ISBN: 978-981-18-3959-7
2022 the 12th International Workshop on Computer Science and Engineering (WCSE 2022)
doi: 10.18178/wcse.2022.06.039

Deeptab: A Deep Neural Network for Non-uniform Tabular Data

Xinyu Jiang [1], Hongyu Guo [1] and Qian Li [1+]

[1] The 15th Research Institute of China Electronics Technology Group Corporation

Abstract. Over the last years, deep neural networks have been the optimal solution for most tasks in several fields. Moreover, various algorithms have been implemented in industrial applications, such as face recognition, language translation, object classification, and object detection. However, when we use deep neural networks for practical applications on tabular data, we find the problem of non-uniform training data and test data. The training data usually exhibit high-quality forms, while the actual prediction data are lower quality than the training data. We propose Deeptab, a deep neural network for non-uniform tabular data based on this problem. Specifically, we make the training data mimic the actual data situation as much as possible and improve the robustness and flexibility of the network by coding each feature separately. We eventually tested our approach on public datasets and our real datasets and achieved better results. We also tested the impact of training data quality and found a better treatment for non-uniform tabular data.

Keywords: tabular data, data mask, deep neural network.

1. Introduction

Deep neural networks have achieved great success in several fields such as image[1], video[2], and text[3]. They occupy the optimal solution for several tasks in real-world applications, such as face recognition[5], language translation[6], object classification[7][8], and object detection[4]. However, deep neural networks are not widely used in the largest dataset of real-world applications, tabular data[9]. So we hope to use deep neural networks for real-world applications concerning tabular data.

When applying deep neural networks to real-world applications, we found a problem that the data used for training are, in most cases, of high quality relative to the data used in real-world applications, meaning that they have low or no missing data. This data feature is not fully representative of the actual situation at the time of application. In real use, missing data can be caused by reasons such as insufficient collection or users without data on that feature. Such missings are unpredictable and cannot be determined during training data. We call this kind of data with high-quality training data, and low-quality test data non-uniform data.

When faced with actual prediction samples, the models trained by such "high quality" data often do not give the best results. The models we train are not robust, so we started to think about how to use the current high-quality data to train models that can be used in real-world scenarios. Although the absence of accurate data is unpredictable, there are ways to simulate the raw data and bring it as close as possible to the target data features. In turn, we can use deep neural networks to train on such data to achieve better results.

So we proposed a data simulation method that uses the data mask to simulate the missing cases inaccurate data to improve the robustness of the network. We also change the coding of the neural network to improve its flexibility of the network. In these ways, we improve the capability of deep neural networks. We applied our network to public datasets and our own data and found that our network worked better after experimentation.

2. Background

2.1. Deep Neural Networks on the Tabular Data

Deep neural networks, using their multi-layer structure powerful parameter adjustment ability, have achieved excellent results in many fields such as text[10], image[11], video[12], and audio[13]. However,

[+] Corresponding author. Tel.: +86 18612929933.
E-mail address: qian.li.china@hotmail.com.

extensive research on tabular data is missing, partly because machine learning methods still occupy the optimal solution for most of the table tasks[9].

However, the deep neural network has a series of unique advantages, such as deep neural networks can perform Embedding learning, achieve more prosperous feature extraction, and flexible loss design, suitable for some complex task scenarios[14]. In addition, the deep neural network can also design the network according to the characteristics of tasks and data, which is more flexible and closer to the practical task.

2.2. Data Mask

Data mask are often used in natural language processing and computer vision[15][16]. People mask process parts of the comprehensive data to improve the network's ability. The general mask methods include random mask, block mask, and grid mask[17]. For mask data, the position of the mask is set to empty, and in image processing, the position of the mask is set to 0. For example, in the field of image, researchers often use the mask in the field of self-supervision, taking the unprocessed picture themselves as the target, hoping that the network can fill the mask points in the image to improve the robustness of the network and strengthen the coding ability of the network, namely the feature extraction ability. This mask can exploit the features of deep neural networks to enhance their capabilities.

3. Methods

For neural networks, it is a truth that better data corresponds to better networks, but this situation assumes that the current data is roughly the same quality as the data we will face in the future. However, we often encounter the problem that the quality of the data used for prediction is often not as high as the quality of the training data we have carefully collected when applying the network. The trained model with such high-quality data will have a poor prediction rate for low-quality data when it is actually used in the future.

To cope with this phenomenon, we consider making the data fit as closely as possible to the scenarios of future practical applications during training to enhance the robustness of the model and, at the same time, use the existing high-quality data to ensure the correctness of the model. We propose Deeptab, a deep neural network for non-uniform tabular data based on the above requirements. The overall structure of the network is shown in Fig. 1

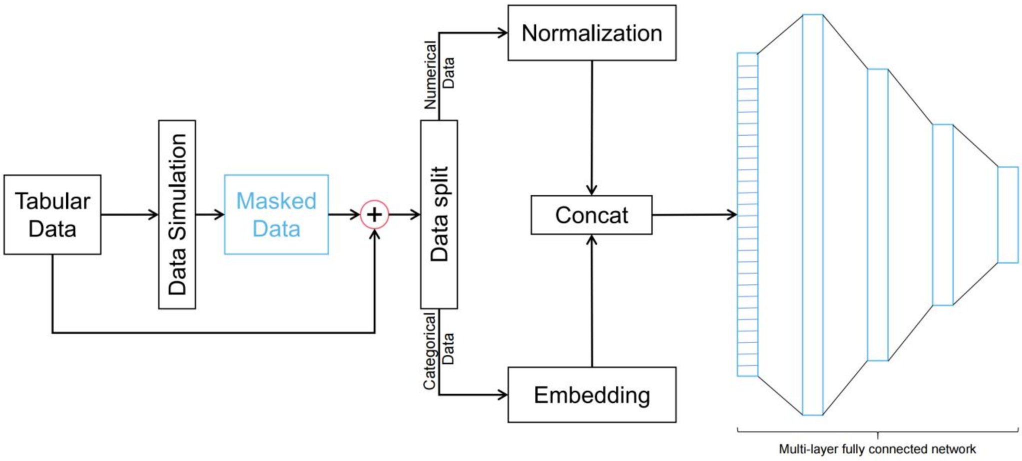
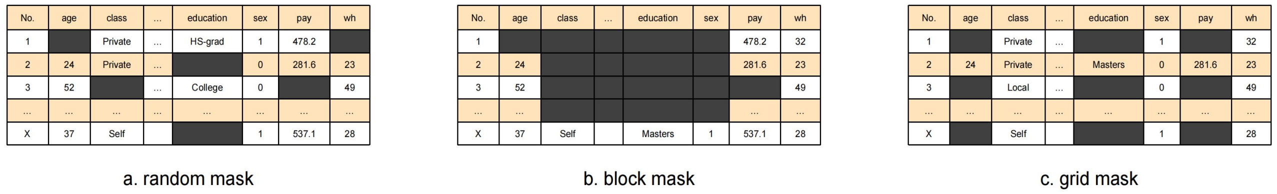


Fig. 1: The network structure of Deeptab. The data mask module completes the data simulation, the data split module completes the data segmentation, the Normalization module and the Embedding module encode the categorical data and the numerical data, respectively, and the Concat module splices the encoded data.

The overall structure consists of two parts. The first is the data simulation part. In the data simulation module, the original "high-quality" data is processed into the missing "low-quality" data to complete the simulation of the training data to the actual prediction data, and the processed data is stitched with the original data and used as training data. The other part is the neural network part. After the data simulation part, the final training data set is fed into the deep neural network, which encodes and splices the different data. The original data is transformed into vector form, and the data features are extracted from it.

3.1. Data Simulation

Data simulation, i.e., the simulation of missing data, aims to make its missing situation as close as possible to the situation of actual data. In order to improve the missing rate of the training data, we adopt the way of masking the data. The current mainstream mask methods are random mask, block mask and grid mask as shown in Fig. 2.



No.	age	class	...	education	sex	pay	wh
1	■	Private	...	HS-grad	1	478.2	■
2	24	Private	...	■	0	281.6	23
3	52	■	...	College	0	■	49
...	...	...	...	...	...	...	...
X	37	Self		■	1	537.1	28

a. random mask

No.	age	class	...	education	sex	pay	wh
1	■	■	■	■	■	478.2	32
2	24	■	■	■	■	281.6	23
3	52	■	■	■	■	■	49
...	...	■	■	■	■	...	...
X	37	Self		Masters	1	537.1	28

b. block mask

No.	age	class	...	education	sex	pay	wh
1	■	Private	...	■	1	■	32
2	24	Private	...	Masters	0	281.6	23
3	■	Local	...	■	0	■	49
...	...	...	...	...	...	...	...
X	■	Self		■	1	■	28

c. grid mask

Fig. 2: Schematic figure of random mask, block mask and grid mask.

In addition to simulating the missing rate, we also need to make the missing data as similar as possible to the real data. The chunks and regular missingness of the real data are very low, so we use random mask for the data. For each sample s in original dataset N_o, every features f_i will be applied with random mask,

$$S_{f_i} = S_{f_i} \times P(f_i), \tag{1}$$

where $P(f_i) = \{0, 1\}$ is the probability of masking f_i in accordance. The simulated data N_P is spliced with the original data N_o, satisfying both the missing data in the training set as well as the high-quality data that can guide the correctness of the network.

3.2. Network Structure

The neural network part first partitions the data according to numerical data and categorical data, and then encodes them separately to vectorize them, where numerical data is normalized and categorical data is encoded using Embedding. To enhance the flexibility of the network, we encode all the features f with the data set N separately, and for the encoded features, we splice the data by concat to preserve their native information and obtain the native vector of the sample. Then we classify them by multilayer perceptron (MLP). The loss function is set as cross-entropy loss,

$$L = -\sum_{c=1}^{M} y_c \log(\hat{y}_c), \tag{2}$$

where M is the number of categories, y_c is the filter for the category, and y is 1 if y is the same as c; otherwise, y is 0. $\hat{y}_c$ is the predicted probability of the current category.

4. Experiment

4.1. Settings

For the comparison methods in the experiments, we have chosen typical methods in machine learning such as Decision tree[18], XGBoost[19], lightGBM[20], and deep neural network method MLP[21].

The dataset is the Adults dataset and the Company Data dataset. The Company Data is a dataset of 5000 samples collected according to certain metrics based on the actual task requirements, is divided into 5 categories containing 52 features, including numerical data and category data.

4.2. Comparison of Methods

We first simulated the data with a missing probability of 20%, i.e., for each feature of each sample in the original dataset N_o, the probability of its nulling is 20%. The processed data N_p is obtained in this way, and then this data is combined with the original data to obtain the complete data set N_c, which is input as training data into the neural network for training. We tested the model trained on the original data set N_o, the model trained only on the simulated data set N_p, and the model trained on the complete data set N_c. All three models were tested, and the results shown in Table 1 were obtained.

Table 1: Performance on Adults dataset and our Company Data dataset use accuracy(%)

Methods	Adults	Company Data
DT[18]	68.2±0.2	57.3±0.2
XGBoost[19]	82.6±0.2	72.4±0.2
LightGBM[20]	85.7±0.2	78.8±0.2
MLP[21]	83.3±0.3	74.1±0.2
Deeptab	**87.4±0.2**	**81.2±0.2**

Table 1 shows that our method has some improvement over the current mainstream tabular data processing methods and outperforms other methods on public data sets. Furthermore, our approach is more than 2% better than the industry's current popular machine learning methods. This indicates that the network's ability is improved by mask when applied to have missing test data. Our method has a certain degree of improvement in the robustness of the network.

4.3. Effect of Missing Rate

Real-world missing rates are often unpredictable, and it is essential to consider as many scenarios as possible when training the network. So we explored the missing rate of training data versus the missing rate of test data. When we simulated the data, we processed multiple sets of data simulations based on the probability $p \in P$, where the missing probability $p_i = \{10, 20, 30, 40\}$.

We experimented with all the data N_{p_i} under each probability p_i and recorded the accuracy of the trained neural network by predicting it on the test set with different missing rates. We also spliced all the N_{p_i} to obtain N_m for training and tested the performance of the network as well. The parameters were set to learning rate=0.01 and trained for 1000 epochs, and the results are shown in Table 2.

Table 2: Performance on Adults dataset for different missing rate use accuracy(%).

methods	Test_10	Test_20	Test_30
p_1	85.8±0.2	85.1±0.1	78.9±0.4
p_2	85.9±0.2	85.5±0.2	78.3±0.4
p_3	83.2±0.2	84.2±0.2	77.8±0.4
p_4	82.4±0.2	81.9±0.3	77.6±0.4
Deeptab_m	**88.6±0.2**	**88.3±0.2**	**86.6±0.2**

We found that different missing rates, such as 10% and 20%, do not have much effect on the lower missing rate training set. Instead, the accuracy decreases when it reaches more than 30%, possibly because the neural network does not acquire the main features due to too much missing data during training. However, when we stitch all the datasets together for training, the obtained models show excellent results in the training sets with different missing rates. We use high-quality data to enhance the correctness of the network while incorporating low-quality data to train the robustness of the network.

5. Conclusion

This study uses the mask approach to simulate the data, expecting that the network model can still perform well in real applications when faced with low-quality data. Each feature of the data is coded separately to enhance the flexibility of the neural network. After our experiments, this approach can improve the robustness and flexibility of the network to a certain extent and achieves good results on both public and our real dataset.

6. Acknowledgement

This work was supported in part by the National Key Research and Development Program of China under Grant No. 2020AAA0105100

7. References

[1] Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[2] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[3] Oord A, Dieleman S, Zen H, et al. Wavenet: A generative model for raw audio[J]. arXiv preprint arXiv:1609.03499, 2016.

[4] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.

[5] Kortli Y, Jridi M, Al Falou A, et al. Face recognition systems: A survey[J]. Sensors, 2020, 20(2): 342.

[6] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.

[7] Zhao B, Feng J, Wu X, et al. A survey on deep learning-based fine-grained object classification and semantic segmentation[J]. International Journal of Automation and Computing, 2017, 14(2): 119-135.

[8] Chunhua Jia, Shuai Zhu, Wenhai Yi, Yu Wu, Leilei Wu, WeiweiCai , "GraftNet: An Efficient and Flexible Multi-label Image Classification and Its Application," 2021 The 11th International Workshop on Computer Science and Engineering (WCSE 2021), pp. 70-78, Shanghai, China, June 19-21, 2021.

[9] Shwartz-Ziv R, Armon A. Tabular data: Deep learning is not all you need[J]. Information Fusion, 2022, 81: 84-90.

[10] Minaee S, Kalchbrenner N, Cambria E, et al. Deep learning--based text classification: a comprehensive review[J]. ACM Computing Surveys (CSUR), 2021, 54(3): 1-40.

[11] Minaee S, Boykov Y Y, Porikli F, et al. Image segmentation using deep learning: A survey[J]. IEEE transactions on pattern analysis and machine intelligence, 2021.

[12] Xu Y, Wei H, Lin M, et al. Transformers in computational visual media: A survey[J]. Computational Visual Media, 2022, 8(1): 33-62.

[13] Zhu H, Luo M D, Wang R, et al. Deep audio-visual learning: A survey[J]. International Journal of Automation and Computing, 2021, 18(3): 351-376.

[14] C. Zhu, L. Cao and J. Yin, "Unsupervised Heterogeneous Coupling Learning for Categorical Representation," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 1, pp. 533-549, 1 Jan. 2022, doi: 10.1109/TPAMI.2020.3010953.

[15] Shorten C, Khoshgoftaar T M. A survey on image data augmentation for deep learning[J]. Journal of big data, 2019, 6(1): 1-48.

[16] Wei C, Fan H, Xie S, et al. Masked Feature Prediction for Self-Supervised Visual Pre-Training[J]. arXiv preprint arXiv:2112.09133, 2021.

[17] He K, Chen X, Xie S, et al. Masked autoencoders are scalable vision learners[J]. arXiv preprint arXiv:2111.06377, 2021.

[18] Myles A J, Feudale R N, Liu Y, et al. An introduction to decision tree modeling[J]. Journal of Chemometrics: A Journal of the Chemometrics Society, 2004, 18(6): 275-285.

[19] Chen T, He T, Benesty M, et al. Xgboost: extreme gradient boosting[J]. R package version 0.4-2, 2015, 1(4): 1-4.

[20] Ke G, Meng Q, Finley T, et al. Lightgbm: A highly efficient gradient boosting decision tree[J]. Advances in neural information processing systems, 2017, 30.

[21] Gardner M W, Dorling S R. Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences[J]. Atmospheric environment, 1998, 32(14-15): 2627-2636.