Deep learning based RF fingerprinting for device identification and wireless security

Qingyang Wu, Carlos Feres, Daniel Kuzmenko, Ding Zhi, Zhou Yu, Xin Liu and Xiaoguang 'Leo' Liu✉

RF fingerprinting is an emerging technology for identifying hardware-specific features of wireless transmitters and may find important applications in wireless security. In this study, the authors present a new RF fingerprinting scheme using deep neural networks. In particular, a long short-term memory based recurrent neural network is proposed and used for automatically identifying hardware-specific features and classifying transmitters. Experimental studies using identical RF transmitters showed very high detection accuracy in the presence of strong noise (signal-to-noise ratio as low as −12 dB) and demonstrated the effectiveness of the proposed scheme.

Introduction: 'RF fingerprinting' generally refers to the process of identifying the unique characteristics of a wireless transmitter hardware imposed on the transmitted signals. RF fingerprinting can be used to effectively prevent node impersonation, in which legitimate security credentials are obtained by an adversary to compromise the security [1].

Many hardware-dependent features have been explored for RF fingerprinting. These features exist due to variations in the manufacture process of wireless transmitters. These variations are small enough to meet the requirements of communication standards but allow for unique device-dependent features to be identified. Examples of such features include the turn-on transient phase of the signals [2, 3], power amplifier imperfections [4, 5], magnitude and phase errors, I/Q dc offset [6], carrier frequency differences, phase offset, and second-order cyclostationary features [7], clock offset [8].

Existing fingerprinting algorithms include white-list based algorithms and unsupervised learning based algorithms. The former requires legitimate devices to register and training a prior to setup a database for their feature space. The latter does not require such prior knowledge, and as such does not differentiate legitimate features from illegitimate ones. Both methods are useful in detecting and identifying spoofing. However, all existing works on RF fingerprinting depend on a set of *human engineered* features from various layers of the protocol stack [1]. In this work, we will demonstrate that deep neural networks can be used to effectively implement device identification with high accuracy through automatic learning of device-dependent RF fingerprints. In contrast to existing works, the proposed approach does not require human intervention in defining what features should be used in the RF fingerprinting process.

Recurrent neural networks (RNNs) for RF fingerprinting: Unlike standard feedforward neural networks, RNNs are neural networks that can present the dynamics of sequences using cycles within the network so that they retain a state that capture information from an arbitrarily long context window. Therefore, RNN models are especially suitable for sequence data. Traditionally, RNNs with millions of parameters were difficult to train. Recently, long short-term memory (LSTM) architecture has been developed to successfully address the problem of the vanishing gradients for RNNs [9]. LSTM systems have demonstrated ground-breaking performance on tasks with sequence data, including speech recognition [10], POS tagging [11], parsing [12], among many others.

Simple RNNs have long-term memory in the form of weights which change slowly during training, encoding general knowledge about the data. In 'LSTM' RNNs, short-term memory also exists in the form of ephemeral activations passing from each node to successive ones. Therefore, we expect LSTM would capture types of features that are long term, such as frequency drift, and features that are short term, such as the ramp-up trend in the initial sequence.

The LSTM architecture is made of a group of recurrently connected unit subnetworks, known as memory blocks. Each memory block contains one or more self-connected memory cells and three multiplicative units, the input, output and forget gates, that provide perform write, read and reset operations for the cells.

Fig. 1 shows a single LSTM memory cell. The multiplicative gates allow LSTM memory cells to retain information over long periods of time. This solves the problem of vanishing gradient during training. For example, the activation of the cell will not be overwritten by the new inputs as long as the input gate remains closed. By opening the gate at a much later time, the cell can be made available to the net much later in the sequence.

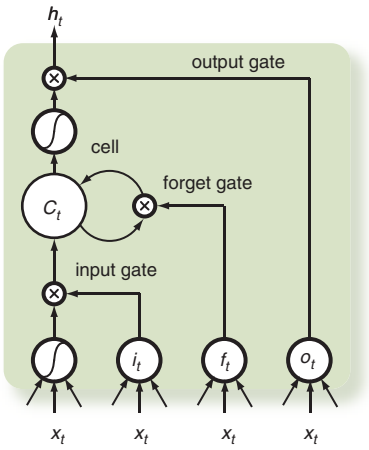

Fig. 1 *Long short-term memory cell*

Given an input $\boldsymbol{x} = (x_1, \ldots, x_T)$, the hidden vector $\boldsymbol{h} = (h_1, \ldots, h_T)$ and output vector $\boldsymbol{y} = (y_1, \ldots, y_T)$ are calculated by the following iterative equations from $t = 1$ to T in a conventional RNN:

$$h_t = \mathcal{H}(\boldsymbol{W}_{xh}x_t + \boldsymbol{W}_{hh}\boldsymbol{h}_{t-1} + \boldsymbol{b}_h),$$
$$y_t = \boldsymbol{W}_{hy}\boldsymbol{h}_t + \boldsymbol{b}_0,$$

where the $\boldsymbol{W}$ terms present weight matrices, the $\boldsymbol{b}$ terms present bias vectors, and $\mathcal{H}$ the hidden layer function. $\mathcal{H}$ is usually an element wise application of a sigmoid function. Each memory cell contains a node with a self-connected recurrent edge of fixed weight one to ensure that the gradient can pass across multiple time steps without vanishing or exploding.

Based on its unique features, we anticipate that LSTM-based RNN should be a suitable choice for RF fingerprinting applications. Unlike the existing literatures on RF fingerprinting, this approach does not use human-engineered features.

Data collection: To collect the necessary training and evaluation data, we conducted transmission/reception experiments using seven identical National Instruments USRP software defined radio modules (model USRP-2900) (six as transmitters and one as receiver). In each experiment, the transmitter and the receiver are connected by an RF test cable to have a well-controlled signal-to-noise ratio (SNR). Random data is generated and packaged into a WiFi-like packet before being sent to the transmitter. Single-carrier (925 MHz) QPSK modulation is used in this study. Gaussian noise is added to the received signal to simulate the degradation of the SNR due to channel loss and external and system noise (Fig. 2).

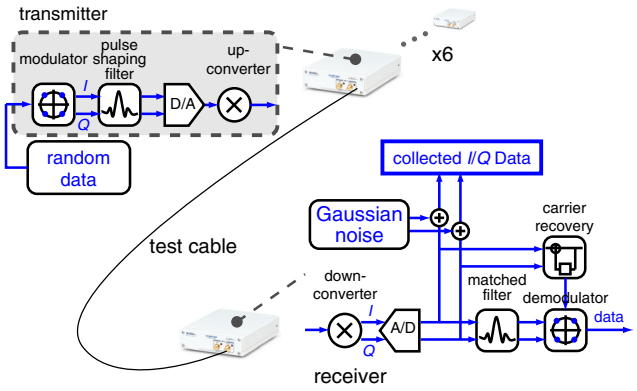

Fig. 2 *Experimental setup for training and evaluation data collection in the preliminary study*

In the experiments, each transmitter sends a total of 6000 packets, where each packet contains 250 symbols (500 bits). The receiver's analogue-to-digital (ADC) converter over-samples at 12 times the bit rate and results in 6000 samples per packet.

Baseband inphase (I) and quadrature (Q) samples are collected from the receiver. A subtle but important question is whether some level of physical layer correction should be implemented at the receiver. In a typical digital communication receiver, carrier frequency and phase recoveries are usually performed in order to synchronise the local oscillators (LO) of the receiver to that of transmitter. However, the current RF fingerprinting literature suggests that imperfect transmitter LO signal characteristics are a major part of hardware-specific fingerprint signatures. As such, the frequency and phase synchronisation may remove important features from the received data. Therefore, in our experiments, I/Q data is collected at the output of the ADC, before any matched filtering or carrier recovery is performed.

Model training: In this work, the hidden layer function $\mathcal{H}$ of an LSTM block is implemented as follows:

$$\begin{aligned}
\boldsymbol{i}_t &= \sigma(\boldsymbol{W}_{xi}x_t + \boldsymbol{W}_{\text{hi}}\boldsymbol{h}_{t-1} + \boldsymbol{W}_{ci}\boldsymbol{c}_{t-1} + \boldsymbol{b}_i),\\
\boldsymbol{f}_t &= \sigma(\boldsymbol{W}_{xf}\boldsymbol{x}_t + \boldsymbol{W}_{hf}\boldsymbol{h}_{t-1} + \boldsymbol{W}_{cf}\boldsymbol{c}_{t-1} + \boldsymbol{b}_f),\\
\boldsymbol{c}_t &= \boldsymbol{f}_t\boldsymbol{c}_{t-1} + \boldsymbol{i}_t \tanh(\boldsymbol{W}_{xc}\boldsymbol{x}_t + \boldsymbol{W}_{hc}\boldsymbol{h}_{t-1} + \boldsymbol{b}_c),\\
\boldsymbol{o}_t &= \sum(\boldsymbol{W}_{xo}\boldsymbol{x}_t + \boldsymbol{W}_{ho}\boldsymbol{h}_{t-1} + \boldsymbol{W}_{co}\boldsymbol{c}_t + \boldsymbol{b}_o),\\
\boldsymbol{h}_t &= \boldsymbol{o}_t \tanh(\boldsymbol{c}_t),
\end{aligned}$$

where σ is the logistic sigmoid function, and $\boldsymbol{i}$, $\boldsymbol{f}$, $\boldsymbol{o}$, and $\boldsymbol{c}$ are, respectively, the input gate, forget gate, output gate, and cell activation vectors, all of which are the same size as the hidden vector $\boldsymbol{h}$. $\boldsymbol{W}_{hi}$ is the hidden-input gate matrix, $\boldsymbol{W}_{xo}$ is the input-output gate matrix. The weight matrix from the cell to gate vectors (e.g. $\boldsymbol{W}_{ci}$) are diagonal, so element m in each gate vector only receives input from element m of the cell vector. The bias terms have been omitted for clarity.

The collected data/samples are divided into sequences of one packet per sequence. The collection of data is divided into a training dataset (80%) and an evaluation dataset (20%). Using these data, we train an RNN with one hidden layer of 50 LSTM cells and 1 softmax output layer.

Performance: Although the trained LSTM model has a relatively simple structure, its effectiveness in classifying the transmitters in the presence of strong noise and interference is surprisingly high.

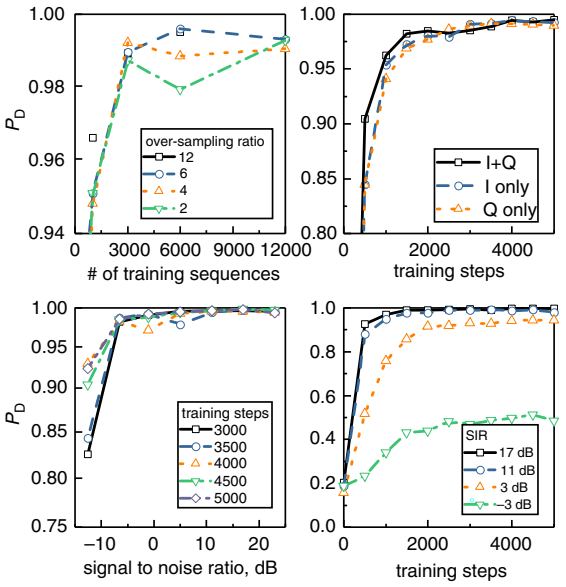

Fig. 3 *Model P_D of the preliminary study using a simple LSTM network versus*

a Over-sampling ratio and number of training sequences
b I/Q data
c SNR
d Signal-to-interference ratio (SIR). Red lines in each sub-plot indicates the $P_D = 0.95$ level

Fig. 3*a* shows the probability of detection (P_D) of the classification for various effective over-sampling ratio and number of sequences used in the training. Above 95% P_D is achieved for as low as 1000 sequences (packets) and an over-sampling ratio of 2 (1000 samples per packet). As expected for deep learning models, the performance improves as we increase the training data size. For all over-sampling ratios, P_D is greater than 0.99 with 6000 training sequences. We have not observed a significant performance difference with respect to whether I, Q, or both data is used (Fig. 3*b*).

Fig. 3*c* shows the performance with respect to the SNR. As expected, a degradation in SNR results in a degradation in P_D. The loss in accuracy can be partially recovered by increasing the number of training steps. At 4000 training steps, the model remains surprisingly accurate with $P_D > 95\%$ at an SNR of -12 dB. As the number of training steps is further increased, we start to observe overfitting of the model and a degradation in P_D.

The LSTM model is also very resilient to interference with $P_D > 0.9$ at an SIR of 3 dB (Fig. 3*d*. Here, we emulate the effect of interference by adding to a sequence $x[n]$ of transmitter i a scaled version of a random sequence $y[n]$ of a random transmitter $j \neq i$, i.e. $x'[n] \Leftarrow x[n] + \alpha y[n]$. The scaling factor α is varied to give an effective SIR between -3 and 17 dB.

Conclusion: In this work, we present a deep neural network based RF fingerprinting scheme for wireless device identification and wireless security applications. We propose an LSTM based RNN model that effectively captures the long-term and short-term hardware-specific features of a wireless transmitter. We validated the effectiveness of the proposed scheme by experimentally showing very high detection accuracy in the presence of strong noise and interference. This first work opens up the door to follow-on studies that will further improve the effectiveness of the proposed scheme.

Submitted: *24 July 2018*
doi: 10.1049/el.2018.6404
One or more of the Figures in this Letter are available in colour online.

Qingyang Wu, Carlos Feres, Daniel Kuzmenko, Ding Zhi, Zhou Yu, Xin Liu and Xiaoguang 'Leo' Liu (*University of California, Davis, USA*)

✉ E-mail: lxgliu@ucdavis.edu

References

1 Xu, Q., Zheng, R., Saad, W., *et al.*: 'Device fingerprinting in wireless networks: challenges and opportunities', *IEEE Commun. Surveys Tutor.*, 2016, **18**, (1), pp. 94–104
2 Hall, J., Barbeau, M., and Kranakis, E.: 'Enhancing intrusion detection in wireless networks using radio frequency fingerprinting', Communications, internet, and information technology, St. Thomas, US Virgin Islands, November 2004, pp. 201–206
3 Ureten, O., and Serinken, N.: 'Wireless security through rf fingerprinting', *Can. J. Electr. Comput. Eng.*, 2007, **32**, (1), pp. 27–33
4 Dolatshahi, S., Polak, A., and Goeckel, D.L.: 'Identification of wireless users via power amplifier imperfections'. Signals, Systems and Computers (ASILOMAR), 2010 Conf. Record of the Forty Fourth Asilomar Conf., Pacific Grove, CA, USA, November 2010, pp. 1553–1557
5 Polak, A.C., Dolatshahi, S., and Goeckel, D.L.: 'Identifying wireless users via transmitter imperfections', *IEEE J. Sel. Areas Commun.*, 2011, **29**, (7), pp. 1469–1479
6 Brik, V., Banerjee, S., Gruteser, M., *et al.*: 'Wireless device identification with radiometric signatures'. Proc. of the 14th ACM Int. Conf. on Mobile Computing and Networking, San Francisco, CA, USA, September 2008, pp. 116–127
7 Nguyen, N.T., Zheng, G., Han, Z., *et al.*: 'Device fingerprinting to enhance wireless security using nonparametric bayesian method'. 2011 Proceedings IEEE INFOCOM, Shanghai, China, April 2011, pp. 1404–1412
8 Rahman, M.M.U., Yasmeen, A., and Gross, J.: 'Phy layer authentication via drifting oscillators'. 2014 IEEE Global Communications Conference, Austin, TX, USA, December 2014, pp. 716–721
9 Hochreiter, S., and Schmidhuber, J.: 'Long short-term memory', *Neural Comput.*, 1997, **9**, (8), pp. 1735–1780
10 Graves, A., Mohamed, A.-R., and Hinton, G.: 'Speech recognition with deep recurrent neural networks'. 2013 IEEE Int. Conf. Acoustics, Speech and Signal Processing (ICASSP), Vancouver, Canada, May 2013, pp. 6645–6649
11 Huang, Z., Xu, W., and Yu, K.: 'Bidirectional lstm-crf models for sequence tagging', *arXiv preprint arXiv:1508.01991*, 2015
12 Kiperwasser, E., and Goldberg, Y.: 'Simple and accurate dependency parsing using bidirectional lstm feature representations', *arXiv preprint arXiv:1603.04351*, 2016