Zhang *et al. BMC Medical Informatics and Decision Making* 2020, **20**(Suppl 3):123
https://doi.org/10.1186/s12911-020-1118-z

BMC Medical Informatics and Decision Making

RESEARCH Open Access

Enhanced character-level deep convolutional neural networks for cardiovascular disease prediction

Zhichang Zhang[1*†], Yanlong Qiu[2†], Xiaoli Yang[3] and Minyu Zhang[4]

From 5th China Health Information Processing Conference
Guangzhou, China. 22-24 November 2019

Abstract

Background: Electronic medical records contain a variety of valuable medical information for patients. So, when we are able to recognize and extract risk factors for disease from EMRs of patients with cardiovascular disease (CVD), and are able to use them to predict CVD, we have the ability to automatically process clinical texts, resulting in an improved accuracy of supporting doctors for the clinical diagnosis of CVD. In the case where CVD is becoming more worldwide, predictive CVD based on EMRs has been studied by many researchers to address this important aspect of improving diagnostic efficiency.

Methods: This paper proposes an Enhanced Character-level Deep Convolutional Neural Networks (EnDCNN) model for cardiovascular disease prediction.

Results: On the manually annotated Chinese EMRs corpus, our risk factor identification extraction model achieved 0.9073 of F-score, our prediction model achieved 0.9516 of F-score, and the prediction result is better than the most previous methods.

Conclusions: The character-level model based on text region embedding can well map risk factors and their labels as a unit into a vector, and downsampling plays a crucial role in improving the training efficiency of deep CNN. What's more, the shortcut connections with pre-activation used in our model architecture implements dimension-matching free in training.

Keywords: Chinese electronic medical record, CVD risk factors extraction, CVD prediction, Downsampling, Pre-activation, Dimension-matching free, Text region embedding

*Correspondence: zzc@nwnu.edu.cn
†Zhichang Zhang and Yanlong Qiu contributed equally to this work and are co-first authors
[1] College of Computer Science and Engineering, Northwest Normal University, 967 Anning East Road, 730070, Lanzhou, China
Full list of author information is available at the end of the article

Background

CVD is becoming more common worldwide and is becoming younger. According to data released by the World Health Organization, CVD is the number one cause of death worldwide, with more deaths from cardiovascular disease each year than any other cause of death. In 2016, an estimated 17.9 million people died of cardiovascular disease, accounting for 31% of all deaths worldwide. In its 2018 report, China's National Center for Cardiovascular Disease noted that CVD mortality remained at the top of 2016, higher than cancer and other diseases, and the number of patients was as high as 290 million.

As CVD risk increases in China, interest in strategies to mitigate it is growing. However, information on the prevalence and treatment of CVD in daily life is limited. But in the medical field, many hospitals have been able to systematically accumulate medical records for a large number of patients by introducing an EMR system. Deep learning has been successfully applied to medical field based on accumulated EMR data [1, 2]. In particular, many studies have been conducted to predict the risk of cardiovascular disease in order to prevent cardiovascular diseases with a high mortality rate globally [3]. Because EMR data is recorded based on patients visiting the hospital, and it contains information on the pathogenesis of cardiovascular disease. Therefore, we intend to extract key information using Convolutional Neural Networks (CNN). Table 1 is the key information we consider, including twelve risk factors. However, we found that there was a large amount of irrelevant information in most EMRs. For example, a record of a complete medical record contains only 10 records of information that is effective in causing disease. These excessively irrelevant information not only reduces CNN's emphasis on effective disease information, but also greatly delays the training time of neural networks. In this regard, we propose to extract the risk factors that cause disease in EMRs and bring along the Time Attribute of these risk factors. In fact, although the training time has decreased, the experimental results are not optimistic. After experimental analysis, we believe that the main reason is the lack of certain context information. In response to this situation, we proposed the EnDCNN. The region embedding method used by EnDCNN can enhance the correlation between risk factors. For example, an increased correlation between the risk of hypertension and the risk of controlling blood pressure can better predict whether the patient has heart disease. At the same time, our inspiration from the ResNet network proposed by He et al. [4] has deepened our own neural network to better extract key information. And for our deep CNN model features, we also have the downsampling method to further reduce training time. This makes it not only speed up the training time through our method, but also the experimental result F-score reaches 0.9516, which fully demonstrates the method we proposed and the efficiency of the network architecture we took out. In summary, our contribution is two-fold, which can be concluded as follows:

Our innovation proposes to extract the risk factor identification and bring along its corresponding label as the basis for CVD prediction. Recurrent Neural Networks (RNNs) generally read the whole text from beginning to end or vice versa sometimes, which makes it inefficient to process long texts [5]. In this regard, Huang et al. deal with Long Short-Term Memory (LSTM) in English text. In view of this, we propose to extract the risk factors and their corresponding labels recognition for the characteristics of the CNN network we use. This method not only avoids a large amount of non-critical information, but also reduces the time spent on model architecture training to some extent.

We propose the EnDCNN model architecture. Inspired by the application of region embedding by Johnson et al. [6] and the ResNet for image model architecture [4] proposed by He et al. We first convert the risk factors and their corresponding tags into corresponding vectors in characters by our character embedding trained in a specific field. Then, we built deep CNN to better extract key information. Finally, for our deep model architecture, we used the downsampling method to further speed up the training, and the final effect of the model is optimistic.

Methods

The main idea of this paper is to predict whether a patient has CVD by focusing on the risk factors in EMRs. First of all, we need to prepare the data we need. The user enters the appropriate input values from his/her EMR report. After this, the historical dataset is uploaded. The fact that most medical dataset may contain missing values makes this accurate prediction difficult. So, for this missing data, we have to transform the missing data into structured data with the help of a data cleaning and data imputation process. After preparing the data, we mainly perform the following two steps. Firstly, the risk factors in the EMRs and their corresponding labels identification are then extracted using the relatively mature entity recognition technology that has been developed. In addition to Age and Gender, the labels for other risk factors include the type of the risk factor and its temporal attributes. We only use the Conditional Random Field (CRF) layer to identify the F-score of the extraction result to reach 0.8994. When we use bidirectional LSTM with a CRF layer (BiLSTM-CRF), the F-score identifying the extraction results reached 0.9073. We did a lot of experiments and summarized why there is such a high F-score based on experimental and EMRs data analysis. Because there are 12 risk factors in the entire data, these risk factors are largely repeated in EMRs. This is great for the system we

Table 1 Attributes of CVD

No.	Attributes	Description
1.	Overweight/Obesity (O2)	A diagnosis of patient overweight or obesity
2.	Hypertension	A diagnosis or history of hypertension
3.	Diabetes	A diagnosis or a history of diabetes
4.	Dyslipidemia	A diagnosis of dyslipidemia, hyperlipidemia or a history of hyperlipidemia
5.	Chronic Kidney Disease (CKD)	A diagnosis of CKD
6.	Atherosis	A diagnosis of atherosclerosis or atherosclerotic plaque
7.	Obstructive Sleep Apnea Syndrome (OSAS)	A diagnosis of OSAS
8.	Smoking	Smoking or a patient history of smoking
9.	Alcohol Abuse (A2)	Alcohol abuse
10.	Family History of CVD (FHCVD)	Patient has a family history of CVD or has a first-degree relative (parents,siblings, or children) who has a history of CVD
11.	Age	The age of the patient
12.	Gender	The gender of patient

have proposed. In contrast, the BiLSTM-CRF model has better recognition performance, so we consider using it to extract the risk factors and corresponding labels in EMRs. These extracted risk factors that carry the corresponding labels serve as the basis for input and predict CVD. In the end, by using the EnDCNN model architecture, we can predict whether a patient has CVD. Figure 1 shows our entire model architecture.

BiLSTM-CRF

The architectures of BiLSTM-CRF model illustrated in Fig. 2. In the model, the BIO (Begin, Inside, Outside) tagging scheme is used.The $\mathbf{Q} = (\mathbf{q}_1, ..., \mathbf{q}_{k-3}, ..., \mathbf{q}_k)$ represents the context information carried by the character embedding trained by Skip-Gram.The HyC is represented as Hypertension and its temporal attribute is *Continue*. The HyD is represented as Hypertension and its temporal attribute is *During*. It is similar to the ones presented by Huang et al. [7], Lample et al. [8] and Ma and Hovy [9].

Given a sentence, the model predicts a label corresponding to each of the input tokens in the sentence. Firstly, through the embedding layer, the sentence is represented as a sequence of vectors $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_t, ..., \mathbf{x}_n)$ where n is the length of the sentence. Next, the embeddings are given as input to a BiLSTM layer. In the BiLSTM layer, a forward LSTM computes a representation $\overrightarrow{\mathbf{h}}_t$ of the sequence from left to right at every character t, and another backward LSTM computes a representation $\overleftarrow{\mathbf{h}}_t$ of the same sequence in reverse. These two distinct networks use different parameters, and then the representation of a character $\mathbf{h}_t = \left[\overrightarrow{\mathbf{h}}_t; \overleftarrow{\mathbf{h}}_t\right]$ is obtained by concatenating its left and right context representations. LSTM memory cell is implemented as Lample et al. [8] did.

Then a tanh layer on top of the BiLSTM is used to predict confidence scores for the character having each of the possible labels as the output scores of the network.

$$\mathbf{e}_t = \tanh(\mathbf{W}_e \mathbf{h}_t), \tag{1}$$

where the weight matrix $\mathbf{W}_e$ is the parameter of the model to be learned in training.

Finally, instead of modeling tagging decisions independently, the CRF layer is added to decode the best tag path in all possible tag paths. We consider $\mathbf{P}$ to be the matrix of scores output by the network. The t^{th} column is the vector e_t obtained by the Equation (1). The element $P_{i,j}$ of the matrix is the score of the j^{th} tag of the i^{th} character in the sentence. We introduce a tagging transition matrix $\mathbf{T}$, where $T_{i,j}$ represents the score of transition from tag i to tag j in successive characters and $T_{0,j}$ as the initial score for starting from tag j. This transition matrix will be trained as the parameter of model. The score of the sentence $\mathbf{X}$ along with a sequence of predictions $\mathbf{y} = (y_1, ..., y_t, ..., y_n)$ is then given by the sum of transition scores and network scores:

$$s(\mathbf{X}, \mathbf{y}) = \sum_{i=1}^{N} \left(T_{y_{i-1}, y_i} + P_{i, y_i}\right), \tag{2}$$

Then a softmax function is used to yield the conditional probability of the path $\mathbf{y}$ by normalizing the above score over all possible tag paths $\tilde{\mathbf{y}}$:

$$p\left(\mathbf{y}|\mathbf{X}\right) = \frac{e^{s(\mathbf{X},\mathbf{y})}}{\sum_{\tilde{\mathbf{y}}} e^{s}\left(\mathbf{X}, \tilde{\mathbf{y}}\right)}, \tag{3}$$

During the training phase, the objective of the model is to maximize the log-probability of the correct tag sequence.

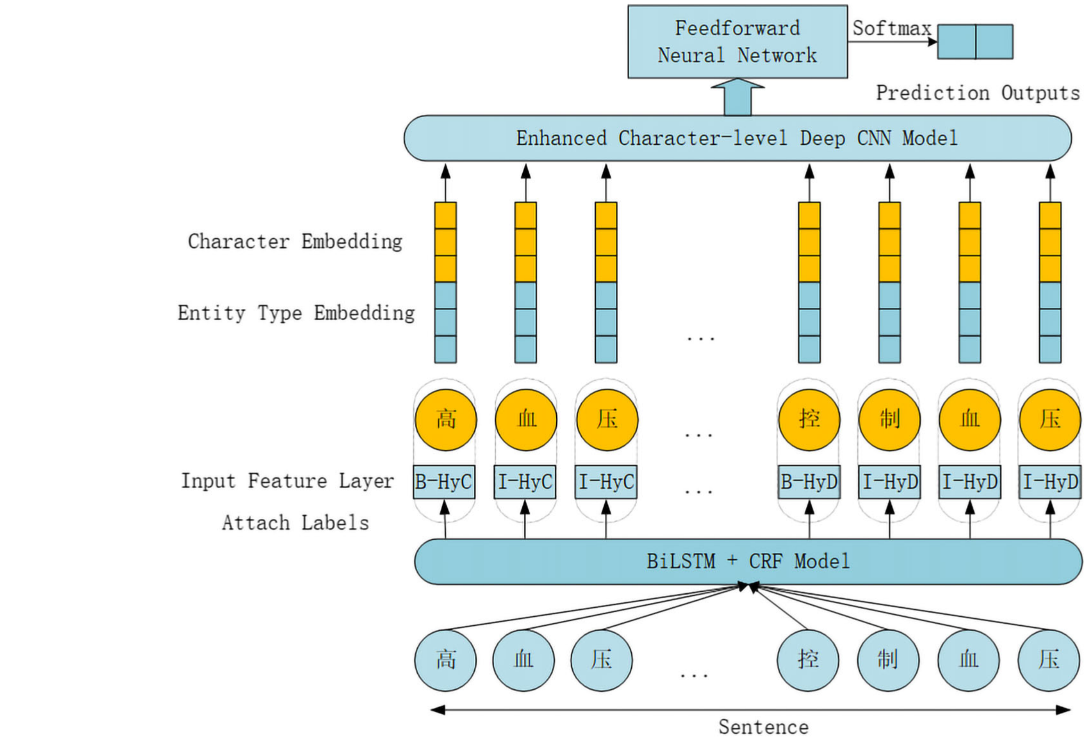


Fig. 1 The entire model architecture of our proposed

At inference time, we predict the best tag path that obtains the maximum score given by:

$$\arg_{\tilde{\mathbf{y}}} \max s\left(\mathbf{X}, \tilde{\mathbf{y}}\right), \tag{4}$$

This can be computed using dynamic programming, and the Viterbi algorithm [10] is chosen for this inference.

Overview of EnDCNN

Figure 3a is our proposed model EnDCNN. Figure 3b is He et al. proposed the ResNet network architecture for image. ⊕ indicates addition. The dotted red shortcuts in Fig. 3b perform dimension matching. EnDCNN is dimension-matching free. The first layer of our model performs text region embedding, which generalizes commonly used character embedding to the embedding of text

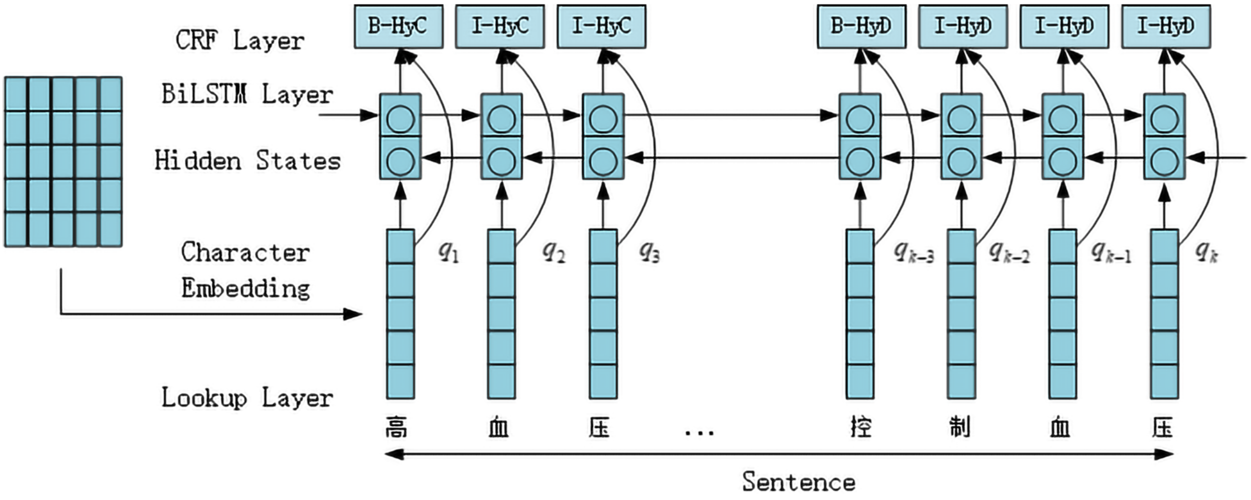
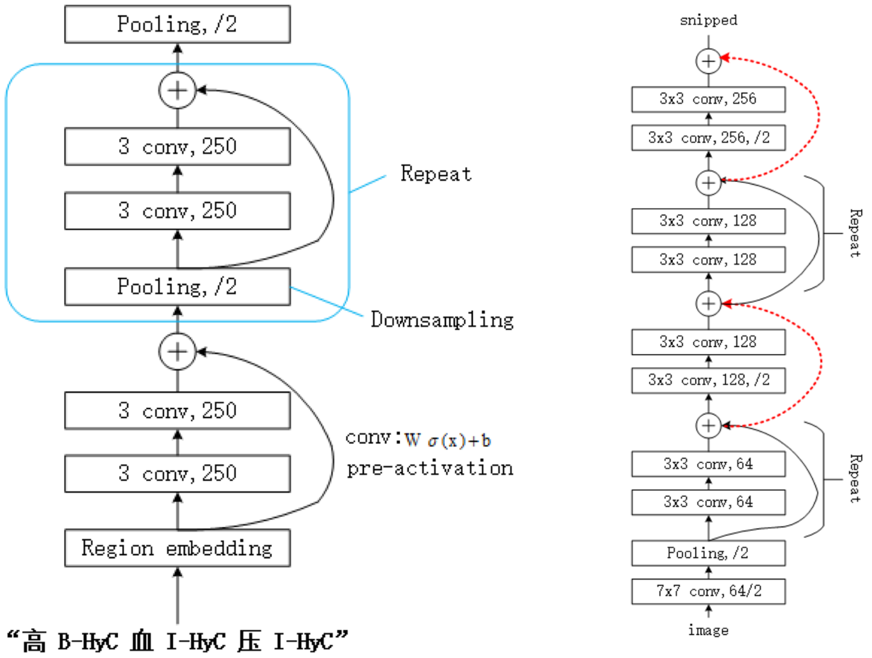


Fig. 2 The architecture of BiLSTM-CRF model



(a) Our proposed model EnDCNN (b) cf. ResNet for image [HZRS15]

Fig. 3 The architectures of our model and ResNet

regions covering one or more characters. It is followed by stacking of convolution blocks (two convolution layers and a shortcut) interleaved with pooling layers with stride 2 for downsampling. The final pooling layer aggregates internal data for each document into one vector. We use max pooling for all pooling layers. The key features of EnDCNN are as follows.

Downsampling without increasing the number of feature maps (dimensionality of layer output, 250 in Fig. 3a). Downsampling enables efficient represent-ation of long-range associations (and so more global information) in the text. By keeping the same number of feature maps, every 2-stride downsampling reduces the per-block computation by half and thus the total computation time is bounded by a constant.

Shortcut connections with pre-activation and identity mapping [11] for enabling training of deep networks.

Text region embedding enhances the relevance of individual character to character, the relevance of risk factors to their corresponding tags. When we use risk factors and their corresponding labels as a unit, text region embedding can also enhance the correlation between each unit. Therefore, accuracy can be improved by text region embedding.

Network architecture of EnDCNN

Downsampling with the number of feature maps fixed

After each convolution block, we perform max-pooling with size 3 and stride 2. That is, the pooling layer produces a new internal representation of a document by taking the component-wise maximum over 3 contiguous internal vectors, representing 3 overlapping text regions, but it does this only for every other possible triplet (stride 2) instead of all the possible triplets (stride 1). This 2-stride downsampling reduces the size of the internal representation of each document by half.

A number of models (Simonyan and Zisserman, 2015 [12]; He et al., 2015 [13], 2016 [11]; Conneau et al., 2016 [14]) increase the number of feature maps whenever downsampling is performed, causing the total computational complexity to be a function of the depth. In contrast, we fix the number of feature maps, as we found that increasing the number of feature maps only does harm - increasing computation time substantially without accuracy improvement, as shown later in the experiments. With the number of feature maps fixed, the computation time for each convolution layer is halved (as the data size is halved) whenever 2-stride downsampling is performed. Therefore, with EnDCNN, the total computation time is bounded by a constant - twice the computation time of a single block, which makes our deep networks computationally attractive.

In addition, downsampling with stride 2 essentially doubles the effective coverage (i.e., coverage in the original document) of the convolution kernel; therefore, after going through downsampling L times, associations among

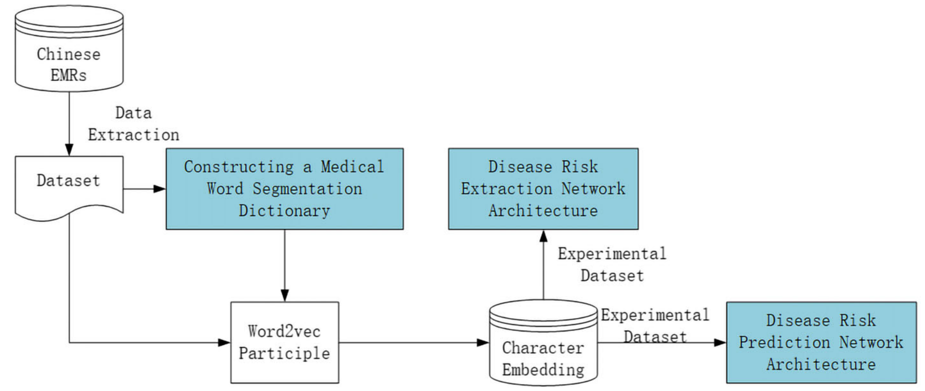

Chinese EMRs
Data Extraction
Dataset
Constructing a Medical Word Segmentation Dictionary
Disease Risk Extraction Network Architecture
Experimental Dataset
Word2vec Participle
Character Embedding
Experimental Dataset
Disease Risk Prediction Network Architecture

Fig. 4 Generate the character embedding for experiments

characters within a distance in the order of 2^L can be represented. Thus, deep CNN is computationally efficient for representing long-range associations and so more global information.

Shortcut connections with pre-activation To enable training of deep networks, we use additive shortcut connections with identity mapping, which can be written as $\mathbf{z}+f(\mathbf{z})$ where f represents the skipped layers [11]. In EnDCNN, the skipped layers $f(\mathbf{z})$ are two convolution layers with *pre-activation*. Here, pre-activation refers to activation being done *before* weighting instead of *after* as is typically done. That is, in the convolution layer of EnDCNN, $\mathbf{W}\sigma(\mathbf{x}) + \mathbf{b}$ is computed at every location of each document where a column vector $\mathbf{x}$ represents a small region (overlapping with each other) of input at each location, $\sigma(\cdot)$ is a component-wise nonlinear activation, and weights $\mathbf{W}$ and biases $\mathbf{b}$ (unique to each layer) are the parameters to be trained. The number of $\mathbf{W}$'s rows is the *number of feature maps* (also called the *number of filters* [13]) of this layer.We set activation $\sigma(\cdot)$ to the rectifier $\sigma(x) = max(x, 0)$. In our implementation, we fixed the number of feature maps to 250 and the kernel size (the size of the small region covered by x) to 3, as shown in Fig. 3a.

With pre-activation, it is the results of linear weighting $\mathbf{W}\sigma(\mathbf{x}) + \mathbf{b}$ that travel through the shortcut, and what is added to them at a ⊕ (Fig. 3a) is also the results of linear weighting, instead of the results of nonlinear activation ($\sigma(\mathbf{Wx}+\mathbf{b})$). Intuitively, such 'linearity' eases training of deep networks, similar to the role of constant error carousels in LSTM [15]. We empirically observed that pre-activation indeed outperformed 'post-activation', which is in line with the image results [11].

No need for dimension matching Although the shortcut with pre-activation was adopted from the *improved ResNet* of [11], our model is simpler than ResNet (Fig. 3b), as all the shortcuts are exactly simple *identity mapping* (i.e., passing data exactly as it is) without any complication for dimension matching. When a shortcut meets the 'main street', the data from two paths need to have the same dimensionality so that they can be added; therefore, if a shortcut skips a layer that changes the dimensionality, e.g., by downsampling or by use of a different number of feature maps, then a shortcut must perform dimension matching. Dimension matching for increased number of feature maps, in particular, is typically done by projection, introducing more weight parameters to be trained. We eliminate the complication of dimension matching by not letting any shortcut skip a downsampling layer, and by fixing the number of feature maps throughout the network. The latter also substantially saves computation time as mentioned above, and we will show later in our experiments that on our tasks, we do not sacrifice anything for such a substantial efficiency gain.

Text region embedding for EnDCNN

A CNN for disease prediction typically starts with converting each character in the text to a character vector

Table 2 Hyper parameters of EnDCNN

Parameter	Description	Value
d_w	Dimension of word embedding	100
lr	Learning rate	0.001
B	Batch size	64
kp	Each neuron's keep rate	0.5
dr	Decay rate for *lr*	0.99
ds	Number of decay steps	500
ks	Window size	3
m	Number of filters	250
s	Number of strides	2
n	Number of epochs	30
d_p	The depth of EnDCNN	15

Table 3 The comparison of each model for CVD prediction results

Model	*k*	Accuracy%	Precision%	Recall%	F-score%
EnDCNN	1	93.91	93.87	93.91	93.89
	7	**95.22**	**95.14**	**95.22**	**95.16**
	14	94.35	94.31	94.35	94.32
EnDCNN$_{(no\ labels)}$	3	**93.91**	**93.83**	**93.91**	**93.86**
	4	89.57	90.19	89.57	89.87
	5	86.52	86.41	86.52	86.47
SVM$_{(raw)}$		90.91	90.91	90.91	90.91
SVM$_{(no\ labels)}$		89.39	89.03	89.39	89.21
ConvNets$_{(raw)}$		92.83	92.64	92.83	92.73
LSTM$_{(raw)}$		92.24	93.46	92.73	93.09
LSTM$_{(risks\ with\ labels)}$		82.58	81.35	83.01	82.17

(character embedding). There is no exception in our mission, we need to put each Chinese character and its corresponding label. This is the entity type embedding layer and character embedding layer in Fig. 1. Then, we take a more general viewpoint as in [16] and consider text region embedding - embedding of a region of text covering one or more characters.

In the region embedding layer we compute $\mathbf{Wx} + \mathbf{b}$ for each character of a document where input $\mathbf{x}$ represents a k-character region (i.e., window) around the character in some straightforward manner, and weights $\mathbf{W}$ and bias $\mathbf{b}$ are trained with the parameters of other layers. Activation is delayed to the pre-activation of the next layer. Now let v be the size of vocabulary, and let us consider the types of straightforward representation of a k-character region for $\mathbf{x}$. We chose sequential input: the kv-dimensional concatenation of k one-hot vectors.

A region embedding layer with region size $k > 1$ seeks to capture more complex concepts than single characters in one weight layer, whereas a network with character embedding uses *multiple* weight layers to do this, e.g., character embedding followed by a convolution layer. Because we want to enter the risk factor and bring along its corresponding label, such as "高 B-HyC", we adjust the parameter k in units of 7 according to the actual situation of the experimental data.

Results

Dataset and evaluation metrics

Our dataset contains two corpora. The first corpus came from a hospital in Gansu Province with 800,000 unlabeled EMRs of internal medicine. The dataset was mainly used to train and generate our character embedding. In Fig. 4, we also added a dictionary of risk factors during the training. In this way, the Skip-Gram model in word2vec we use can better make each character in the risk factor more compact. The other one is from the Network Intelligence Research Laboratory of the Language Technology Research Center of the School of Computer Science, Harbin Institute of Technology, which contains 1186 EMRs. This corpus intends to be used to develop a risk factor information extraction system that, in turn, can be applied as a foundation for the further study of the progress of risk factors and CVD [17]. For the corpus, we divided it into CVD and no CVD according to the clinically diagnosed disease in the electronic medical record. In the corpus we used, there were 527 EMRs and 132 EMRs. The basis comes from the following two parts: On the one hand, according to the definition of CVD by the World Health Organization[18]. On the other hand, the first (symptoms) and the third (diseases) in the book of *Clinical Practical Cardiology* [19].

Our experiments involve 12 risk factors which are Overweight/Obesity (O2), Hypertension, Diabetes, Dyslipidemia, Chronic Kidney Disease (CKD), Atherosis, Obstructive Sleep Apnea Syndrome (OSAS), Smoking, Age, Gender, Abuse (A2), Family History of CVD

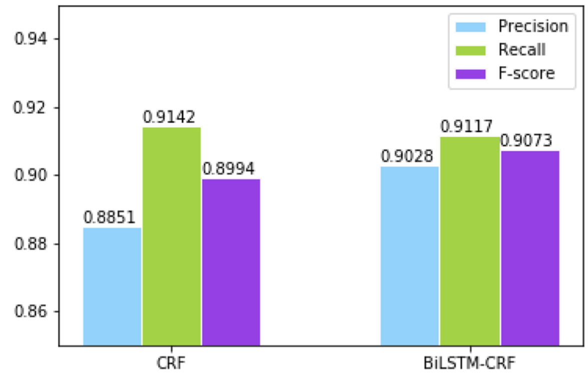


Fig. 5 Comparison of CRF and BiLSTM-CRF models

Table 4 The performance of each model at different embedding

Model	Accuracy%	Precision%	Recall%	F-score%
$ConvNets_{(no\ pre-emd)}$	91.67	90.87	91.24	91.05
$LSTM_{(no\ pre-emd)}$	81.82	79.45	82.36	80.88
$EnDCNN_{(no\ pre-emd,k=7)}$	94.13	94.06	94.13	94.09
$EnDCNN_{(no\ re-emd)}$	88.04	87.54	88.04	87.79

(FHCVD), as shown in Table 1. In addition, the dataset also includes 4 temporal attributes which are *Continue*, *During*, *After*, *Before*. Since risk factors of Age and Gender have no temporal attributes, we have added a temporal attribute: None.

In dataset, it consists of unstructured data, meaning data which is not in well-formed data. Mostly medical data is not in proper format. For the missing data, imputation and data cleaning are necessary. The unwanted data and noisy data must remove from dataset so that we get structured data.

In the experiment, our training set contained 461 EMRs, the test set contained 132 EMRs, and the development set contained 66 EMRs. *Accuracy*, *Precision*, *Recall* and *F-score* are used as evaluation metrics.

Models and parameters

We carry out the experiments to compare the performance of our model with others described in the following.

CRF: This model was used by Mao, et al. [20] recognized the named entity in the electronic medical records based on Conditional Random Field.

BiLSTM-CRF: This model was used by Li, et al. In order to realize automatic recognition and extraction of entities in unstructured medical texts, a model combining language model conditional random field algorithm (CRF) and Bi-directional Long Short-term Memory networks (BiLSTM) is proposed [21].

SVM: This model was used by S. Menaria, et al. As a traditional machine learning method, the support vector machine algorithm performs well in [22].

ConvNets: This model was used by Xiang, et al. [23] offers an empirical exploration on the use of character-level convolutional networks (ConvNets) for text classification.

LSTM: This model was used by Xin, et al. [24], which proposed an LSTM network with fully connected layer and activation layers.

EnDCNN: This is the model proposed in this paper. Table 2 gives the chosen hyper-parameters for all experiments. We tune the hyper-parameters on the development set by random search. We try to share as many hyper-parameters as possible in experiments.

Experimental results

We did a rich comparative experiment on our own model itself and other models:

In Fig. 5, we performed a comparison of the CRF and BiLSTM-CRF models for the identification of risk factors in EMRs. The precision values for the two models are: 0.8851 and 0.9028; The recall values for the two models are: 0.9142 and 0.9117; The F-score values for the two models are: 0.8994 and 0.9073, respectively. It is clear that the BiLSTM-CRF model outperforms the CRF. So, we

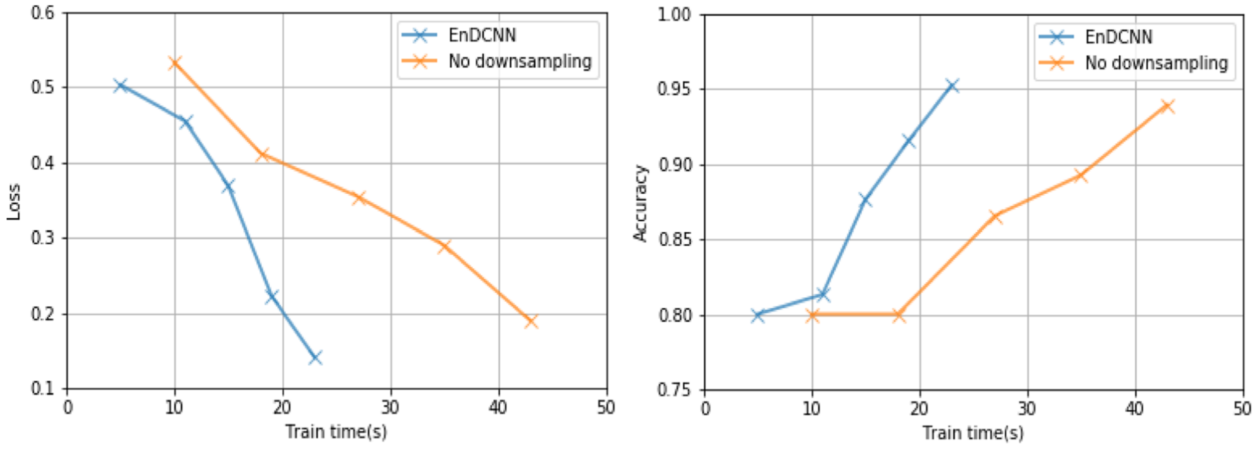


Fig. 6 Training efficiency

chose the BiLSTM-CRF model as our extractor for risk factors in EMRs.

In Table 3, we show the comparison between the previous model and our proposed EnDCNN model for accuracy, precision, recall, and F-score. In addition, we also compared the performance of EnDCNN models with different ranges (*k* values) for region embedding. And the performance of each model when the dataset is the original EMRs, the risk factor with the label, or the risk factor without the label.

In Table 4, we compared four cases: (1) The performance of the ConvNets model in random embedding; (2) The performance of the LSTM model in random embedding; (3) When our model is in random embedding and the region embedding size is 7; (4) The performance of our model without region embedding, that is, only use our pre-trained embedding.

In Fig. 6, we show a comparison of the training efficiencies of our model without downsampling and with downsampling. We plot the loss and accuracy in relation to the computation time - the time spent for training task using our performances on a GPU. We recorded five iterations from the start of training to the optimal training.

Discussion

Table 3 shows the overall classification performance of different models on our evaluation corpus. It can be seen that when the region embedding range size is 7, our EnDCNN method is superior to other methods in all evaluation indicators. On the data without the risk factor labels, our model performed well when the region embedding was 3. From the performance of our model in Tables 3 and 4, we can use the character embedding pre-trained by medical EMRs to help improve the performance of the model. Not only that, but from the performance of our model in Table 4 without region embedding, the importance of region embedding to the performance of our model. For Fig. 6, we can clearly see that downsampling is critical to the training efficiency of deep convolutional neural networks like ours.

Conclusions

In this paper, the disease prediction experiment was carried out on the EnDCNN algorithm using structured data. We used CRF and BiLSTM-CRF algorithm to identify the risk of CVD and its corresponding risk factors. We have compared the results of the CRF algorithm with the BiLSTM-CRF algorithm and the accuracy of BiLSTM-CRF 90.73% which is more than CRF algorithm. With the help of region embedding, we used the character-level embedding to achieve greater results, and the disease prediction F-score reached 95.16%. On the other hand, the downsampling technique solves the problem of slower training time in deep CNN. What's more, the shortcut connections with pre-activation used in our model architecture implements dimension-matching free in training. In the end, we got accurate disease prediction as output, by giving the input as patients EMRs which help us to understand the level of disease prediction. This output predicted whether to have or not to have heart disease. Because of this system may leads in low time consumption and minimal cost possible for disease risk prediction. In the future, we will strengthen research on the pathogenic factors of CVD and improve the accuracy of CVD prediction as much as possible.

Abbreviations

CVD: Cardiovascular Disease; EnDCNN: Enhanced Character-level Deep Convolutional Neural Networks; CNN: Convolutional Neural Networks; RNNs: Recurrent Neural Networks; LSTM: Long Short-Term Memory; CRF: Conditional Random Field; BiLSTM: Bi-directional Long Short-term Memory Networks; BiLSTM-CRF: Bidirectional LSTM with a CRF Layer; SVM: Support Vector Machine; ConvNets: Character-level Convolutional Networks

Acknowledgements

We thank the anonymous reviewers for their insightful comments.

About this supplement

This article has been published as part of *BMC Medical Informatics and Decision Making Volume 20 Supplement 3, 2020: Health Information Processing*. The full contents of the supplement are available online at https://bmcmedinformdecismak.biomedcentral.com/articles/supplements/volume-20-supplement-3.

Authors' contributions

ZZ and QY had a part in conceiving the study, and have substantially contributed in writing and revising the manuscript. YX and ZM analyzed the risk factors for disease in Chinese electronic medical records and were the main contributors to the processing of experimental data. All author(s) read and reviewed the final manuscript.

Funding

The publication cost of this paper was supported by the National Natural Science Foundation of China(No. 61762081, No.61662067, No.61662068) and the Key Research and Development Project of Gansu Province (No.17YF1GA016). The funding body had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Availability of data and materials

The datasets used and analyzed during the current study are available from the corresponding author upon reasonable requests.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

[1]College of Computer Science and Engineering, Northwest Normal University, 967 Anning East Road, 730070, Lanzhou, China. [2]College of Computer Science and Engineering, Northwest Normal University, 967 Anning East Road, 730070, Lanzhou, China. [3]College of Computer Science and Engineering, Northwest Normal University, 967 Anning East Road, 730070, Lanzhou, China. [4]College of Computer Science and Engineering, Northwest Normal University, 967 Anning East Road, 730070, Lanzhou, China.

Published: 9 July 2020

References

1. Yue-Hei Ng J, Hausknecht M, Vijayanarasimhan S, Vinyals O, Monga R, Toderici G. Beyond short snippets: Deep networks for video classification. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Boston: IEEE; 2015.
2. Liang Z, Zhang G, Huang JX, Hu QV. Deep learning for healthcare decision making with EMRS. In: 2014 IEEE International Conference on Bioinformatics and Biomedicine, BIBM 2014, November 2-5, 2014. Belfast, United Kingdom: IEEE; 2014. p. 556–9.
3. Wang J, Ding H, Bidgoli FA, Zhou B, Iribarren C, Molloi S, Baldi P. Detecting cardiovascular disease from mammograms with deep learning. IEEE Trans Med Imaging. 2017;36(5):1172–81.
4. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, June 27-30, 2016. Las Vegas, NV, USA: IEEE; 2016. p. 770–8.
5. Huang T, Shen G, Deng Z. Leap-lstm: Enhancing long short-term memory for text categorization. In: Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, August 10-16, 2019. Macao, China: International Joint Conferences on Artificial Intelligence Organization; 2019. p. 5017–23.
6. Qiao C, Huang B, Niu G, Li D, Dong D, He W, Yu D, Wu H. A new method of region embedding for text classification. In: 6th International Conference on Learning Representations, ICLR 2018, April 30 - May 3, 2018, Conference Track Proceedings. Vancouver: OpenReview.net; 2018.
7. Huang Z, Xu W, Yu K. Bidirectional LSTM-CRF models for sequence tagging. CoRR 1508.01991. 2015;abs/1508.01991.
8. Lample G, Ballesteros M, Subramanian S, Kawakami K, Dyer C. Neural architectures for named entity recognition. In: NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, June 12-17, 2016. San Diego California, USA: Association for Computational Linguistics; 2016. p. 260–70.
9. Ma X, Hovy EH. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Volume 1: Long Papers. Berlin, Germany: Association for Computational Linguistics; 2016.
10. Viterbi AJ. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. IEEE Trans Inf Theory. 1967;13(2):260–9.
11. He K, Zhang X, Ren S, Sun J. Identity mappings in deep residual networks. In: Computer Vision - ECCV 2016 - 14th European Conference, October 11-14, 2016, Proceedings, Part IV. Amsterdam, The Netherlands: Springer International Publishing; 2016. p. 630–45.
12. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. In: 3rd International Conference on Learning Representations, ICLR 2015, May 7-9, 2015, Conference Track Proceedings. San Diego: Arxiv.org; 2015.
13. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, June 27-30, 2016. Las Vegas, NV, USA: IEEE; 2016. p. 770–8.
14. Conneau A, Schwenk H, Barrault L, LeCun Y. Very deep convolutional networks for natural language processing. CoRR 1606.01781. 2016;abs/1606.01781.
15. Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. 1997;9(8):1735–80.
16. Johnson R, Zhang T. Semi-supervised convolutional neural networks for text categorization via region embedding. In: Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015. Montreal: Curran Associates, Inc.; 2015. p. 919–27.
17. Su J, He B, Guan Y, Jiang J, Yang J. Developing a cardiovascular disease risk factor annotated corpus of chinese electronic medical records. BMC Med Inf Decis Making. 2017;17(1):117–1711,.
18. The details of Cardiovascular diseases (CVDs) come from World Health Organization (WHO). https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds). Accessed 17 Feb 2020.
19. Guo J. Clinical Practical Cardiology. New Haven: Peking University Medical Press; 2015.
20. Mao X, Li F, Duan Y, Wang H. Named entity recognition of electronic medical record in ophthalmology based on crf model. In: 2017 International Conference on Computer Technology, Electronics and Communication (ICCTEC). Dalian: IEEE; 2017. p. 785–8.
21. Li W, Song W, Jia X, Yang J, Wang Q, Lei Y, Huang K, Li J, Yang T. Drug specification named entity recognition base on BILSTM-CRF model. In: 43rd IEEE Annual Computer Software and Applications Conference, COMPSAC 2019, July 15-19, 2019, Volume 2. Milwaukee, WI, USA: IEEE; 2019. p. 429–33.
22. Mareeswari V, Saranya R, Mahalakshmi R, Preethi E. Prediction of diabetes using data mining techniques. Res J Pharm Technol. 2017;10(4): 1098. https://ieeexplore.ieee.org/document/8553959.
23. Zhang X, Zhao JJ, LeCun Y. Character-level convolutional networks for text classification. In: Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015. Montreal: Curran Associates, Inc.; 2015. p. 649–57.
24. Hong X, Lin R, Yang C, Zeng N, Cai C, Gou J. Predicting alzheimer's disease using lstm. IEEE Access. 2019;PP:1–1.

Publisher's Note

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