Distributed Deep Learning Models for Wireless Signal Classification with Low-Cost Spectrum Sensors

Sreeraj Rajendran, *Student Member, IEEE,* Wannes Meert, Domenico Giustiniano, *Senior Member, IEEE,* Vincent Lenders and Sofie Pollin, *Senior Member, IEEE.*

***Abstract*—This paper looks into the technology classification problem for a distributed wireless spectrum sensing network. First, a new data-driven model for Automatic Modulation Classification (AMC) based on long short term memory (LSTM) is proposed. The model learns from the time domain amplitude and phase information of the modulation schemes present in the training data without requiring expert features like higher order cyclic moments. Analyses show that the proposed model yields an average classification accuracy of close to 90% at varying SNR conditions ranging from 0dB to 20dB. Further, we explore the utility of this LSTM model for a variable symbol rate scenario. We show that a LSTM based model can learn good representations of variable length time domain sequences, which is useful in classifying modulation signals with different symbol rates. The achieved accuracy of 75% on an input sample length of 64 for which it was not trained, substantiates the representation power of the model. To reduce the data communication overhead from distributed sensors, the feasibility of classification using averaged magnitude spectrum data, or online classification on the low cost sensors is studied. Furthermore, quantized realizations of the proposed models are analyzed for deployment on sensors with low processing power.**

***Index Terms*—Deep learning, Modulation classification, LSTM, CNN, Spectrum sensing.**

I. Introduction

Wireless spectrum monitoring over frequency, time and space is important for a wide range of applications such as spectrum enforcement for regulatory bodies, generating coverage maps for wireless operators, and applications including wireless signal detection and positioning. Continuous spectrum monitoring over a large geographical area is extremely challenging mainly due to the multidisciplinary nature of the solution. The monitoring infrastructure requires proper integration of new disruptive technologies than can flexibly address the variability and cost of the used sensors, large spectrum data management, sensor reliability, security and privacy concerns, which can also target a wide variety of the use cases. Electrosense was designed to address address these challenges and support a diverse set of applications [1]. Electrosense is a crowd-sourced spectrum monitoring solution deployed on a large scale using low cost sensors.

One of the main goals of Electrosense is to accomplish automated wireless spectrum anomaly detection, thus enabling efficient spectrum enforcement. Technology classification or specifically AMC is an integral part of spectrum enforcement. Such a classifier can help in identifying suspicious transmissions in a particular wireless band. Furthermore, technology classification modules are fundamental for interference detection and wireless environment analysis. Technology classification is also used in Wireless Sensor Networks (WSN) for adapting the schedule of the wireless sensors as function of the detected interfering technology. Considering the aforementioned large application space this paper looks into two key aspects: Is efficient wireless technology classification achievable on a large scale with low cost sensor networks and limited uplink communication bandwidth? If possible, which are the key classification models suitable for the same.

The number of publications related to AMC appearing in literature is large [2]–[7] mainly due to the broad range of problems associated with AMC and huge interest in the problem itself for surveillance applications. AMC helps a radio system for environment identification, defining policies and taking actions for throughput or reliability improvements. It is also used for applications like transmitter identification, anomaly detection and localization of interference [2,3].

Various approaches for modulation classification discussed in literature can be brought down to two categories [4], one being the *decision theoretic* approach and the other the *feature based* approach. In decision-theoretic approaches the modulation classification problem is presented as a multiple hypothesis testing problem [4]. The maximum likelihood criterion is applied to the received signal directly or after some simple transformations such as averaging. Even though decision-theoretic classifiers are optimal in the sense that they minimize the probability of miss-classifications, practical implementations of such systems suffer from computational complexity as they typically require buffering a large number of samples. These methods are also not robust in the presence of unknown channel conditions and other receiver discrepancies like clock frequency offset.

Conventional feature-based approaches for AMC make use of expert features like cyclic moments [5]. Spectral correlation functions of various analog and digital modulation schemes covered in [8] and [9] respectively are the popularly used features for classification. Detailed analysis of various methods using these cyclostationary features for modulation classification are presented in [6]. Various statistical tests for detecting the presence of cycles in the kth-order cyclic cumulants without assuming any specific distribution on the data are presented in [7]. In [10] authors used a multilayer linear perceptron network over spectral correlation functions for classifying some basic modulation types. Another method makes use of the cyclic prefix [11] to distinguish between multi-carrier and single carrier modulation schemes which is used for OFDM signal identification.

All these aforementioned model driven approaches exploit knowledge about the structure of different modulation schemes to define the rules for AMC. This manual selection of expert features is tedious which makes it difficult to model all channel discrepancies. For instance, it is quite challenging to develop models which are robust to fading, pathloss, time shift and sample rate variations. In addition, a distributed collection of in-phase and quadrature phase (IQ) data over frequency, space and time is expensive in terms of transmission bandwidth and storage. Furthermore, most of these algorithms are processor intensive and could not be easily deployed on low-end distributed sensors.

Recently, deep learning has been shown to be effective in various tasks such as image classification, machine translation and automatic speech recognition [12], thanks to multiple hidden layers with non-linear logistic functions which enable learning higher-level information hidden in the data. A recently proposed deep learning based model for AMC makes use of a Convolutional Neural Network (CNN) based classifier [13]. The CNN model operates on the time domain IQ data and learns different matched filters for various signal-to-noise ratio (SNR). However, this model may not be efficient on data with unknown sampling rates and pulse shaping filters which the model has never encountered during the training phase. Also being a fixed input length model, the number of modulated symbols the model can process remains limited across various symbol rates. Furthermore, the training and computational complexity of the model increases with increasing input sample length. In [14], the authors extended the analysis on the effect of CNN layer sizes and depths on classification accuracy. They also proposed complex inception modules combining CNN and Long Short Term Memory (LSTM) modules for improving the classification results. In this paper we show that simple LSTM models can itself achieve good accuracy, if input data is formatted as amplitude and phase (polar coordinates) instead of IQ samples (rectangular coordinates).

This paper proposes a LSTM [15] based deep learning classifier solution, which can learn long term temporal representations, to address the aforementioned issues. The proposed variable input length model can capture sample rate variations without explicit feature extraction. We first train the LSTM model to classify 11 typical modulation types, as also used in [16], and show our approach outperforms the state-of-the-art (SoA). Being a variable input length model we also show that the model enables efficient classification on variable sample rates and sequence lengths. Even though these deep learning models can provide good classification accuracies on lower input sample lengths, their computational power requirements are still high preventing them from low-end sensor deployment as Electrosense.

The wireless sensing nodes deployed in the Electrosense network consist of a low-cost and bandwidth limited Software Defined Radio (SDR) interfaced with a small sized embedded platform [1]. Averaged magnitude Fast Fourier Transform (FFT) and IQ pipelines are enabled in the sensor to support various applications producing data in the order of 50-100 Kbps and 50 Mbps respectively. First, the embedded hardware of the sensors is not powerful enough to handle performance intensive AMC algorithms. Second, transferring IQ samples to the backend for classification by enabling the IQ pipeline is not a scalable solution as it is expensive in terms of data transfer and storage. Finally, the sensors are bandwidth limited which prevents them from acquiring wideband signals.

To enable instantiation of the newly proposed LSTM model for modulation classification in a large distributed network of low cost sensor nodes, we compare various approaches to decrease the implementation cost of the classifier. In the first approach we study the advantages and limitations of classification models for technology classification on a deployed distributed sensor network with limited bandwidth sensors based on averaged magnitude FFT data which decreases the communication cost by a factor 1000. Moreover, quantized versions of the proposed models are studied in detail for sensor deployment. These quantized versions can be run on a low cost sensor and don't require the instantiation of the classifier in the cloud. As a result, the sensor should only communicate the decision variable, which further decreases the communication cost. The code and datasets for all the deep learning models are made public for future research[1]. The models are also available for use through Electrosense.

The contribution of this paper is thus threefold. First, we develop a new LSTM based deep learning solution using time domain amplitude and phase samples which provides SoA results for high SNRs on a standard dataset. Second, we explore the use of deep learning models for technology classification task in a distributed sensor network only using averaged magnitude FFT data. Finally, we explore the model performance by quantizing the deep neural networks for sensor deployment.

The rest of the paper is organized as follows. The classification problem is clearly stated in Section II. A brief overview of the modulation dataset and the channel models used are presented in Section III. Section IV explains the LSTM model used for classification and the parameters used for training along with other implementation details. Section V details the classification results and discusses the advantages of the proposed model. Low-implementation cost models are discussed in Section VI. Conclusions and future work are presented in Section VII.

II. Problem Statement

Technology or modulation recognition can be framed as a N-class classification problem in general. A general representation for the received signal is given by

$$r(t) = s(t) * c(t) + n(t), \tag{1}$$

where $s(t)$ is the noise free complex baseband envelope of the received signal, $n(t)$ is Additive White Gaussian Noise (AWGN) with zero mean and variance σ_n^2 and $c(t)$ is the time varying impulse response of the transmitted wireless channel. The basic aim of any modulation classifier is to give out

[1] https://github.com/zeroXzero/modulation_classif.git

Modulations	8PSK, AM-DSB, AM-SSB, BPSK, CPFSK, GFSK, PAM4, QAM16, QAM64, QPSK, WBFM
Samples per symbol	4
Sample length	128
SNR Range	-20dB to +20dB
Number of training samples	82500 vectors
Number of test samples	82500 vectors

TABLE I: RadioML2016.10a dataset parameters.

Modulations	8PSK, AM-DSB, AM-SSB, BPSK, CPFSK, GFSK, PAM4, QAM16, QAM64, QPSK, WBFM
Samples per symbol	4 and 8 sps
Sample length	128 to 512
SNR Range	-20dB to +20dB
Number of training samples	165000 vectors
Number of test samples	165000 vectors

TABLE II: Modified complex RadioML dataset parameters.

$P(s(t) \in N_i | r(t))$ with $r(t)$ as the only signal for reference and N_i represents the ith class. The received signal $r(t)$ is commonly represented in IQ format due to its flexibility and simplicity for mathematical operations and hardware design. The in-phase and quadrature components are expressed as $I = Acos(\phi)$ and $Q = Asin(\phi)$, where A and ϕ are the instantaneous amplitude and phase of the received signal $r(t)$.

The RadioML and modified RadioML datasets used for testing the proposed model, presented in the next section of this paper, follow the signal representation as given in equation 1. These datasets make a practical assumption that the sensor's sampling rate is high enough to receive the full-bandwidth signal of interest at the receiver end as $r(t)$. The datasets also take into account complex receiver imperfections which are explained in detail in Section III. The *samples per symbol* parameter used in the tables I and II specify the number of samples representing each modulated symbol which is a modulation characteristic. Similarly *sample length* parameter specifies the number of received signal samples used for classification.

III. Modulation Datasets

A publicly available dataset used for evaluating the performance of the proposed model is detailed in this section. The standard dataset is also extended to evaluate the sample rate dependence of the proposed model.

A. RadioML dataset

A standard modulation dataset presented in [16] is used as the baseline for training and evaluating the performance of the proposed classifier. The used RadioML2016.10a dataset is a synthetically generated dataset using GNU Radio [17] with commercially used modulation parameters. This dataset also includes a number of realistic channel imperfections such as channel frequency offset, sample rate offset, additive white gaussian noise along with multipath fading. It contains modulated signals with 4 samples/symbol (sps) and a sample length of 128 samples. Used modulations along with the complete parameter list can be found in Table I. Detailed specifications and generation details of the dataset can be found in [16].

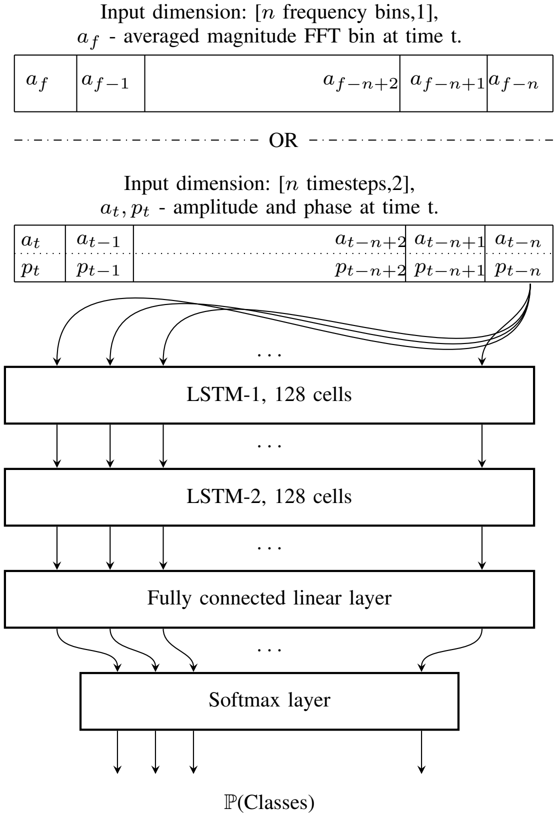

Fig. 1: Two layer LSTM model for classification. The model is trained and deployed for technology classification using either amplitude-phase signal or the averaged magnitude-FFT signal as input.

B. Modified RadioML dataset

The standard radioML dataset is extended using the generation code[2] by varying the *samples per symbol* and *sample length* parameters for evaluating the sample rate dependencies of the LSTM model. The extended parameters of the used dataset are listed in the Table II. The extended dataset contains signals with 4 and 8 samples per symbol. This dataset is generated to evaluate the robustness of the model in varying symbol rate scenarios.

IV. Model Description

The proposed LSTM model, that works on the time domain amplitude and phase signal, is introduced in the following subsection. In addition, the baseline CNN model used for comparisons is also detailed.

A. Model for complex signals

A LSTM network with different layers is used for complex data classification as shown in Figure 1. The amplitude and phase of the time domain modulated signal are fed to all cells of the LSTM model as a two dimensional vector, at each time step for classification. The amplitude vector is L2 normalized and the phase, which is in radians is normalized between -1 and 1. The first two layers are comprised of 128 LSTM cells each. The final output from the second LSTM layer, a vector

[2] https://github.com/radioML/dataset

of dimension 128, after all time steps, is fed to the last dense layer of the model. The final layer is a dense softmax layer which maps the classified features to one of the 11 output classes representing the modulation schemes. The two layer model is selected after detailed analysis varying the cell size and layer depths which are detailed in Section V-D.

The intuition to use a LSTM model for classification is based on the fact that different modulation schemes exhibit different amplitude and phase characteristics and the model can learn these temporal dependencies effectively. Even though fading and other real world effects may slightly hamper the characteristics of the signal, we expect the model to classify signals efficiently by learning good fading resistant representations. Since the proposed model can work on variable length input time domain samples, we expect the model to learn useful symbol rate independent representations for classification. In addition, the importance of the number of LSTM cells and layer depth are further investigated by varying these parameters. Model classification accuracies are analyzed with varying layer depth from 1 to 3 and number of cells from 16 to 256. We further analyze these aspects in detail in Section V.

B. Baseline IQ model

The two layer CNN 8 tap model presented in [14] is used as the baseline model for further comparisons. The baseline model uses 256 and 80 filters in the first two convolutional layers respectively. A publicly available training model is used for generating the baseline performance graph [13].

C. Model training and testing

Each of the datasets mentioned in Tables I and II are split into two, one training set and the other testing set. A seed is used to generate random mutually exclusive array indices, which are then used to split the data into two ascertaining the training and testing sets are entirely different. The number of the training and testing vectors are listed in the corresponding tables. A softmax cross entropy with logits, that measures the probability error in discrete classification tasks in which the classes are mutually exclusive, is used as the loss function. Stochastic gradient descent with a minibatch size of 400 vectors is used to avoid local optima. We use the Adam optimizer [18], a first-order gradient based optimizer with a learning rate of 0.001. The complex two layer LSTM network is trained for 70 epochs which takes around an hour of training time on a x86 PC with Nvidia GeForce GTX 980 Ti graphics card. We use a basic LSTM cell with starting training forget bias set to one. While initializing the network, it is helpful to keep the scale of the input variance constant, so that it does not explode or diminish by reaching the final layer. To achieve this LSTM weights are initialized with a default uniform unit scaling initializer which generates weights with a uniform variance. All the models use the same training parameters unless specified explicitly.

D. Implementation details

The neural network is implemented using TensorFlow [19], a data flow graph based numerical computation library from Google. Python and C++ bindings of Tensorflow makes the usage of the final trained model easily portable to host based SDR frameworks like GNU Radio [17]. The trained model can be easily imported as a block in GNU Radio which can be readily used in practice with any supported hardware front-end.

V. Results and discussion

The classification accuracies of the model for the aforementioned datasets along with the learned representations are discussed in the following subsections.

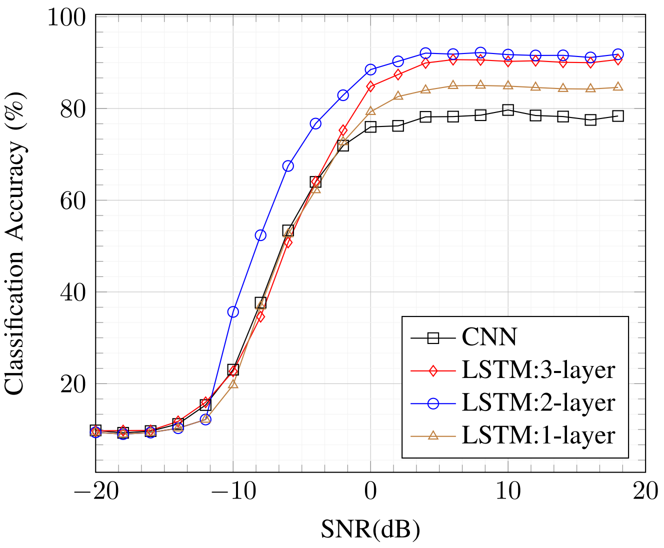

Fig. 2: Classification accuracy comparison of hyper-parameter optimized 2-layer LSTM model with others on RadioML dataset.

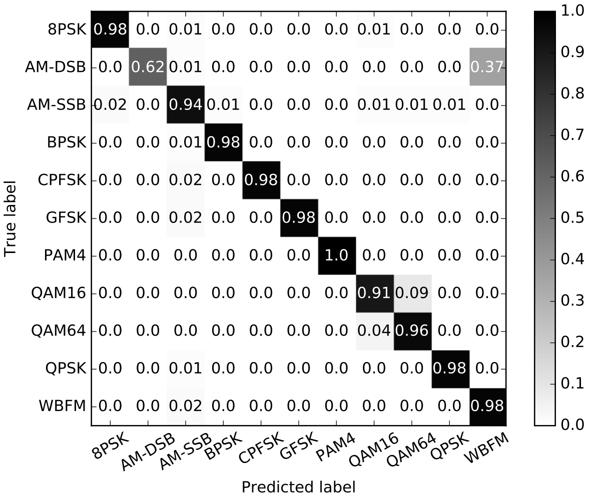

Fig. 3: Confusion matrix for 18dB SNR

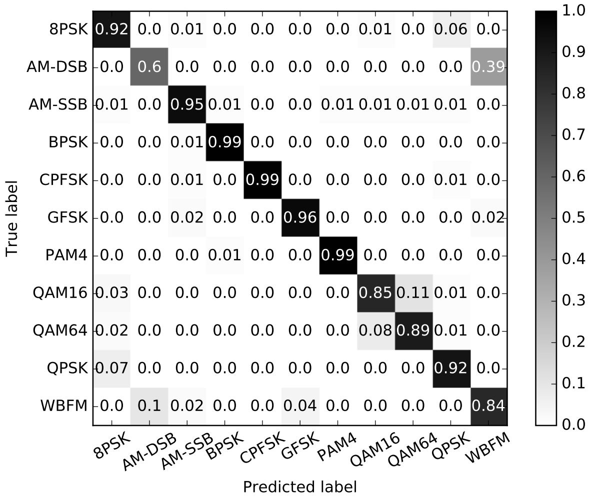

Fig. 4: Confusion matrix for 0dB SNR

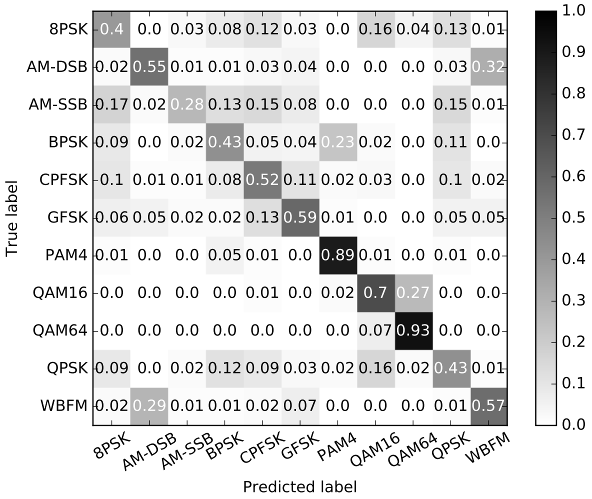

Fig. 5: Confusion matrix for -8dB SNR

A. *Classification accuracy on RadioML dataset*

The two layer amplitude-phase LSTM model, shown in Figure 1, is trained on SNR ranges from -10dB to 20dB. Training vectors with SNR ranges below -10dB were not used as the model was converging slowly when those vectors were used. Alternate models with varying LSTM layer depths are also trained to understand the performance improvements provided by the different layer depths. The classification accuracy of all the four models are presented in Figure 2. The two layer LSTM model gave an average accuracy of 90% in SNR ranges from 0dB to 20dB. It can be noticed that the single layer LSTM also reaches a high accuracy at high SNRs, 6% less than the two layer model. It was also noticed that the classification accuracy saturates for layer depths of two. Hence, layer depth of two is selected for the final model and its parameters are fine tuned to achieve the best test performance as shown in Figure 2. Rigorous fine tuning was not performed on layer depths other than two accounting for a slightly lower accuracy levels, for instance the accuracy level for layer depth 3 is slightly lower than layer depth two. The performance of the baseline CNN model was shown to be much better on the low SNR regions in [14]. We were not able to reproduce the reported results on the low SNR regions after various attempts, which may be because of the difference in hyper-parameter tuning. Though, the high SNR results of the baseline model matches with that of the reported ones in the paper. Detailed discussions on the effect of layer depth and number of LSTM cells are presented in Section V-D.

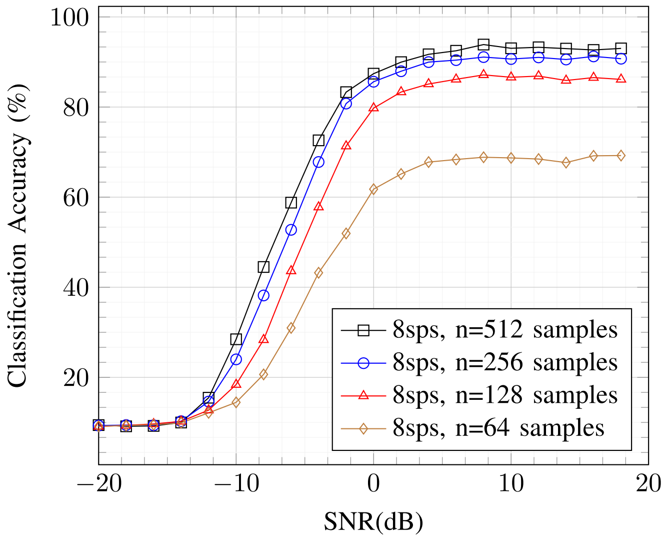

Fig. 6: Classification accuracy on modified RadioML dataset for 8 samples/symbol. The model is trained only on input sample lengths (n) from 128 to 512. The model gives close to 70% accuracy on 64 input samples which it is not trained for.

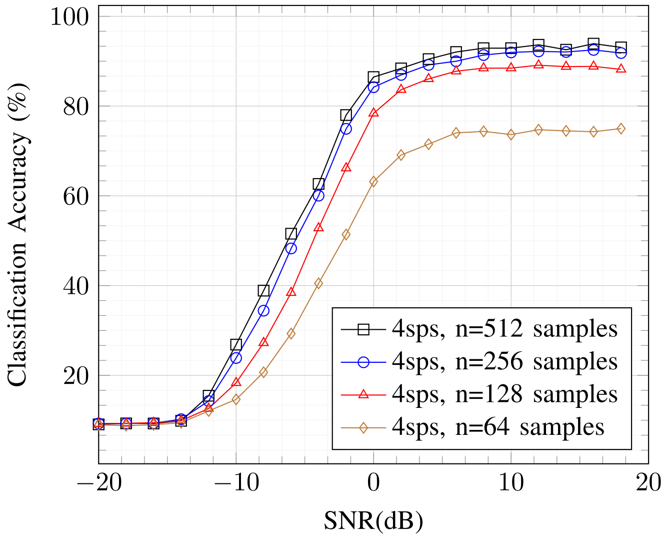

Fig. 7: Classification accuracy on modified RadioML dataset for 4 samples/symbol. The model is trained only on input sample lengths (n) from 128 to 512. The model gives close to 75% accuracy on 64 input samples which it is not trained for.

To understand the results better confusion matrices for the two layer LSTM model for various SNRs are also included. It can be seen in Figure 3 that at a high SNR of 18dB the diagonal is much more sharp even though there are difficulties in separating AM-DSB and WBFM signals. This is mainly due to the silence periods of audio as the modulated signals are

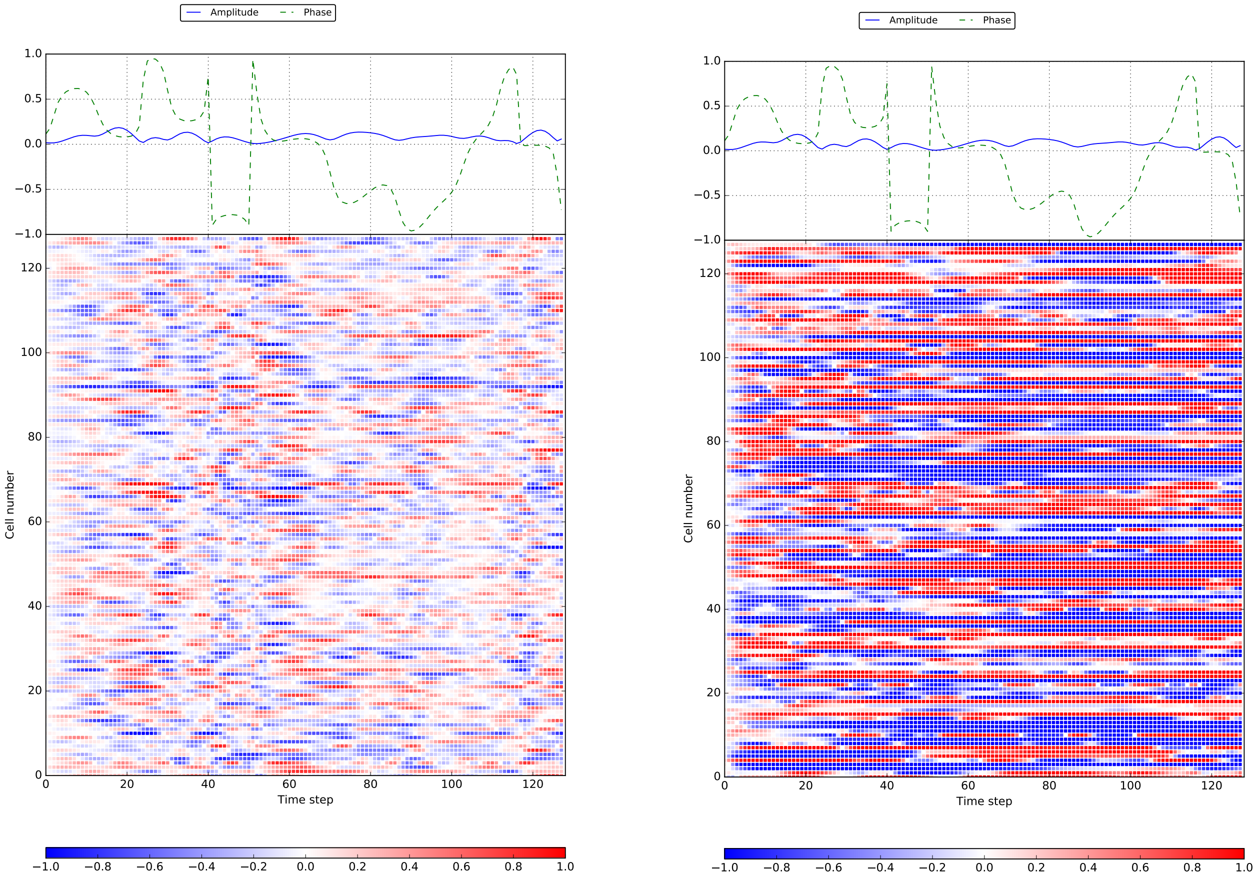

Fig. 8: Layer-1 (left) and layer-2 (right) LSTM temporal activations for a QAM64 input vector. On the top the amplitude and phase of the input signal is plotted at each time step. Blue denotes *tanh(c)* activations of value -1 and red denotes a value of +1.

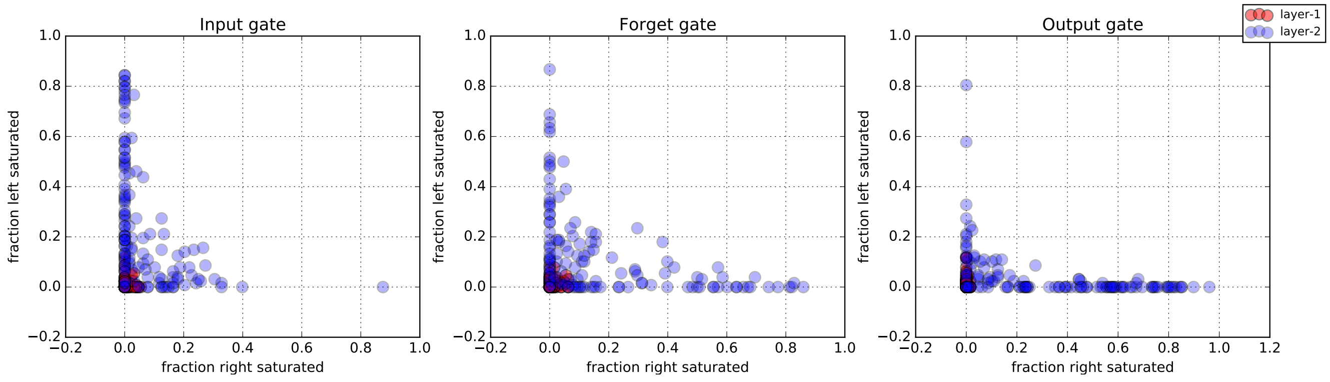

Fig. 9: Gate saturation plots for LSTM for the same a QAM64 input vector. A circle represents a gate in a particular LSTM cell.

generated from real audio streams. Similarly in Figure 4, at 0dB SNR it is noticed that there is some level of confusion regarding QAM16 and QAM64 as the former is a subset of the the latter. The confusion increases further at low SNRs as shown in Figure 5. From these basic analysis it is clear that deep complex structures as mentioned in [14] are not required to achieved good SoA classification accuracy at high SNRs. However, use of convolutional layers might turn useful at low SNRs as reported in [14]. In our experiments we also noticed that simply providing IQ samples to the LSTM model yielded poor results while normalized amplitude and phase interpretation provided good results. The models even failed to reduce the training loss when fed with time domain IQ samples, giving a constant accuracy of 9% on the radioML dataset, as the LSTMs were not able to extract any meaningful representations.

B. Classification accuracy on modified RadioML dataset

The same two layer LSTM model is trained on SNRs ranging from -20dB to 20dB and input sample lengths from 128 to 512 samples. The accuracy of the model is tested on the full range of SNRs and also on input sample length that is smaller than the training set (e.g, 64). It is evident from the results in Figures 6 and 7 that the classification accuracy improves as the model sees more modulated symbols. Even though the model is trained on varying data lengths from 128 to 512 samples, it gives an average accuracy of 75% with 64 samples and 4 samples per symbol scenario for which it was not trained, which confirms the model's generalization capabilities.

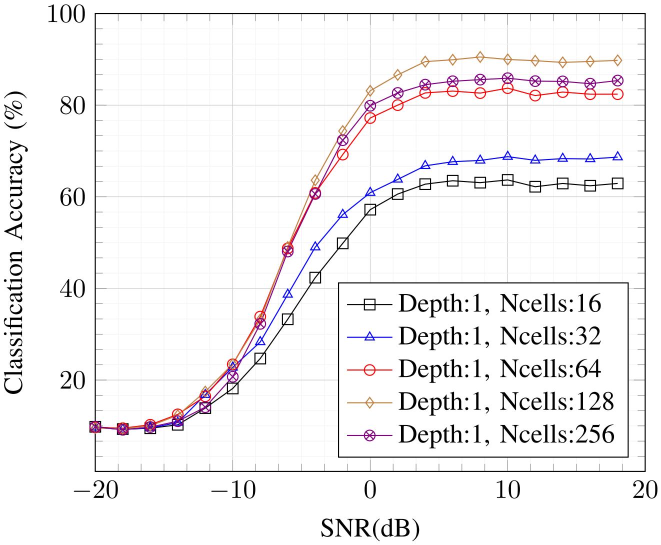

Fig. 10: Classification accuracy of single layer LSTM model for different cell size.

C. Learned representations

The inherent non-linearity and deep structures makes understanding the representations learned by LSTMs difficult. In order to obtain some good insights we use visualization techniques similar to the ones presented in [20]. Figures 8 and 9 presents the gate activation and saturation of the trained two layer LSTM model for a QAM64 input signal with 18dB SNR. A gate is said to be left saturated if its activation is less than 0.1 and right saturated if the activation is greater than 0.9. The fraction of time for which the gate is in left or right saturated mode in the entire 128 samples time is plotted in Figure 9. On the first layer, it can be noticed that all the three gates are confined close to the origin showing that they are not highly left or right saturated. The absence of right saturation in the first layer forget gates, confirms that the cells do not store information for long term. This is also visible in the activation plots of the first layer in Figure 8. The activations are short when compared to the second layer and it can be noticed in Figure 8 that many cell activations follow the input amplitude and phase changes in the input waveform. The second layer stores much long term dependencies from the fine grained representations generated from the first layer.

D. Effect of cell size and layer depth

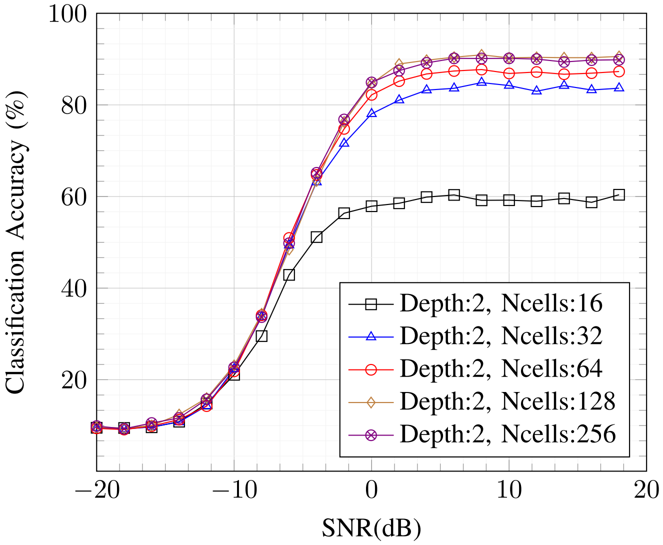

Fig. 11: Classification accuracy of two layer LSTM model for different cell sizes.

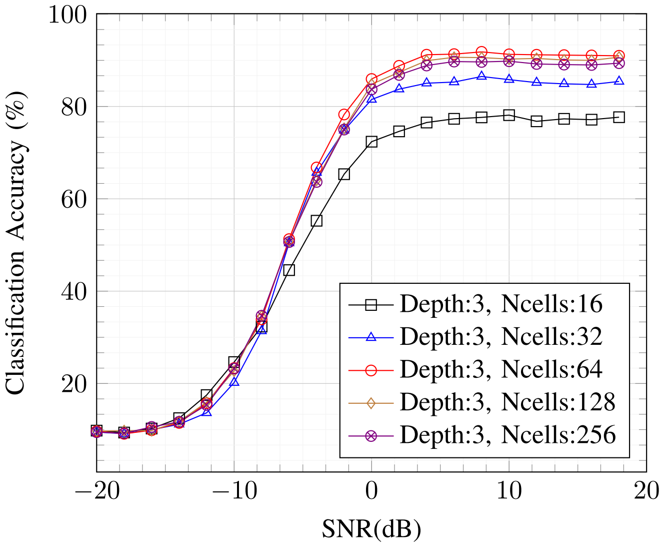

Fig. 12: Classification accuracy of three layer LSTM model for different cell sizes.

A comprehensive study is also performed to understand the effect of the number cells and layer depth on the model performance. The number of LSTM cells and layer depth are varied from 16 to 256 and 1 to 3 respectively. The models are trained on RadioML dataset on all SNRs. The accuracy levels for various layer depths are presented in Figures 10, 11 and 12. An initial analysis clearly shows that the model accuracy increases with increasing layer depth for mostly all cell sizes. It can be also noted that as the depth of the model increases, increasing the number of cells doesn't give much performance improvements. For instance, at depths 2 and 3 increasing the cell numbers from 128 to 256 doesn't provide any performance improvements.

VI. Resource Friendly Models

As mentioned earlier in the introduction, it is quite difficult to deploy SoA AMC algorithms on low-end distributed sensors such as the ones in Electrosense. We extend our study in two directions to reduce the resource requirements in terms of data transfer rate to the cloud, data storage and computational power. First, a study is conducted to understand to what extent technology classification can be done using averaged magnitude FFT data, as the FFT pipeline being the default enabled one in the Electrosense sensors with medium data transfer costs. As the sensors are sequentially scanning the spectrum, they are capable of generating magnitude spectrum information for wideband signals which is an added advantage. Second, the performance of quantized versions of the proposed deep learning models are analyzed which can reduce the computational cost of the models enabling deployment of these models on the sensors itself. The averaged magnitude FFT signal sent by the sensor, the selected dataset for testing the model, averaged magnitude FFT classification model, other quantized models and the classification results are detailed in the following subsections.

A. Received averaged magnitude FFT signal

Electrosense sensors scan the wireless spectrum sequentially. The sensor samples the wireless spectrum at a fixed sampling rate $N = 2.4\ MS/s$ tuned to a particular centre frequency f_x. As the sensor's sampling rate is limited, a wideband signal's magnitude spectrum can be received only by sequential scans to cover the entire bandwidth as given in equation 2.

$$\begin{aligned} R(f) = \frac{1}{M} \sum_{m=0}^{M} & |FFT_m(e^{-j2\pi f_1(t_0+mD_t)} s(t_0+mD_t)* \\ & c(t_0+mD_t) + n(t_0+mD_t))| \\ & || \frac{1}{M} \sum_{m=0}^{M} |FFT_m(e^{-j2\pi f_2(t_1+mD_t)} s(t_1+mD_t)* \\ & c(t_1+mD_t) + n(t_1+mD_t))| \\ & || \cdots \end{aligned} \tag{2}$$

In equation 2 $||$ represents the concatenation operation where the full bandwidth of the signal of interest is captured by a sequentially scanning sensor sampling at a lower sampling rate, similar to the Electrosense dataset mentioned in the following subsection. The averaged magnitude FFT signal at centre frequencies f_i, where $f_i \in (50\ MHz, 6\ GHz)$ based on the sensor sampling rate and frequency range, are sent to the cloud where they are concatenated together. In equation 2, M represents the magnitude-FFT averaging factor and t_x the sequential sampling time. For instance, $t_n = t_{n-1} + T$, where $T = MD_t$ is the amount of time spend at a particular centre frequency and D_t being the time for collecting *fft_size* samples for a single FFT input.

Technology	WFM, TETRA, DVB, RADAR, LTE, GSM
Time resolution	60s
Frequency resolution	10 kHz
Sensor sampling rate	2.4 MS/s
Scanning strategy	Sequential
FFT averaging factor	5
Sensor location and antenna type	Indoor, Omni-directional
Number of training samples	3100 vectors
Number of test samples	3100 vectors

TABLE III: Averaged magnitude FFT dataset parameters.

B. Electrosense dataset

Six commercially deployed technologies are selected to validate the classification accuracy using averaged magnitude FFT data as given in Table III. Over-the-air data from multiple Electrosense sensors are retrieved through the Electrosense API[3] with a spectral resolution of 10 kHz and time resolution of 60 seconds. The data is collected from sensors with omni-directional antennas which are deployed indoors. The sensors follow sequential scanning of the spectrum with an FFT size set to 256 giving a frequency resolution close to 10 kHz. With a FFT size of 256 and sensor ADC bit-width of 8, we get an effective bitwidth of 12 resulting in a theoretical dynamic range of 74dB. Practical dynamic range depends on the ADC frontend stages and the noise level, which may vary between 60 to 65dB. Five FFT vectors are averaged for reducing the thermal noise of the receiver. Some of the selected technologies such as LTE and DVB have an effective bandwidth which is higher than the sampling bandwidth of the of the low-end SDR. As the sensor is sequentially scanning, full spectrum shapes of these wideband signals are obtained by combining FFT outputs of these sequential scans. The entire data is split into two, one half for training and the other half for testing the model.

C. Averaged Magnitude FFT model

Sequentially sensed frequency spectrum data from the sensors contain signals of different bandwidth. The model should be able to process this variable length data and classify them to proper groups. We use the same LSTM model used for classifying complex input data as shown in Figure 1. The averaged magnitude FFT signal is fed to the model as a sequence as presented in Figure 1. The same LSTM model is chosen as it can handle variable length input signals and is also good at learning long term dependencies. The final output of the LSTM model after feeding n frequency bins is given as input to the softmax layer through a fully connected linear layer. The softmax layer outputs the probability $P(y = l|a; \theta)$ for $l \in \{0, 1, .., 5\}$ where a denotes the input averaged FFT bins, θ the model parameters and l the enumerated label for one of the six technologies as listed in Table III.

D. Classification results

An initial study is conducted to understand the technology classification accuracy of the averaged magnitude FFT model when compared to full IQ information. On the test dataset the

[3] https://electrosense.org/open-api-spec.html

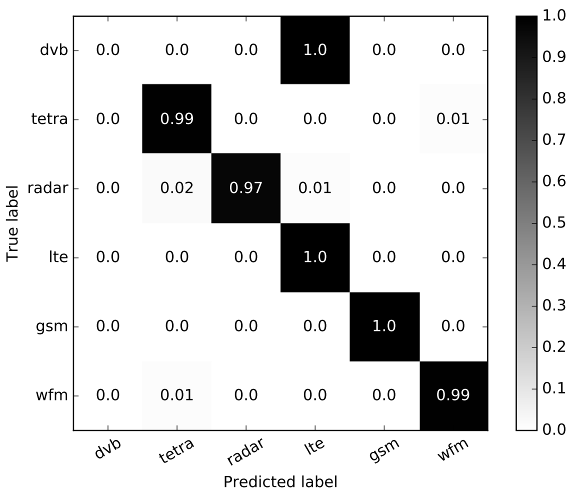

Fig. 13: Confusion matrix for technology classification

Layer-depth	Number of cells			
	16	32	64	128
1	70.02%	74.39%	79.38%	81.05%
2	72.69%	75.93%	80.92%	81.68%

TABLE IV: Classification accuracy on Electrosense dataset for varying layer depths and cell numbers.

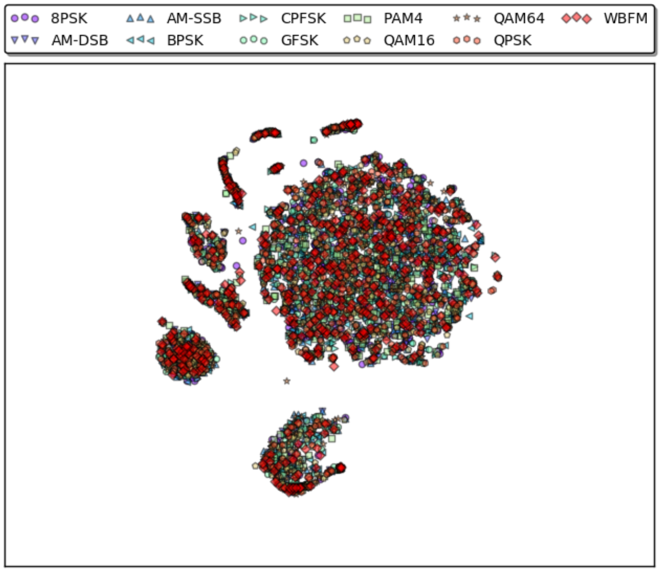

Fig. 14: t-SNE plot for magnitude FFT output on radioml dataset.

proposed model achieves a classification accuracy of 80%. The confusion matrix for the same is shown in Figure 13. From the confusion matrix it is clear that there is a large confusion between LTE and DVB. This is expected as the power spectra of both DVB and LTE looks very similar as both of them are based on Orthogonal Frequency Division Multiplexing (OFDM). We also investigated the effect of number of LSTM cells and layer depth on this dataset whose results are summarized in Table IV. Increasing the layer depth did not contribute significantly to the classification accuracy as there might be no more low level abstract features to be learned from the magnitude spectrum information. Furthermore, there are a large number of modulation schemes which exhibit the same power spectral density making averaged spectrum a sub-optimal feature for classification. For instance, the power spectral densities of different modulations schemes such as 8PSK, QAM16 and QPSK are identical once passed through a pulse shaping filter with a fixed roll-off factor. This can be theoretically shown and easily verified with manual inspection.

To further validate the argument, magnitude-FFT is calculated on the radioML dataset. To get a better understanding of the generated magnitude-FFT dataset, a visualization of a subset of the data in two dimensions is provided. For reducing the dimensionality of the data to 2 and for the ease of plotting, the t-SNE technique [21] is used. A small subset of the radioML dataset of 5000 vectors, containing 128 point magnitude FFT of all generated modulation schemes with varying SNR ranges from +20dB to +20dB are fed to the t-SNE algorithm. t-SNE is useful for a preliminary analysis to check whether classes are separable in some linear or nonlinear representation. The representation generated by t-SNE on the data subset is presented in Figure 14. It can be seen that the representation overlap is very high and t-SNE could not generate any meaningful clustering as the phase information is completely lost when computing magnitude-FFT, leaving identical magnitude spectrum for many modulation schemes. The obvious solution is to switch to IQ pipeline and deploy optimized versions of complex input signal models on the sensors itself, thus reducing the uplink data transfer rate. This is further investigated in the following subsection.

E. Quantized models

Deep learning models are processor intensive which makes them quite difficult to be deployed on low-end sensor networks. As mentioned in the introduction, transferring pure IQ data to the backend server is also not a scalable solution. Although our results indicate that some signals require IQ information for correct classification. To enable low-end sensor deployment, a feasibility study is conducted by quantizing the weights and activations of the newly proposed as well as baseline neural network models. Binarized networks can exceptionally reduce the required memory size for holding intermediate results and replace most of the arithmetic operations with bitwise operations [22]. For instance, when compared to a full precision network with 32 bits weights and activation, a binarized network only needs 32 times smaller memory resulting in a reduced required memory size and memory access cost. In [22] the authors had already noticed that binarizing LSTMs results in very poor accuracy. We confirm the same observation with our LSTM models too. However, models with binarized CNNs have been reported to provide accuracy close to their full precision variants. To validate this, the performance of the binarized baseline CNN model is also investigated on the radioML modulation dataset. Furthermore, by allowing more quantization levels on the LSTM models a higher accuracy can be achieved while still reducing the computational cost. Two quantized LSTM model variants are tested, one with ternary weights (-1, 0, +1) and full precision activation (TW_FA) and the other

Model	Weights count	MUL/ XNOR count	Memory
CNN: full precision	5451096	5451096	174.44 MB
CNN: binary	5451096	5451096	5.45 MB
LSTM: full precision	200075	262144	6.4 MB
LSTM: TW_4BA	200075	262144	400.15 KB
LSTM: TW_FA	200075	262144	400.15 KB

TABLE V: Memory size and number of multiplications for the discussed models (excluding nonlinear activations).

with ternary weights and four bits activation (TW_4BA). The accuracy results of these models are summarized in Figure 15. Results show that LSTM models with ternary weights and 4bit activation can provide close to 80% accuracy reducing the very high computational power required for full precision models. Binary CNN models also provided an accuracy level 10% below the full precision variants. We believe the classification accuracy can be further improved by proper hyper-parameter tuning and longer training.

The theoretical memory requirements for the trained weights along with number of multiplications required for the entire model, excluding activations, are summarized in Table V. A binarized neural network can drastically reduce the processing power requirements of the model. For instance, in a binarized network all weights and activation are either -1 or +1, replacing all multiply operations by XNORs. The multipy-accumulate, which is the core operation in neural networks, can be replaced by 1-bit XNOR-count operation [22]. Convolutions also comprises of multiply and accumulate operations which can also be replaced by its binarized variants. Thus the baseline CNN model can provide very good performance improvements on the general purpose ARM based Electrosense sensors. For the CNN models the convolutional layer output numbers are high, as we are not using any pooling layers, which accounts for the larger memory size in the succeeding dense layers. We would like to emphasize the fact that the given memory sizes are for the entire model and the weights that should be hold in the memory might vary based on practical implementations.

As binarized LSTM models did not provide good accuracy, we are forced to use 4-bit quantized variants of the same. Even though the performance improvements are not that extreme similar to binarized models, quantized LSTMs can also reduce the resource consumption. First of all, as no large dynamic range is required all the 4-bit multiply-accumulate operations can be implemented in fixed point arithmetic, which is much more faster in ARM CPUs when compared to their floating point versions. Secondly, routines can also be implemented to reduce the space requirements to hold intermediate results and the activations can be implemented as look-up tables. We would also like to emphasize the fact thatOn a special purpose hardware, such as FPGAs, quantized models can obviously reduce the space used and power-consumption as the multiply-accumulate units have smaller bit-widths.

VII. Conclusion and Future work

Wireless spectrum monitoring and signal classification over frequency, time and space dimensions is still an active research problem. In this paper we proposed a new LSTM based model which can classify signals with time domain amplitude and phase as input. SoA results on high SNRs (0 to 20dB) is achieved without using complex CNN-LSTM models as mentioned in [14]. Being a recurrent model we showed that the model can handle variable length inputs thus can capture sample rate variations effectively.

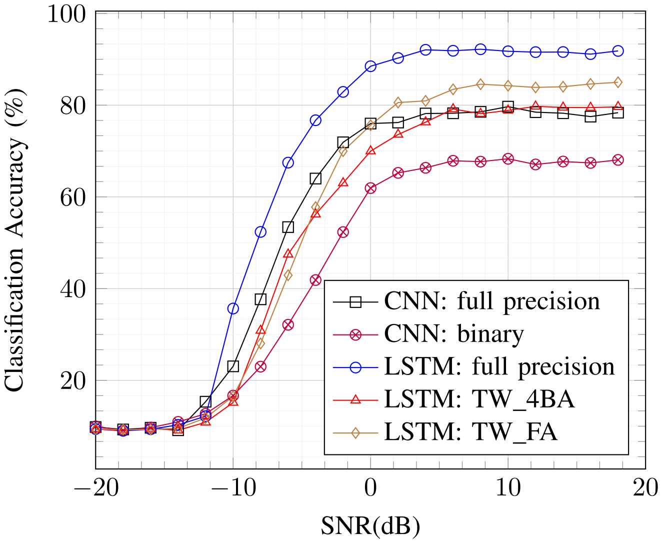

Fig. 15: Classification accuracy of two layer quantized models on RadioML dataset.

Though neural networks are good at function approximation, our experiments emphasize the fact that data preprocessing and proper representation is equally important. This claim is substantiated from our experiments with the LSTM model where the model gave poor results when fed with time domain IQ samples while it gave accuracies close to 90% for high SNRs when provided with time domain amplitude and phase information. As shown by various SoA models in speech and image domains, performance improvements are seen with increasing layer depth which saturates after a few levels. In addition, we showed that basic technology classification is achievable by only using averaged magnitude FFT information over a distributed set of sensors complying with the uplink bandwidth resource constraints. Furthermore, experiments showed that quantized LSTM models can achieve good classification results thus reducing the processing power requirements at the cost of 10% accuracy loss. This allows the deployment of these models on low cost sensors networks such as Electrosense enabling a wide area deployment. It is also remarkable that these deep learning models can classify signals with a fewer number of samples when compared to the expert feature variants, such as cyclic frequency estimators, enabling faster classification.

Although the LSTM models perform very well at high SNR conditions, CNN models seems to provide an additional 5-10% accuracy on the low SNR conditions (SNRs below -2dB) as shown in [14]. Even though we are not able to replicate the results in [14] (because of hyperparameter tuning), it is reasonable to conclude that the learned filters in CNN for a fixed sample rate might give performance improvements for low SNR values. Furthermore, all the implemented code for the proposed models are made publicly available for reproducing and verifying the results presented in this paper and for supporting future research.

Low SNR performance of these SoA deep learning models could be further improved with the help of efficient blind denoising models. Models which can perform automated channel equalization and compensate receiver imperfections such as frequency offset can further improve the classification performance. The current radio deep learning models make use of layers which basically applies non-linearity after simple multiply-accumulate-add operations while it is well established in the research community that cyclic cumulants, which are generated by time-shifted multiplication and averaging of the input itself, performs well in the expert feature space. Deep learning models which can extract features similar to cyclic cumulants might improve the performance metrics.

The analysis wouldn't be complete without emphasizing the limitations of the SoA deep learning models. First, the current complex models are tested on a dataset with normalized bandwidth parameters. Real life transmitted signals generally have varying symbol rates and bandwidths. Even though the variable length LSTM model is shown to be capable of adapting to these scenarios, further analysis is required to validate the claim. In future, models that can handle all possible spread spectrum modulations should be also tested. Second, the generalization capabilities of these models should be further investigated, in terms of performance of these models in unknown channel conditions and modulation parameters. Finally, most of the successful SoA models are supervised models which requires labeled training data. Labeling is a very tedious task which projects the importance of semi-supervised models for classification tasks, which is also an active direction for future research. We believe deep learning models adapted to radio domain can help in understanding, analyzing and decision making in future complex radio environments.

References

[1] S. Rajendran, R. Calvo-Palomino, M. Fuchs, B. Van den Bergh, H. Cordobés, D. Giustiniano, S. Pollin, and V. Lenders, "Electrosense: Open and Big Spectrum Data," *ArXiv e-prints*, Mar. 2017.

[2] M. Zheleva, R. Chandra, A. Chowdhery, A. Kapoor, and P. Garnett, "Txminer: Identifying transmitters in real-world spectrum measurements," in *2015 IEEE International Symposium on Dynamic Spectrum Access Networks (DySPAN)*, Sept 2015, pp. 94–105.

[3] S. S. Hong and S. R. Katti, "Dof: A local wireless information plane," in *Proceedings of the ACM SIGCOMM 2011 Conference*, ser. SIGCOMM '11. New York, NY, USA: ACM, 2011, pp. 230–241. [Online]. Available: http://doi.acm.org/10.1145/2018436.2018463

[4] O. A. Dobre, A. Abdi, Y. Bar-Ness, and W. Su, "Survey of automatic modulation classification techniques: classical approaches and new trends," *IET Communications*, vol. 1, no. 2, pp. 137–156, April 2007.

[5] W. A. Gardner, "Signal interception: a unifying theoretical framework for feature detection," *IEEE Transactions on Communications*, vol. 36, no. 8, pp. 897–906, Aug 1988.

[6] Z. Yu, "Automatic modulation classification of communication signals," Ph.D. dissertation, Department of Electrical and Computer Engineering, New Jersey Institute of Technology, 2006.

[7] A. V. Dandawate and G. B. Giannakis, "Statistical tests for presence of cyclostationarity," *IEEE Transactions on Signal Processing*, vol. 42, no. 9, pp. 2355–2369, Sep 1994.

[8] W. Gardner, "Spectral correlation of modulated signals: Part i - analog modulation," *IEEE Transactions on Communications*, vol. 35, no. 6, pp. 584–594, June 1987.

[9] W. Gardner, W. Brown, and C.-K. Chen, "Spectral correlation of modulated signals: Part ii - digital modulation," *IEEE Transactions on Communications*, vol. 35, no. 6, pp. 595–601, June 1987.

[10] A. Fehske, J. Gaeddert, and J. H. Reed, "A new approach to signal classification using spectral correlation and neural networks," in *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005.*, Nov 2005, pp. 144–150.

[11] T. Yucek and H. Arslan, "Ofdm signal identification and transmission parameter estimation for cognitive radio applications," in *IEEE GLOBECOM 2007 - IEEE Global Telecommunications Conference*, Nov 2007, pp. 4056–4060.

[12] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, insight. [Online]. Available: http://dx.doi.org/10.1038/nature14539

[13] T. J. O'Shea, J. Corgan, and T. C. Clancy, "Convolutional radio modulation recognition networks," in *International Conference on Engineering Applications of Neural Networks*. Springer, 2016, pp. 213–226.

[14] N. E. West and T. J. O'Shea, "Deep architectures for modulation recognition," *CoRR*, vol. abs/1703.09197, 2017. [Online]. Available: http://arxiv.org/abs/1703.09197

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997. [Online]. Available: http://dx.doi.org/10.1162/neco.1997.9.8.1735

[16] T. J. O'Shea and N. West, "Radio machine learning dataset generation with gnu radio," in *Proceedings of the GNU Radio Conference*, vol. 1, no. 1, 2016.

[17] "GNU Radio website," http://www.gnuradio.org, accessed: 2017-04-04.

[18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2014. [Online]. Available: http://arxiv.org/abs/1412.6980

[19] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[20] A. Karpathy, J. Johnson, and F. Li, "Visualizing and understanding recurrent networks," *CoRR*, vol. abs/1506.02078, 2015. [Online]. Available: http://arxiv.org/abs/1506.02078

[21] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[22] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Quantized Neural Networks: Training Neural Networks with Low Precision Weights and Activations," *ArXiv e-prints*, Sep. 2016.