EXTRACTING SPATIAL-TEMPORAL FEATURES USING DEEP LEARNING IN COOPERATIVE SPECTRUM SENSING

Doi Thi Lan[1,*]

Abstract

In cognitive radio systems, spectrum sensing (SS) plays a vital role in detecting the presence of the primary user (PU). In this work, a cooperative spectrum sensing (CSS) model based on a graph convolution network and bidirectional long short-term memory (GCN-BiLSTM) is proposed. Specifically, the GCN architecture is applied to extract the relationship between the secondary users (SUs). Besides, the BiLSTM architecture learns the temporal correlation of sensing information at SUs. The presence of PU is decided based on spatial-temporal features, which are combined from the outputs of GCN and BiLSTM. The proposed model is evaluated in a scenario of a dynamic channel (i.e., fading channel). Experimental results show that the GCN-BiLSTM model obtains a detection probability (P_d) of 84.5% and an accuracy of 87.25% at a Signal-to-Noise Ratio (SNR) of -14 dB, demonstrating superior performance compared to the baseline models.

Index terms

Radio cognitive network; CSS; GCN; BiLSTM.

1. Introduction

Due to the rapid growth of wireless communication systems, the spectrum resource demand has become urgent. Besides, available spectrum resources are limited. To solve the spectrum scarcity, the cognitive radio network (CRN) is proposed as a promising solution [1], [2]. In CRN, there are two types of users: PU and secondary user (SU). PU is also known as the licensed user, and they are prioritised accessing the licensed spectrum. Besides, SU can use the licensed spectrum when PU is not being accessed. SU needs to ensure that there is no interference with PU. Thus, detecting whether PU occupies available spectrum bands remains a challenging aspect of CRN.

Over the past decades, many SS algorithms have been proposed. For example, due to the low complexity, the energy detection method (ED) is applied widely in SS [3], [4].

[1]Faculty of Radio and Electronic Engineering, Le Quy Don Technical University
[*]Corresponding author, email: doilan@lqdtu.edu.vn
DOI: 10.56651/lqdtu.jst.v14.n01.1041.ict

However, ED performs poorly at scenarios of low SNR, while cyclostationary detection overcomes this limitation under strong noise [5], [6]. However, cyclostationary feature-based detection requires prior information on the PU's signal and high computational complexity. If there are only a few samples of the received PU's signal, the matched filter method is efficient [7], [8]. This method also needs to know fully the PU's signal and the channel.

Due to the powerful ability of machine/deep learning algorithms (ML/DL), many SS methods have been proposed [9], [10]. For instance, the authors in [11] considered a CSS approach based on K-means clustering algorithm. Energy vectors from SUs are collected to classify using K-means algorithm. After training model, the centroid of K clusters is used to detect the presentation of PU. The studies in [12], [13] are proposed to detect the presence of PU using LSTM-based models. In these works, the LSTM architecture exploite the temporal correlation of the received signal. In [14], [15], Convolutional Neural Network (CNN)-based models are proposed to sense the spectrum in CRN. The one dimension (1D)-CNN architecture is utilized to learn the energy correlation features of the PU signal [14]. To improve the performance of SS, the sample covariance matrix (CM) between received signals at SUs is used as the input of the CNN model in [15]. Note that the two dimensional (2D)-CNN architecture is applied to extract the spatial correlation between SUs in [15]. To learn the spatial and temporal features of signal for each single node in CRN, the authors in [16] introduced a CNN- Gated Recurrent Unit (GRU) model for SS. First, the PU's signal features are learned at each SU. Then, the final cooperation result is decided using the features achieved by all PUs.

In this work, a novel CSS model is proposed based on a graph convolution network and bidirectional long short-term memory architecture (GCN-BiLSTM model). To improve the performance of SS in the wireless environment with dynamic changes, the proposed model exploits correlation features of received signals at SUs. In particular, GCN is known as the network architecture that can discover the relationships of structural information between entities. Thus, in our work, the GCN is utilized to learn the spatial correlation between the received signals by SUs. The GCN architecture also applied to improve the detection performance of CSS in [17]. However, in the work, they only learn the spatial relationships between SUs using the GCN and ignore the temporal relationship. Meanwhile, in the proposed model, the BiLSTM is applied to exploit the temporal features of PU at SUs. To learn both sides of received signals, the BiLSTM architecture is used instead of the LSTM architecture in this work.

To evaluate the proposed model, experiments are performed in a scenario of dynamic channel. Results show that our method outperforms the compared models.

The main contribution of this paper can be summarized as follows:

- We propose a novel CSS framework based on GCN and BiLSTM to extract correlation features of received signals at SUs. In the framework, the spatial correlation between signals at other SUs is extracted using GCN. Besides, the temporal correlation of the received signals is learned using BiLSTM. The

achieved spatial-temporal features are combined to detect the presence of PU.
- The dynamic channel condition between PU and SUs is studied in our work (i.e., the fading channel model). Besides, we evaluate the effect of the SU number on the CSS performance. Results show that the proposed model achieves 84.5% in terms of P_d at a low SNR of -14 dB.

The work is organized as follows. First, Section 2 presents the problem formulation. Then, the methodology is shown in Section 3. Experimental results are depicted in Section 4. Finally, a conclusion is given in Section 5.

2. Problem formulation

In this work, a general scenario of CRN, which contains one PU, M SUs and one fusion center (FC), is considered. In CSS, SUs first sense the transmitted signal from the PU. Then, SUs upload their sensing information to FC through the reporting channel. Finally, the PU's presence is decided at FC. Note that the reporting channel is assumed to be perfect in this work. Besides, CRN is considered in the scenario of dynamic changes in a wireless environment (i.e., the channel can be affected by fading).

At SU, the received signal from PU can be written as

$$y_m(n) = \begin{cases} h_m(n)x(n) + \eta_m(n), & H_1 \\ \eta_m(n), & H_0 \end{cases} \quad (1)$$

here, $y_m(n)$ is nth received signal sample during a sensing period at mth SU, and $n = 1, 2, 3, ...N$; N is the total number of received signal samples at each SU in a sensing period; $x(n)$ is transmitted signal from PU, and $\eta_m(n)$ is additive Gaussian noise which has a zero mean and a variance of σ^2; $h_m(n)$ is the channel gain; two hypotheses are considered in the SS problem: H_1 represents the presence of PU on the channel and H_0 indicates the absence of PU.

To solve the SS problem, we proposed a novel deep learning model based on GCN and BiLSTM. First, sensing data from all SUs at the FC is preprocessed. Then, the model is trained using the preprocessed data. Finally, the presence of PU is checked.

3. Methodology

3.1. Proposed framework

In this subsection, a novel CSS framework is introduced in Figure 1. In particular, the received signals at SUs are first pre-processed to obtain the energy vectors. Then, a feature matrix and a graph are built using the energy vectors, which are fed into GCN and BiLSTM to extract signal features. In particular, GCN learns the high-level relationship between received signals from other SUs. Besides, Bi-LSTM architecture extracts the temporal relationship of received signals in a sensing period. Finally, the extracted features are combined to make the final decision about the presence of PU.

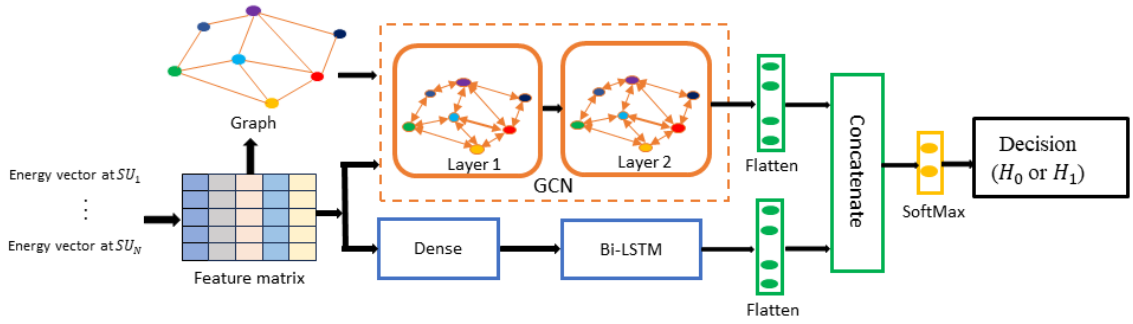

Fig. 1. Introducing a novel CSS framework based on GCN and BiLSTM.

3.2. Pre-processing data

Before feeding the model, received signals at SUs need to be preprocessed. In this work, the energy level of received signals at SUs is used to extract correlation features.

In particular, during a sensing duration, the received signal strength (RSS) at each SU is accumulated as follows:

$$x_m = \frac{1}{N}\sum_{n=1}^{N} |y_m(n)|^2 , \tag{2}$$

where x_m is accumulated RSS of received signal y_m during a sensing period at mth SU; RSS values at SUs are used to build the feature matrix and the graph as inputs of the model.

3.3. Learning correlation feature of SUs using GCN

This subsection first presents the fundamental of GCN. Then, the construction of the feature matrix and the graph from RSS values is introduced. Finally, learning spatial correlation features from received signals at other SUs using GCN layers are discussed.

The GCN architecture is proposed to learn high-level relationships of graph-structured data [18]. An undirected graph is defined as $G = (V, A, E)$, where V is a set of nodes or vertices of the graph; A is defined as an adjacency matrix, which represents the relationship between nodes in the graph; E denotes a set of edges in the graph.

From RSS values at SUs, a feature matrix X of dimension MXD is built as

$$X = \begin{bmatrix} x_{11} & x_{12} & x_{13} & \cdots & \cdots & x_{1D} \\ x_{21} & x_{22} & x_{23} & \cdots & \cdots & x_{2D} \\ \vdots & \vdots & \vdots & \cdots & \cdots & \vdots \\ x_{M1} & x_{M2} & x_{M3} & \cdots & \cdots & x_{MD} \end{bmatrix} \tag{3}$$

where $X \in \mathbb{R}^{M\times D}$; D is feature dimension of each node, and $D = 1$ in this work.

If there is an edge between nodes v_i and v_j, the correlation weight between them is determined as the element $A(i,j)$ of matrix A. Here $A(i,j)$ can be determined from the radial basis function:

$$A(i,j) = \exp\left(-\frac{\|x_i - x_j\|^2}{\rho^2}\right), \tag{4}$$

where x_i and x_j are feature vectors of nodes v_i and v_j, respectively; ρ controls the width of the radial basis function.

The GCN architecture is known as an extension of CNN for extracting spatial features of the graph-structured data. In this work, the GCN is used to capture the complex correlations between received signals from other SUs. A multi-layer GCN is defined as follows [19]:

$$X^{l+1} = \sigma\left(\widetilde{D}^{-1/2}\widetilde{A}\widetilde{D}^{-1/2}X^l W^l\right), \tag{5}$$

here, $\widetilde{D}$ is a degree matrix, and $\widetilde{D}_{ii} = \sum_j \widetilde{A}_{ij}$; $\widetilde{A} = A + I_N$ is the adjacency matrix with added self-loops; X^l and W^l are matrices of input and trainable weights at the l^{th} layer, respectively; X^{l+1} is the output of GCN at the l^{th} layer. Note that $X^0 = X$. As increasing the number of GCN layers reduces information transmission efficiency [20], we adopt two GCN layers.

3.4. Learning temporal correlation using BiLSTM

The architecture of BiLSTM is used efficiently for extracting correlation features of sequences in the time domain. With the learning ability from two sides of the input sequences, BiLSTM can perform better than LSTM. In particular, BiLSTM contains two layers of transmission information following the reversed directions: the forward and backward layers. In the former, information on the input sequence is transmitted from left to right. In contrast, the latter gets the input from right to left of the sequence.

In this work, BiLSTM is applied to capture the temporal correlation of received signals at SUs in a sensing period. The correlation also represents the activity pattern of PU. The feature vector, which contains energy information of received signals from all SUs in each sensing period, is fed into BiLSTM. Since dimension of energy vector at each SU in a sensing duration is one (i.e., $D = 1$), the energy vector is first projected into a higher space through the dense layer. Then, the projected energy vector is used as input of the BiLSTM layer for extracting the temporal correlation.

The spatial-temporal correlation features of SUs extracted from GCN and BiLSTM are combined using a concatenate layer. The output of the concatenate layer is a representation of the received signals at SUs, which captures the spatial-temporal informational features of signals. The signal representation is used to detect the presence of PU through a fully connected layer and a softmax function.

3.5. Sensing the presence of PU

To decide the presence of PU on the channel, the output of the softmax function is used. In particular, a detection threshold is determined. The decision about the state of PU as follows:

$$\mathrm{Pr}_{[\mathrm{PU}|H_1]} \underset{H_1}{\overset{H_0}{\lessgtr}} \gamma \tag{6}$$

here, $\mathrm{Pr}_{[\mathrm{PU}|H_1]}$ is the presence probability of PU, which is obtained from the output of the trained model.

The detection threshold γ is a crucial parameter in the CSS problem. If the value of γ is low, the model can detect noise as the PU signal, leading to a high false alarm probability (P_f). This means that the channel is idle, while the model shows that the channel is occupied by PU. In this case, the spectrum holes may not be detected and the spectrum utilization efficiency is decreased. In contrast, a high value of γ can make the model fail to detect the weak PU signal, causing interference to the PU. This means that the PU is active while the model can not detect it. Thus, the SU occupies the channel and interferes with the PU. The missed detection is direr than the false alarm problem. In other words, avoiding interference for PU is a higher priority. Thus, the model tends to improve the correct detection probability (P_d) (i.e., decreasing the missed detection probability) at an accepted value of P_f. From this point, the detection threshold γ is chosen following the predefined value of P_f.

Table 1 depicts the value of the threshold γ when changing the P_f. The result is obtained when evaluating the proposed model at SNR of -14 dB.

Table 1. The value of the detection threshold γ when changing P_f

P_f	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
γ	0.917	0.681	0.557	0.472	0.397	0.329	0.275	0.224	0.171	0.117	0.030

From Table 1, the value of γ is low when P_f is high and otherwise. The results are suitable for the above statement about the relationship between γ and P_f.

4. Experimental results

This section evaluates the proposed framework on generated data, comparing its performance with five baseline models: Kmeans [11], CNN [14], CNN-GRU [16], CM-CNN [15] and GCN [17] models.

4.1. Dataset preparation

Due to the lack of public datasets in the considered scenarios, data for evaluating the existing models have been simulated [11], [14], [17]. In this work, the detailed description of generating dataset is presented as follows. First, the channel between PU

and SUs is assumed to follow a Rayleigh fading model. This means that the probability density function of the channel gain follows the Rayleigh distribution, as given in (7),

$$f(h) = \frac{h}{\sigma^2} \exp\left(-\frac{h^2}{2\sigma^2}\right), \quad h \geq 0 \tag{7}$$

where the channel gain h is the envelope of Gaussian random variables with zero mean and variance be σ. Note that the channel coherent time can be assumed to be longer than the sensing period. Therefore, the channel gain remains constant during the sensing duration. Besides, the severity of fading channel can be controlled by the scale parameter σ in (7). Larger σ values indicate more severe fading; in this work, σ is randomly selected from $[0.25, 0.5]$. From the probability density function in (7), the channel gain between PU and SUs is generated to simulate the sensing signals in (1).

Next, the transmitted signal $x(n)$ from the PU is assumed to use BPSK modulation. The number of signal samples during each sensing period is set to 512 (i.e., $N = 512$). The transmitted signal power is assumed to be 1 W. The SNR of the received signal varies between -20 dB and -8 dB. The transmitted signal and noise vectors are generated based on these assumptions.

Finally, the received signal at the SUs is generated using (1). In this experiment, the number of SUs is set to 20 (i.e., $M = 20$). The dimension of the collected signal at FC is (20×512). Note that the channel between SUs and FC is assumed to be perfect.

The received signal vectors are processed to train the model. In particular, the RSS of SUs in each sensing period is determined using (2). The feature matrix X and the adjacent matrix A are determined using (3) and (4), respectively. The dimension of the matrices X and A is 20×1 and 20×20, respectively. These matrices are used as inputs to train the model.

The model is trained and validated using 20,000 and 4,000 data samples, respectively. The test dataset has 2,000 samples. To evaluate exactly the model, the number of received signal samples in two hypothesizes and is the same in each dataset.

4.2. Results

Experiments are implemented in Python 3.9.5, using the Spektral library to design the graph convolutional network and Tensorflow 2.10.1 to build and train the model. The proposed model is trained at a learning rate of 0.0005, with a batch size of 128 and an epoch number of 100. Table 2 shows a list of parameters used for training the GCN-BiLSTM model.

Figures 2, 3 and 4 depict the CSS performance of the GCN-BiLSTM model compared with other models.

In Figure 2, P_d of the CSS models is evaluated versus P_f at an SNR of -14 dB. Note that when the number of false alarms (i.e., according to P_f) is large, the detection threshold γ in (6) is small (Table 1). In other words, a test PU signal is detected easily. Thus, all models achieve higher value of P_d when increasing P_f. ROC

Table 2. Parameters of proposed model

Parameter	Value
No. of units in Conv layer of GCN1	64
No. of units in Conv layer of GCN2	128
Hidden nodes per BiLSTM layer	32
Dense1	32
Dense2	2

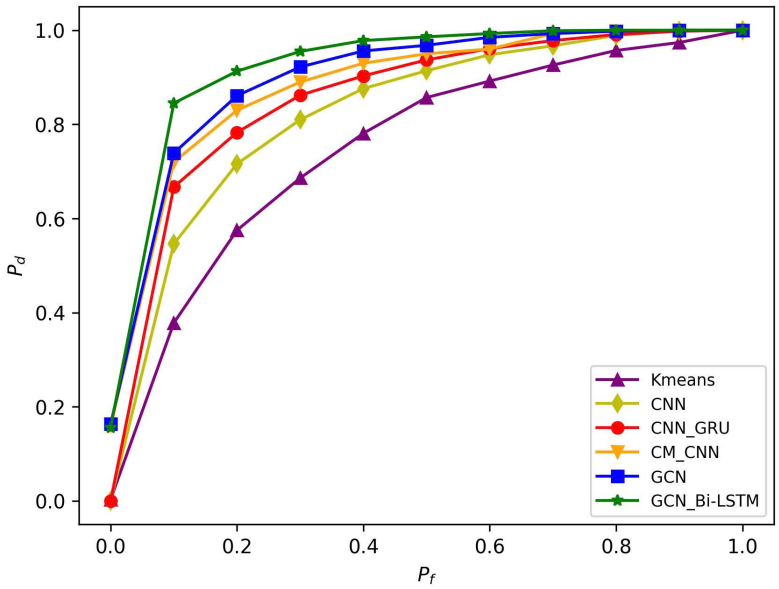

Fig. 2. Receiver Operating Characteristic (ROC) curver at SNR=-14 dB.

curves (i.e., P_d versus P_f) also show that the proposed model (i.e., GCN-BiLSTM model) performs better than all compared models. For example, the GCN-BiLSTM model outperforms over the GCN model (i.e., the best baseline model) about 5% and K-means (i.e., the model achieved the lowest performance) about 30% at $P_f = 0.2$. This can be explained that the proposed model extracts both the spatial and temporal correlations using the GCN and the BiLSTM, respectively. Meanwhile, the GCN model only learns the spatial correlations and ignores the temporal relationship in the received signals, so the performance of the GCN is less than the GCN-BiLSTM.

However, the ROC of the GCN is better than of the remaining models. This shows that GCN is more efficient than CNN in extracting spatial correlations between SUs. Note that although the CNN-GRU learns both spatial-temporal relationships of received signals, the performance is still less than GCN and CM-CNN. A possible reason is given that in the CNN-GRU, the architecture 1D-CNN is used to learn local features of each SU before combing them from all SUs. They do not learn directly the relationship between SUs from sensing signals. In other words, the spatial features in this model are

just discovered at level of single SU. In contrast, the spatial correlations are extracted directly from all SUs in the GCN and CM-CNN models. In particular, the GCN model learns the spatial features from the graph-structured input built from all received signals of SUs. In the CM-CNN model, the architecture 2D-CNN learns the spatial features using CM (i.e., the CM between SUs' signals). The efficiency of extracting the features directly from SUs shows that the GCN and 2D-CNN are better than 1D-CNN, which is applied in CNN and CNN-GRU baselines.

Furthermore, in the proposed model, the BiLSTM is better than GRU in extracting the temporal correlation. Therefore, the relationships of SUs in the proposed model are learned much more efficiently compared with CNN-GRU model. Besides, Figure 2 also depicts that all DL models are much better than the traditional ML model (i.e., the K-means model). This can be explained that the DL models is more efficient than the traditional ML in extracting features of the received signals.

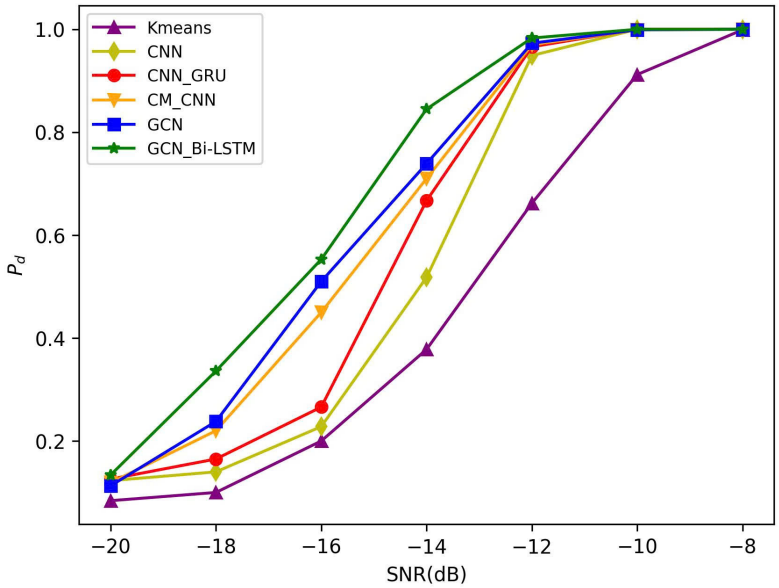

Fig. 3. P_d versus SNR at $P_f = 0.1$.

In Figure 3, P_d is evaluated versus SNR. In the experiment, the value of P_f is chosen at 0.1, following the IEEE 802.22 standard. The SNR ranges from -20 dB to -8 dB. Similar to the ROC curve, the P_d versus SNR of the proposed model outperforms the all compared models. Specifically, the GCN-BiLSTM model performs significantly better than other baselines at low SNRs (e.g., higher than K-means model about 5% and 20% at -20 dB and -18 dB, respectively).

Figure 4 shows that the proposed model outperforms the baselines in accuracy across SNRs. Besides, Figures 3 and 4 shows that at the high SNRs (i.e., -10 dB and -8 dB), all DL models obtain similar performance and approximate to the maximum values. This can be explained that when the SNR is high, the signal is much stronger compared

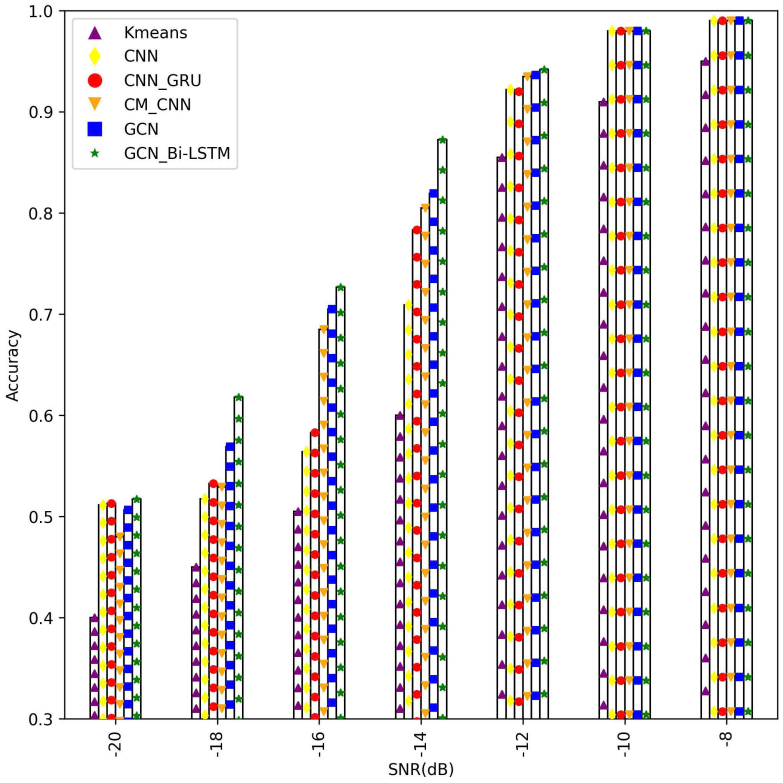

Fig. 4. Accuracy versus SNR at $P_f = 0.1$.

to the noise. Thus, the models are easier to detect the signal of PU from the noise. As a results, the simple models (i.e., CNN, CNN-GRU models) also achieve high performance at these SNRs. With the K-means model (i.e., the model with the lowest performance), the P_d and accuracy also approximate the maximum values and are equal to those of the DL models when SNR equals -8 dB.

The effect of number of SUs on the proposed model's performance is illustrated in Figures 5, 6 and 7. In these experiments, the GCN-BiLSTM model is evaluated at different values of SUs (i.e., 5, 10, 15 and 20). We can observe that when the number of SUs increases, the CSS performance of the proposed model is better. For example, in Figure 6, the P_d obtains 26%, 39%, 48.5% and 55.3% according to the number of SUs at 5, 10, 15 and 20, respectively, at $SNR = $ -18 dB. This can be because when the number of SUs increases, the spatial diversity gain increases and more information about PU is used for training model. Thus, the model decides more precisely the presence of PU. Besides, Figures 6 and 7 also show that the detection performance of the proposed model is improved significantly at the high SNRs when SUs increases. For example, considering the number of SUs at two values of 5 and 20 in Figure 6, the P_d increases only about 3% at $SNR = $ -20 dB while this metric is improved by about 30% at $SNR = $ -14 dB.

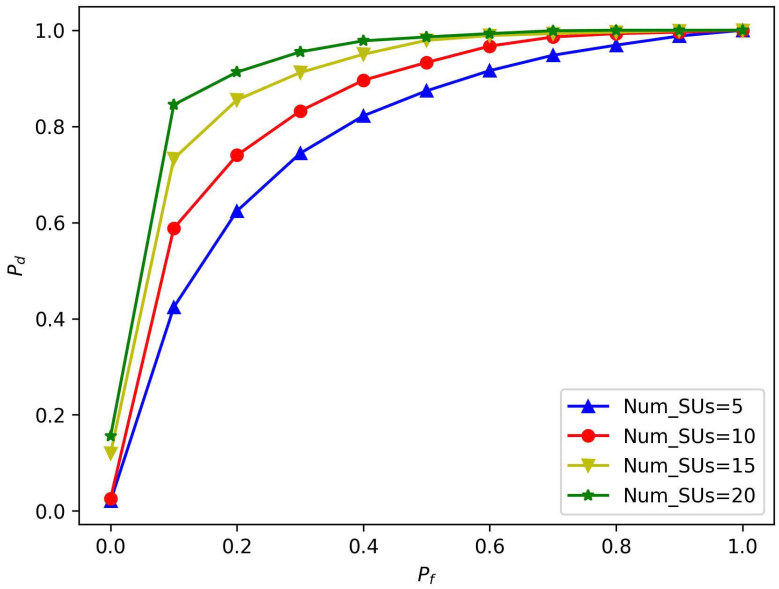

Fig. 5. ROC curve at different Num_SUs.

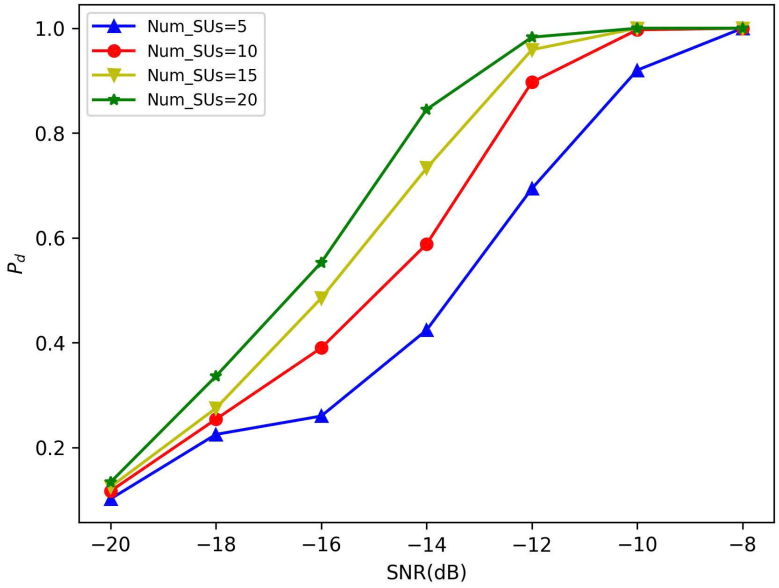

Fig. 6. P_d versus SNR at different Num_SUs.

However, when the number of SUs increases, the model becomes more complex and requires longer training time. Table 3 shows that when the number of SUs increases from 5 to 20, the parameter number of the model rises than 1.5 times (i.e., 7,874 parameters at $M = 5$ and 11,714 parameters at $M = 20$). Besides, in this work, the model is trained using a computer with Intel(R) Core(TM) i7-14700K and NVIDIA

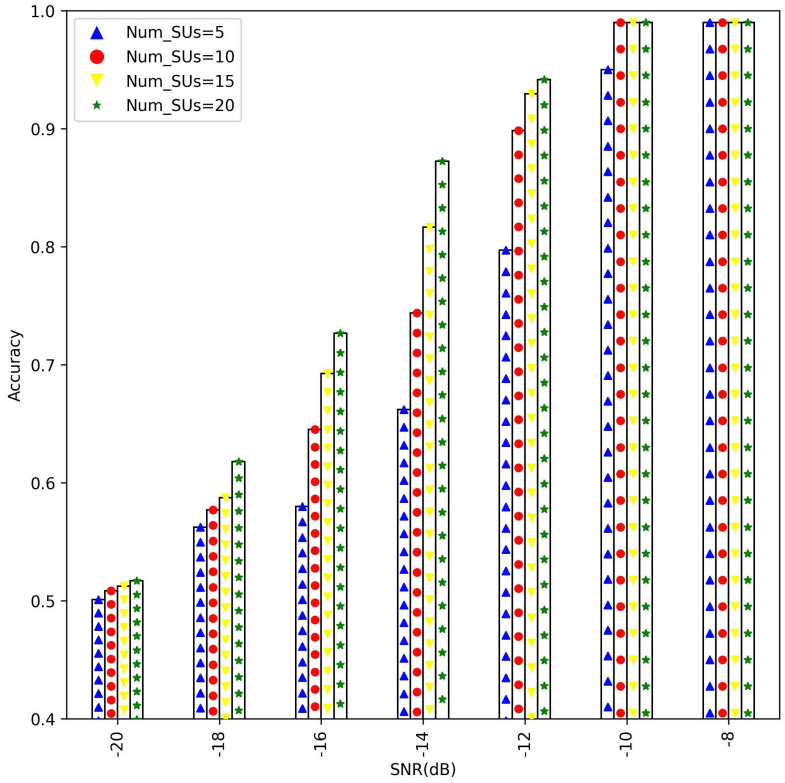

Fig. 7. Accuracy versur SNR at different Num_SUs.

Table 3. Statistic list of the proposed model when increasing the number of SUs

Num_SUs	Training time (seconds)	Total params
5	32.06	7,874
10	32.36	9,154
15	32.55	10,934
20	32.65	11,714

GeForce RTX 4060 Ti graphics cards. According to this hardware configuration, the training time increases by about 0.6 seconds when the number of SUs changes from 5 to 20. However, if the computer with the hardware configuration is limited, the training time can be much longer, evenly it is difficult to train model.

In this work, Table 4 presents the model complexity and training time. In all experiments, the number of SUs is fixed at 20. During training, the same hyperparameters (epoch, batch size, learning rate) are applied to ensure fair comparison among models. Table 4 shows that the GCN-BiLSTM model has less complexity than the CNN and CNN-GRU models. Since the CNN and CNN-GRU models extract local features for each SU, the models need more layers of CNN and

Table 4. Statistic list of the models

Method	Training time (seconds)	Total params
GCN-BiLSTM	32.65	11,714
GCN	15.86	6,850
CM-CNN	23.89	6,916
CNN	196.37	334,666
CNN-GRU	9395.57	366,666

GRU when the number of SUs increases. In contrast, the number of GCN and BiLSTM layers of the proposed model does not depend on the number of SUs. Besides, the training time of the proposed model is also much less than that of the CNN and CNN-GRU models. With GCN and CM-CNN models, they only extract the spatial features and ignore the temporal information, so the models are simpler than the GCN-BiLSTM model (i.e., the parameter number is 6,850, 6,916 and 11,714 according to the GCN, CM-CNN and GCN-BiLSTM models, respectively). However, to improve the detection performance of PU, with the training time only 32.65 seconds, the proposed model is a potential choice in solving CSS problem.

In this work, the proposed model is designed for the CSS problem with one PU. Therefore, the model is unsuitable for detecting spectrum holes in the real scenario with multiple PUs. Besides, a computer with high hardware configuration can be required to train the model and perform the PU detection, especially with the large number of SUs. Furthermore, this work only evaluates the proposed model in the fading channel environment. In the future, the authors plan to design the model for the CSS problem with multiple PUs and evaluate the model in more complex channel environments.

5. Conclusion

In the paper, a model for CSS based on GCN and BiLSTM was proposed. The GCN architecture was applied to extract spatial relationships between SUs. Besides, BiLSTM architecture, which can learn features from both sides of input sequences, was used to capture the temporal correlation of sensing signals at SUs. The extracted spatial-temporal features from GCN and BiLSTM were combined to decide the presence of PU. The proposed model was training using sensing signals at SUs, transmitted from PU over a fading channel. Besides, the effect of the SU number in CRN on the CSS performance was evaluated in this work. The results showed that our model performed better than CNN and CNN-GRU models.

References

[1] I. F. Akyildiz, L. Won-Yeol, M. C. Vuran, S. Mohanty, “Next generation/dynamic spectrum access/cognitive radio wireless networks: A survey,” *Computer Networks*, vol. 50, no. 13, pp. 2127–2159, 2006. DOI: 10.1016/j.comnet.2006.05.001

[2] S. Haykin, "Cognitive radio: Brain-empowered wireless communications," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 2, pp. 201–220, 2005. DOI: 10.1109/JSAC.2004.839380

[3] S. Yoo, P. Sofotasios, S. Cotton, S. Muhaidat, O. Badarneh, G. Karagiannidis, "Entropy and energy detection-based spectrum sensing over F-composite fading channels," *IEEE Transactions on Communications*, vol. 67, no. 7, pp. 4641–4653, 2019. DOI: 10.1109/TCOMM.2019.2900627

[4] F. F. Digham, M.-S. Alouini, M. K. Simon, "On the energy detection of unknown signals over fading channels," *IEEE Transactions on Communications*, vol. 55, no. 1, pp. 21–24, 2007. DOI: 10.1109/TCOMM.2006.887483

[5] I. Alexander, M. Albena, K. Tonchev, V. Poulkov, "Real-time adaptive spectrum sensing for cyclostationary and energy detectors," *IEEE Aerospace and Electronic Systems Magazine*, vol. 33, no. 5-6, pp. 20–33, 2018. DOI: 10.1109/MAES.2018.170098

[6] M. Nouri, H. Behroozi, N. Mallat, S. Aghdam, "A wideband 5G cyclostationary spectrum sensing method by kernel least mean square algorithm for cognitive radio networks," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 7, pp. 2700–2704, 2021. DOI: 10.1109/TCSII.2021.3051087

[7] F. Salahdine, H. Ghazi, N. Kaabouch, W. Fihri, "Matched filter detection with dynamic threshold for cognitive radio networks," in *2015 IEEE International Conference on Wireless Networks and Mobile Communications (WINCOM),* Morocco, 2015, pp. 1–6. DOI: 10.1109/WINCOM.2015.7381345

[8] X. Zhang, R. Chai, F. Gao, "Matched filter based spectrum sensing and power level detection for cognitive radio network," in *2014 IEEE Global Conference on Signal and Information Processing (GlobalSIP),* USA, 2014, pp. 1267–1270. DOI: 10.1109/GlobalSIP.2014.7032326

[9] A. Upadhye, P. Saravanan, S. S. Chandra, S. Gurugopinath, "A survey on machine learning algorithms for applications in cognitive radio networks," in *2021 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT),* India, 2021, pp. 1–6. DOI: 10.1109/CONECCT52877.2021.9622610

[10] S. Khamaysa, A. Halawani, "Cooperative spectrum sensing in cognitive radio networks: A survey on machine learning-based methods," *Journal of Telecommunications and Information Technology*, vol. 3, pp. 36–46, 2020. DOI: 10.26636/jtit.2020.137219

[11] V. Kumar, D. C. Kandpal, M. Jain, R. Gangopadhyay, S. Debnath, "K-mean clustering based cooperative spectrum sensing in generalized k-μ fading channels," in *2016 Twenty Second National Conference on Communication (NCC),* India, 2016, pp. 1–5. DOI: 10.1109/NCC.2016.7561130

[12] N. Balwani, D. K. Patel, B. Soni, M. Lopez-Benitez, "Long short-term memory based spectrum sensing scheme for cognitive radio," in *2019 IEEE 30th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC),* Turkey, 2019, pp. 1–6. DOI: 10.1109/PIMRC.2019.8904422

[13] B. Soni, D. K. Patel, M. López-Benítez, "Long short-term memory based spectrum sensing scheme for cognitive radio using primary activity statistics," *IEEE Access*, vol. 8, pp. 97 437–97 451, 2020. DOI: 10.1109/ACCESS.2020.2995633

[14] L. Chang, L. Xuemeng, L. Ying-Chang, "Deep CNN for spectrum sensing in cognitive radio," in *2019 IEEE International Conference on Communications (ICC),* China, 2019, pp. 1–6. DOI: 10.1109/ICC.2019.8761360

[15] L. Chang, W. Jie, L. Xuemeng, L. Ying-Chang, "Deep CM-CNN for spectrum sensing in cognitive radio," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 10, pp. 2306–2321, 2019. DOI: 10.1109/JSAC.2019.2933892

[16] X. Mingdong, Y. Zhendong, Z. Yanlong, W. Zhilu, "Cooperative spectrum sensing based on multi-features combination network in cognitive radio network," *Entropy*, vol. 24, no. 1, p. 129, 2022. DOI: 10.3390/e24010129

[17] D. Janu, S. Kumar, K. Singh, "A graph convolution network based adaptive cooperative spectrum sensing in cognitive radio network," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 2, pp. 2269–2279, 2023. DOI: 10.1109/TVT.2022.3214348

[18] Z. Si, T. Hanghang, X. Jiejun, M. Ross, "Graph convolutional networks: a comprehensive review," *Computational Social Networks*, vol. 6, no. 1, pp. 1–23, 2019. DOI: 10.1186/s40649-019-0069-y

[19] T. N. Kipf, M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proceedings of the International Conference on Learning Representations (ICLR),* France, 2016. DOI: 10.48550/arXiv.1609.02907

[20] C. Wei-Lin, L. Xuanqing, S. Si, L. Yang, B. Samy, H. Cho-Jui, "Cluster-GCN: An efficient algorithm for training deep and large graph convolutional networks," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining,* USA, 2019, pp. 257–266. DOI: 10.1145/3292500.3330925

Manuscript received 04-11-2024; Accepted 15-6-2025.

■

Doi Thi Lan received the B.S. degree in Information and Communication Engineering from Le Quy Don Technical University, Vietnam, in 2013. She then obtained the M.S. degree in Electrical and Electronic Engineering from the same university in 2018. In 2024, she took the Ph.D. degree in Electrical, Electronic and Computer Engineering from the University of Ulsan, South Korea. She is currently serving as a lecturer in the Faculty of Radio and Electronic Engineering, Le Quy Don Technical University, Vietnam. Her current research interests include machine learning, deep learning, anomaly detection, and their applications in communication networks. E-mail: doilan@lqdtu.edu.vn

TÁCH CÁC ĐẶC ĐIỂM KHÔNG GIAN-THỜI GIAN SỬ DỤNG HỌC SÂU TRONG CẢM BIẾN PHỔ HỢP TÁC

Đới Thị Lan

Tóm tắt

Trong các hệ thống vô tuyến nhận thức, cảm biến phổ đóng vai trò quan trọng trong việc phát hiện sự có mặt của người dùng sơ cấp (PU). Trong công trình này, một mô hình cảm biến phổ hợp tác (CSS) dựa trên các kiến trúc GCN và BiLSTM (GCN-BiLSTM) được đề xuất. Cụ thể, kiến trúc GCN được áp dụng để trích xuất mối quan hệ giữa những người dùng thứ cấp (SUs). Bên cạnh đó, kiến trúc BiLSTM học mối tương quan thời gian của thông tin cảm biến tại các SUs. Sự hiện diện của PU được quyết định dựa trên các đặc điểm không gian-thời gian, được kết hợp từ đầu ra của các mạng GCN và BiLSTM. Mô hình đề xuất được đánh giá trong một kịch bản kênh truyền động (tức là kênh truyền fading). Kết quả thử nghiệm cho thấy mô hình GCN-BiLSTM có chất lượng cảm biến phổ hợp tác tốt hơn so với các mô hình CNN và CNN-GRU.

Từ khóa

Mạng vô tuyến nhận thức; cảm biến phổ hợp tác; GCN; BiLSTM.