theta(x_i) - \log \sum_{j \in R(T_i)} \exp(h_\theta(x_j)) \right), \tag{2}$$

where $N_{D=1}$ is the number of patients with an observable death. The risk set $R_i = \{j : T_j \geq T_i\}$ is the set of those patients under risk at T_i.
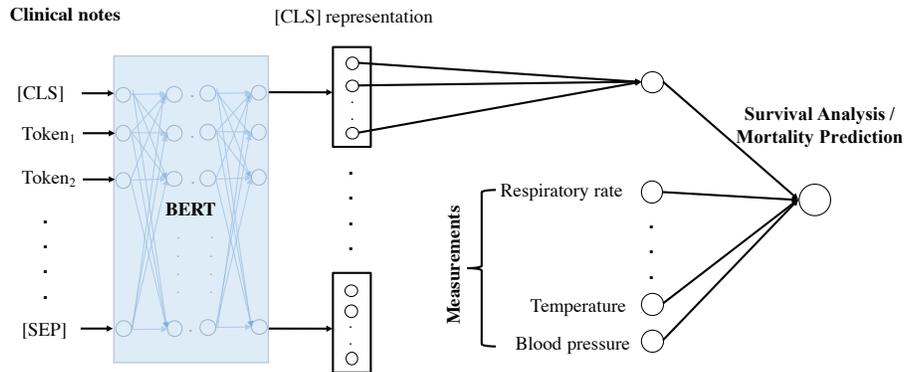

Fig. 1: The framework of BERTSurv. [CLS] is a special symbol added in front of every clinical note sample, and [SEP] stands for a special separator token. BERTSurv consists of three main parts: BERT, measurements and output layer for mortality prediction or survival analysis. First, we input a set of diagnostics and nurse notes to BERT pretrained on masked language modeling and next sentence prediction. The [CLS] representation, is treated as the representation of the input notes. Then we concatenate the [CLS] representation and measurements as input and fine-tune BERTSurv for downstream survival analysis.

We use batch normalization through normalization of the input layer by re-centering and re-scaling [18]. We apply rectified linear unit(ReLU) or scaled exponential linear units (SELU) as the activation function. For regularization, dropout [19] is implemented to avoid overfitting. Dropout prevents co-adaptation of hidden units by randomly dropping out a proportion of the hidden units during backpropagation. BCE/PLL loss is minimized with the Adam optimizer [20] for training.

BERTSurv is implemented in Pytorch [21]. We use a Dell 32GB NVIDIA Tesla M10 Graphics Card GPU (and significant CPU power and memory for pre-processing tasks) for training, validation and testing. The hyperparameters of the network include: BERT choice ($BERT_{BASE}$ or clinical BERT [13]), sequence length, batch size, learning rate, dropout rate, training epochs and activation function (ReLU or SELU).

3.2   BERT

A key component of BERTSurv is the BERT language representation model. BERT is a Transformer-based language representation model, which is designed to pre-train deep bidirectional representations from unlabeled text by jointly considering context from both directions (left and right). Using BERT, the input representation for each token in the clinical notes is the sum of the corresponding token embeddings, segmentation embeddings and position embeddings. WordPiece token embeddings [22] with a 30,000 token vocabulary are applied as input to BERT. The segment embeddings identify which sentence the token is associated with. The position embeddings of a token are a learned set of parameters corresponding to the token's position in the input sequence. An attention function maps a query and a set of key-value pairs to an output. The attention function takes a set of queries Q, keys K, and values V as inputs and is computed on an input sequence using the embeddings associated with the input tokens. To construct Q, K and V, every input embedding is multiplied by the learned sets of weights. The attention function is

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}}V), \qquad (3)$$

where d_k is the dimensionality of Q and K. The dimension of V is d_v. A multi-head attention mechanism allows BERT to jointly deal with information from different representation subspaces at different positions with several (h) attention layers running in parallel:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_h)W^O, \qquad (4)$$

where $\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$. Parameter matrices $W_i^Q \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$, $W_i^V \in \mathbb{R}^{d_{\text{model}} \times d_v}$ and $W^O \in \mathbb{R}^{hd_v \times d_{\text{model}}}$ are the learned linear projections from Q, K, V to d_k, d_k and d_v dimensions.

In BERTSurv, we use pretrained BERT of $BERT_{BASE}$ and clinical BERT [13] for clinical note embedding, and focus on fine-tuning for survival analysis.

3.3   Survival Analysis

Another key component of BERTSurv is survival analysis. Survival analysis [23, 24] is a statistical methodology for analyzing the expected duration until one or more events occur. The survival function $S(t)$, defined as $S(t) = P(T \geq t)$, gives the probability that the time to the event occurs later than a given time t. The cumulative distribution function (CDF) of the time to event gives the cumulative probability for a given t-value:

$$F(t) = P(T < t) = 1 - S(t). \tag{5}$$

The hazard function $h(t)$ models the probability that an event will occur in the time interval $[t, t + \Delta t)$ given that the event has not occurred before:

$$h(t) = \lim_{\Delta t \to 0} \frac{P(t \leq T < t + \Delta t \mid T \geq t)}{\Delta t} = \frac{f(t)}{S(t)}, \tag{6}$$

where $f(t)$ is the probability density function (PDF) of the time to event. A greater hazard implies a greater probability of event occurrence. Note from Equ. 5 that $-f(t)$ is the derivative of $S(t)$. Thus Equ. 6 can be rewritten as

$$h(t) = -\frac{dS(t)}{dt} * \frac{1}{S(t)} = -\frac{d}{dt} \log(S(t)). \tag{7}$$

By solving Equ. 7 and introducing the boundary condition $S(0) = 1$ (the event can not occur before duration 0), the relationship between $S(t)$ and $h(t)$ is given by

$$S(t) = \exp\left(-\int_0^t h(s)ds\right). \tag{8}$$

The Cox model [4] is a well-recognized survival model. It defines the hazard function given input data $h(t \mid \mathbf{y}, \boldsymbol{\eta})$ to be the product of a baseline function, which is a function of time, and a parametric function of the input data $\mathbf{y}$ and $\boldsymbol{\eta}$. $\mathbf{y}$ and $\boldsymbol{\eta}$ denote endogenous measurements and exogenous measurements, respectively. Using the assumption of a linear relationship between the log-risk function and the covariates, the Cox model has the form

$$h(t \mid \mathbf{y}, \boldsymbol{\eta}) = h_0(t) \exp(\boldsymbol{\tau}^T \mathbf{y} + \boldsymbol{\gamma}^T \boldsymbol{\eta}), \tag{9}$$

where $h_0(t)$ is the baseline hazard function, and $\boldsymbol{\tau}$ and $\boldsymbol{\gamma}$ are the vectors of weights for $\mathbf{y}$ and $\boldsymbol{\eta}$.

In BERTSurv, the log-risk function $h_\theta(\mathbf{x})$ is the output node from the neural network:

$$h(t \mid \mathbf{y}, \boldsymbol{\eta}) = h_0(t) \exp(h_\theta(\mathbf{x})), \tag{10}$$

where the input $\mathbf{x}$ includes $\mathbf{y}$, $\boldsymbol{\eta}$ and clinical notes. The likelihood function for the survival model is as follows:

$$p(T, \delta) = h(T)^\delta S(T). \tag{11}$$

When $\delta = 1$, it means that the event is observed at time T. When $\delta = 0$, the event has not occurred before T and it will be unknown after T. The time T when $\delta = 0$ is called the censoring time, which means the event is no longer observable.

The Cox partial likelihood is parameterized by $\boldsymbol{\tau}$ and $\boldsymbol{\gamma}$ and defined as

$$\mathrm{PL}(\boldsymbol{\tau}, \boldsymbol{\gamma}) = \prod_{i=1}^{n} \{ \frac{\exp(\boldsymbol{\tau}^T \mathbf{y} + \boldsymbol{\gamma}^T \boldsymbol{\eta})}{\sum_{j \in R_i} \exp(\boldsymbol{\tau}^T \mathbf{y} + \boldsymbol{\gamma}^T \boldsymbol{\eta})} \}^{\Delta_i}, \tag{12}$$

where $\Delta_i = I(T_i \leq C_i)$. C_i is the censoring time for the ith patient, and $I(*)$ is the indicator function.

We use the Breslow estimator [25] to estimate the cumulative baseline hazard $\widehat{H}_0(t) = \int_0^t \widehat{h_0}(u) du$:

$$\widehat{H}_0(t) = \sum_{i=1}^{n} \frac{I(T_i \leq t) \Delta_i}{\sum_{j \in R_i} \exp(\boldsymbol{\tau}^T \mathbf{y} + \boldsymbol{\gamma}^T \boldsymbol{\eta})}. \tag{13}$$

4   Experiments and Analysis

Throughout this section, we randomly pick 70% of the trauma data as training and the rest as testing. Considering the size of our dataset and the training time, we apply 5-fold cross-validation on the trauma training dataset and grid search for hyperparameter choice. Our hyperparameters are described in Table 1. Note that the sequence length and batch size choices are limited by GPU memory.

Table 1: Hyperparameters

Hyperparameters	Survival analysis	Mortality prediction
Batch size	24	16
Sequence length	512	512
Epoch	4	4
Dropout rate	0.1	0.1
Learning rate	1e-2	4e-2
BERT choice	clinical BERT	clinical BERT
Activation	SELU	ReLU

Using the clinical notes and measurements, we formulate the mortality prediction problem as a binary classification problem. Fig. 2 shows the averaged cross validation confusion matrix for mortality prediction in the trauma training dataset. The testing confusion matrix for mortality prediction is presented in Fig. 3. Dominant numbers on the diagonals of both confusion matrices indicate that BERTSurv achieves high accuracy for both of the outcomes (death/discharge).

With BCE loss, we apply two baselines: MLP without notes and the TF-IDF mortality model. In MLP without notes, we consider only the measurements and build a MLP with 3 feed-forward layers for mortality outcomes. In the TF-IDF mortality model, we represent notes with TF-IDF vectors and build a support vector machine (SVM) on TF-IDF vectors combined with measurements for mortality prediction. We use AUC-ROC as our performance metric for mortality prediction, as it is commonly used in survival prediction [26, 27]. AUC-ROC represents the probability that a classifier ranks the risk of a randomly chosen death patient (class 1) higher than a randomly chosen discharged patient (class 0). As is shown in Fig. 5, BERTSurv achieved an AUC-ROC of 0.86, which outperforms MLP without notes by 3.6%. BERTSurv also outperforms MLP without notes, with 5-fold cross validation as shown for our trauma training dataset in Fig. 4.
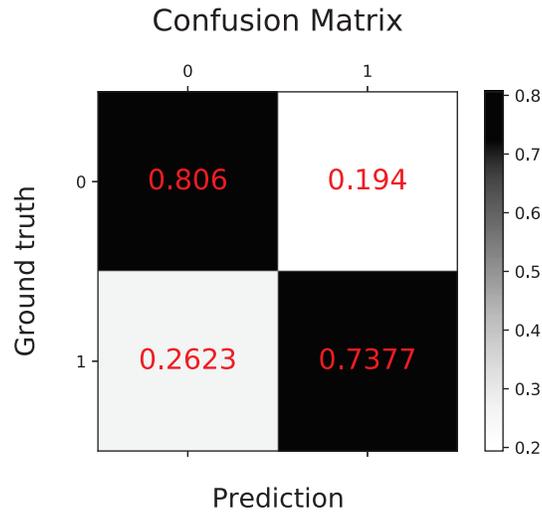

Fig. 2: Averaged confusion matrix for mortality prediction over 5-fold cross validation on our trauma training dataset.

To evaluate the model's predictive performance with PLL loss on survival analysis, we measure the concordance-index (C-index) as outlined by [28]. BERTSurv achieved a C-index of 0.7 on trauma patients, which outperforms a Cox model with a C-index of 0.68. To reduce the risk of ICU trauma patients progressing to an irreversible stage, accurate and early prediction of patient condition is crucial for timely medical decisions. Mortality and cumulative hazard curves for two patients with different outcomes from BERTSurv are shown in Fig. 6 and Fig. 7. Fig. 6(c) indicates that an earlier discharged patients have a lower risk than later discharged patients, while Fig. 6(b) shows that patients who die
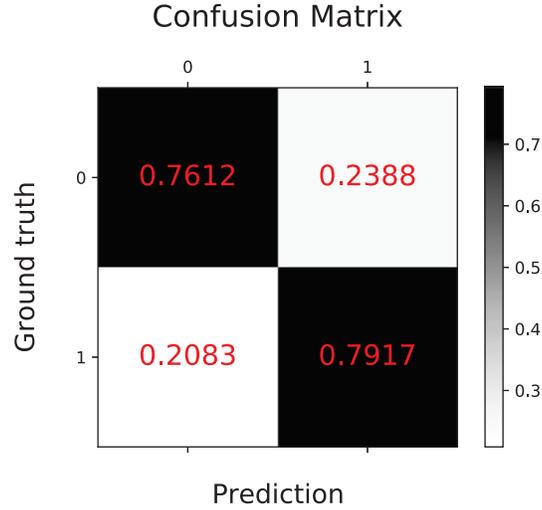
Confusion Matrix

Fig. 3: Confusion matrix for mortality prediction on trauma testing dataset.

early are at a relatively higher risk compared with patients who die later. Comparing Fig. 6(a) and Fig. 6(d), the gap between early discharge vs. early death is larger than that of late discharge vs. late death. Similar conclusions can be drawn from the hazard curves in Fig. 7. Such survival and hazard curves can provide physicians with comprehensive insight into patients' condition change with time.

Fig. 8 depicts four self-attention mechanisms in BERTSurv which help to understand patterns in the clinical notes. In all of the panels, the x-axis represents the query tokens and the y-axis represents the key tokens. In panels (a) and (b), we analyze a clinical note "left apical cap and left lateral pneumothorax suggests severe chest trauma " from a patient that died at hour 76. Panels (a) and (b) are two different head attention mechanisms. Panel (a) indicates "severe chest" and panel (b) extracts "trauma" as prominent patterns, respectively. Similarly, panels (c) and (d) are two head attention mechanisms for a patient discharged at hour 85. The input note to BERTSurv is "the endotracheal tube terminates in good position approximately 4 cm above the carina". BERTSurv finds "good" and "good position" in panels (c) and (d), respectively. Both "severe chest" and "good position" help in understanding the patients' conditions and strongly correlate with the final outcomes. The indications from extracted patterns to patient outcomes show the effectiveness of BERT representation for clinical notes.
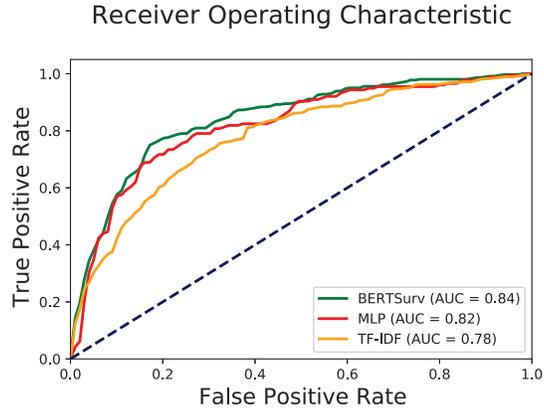
Receiver Operating Characteristic

Fig. 4: Receiver operating characteristic (ROC) curve for mortality prediction over 5-fold cross validation on our trauma training dataset. BERTSurv outperforms both baselines.
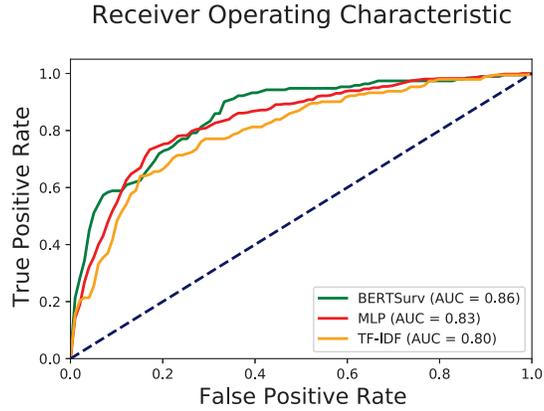
Receiver Operating Characteristic

Fig. 5: Receiver operating characteristic (ROC) curve for mortality prediction in trauma testing dataset. BERTSurv outperforms both baselines.
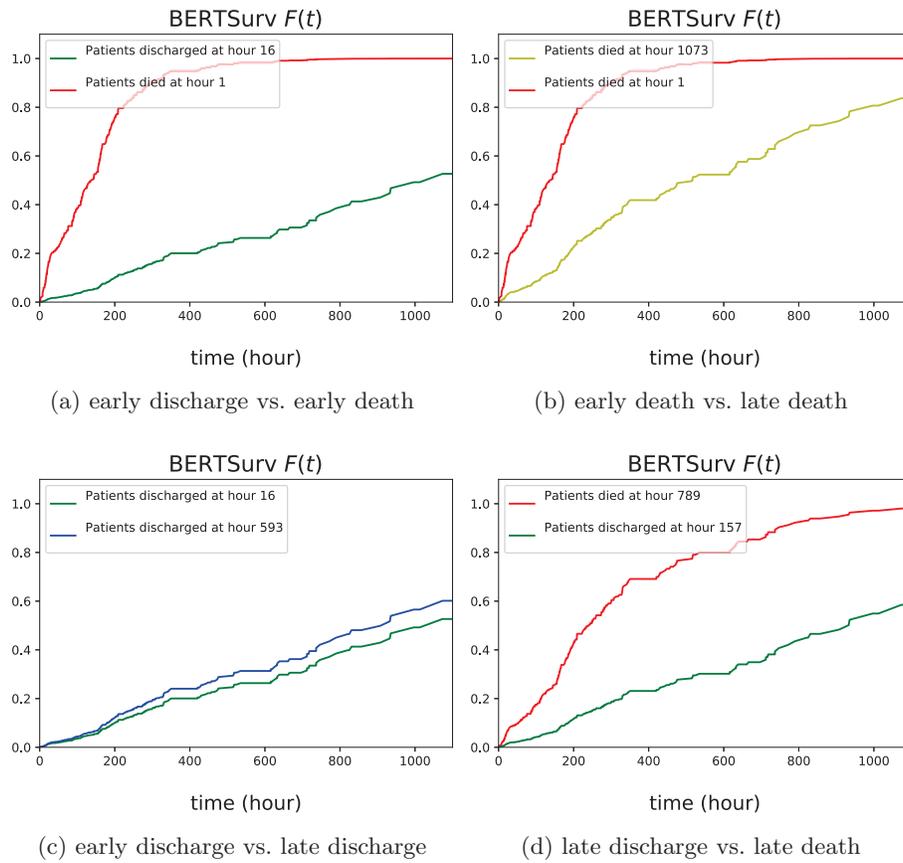
(a) early discharge vs. early death

(b) early death vs. late death

(c) early discharge vs. late discharge

(d) late discharge vs. late death

Fig. 6: Prediction of mortality as a function of time after admission to ICU using BERTSurv.

(a) early discharge vs. early death

(b) early death vs. late death

(c) early discharge vs. late discharge

(d) late discharge vs. late death
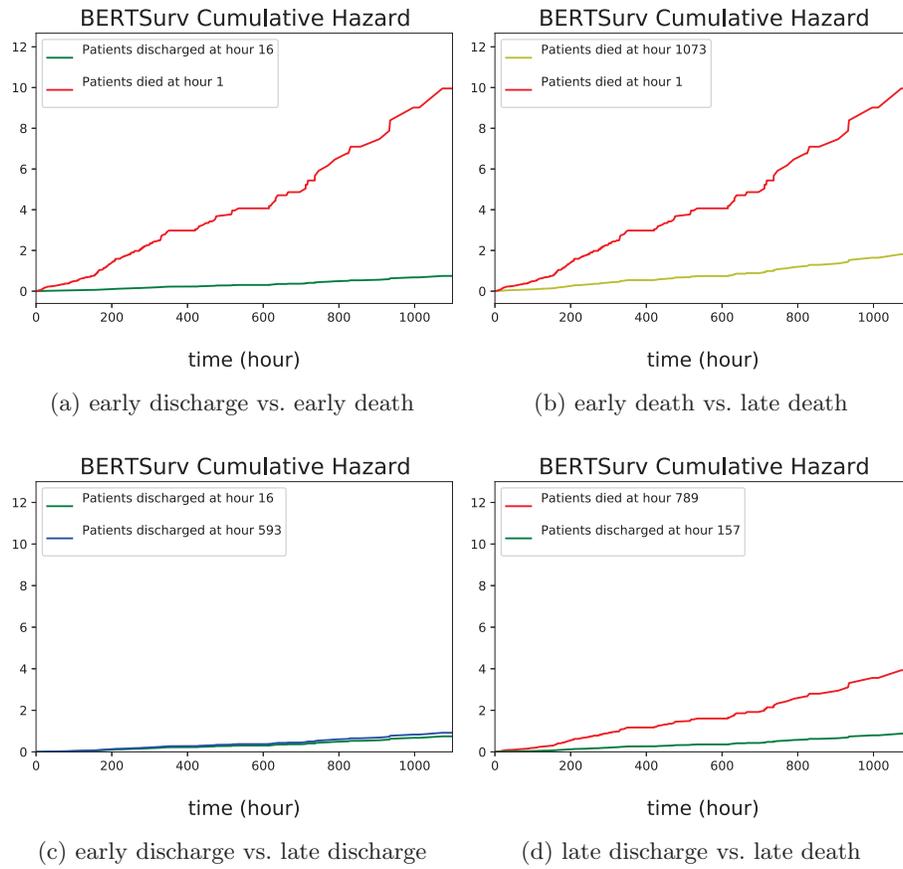
Fig. 7: Prediction of cumulative hazard function as a function of time after admission to ICU using BERTSurv.

(a) patient died at hour 76

(b) patient died at hour 76

(c) patient discharged at hour 85

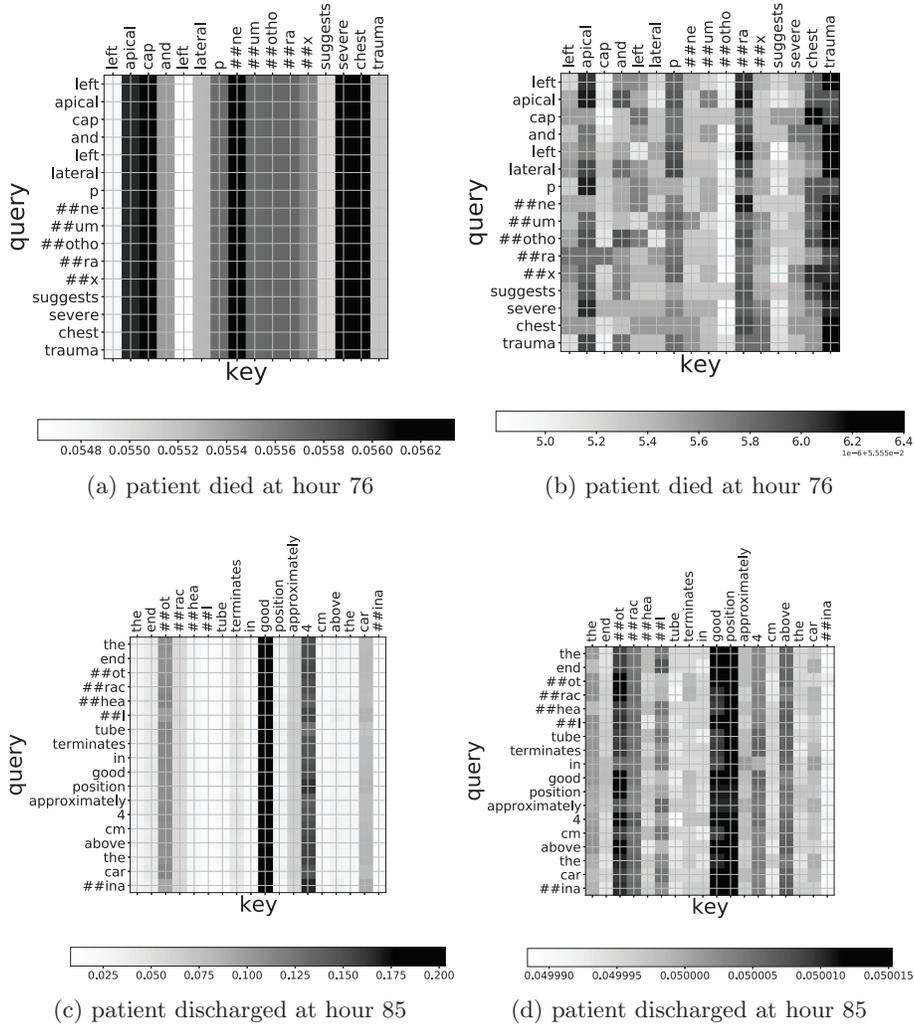(d) patient discharged at hour 85

Fig. 8: BERT visualization. The x-axis are the query tokens and the y-axis are the key tokens. Panels (a) and (b) are two head attention mechanisms for a patient that died at hour 76. The input notes to BERTSurv read "left apical cap and left lateral pneumothorax suggests severe chest trauma". Panels (a) and (b) extract "severe chest" and "trauma" as prominent patterns from the two heads, respectively. "severe chest" and "trauma" provide insight on the patient's critically ill condition. Similarly, panels (b) and (c) are two head attention mechanisms for a patient discharged at hour 85. The input notes include "the endotracheal tube terminates in good position approximately 4 cm above the carina". "good" stands out in panel (c) and "good position" emerges in panel (d). Both "good" and "good position" are strong indications that the patient is in a relatively benign condition.

5   Discussion

We have proposed a deep learning framework based on BERT for survival analysis to include unstructured clinical notes and measurements. Our results, based on MIMIC III trauma patient data, indicate that BERTSurv outperforms the Cox model and two other baselines. We also extracted patterns in the clinical texts with attention mechanism visualization and correlated the assigned weights with survival outcomes. This paper is a proof of principle for the incorporation of clinical notes into survival analysis with deep learning models. Given the current human and financial resources allocated in preliminary clinical note analysis, our method has foreseeable potential to save labor costs, and further improve trauma care. Additional data and work are needed, however, before the extent to which survival analysis can benefit from deep learning and NLP methods can be determined.

6   Acknowledgments

References

1. WISQARS Data Visualization, https://wisqars-viz.cdc.gov:8006/lcd/home.
2. Cullen, D., Sweitzer, B., Bates, D., Burdick, E., Edmondson, A., Leape, L.: Preventable adverse drug events in hospitalized patients. Critical Care Medicine. 25, 1289-1297 (1997).
3. Zhang, Y., Jiang, R., Petzold, L.: Survival topic models for predicting outcomes for trauma patients. In 2017 IEEE 33rd International Conference on Data Engineering (ICDE), 1497-1504 (2017).
4. Cox, D.: Regression models and life-tables. Journal of the Royal Statistical Society: Series B (Methodological). 34(2), 187-202 (1972).
5. Faraggi, D., Simon, R.: A neural network model for survival data. Statistics in Medicine. 14, 73-82 (1995).
6. Katzman, J., Shaham, U., Cloninger, A., Bates, J., Jiang, T., Kluger, Y.: DeepSurv: personalized treatment recommender system using a Cox proportional hazards deep neural network. BMC Medical Research Methodology. 18 (2018).
7. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., Polosukhin, I.: Attention is all you need. In Advances in Neural Information Processing Systems, 5998-6008 (2017).
8. Devlin, J., Chang, M., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018).
9. Darabi, H., Tsinis, D., Zecchini, K., Whitcomb, W., Liss, A.: Forecasting mortality risk for patients admitted to intensive care units using machine learning. Procedia Computer Science, 140, 306-313 (2018).

10. Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C., Kang, J.: BioBERT: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics, 36(4), 1234-1240 (2020).
11. Rasmy, L., Xiang, Y., Xie, Z., Tao, C., Zhi, D.: Med-BERT: pre-trained contextualized embeddings on large-scale structured electronic health records for disease prediction. arXiv preprint arXiv:2005.12833 (2020).
12. Shang, J., Ma, T., Xiao, C., Sun, J.: Pre-training of graph augmented transformers for medication recommendation. arXiv preprint arXiv:1906.00346 (2019).
13. Alsentzer, E., Murphy, J., Boag, W., Weng, W., Jin, D., Naumann, T., McDermott, M.: Publicly available clinical BERT embeddings. arXiv preprint arXiv:1904.03323. (2019).
14. Huang, K., Altosaar, J., Ranganath, R.: Clinicalbert: Modeling clinical notes and predicting hospital readmission. arXiv preprint arXiv:1904.05342 (2019).
15. Wa, K., Wagner, D., Draper, E.: The APACHE III prognostic system. Risk prediction of hospital mortality for critically ill hospitalized adults. Chest, 100(6) 1619-1636 (1991).
16. Johnson, A., Pollard, T., Shen, L., et al.: MIMIC-III, a freely accessible critical care database. Scientific data, 3(1), 1-9 (2016).
17. Buuren, S., Groothuis-Oudshoorn, K.: mice: Multivariate imputation by chained equations in R. Journal of statistical software, 1-68. Chicago (2010).
18. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015).
19. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. The journal of machine learning research, 15(1), 1929-1958 (2014).
20. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014).
21. Paszke, A., Gross, S., Massa, F., et al.: Pytorch: An imperative style, high-performance deep learning library. In Advances in neural information processing systems, 8026-8037 (2019).
22. Wu, Y., Schuster, M., Chen, Z., et al.: Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv preprint arXiv:1609.08144 (2016).
23. Klein, J., Moeschberger, M.: Survival analysis: techniques for censored and truncated data. Springer Science & Business Media (2006).
24. Kalbfleisch, J., Prentice, R.: The statistical analysis of failure time data. Vol. 360. John Wiley & Sons (2011).
25. Prentice, R., Breslow, N.: Retrospective studies and failure time models. Biometrika, 65(1), 153-158 (1978).
26. Hung, H., Chiang, C.: Estimation methods for time-dependent AUC models with survival data. Canadian Journal of Statistics, 38(1), 8-26 (2010).
27. Chambless, L., Cummiskey, C., Cui, G.: Several methods to assess improvement in risk prediction models: extension to survival analysis. Statistics in Medicine, 30(1), 22-38 (2011)..
28. Harrell Jr, F., Lee, K., Califf, R., Pryor, D., Rosati, R.: Regression modelling strategies for improved prognostic prediction. Statistics in Medicine, 3(2), 143-152 (1984).