International Journal of Research in Engineering, Science and Management
Volume-1, Issue-7, July 2018
www.ijresm.com
ISSN (Online): 2581-5782

Analysis and Removal of Interference in Cognitive Radio Using RBFN

Charu Gupta[1], Prabhdeep Singh[2]

[1]*Student, Department of Electronics and Communications, RPIIT, Bastara, India*
[2]*Associate Professor, Department of Electronics and Communications, RPIIT, Bastara, India*

***Abstract*: Spectrum prediction is a key technology of cognitive radio, which can help unlicensed users to determine whether the licensed user's spectrum is idle. Based on radial-basis function (RBF) neural network, this paper proposed a spectrum prediction algorithm with K-means clustering algorithm (K-RBF). This algorithm could predict the spectrum holes according to the historical information of the licensed user's spectrum. It not only increases the veracity of spectrum sensing, but also improves the efficiency of spectrum sensing. Simulation results showed that this prediction algorithm can predict the spectrum accessing of the licensed user accurately and the prediction error is only one-third of that of the RBF neural network.**

Key Words: Cognitive Radio, Spectrum Prediction, K-means Clustering, RBF Neural Network.

1. Introduction

The electromagnetic spectrum has been exclusively allocated to different wireless services by government, although some of the frequency bands in the spectrum are unoccupied most of the time or only partially occupied. How to maximize the use of the existing spectrum resources is an urgent problem to be solved. Cognitive radio (CR) is a kind of intelligent spectrum sharing technology, which can rely on artificial intelligence support to adjust the transmission parameters (such as transmission power, data rate, carrier frequency, etc.). CR can effectively use idle spectrum and greatly reduce the restriction to the development of wireless technology by the spectrum and the limitation of bandwidth.

CR network is composed of two parts of the users – licensed user (also known as primary user) and unlicensed user (also known as second user). In each time slot, the unlicensed user must perceive the short-term activities of the licensed user and access slot when it is idle (the idle time slot is also known as spectrum holes). To minimize the interference to licensed users, unlicensed users need a reliable spectrum sensing mechanism. Spectrum prediction is important to effective spectrum of CR network and has become a hot topic in CR. A prediction model using sliding window was established to predict licensed users' future spectrum activity. This model sets a threshold value through the adaptive filter. The frequency band which is lower than the threshold will be set to be unreliable and do not allow unlicensed users to access this band. A multilayer perceptron for spectrum prediction was proposed. However, the multilayer perceptron uses traditional unconstrained minimization method to achieve minimization of the error function. Therefore, it inevitably has local minima problem. Subsequent studies have proposed ON-OFF, Blackman window, POMDP, and other prediction mechanisms (Federal Communications Commission, 2002); (Acharya et al., 2006); (Jianli et al., 2011). In (Marko et al., 2008), a dynamic spectrum access algorithm based on probably density estimation was proposed to predict channel state with flexibility and availability.

However, when the CR node sensing the spectrum, it will detect the whole spectrum concerned every time. It will consume a lot of network resources. We addressed the problem in this paper and proposed a spectrum prediction algorithm using radial-basis function (RBF) neural network based on K-means clustering algorithm (K-RBF). In the algorithm, the spectrum holes are predicted according to the licensed users's historical information. Then, appropriate spectrum bands are chosen for the unlicensed users to detect. It can greatly reduce the resources consumed in spectrum sensing.

The rest of the paper is organized as follows. In Section II, we present the system model of spectrum prediction with RBF neural network. In Section III, we propose the improved prediction algorithm of spectrum. In Section IV, we provide the simulations for the improved prediction algorithm and demonstrate the effect of spectrum prediction. Finally, Section V concludes this paper.

Whether the licenced user's spectrum is idle can be modeled as a binary series prediction problem. We design the binary series predictor using neural networks. Neural network are nonlinear parametric models which create a mapping finction between the input and output data.

The most basic form of RBF neural network is a three layer forward network, which includes the input layer, the hidden layer, and the output layer. The input layer has some source nodes connecting to the external environment, while hidden layer has a variable no. of neurons. The neurons in the hidden layer contain Gaussian transfer function whose outputs are inversely proportional to the distance from the centre of the neuron. The output layer produces response to the input nodes.

2. Radial Basis Function

A Radial Basis Function (RBF) neural network has an input layer, a hidden layer and an output layer. The neurons in the hidden layer contain Gaussian transfer functions whose outputs are inversely proportional to the distance from the center of the neuron. RBF networks are similar to K-Means clustering and PNN/GRNN networks. The main difference is that PNN/GRNN networks have one neuron for each point in the training file, whereas RBF networks have a variable number of neurons that is usually much less than the number of training points. For problems with small to medium size training sets, PNN/GRNN networks are usually more accurate than RBF networks, but PNN/GRNN networks are impractical for large training sets.

How RBF networks work:

Although the implementation is very different, RBF neural networks are conceptually similar to K-Nearest Neighbour (k-NN) models. The basic idea is that a predicted target value of an item is likely to be about the same as other items that have close values of the predictor variables.

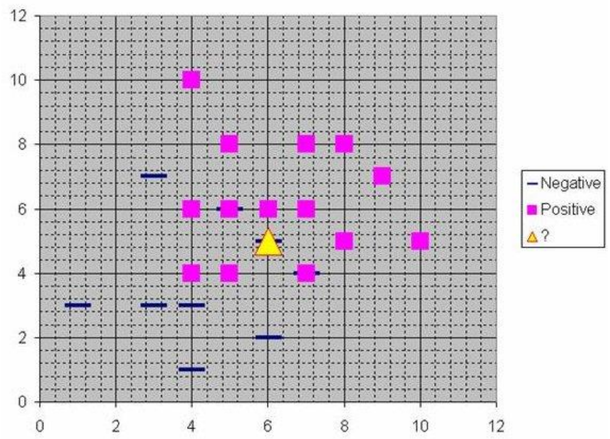

Fig. 1. Prediction (positive or negative)

Assume that each case in the training set has two predictor variables, x and y. The cases are plotted using their x, y coordinates as shown in the figure. Also assume that the target variable has two categories, positive which is denoted by a square and negative which is denoted by a dash. Now, suppose we are trying to predict the value of a new case represented by the triangle with predictor values x=6, y=5.1. Should we predict the target as positive or negative?

Notice that the triangle is position almost exactly on top of a dash representing a negative value. But that dash is in a fairly unusual position compared to the other dashes which are clustered below the squares and left of center. So it could be that the underlying negative value is an odd case.

The nearest neighbour classification performed for this example depends on how many neighbouring points are considered. If 1-NN is used and only the closest point is considered, then clearly the new point should be classified as negative since it is on top of a known negative point. On the other hand, if 9-NN classification is used and the closest 9 points are considered, then the effect of the surrounding 8 positive points may overbalance the close negative point.

An RBF network positions one or more RBF neurons in the space described by the predictor variables (x, y in this example). This space has as many dimensions as there are predictor variables. The Euclidean distance is computed from the point being evaluated (e.g., the triangle in this figure) to the center of each neuron, and a radial basis function (RBF) (also called a kernel function) is applied to the distance to compute the weight (influence) for each neuron. The radial basis function is so named because the radius distance is the argument to the function.

Weight = RBF (distance)

The further a neuron is from the point being evaluated, the less influence it has.

Training RBF Networks:

The following parameters are determined by the training process:

1. The number of neurons in the hidden layer.
2. The coordinates of the center of each hidden-layer RBF function.
3. The radius (spread) of each RBF function in each dimension.
4. The weights applied to the RBF function outputs as they are passed to the summation layer.

Various methods have been used to train RBF networks. One approach first uses K-means clustering to find cluster centers which are then used as the centers for the RBF functions. However, K-means clustering is a computationally intensive procedure, and it often does not generate the optimal number of centers. Another approach is to use a random subset of the training points as the centers.

The computation of the optimal weights between the neurons in the hidden layer and the summation layer is done using ridge regression. An iterative procedure developed by Mark Orr (Orr, 1966) is used to compute the optimal regularization Lambda parameter that minimizes generalized cross-validation (GCV) error.

3. K-Means Clustering

K-means is one of the simplest unsupervised learning algorithms that solve the well-known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed apriori. The main idea is to define k centers, one for each cluster. These centers should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest center. When no point is pending, the first step is completed and an early group age is done. At this point we need to re-calculate k new centroids as barycenter of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new center. A loop has been generated. As a result of this loop we may notice that the k centers change their location step by step until no more changes are done or in other words centers do not move any more. Finally, this algorithm aims at minimizing an objective function know as squared error function given by:

$$J(V) = \sum_{i=1}^{c} \sum_{j=1}^{c_i} \left(\lVert x_i - v_j \rVert \right)^2 \quad (1)$$

Where, '$||x_i - v_j||$' is the Euclidean distance between x_i and v_j.
'c_i' is the number of data points in i^{th} cluster.
'c' is the number of cluster centers.

Algorithmic steps for k-means clustering:
Let $X = \{x_1,x_2,x_3,\ldots\ldots,x_n\}$ be the set of data points and $V = \{v_1,v_2,\ldots\ldots,v_c\}$ be the set of centers.
1) Randomly select 'c' cluster centers.
2) Calculate the distance between each data point and cluster centers.
3) Assign the data point to the cluster center whose distance from the cluster center is minimum of all the cluster centers.
4) Recalculate the new cluster center using:

$$v_i = (1/c_i) \sum_{j=1}^{c_i} x_i \quad (2)$$

Where, 'c_i' represents the number of data points in i^{th} cluster.
5) Recalculate the distance between each data point and new obtained cluster centers.
6) If no data point was reassigned then stop, otherwise repeat from step 3.

4. Simulation and Analysis

The channel state is divided into two types: occupation and idle. In the simulation, we use the cluster threshold value simulate the channel state occupied by licensed user respectively. We took the first 350 data to train the neural network and the last 70 data as the test data to test the neural network. First, we used k cluster to simulate. The different cluster is shown in Fig. 2. The comparison of the cluster with the nearby distance is shown in Fig. 3, and the threshold value is presented in Fig. 4. Fig. 5, shows the threshold value for the clusters. They showed that K-RBF define threshold value for the occupied users and non-occupied users. The Fig. 5, Small threshold value for non-occupied users, both the figures differentiate between the occupied user and non-occupied users.

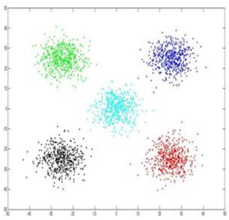

Fig. 2. Different clusters

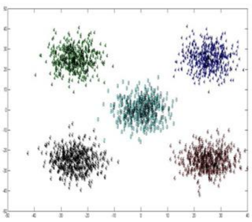

Fig. 3. Total clusters in the data

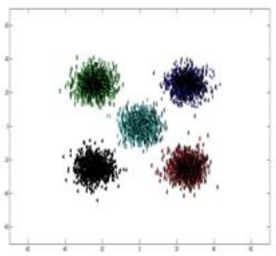

Fig. 4. Threshold value for occupied users

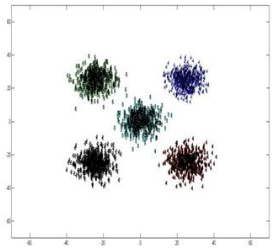

Fig. 5. Small threshold value for non-occupied users

5. Conclusion

In this paper, we proposed a spectrum prediction algorithm: K-RBF algorithm. We used K-means clustering algorithm to obtain the hidden nodes center and base function width of the RBF neural network. Then, we train and form more appropriate neural network. Through simulation, we found that K-RBF algorithm can achieve better predicting precision. Thus, we have used the prediction information to sensing the licensed user spectrum more simply. And, it will reduce the resource consumed.

References

[1] Federal Communications Commission. "Spectrum policy task force," ET Docket no. 02-135, Nov. 2002.
[2] Acharya P. A., Singh S., Zheng H., (2006) Reliable open spectrum communications through proactive spectrum access. Proc of the 1st International Workshop on Technology and Policy for Accessing Spectrum (TAPAS06):1-8.
[3] Zhao Jianli, Wang Mingwei, Yuan Jinsha, (2011). Based on neural network spectrum prediction of cognitive radio. 2011 International Conference on Electronics, Communication and Control (ICECC): 762-765.
[4] Marko H., Sofue P., Aarne M., (2008). Performance improvement with predictive channel selection for cognitive radios. Proc of the 1st International Workshop on Cognitive Radio and Advanced Spectrum Management.1-5
[5] Stefan G., Lang T., Brain M., (2008). Interference-aware OFDMA resource allocation: a predictive approach. Proc of IEEE Military Communications Conference. 1-5.
[6] Zhao Q., Tong L., Swami A. et al., (2007). Decentralized cognitive MAC for opportunistic spectrum access in Ad hoc networks: a POMDP framework. IEEE J Select Areas Communication: Special Issue Adaptive, Spectrum Agile Cognitive Wireless Networks, 25(3):589-600
[7] Broomhead, D. Lowee, (1988). Multivariable function interpolation and adaptive networks. Complex system. (2): 321~355.
[8] Khaled Alsabti, Sanjay Ranka, Vineet Singh, (1997). An efficient K-means clustering algorithm. Electrical Engineering and Computer Science. 1(1):43-39.