Improved Sensing Performance with Autoencoder and Ensemble Classifier

Noor Gul
Department of Electronics
University of Peshawar
Peshawar, Pakistan
noor@uop.edu.pk

Saeed Ahmed
Department of Electrical Engineering
Mirpur University of Science and Technology
AJK, Pakistan
saeed.ahmed@must.edu.pk

Su Min Kim
Department of Electronics Engineering
Tech University of Korea
Gyeonggi-do, Korea
suminkim@tukorea.ac.kr

Junsu Kim
Department of Electronics Engineering
Tech University of Korea
Gyeonggi-do, Korea
junsukim@tukorea.ac.kr

***Abstract*—Cognitive Radios (CRs) improve spectrum utilization by intelligently sensing and learning from the radio environment. Spectrum sensing is one of the most pre-requisite jobs for the SUs before they access the PU channel opportunistically. This paper employs various machine learning tools in a collaborative environment to perform sensing. First, the sensing time is reconfigured to maximize the spectrum utilization leading to low cost and high channel throughput. In the first step ensembled classifier estimates sensing time. Next, the soft energies of the different categories of false sensing users (FSUs) are cleaned using the denoising Autoencoder before soft decision fusion schemes. The ensemble classification method in this paper leads to an accurate estimation of the channel for the target detection, false alarm, and wireless channel conditions. The result shows that the ensemble classifier with the AdaBoost method can predict efficiently with a better F1 score, accuracy, and Matthews correlation coefficient (MCC).**

***Index Terms*—Reconfigurable Sensing Time, Ensemble Classifier, Random Forest Algorithm, Neural Network, AdaBoost, Autoencoder**

I. Introduction

The radio spectrum investigations show that the allocated spectrum to licensed users is underutilized. Some of the frequency bands are fully utilized, some are partially in use, and the others are mostly unemployed. Cognitive radio (CR) has the intelligence that senses and learns from the environment. It adapts its internal states, such as the transmitting power, carrier frequency, and modulation scheme to provide reliable communication [1]. Secondary users (SUs), as opportunistic users, try to access the primary user (PU) channel when the PU is not active [2]. Cooperative sensing is a more appropriate choice to deal with multipath fading and ambiguities in the wireless channel [3]. A centralized collaborative model has a fusion center (FC) that collects sensing notifications from the individual contributing users and takes final using hard, and soft-decision schemes [4].

One of the threats to the centralized cooperative model is the presence of false sensing users (FSUs) with fake sensing notifications to the FC. The effects of malicious users (MUs) and their prevention schemes are studied in [5]. Heuristic algorithms are studied in [6] to optimize detection and false alarm probabilities. A reconfigurable wireless network is discussed in [7] that delivers a thorough analysis of the method and strategies. Optimization of the individual channel parameters such as sensing period to maximize throughput with the constraint to reduce interference for the PU is discussed. The CRN throughput is maximized by keeping the quality of service (QoS) for the PUs with optimal power allocation and sensing time [8]. A concise survey is illustrated in [9] that shows the role and importance of machine learning (ML) and AI-based learning methods. [11] has realized the reinforcement learning concept to achieve cognition cycle (CC) for unlicensed users. A considerable time is consumed in spectrum sensing when using traditional spectrum sensing approaches. Therefore, the work in [12] has examined deep learning and convolutional neural network (CNN) based spectrum sensing systems.

In the cooperative model, an increase in the number of sensing samples for the collaborative users guides to higher sensing accuracy, resulting in more energy consumption and a reduction in the channel throughput. On the contrary, optimal sensing sample choice for the individual and user in cooperation benefits-improved throughput, low sensing cost, and guaranteed sense performance with minimum error in sensing. An ensemble classifier is suggested using the AdaBoost assembling method along with denoising autoencoder that cleans the

This work was supported in part by the Ministry of Science and ICT (MSIT), Korea, under the Information and Technology Research Center (ITRC) support program (IITP-2022-2018-0-01426) and by the National Research Foundation of Korea (NRF) funded by the Korea government (MIST) (No.2021R1A2C1013150).

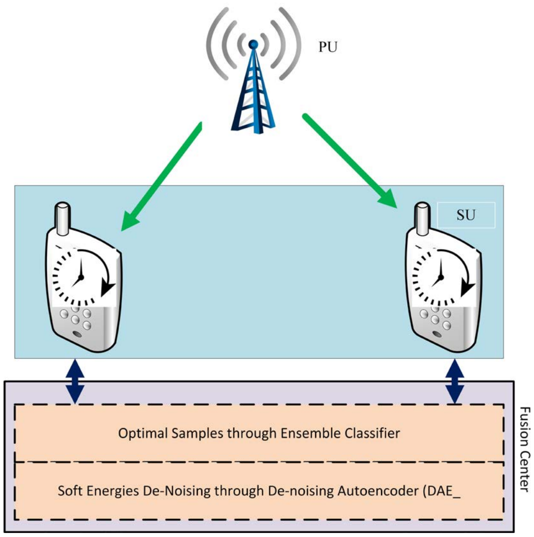

Fig. 1. CSS with reconfigurable sensing time.

sensing data contents from the reports of FSUs.

II. System Model and Background

This work carries a single PU channel and pursues an energy detector for spectrum sensing in Figure 1. The H_0 and H_1 hypothesis test of the j^{th} user out of m users for sensing is as

$$\begin{Bmatrix} H_0, \; x_j(l) = v_j(l) \\ H_1, \; x_j(l) = g_j c(l) + v_j(l) \end{Bmatrix}, j \in \{1, ..., m\}, l \in \{1, ..., k\}. \tag{1}$$

Eq. (1) has g_j as the channel gain for the j^{th} user. $c(l)$ is the PU signal in the l^{th} slot. The $v_j(l)$ is Gaussian noise with mean zero and variance $\sigma_{v_j}^2$ between the j^{th} user and PU. The sensing energy of the users has total sensing samples $k = 2B\tau_s$. Here B shows the bandwidth with τ_s sensing period. An energy representation of (1) in the k number of sensing samples are as

$$E_j(i) = \begin{Bmatrix} \sum_{l=l_i}^{l_i+k-1} |v_j(l)|^2, & H_0 \\ \sum_{l=l_i}^{l_i+k-1} |g_j c(l) + v_j(l)|^2, & H_1 \end{Bmatrix}. \tag{2}$$

The soft energy reports under satisfactory sensing samples converge to the Gaussian random variable is

$$E_j \sim \begin{Bmatrix} N\left(\mu_0 = k, \, \sigma_0^2 = 2k\right), & H_0 \\ N\left(\mu_1 = k(\eta_j + 1), \sigma_1^2 = 2k(\eta_j + 1)\right), & H_1 \end{Bmatrix}, \tag{3}$$

where $\left(\mu_0, \, \sigma_0^2\right)$ and $\left(\mu_1, \, \sigma_1^2\right)$ are the mean and variance results of the energy distribution, while η_j is j^{th} user channel signal-to-noise-ratios (SNRs).

Figure 2 shows the sensing and data transmission time of the cooperative users are τ_s and $T - \tau_s$. The sensing time between the cooperative users is assumed to be synchronized.

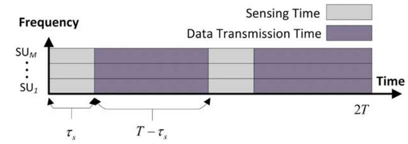

Fig. 2. Sensing and data transmission time.

The cooperative environment in this paper is also considered to be protected against the false sensing notices of Yes False Sensing (YFS) and No False Sensing (NFS) users. The normal SUs have entirely different energy distributions in the H_0 and H_1 hypothesis. The regular users report high energy statistics in the H_1 and low energy statistics in H_0 to the FC. Similarly, the YFS reports high energy statistics in both hypotheses to the FC, while the NFS always reports low energy. The YFS contribution reduces the throughput of the SUs, and the NFS presence is the primary cause of interference for the licensed PUs.

III. Sensing time reconfiguration and data denoising

This section illustrates sensing time estimation using the proposed ensembled classifier. The estimated sensing samples is then used in the energy reports of the cooperative users. Finally, sensing data is denoised and cleaned with the DAE before employing soft combinations.

A. Dataset Reconfigure Optimal Sensing Time

This section examines the construction of the dataset for the ML classifier. The false alarm probability representation is as

$$P_f = Q\left(\frac{(\beta - \mu_0)}{\left(\sqrt{\sigma_0^2}\right)}\right) = Q\left(\frac{(\beta - k)}{\left(\sqrt{2k}\right)}\right), \tag{4}$$

where B is the threshold and P_f is the false alarm probability. Likewise, expression of the probability of detection P_d when H_1 is true are as

$$P_d = Q\left(\frac{(\beta - \mu_1)}{\left(\sqrt{\sigma_1^2}\right)}\right) = Q\left(\frac{(\beta - k(\eta_j + 1))}{\left(\sqrt{2k(\eta_j + 1)}\right)}\right). \tag{5}$$

The ensemble classifier finds the optimal sensing samples using detection probability as

$$P_d = Q\left(\frac{\left(\sqrt{2}Q^{-1}(P_f) - \sqrt{k}\eta_j\right)}{\left(\sqrt{2(\eta_j + 1)}\right)}\right), \tag{6}$$

where $Q(\,.\,)$, and $Q^{-1}(\,.\,)$ shows the complementary and inverse complementary Gaussian distribution functions. This develops a dataset to train the ensemble classifier and to predict sensing samples. The feature vector or example of a data point in the dataset is represented with a vector $x = \begin{bmatrix} \eta_j & P_f & k & P_d \end{bmatrix}$. This allows the ensemble classifier to be trained with $(\eta_j, \, P_f, \, P_d)$ input and k_j output label.

B. Classification with the AdaBoost

As individual classifiers may result in biased prediction, ensembling the categories of different classifiers is a more acceptable option.

The soft energy reports of the users found a history matrix with normal and corrupted data as

$$X = \begin{bmatrix} x_1 & x_2 & \dots & x_n \end{bmatrix}^T, i \in \{1, 2, ..., n\}, \tag{7}$$

where x_1,x_2,x_3 and x_n are the n feature vectors that consist of the target detection, false alarm and SNRs as the input and sensing samples as the response. The adaptive boosting (AdaBoost) construct more strong classifier with ensembling of weak classifiers. In this work the data set is represented as: $T = \{(x_i,\ y_i)\}_{i=1}^{n}$ The training set T is an $n\times(m+1)$ matrix also represented as

$$T = \left[x_i \left| y_i \right.\right], i \in \{1, 2, ..., n\}, \tag{8}$$

where x_i are the input features vectors to be classified with the ensembling method, the input training data to the ensemble classifier in the matrix form T has S and Y submatrices. Here S is the n feature matrix and Y as the $n\times 1$ output label matrix consisting of H_0 and H_1 hypotheses.

In AdaBoost, each r^{th} classifier is set with decision weights by knowing the predictions of the $(r-1)$ classifiers to express the boosted classifier as

$$e_{r-1}(x_i) = \sum_{p=1}^{r-1} \alpha_p h_p(x_i),\ p \in \{1, ..., r-1\},\ i \in \{1, ..., n\}, \tag{9}$$

where $h_p(x_i)$ is the value predicted by the p^{th} classifier and α_p is the weight assigned to the classifier prediction. Similarly, the r^{th} classifier prediction is included with $h_r(x_i)$ as prediction and α_r optimum weight. The above can be written as

$$e_r(x_i) = e_{r-1}(x_i) + \alpha_r h_r(x_i), \tag{10}$$

where $e_r(x_i)$ is the compound predicted value of r classifiers. We are interested in a closed form formula for α_r, which assigns α_r a value such that the total error of prediction is minimized. The expression for the weight α_r in final form is

$$\alpha_r = \frac{1}{2} \ln\left(\frac{1-e_m}{e_m}\right), \tag{11}$$

where $e_m = {}^{W_e}/_{W}$ is the weighted error rate of the weak classifier h_r.

In case of fixed T reduction in the sensing duration τ_j leads to a higher throughput. As $\tau_j = \frac{k_j}{f_s}$ has a sampling frequency f_s. Therefore, higher throughput is obtained by lowing the τ_j through reduced k_j. The re-configurable network estimates the optimum samples k_j using the ensemble classifier. The energy spent in sensing is directly proportional to the sensing duration and is represented as

$$s(i) = \sum_{j=1}^{m} (E_j(i) \times \tau_j), \tag{12}$$

where $s(i)$ is the sensing energy consumed by the cooperative users based on the reconfigured sensing samples k_j, therefore, the increase in $k_j = \tau_j \times f_s$ is expected to increase the sensing cost of the system.

C. Data denoising through Autoencoder

The Autoencoder is the fully connected feed-forward neural network, where the inputs are equal to the output and trained without any label information. This paper employs a variant of the simple Autoencoder that is DAE. The DAE offers more features than the simple AE, such as reconstructing the input from attacked or corrupted sensing data. For example, for a given x in Figure 3, the encoder transforms deteriorated and attacked input into hidden or latent space representation as h such as

$$h = f(W_1 s + b). \tag{13}$$

This f is a non-linear activation function such as the sigmoid function. $W_1 \in IR^{n\times m}$ is the weight matrix, and $b \in IR^n$ is optimized in encoder with n nodes in latent space. Then, the decoder evolves the latent space into a reconstructed vector, $\hat{s}$, at the output layer as

$$\hat{s} = g(W_1 s + c). \tag{14}$$

In order to improve the learning efficiency $W_1 = {W_2}^T$. The training objective of the DAE is to find the optimal parameters, $\psi = \{W_1,\ b,\ c\}$ that minimize the reconstruction error as

$$\min_{\psi} \sum_{i=1}^{n} \left\| s_i - \hat{s}_i \right\|^2. \tag{15}$$

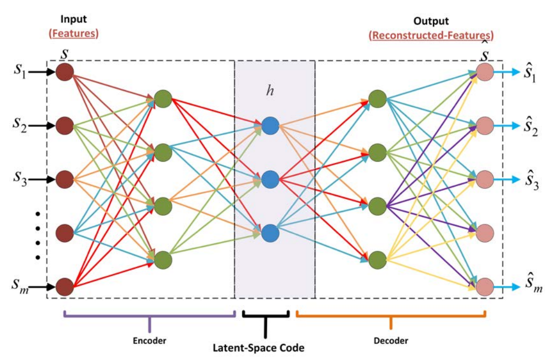

Fig. 3. Denoising Autoencoder.

IV. Result and Discussions

This section shows a performance comparison of the different classifiers such as k-nearest neighbor (KNN), neural network, decision tree (DT), and proposed ensemble classifier. First, the optimum sensing samples are estimated with the ML classification. An energy detector is then used to follow the modified sensing samples. This article evaluates the required samples with the target detection, false alarm, and SNRs expected from the cooperative SUs. Sensing performance is then improved by cleaning the sensing contents using the

DAE scheme. The dataset used to train and test the ML classifiers is formed with target SNRs from -15dB to -10.25dB, total reporting users of ten, and target detection probability from 0.01 to 0.99. The associated number of samples is computed with equation (6) in the first step. Therefore, we added 40,000 pieces (features). For 20 levels of detection probability ranging between 0.01 to 0.99, the false alarm probabilities in the dataset are ten. Finally, the model is trained using SNRs, false alarms, and detection probability as the input features and samples as the output feature. The sampling frequency and signal bandwidth values are 40KHz with a frame duration of 100ms. Figure 4 shows the classifier accuracy performance that shows high accuracy results of the proposed ensemble classification method. The ensemble classifier accuracy is followed by the DT, neural network, and KNN. The result shows the worst accuracy results by the GNB classifier. Next, in Figure 5, F1-score performance shows high F1-score results for the ensemble classifier. Finally, Figure 6 shows the classifiers' authenticity with the Matthews correlation coefficient (MCC) results.

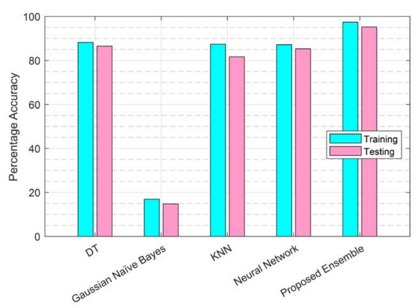

Fig. 4. Classifiers Accuracy in estimating optimal sensing samples.

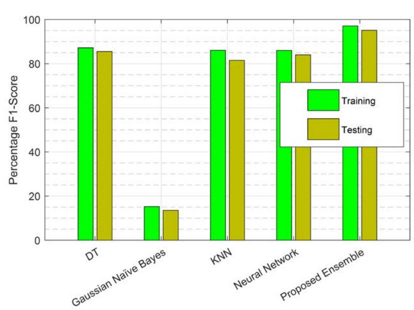

Fig. 5. Classifiers F1-Score in estimating optimal sensing samples.

V. Conclusions

The Raleigh fading environment follows CSS to achieve better sensing reliability. To reduce the slow convergence of the CSS, SUs have to report instantly with less time consumption in sensing to ensure decision reliability. The estimated optimal sensing time increases the channel throughput and reduces sensing cost in the adjustable sensing time through the proposed ensemble classifier. This paper proposed an ensemble

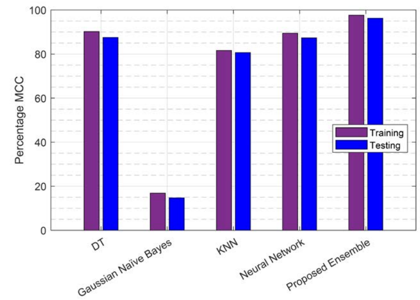

Fig. 6. Classifiers MCC in estimating optimal sensing samples.

classifier method to determine optimal sensing samples. The users then use this modified sensing time, and soft energy reports are determined. The energy reports are finally cleaned and denoised through the DAE scheme in the paper. Accuracy, F1-score, and MCC comparison are shown for various ML classifiers.

References

[1] J. Mitola, and G. Q. Maguire, "Cognitive radio: making software radios more personal," IEEE Personal Communications, vol. 6, no. 4, pp. 13–18, 1999.

[2] Q. Wu, G. Ding, Y. Xu, S. Feng, and Z. Du., "Cognitive internet of things: A new paradigm beyond connection," IEEE Internet of Things Journal, vol. 1, no. 2, pp. 129–143, 2014.

[3] I. F. Akyildiz, B. F. Lo and R. Balakrishnan, "Cooperative spectrum sensing in cognitive radio networks: A survey," Physical Communication, vol. 4, no. 1, pp. 40–62, 2011.

[4] N. Gul, M. S. Khan, J. Kim and S. M. Kim, "Robust spectrum sensing via double-sided neighbor distance based on genetic algorithm in cognitive radio networks," Mobile Information Systems, vol. 2020, pp. 1-10, 2020.

[5] N. Gul, I. M. Qureshi, A. Naveed, A. Elahi and I. Rasool, "Secured soft combination schemes against malicious-users in cooperative spectrum sensing," Wireless Personal Communications, vol. 108, no. 4, pp. 389–408, 2019.

[6] N. Gul, M. S. Khan, S. M. Kim, J. Kim and I. Ullah, "Particle swarm optimization in the presence of malicious users in cognitive IoT networks with data," Scientific Programming, vol. 2020, pp. 1-11, 2020.

[7] A. El-Mougy, M. Ibnkahla, G. Hattab and W. Ejaz, "Reconfigurable Wireless Networks," Proceedings of the IEEE, vol. 103, no. 7, pp. 1125–1158, 2015.

[8] Y. Pei, Y. C. Liang, K. C. Teh and K. H. Li, "Sensing-throughput tradeoff for cognitive radio networks: A multiple-channel scenario," IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC, Tokyo, Japan, pp 1257–1261, 2009.

[9] N. Abbas, Y. Nasser and K. E. Ahmad, "Recent advances on artificial intelligence and learning techniques in cognitive radio networks," Eurasip Journal on Wireless Communications and Networking, vol. 2015, pp. 1–20, 2015.

[10] Z. Qin, X. Zhou, L. Zhang, Y. Gao, Y. C. Liang et al., "20 years of evolution from cognitive to intelligent communications," IEEE Transactions on Cognitive Communications and Networking, vol. 6, no. 1, pp. 6–20, 2020.

[11] K. A. Yau, P. Komisarczuk and P. D. Teal, "Applications of reinforcement learning to cognitive radio networks," 2010 IEEE Int Conf Commun Work ICC 2010, Cape Town, South Africa, pp 1–6, 2010.

[12] S. Zheng, S. Chen, P. Qi, H. Zhou and X. Yang, "Spectrum sensing based on deep learning classification for cognitive radios," China Communications, vol. 17, no. 2, pp. 138–148, 2020.