Identification of wireless devices in the 2.4GHz band using machine learning

Luuk Vink

***Abstract*—With the increasing popularity of wireless devices, it has become increasingly important to monitor the electromagnetic spectrum and enforce rules on the use of it. The radio communications agency of the Netherlands, Rijksinspectie Digitale Infrastructuur, is responsible for monitoring the frequency spectrum and ensuring its proper use. Processing the data measured by the measurement setups spread throughout the Netherlands is labour-intensive which is why there are machine-learning models in place for processing data to detect outlandish signals, reducing the need for human intervention.**

Further analysis of the 2.4GHz unlicensed band is requested, with focus on detecting specific devices transmitting at this frequency. This can later be used for statistical purposes or to enforce regulations. The analysis will be done using supervised machine learning due to its proven effectiveness and faster data processing capabilities compared to traditional signal processing methods.

This paper explores the use of the Random Forest model and the Gradient Boosted Trees model for detecting specific devices in the 2.4 GHz unlicensed band. First, a dataset will be made, consisting of spectrograms of WiFi, Bluetooth, and Fixed carrier (Babyphone) signals. The dataset is later expanded upon with several combinations of the three signals. Next, a Random Forest and Gradient Boosted Trees model is trained, compared, and shown to be effective. After the expansion of the dataset, the models are retrained and shown to be working. Finally, a real-world data sample is input to both models where it is shown that they correctly identify the specific signal. The best-performing algorithm is the Gradient Boosted Trees model, achieving an accuracy of 97.3%.

I. Introduction

THE usage of the radio frequency (RF) spectrum keeps growing. With a growing number of wireless devices, more supervision is necessary to ensure compatibility between devices and allow for a functioning wireless spectrum. The Radio Communications Agency of the Netherlands, Rijksinspectie Digitale Infrastructuur (RDI), is responsible for monitoring the RF spectrum and enforcing the legislation around wireless communications. This monitoring is done by utilising several RF spectrum measuring setups in the Netherlands, with nine in urban areas and six in the countryside. [1] These measuring setups generate a lot of data on a daily basis, which has to be processed. RDI used to have people manually processing the data, with employees examining the generated data and drawing conclusions, which is inefficient, costly, and prone to error. Research into using machine learning for processing the measurements has been conducted and is shown to work, [2] [3] reducing the workload for people at RDI.

The 2.4GHz band, which is unlicensed but not unregulated, is used for daily household electronics, like a WiFi router, or Bluetooth devices. Even though the 2.4GHz band is unlicensed, RDI is still interested in this band as it is not entirely unregulated, with heavier monitoring and more restrictions being put on the band in case of large events due to economic interests. When processing this band, RDI is interested in identifying the type of device, such as a WiFi access point or a baby monitor, in addition to ensuring the legality of the signals. This information allows for statistical analysis and makes it easier to localize possible rule breakers.

Previous research shows that it is possible to get meaningful results from a spectrogram using supervised machine learning. [2][3] These results were obtained using real-world measurement data from the measurement setups operated by RDI, which will also be the case in this study. A different paper explores the identification of WiFi and Bluetooth devices using unsupervised machine learning showing that it is possible to identify devices based on their RF presence. [4]

The authors of [1][2][3] focused on using supervised machine learning or signal processing for detecting outlandish signals with the measurement data at RDI. However, more research is needed to show the feasibility of supervised machine learning for identifying specific devices which are transmitting in the 2.4GHz band, specifically using spectrograms obtained by measurements of RDI. This technology can later be used on a large scale for statistical analysis and tracking down illegal use of the spectrum.

This research explores the possibility of using supervised machine learning to identify different types of wireless devices. Supervised machine learning requires a labelled dataset which is as close as possible to real-world measurements. The devices considered in this research are WiFi routers, Bluetooth devices and Babyphones. The RF signals of these had to be simulated accurately to enable the creation of a comprehensive dataset. The machine learning algorithms used are Random Forest and Gradient Boosted Trees, which are decision tree algorithms. Their performance will be tested and compared.

A. Research question

The main research question which will be investigated is:

What accuracy can be achieved when identifying different types of wireless devices using supervised machine learning in the 2.4GHz band?

II. Method

A. Channel

All signals pass through a channel when travelling from transmitter to receiver. A generic wireless communication setup can be viewed in fig. 1. This paper considers the channel to be an additive white Gaussian noise (AWGN) channel. Other influences such as multipath propagation have

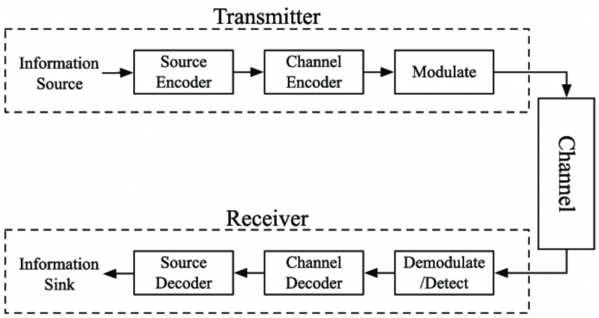


Fig. 1: Generic wireless communication setup
Source: From [5]

been disregarded to reduce computation times and dataset complexity.

In [6], the noise floor in the 2.4GHz band was measured and found to be well above the thermal noise floor for the receiver used (-111dBm), with an average received power between -78 and -101 dBm. General power levels at the receiver for Bluetooth and WiFi are between -30 and -100 dBm. [7][8] Power levels for a fixed carrier can vary more, but it is a safe assumption that they can be in the same range as WiFi and Bluetooth. Using these values, a typical Signal-to-Noise Ratio (SNR) ranging from -22 (no discernible signal) to an absolute maximum of 71 dB (very strong signal) can be utilized for dataset creation.

All the generated waveforms will be passed through the AWGN channel to realise a specific SNR, varying throughout the dataset to encompass the full possible range.

B. Generation of the spectrogram

The generation of a spectrogram is done by taking the short-time Fourier transform (STFT) of a signal sampled in time. The STFT in this paper uses a segment length of 256 with an overlap of 128. A Hanning window is used to reduce spectral leakage.

RDI collects spectrum data at each measurement location, containing the signal strength sampled every minute. [1] These measurements combined result in a spectrogram, which will be saved to the servers for later analysis. Using a spectrogram as the input for the machine learning algorithms allows for minimal preprocessing, speeding up computation times and reducing complexity.

C. Dataset Creation

First, the dataset should be created. To do this, the relevant signals have to be simulated accurately.

1) Fixed Carrier: For the fixed carrier (babyphone), both digital modulation and frequency modulation (FM) were chosen. The digital modulation uses one of three chosen modulation schemes: BPSK, QPSK or 16-QAM. The constellation points for all modulation schemes were normalised such that the average energy per bit is 1. This was done to ensure comparable signals. The data sent for the duration of the signal is randomly generated as we do not care about the actual contents, but mainly about the frequency usage.

The FM signal uses an arbitrary input waveform to modulate a carrier. In this case, a chirp signal was chosen.

2) WiFi: WiFi is simulated according to the 802.11g standard with a modulation coding scheme (MCS) index of 9. This is an OFDM signal with 16-QAM modulation, utilising 52 OFDM subcarriers with a spacing of 0.3125 MHz between subcarriers. Furthermore, after each packet, a short pause is required. The pause is called the short interframe spacing (SIFS) which is 10 μs for the used standard.

3) Bluetooth: Bluetooth operates in the frequency range from 2400 MHz to 2483.5 MHz. This frequency span is partitioned into 79 frequency channels, spaced 1 MHz apart. Each channel n, where $0 \leq n \leq 78$ and $n \in \mathbb{Z}$, is modulated with a carrier frequency of $2402 + n$ MHz. [9]

In this paper, the original 'Basic Rate' modulation scheme is used, which uses Gaussian frequency shift keying (GFSK) modulation with a raw data rate of 1 MBit/sec. The frequency deviation for a logic '1' and a logic '0' is nominally +160 KHz and -160 KHz respectively. This frequency deviation has to be at least ± 115 KHz.

Furthermore, Bluetooth has built-in pseudo-random frequency hopping to avoid interference. There are 1600 time slots per second, allocating 625 μs to each slot, after which the Bluetooth signal can hop. Each packet is transmitted on a different frequency channel with a single data packet using between 1 and 5 slots.

4) Noise: Finally, there should also be a case where no device is sending. In this case, only noise is a factor. Noise will be simulated by only considering the AWGN channel and reading the output (simulated received side).

D. Initial Model

After the creation of the initial dataset, models need to be trained. The chosen models are Random Forest and Gradient Boosted Trees. The decision to go for decision trees is due to previous research finding success in the usage of these methods with spectrogram data. [3]

Random Forest and Gradient Boosted Trees are both ensemble learning methods that leverage multiple decision trees to improve predictive accuracy. A decision tree consists of a series of binary tests that partition data into subsets based on feature values, culminating in leaf nodes that provide the final prediction. For instance, a test might ask, "Are there more than 10 coins in his wallet?", branching the decision process based on 'yes' or 'no' answers.

Random Forest employs two key techniques: bagging and feature randomness. Bagging trains each tree on a different subset of the data, sampled with replacement, while feature randomness considers only a random subset of features at each split. These techniques reduce variance and improve model robustness. Random Forest is inherently parallelizable, enabling faster training, and providing an unbiased error estimate via out-of-bag samples. However, it can become memory-intensive and complex with many trees, potentially slowing prediction times and increasing model size. [10]

Gradient Boosted Trees utilises the boosting method, building trees sequentially, where each tree corrects the errors of the previous ones. This is achieved through gradient descent optimisation on the loss function, with each tree fitting to the negative gradient of the previous tree's errors. The addition of a learning rate moderates each tree's contribution, mitigating overfitting risks. While Gradient Boosted Trees often achieve higher accuracy, they are computationally intensive and slower to train, with sequential tree construction hindering parallelization. Sequential training does allow for the use of fewer trees, speeding up inferencing and reducing model size. [11][12]

To train the model, Python will be used with the Yggdrasil Decision Forests (YDF) library. YDF was chosen for its speed and small size compared to other libraries. [13]

Results will be analysed to determine if changes are necessary for either the models or the dataset.

The machine learning algorithms will mainly be scored based on the accuracy and the F1-score. The accuracy is the ratio of the correctly predicted samples to the total samples. The F1-score is a metric which ranges from 0 to 1, with 1 being the best, that measures the performance of both precision and recall. Precision focuses on the correctness of the prediction, while recall focuses on the completeness of the prediction. Higher precision means fewer false positives and higher recall means fewer false negatives.

E. Dataset Expansion

The dataset will be expanded when the approach is verified to be working. This expansion adds combinations of the previously generated signals. Adding another four possible outcomes for the machine learning model to determine. These being:

- Fixed Carrier + Bluetooth
- Fixed Carrier + WiFi
- Bluetooth + WiFi
- All three combined

F. Final Model

A final model will be trained using the full dataset. Again, both the Random Forest and Gradient Boosted Trees algorithms will be trained, after which, a comparison has to be made.

G. Measurements and Testing

Finally, after a functioning final model is realised, it will be tested using real-world measurement data supplied by RDI.

III. Results and Discussion

The creation of the dataset and the training of the machine learning algorithm were done in Python. The typical SNR levels presented in section II-A are used for the creation of the dataset.

A. Fixed Carrier data generation

One of the chosen devices to identify was a Babyphone, which is essentially a wireless microphone. These devices usually work with a narrow-band fixed carrier for transmitting audio.

Because different brands approach the modulation of a carrier in different ways, the possibilities of both frequency modulation (FM), which is implemented in the hardware, and digital modulation, which is handled by a microcontroller in software, were taken into account.

This part of the dataset consists of both possibilities with varying carrier frequencies and SNR levels.

In fig. 2 the resulting spectrograms can be viewed. Two different fixed carrier modulation techniques are shown with a different SNR and a different carrier frequency.

B. Bluetooth data generation

The Bluetooth data only consists of 'Basic Rate' modulation, Enhanced Data Rate for Bluetooth has not been implemented to limit dataset size.

To generate the signal, a random bit sequence is generated. This is upsampled to match the sampling frequency and pulse shaped with a Gaussian pulse, resulting in a GFSK signal.

After the full signal was generated, frequency hopping was added by modulating the signal with a carrier signal at the proper frequency according to a pseudo-random sequence. This sequence is randomised for each data point in the dataset.

fig. 3 shows the resulting spectrograms of the generated Bluetooth signal, with different hopping sequences and SNR levels.

C. WiFi data generation

The WiFi signal is created by generating a random bitstream which is then grouped into sets of 4 bits. Each set of 4 bits is then mapped to a 16-QAM symbol and OFDM modulated, with each data subcarrier representing a different symbol. The pilot carriers have a known pilot value which is also set. Next, the inverse fast Fourier transform (IFFT) is taken from all subcarriers. Finally, a cyclic prefix is added, which is just a repetition of the final few samples of the generated signal. The cyclic prefix helps mitigate inter-symbol interference (ISI) by acting as a guard interval. This process is repeated until there is no more data to encode. Finally, a short spacing is added between each packet of 10 μs (SIFS).

The result with the generated spectrogram can be seen in fig. 4. Each plot has different packet lengths and a different SNR.

D. Initial Results

Initially, only the case of a single device sending at a time was considered. This was done to verify if the machine-learning approach would work. If this hadn't worked, the approach had to be reconsidered because a larger dataset with more variables would make it more difficult to train a proper machine learning algorithm.

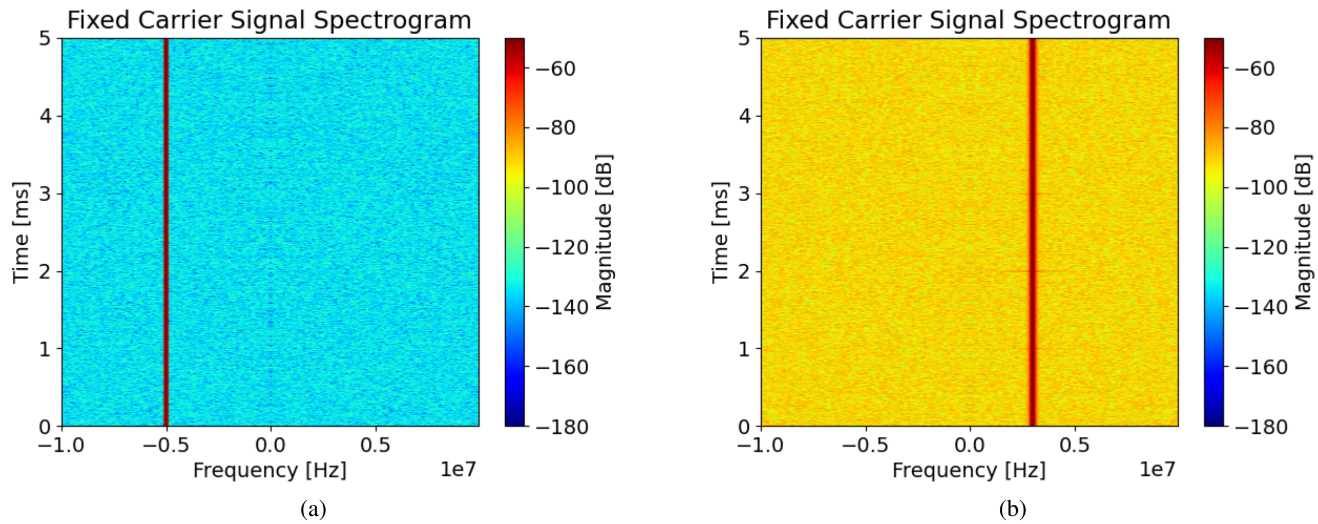


Fig. 2: Fixed carrier spectrograms (a): FM modulated carrier with $f_c = -5[MHz]$ and $SNR = 60[dB]$; (b): QPSK modulated carrier with $f_C = 3[MHz]$ and $SNR = 20[dB]$

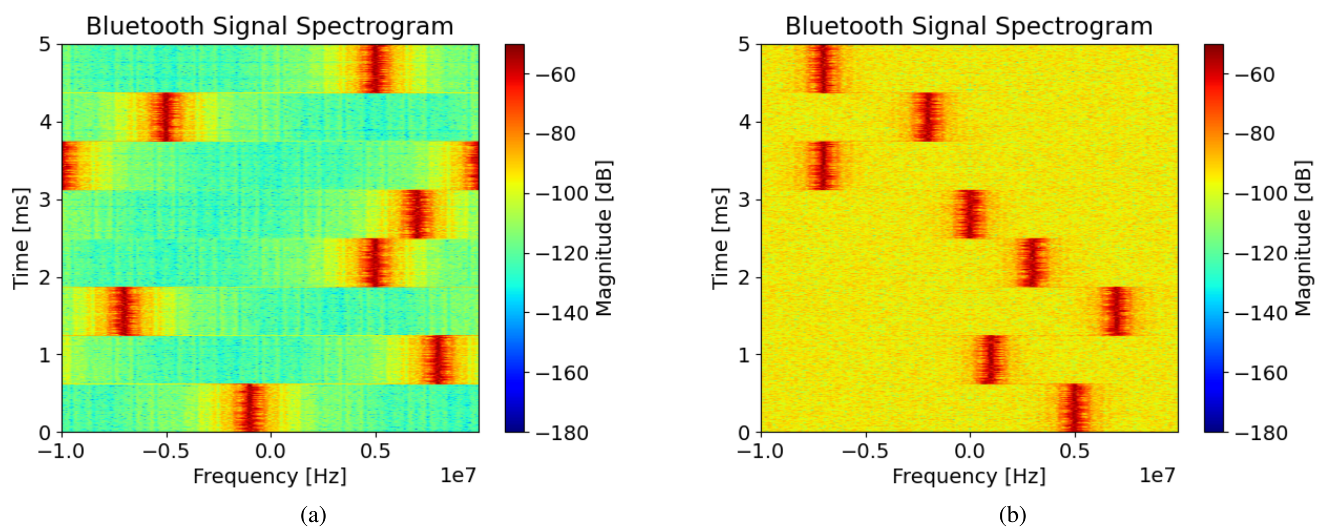


Fig. 3: Bluetooth signal spectrograms with different hopping sequences (a) $SNR = 60[dB]$; (b) $SNR = 20[dB]$

	Gradient Boosted Trees	**Random Forest**
Accuracy [%]	98.5	96.5
F1-Score	0.985	0.966
Training Time	48m 26s	1m 59s
Inference Time [μs]	2.13	3.89
Model Size [MB]	9.9	11.2

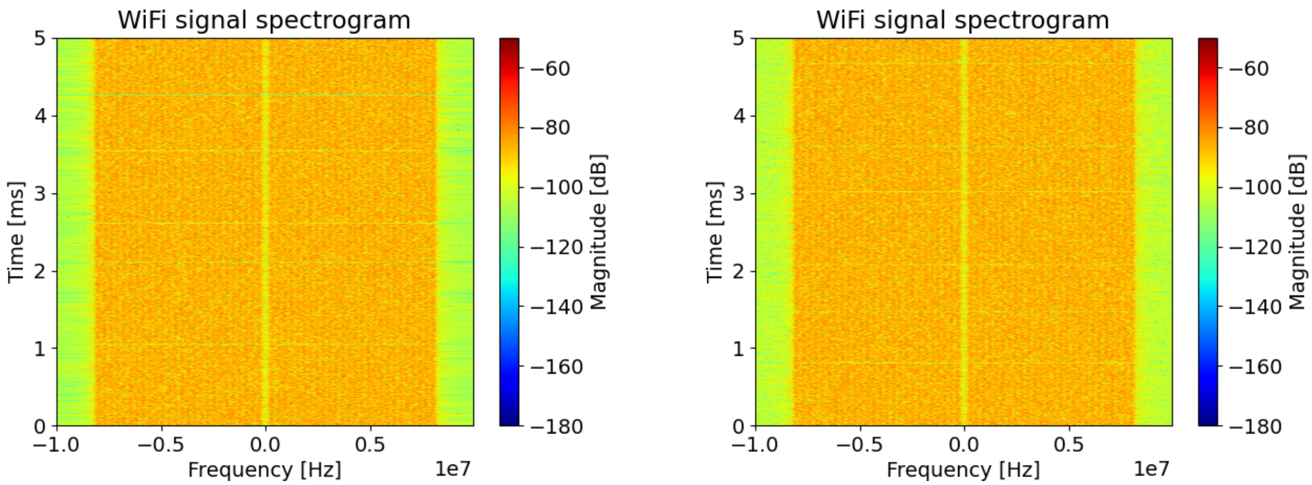
TABLE I: Initial Results of trained machine learning algorithms with limited dataset

table I shows the most important performance metrics of both algorithms, showing that the Gradient Boosted Trees method has the highest accuracy and F1-score with the current dataset. The accuracy is the ratio of correctly predicted values to the total number of predictions. The F1-score is the macro-average F1-score, taking the mean of the individual F1-scores for each possible prediction. The confusion matrices can be found in fig. 5, displaying all the predictions (x-axis) compared with the truth (y-axis). Both confusion matrices display a clear diagonal, indicating that most predicted devices were correct. Any non-zero value outside the diagonal indicates an incorrect prediction. For example, in fig. 5a, 35 data samples were falsely predicted as 'Fixed' when they were, in fact, 'Bluetooth'.

Both models have an accuracy of over 96% (table I), showing that these are viable models, allowing for the expansion of the dataset.

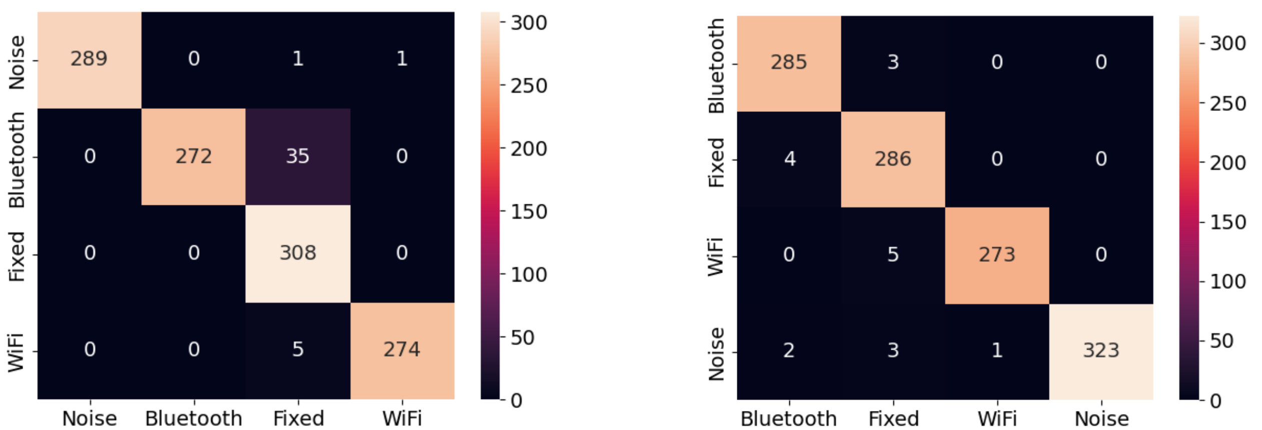
E. Expanded Dataset

After verifying that the chosen supervised machine learning approaches are viable solutions, the dataset had to be expanded to allow for the detection of multiple devices at once.



(a) (b)

Fig. 4: WiFi signal spectrograms with different packet lengths (a) $SNR = 60[dB]$; (b) $SNR = 20[dB]$



(a) (b)

Fig. 5: Confusion matrices of trained machine learning models, Random Forest (a) and Gradient Boosted Trees (b)

In fig. 6 the different combinations can be seen, where the generated data for the full dataset varies in SNR, Carrier frequency for the fixed carrier, hopping sequence for the Bluetooth signal, and packet length for the WiFi signal. Some of these variations can be seen between the subplots, with every sample in the dataset being different from one another. When combining the signals, a clear difference in signal power spectral density (PSD) can be viewed between WiFi and the other signals. This is because the power is more spread out in the frequency domain due to the OFDM modulation, resulting in a lower PSD.

F. Final Results

The most important performance metrics of both final models can be seen in table II. The Gradient Boosted Trees algorithm has the highest overall accuracy, F1-score and lowest inference time. The Random Forest model does have a significantly shorter training time while still achieving good results.

	Gradient Boosted Trees	**Random Forest**
Accuracy [%]	97.3	95.5
F1-Score	0.973	0.957
Training Time	2h 35m 04s	9m 23s
Inference Time [μs]	4.96	6.41
Model Size [MB]	10.4	14.8

TABLE II: Final Results of trained machine learning algorithms with complete dataset

The best-achieved accuracy was 97.3% (table II) with the Gradient Boosted Trees model.

The final confusion matrices can be seen in fig. 7. With a clearly defined diagonal, most predictions are correct. With the largest outlier, shown in fig. 7a, being the 36 false predictions of 'Fixed' instead of 'Bluetooth'.

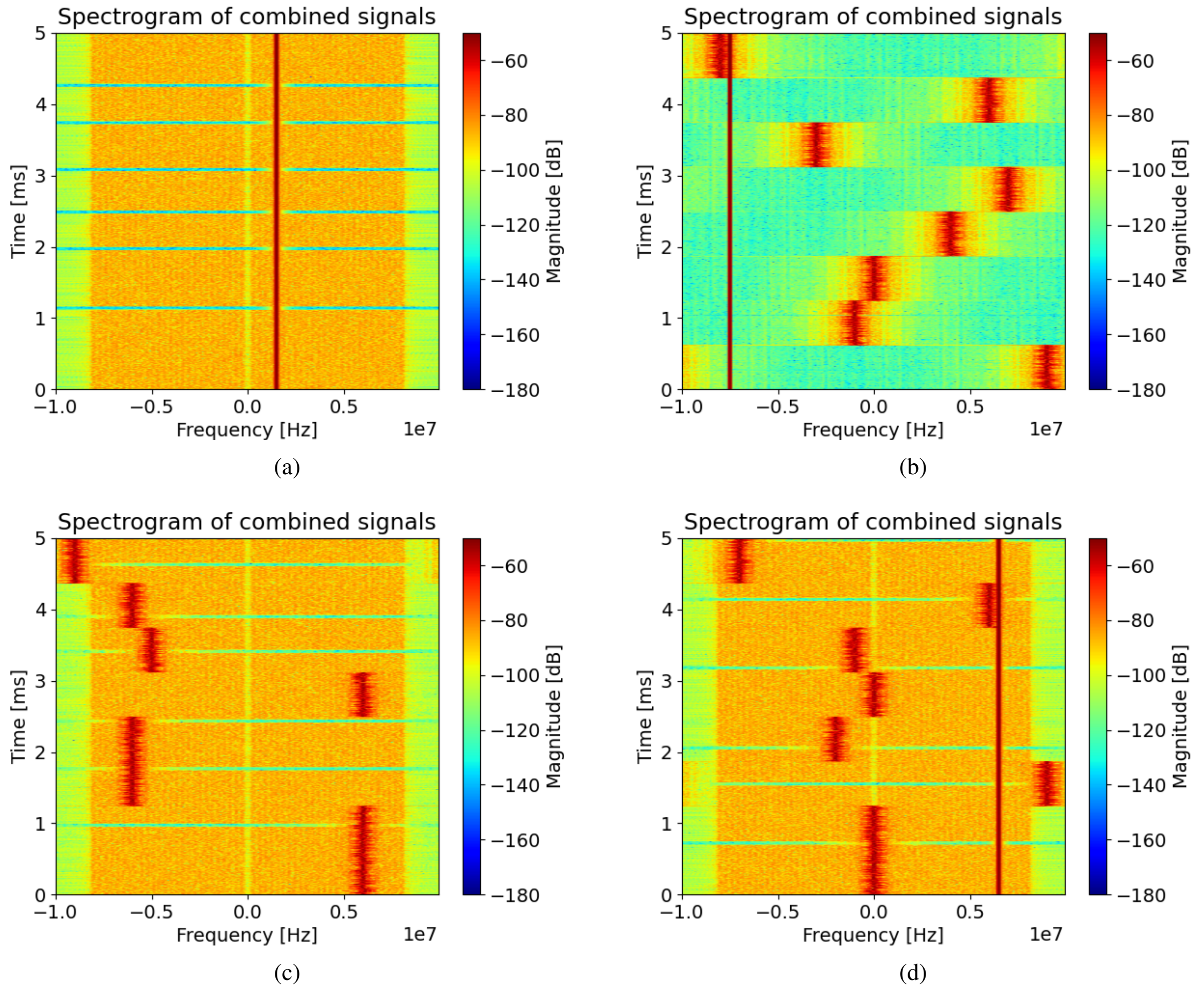


Fig. 6: Spectrograms of all possible combinations of signals, each with an SNR of 60 [dB] (a) Fixed & WiFi; (b) Fixed & Bluetooth; (c) Bluetooth & WiFi; (d) Combination of all three

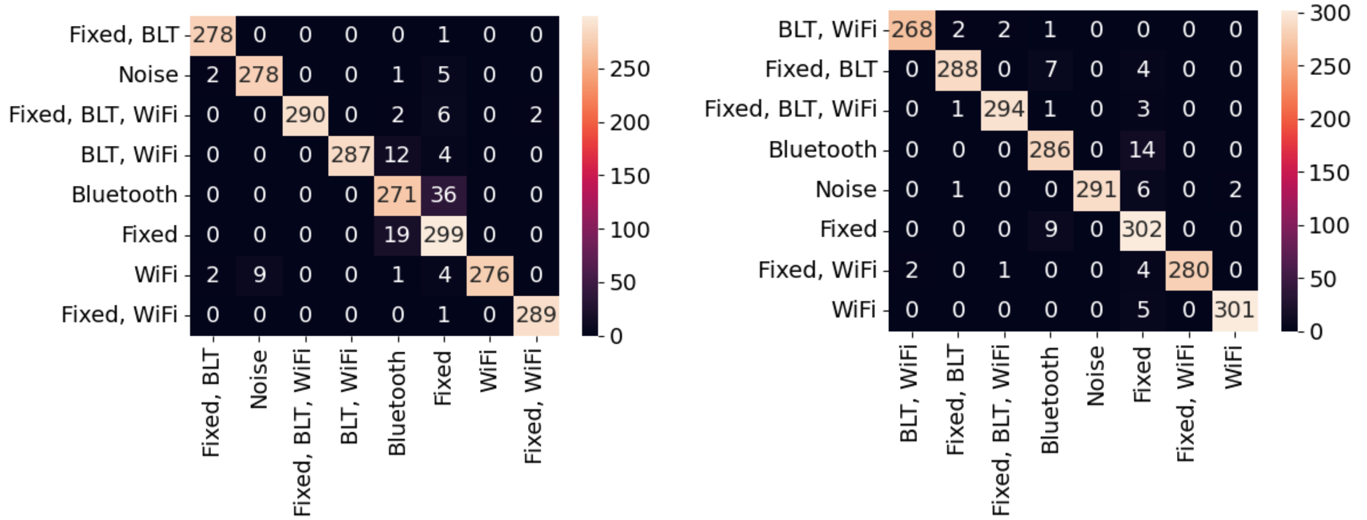
G. Real World Data

After successfully training a machine learning model with a generated dataset, RDI provided real-world measurement data (fig. 8). The data supplied by RDI consists of spectrograms from various locations throughout the Netherlands. Very little data preprocessing should be required for the machine learning model to function, as it was trained on spectrograms and expects this type of data. However, there are significant restrictions when using the data from the measurement setups at RDI. The main differences between the data supplied by RDI and the generated dataset are the timestep and the frequency span, with RDI doing measurements every minute and the generated dataset roughly every 6 μs, and the frequency span of the generated dataset spanning from -10 to 10 MHz (20MHz span). The supplied RDI data spans from 1.9GHz to 3GHz (1.1GHz span).

The choice of having the generated dataset only span 20 MHz was set to achieve enough detail in the measurements while keeping the size of the dataset manageable.

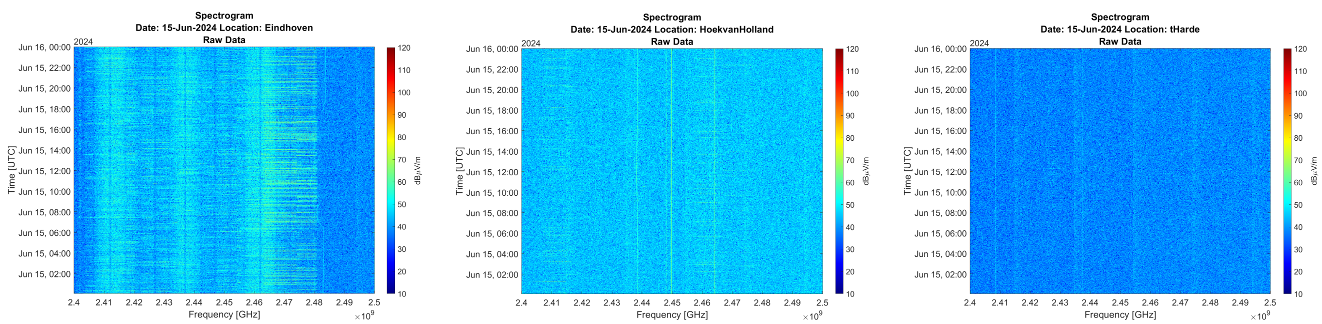
To be able to process real data with the current model, the frequency span was decreased to match the number of frequency bins in the training data. Additionally, the data was truncated to match the same length.

Since it is not known what kind of devices are sending, it is difficult to classify if the algorithm is right or wrong. This is why a sample has been hand-picked to ensure a visible fixed carrier. The hand-picked sample can be seen in fig. 9 with a clear fixed carrier around 2.45 GHz. The decision was made to search for a fixed carrier because the time intervals of the actual data are too large to detect Bluetooth hops. Additionally, it is more challenging to locate a WiFi signal due to the difficulty in confirming the presence of a WiFi signal with certainty from the measured data.



(a) (b)

Fig. 7: Confusion Matrices of the final models trained using the full dataset. (a) Random Forest; (b) Gradient Boosted Trees



(a) (b) (c)

Fig. 8: Real-world measurement data generated by RDI around the 2.4GHz band, (a)-(c) show different locations with clear differences between the spectrograms

	Noise	*Fixed*	BLT	WiFi	Fixed + BLT	Fixed + WiFi	WiFi + BLT	Fixed + BLT + WiFi
Gradient Boosted Trees	0.09	***0.48***	0.40	0	0	0.01	0	0.01
Random Forest	0.09	***0.43***	0.26	0	0.04	0.03	0.01	0.14

TABLE III: Prediction of both models on real-world data from RDI, each value representing the confidence of the prediction

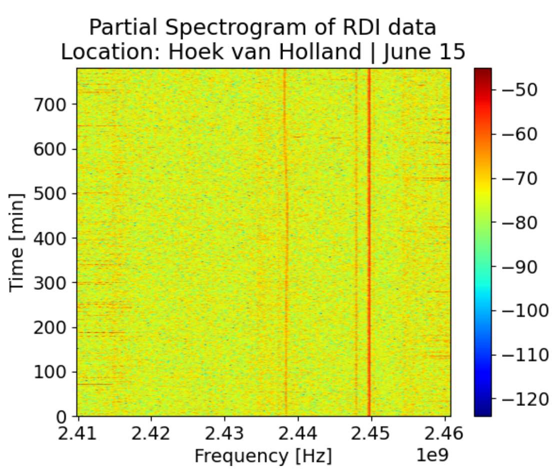


Fig. 9: Real-world measurement data from RDI clearly showing the presence of a fixed carrier, the data is zoomed and truncated to match the input shape of the machine learning algorithms

Finally, processing this hand-picked sample with both models results in the predictions in table III. Both the Random Forest and the Gradient Boosted Trees models correctly predict the fixed carrier with confidence levels of 43% and 48% respectively. Both models rank both the fixed carrier and Bluetooth high, the latter is likely due to the additional artefacts visible in fig. 9. Between these two options, the GBT ranks the fixed carrier with the highest confidence. To verify the other cases of the algorithms, measurement data should be gathered in a more controlled environment where the present devices are known. Since this is not possible with the current measurement setup at RDI a custom setup has to be made.

IV. Conclusion

The usage of supervised machine learning for the identification of WiFi, Bluetooth, and fixed carrier (Babyphone) devices has been explored in this research. Random Forest and Gradient Boosted Trees models were trained using a dataset closely resembling the real-world signals emitted by the aforementioned devices. The Random Forest model achieved 95.5% accuracy, while the Gradient Boosted Trees model achieved 97.3% accuracy in the experiments. The Gradient Boosted Trees model also had the fastest inference time at 4.96 μs, compared to 6.41 μs for the Random Forest model. Additionally, the Gradient Boosted Trees model had the smallest size at 10.4 MB. Despite the excellent performance, the main downside of the Gradient Boosted Trees model compared to the Random Forest model is the significantly longer training time.

The model was fed real-world data, and both algorithms accurately predicted the correct device. Gradient Boosted Trees was the most confident in its prediction. However, the analysis was limited to only a single hand-picked sample with a fixed carrier due to the measurement setup. A revised measurement setup is needed to thoroughly validate the model and draw a more conclusive assessment of its accuracy with real-world data.

Finally, it is concluded that supervised machine learning is a viable solution to the problem of identifying devices in the 2.4GHz spectrum, achieving an accuracy of 97.3% using the Gradient Boosted Trees model. This technology can be implemented to reduce the human workload and increase the processing speed of measurement data.

References

[1] W. Bakker, “Automatic detection of outlandish signal behaviour in the spectrum of cellular networks,” Master’s thesis, University of Twente, May 2019.

[2] F. Billon, *Data analysis for spectrum monitoring by the radio communications agency of the Netherlands*, 2022.

[3] B. Somers, “Finding forensic artefacts in long term frequency band occupancy measurements using statistics and machine learning,” minor thesis, University College of Dublin, August 2022.

[4] V. Iyer, F. Hermans, and T. Voigt, “Detecting and avoiding multiple sources of interference in the 2.4 ghz spectrum,” vol. 8965, February 2015, pp. 35–51.

[5] Unacademy. (2022, March) Types of communication systems. Accessed on 25.06.2024. [Online]. Available: https://unacademy.com/content/nda/study-material/physics/types-of-communication-system/

[6] J. D. Per Enge, Dennis Akos and V. S. Joel B. Simoneau, L. Wilson Pearson, “Measurements of man-made spectrum noise floor.” Washington, DC: NASA, 2004. [Online]. Available: https://ntrs.nasa.gov/citations/20050041714

[7] M. Li. (2022, January) Understanding the measures of bluetooth rssi. Accessed on 27.06.2024. [Online]. Available: https://www.mokoblue.com/measures-of-bluetooth-rssi/

[8] Metageek. Wi-fi signal strength basics. Accessed on 27.06.2024. [Online]. Available: https://www.metageek.com/training/resources/wifi-signal-strength-basics/

[9] P. Dziwior. (2004, September) Bluetooth - physical interface. Accessed on 28.06.2024. [Online]. Available: http://www.dziwior.org/Bluetooth/Physical.html

[10] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, pp. 5–32, 10 2001.

[11] Gaurav. An introduction to gradient boosting decision trees. Accessed on 28.06.2024. [Online]. Available: https://www.machinelearningplus.com/machine-learning/an-introduction-to-gradient-boosting-decision-trees/

[12] Google. (2022, September) Gradient boosted decision trees. Accessed on 28.06.2024. [Online]. Available: https://developers.google.com/machine-learning/decision-forests/intro-to-gbdt

[13] M. Guillame-Bert, S. Bruch, R. Stotz, and J. Pfeifer, “Yggdrasil decision forests: A fast and extensible decision forests library,” in *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, ser. KDD ’23. New York, NY, USA: Association for Computing Machinery, 2023, p. 4068–4077. [Online]. Available: https://doi.org/10.1145/3580305.3599933

Appendix

A. Use of AI

During the preparation of this work, the author used Grammarly in order to check for and correct spelling, grammar, and sentence structure errors. After using this tool/service, the author reviewed and edited the content as needed and takes full responsibility for the content of the work.

B. Code

Requesting the code used in this research is possible via e-mail: l.w.a.vink@student.utwente.nl