Applied Intelligence
https://doi.org/10.1007/s10489-021-02509-w

Pattern recognition of epilepsy using parallel probabilistic neural network

Chen Gong[1,2,3] · Xingchen Zhou[1] · Yunyun Niu[1]

Accepted: 5 May 2021

Abstract

Accurate and rapid pattern recognition of epilepsy from intracranial electroencephalogram (iEEG) recordings is important for medical diagnostics. In this paper, three algorithms based on discrete wavelet transform (DWT) analysis and parallel probabilistic neural network, SA-PNN, SA-PPNN, and LSA-PPNN, are presented to identify iEEG recordings and detect epileptic seizures. Simulated annealing (SA) and local simulated annealing (LSA) are utilized to optimize network parameters of probabilistic neural network classifier, respectively. The combinations of different features are utilized as the input vectors of classifiers to complete classification tasks. Experiments are conducted to deal with five different classification tasks. Compared with non-parallel probabilistic neural network algorithm (SA-PNN), the running time of parallel probabilistic neural network algorithm (SA-PPNN) is shortened by 2.18 times. Compared with SA-PPNN, the average operating time of LSA-PPNN is reduced by 9.97 times. The reason is that LSA-PPNN trains and optimizes parameters with local data firstly and then brings the parameters into the global training data sets to train the network for a test. As the amount of data increases, the superiority over LSA-PPNN is getting more distinct. Our methods are also compared with other existing relative research. Experimental results prove that our methods are much more competitive. In particular, for the classification task C-D, the classification accuracy of our method reaches 83.3%, which is much higher than previous results.

Keywords Intracranial electroencephalogram (iEEG) · Epilepsy · Discrete wavelet transform (DWT) · Parallel computing · Local simulated annealing (LSA) · Probabilistic neural network (PNN)

1 Introduction

Recently, epilepsy becomes the most common neurological diseases in the world. It is result from plenty of abnormal activities with nerve cell. Not only can epilepsy make several potential health problems, but also influence normal life seriously due to paroxysmal. In clinic, we usually utilize intracranial EEG (iEEG) signals measuring activities of nerve cell in our brain. Under epilepsy condition, the activities of the iEEG are usually divided into four classes: normal, pre-ictal, post-ictal, epileptic, where ictal is the stage from the start to the end of epileptic seizure activities [1]. In epilepsy research, the essential part is how to distinguish pre-ictal, epileptic seizures, and post-ictal states with existing iEEG signals [2]. This problem seems simple enough, but it is so time-consuming that neuroscientists are supposed to analyze and classify numerous iEEG signals by visual inspection. Furthermore, according to [3], there are only 92% of the inter-expert sensitivity achieved with four human experts, so it is necessary to enhance the classification accuracy of visible inspection. Nowadays, so many researches attempt to design an automatic pattern classification methods for iEEG signals, in order to help neuroscientists completing the classification tasks simply.

This work was supported by the National Natural Science Foundation of China [grant numbers 61872325]; the Fundamental Research Funds for the Central Universities [grant number 2652019028].

✉ Yunyun Niu
yniu@cugb.edu.cn

Chen Gong
ChenG_abc@outlook.com

1 School of Information Engineering, China University of Geosciences in Beijing, Beijing, 100083, China

2 Institute of Automation, Chinese Academy of Sciences, Beijing, China

3 School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China

Published online: 02 June 2021

In automatic epileptic seizure recognition methods, usually it can be divided into two parts, feature extraction and signal recognition. Therefore, there are many feature extraction algorithms combined with classifiers based on supervised or unsupervised learning that have been utilized in detection of epileptic seizures recently.

Generally speaking, we often have three groups feature extraction methods, namely time domain, frequency domain, and combine time and frequency domain analysis methods. On the one hand, time-domain analysis methods usually extract entropy to represent complete signal, including Shannon entropy [2], Approximate entropy [2, 4, 5], Sample entropy [6], Log energy entropy [2], Wavelet entropy [7], and so on. On the other hand, frequency-domain analysis methods includes principal component analysis [8], autoregressive (AR) model [9], short time fourier transform (STFT) [10] and so on. As for time-frequency analysis, Wavelet transform (WT) considers both time and frequency point of views. So it possible to capture and localize transient features in the epileptic spikes precisely [11]; multifractional analysis method, discrete wavelet transformation (DWT), have been utilized to many classification problem, achieving good performance [12].

After features are extracted from iEEG signals, the classification algorithms are supposed to focus next. Pattern classification is the other essential part in our research. Lots of classifiers have been previously used to solve the problem, such as support vector machines (SVM) [13, 14], linear discriminant analysis (LDA) [15], naive Bayes algorithm [16], and so on. Among them, neural network (NN) is the most popular method, due to its amazing performances of adaptability [11, 12]. However, ANN algorithms contain too much parameters, and the performance of it depends heavily on parameters' quality, so how to select properly parameters become a serious problem. This paper utilize probabilistic neural network (PNN) as a classifier analyzing iEEG signals. Considerately, PNN classifier needs only one parameter whose value can be determined adaptively. In this paper, the contributions can be summarised as follows.

1. This paper presents three algorithms based on DWT analysis and PNN, namely SA-PNN, SA-PPNN and LSA-PPNN to identify iEEG recordings. SA and LSA algorithm are utilized to optimize network parameters of probabilistic neural network classifier, respectively. Comparing with existing relevant work, our methods can get better classification accuracy.
2. In order to reduce the running time of PNN classifier, parallel mechanism is adopted in our work. Input weight is calculated on several different processors simultaneously. Compared with serial PNN algorithm, the time consumption of weight computing between the input layer and the radial base layer can be reduced significantly. Parallel PNN (PPNN) improves computing efficiency and it is more suitable than PNN for large-scale data processing.
3. LSA-PPNN algorithm trains and optimizes parameters with local data firstly, and then brings the parameters into the global training data sets to train the network for test. The LSA algorithm helps to optimize the spreading factor of the PPNN classifier. Experiment results show that the LSA-PPNN algorithm not only greatly shortens the running time, but also obtains better accuracy.

2 Material and methods

2.1 Data segmentation

In this paper, we utilize the public data set which obtain the data sets previous ictal (C), post ictal (D) and epileptic (E) from the Department of Epileptology, University of Bonn. Andrzejak et al. [17]. Set C is obtained from the hippocampus of the opposite hemisphere of 5 epileptic patients. Set D is acquired from an epileptogenic zone. Set E is collected by measuring the pathogenic areas of epilepsy during epileptic seizures. The description of three data sets is presented in Table 1. In this paper, we mainly research the following five tasks.

In this photograph, we will introduce our data segmentation method for efficient use of samples. The data set is recorded by the 128 channels signal amplifiers. And the amplifiers are digitized with a sampling rate of 173.61 Hz, and the band pass filter sets between 0.53 HZ and 40.0 Hz. Sampling time is 23.6 second . Therefore, there are 4097 samples in each group data. Samples are divided into 8 equal data segmentation of size 512. Therefore, 800 data segmentation are obtained from 100 single channels. The iEEG signals from set C to set E are included in the samples, which are shown in Fig. 1.

2.2 Proposed method

In this paper, three kinds of pattern recognition methods based on PNN classifier (SA-PNN, SA-PPNN and LSA-PPNN) are proposed to identify iEEG signals and detect epileptic seizures. The architecture of the whole process are presented in Fig. 2. First, IEEG signals are processed by using the DWT, and decomposed into multiple kinds of sub-signals by the fifth level decomposition with 'db5' wavelet function. Then, statistical features from sub-signals,

Table 1 The summary of data set of Previous ictal, Post ictal and Epileptic

	SET C	SET D	SET E
sampling rate	173.6HZ	173.6HZ	173.6HZ
No. of epochs	100	100	100
Epoch duration	23.6s	23.6s	23.6s
Electrodes placement	Within epileptogenic zone	Opposite to epileptogenic zone	Within epileptogenic zone

mean absolute value (MAV), root mean square (RMS) and standard deviation (SD), are extracted. The selection of those statistical features refer to the Sharmila et al.'s research [12]. Next, three methods based on PNN, SA-PNN, SA-PPNN, LSA-PPNN, are proposed and deal with classification tasks.

2.3 Extracting features by discrete wavelet transforms (DWT)

2.3.1 Feature extraction

WT has inherited and developed from the idea of short-time Fourier transform (FT) method. Compared with original algorithm FT, WT solves the disadvantages of window size without changing with frequency, and provide a "time-frequency" window which changes with frequency so that extracting more effect information from signals. The characteristic of wavelet transform method is time-frequency localization and multi-scale refinement of signal through scale transformation. Simultaneously, the WT achieves both frequency and time subdivision for all frequencies, so that WT can adaptively complete the requirements for the analysis of different time and frequency [18]. Conclusively, WT is one of the ideal algorithm for signal process analysis, and especially for the irregular signals. The DWT [19] for a signal $s(t)$ is defined as (1):

$$\mathrm{DWT}(m,n)=\int_{-\infty}^{+\infty}\frac{s(t)}{\sqrt{|2^j|}}\psi\left(\frac{t-2^m n}{2^j}\right)dt \qquad (m\in Z) \tag{1}$$

where $\psi(\cdot)$ is a wavelet basis function.

In fact, to realize DWT method, quadrature mirror filters (QMF) can efficient calculate it simply. QMF utilize a series of high−pass (HP) and low−pass (LP) filters. The HP and LP filters are represented by $g[\cdot]$ and $h[\cdot]$ respectively. In QMF, a signal are decomposed by using a series of HP and LP filters [19]. In our algorithm, for the first level decomposition, results D1 and A1 is the output of $s(t)$ by passing $h[n]$ and $g[n]$, among that A1 is dominant in frequency of original signal. The approximation coefficient of each level are supposed to be decomposed continue, and this process are repetitive executed four times to obtain the final sub-signals. The results of five level decomposition with $s(t)$ can be obtained, including D1, D2, D3, D4, D5. They are the frequency content of primary signals within the bands with fs/4−fs/2, fs/8−fs/4, fs/16−fs/8, fs/32−fs/16

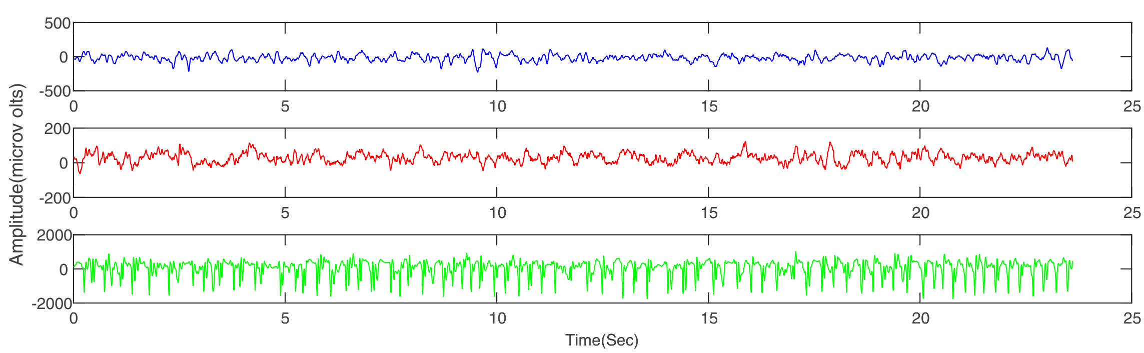

Fig. 1 The records of iEEG signal for C to E. Blue line is an example in set C; red line is an example in set D; green line is an example in set E

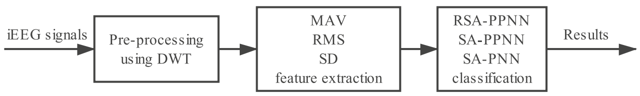

Fig. 2 The architecture of the proposed method

and fs/64−fs/32 respectively, among that the fs is 173.6 HZ [11]. The algorithm structure of five level with wavelet decomposition for iEEG signals is presented in Fig. 3, and the sub-bands corresponds to the frequency range in details are presented in Table 2.

2.3.2 Analysis of variance

Analysis of Variance (ANOVA) usually analyzes test results by mathematical statistics, which is a effective method for identifying the influence of each factor [20]. In our paper, ANOVA is used for 18 kinds of features from 6 sub-bands to confirm the effectiveness of feature extraction method.

2.4 Parallel probabilistic neural network

Probabilistic neural network (PNN) is a kind of radial basis network (RBF) [21]. As the structure of PNN is simple and easily design algorithms, PNN has been widely utilized in pattern classification task [22]. The combinations of different statistical features are utilized as the input vector of PNN. Besides, The number of radial basis layer neurons equals to the number of input vector's training samples.

The input weights are determined by the j_{th} neuron of class i in radial basis layer [7],

$$\psi_{ij}(x) = \frac{1}{\sqrt{2\pi}\sigma^d} e^{-\frac{(x-x_{ij})(x-x_{ij})^T}{\sigma^2}} \quad (i = 1, 2, \ldots, M) \quad (2)$$

where x is the input of radial basis layer; M is the number of classes to be classified; d is the dimension of the sample space; x_{ij} is j_{th} center of the class i sample; σ is a spreading factor, which is key hyper-parameter in our network.

The competitive layer averages the output of radial basis neurons in the same class,

$$v_i = \frac{\sum_{j=1}^{L} \psi_{ij}}{L} \quad (3)$$

where v_j is the output of i_{th} class, and L represents the number of i_{th} class neurons.

The output layer selects the competition layer which has the maximum value.

$$y = \max\{v_1, v_2, \ldots, v_N\} \quad (4)$$

We use one-hot coding method in the competitive layer, so the neuron output is 1 while the neuron has the max posterior probability density, and the others are 0.

In this paper, a parallel PNN (PPNN) classifier is designed for improving computing efficiency. In PPNN method, the calculation of $\varphi_{ij}(x)$ is divided into several parts, and every part is calculated on different processors simultaneously. Therefore, PPNN is more suitable than PNN method for large-scale data processing. The CPU with four processors is used for running PPNN. The data sets are separated into two parts. 70% of data are used to train the network, and 30% are used to test the network's performance. Our results are average by 10 times cross validation. The architecture of this PPNN classifier is illustrated in Fig. 4.

2.5 PPNN combined with simulated annealing algorithm

Due to the largely dependent on the value of σ for the performance of PNN, this paper proposes simulated annealing (SA), and local simulated annealing (LSA) to optimize parameter σ for the PPNN classifier.

The idea of SA algorithm was proposed by Metropolis in 1953. Kirkpatrick applied it to solving optimization problems firstly in 1983 [7]. SA is a stochastic optimization algorithm based on Monte-Carlo iterative solution strategy. The origin of SA is based on the similarity between the annealing process of solid matter in physics and the general combinatorial optimization problem. The core idea of the SA algorithm is to reject the solution of the local minimum problem with a certain probability. In this way, the local optimal solution could be abandoned, and other state solutions of the state space could be explored. The global optimal solution of the problem could be found [7].

In order to make all the solutions acceptable and to avoid the algorithm falling into the local optimal solution, the value of initial temperature T is supposed to be high enough. Here it is set to 100, the maximum number of iterations num_max. The set of states is defined as $S = \{s_1, s_2, s_3, \cdots, s_n\}$. The spreading factor σ is discretized and updated as follows in each iteration,

$$\sigma_{k+1} = \sigma_k - (\epsilon \times 2 - 1) \times 0.02 \quad (5)$$

where $\epsilon \sim U(0, 1)$. The energy function of solutions in time step k is $E(\sigma_k)$. The CA of PNN classifier is calculated with the spreading factor σ_k. The value of CA equals to the value of energy function $E(\sigma_k)$ in our algorithm. The global optimal solution is denoted by σ^* and $E(\sigma^*) = \min\{\sigma_i \,|\sigma_i \in S\}$.

The concrete steps of SA are listed below:

1. Select the initial solution s_0 and set the initial temperature to 100.
2. The states produce random perturbations Δs. The change of energy function is calculated by (6).

$$\Delta E = E(\sigma_{k+1}) - E(\sigma_k) \quad (6)$$

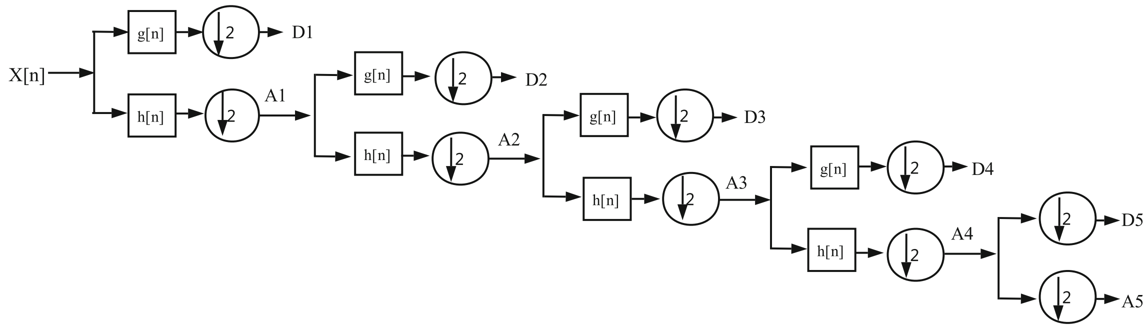

Fig. 3 Fifth level wavelet decomposition of iEEG

3. The probability of $P(\sigma_{k+1} = \sigma_k)$ is calculated by (7)

$$P(\sigma_{k+1} = \sigma_k) = \begin{cases} 1 & \Delta E \leq 0 \\ \exp\left\{-\dfrac{\Delta E}{T}\right\} & \Delta E > 0 \end{cases} \tag{7}$$

4. Choose a noise from a uniform distribution: $\epsilon \sim U(0, 1)$. According to $P(\sigma_{k+1} = \sigma_k)$, σ_{k+1} is selected as follow

$$\sigma_{k+1} = \begin{cases} \sigma_k - (\epsilon \times 2 - 1) \times 0.02 & \epsilon < P(\sigma_{k+1} = \sigma_k) \\ \sigma_k & \text{else} \end{cases} \tag{8}$$

5. Check the stability of the system at temperature T. If the system is unstable, the second step will be executed.
6. Temperature T is declined by $T = \lambda T$, where λ is equal to 0.98. If the termination condition of the algorithm is satisfied ($k \geq num\_max$), the annealing process ends. Otherwise move to step 2.

In order to reduce computational overhead while ensuring classification performance, local simulated annealing (LSA) is proposed to optimize parameter σ for the PPNN classifier. Compared with the previous strategy that uses all the data in data sets for training, LSA samples only a small amount of data from data sets for training and testing. In this work, the data is divided into training data set and test data set. In each iteration, 20% of data is randomly selected as the training data sets. About 50% of the rest data is selected to form test data sets. Partial data is enough to calculate parameters in our model and avoids over fitting problem.

These sub data sets are used to train the model and test the performance of networks. The initial value of σ is set to 0.5. Classification accuracy (CA) of PPNN is used as energy function in LSA. LSA algorithm decides whether to change the value of σ or accept the current value of σ as the result of the algorithm. Then repeat the process until the anneal process is accomplished. After LSA algorithm, the value of σ with the highest CA is selected in PPNN. Then, PPNN is trained with training data sets. At last, the performance of PPNN is tested by using the test data sets. LSA algorithm utilizes local data for training and testing, and obtains the optimal parameter in shorter time.

2.6 Performance evaluation parameters

The performance of the proposed method can be evaluated by the following parameters [23]. Sensitivity, Specificity, Classification Accuracy are mainly used to describe the performance of accuracy, and Running Time and Speedup are implemented to describe performance in terms of time. Speedup represents the radio of the running time with

Table 2 Frequency band of iEEG signals using DWT

Decomposition levels	Sub-bands	Frequency range (HZ)
1	D1	43.4-86.8
2	D2	21.7-43.4
3	D3	10.8-21.7
4	D4	5.4-10.8
5	D5	2.7-5.4
6	A5	0-2.7

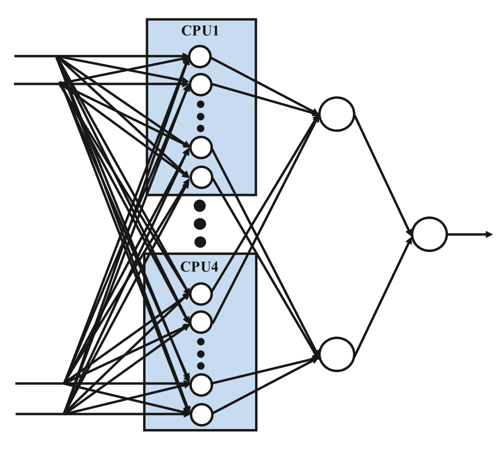

Fig. 4 The architecture of a PPNN

the single computing, and the running time with parallel computing.

3 Results

3.1 The results of feature extraction

The statistical features are extracted the A5, D1, D2, D3, D4, and D5 coefficients by using DWT [24]. According to the results in this table, the value of epileptic is so much higher than that of pre-ictal and post-ictal in the 18 features, while the value of post-ictal is slightly higher than that of pre-ictal. The results also point out that the pre-ictal features could be applied to the prediction of epilepsy [25]. The post-ictal values help researchers realize the clinical significance of post-ictal EEG activity [26]. ANOVA is utilized to 18 features, which can be proved that the features are effective in classification, as p-value ≤ 0.0001 for all combinations [20].

3.2 The result of SA-PNN, SA-PPNN, and LSA-PPNN

In this subsection, PNN based on SA algorithm (SA-PNN), PPNN based on SA algorithm (SA-PPNN), and PPNN based on LSA algorithm (LSA-PPNN), are used to do the five classification tasks with seven different combinations of statistics features including RMS, SD, MAV, SD+RMS, MAV+RMS, MAV+SD, and MAV+SD+RMS.

The results obtained by the SA-PNN classifier are shown in Table 3. For the classification tasks C-D, C-E, D-E, CD-E, and C-D-E, the highest CA values are 83.33%, 99.79%, 99.17%, 98.75% and 85.73%, respectively. They are obtained by different combinations of features (MAV+SD, MAV, MAV+RMS, RMS+SD, and MAV+SD). The range of CA for all the classification tasks using a PNN based on SA algorithm is 83.33% to 99.79%. The shortest running time in different combinations of features for C-D, C-E, D-E, CD-E and C-D-E classification tasks are 2270.21, 2320.80, 2298.54, 5131.98, and 5115.80 seconds, respectively.

The results obtained by the SA-PPNN classifier are shown in Table 4. For C-D, C-E, D-E, CD-E, and C-D-E classification tasks, the highest CA are 83.33%, 99.79%, 99.17%, 98.75%, and 85.73% obtained by the different combinations of features: MAV+SD, MAV, RMS+SD, MAV+RMS, and MAV+SD, respectively. The shortest running time in different combinations of features for C-D, C-E, D-E, CD-E and C-D-E classification tasks are 1106.04, 1104.42, 1159.71, 2151.70, and 2216.21 seconds.

What is different from SA-PNN to SA-PPNN is that the calculation of $\varphi_{ij}(x)$ in SA-PPNN is divided into several parts, and every part is calculated on different processors simultaneously. But the calculation of $\varphi_{ij}(x)$ is assigned to the only processor in SA-PNN. Through Tables 3 and 4, we

Table 3 Performance for the combinations of different features using SA-PNN

Task		MAV	RMS	SD	MAV+SD	RMS+MAV	RMS+SD	MAV+SD+RMS
C-D	Running Time (s)	2280.29	**2270.21**	2300.56	2332.20	2309.27	2304.24	2339.19
	CA(%)	80.42	78.96	79.58	**83.33**	80.42	79.79	81.67
	Sensitivity(%)	77.92	80.42	83.33	85.00	81.25	83.33	83.33
	Specificity(%)	82.92	77.50	75.83	81.67	79.58	76.25	80.00
C-E	Running Time (s)	**2300.59**	2359.24	2324.75	2390.31	2320.80	2310.03	2441.35
	CA(%)	**99.79**	99.38	99.17	99.38	99.58	99.17	99.38
	Sensitivity(%)	100.00	99.58	99.58	99.58	100.00	99.58	100.00
	Specificity(%)	99.58	99.17	98.75	99.17	99.17	98.75	98.75
D-E	Running Time (s)	2561.41	**2298.54**	2311.66	2357.53	2399.30	2453.87	2464.23
	CA(%)	98.33	**98.96**	98.75	98.54	98.75	99.17	98.33
	Sensitivity(%)	98.33	98.75	99.17	98.33	98.17	99.17	97.92
	Specificity(%)	98.33	99.17	98.33	98.75	98.75	99.17	98.75
CD-E	Running Time (s)	**5131.98**	5321.63	5267.33	5162.39	5201.16	5197.45	5245.93
	CA(%)	92.36	98.19	97.92	**98.75**	**98.75**	98.06	98.47
	Sensitivity(%)	98.13	98.33	97.92	98.96	98.96	97.92	98.75
	Specificity(%)	80.83	97.92	97.92	98.33	98.33	98.33	97.92
C-D-E	Running Time (s)	5128.79	5130.177	**5115.80**	5236.11	5152.70	5279.96	5184.32
	CA(%)	82.64	80.28	80.28	85.73	83.19	82.36	83.61

Table 4 Summary of performance for different features combination using SA-PPNN

Task		MAV	RMS	SD	MAV+SD	MAV+RMS	SD+RMS	MAV+RMS+SD
C-D	Running Time (s)	**1106.78**	1155.90	1163.16	1157.14	1261.37	1162.69	1155.22
	CA(%)	80.42	78.96	79.58	**83.33**	80.42	79.79	81.67
	Sensitivity(%)	77.92	80.42	83.33	85.00	81.25	83.33	83.33
	Specificity(%)	82.92	77.50	75.83	81.67	79.58	76.25	80.00
C-E	Running Time (s)	1251.74	1116.06	1139.76	**1104.42**	1241.48	1130.80	1131.32
	CA(%)	**99.79**	99.38	99.17	99.38	99.58	99.17	99.38
	Sensitivity(%)	100.00	99.58	99.58	99.58	100.00	99.58	100.00
	Specificity(%)	99.58	99.17	98.75	99.17	99.17	98.75	98.75
D-E	Running Time (s)	1253.11	1220.53	1272.97	1229.31	1168.38	1281.22	**1159.71**
	CA(%)	98.33	**98.96**	98.75	98.54	98.75	99.17	98.33
	Sensitivity(%)	98.33	98.75	99.17	98.33	98.17	99.17	97.92
	Specificity(%)	98.33	99.17	98.33	98.75	98.75	99.17	98.75
CD-E	Running Time (s)	2151.70	2275.01	2332.29	2318.26	**1168.38**	2416.72	2237.92
	CA(%)	92.36	98.19	97.92	**98.75**	**98.75**	98.06	98.47
	Sensitivity(%)	98.13	98.33	97.92	98.96	98.96	97.92	98.75
	Specificity(%)	80.83	97.92	97.92	98.33	98.33	98.33	97.92
C-D-E	Running Time (s)	2216.21	2244.04	**2121.64**	2298.49	2342.93	2234.8	2289.53
	CA(%)	82.64	80.28	80.28	**85.73**	83.19	82.36	83.61

can observe that the results are samely expect the running time. Since parallel computing is chunking the matrix, the running time of the computing can be shorten.

The results obtained by the LSA-PPNN classifier shown in Table 5. For C-D, C-E, D-E, CD-E, and C-D-E classification tasks, the highest CA are 83.33%, 99.79%, 98.96%, 98.61%, and 84.61% obtained by the different combinations of features: MAV+SD, MAV, RMS, MAV+SD, and MAV+SD, respectively. The shortest running time in different combinations of features for C-D, C-E, D-E, CD-E, and C-D-E classification tasks are 305.61, 303.92, 304.36, 415.23, and 414.18 seconds. The CA of

Table 5 Summary of performance for different features combination using LSA-PPNN

Task		MAV	RMS	SD	MAV+SD	MAV+RMS	SD+RMS	MAV+RMS+SD
C-D	Running Time (s)	308.24	309.37	309.19	308.98	308.37	308.72	**305.61**
	CA(%)	80.21	57.50	75.42	**83.33**	63.33	79.38	81.46
	Sensitivity(%)	77.92	99.58	87.50	85.40	91.25	83.33	84.17
	Specificity(%)	82.50	15.42	63.33	81.25	35.42	75.42	78.75
C-E	Running Time (s)	306.40	306.13	303.30	305.17	303.92	**303.07**	305.64
	CA(%)	**99.79**	98.54	99.17	99.17	99.17	99.17	99.38
	Sensitivity(%)	100.00	99.17	99.58	100.00	100.00	99.58	100.00
	Specificity(%)	99.58	97.92	98.75	98.33	98.33	98.75	98.75
D-E	Running Time (s)	305.42	306.96	308.37	306.72	306.53	306.96	**304.36**
	CA(%)	97.29	**98.54**	98.33	97.92	98.33	**98.54**	98.13
	Sensitivity(%)	99.58	99.17	98.33	97.08	98.33	99.17	97.92
	Specificity(%)	95.00	97.92	98.33	98.75	98.33	97.92	98.33
CD-E	Running Time (s)	415.39	420.59	**415.23**	448.61	423.88	417.93	422.78
	CA(%)	98.19	98.19	97.92	**98.61**	97.78	97.78	98.47
	Sensitivity(%)	99.17	98.13	97.92	99.17	99.17	97.92	99.17
	Specificity(%)	97.92	98.33	97.92	97.92	97.92	97.50	97.92
C-D-E	Running Time (s)	442.12	**417.72**	568.84	451.76	418.84	424.64	417.78
	CA(%)	81.67	79.17	77.78	**84.61**	82.22	82.22	83.61

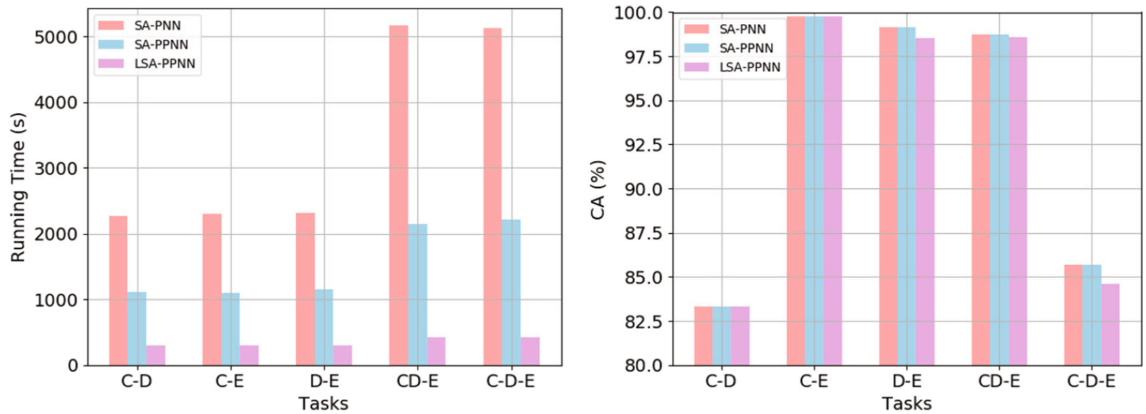

Fig. 5 The best performance achieved by SA-PNN, SA-PPNN, and LSA-PPNN. Left figure is the running time for different methods and right figure is CA

classification results obtained by LSA-PPNN classifier is similar to that obtained by the SA-PPNN classifier. However, the running time is much shorter than the SA-PNN and the SA-PPNN classifiers. The best performance achieved by SA-PNN, SA-PPNN, and LSA-PPNN is shown in Table 7 (see Appendix) and Fig. 5. The average CA of SA-PNN, SA-PPNN and LSA-PPNN is quite similar (93.35%, 93.35%, and 93.06%, respectively). LSA-PPNN uses the idea of training and optimizing parameters with local data firstly, and then brings the parameters into the global training data sets to train the network for testing. Therefore, the running time will be greatly reduced.

The speedup between SA-PNN and SA-PPNN, SA-PNN and LSA-PPNN is shown in Table 8 (see Appendix) and Fig. 6. For five classification tasks, the highest speedups between SA-PNN and SA-PPNN, are 2.06, 2.16, 2.12, 2.39, and 2.41 respectively. The highest speedups between SA-PNN and LSA-PPNN, are 7.65, 7.99, 8.39, 12.51, and 12.41 respectively. The CPU with 4 processors is used for running the procedures of SA-PPNN. However, in parallel computing, processors need to pass messages to each other, and it requires consumption. Obviously, it is impossible that the running time is reduced by many times as many processors. In general, for the same type of calculation, the speedup will increase as the data increases, because the ratio of the spending time of processors' communication and the reduced time of parallel computing becomes smaller and smaller. As a result, the speedups of tasks with 3 data sets (CD-E, C-D-E) are higher than tasks with 2 data sets (C-D, C-E, D-E).

3.3 Discussion

We compare the performance of our methods with other previous methods [2, 7, 12, 13] and [27–29]. The results obtained from the proposed methods and others on the iEEG

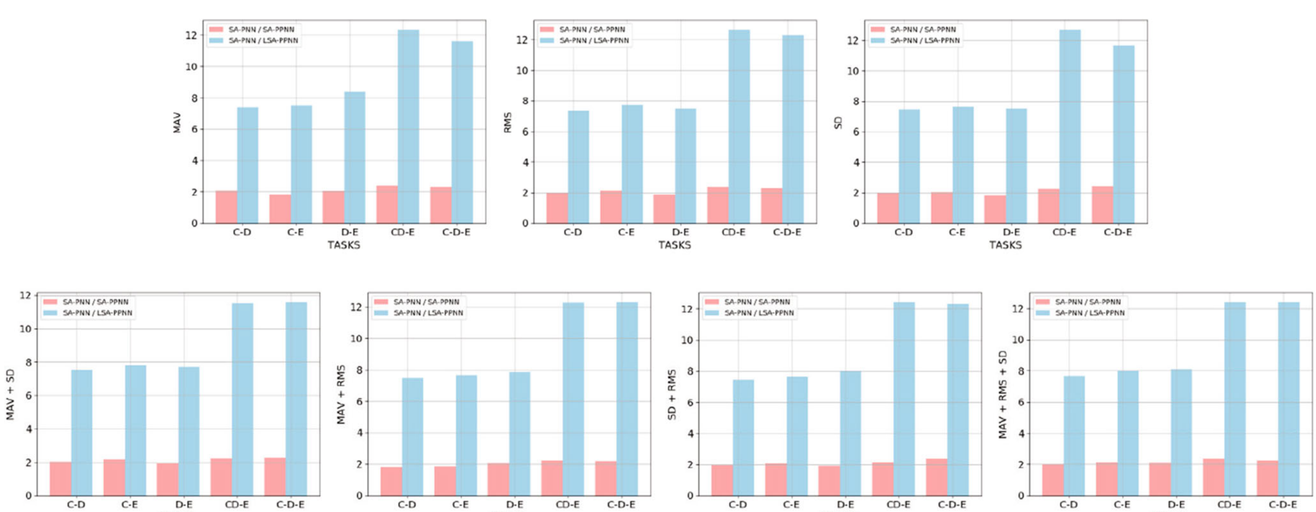

Fig. 6 The speedup between SA-PNN and SA-PPNN, SA-PNN and LSA-PPNN for different classification tasks

Table 6 The comparison results of the proposed method and other previous algorithms

Year	Method	Task	CA(%)	References
2012	Permutation entropy combining with SVM	C-E	88.00	[13]
		D-E	79.94	
2012	DWT combined with ApEn, SVM as classifier	C-E	99.00	[7]
		D-E	95.94	
2012	DWT combined with Relative wavelet energy (RWE) and Wavelet entropy (WE), SVM for classification	C-E	97.50	[7]
		D-E	97.50	
2014	DWT based on ApEn neural network for classification	C-E	98.00	[11]
		D-E	94.00	
2014	One-dimensional local binary pattern	CD-E	97.67	
2015	Empirical mode decomposition (EMD) and LS-SVM	CD-E	98.67	[29]
2016	DWT-Naive Bayes classifer combined with k-NN classifiers	C-E	99.62	[12]
		D-E	95.62	
		CD-E	98.75	
2016	Wavelet-based nonlinear analysis and SVM	CD-E	85.00	[28]
2017	Log energy, Shannon, Renyi entropy and Spectral using MLP for classification	C-E	97.68	[2]
		D-E	94.56	
		C-D-E	84.58	
		C-D	57.80	
2019	DWT extract features using elman neural network models	C-E	99.37	[24]
		D-E	98.25	
		CD-E	97.50	
		C-D	71.25	
		C-D-E	73.25	
2020	DWT extract features using LSTM	C-E	**100.00**	[30]
2020	DWT extract features using Simulated Annealing-Based Parallel Probabilistic Neural Network (SA-PPNN)	C-E	99.79	
		D-E	**99.17**	
		CD-E	**98.75**	
		C-D	**83.33**	
		C-D-E	**85.73**	
2020	DWT extract features using Rondom Simulated Annealing-Based P-arallel Probabilistic Neural Network (LSA-PPNN)	C-E	99.79	
		D-E	98.54	
		CD-E	98.61	
		C-D	**83.33**	
		C-D-E	84.61	

signal classification tasks are presented in Table 6. It can be observed that the SA-PPNN obtains the best CA for all classification tasks. The CA of LSA-PPNN for tasks C-E, D-E, C-D and C-D-E is also higher than existing methods. However, the CA of LSA-PPNN for task CD-E is 0.14% lower than A. Sharmila's work [12], and the CA of LSA-PPNN for task C-E is 0.21% lower than the research in [30]. There are some methods using deep learning as classifiers, which can achieve accuracies of 100% in the task of detecting epileptic seizures [30]. In this paper, we studied pattern recognition of iEEG signals for pre-ictal, post-ictal, and epileptic, not only detection of epileptic seizures. For the classification of detection of epileptic seizures (C-E), we achieved accuracy of 99.79%, which is a little lower than that achieved by some deep learning methods, but the structure of our method is much simpler than deep learning methods.

In this study, each iEEG signal is decomposed into six sub-signals by using the DWT, and statistics features are extracted from these sub-signals. The ANOVA analysis indicates that features selected for the experiment distinguish between the classes very well. That is, effective features are obtained, paving the path for classification. PNN is selected as the classifier because it has only one key parameter, spreading factor. It is much easier to adjust parameter than most other machine learning algorithms [22]. A LSA algorithm is proposed to optimize spreading factor, which helps obtain good results with local data. Finally, the complex matrix computation in PNN is realized by parallel setting (PPNN), which greatly improves the running speed of our algorithm. Our algorithm is competitive comparing with other methods.

4 Conclusions

In order to improve the classification accuracy of iEEG signals and reduce the running time, three probabilistic neural network classifier, SA-PNN, SA-PPNN, and LSA-PPNN, are proposed with different feature combinations as input vectors. Each of these PNN classifiers has only one parameter, whose value can be obtained automatically by simulated annealing algorithm.

Experimental results show that parallel mechanism has positive impact on reducing the running time. LSA-PPNN algorithm trains and optimizes parameters with local data, but achieves a classification effect no less than SA-PPNN. The running time of LSA-PPNN is much shorter than SA-PPNN. As the amount of data increases, the superiority over LSA-PPNN is getting more obvious.

Comparing with other existing methods, the SA-PPNN obtains the best CA for all classification tasks. The CA of LSA-PPNN for four tasks C-E, D-E, C-D, and C-D-E is also higher than existing methods. Especially for classification task C-D, LSA-PPNN gets 83.3% CA, which is much higher than previous methods. As a result, the LSA-PPNN is expected to be an economical and effective algorithm in the practical application of epilepsy medical diagnosis.

Acknowledgements This work was supported by the National Natural Science Foundation of China [grant numbers 61872325]; the Fundamental Research Funds for the Central Universities [grant number 2652019028].

Appendix

The best performance achieved by SA-PNN, SA-PPNN, and LSA-PPNN is shown in Table 7. The average CA of SA-PNN, SA-PPNN, and LSA-PPNN is quite similar (93.35%, 93.35% and 93.06%, respectively). The speedup between SA-PNN and SA-PPNN, SA-PNN, and LSA-PPNN is shown in Table 8. For five classification tasks, the highest speedups between SA-PNN and SA-PPNN, are 2.06, 2.16, 2.12, 2.39, and 2.41 respectively. The highest speedups between SA-PNN and LSA-PPNN, are 7.65, 7.99, 8.39, 12.51, and 12.41 respectively.

Table 7 The best performance achieved by SA-PNN, SA-PPNN and LSA-PPNN

Task		SA-PNN	SA-PPNN	LSA-PPNN
C-D	Running Time (s)	2270.21	1106.78	**304.36**
	CA(%)	83.33	83.33	**83.33**
C-E	Running Time (s)	2300.59	1104.42	**303.07**
	CA(%)	99.79	99.79	**99.79**
D-E	Running Time (s)	2311.66	1159.71	**304.36**
	CA(%)	99.17	**99.17**	98.54
CD-E	Running Time (s)	5162.39	2151.70	**417.93**
	CA(%)	98.75	**98.75**	98.61
C-D-E	Running Time (s)	5130.17	2216.21	**417.72**
	CA(%)	85.7	**85.7**	84.61

Table 8 Speedup between SA-PNN and SA-PPNN, SA-PNN and LSA-PPNN

Task	Speedup	MAV	RMS	SD	MAV+SD	MAV+RMS	SD+RMS	MAV+RMS+SD
C-D	SA-PNN / SA-PPNN	**2.06**	1.96	1.98	2.02	1.83	1.98	2.02
	SA-PNN / LSA-PPNN	7.40	7.34	7.44	7.55	7.49	7.46	**7.65**
C-E	SA-PNN / SA-PPNN	1.84	2.11	2.04	2.16	1.87	2.04	**2.16**
	SA-PNN / LSA-PPNN	7.51	7.71	7.66	7.83	7.64	7.62	**7.99**
D-E	SA-PNN / SA-PPNN	2.04	1.88	1.82	1.92	2.05	1.92	**2.12**
	SA-PNN / LSA-PPNN	**8.39**	7.49	7.50	7.69	7.83	7.99	8.10
CD-E	SA-PNN / SA-PPNN	**2.39**	2.34	2.26	2.23	2.23	2.15	2.34
	SA-PNN / LSA-PPNN	12.35	12.65	**12.69**	11.51	12.27	12.44	12.41
C-D-E	SA-PNN / SA-PPNN	2.31	2.29	2.41	2.28	2.20	**2.36**	2.26
	SA-PNN / LSA-PPNN	11.60	12.28	11.66	11.59	12.30	12.33	**12.41**

References

1. (2014) Health communities. http://www.healthcommunities.com/epilepsy-seizures/seizures-phases.html
2. Raghu S, Sriraam N (2017) Optimal configuration of multilayer perceptron neural network classifier for recognition of intracranial epileptic seizures. Expert Syst Appl 89:205–221
3. Wilson SB, Scheuer ML, Plummer C, Young B, Pacia S (2003) Seizure detection: correlation of human experts. Clin Neurophysiol 114(11):2156–2164
4. Guo L, Rivero D, Pazos A (2010) Epileptic seizure detection using multiwavelet transform based approximate entropy and artificial neural networks. J Neurosci Methods 193(1):156–163
5. Srinivasan V, Eswaran C, Sriraam N (2007) Approximate entropy-based epileptic eeg detection using artificial neural networks. IEEE Trans Inf Technol Biomed 11(3):288–295
6. Acharya UR, Molinari F, Sree SV, Chattopadhyay S, Ng K-H, Suri JS (2012) Automated diagnosis of epileptic eeg using entropies. Biomed Signal Process Control 7(4):401–408
7. Kumar Y, Dewal ML, Anand RS (2012) Relative wavelet energy and wavelet entropy based epileptic brain signals classification. Biomed Eng Lett 2(3):147–157
8. Polat K, Güneş S (2008) Artificial immune recognition system with fuzzy resource allocation mechanism classifier, principal component analysis and fft method based new hybrid automated identification system for classification of eeg signals. Expert Syst Appl 34(3):2039–2048
9. Mousavi SR, Niknazar M, Vahdat BV (2008) Epileptic seizure detection using ar model on eeg signals. In: 2008 Cairo International Biomedical Engineering Conference. IEEE, pp 1–4
10. Zhan Y, Halliday D, Jiang P, Liu X, Feng J (2006) Detecting time-dependent coherence between non-stationary electrophysiological signals—a combined statistical and time–frequency approach. J Neurosci Methods 156(1-2):322–332
11. Kumar Y, Dewal ML, Anand RS (2014) Epileptic seizures detection in eeg using dwt-based apen and artificial neural network. SIViP 8(7):1323–1334
12. Sharmila A, Geethanjali P (2016) Dwt based detection of epileptic seizure from eeg signals using naive bayes and k-nn classifiers. Ieee Access 4:7716–7727
13. Nicolaou N, Georgiou J (2012) Detection of epileptic electroencephalogram based on permutation entropy and support vector machines. Expert Syst Appl 39(1):202–209
14. Tang Y, Durand DM (2012) A tunable support vector machine assembly classifier for epileptic seizure detection. Expert Syst Appl 39(4):3925–3938
15. Naseer N, Hong K-S (2015) fnirs-based brain-computer interfaces: a review. Front Human Neurosci 9:3
16. Orhan U, Hekim M, Ozer M (2011) Eeg signals classification using the k-means clustering and a multilayer perceptron neural network model. Expert Syst Appl 38(10):13475–13481
17. Andrzejak RG, Lehnertz K, Mormann F, Rieke C, David P, Elger CE (2001) Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state. Phys Rev E 64(6):061907
18. Adeli H, Zhou Z, Dadmehr N (2003) Analysis of eeg records in an epileptic patient using wavelet transform. J Neurosci Methods 123(1):69–87
19. Mallat SG (1989) A theory for multiresolution signal decomposition: the wavelet representation. IEEE Transa Pattern Anal Mach Intell 11(7):674–693
20. Webster R, Lark RM (2018) Analysis of variance in soil research: let the analysis fit the design. Eur J Soil Sci 69(1):126–139
21. Specht DF (1990) Probabilistic neural networks. Neural Netw 3(1):109–118
22. Huang C-J, Liao W-C (2003) A comparative study of feature selection methods for probabilistic neural networks in cancer classification. In: Proceedings. 15th IEEE International Conference on Tools with Artificial Intelligence. IEEE, pp 451–458
23. Pang CCC, Upton ARM, Shine G, Kamath MV (2003) A comparison of algorithms for detection of spikes in the electroencephalogram. IEEE Trans Biomed Eng 50(4):521–526
24. Gong C, Zhang X, Niu Y (2020) Identification of epilepsy from intracranial eeg signals by using different neural network models. Comput Biol Chem:107310
25. Besag FMC, Vasey MJ (2018) Prodrome in epilepsy. Epilepsy Behav 83:219–233
26. Tomlinson SB, Khambhati AN, Bermudez C, Kamens RM, Heuer GG, Porter BE, Marsh ED (2018) Alterations of network synchrony after epileptic seizures: An analysis of post-ictal intracranial recordings in pediatric epilepsy patients. Epilepsy Res 143:41–49
27. Leung StephenCH, Zheng J, Zhang D, Zhou X (2010) Simulated annealing for the vehicle routing problem with two-dimensional loading constraints. Flexible Serv Manuf J 22(1-2):61–82

28. Li M, Chen W, Zhang T (2016) Automatic epilepsy detection using wavelet-based nonlinear analysis and optimized svm. Biocybern Biomed Eng 36(4):708–718
29. Sharma R, Pachori RB (2015) Classification of epileptic seizures in eeg signals based on phase space representation of intrinsic mode functions. Expert Syst Appl 42(3):1106–1117
30. Shoeibi A, Ghassemi N, Khodatars M, Jafari M, Hussain S, Alizadehsani R, Moridian P, Khosravi A, Hosseini-Nejad H, Rouhani M et al (2020) Epileptic seizure detection using deep learning techniques: A review. arXiv:2007.01276

Chen Gong received the B.Sc. degree from China University of Geosciences, Beijing, China, in 2020. He is a master student at the Institute of Automation, Chinese Academy of Sciences. His current research interests include mainly medical intelligence and reinforcement learning.

Xingchen Zhou received the B.E. degree in computer science and technology at China University of Geosciences (Beijing) in 2020. He now is a master student at school of computer science and engineering, Southeast University. His research interests include knowledge graph and natural language processing.

Yunyun Niu received the Ph.D. degree in systems analysis and integration from the Huazhong University of Science and Technology, Wuhan, China, in 2012. She is currently an Associate Professor with the School of Information Engineering, China University of Geosciences, Beijing, China. Her current research interests include bioinformatics and artificial intelligence.