Transformer Network for Fast RF Signal Classification

Dariush Hassan, Jose Lorente, George Sklivanitis
Center for Connected Autonomy and AI, Florida Atlantic University, Boca Raton, FL 33431
E-mail: {dhassan2019, jlorente, gsklivanitis}@fau.edu

***Abstract*—In recent years, deep learning has proven to be an effective tool for the task of radio frequency signal classification. Architectures such as the convolutional and residual neural network have shown the greatest performance so far, but can require long training times. We propose the use of a fairly new architecture, the Transformer, which is based merely on attention mechanisms and can be easily parallelized, requiring far less training time. Our model provides a classification accuracy of 75% at high SNR, with only a training time of four seconds per epoch, proving to be almost 10x faster than current state of the art architectures.**

I. Introduction

Rapid radio spectrum characterization and radio-frequency signal classification in congested (and sometimes contested) environments plays an important role toward autonomous spectrum management and enforcement of policy/regulations for future spectrum sharing applications [1]. In parallel, high-quality spectrum analytics (at either the waveform/modulation or the network protocol or the device level) offer an opportunity to recognize unlicensed spectrum/interfering users, malfunctioning equipment and take action. Existing approaches require expensive, high-maintenance expert systems that rely on prior knowledge of signal properties, features and decision statistics and focus on energy detection, localization and classification of spectrum activity under simplified hardware, propagation and radio environment models. Additionally, characterization of spectrum activity and of the corresponding radio devices requires tuning to the band and signal of interest to perform comparisons with existing baseline signal databases, thus incurring significant computational power and implementation/deployment cost before taking further action.

In recent years, deep learning [2] has proven to be an extremely effective tool for this task. Although the sophistication of deep neural networks is unparalleled, the time it takes to train them can be very lengthy, even with expensive hardware. This is especially more true for situations where computational power is lacking. As a result, an individual using deep learning for signal classification is limited in two scenarios: one where they need experimental results with a narrow time constraint, and one where they would like to utilize online learning [3] in an efficient manner. The latter has demonstrated to be of higher significance when the model's adaptation to changing environmental conditions is necessary.

In this work, we propose to leverage the Transformer [4] -the first neural network architecture based solely on attention mechanisms– to reduce training time, while maintaining RF signal classification accuracy. Previously, Long Short-Term Memory (LSTM) networks [5] provided high accuracy for RF signal classification tasks, but came with some concerning properties. Since this work primarily focuses on fast training speeds, we will only discuss the main issue: the LSTM utilizes sequential processing, where vectors must be processed slice by slice. This means that the model can't be trained in parallel. On the other hand, Transformers process vectors as a whole. As a result, the Transformer trains significantly faster.

With all the excitement around this fairly new architecture, we decide to test it's application with the synthetic RF signal data set. Our hopes are that the Transformer will perform equally well to training without the traditional natural language data. We compare the performance of the Transformer network to that of other state-of-the-art deep neural network architectures, such as a Residual Network (ResNet) [6]. Our proposed Transformer model provides a classification accuracy of 75% at high SNR, with only a training time of four seconds per epoch.

The rest of the paper is organized as follows. Section II contains background of the experiment. The Transformer Network is detailed in section III. The experiments and results are reported in section IV. Finally, in section V, the final observations and further action items with the conclusion of the work.

II. Background

A. Deep Learning for RF Signal Classification

With recent successful leaps being made in deep learning, specifically in computer vision and natural language processing (NLP), researchers began investigating if this

This work was supported by the National Science Foundation under Grant ECCS-2030234.

technology could be utilized for advances in the signal classification domain as well. In [7] and [8], researchers lay out a broad survey on the deep learning algorithms that provide advantages over previous baseline signal classification approaches. Specifically, the CNN and ResNet, which are primarily used in image recognition, repeatedly show great promise in being state of the art for the task of RF signal classification [1], [9], [10].

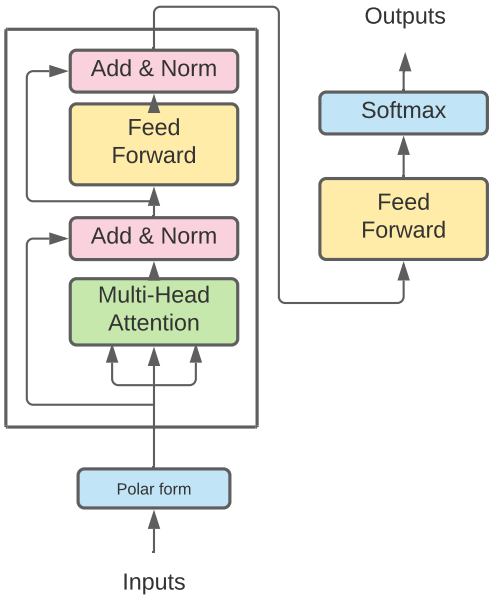

Fig. 1: Proposed radio Transformer architecture.

B. Attention Models for RF Signal Classification

In [11]–[14], researchers propose utilizing attention mechanisms for the task of RF signal classification. While the researchers show that it is highly beneficial to do so, attention is always accompanied by other complex layers, be it either convolutional or recurrent. In contrast, we introduce the method of solely using attention mechanisms for the task of RF signal classifications. To the best of our knowledge, this work is the first to approach this task in such a manner.

C. Speed Up Training

In [15], researchers also identify the problem of slow training speeds with deep learning approaches and make it an objective to investigate possible solutions. They identify and utilize three methods to decrease training times: PCA, subsampling [16], and SNR selection (training only with data of a selected SNR value). They apply these methods across multiple deep learning architectures, such as the CLDNN [17], CNN, ResNet, and long short-term memory networks (LSTM) [5] for the purpose of modulation classification. For our work, we choose to focus on PCA and SNR selection with the ResNet architecture, as these techniques were reported to perform the best in [15]. They find that a linear drop in training time is observed with a drop in the number of dimensions of the input vector. After replicating their experiments, we find very similar results. We compare the derived results of these techniques with those of the Transformer and present them in section IV.

III. Transformer Network

The Transformer was first introduced in [4] and was the first architecture based solely on attention mechanisms, dispensing with recurrence and convolutions entirely [4]. In this work, we only use the encoder section of the Transformer, due to this being a classification task. We only use one layer in the encoder, which consists of two sub-layers.

First, the input data $X \in \mathbb{C}^{N \times M \times T}$ where dimension N represents the number of samples, dimension M represents the IQ sample, and dimension T represents the sample length.

The polar form representation is computed with the amplitude and phase of the IQ sample. The amplitude and phase are obtained with the provided equations:

$$amp = |a + bi| = \sqrt{a^2 + b^2} \tag{1}$$

$$phase = \frac{arctan(a, b)}{\pi} \tag{2}$$

The data is then run through zero padding.

In [4], researchers initially use an input embedding layer, which transforms words into dense vector representations. Then, they add a positional encoding vector to the new word vector, which is a method to keep track of the order of words in the sentence. Since our sample data is already in vector form, there is no need to first pass the data through an embedding layer. We also find that there is no need for positional encoding either and obtain significant performance improvements without it. Instead, the data is passed straight to three fully connected layers to create the query, key, and value vectors. These vectors place emphasis on segments of the data that are more important than others.

In this work, we use scaled dot-product attention [4] where input contains the obtained queries and keys of dimension d_k, and values of dimension d_v, or in this case the size of our input data split into individual vectors. The set of all queries, keys, and values are stored respectively in matrices $\mathbf{Q}$, $\mathbf{K}$ and $\mathbf{V}$, in which the number of columns are determined by the batch size of the input data. The outputs are:

$$Attention(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = softmax\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}}\right)\mathbf{V} \quad (3)$$

In the first sub-layer, each attention function, called a head, is run in parallel. Then the outputs are concatenated, before being passed through a linear layer to receive the final values. Since there are multiple attention heads processing the data, it is called multi-Head attention [4]. This is more beneficial than using a single attention function, since each attention head learns information from different representation subspaces, therefore gaining more representation capability. In this work, we use two attention heads. This number of heads was the optimal choice in order to reduce redundant processing [18] and maintain an ideal training time to accuracy ratio. The results from this testing is displayed in Table II.

The second sub-layer is a simple fully connected feed forward network. A residual connection [6] is utilized around each sub-layer, as well as layer normalization [19]. Finally, for our classification purposes, we use an additional feed forward network for further processing and obtain outputs through a softmax layer. The architecture used in this work is portrayed in Fig. 1.

IV. Experiments

A. Dataset

We use the RadioML2016.10b data set generated in [20] as input. This data set has 128 samples time-domain IQ with 10 modulation types – as digital modulation BPSK, QPSK, 8PSK, QAM16, QAM64, BFSK, CPFSK, and PAM4, and analog modulation WB-FM, and AM-DSB. In addition, the samples are uniformly distributed in different SNR values from -20dB to 18dB, with steps of 2dBs. The whole dataset consists of 1,200,00 examples of size 2x128 that are subsequently divided into 50% for training, 50% for testing, and a validation split of 25% of the training set.

B. Experimental Setup

For all of our experiments, we use Keras with Tensorflow. We use a GPU server with a GTX 970 GPU. The models are trained with the Adam optimizer [21], a learning rate of 0.001, a batch size of 1024, and a categorical cross entropy loss function. We also use an early stopping mechanism from Keras that halts training when validation loss stops improving, to combat overfitting. This should be kept in mind when viewing total training time for the architectures provided. The ResNet and CNN networks we use to compare our results are inspired from [9]. For PCA purposes, we use the PCA function provided by the scikit-learn library. We experiment with a number of components in the range of 32 to 128, but use the high and low of that range for comparisons.

TABLE I: Architecture performance.

NN Model	*Test Accuracy (SNR=18dB)*	*Train Time (per epoch)*	*Total Train Time*
ResNet [1]	92%	39 s	1800 s
ResNet [1] (0dB, 18dB)	80%	3 s	212 s
Transformer [2]	75%	4 s	300 s
ResNet [1] (1/2 PCA)	69%	24 s	690 s

TABLE II: Transformer performance with changes in quantity of attention heads

Num. of Heads	*Test Accuracy (SNR=18dB)*	*Train Time (per epoch)*	*Total Train Time*
2	75%	4 s	300 s
4	73%	5 s	375 s
8	74%	5 s	375 s
16	74%	7 s	525 s

C. Experimental Results

The Transformer provides a classification accuracy of 75% at high SNR, with a training time of only four seconds per epoch. This is significantly faster than the baseline ResNet, which requires a training time of 39 seconds per epoch, but which also respectively provides a classification accuracy of 92% at high SNR. We also observe that with the Transformer we receive better performance when the samples are represented in polar form, in contrast to rectangular form. The Transformer is especially good at differentiating between different QAM constellations, due to it's ability to identify repeating patterns of changes in amplitude and phase [15].

With PCA, at a 1/2 dimensionality reduction the ResNet provides a classification accuracy of 69% at high SNR, with a training time of 24 seconds, and at a 1/16 dimensionality reduction it provides a classification accuracy of 52% at high SNR, with a training time of nine seconds. While we do see an impressive reduction in training time with PCA, the accuracy steers behind when trying to get to the four seconds per epoch mark of the Transformer.

When the ResNet is trained only on data from the selected 0dB and 18dB SNR values, it provides a classification accuracy of 80% at high SNR, with a training time of three seconds per epoch. This is an impressive training speed, with a relatively high accuracy rate compared to the Transformer. Although, we find that when training with such a low data sample size, the model tends to initially show vast differences between the training and validation loss. This tells us that the

model significantly overfits the data before adjusting itself, but still has volatility spikes in validation loss afterwards. This is shown in Fig. 4 and compared with the Transformer learning curve. We also find that the Transformer provides higher classification accuracy than the SNR selection model at low SNR, which is shown in Fig. 3.

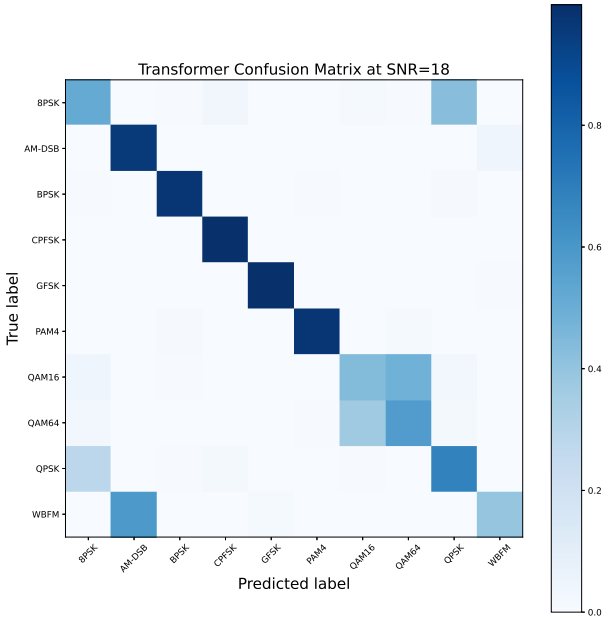

Fig. 2: 10-modulation confusion matrix for Transformer trained on the entire range of SNRs from -20:18 dB and tested with SNR=18 dB data.

Fig. 2 depicts the confusion matrix for the proposed Transformer network for signals of SNR=18dB, across 10 different modulations.

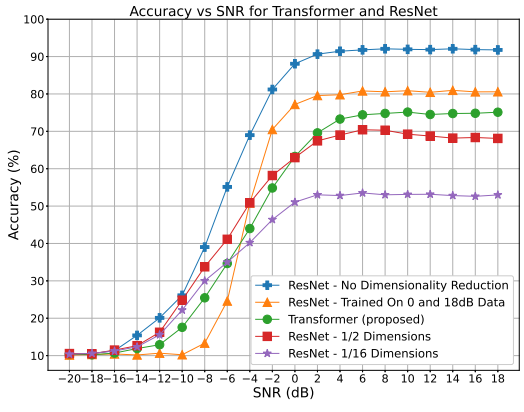

Fig. 3: XXXX

Fig. 3 illustrates testing accuracy across all SNR values for the baseline ResNet and Transformer. It also includes accuracy for 1/2 and 1/16 data dimensionality reduction with PCA, applied to the ResNet, as well as the accuracy obtained when training only on 0dB and 18dB data.

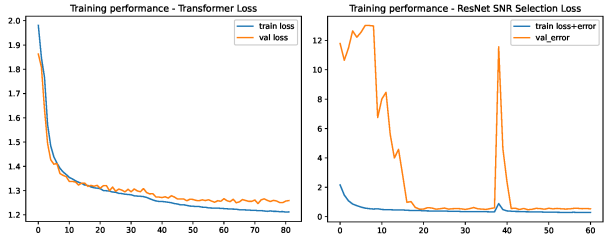

Fig. 4: Learning curves for the Transformer and the ResNet trained on 0dB and 18dB SNR data.

V. Conclusions

We show that the Transformer can be an effective neural network architecture for radio frequency signal classification and demonstrate it to be an ideal candidate where fast training times are a necessity. While the 75% classification accuracy is certainly behind current state of the art architectures for this task, we will continue to work on improving it by further tuning of certain hyper-parameters. We will also investigate further training and testing on new data sets, such as the RadioML2018.10A data set [9], and observe if the Transformer maintains valid performance. We hope others in this field recognize the power of this parallelizable architecture for fast training times and build upon our work.

References

[1] S. Zhou, Z. Yin, Z. Wu, Y. Chen, N. Zhao, and Z. Yang, "A robust modulation classification method using convolutional neural networks," *EURASIP Journal on Advances in Signal Processing*, vol. 2019, no. 1, 2019.

[2] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[3] D. Sahoo, Q. Pham, J. Lu, and S. C. H. Hoi, "Online deep learning: Learning deep neural networks on the fly," 2017.

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.

[5] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015.

[7] X. Li, F. Dong, S. Zhang, W. Guo, and H. Wu, "A survey on deep learning techniques in wireless signal recognition," vol. 2019, Jan. 2019. [Online]. Available: https://doi.org/10.1155/2019/5629572

[8] R. Zhou, F. Liu, and C. W. Gravelle, "Deep learning for modulation recognition: A survey with a demonstration," *IEEE Access*, vol. 8, pp. 67 366–67 376, 2020.

[9] T. J. O'Shea, T. Roy, and T. C. Clancy, "Over-the-air deep learning based radio signal classification," *IEEE Journal of Selected Topics in Signal Processing*, vol. 12, no. 1, pp. 168–179, 2018.

[10] T. J. O'Shea and J. Corgan, "Convolutional radio modulation recognition networks," *CoRR*, vol. abs/1602.04105, 2016. [Online]. Available: http://arxiv.org/abs/1602.04105

[11] T. J. O'Shea, L. Pemula, D. Batra, and T. C. Clancy, "Radio transformer networks: Attention models for learning to synchronize in wireless systems," 2016.

[12] R. Luo, T. Hu, Z. Tang, C. Wang, X. Gong, and H. Tu, "A radio signal modulation recognition algorithm based on residual networks and attention mechanisms," 2019.

[13] R. Zhang, Z. Yin, Z. Wu, and S. Zhou, "A novel automatic modulation classification method using attention mechanism and hybrid parallel neural network," *Applied Sciences*, vol. 11, no. 3, 2021. [Online]. Available: https://www.mdpi.com/2076-3417/11/3/1327

[14] Y. Chen, W. Shao, J. Liu, L. Yu, and Z. Qian, "Automatic modulation classification scheme based on lstm with random erasing and attention mechanism," *IEEE Access*, vol. 8, pp. 154 290–154 300, 2020.

[15] S. Ramjee, S. Ju, D. Yang, X. Liu, A. E. Gamal, and Y. C. Eldar, "Fast deep learning for automatic modulation classification," 2019.

[16] Y. C. Eldar, *Sampling Theory: Beyond Bandlimited Systems*, 1st ed. USA: Cambridge University Press, 2015.

[17] T. N. Sainath, O. Vinyals, A. Senior, and H. Sak, "Convolutional, long short-term memory, fully connected deep neural networks," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4580–4584.

[18] P. Michel, O. Levy, and G. Neubig, "Are sixteen heads really better than one?" 2019.

[19] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," 2016.

[20] T. O'Shea and N. West, "Radio machine learning dataset generation with gnu radio," *Proceedings of the GNU Radio Conference*, vol. 1, no. 1, 2016. [Online]. Available: https://pubs.gnuradio.org/index.php/grcon/article/view/11

[21] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017.