Classifying Wi-Fi from Raw Power Measurements using a Neural Network Adapted from WaveNet

Robert J. Estes
School of Electrical and Computer Engineering
The University of Oklahoma
Norman, USA
robert.j.estes@ou.edu

Derrick Willis
School of Electrical and Computer Engineering
The University of Oklahoma
Norman, USA
derrick.willis@ou.edu

Hazem H. Refai
School of Electrical and Computer Engineering
The University of Oklahoma
Norman, USA
hazem@ou.edu

Abstract—Research into identifying coexisting wireless devices and technologies has been growing in importance as the number of wireless devices increase every year. While many various methods have been proposed over the years, most of them usually rely on using features from the frequency domain. These can usually be distinguishable among different wireless standards. The feature space in the frequency domain is usually limited but less complex than those in the time domain. However, when trying to identify wireless devices that use different specifications of the same Wi-Fi standard, utilizing the frequency domain feature is difficult as the features are too similar between specifications.

In this paper, we investigate utilizing a neural network to identify the Wi-Fi standard using raw power measurements. More specifically, we adapt the WaveNet model to take advantage of its capability to handle time series data with thousands of timesteps, an advantage that both recurrent neural networks (RNN) and long short-term memory (LSTM) networks do not share. Wi-Fi signals are collected across versions 802.11n, 802.11ac, and 802.11ax both individually and with multiple standards coexisting across a range of different throughputs. The data is then pre-processed and used to train a neural network adapted from the WaveNet model. Results indicate that high accuracy detection can be achieved by utilizing this method.

Keywords—Deep Neural Network, Dilated CNN, Time Series Analysis, WaveNet, Wi-Fi Identification

I. INTRODUCTION

Concerns about coexistence between wireless devices and technologies has been growing as new devices and technologies are developed that use the same ISM band. In the age of Internet of Things (IoT), virtual assistance technology is more commonplace, and many household devices and appliances are being redesigned to be “smart” by adding connectivity to them adding to the growing concerns. Spectrum congestion is another concern related to the growing numbers which resulted in moving Wi-Fi communication to 5 GHz from 2.4 GHz. With this growth in the number of wireless devices and technologies, research into coexistence has been growing as well.

In this work, we will investigate adapting a WaveNet architecture to use in the process of classifying coexisting wireless devices that are using different Wi-Fi specifications.

Specifically, this work will focus on classifying individual for base line technology features and coexisting cases of wireless devices: 802.11n, 802.11ac, and 802.11ax—operating on the 2.4 GHz band while processing raw power measurements obtained while the devices are operating in a shared environment.

II. RELATED WORK

Past work in using a neural network to identify coexisting wireless technologies has been done by utilizing features primarily from the frequency domain. A convolutional neural network (CNN) was used in [4] to identify cases of individual and coexisting wireless technologies 802.11n, Bluetooth, and Zigbee. Raw power measurements were recorded across the 80 MHz spectrum at various SNRs and used to train the CNN which achieved an accuracy of 93% with the highest SNR. Their work shows that the spectrograms for ZigBee, Wi-Fi, and Bluetooth are unique and distinguishable between these standards such that the CNN they proposed achieved high accuracy. A similar method for identifying coexisting Wi-Fi specifications would be difficult as spectrograms for the Wi-Fi specifications 802.11ac and 802.11ax would be similar to the spectrogram for 802.11n shown in [4].

When dealing with time series data, it is common to use either RNN or LSTM architectures for neural networks. However, neither of these networks handle data with thousands of timesteps very well due, to the issue of vanishing gradients after ~50 timesteps for RNNs and ~500 timesteps for LSTMs. As detailed in [1], WaveNet was initially designed as a generative model for speech synthesis. Audio samples being 16 kHz means that the model was designed to handle sequenced data of 16000 samples per second, much larger than anything RNN or LSTM networks can handle. Work in [6] indicates that the model architecture can also be adapted to work as a classifier. They used a network architecture based on WaveNet for artist classification of audio samples. Their architecture essentially encoded the 16 kHz audio sample to a feature space of (16000, 40) that was fed into the classifier portion of the network which contained pooling and convolution layers to down sample the encoded signal before using a softmax layer to classify the sample as 1 of 20 artists. Utilizing this method, they were able to achieve a F1 score of 0.854 which they indicated was better than current state-of-the-art methods employed for
artist classification. This study, to the best of authors' knowledge, is the first use of WaveNet for classifying wireless technologies in a coexistence analysis.

III. METHODOLOGY

A. Devices

The Asus RT-AX88U Router is 802.11ax compliant and was released in October 2018. It features Orthogonal Frequency-Division Multiple Access (OFDMA) and 4x4 Multiple User Multiple-Input Multiple-Output (MU-MIMO) technology that contributes to efficiency in traffic-dense heterogeneous and homogeneous environments. It supports single-carrier data rate speeds of 4.8 Gbps on the 5 GHz band with 160 MHz of channel bandwidth. Powered by the BCM49408 chipset, it is supplied by Broadcom for wave 2 wireless applications.

The MikroTik RouterBOARD 953GS-5HnT-RP provides a dual band (802.11ac/n) Wi-Fi solution with 3x3 MIMO Triple Chain support. It also shares wave 2 wireless application like the Asus RT-AX88U router. The Qualcomm Atheros QCA9558 chipset is capable of peak speeds of 1.7 Gbps and has channel bandwidth of up to 40 MHz. Radio card R11e-2HPnD was used to specifically generate 802.11ac while 802.11n was broadcasted through onboard capability of the router boards.

The NI PXIe-5644R is a reconfigurable 6 GHz RF vector signal transceiver (VST) that was configured on a complete PXI platform. The platform includes the NI PXIe-8133 1.73 GHz Quad-Core PXI Express Embedded Controller, NI PXIe-1082 8-Slot 3U PXI Express Chassis and VERT2450 Antenna. The VST is FPGA-based and capable of real-time signal processing and control. The process of measuring the Radio Frequency (RF) is broken into acquiring I/Q samples, transferring I/Q samples into a host PC, and performing a proprietary measurement algorithm based on LabVIEW.

The Netgear Gigabit Switch GS116 facilitates remote connection between NI PXIe-5644R platform, MikroTik router boards, and their host laptop Toshiba Portege R835-P56X. A Dell Precision 5540 was ethernet wire connected to the Asus RT-AX88U router for data logging purposes and functioned as the server through the Iperf3 network performance measurement software tool. A final laptop, Lenovo ThinkPad T470s, is equipped with the Killer Wi-Fi 6 AX1650 module and communicates on the 802.11ax standard. It is used in conjunction with Iperf3 to test network traffic with the Asus RT-AX88U router.

B. Data Collection

Tests were performed in Building 5 blockhouse at the Oklahoma University campus in Tulsa. We first baselined the ambient noise to ensure no interfering signals were present. This was re-affirmed by daily testing and ambient noise remained minimum and constant of less than -73 dBm. Three separate Wi-Fi networks were configured to operate 802.11n, 802.11ac, and 802.11ax wireless devices.

Each network had a pair of devices that acted as access point (Tx) and station (Rx) for their respective networks. Equipment was placed on wood tables elevated at a height of 2.5 ft and separated at a prefixed distance. Baseline testing was conducted and evaluated for maximum throughput performance using the Transmission Control Protocol (TCP) with downlink transmissions. The experimental setup and device layout are shown in Fig. 1 and the labeled devices with their interface (IF), operating system (OS) and iPerf version are detailed in Table I.

![WiFi Classifier Network Structure based on WaveNet](image)

| ID | Device | IF    | OS     | iPerf |
|----|--------|-------|--------|-------|
| 1  | NI PXIe-5644R | Ethernet | Windows 7 | -     |
| 2  | Asus RT-AX88U | WLAN   | Windows 10 | 3.1.3 |
| 3  | MikroTik RouterBOARD 953GS-5HnT-RP | WLAN | RouterOS | -     |
| 4  | Toshiba R835-P56X | Ethernet | Windows 10 | -     |
| 5  | Dell Precision 5540 | Ethernet | Windows 10 | -     |
| 6  | Lenovo ThinkPad T470s | WLAN | Windows 10 | 3.1.3 |

802.11ax, 802.11ac, and 802.11n was tested five times and achieved a maximum throughput of 956 Mbps, 340 Mbps, and 250 Mbps, respectively. This maximum data rate was then
divided equally into five limits that were used to artificially limit the throughput through the iPerf3 software tool. These limits allowed us to simulate signal quality with lower throughputs simulating low quality and higher throughputs simulating high quality. The tests for each of the limits was repeated 5 times. Additional coexistence tests were conducted between 802.11ax with 802.11ac and 802.11ax with 802.11n and repeated 5 times for each possible combination of limited throughputs. The limits are detailed in Table II.

| Specification | Limits (Mbps) |
|---------------|--------------|
| 802.11ax      | 190.0 380.0 570.0 760.0 956.0 |
| 802.11ac      | 68.0 136.0 204.0 272.0 340.0 |
| 802.11n       | 50.0 100.0 150.0 200.0 250.0 |

The Asus RT-AX88U generated the 802.11ax network with channel bonding enabled to allow full channel utilization of 160 MHz width. Network traffic was set to wireless local area network (WLAN) channel 36 of the 5 GHz ISM Band with a center frequency of 5180 MHz. Both pairs of MikroTik router boards had channel bonding enabled and generated the 802.11n and 802.11ac networks with 40 MHz width of channel utilization. They were configured to operate on the same channel, ISM band, and share the center frequency with the Asus RT-AX88U.

The PXIe-5644R used a proprietary LabVIEW software [5] to extract features of these measurements that included the raw power value, and time domain activity. Power values were expressed as In-Phase and Quadrature (IQ) components with duty cycle extracted from the time domain. Settings of the proprietary LabVIEW tool during testing included an IQ sampling rate of 10 MS/s of real-time bandwidth analyzation. All testing proceeded for a duration of 60 seconds and produced 72,600 samples per frame. The threshold was set to -59 dBm which was at least 5 dBm above the noise floor measured in the testing environment for accurate classification.

**C. Data Preprocessing**

A total of 325 tests were performed resulting in the same number of binary files to process. The files totaled in size of approximately 319 GB. Each binary file contained around 1.054 billion interleaved IQ components of raw power measurements in dBm. The interleaved data was separated into the individual components each totaling around 527 million data points. The beginning and end of the I component was trimmed to range 100 to 500 million and the Q component was discarded as it was not used for classification. Random samples of length 15,000 were then taken with random space in between each sample. Classes were defined as 1 to 5 indicating 802.11ax, 802.11ac, 802.11n, 802.11ax with 802.11ac, and 802.11ax with 802.11n, respectfully. The number of samples taken from each was set to ensure that we ended up with a total of 75,000 samples of length 15,000 that was perfectly balanced at 15,000 samples for each class. Furthermore, the 15,000 samples were taken in a way that ensured the limits were balanced as well.

Normalization was done in which the signal characteristics were maintained. Each sample was offset by 59.0 dBm which was determined by the threshold used during collection. The samples were then divided by the max value determined to be 85.0 dBm which scaled the data between -1.0 and 1.0. Fig. 2 compares a random sample of the I-component of the raw power measurements taken for coexisting 802.11ax and 802.11n both before and after normalization. It can clearly be seen that the characteristics of the signal was preserved after we performed the aforementioned normalization. Fig. 3 shows a random sample of the I-component of the raw power measurements taken during each test for every throughput limit after normalization in order to show the characteristics of the captured signals. Signal characteristics confirms the difficulty in distinguishing between them in the time domain. The size of the sampled data, after normalization, totaled approximately 17 GB, which was used for training.
IV. NEURAL NETWORK

A network structure was developed that consisted of an “encoding” portion and a “classifier” portion. The encoding portion was based on WaveNet and encoded the raw power signal samples. The classifier portion consisted of dense layers combined with softmax to classify the samples as 1 of 5 classes. Raw power data consisted of In-Phase and Quadrature (IQ) components of which only In-Phase was used to train the network. No further preprocessing besides that which is described in Section III-A was done. Label data consisted of integer values in the range of 1-5 to identify the sample as 802.11ax, 802.11ac, 802.11n, 802.11ax with 802.11ac, and 802.11ax with 802.11n respectively. These labels were one-hot encoded before being used to train the network. Fig. 4 depicts the layer structure of the developed network with output sizes indicated.

**A. Network Layers**

The WaveNet layers consist of a causal convolution followed by a series of stacked residual blocks. The residual block consists of a dilated convolution in which the ReLu activation function is replaced with a multiplicative combination of sigmoid and tanh functions. The sigmoid acts as a forget gate since it outputs a value between 0 and 1 while the tanh acts as an activation. The result of this dilated convolution is both fed out of the block through a skip connection and added to the residual which is then fed to the next block. The skip connections are added together before being fed into a final convolution layer with ReLu activations before and after this final convolution. At this point the network enters the classifier portion and the WaveNet portion ends. The size of the output from the residual blocks is dependent on the number of timesteps and the number of filters used in the convolutions which is a hyperparameter. The convolution layer before and after the residual block each has a kernel size whose value is a hyperparameter and is referenced as $K_1$ and $K_2$ in Table III.

The number of residual blocks is dependent on the dilation depth which is a hyperparameter. The dilation depth will directly dictate the width of the receptive field. The receptive field for dilated convolution grows exponentially as more layers are added while causal convolution grows linearly. For example, a dilation depth of 3 results in a receptive field of 8 while a causal convolution with 3 layers results in a receptive field of only 4 as shown in Fig. 5. Furthermore, varying the dilation depth from 3 to 10 results in receptive field increases of 16, 32, 64, 128, 256, 512, and 1024 respectfully. So, with a dilation depth of 10 each encoded timestep can take into consideration 1024 previous timesteps. If causal convolution were used then to achieve the same 1024 width receptive field, 1023 convolution layers would be needed.

**B. Classifier Layers**

The classifier layers consist of an average pooling layer followed by a flatten layer, 2 dense layers, and a softmax layer. The average pooling layer is used to decrease the dimensionality of the encoded signal. It will have a pool size equal to the number of filters specified and a stride, which is a hyperparameter, that determines by what factor the feature dimensions will be reduced. For example, if the encoded signal is of size (15000, 40) where 15000 is the number of timesteps and 40 is the number of filters specified in the convolutions, then a stride of 50 will reduce this to size (300, 40). The flatten layer flattens the input to 1 dimension before feeding into the dense layers. Continuing the example above, the size (300, 40) would be flattened to size (12000, ). The 2 dense layers are fully

---

**Fig. 4. WiFi Classifier Network Structure based on WaveNet**

**Fig. 5. Receptive Field Width for Dilated versus Causal Convolution**
connected layers whose sizes are hyperparameters and are references as $D_1$ and $D_2$ in Table III. The layers use ReLu activation and He uniform variance scaling as the kernel initializer. The softmax layer is a fully connected layer whose size is equal to the number of classes and activation is softmax. It will output a vector which is the class probability distribution of the sample.

### B. Optimization and Network Parameters

For the optimization algorithm, we chose to use Adam. This optimization has four additional parameters which include $\alpha$, $\beta_1$, $\beta_2$, and $\epsilon$. The learning rate is $\alpha$ and sets the step size of each iteration during optimization. $\beta_1$ and $\beta_2$ are the exponential decay rates for the first and second moment, respectively. Lastly, $\epsilon$ is as small constant introduced for stability. For loss calculation, categorical cross entropy was used. Epochs and batch size are additional parameters. Epochs tells the network how many times to iterate over the entire training dataset and batch size tells the network how many samples to process before updating the model. The values used for each parameter described here and in Section IV-A is detailed in Table III.

| Parameter   | Value   | Description                             |
|-------------|---------|-----------------------------------------|
| Depth       | 10      | Number of stacked residual blocks.      |
| Filters     | 40      | Number of convolution filters.          |
| $K_1$       | 2       | Kernel size for first causal convolution.|
| $K_2$       | 50      | Kernel size for last convolution after add skips. |
| Stride      | 50      | Average pooling stride size.            |
| $D_1$       | 4000    | Size of 1st fully connected dense layer.|
| $D_2$       | 40      | Size of 2nd fully connected dense layer.|
| $\alpha$    | 0.00001 | Learning rate for Adam optimizer.       |
| $\beta_1$   | 0.9     | Decay rate for 1st moment in Adam optimizer. |
| $\beta_2$   | 0.999   | Decay rate for 2nd moment in Adam optimizer. |
| $\epsilon$  | 0.000001| Stability constant for Adam optimizer.   |
| Epochs      | 10      | Number of passes over data for training.|
| Batch       | 20      | Number of samples to process before update. |

### V. RESULTS

Training ran for a total of 10 epochs which took approximately 14 hours with a sample size of 60,000 samples. Prediction was performed on a validation set of size 15,000 and took approximately 6 minutes to run. One should note that the prediction time is not linear as a prediction of 1 sample takes around 0.4 seconds which indicates that the network is efficient at predicting large amounts of data at once.

Table IV details the training loss, training accuracy, validation loss, and validation accuracy for each epoch. The network was able to achieve a top training accuracy of 97.41% and a top validation accuracy of 95.39%. Most of the performance increase was seen early in the training in the first 4 epochs. After which, training accuracy stayed around or below approximately 95%. The small increases in validation accuracy and small decreases in validation loss after epoch 6 indicates that the training has plateaued; hence continued training would have little to no positive effect on validation accuracy. If additional performance improvement (above 95.39%) is desired, further parameter tuning or changes to the network structure.

| Epoch | Accuracy | Loss | Val. Accuracy | Val. Loss |
|-------|----------|------|---------------|----------|
| 0     | 0.79188  | 0.52341 | 0.84840       | 0.36957  |
| 1     | 0.88322  | 0.31179 | 0.88507       | 0.30241  |
| 2     | 0.91300  | 0.24227 | 0.92807       | 0.22684  |
| 3     | 0.93300  | 0.19104 | 0.92340       | 0.20608  |
| 4     | 0.94415  | 0.15750 | 0.94453       | 0.16166  |
| 5     | 0.95585  | 0.12745 | 0.92987       | 0.19309  |
| 6     | 0.96035  | 0.11201 | 0.95393       | 0.13832  |
| 7     | 0.96647  | 0.09618 | 0.94980       | 0.13700  |
| 8     | 0.96957  | 0.08865 | 0.95240       | 0.14093  |
| 9     | 0.97413  | 0.07582 | 0.95133       | 0.14125  |

![Confusion Matrix for Validation Data Set of 15,000 Samples](image-url) Fig. 6. Confusion Matrix for Validation Data Set of 15,000 Samples

### VI. CONCLUSION

In this work, we developed a neural network adapted from WaveNet employed to classify samples from the In-Phase component of raw power measurements based on the Wi-Fi specification that the devices were using. WaveNet was originally designed for speech synthesis. While previous work [6] used an adapted WaveNet for classification, this paper is the first use for classifying wireless technologies in a coexistence analysis. Results show that using this type of neural network performs extremely well at discerning individual and coexisting technologies.
cases of devices using different Wi-Fi specifications across various throughputs.

While this study covered multiple cases of coexistence, the network still needs to be trained on additional cases. Future work will include adding in the final cases of coexistence which include 802.11ax with 802.11n, and 802.11ax with 802.11ac with 802.11n. It is expected that the current network would have similar performance for the case of 802.11ax with 802.11n. For the case of 802.11ax with 802.11ac with 802.11n, which has 3 technologies coexisting, it is expected that the accuracy would drop below 90%. Network optimization will also be looked at to try and decrease complexity and number of learning parameters which could include changes to the network structure as well as further tuning of the hyperparameters. This optimization would have the goal of increasing prediction accuracy, especially in the cases of coexistence where the accuracy was around 90%, and decreasing training and prediction times. In fact, one shortcoming of this network is the time it takes to predict one sample. At 0.4 seconds, which is rather slow in digital terms, integrating this network into a pipeline which requires predictions at a much faster rate would not be possible as it stands for near real-time prediction. Benchmarking the proposed network against other machine learning algorithms and other deep learning networks is planned. The network will also need to be tested in environments with coexisting technologies where one or more of those was not used to train the network. This would allow us to determine the performance of the network when presented with data samples captured in different environments. Additionally, other wireless technologies besides the 802.11 family, like Bluetooth, could be added in future work, although similar performance would be expected with performance decreasing as the number of coexisting technologies increases above two.

VII. ACKNOWLEDGMENT

The authors would like to thank Mr. Alex Fortney—performed a summer internship with Dr. Refai—for collecting part of the data used for this study. Mr. Fortney is currently an Electrical and Computer Engineering student at the MST in Rolla, MO.

REFERENCES

[1] A. Oord, et al, “WaveNet: A Generative Model for Raw Audio,” 2016, arXiv:1609.03499.
[2] E. D. Ngangue Ndih and S. Cherkaoui, "On Enhancing Technology Coexistence in the IoT Era: ZigBee and 802.11 Case," in IEEE Access, vol. 4, pp. 1835-1844, 2016, doi: 10.1109/ACCESS.2016.2553150.
[3] E. D. Ngangue Ndih, S. Cherkaoui and I. Dayoub, "Analytic Modeling of the Coexistence of IEEE 802.15.4 and IEEE 802.11 in Saturation Conditions," in IEEE Communications Letters, vol. 19, no. 11, pp. 1981-1984, Nov. 2015, doi: 10.1109/LCOMM.2015.2449298.
[4] N. Bitar, S. Muhammad and H. H. Refai, "Wireless technology identification using deep Convolutional Neural Networks," 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Montreal, QC, 2017, pp. 1-6, doi: 10.1109/PIMRC.2017.8292183.
[5] W. Balid, M. O. Al Kalaa, S. Rajab, T. Tafish and H. H. Refai, "Development of measurement techniques and tools for coexistence testing of wireless medical devices," 2016 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), Doha, 2016, pp. 449-454, doi: 10.1109/WCNCW.2016.7552741.
[6] X. Zhang, Y. Gao, Y. Yu, and W. Li, “Music Artist Classification with WaveNet Classifier for Raw Waveform Audio Data,” 2020, arXiv: 2004.043