

DETECTION OF PARKINSON DISEASE USING COMPUTATIONAL INTELLIGENCE METHODS

Key words: *Parkinson's Disease, Artificial Neural Network, Differential Evolution Optimization, Computational Intelligence*

Açar sözlər: *Parkinson Xəstəliyi, Süni Sinir Şəbəkəsi, Diferensial Təkamül Optimallaşdırılması, Hesablama Zəkası*

Ключевые слова: *болезнь Паркинсона, искусственная нейронная сеть, оптимизация дифференциальной эволюции, вычислительный интеллект*

Introduction

Parkinson's disease is a neuro-degenerative movement disorder that causes voice/speech, and behavioral impairments. The disease causes partial or full loss in motor reflexes, speech, behavior, mental processing, and other vital functions [1]. The early detection of disease symptoms is vitally important in order to prevent further disease complications. Using recorded data including voice/speech and physical behavior/movement characteristics from healthy and sick people it is possible to create models, which would allow fast noninvasive diagnostic of the disease.

Appropriate models include Support Vector Machines, Rule Based Systems, Artificial Neural Networks and others. Most existing approaches utilize only voice/speech data [2]. In our research to improve detection accuracy, we use also physical behavior/movement characteristics obtained from different subjects.

Among all possible methods to create required model, we have chosen multi-layer deep feed-forward neural networks for a number of reasons. First, because they are indeed universal approximators and can be used to reveal any complex relationships in large data sets. Second, because recent developments in the theory and technology have significantly increased efficiency of neural networks. For instance, increased processing power and parallel processing abilities of modern computers allow efficient use of new evolutionary training approaches to effectively battle such bottleneck of large multi-layer neural networks as time-consuming parameter adaptation. The global parameter search, which avoids local minima trapping, is now much faster than ever. Third, because, neuron models are not now required to be constrained by smooth differentiable transfer functions, connection weights by simple numerical values and network architecture for large input/output systems by single hidden layer of neurons.

Method

The used detection model is multi-layer feed-forward neural network with non-linear transfer function based neurons in hidden layers and linear neurons in input and output layers.

Given particular values for the neural network parameters, and given values for the inputs, a neural network generates a value for each output:

$$y_i = NN_w \mathbf{x} ,$$

The operation of an L -layer feed-forward perceptron neural network at each layer $l(l=1, \dots, L-1)$ can be described by the following equation:

$$y_i^{(l)} = f^{(l)} \left(\left(\sum w_{i,j}^{(l)} x_j^{(l)} \right) + \theta_i^{(l)} \right),$$

Where $f^{(l)}(\cdot)$ is the activation function used at network layer l .

In the vector form this can be written more compactly:

$$\mathbf{y}^{(l)} = f^{(l)} \mathbf{w}^{(l)} \mathbf{x}^{(l)}$$

Alternatively, based on only the network activations as:

$$\mathbf{y}^{(l)} = f^{(l)} \mathbf{w}^{(l)} \mathbf{y}^{(l-1)}$$

Matrix $w^{(l)}$ will denote weights connecting all neurons of layer l with all neurons of layer $(l+1)$. Thus for an L -layered NN set W will contain matrixes $w^{(0)}, w^{(1)}, \dots, w^{(L-1)}$.

$w_{i,j}^{(l)}$ is the weight of connection to neuron i at layer l from neuron j at the previous layer $(l-1)$, $\theta_i^{(l)}$ is the threshold parameter of neuron i at layer l

The total number of connection weights and thresholds (i.e. number of elements in the set W) for a feed-forward neural network is

$$N = n_0 + 1 n_1 + n_1 + 1 n_2 + \dots + (n_{L-2} + 1)n_{L-1}.$$

The evolutionary algorithm used for training is Differential Evolution [5], which is one of the fastest population based algorithms for global search in multi-dimensional vector space.

The DE algorithm in a basic form can be described in Figure 1.

```

Step 0. Initialize DE
    Step 0.1. Set algorithm parameters:  $f$  (mutation rate),  $cr$  (crossover rate),
              and  $ps$  (size of population)
    Step 0.2. Define the cost function as function of error function of current
              NN parameters
Step 1. Randomly generate  $ps$  vectors (potential network parameter sets) from
        respective parameter spaces (e.g. in the range  $[-1, 1]$ ) and form a
        population  $P = \{X_1, X_2, \dots, X_{ps}\}$ 
Step 2. While Termination condition is not met generate new parameter sets:
    Step 2.1. Choose a next vector  $X_i$  ( $i=1, \dots, ps$ )
        Step 2.2. Choose randomly different 3 vectors from  $P$ :  $X_{r1}, X_{r2}, X_{r3}$ 
                  each of which is different from current  $X_i$ 
        Step 2.3. Generate trial vector  $X_t = X_{r1} + r'(X_{r2} - X_{r3})$ 
        Step 2.4. Generate new vector from trial vector  $X_t$ . Individual vector
                  parameters of  $X_i$  are inherited with probability  $cr$  into the
                  new vector  $X_{new}$ . If the cost function from  $X_{new}$  is better
                  (lower) than the cost function from  $X_i$ , current  $X_i$  is
                  replaced in population  $P$  by  $X_{new}$ 
    Next  $i$ 
Step 3. Select the vector (NN parameter set) with best cost (training error  $E$ )
        function from population  $P$ 

```

Figure 1. DE Algorithm for training NNs

The suggested Parkinson Disease Detection System is based on two neural network based models:

1. Detection on the basis of patient's behavioral characteristics
2. Detection on the basis of patient's voice features

The results from both systems are combined to provide a quality detection of the disease.

The data to train the systems and test the system's performance are taken from real patients suffering the decease and taking treatment [2].

Input Data for Model 1 accepts the following patient characteristics as its inputs:

- memory
- hallucinations
- mood
- motivation

- speech
- saliva
- swallowing
- handwriting
- cutting_food
- dressing
- hygiene
- turning_in_bed
- falling
- freezing
- walking
- tremors
- numbness

Output Data for Model 1 is the variable evaluating the severity (degree) of the disease:

- Hoehn and Yahr stage (hoehn_yahr)

Table 1 presents a Fragment of data used for training the neural network based Model 1:

memory	hallucinat	mood	motivatio	speech	saliva	swallowir	handwriti	cutting_fd	dressing	hygiene	turning_in_bed	...	hoehn_ya
2	2	4	4	2	0	0	2	1	1	1	1		3
1	1	1	0	2	1	1	1	1	1	1	2		3
0	2	0	1	2	2	1	3	2	2	1	2		3
2	1	2	4	1	0	1	2	1	1	1	1		2
1	0	1	2	0	0	0	1	1	1	1	1		3
0	0	0	1	2	1	0	1	1	1	1	1		3
1	1	0	0	1	0	1	1	0	1	0	1		1
0	1	0	1	1	1	0	1	0	1	1	1		2
0	1	1	0	0	0	1	0	1	1	1	1		1
1	0	0	3	2	2	2	1	0	0	0	0		2
...													

Table 1. A Fragment of data used for training Model 1 NN.

1. Results

Several architectures for Model 1 and Model 2 networks have been used for experimenting.

The parameters for a successful Model 1 network are as follows:

Model 1

4 Layered Feed-Forward NN with non-linear hidden and linear input and output layer neurons

Number of inputs (Layer 1): 19

Number of hidden neurons (Layer 2): 8

Number of hidden neurons (Layer 3): 6

Number of outputs (Layer 4): 1

Number of training/validation input-output data pairs: 300/50

Number of test input-output data pairs: 39

Training based on Differential Evolution algorithm

About 40 training experiments have been performed.

Best network RMSE after training (on training/validation data) obtained after 8000 iterations (generations of DE): 0.55 (Figure 2)

RMSE of trained network on test data was 0.72.

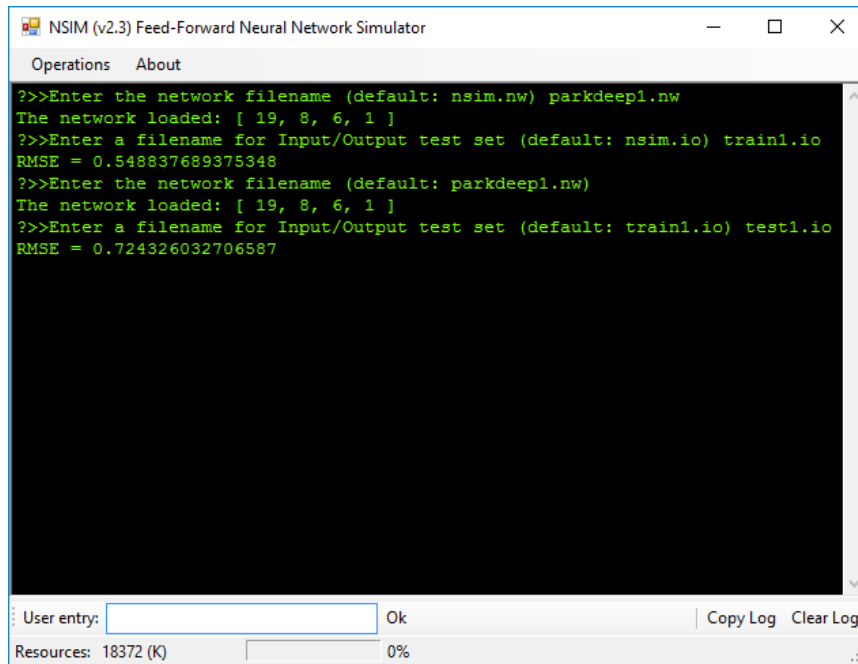


Figure 2. Result of best Model 1 NN training by DE

The parameters for a successful Model 2 network are as follows:

Model 2

3 Layered Feed-Forward NN with non-linear hidden and linear input and output layer neurons

Number of inputs (Layer 1): 38

Number of hidden neurons (Layer 2): 16

Number of outputs (Layer 3): 1

Input Data are 38 measured voice features.

Output is Hoehn and Yahr stage (hoehn_yahr).

Number of training/validation input-output data pairs: 300/40

Number of test input-output data pairs: 34

Training based on Differential Evolution algorithm

RMSE after training (on training/validation data) was 0.92 and RMSE of trained network on test data was 0.93 (Figure 3).

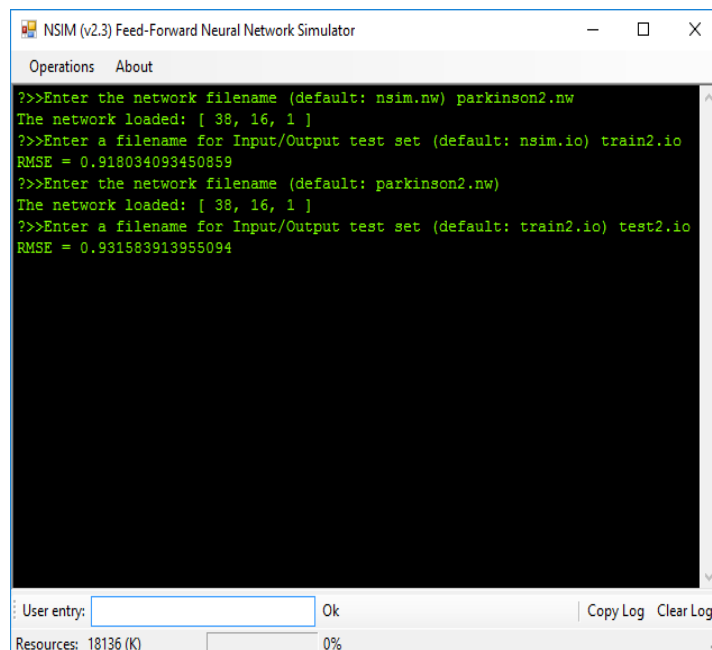


Figure 3. Result of best Model 2 NN training by DE

In the future work we intend to apply other computational intelligence methods as well including Neuro-Fuzzy IS, Type-2 FIS, and Z-FIS to detect Parkinson Disease in patients on the basis of a number of symptoms.

Conclusion

Deep feed-forward neural network based approach is used for detection of Parkinson disease. Two network models trained on separate sets of real patients' characteristics are used for better detection. The networks are trained with Differential Evolution training algorithm on a pair of data sets (training and validation sets) to avoid overfitting and improve model's generalization ability. Results of the experiments has proven the efficiency of the suggested method over the existing approaches.

References

1. Betul Erdogan Sakar, M. Erdem Isenkul, C. Okan Sakar, Ahmet Sertbas, Fikret Gurgen, Sakir Delil, Hulya Apaydin, and Olcay Kursun. Collection and Analysis of a Parkinson Speech Dataset With Multiple Types of Sound Recordings.
2. Ogut F., Yavuzere, Uygur M., Akyurekli O and Kalaycı T., "Evaluation Criteria of Voice Change in Parkinson's Disease", Ege Tıp Dergisi, vol. 39, no. 1, 2000.
3. Little M. A., McSharry P. E., Roberts S. J., Costello D. A. and Moroz I. M., "Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection", Biomed. Eng. Online., Skin. 6, p. 23, 2007.
4. LeCun Y., Bengio Y. and Hinton G., "Deep Learning", Nature, vol.521, pp. 436-444, 2015.
5. Price, K., Storm, R., Lampinen J.: Differential evolution – a practical approach to global optimization. Springer, Berlin (2005).

Hesablama Zəkası Metodlarından istifadə edərək Parkinson xəstəliyinin aşkarlanması

Xülasə

Parkinson xəstəliyi səs / danışma və davranış pozuntularına səbəb olan bir nörodegenerativ hərəkət pozuntusudur. Bir disfunksiya xəstəliyi olaraq, xəstələrin müəyyən simptomları ilə müəyyən edilə bilər. Belə simptomlar arasında həm səs, həm danışma, həm də fiziki davranış / hərəkət xüsusiyyətləri vardır. Daha yaxşı aşkarlanması üçün tədqiqatımızda hər iki xüsusiyyət dəsti istifadə olunur. Bu araşdırmada, diaqnostik bir model olaraq çox qatlı (dərindən) qidalandırıcı sinir şəbəkələrinə əsaslanan bir sistem istifadə edirik. Şəbəkələr, həddindən artıq uyğunlaşmamaq və modelin ümumiləşdirmə qabiliyyətini (öyrətməmiş məlumatlar üzərində işləmə) yaxşılaşdırmaq üçün paralel bir cüt məlumat dəsti (təlim və qiymətləndirmə dəsti) istifadə edərək Diferensial Evolution təlim alqoritmləri ilə öyrədilir. Tətbiq olunan DE alqoritmi təlim əsnasında səhv funksiyasının yerli minimumundan yayınmağa imkan verdi. Üçüncü bir məlumat dəsti təlim keçmiş şəbəkə performansını sınaq üçün istifadə olunur. Əldə olunan nəticələrə görə, bu üsul digər mövcud yanaşmalardan daha yaxşı nəticələr göstərdi.

Обнаружение болезни Паркинсона с использованием методов вычислительного интеллекта

Резюме

Болезнь Паркинсона является нейродегенеративным двигательным расстройством, которое вызывает голос / речь и поведенческие нарушения. Как дисфункциональное заболевание, оно может быть выявлено рядом специфических симптомов пациентов. Такие симптомы включают в себя характеристики голоса / речи и / или физического поведения / движения. Для лучшего обнаружения оба набора характеристик используются в нашем исследовании. В этом исследовании в качестве диагностической модели мы используем систему, основанную на многослойных (глубоких) нейронных сетях с прямой связью. Сети обучаются с помощью алгоритмов обучения дифференциальной эволюции, использующих параллельно пару наборов данных (наборы обучения и проверки), чтобы избежать переобучения и улучшить способность модели к обобщению (производительность на необученных данных). Примененный алгоритм DE позволил избежать локальных минимумов функции ошибок во время обучения. Третий набор данных используется для тестирования производительности обученной сети. Согласно полученным результатам, этот метод показал лучшие результаты, чем другие существующие подходы.