IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 4, 2022, accepted March 2, 2022, date of publication March 14, 2022, date of current version March 22, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3159339

Long Short-Term Memory Recurrent Neural Network for Automatic Speech Recognition

JANE ORUH, SERESTINA VIRIRI, (Senior Member, IEEE), AND ADEKANMI ADEGUN
School of Mathematics, Statistics and Computer Science, University of KwaZulu-Natal, Durban 4000, South Africa
Corresponding author: Serestina Viriri (viriris@ukzn.ac.za)

ABSTRACT Automatic speech recognition (ASR) is one of the most demanding tasks in natural language processing owing to its complexity. Recently, deep learning approaches have been deployed for this task and have been proven to outperform traditional machine learning approaches such as Artificial Neural Network (ANN). In particular, deep-learning methods such as long short-term memory (LSTM) have achieved improved ASR performance. However, this method is limited to processing continuous input streams. Traditional LSTM requires four (4) linear layers (multilayer perceptron (MLP) layer) per cell with a large memory bandwidth for each sequence time step. LSTM cannot accommodate the many computational units required for processing continuous input streams because the system does not have sufficient memory bandwidth to feed the computational units. In this study, an enhanced deep learning LSTM recurrent neural network (RNN) model was proposed to resolve this shortcoming. In the proposed model, the RNN is incorporated as a ''forget gate'' to the memory block to allow the resetting of cell states at the beginning of the sub-sequences. This enables the system to process continuous input streams efficiently without necessarily increasing the required bandwidths. In the proposed model, the standard architecture of the LSTM network is modified to effectively use the model parameters. Some CNN-based and sequential models were used on the same dataset, and the models were compared with the proposed model. LSTM-RNN outperformed the other deep learning models with an accuracy of 99.36% on the well-established public benchmark spoken English digit dataset.

INDEX TERMS Automatic speech recognition, deep supervised learning, recurrent neural network, spoken English digit dataset.

I. INTRODUCTION

Speech comprises a sequence of uttered sounds, which are also known as phonemes. Speech is used to transmit information from one speaker to the other. When the signal from speech is converted into a meaningful message or text, it is called Automatic Speech Recognition (ASR) [1]. The recognition of isolated spoken digits has proven to be a challenging task in ASR owing to its complexity.

A. BACKGROUND

Deep learning is an emerging technology that is regarded as auspicious direction for attaining a height in artificial intelligence [2]. At present, deep learning has been deployed in a wide range of domains, including bioinformatics, computer vision, machine translation, dialogue systems, and natural language processing. One area that has been transplanted by this technology is ASR [3]. In recent times, deep learning has been deployed for ASR [4]–[6], speech recognition systems [7], [8], speech enhancement problems [9]–[11] and has outperformed traditional machine learning approaches such as artificial neural networks (ANN).

Although ANNs can categorize small acoustic-phonetic units such as separate phonemes, they cannot model long-term dependencies in acoustic signals [12]. However, deep neural networks (DNNs) provide restricted temporal modeling of the acoustic frames. However, they cannot deal with data that have longer-term dependencies. Feed-forward neural networks can be expanded for an effective classification. To achieve this, it will require feeding the signals that were fed back into the network from previous time steps. Such networks with recurrent interconnections are called recurrent neural networks (RNNs) [13], [14]. RNNs are restricted because they look back in time for roughly ten time-steps [15].

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Liu.

The connections in RNNs are cyclic, which makes them a dynamic mechanism for modeling sequence data [16]. Thus, RNNs use a dynamic contextual window against a static fixed-size window over sequences. Unfortunately, RNNs are difficult to train using gradient-based back propagation through time (BPTT) [17] and are not likely to demonstrate the full power of recurrent models. This is because of the well-known vanishing and exploding gradient problems [18].

One way to improve the training of RNNs is to use an optimization algorithm with higher-order approximations [19]. However, it is normally at the cost of remarkably increased computational costs, which makes the approach unattractive for language modeling which requires an enormous amount of training data [20]. Hochreiter and Schmidhuber [21] proposed the long short-term memory (LSTM) architecture as a solution to resolve this challenge. LSTMs are specifically designed to avoid the long-term dependency problem. Remembering information for long periods is their default practice. LSTMs have many advantages over conventional feed-forward neural networks and the RNN. This is because of their ability to remember patterns for long durations.

LSTM is a type of recurrent neural network with a strong ability to learn and predict sequential data. Sequence prediction is a long-standing problem. With recent advancements in the field of data science, it is found that for practically all sequence prediction problems, LSTM has been observed as the most successful approach [22]. The core idea behind LSTMs is the cell state and its gates. The cell state conveys the relevant information to the sequence chain.

B. DEEP LEARNING BASED METHODS FOR AUTOMATIC SPEECH RECOGNITION

1) RECURRENT NEURAL NETWORK

Sak *et al.*'s [16] work was found to have introduced the first implementation of LSTM networks on a large-vocabulary Google voice search speech recognition task. They presented an LSTM RNN model architecture that makes use of the model parameter more advantageous for training acoustic models for large-vocabulary tasks. They trained and compared LSTM, RNN, and DNN models using different numbers of parameters and configurations. The results of their experiment show that LSTM models converge quickly and perform best when applied to moderately small-sized frameworks.

Geiger *et al.* [23], proposed an LSTM RNN in a hybrid acoustic modeling structure for robust speech recognition in an environment affected by noise and reverberation. The experiment was conducted using the database of the medium-vocabulary recognition track of the 2nd CHiME speech separation and recognition challenge. The authors compared state prediction networks with networks that predict phonemes using LSTM networks. The result showed that with LSTMs, state prediction is better than networks predicting phonemes.

Recently, there has been a remarkable improvement in RNN-HMM hybrid systems with deep bidirectional (DB) LSTM-based acoustic models for CD phonetic units, states for the LSTM output space and distributed training methods to perform large-scale modeling [24].

2) GATED RECURRENT UNITS (GRU)

Modified gated recurrent units (GRU), known as light-gated recurrent units (Li-GRU), were proposed in [25] for automatic speech recognition across various tasks, features, conditions, and paradigms. The experiment was conducted using TIMIT, DIRHA-English, CHiME, and TED-talk speech recognition corpus in various subsections. The proposed method outperformed the standard GRU in terms of recognition and computational performance and significantly reduced the per-epoch training time by 30% compared to the standard GRU.

Feng *et al.* [26] proposed a projected minimal gated recurrent unit (PmGRU) an improved version of the mGRUIP with context module (mGRUIP-Ctx) for speech recognition acoustic model on five different ASR tasks. The proposed model showed a significant reduction in the word error rate (WER) compared to the WER of mGRUIP-Ctx.

3) END-TO-END SPEECH RECOGNITION

Graves *et al.*'s [7] showed that end-to-end training methods such as connectionist temporal classification (CTC) can be used to train RNNs for sequence-labelling tasks on the TIMIT corpus, where the input-output alignment is not known. They suggested that combining these methods with LSTM RNN architecture is likely to yield state-of-the-art results.

Hannun *et al.* [27] used of a 5-layer RNN with a bidirectional recurrent layer trained with CTC loss and a language model to credibly fix the phonetic transcriptions. The results of this approach exceeded the best results on the switchboard dataset.

Li [28] provided a detailed overview of E2E models and feasible technologies that makes E2E models outperform hybrid models in the industrial world.

4) DEEP BELIEF NETWORK

Mohamed *et al.* [29] conducted the first successful experiment using a hybrid DNN-hidden makov model (HMM) with an acoustic model based on deep belief network (DBN) on the TIMIT dataset. His results outperformed those of previous studies using the same dataset. Over the years, other researchers have used restricted BoltzMann machines (RBMs) and DBNs techniques to explore and demonstrate the results of using them in speech recognition tasks. [30]–[33].

5) CONVOLUTIONAL NEURAL NETWORK

Abdel-Hamid *et al.*'s [34] work using CNN outperformed previously published results used in the hybrid NN-HMM model. Their experimental results showed a remarkable improvement in the recognition performance using local

filtering and max-pooling and achieved over a 10% relative error reduction on the core TIMIT test sets compared to constant neural networks (NNs) with the same number of hidden layers and weights. Abdel-Hamid *et al.*'s work in [35] also, investigated convolution over the time and frequency axes simultaneously.

Sainath *et al.*'s [36] investigated the most suitable approach for making CNNs a more capable model for large-vocabulary continuous speech recognition (LVCSR tasks) than DNNs. They also investigated the actions of NN features extracted from CNNs on a variety of LVCSR tasks, which were compared with DNNs and GMMs. The results of their experiment shows 13-30% and 4-12% relative improvement over GMMs and DNNs respectively, on the 400-hr broadcast news and 300-hr switchboard task. In addition, an experimental investigation of CNN-based acoustic models for low-resource languages has proven that CNNs are better than DBNs in terms of robustness and improved generality [37].

C. PROPOSED MODEL

A modified LSTM RNN model was proposed in this work to perform sequence prediction that will make use of deep supervised learning on the benchmark spoken English digit dataset. The effectiveness of the model will be estimated with respect to training and validation accuracy, and the results will be compared with other studies that used deep learning models for various speech recognition tasks. In addition, the classification performance of the model was evaluated to obtain the average score for precision, recall, f1-score using a confusion matrix. The choice of LSTM RNN is based on the fact that LSTM consists of a standard RNN built up with ''memory units'', that specializes in transferring long-term information, also with a set of ''gating'' units that allows memory units to carefully interrelate with the normal RNN hidden state [19].

Several studies have been conducted using LSTM RNN. LSTM has achieved virtually all thrilling results based on RNNs. Thus, it has become the centre of deep learning in ASR systems [38]. LSTMs have been used extensively in speech recognition tasks because of their powerful learning ability [7], [16], [23], [39], [40], [25], [41], but this is the first time LSTM RNN will be used on the spoken English digit speech recognition dataset.

The contributions of this paper can be summarized as follows;

1) This study reviews existing deep learning methods for sequential data and highlights the limitations of traditional LSTM in processing continuous input streams.
2) A recurrent neural network (RNN) is incorporated as a forget gate to the memory block to allow resetting of the cell states at the beginning of the subsequences.

II. RELATED WORK

Graves *et al.* [7] showed that end-to-end training methods like CTC can be used to train RNNs for sequence labelling tasks on the TIMIT corpus. Merging these methods with LSTM RNN architecture will likely yield state-of-the-art results. In this study, the standard LSTM RNN training method was used to obtain a 99.36% accuracy for the sequence prediction speech recognition task.

Sak *et al.* [16] work, was found to have introduced the first implementation of LSTM networks on the Google voice search speech recognition task. Their proposed model architecture improved the use of model parameters while training acoustic models. The model trained and compared LSTM, RNN, and DNN models with various numbers of parameters and configurations. The results show that the LSTM model was the fastest to converge and performed best when applied to a moderately small-sized sized framework.

Geiger *et al.* [23], proposed an LSTM RNN in a hybrid acoustic modelling structure for robust speech recognition in an environment affected by noise and reverberation. The experiment was conducted using the database of the 2nd CHIME medium-vocabulary recognition track. The authors compared state prediction networks and networks that predict phonemes using LSTM networks. The results of their experiment showed that with the use of LSTMs in a hybrid or double-stream system, the state prediction network is superior to the network prediction phonemes.

He and Droppo [40] proposed a generalized LSTM known as the (G)LSTM-DNN. The strength of the proposed model was first analyzed using a normal 80-hour LVCSR task AMI and then applied to the 2000-hour Switchboard data set. The results of their experiment showed that the proposed (G)LSTM-DNN performs better with more layers and achieved a relative word error rate reduction of 8.2% on the 2000-hour Switchboard data set. One issue discussed in their work is that the model's performance comes at the cost of a large number of parameters, and it is noteworthy to find a system that will save the parameters while maintaining its modeling power.

Tachioka and Ishii [39], proposed LSTM RNN for Bandwidth Extension (BWE) on the TIMIT phoneme recognition task. The proposed LSTM RNN-based BWE was compared to standard gaussian mixture model (GMM)-based BWE. The results of the experiment showed that LSTM RNN-based BWE was more powerful than the GMM-based BWE. In addition, they added that for ASR purposes, it is better to predict MFCC features directly than to predict Mel-cepstrum features. The model used in this study has used the MFCC features for its prediction.

The authors proposed an LSTM-RNN for deep sentence embedding [42]. Here, the RNN is used to accept each word in a sentence sequentially and then map alongside the contextual information into a latent space in a recurrent form. Furthermore, LSTM cells were incorporated into the RNN model (LSTM-RNN) to address the weakness of the RNN in learning long-term memory. As a result of the non-availability of labeled data, user click-through data were used and the model was trained in a weakly supervised form. The proposed LSTM RNN used in this work for the sequence

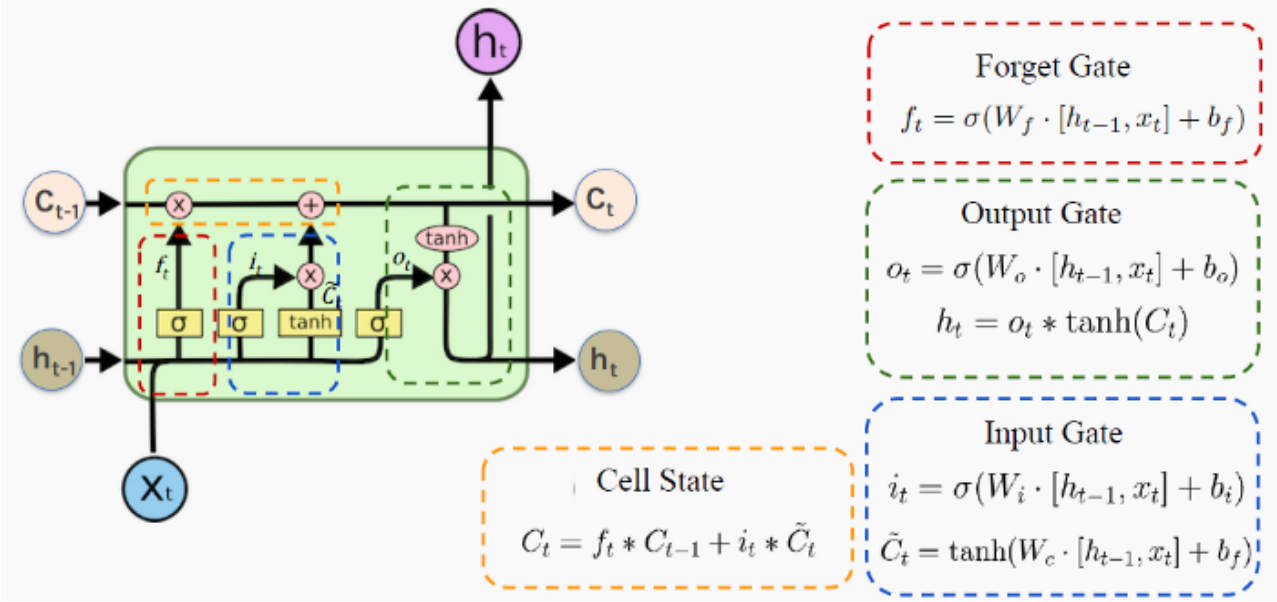


FIGURE 1. The standard LSTM RNN architecture [46].

prediction of the spoken English data, however, was, trained using a strong deep supervised network that helped obtain optimal accuracy.

One of the RNN models, gated recurrent units (GRUs), was revised, and a simpler architecture was proposed in [25] for automatic speech recognition across various tasks, features, conditions and paradigms. The experiment was conducted using TIMIT, the DIRHA-English, CHiME, and TED-talk speech recognition corpus, in various subsections. The proposed method has outperformed the standard GRU in terms of recognition and computational performance and significantly reduced the per epoch training by 30% compared to the standard GRU.

WAZIR and CHUAH [41] proposed an Arabic digits speech recognition model using an RNN with LSTM cells. Their model exhibited an overall accuracy of 94.00% for model training and 69.00% for the model testing. When the standard LSTM was implemented in the spoken English digit speech recognition task, the overall accuracy of 99.36% was achieved for model training, as demonstrated in this work.

III. METHODS AND TECHNIQUES

A. THE STANDARD LSTM ARCHITECTURE

The main structure of LSTM consists of unique segments known as ''memory blocks'' in the hidden layer. The first type of LSTM block consists of cells and the input and output gates. The standard structure of LSTM has a limitation, which was addressed for the first time in [43] through the establishment of a ''forget gate'' that will empower LSTM to adjust its state. The ''forget gate'' f_t resets the cell variable leading to the 'forgetting' of the stored input c_t, whereas the input and output gates manage the reading of inputs from the feature vector, x_t, and writing of output to h_t, respectively [21].

The gates regulate the action of the memory block whereas the ''forget gate'' weighs the information inside the cells, such that anytime previous information becomes unimportant for some cells, it will reset the state of the different cells. ''Forget gates'' also enables continual prediction [44], by making cells forget their previous state, thereby restricting biases in prediction.

The computation operation within an LSTM block is as follows: Input values can only be conserved in the cell state if the input gate allows them. Its input value of i_t and the expected value of the memory cells, $\tilde{C}_t$, at time step, t, is calculated as follows:

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i) \quad (1)$$

$$\tilde{C}_t = tanh(W_c[h_{t-1}, x_t] + b_c) \quad (2)$$

$W[h_{t-1}, x_t]$ and b represent the weight matrices and bias, respectively. The forget gate controls the weight of the state cell unit, and the value of the forget gate is computed as:

$$f_t = \sigma(W_f.[h_{t-1}, x_t + b_f) \quad (3)$$

By this process, the new state of the memory cell is being updated as

$$\tilde{C}_t = i_t.\tilde{C}_t + f_t.\tilde{C_{t-1}} \quad (4)$$

Given a new state memory cell, the output value of the gate is computed as

$$o_t = \sigma(W_x.[h_{t-1}xX_t] + b_o) \quad (5)$$

The final output value of the cell can then be explained as

$$h_t = o_t * tanh(c_t) \quad (6)$$

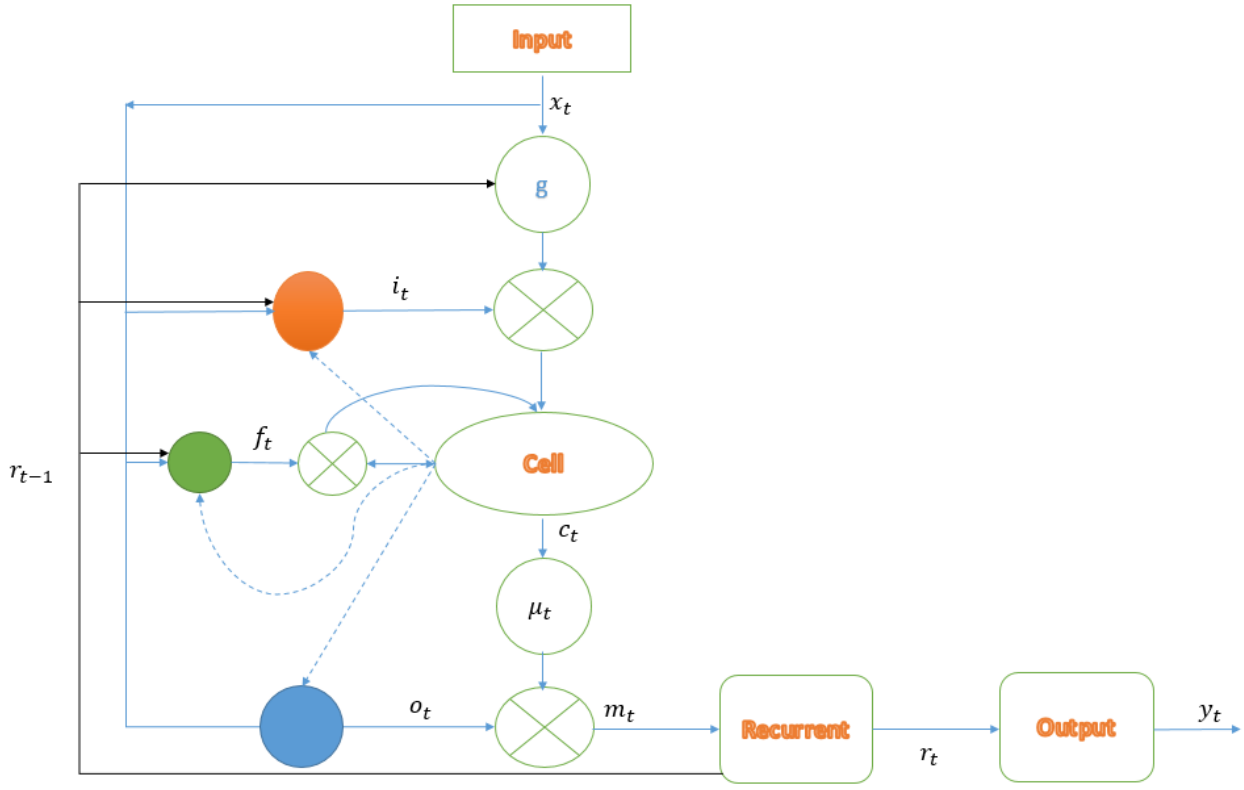


FIGURE 2. Proposed LSTM RNN architecture.

σ, g and h are point-wise nonlinear activation functions, and i, f, o and c are the input, forget, output gate and the cell activation vectors, respectively. All features of the LSTM network architecture can be trained using the sigmoid(ϕ) and *tanh* activation functions.

With this structure in place, the network can store inputs for a long period, thus utilizing a trained number of extended temporal situations [23]. Additionally, the recent LSTM architecture accommodates "peephole connections" from its internal cells, which learns the accurate timing of the output [45]. The standard LSTM Structure is illustrated in Figure 1.

B. THE PROPOSED LSTM ARCHITECTURE

The proposed model avoids the problem of processing continuous input streams that are not segmented into subsequences. This means that streams that are not theoretically subdivided into smaller units are easily processed by the network. The proposed model in turn integrates RNN as a "forget gate" to the memory block to permit cell states to be reset at the beginning of sub-sequences. There is a need to reset the network's internal state to prevent the cell state from growing indefinitely, which may eventually cause the network to break. The memory blocks use their memory cells to store the network's temporal state, and distinctive multiplicative units known as gates to control information flow. The proposed model architecture effectively use model parameters by modifying the standard LSTM architecture. This modification in the LSTM architecture causes changes in the computational cost because of the increase in the computational resources as a result of adding an RNN as a forget gate. Figure 2 shows the proposed LSTM RNN memory block.

Supervised learning is a learning technique that use labelled data. For a supervised deep learning technique, the setting comprises a set of inputs with complementary output $(x_t, y_t) \sim p$. For instance, if for an input x_t, the smart agent predicts $\hat{y} = (x_t)$, and then the agent will obtain a loss value $l = (y_t, \hat{y}_t)$. After successful training, the agent repeatedly adjust the network parameters to obtain an improved approximation of the output, similar to the deep supervised approach used in this study [47].

Algorithm 1 represents the algorithm of the proposed Model

IV. EXPERIMENTS

A. DATASET

The dataset is a *well-established publicly available dataset* under Pannous, a collaboration working on improving speech recognition [48], from the librosa library [49]. Speech data were downloaded using an MFCC batch generator. The file consists of a group of wav files that are in batches alongside its related labels. The audio dataset was pre-processed using the librosa library, Python's library dedicated to analyzing sounds.

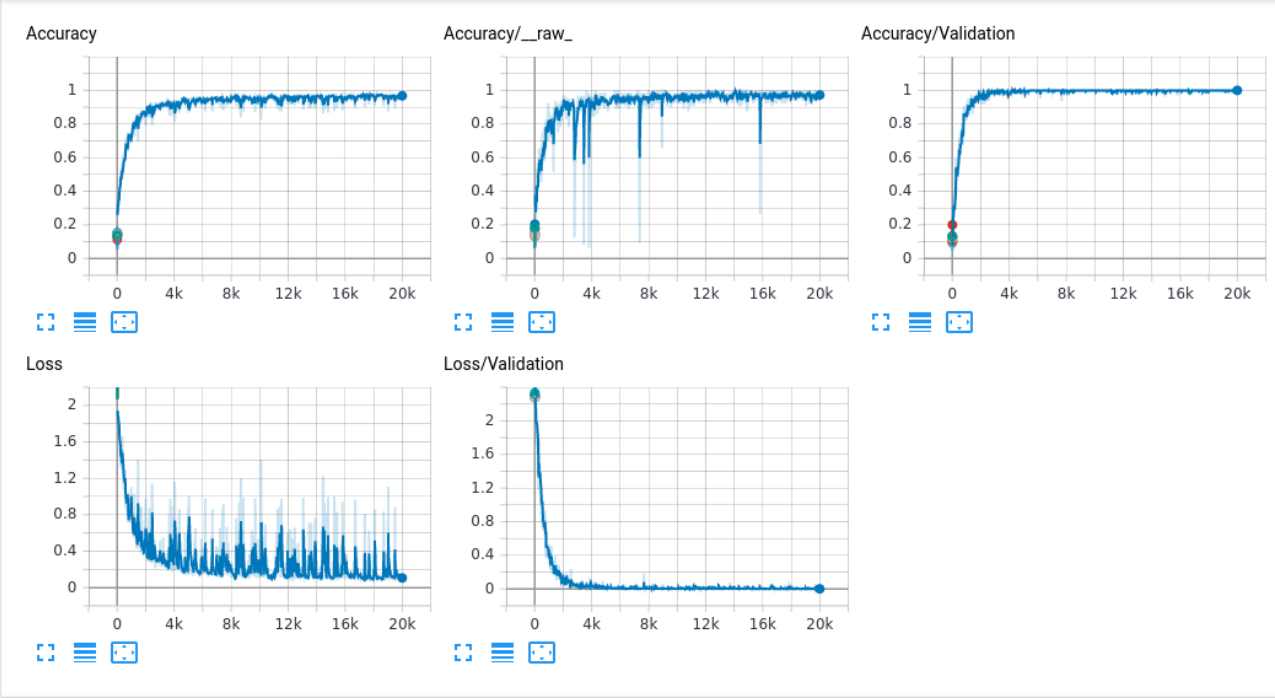


FIGURE 3. Model's accuracy and loss for 10^{-3} learning rate @2000 training iterations.

Algorithm 1 Proposed Speech Recognition Model for the LSTM-RNN Network

procedure ENHANCED LSTM RNN PROCEDURE(X, Y)
    Input Speech (Speech X)
    Extract Feature Map;
    LSTM processing;
    RNN processing - Cell's states memory resetting;
    LSTM processing;
    Model training
    Generate Prediction (Map$Y$);
    Perform Optimal Estimation using Adam Optimization;
    Output Recognized Speech
end procedure

The dataset used in this study consists of isolated spoken digits. It is a tar file consisting of 15 speakers (male and female). Each speaker utters a digit 16 times, leading to 15*16 = 240 instances for each digit. The phrases were English numbers: 0-9. This gives us a total of, 2400 different audio files with wav format for training the proposed system.

The dataset was split into training and validation datasets. Ten percent (10%) of the dataset was used for validation, and the remaining ninety percent (90%) was used for training. The training step output contained validation accuracy and loss as shown in Table 1 because the validation set was introduced as a part of the model fit function during training.

The proposed LSTM RNN network structure comprises four network layers: an input layer, LSTM (dropout) layer, fully connected layer and regression layer. The model was trained using a deep-learning library known as TFLearn.

B. PROPOSED MODEL TRAINING

The learning rate and number of training iterations can affect the accuracy and training time of the proposed model. Therefore, both parameters were adjusted to different values for optimal performance. Given that the learning rate should be considered the most crucial hyperparameter, it might be necessary to understand how to adjust it properly to achieve a positive outcome [50]. The learning rate regulates the speed of the network weight updates. The initial learning rate of the model was set at 10^{-3}.

Next is the training iteration, which was adjusted to the initial value of = 1000 iters. Training iterations were used to multiply the epoch size to obtain the training steps. The training steps, with 10 epochs of batch size 64/64, ranged from 10000 to 20000 training steps. A high accuracy was achieved when the number of training steps was increased.

To reduce LSTM total loss on a set of training sequences, Adam's optimization algorithm was used to improve the parameter of each network weight to the weight parameter using the BPTT method [17], [51], [52]. The BPTT method, used for learning the weight matrices of an RNN unravels the network on time and disseminates error signals backward through time. The major challenge with the BPTT method is the vanishing gradient problem. However, this difficulty is being overcome to a great extent by using LSTM cells [53].

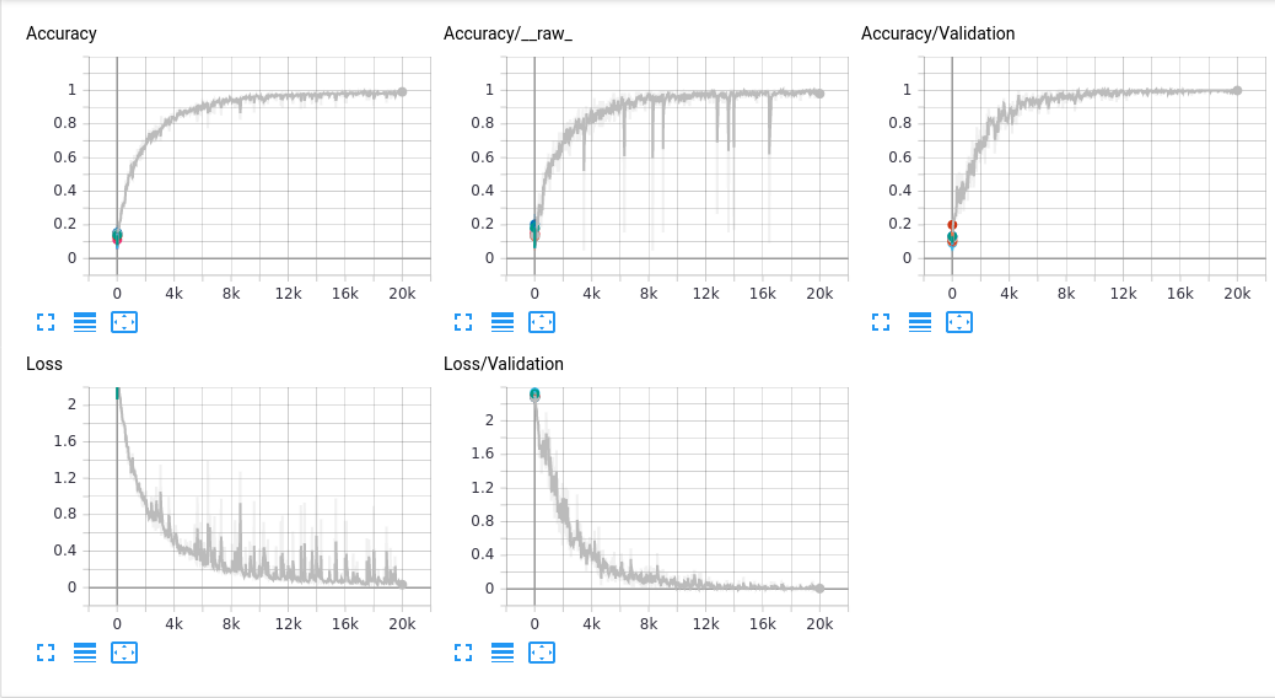


FIGURE 4. Model's accuracy and loss for 10^{-4} learning rate @2000 training iterations.

The cross-entropy loss that used the softmax activation function was used to train the networks. With an initial learning rate of 10^{-3}, the model trained quickly, but started to overfit at some point. It was observed that the accuracy dropped when the model overfitted. By adjusting the learning rate to 10^{-4}, the model was trained slowly, with an increase in network accuracy.

The proposed model was implemented on a multi-core central processing unit (CPU) on a single machine instead of a graphics processing unit (GPU). The choice of using a CPU is made because CPUs are relatively simple to implement and easy to debug. It also allows for easy distributed implementation on a large cluster of machines [54].

The computational graphs of the model's output were visualized using a TensorBoard. It is a visualization extension created by the TensorFlow team to decrease the complexity of neural networks. Time-dependent scalar statistics that vary over time and variations in accuracy and loss performance are visualized in Figures 3 and 4 for 2000 iterations at learning rates of 10^{-3} and 10^{-4}, respectively.

Other deep learning models such as ResNet-18, ResNet-34, DenseNet-121, DenseNet-169, and VGG-16 were used to train the model. The output of the training showing loss and accuracy curves and the bar chart comparing the performances of the deep learning models with the proposed model are shown in Figures 5, 6, respectively.

C. RESULTS AND DISCUSSIONS

The result of the model's training has shown that good hyperparameters such as the learning rate, help to manage a large set of experiments for hyperparameter tuning. This shows that increasing the learning rate leads to fast network training, whereas reducing the learning rate leads to an accurate prediction of the network. Hence, it represents the trade-offs between time and accuracy. Optimum accuracy is possible when the learning rate is reduced and the number of training steps increases.

From the performance results of network training in the proposed model, it is necessary to state that RNNs are at the centre of recent ASR systems. Specifically, LSTM RNN have shown exciting results in numerous speech recognition

TABLE 1. The result for learning rates tuning and its corresponding accuracy.

Training iters	Learning rate	Loss	Accuracy	Val loss	Val acc
1000	0.001	0.49798	0.9197	0.02684	0.9844
1000	0.0001	0.31974	0.9333	0.01019	1.0000
2000	0.001	0.10913	0.9665	0.00189	1.0000
2000	0.0001	0.02656	0.9936	0.00130	1.0000

TABLE 2. Comparing the proposed model accuracy with other deep learning models on the same dataset.

Models	Loss	Val loss	Accuracy
ResNet-18	1.0733	0.9134	0.7283
ResNet-34	0.8990	0.8125	0.7417
DenseNet-121	0.3723	0.3305	0.8967
DenseNet-169	0.5774	0.3744	0.8717
VGG-16	0.7045	0.6663	0.7717
Proposed Model	0.0266	0.0013	**0.9936**

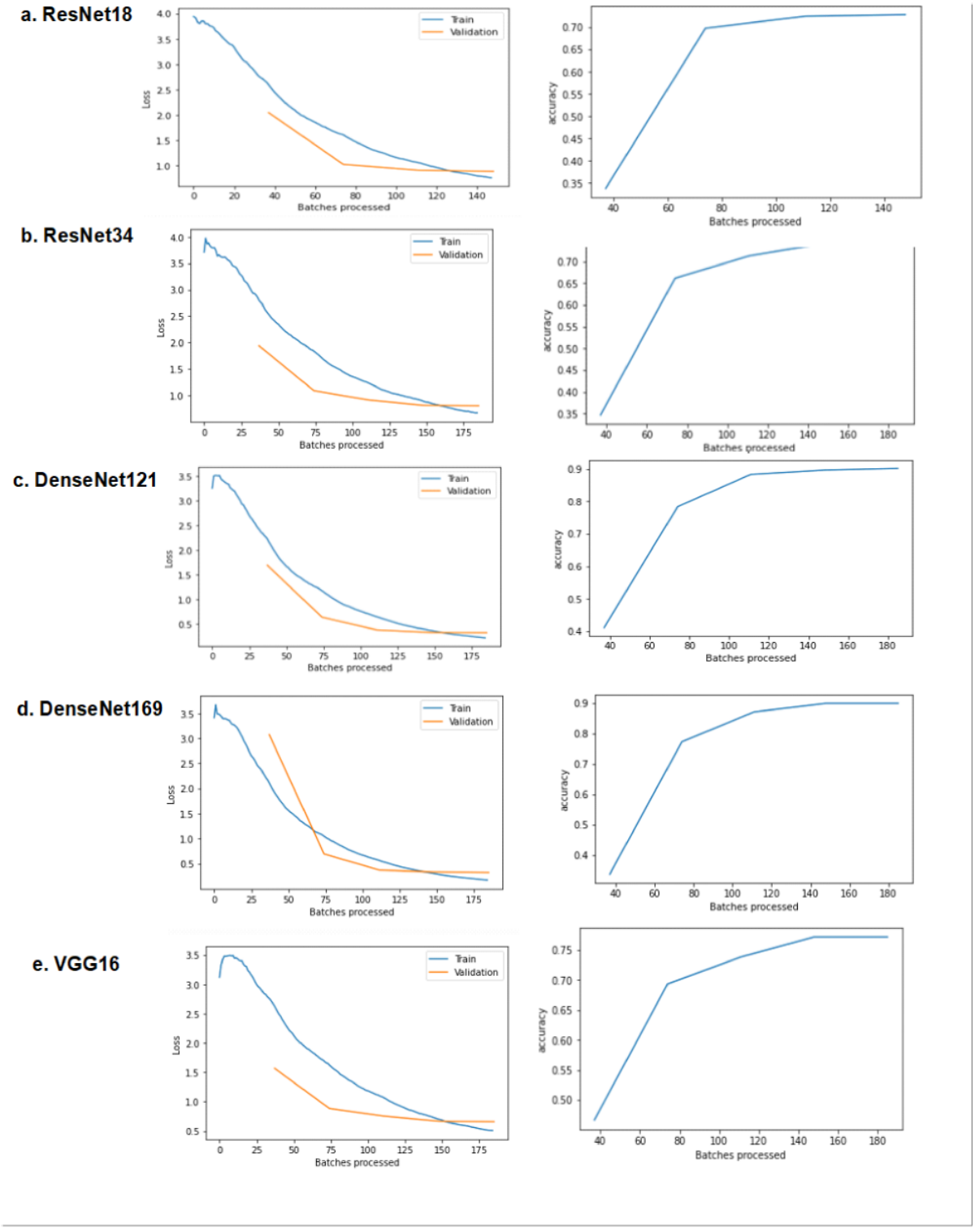


FIGURE 5. Loss and accuracy curves for ResNet-18, ResNet-34, DenseNet-121, DenseNet-169, and VGG-16.

jobs, owing to their capability to represent long-term and short-term dependencies in sequences [55].

The model showed 99.36% accuracy and 100.00% validation accuracy with the least minimal loss of 0.02656 for 2000 training iterations at the learning rate of 10^{-4}, as represented in Table 1. This is to prove that a low learning rate leads to a higher accuracy. Table 1 presents the results of the learning rate tuning of the model.

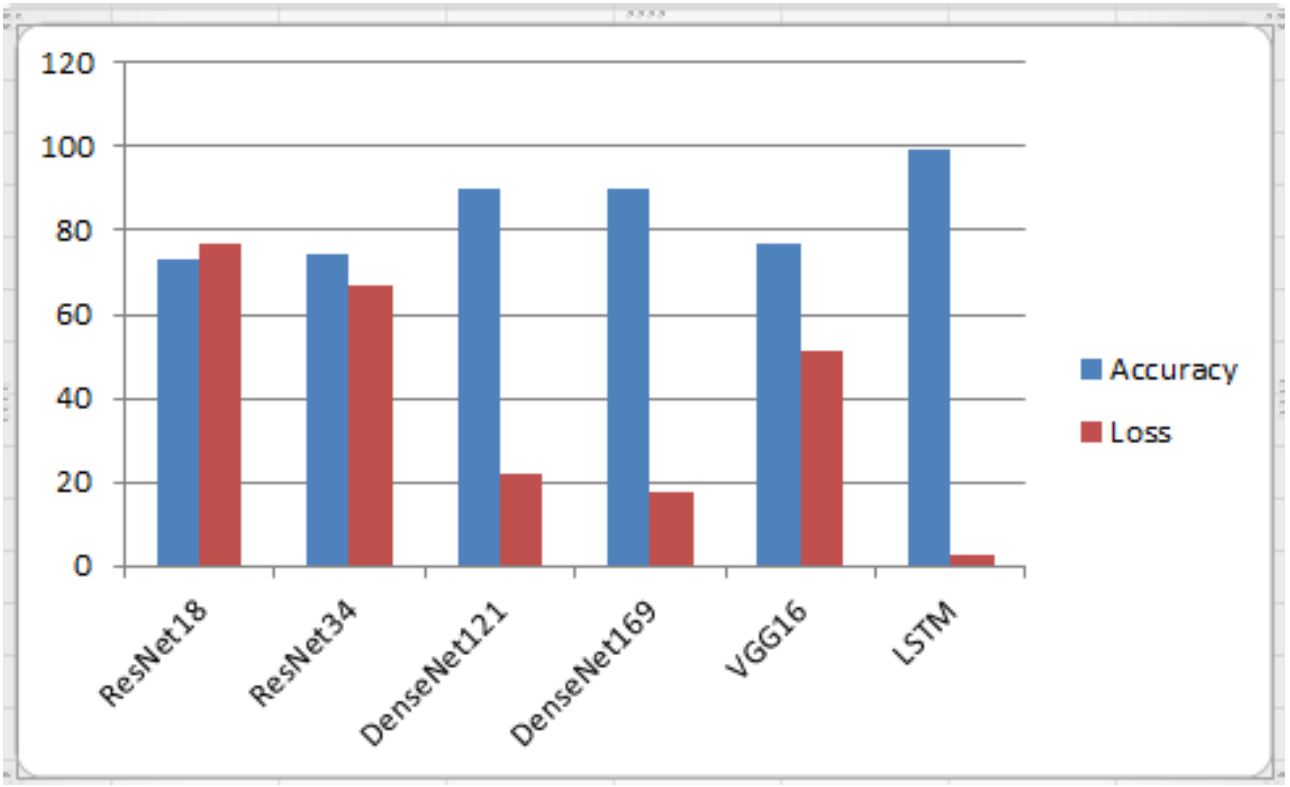


FIGURE 6. Bar chart for comparing the performances of the deep learning models with the proposed model.

TABLE 3. Comparing the proposed model accuracy with some sequential models on the same dataset.

Models	Loss	Val loss	Accuracy
Simple LSTM	0.02420	0.59769	0.8711
Bidirectional LSTM	0.02445	0.63545	0.8086
Simple RNN	0.53050	0.39571	0.8359
GRU	0.03808	0.52378	0.9023
Proposed Model	0.0266	0.0013	**0.9936**

TABLE 4. Summary of the model's classification report using a confusion matrix.

Average Score	Precision	Recall	F1-Score
Macro Average	0.67	0.67	0.61
Weighted Average	**0.70**	**0.60**	**0.60**

The ResNet-18, ResNet-34, DenseNet-121, DenseNet-169, and VGG-16 deep learning models were run on the same dataset and the results were compared with the performance of the proposed model as presented in Table 2. From Table 2, it can be seen that DenseNet-121 and DenseNet-169 showed high accuracies of 89.67% and 87.17% respectively, but LSTM-RNN showed the highest accuracy of 99.36% and outperformed the other deep learning models on the same dataset. A summary of the performance of the deep learning models in comparison with the proposed model is represented as a bar chart in Figure 6. LSTM exhibited the best performance in terms of both accuracy and loss. Sequential models such as GRU, bidirectional LSTM, simple LSTM and RNN were also tested on the same dataset and the performance is compared in Table 3.

To further investigate the model results, a confusion matrix was used to evaluate classification performance. From the model's classification report, the average score for precision, recall, and f1-score was derived from the classification report and served as the performance metric for the evaluation of the proposed model as shown in Table 4.

V. CONCLUSION

In this study, an LSTM-RNN model has been proposed that incorporates an RNN into the LSTM network to overcome the challenges of the traditional LSTM in processing a continuous input stream. The proposed system utilizes an RNN as a forget gate in the network, which allows the resetting of the cell states at the beginning of sub-sequences and consequently improves the performance of the model to make effective use of network parameters. This addresses the computational efficiency problems of large networks for large-vocabulary speech recognition. The proposed model is evaluated using a well-established dataset. Some CNN-based and sequential models were also used on the same dataset, and the performances of the models were compared with the performance of the proposed model. The proposed LSTM-RNN outperformed other deep learning models with an accuracy of 99.36% on the well-established public benchmark spoken English digit dataset.

REFERENCES

[1] P. N. Nasreen, A. C. Kumar, and P. A. Nabeel, "Speech analysis for automatic speech recognition," in *Proc. Int. Conf. Comput., Commun. Sci.*, 2016.

[2] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep Learning*, vol. 1, no. 2. Cambridge, MA, USA: MIT Press, 2016.

[3] D. Yu and L. Deng, *Automatic Speech Recognition*, vol. 1. Berlin, Germany: Springer, 2016.

[4] S. Lin, N. Liu, M. Nazemi, H. Li, C. Ding, Y. Wang, and M. Pedram, "FFT-based deep learning deployment in embedded systems," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2018, pp. 1045–1050.

[5] G. Cheng, P. Zhang, and J. Xu, "Automatic speech recognition system with output-gate projected gated recurrent unit," *IEICE Trans. Inf. Syst.*, vol. 102, no. 2, pp. 355–363, 2019.

[6] J. Oruh and S. Viriri, "Deep learning with optimization techniques for the classification of spoken English digit," in *Proc. Int. Conf. Comput. Collective Intell.* Cham, Switzerland: Springer, Sep. 2021, pp. 494–507.

[7] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 6645–6649.

[8] Y. Zhang, G. Chen, D. Yu, K. Yao, S. Khudanpur, and J. Glass, "Highway long short-term memory RNNS for distant speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2016, pp. 5755–5759.

[9] W. Han, C. Wu, X. Zhang, M. Sun, and G. Min, "Speech enhancement based on improved deep neural networks with MMSE pretreatment features," in *Proc. IEEE 13th Int. Conf. Signal Process. (ICSP)*, Nov. 2016, pp. 1140–1145.

[10] R. Ahmad, S. Zubair, and H. Alquhayz, "Speech enhancement for multimodal speaker diarization system," *IEEE Access*, vol. 8, pp. 126671–126680, 2020.

[11] N. Saleem, M. I. Khattak, M. Al-Hasan, and A. B. Qazi, "On learning spectral masking for single channel speech enhancement using feedforward and recurrent neural networks," *IEEE Access*, vol. 8, pp. 160581–160595, 2020.

[12] K. V. Ramesh and S. Gahankari, "Hybrid artificial neural network and hidden Markov model (ANN/HMM) for speech and speaker recognition," *Int. J. Comput. Appl.*, vol. 975, p. 8887, Mar. 2013.

[13] P. J. Werbos, "Backpropagation through time: What it does and how to do it," *Proc. IEEE*, vol. 78, no. 10, pp. 1550–1560, Oct. 1990.

[14] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural Comput.*, vol. 1, pp. 270–280, Jun. 1989.

[15] M. C. Mozer, "Induction of multiscale temporal structure," in *Proc. Adv. Neural Inf. Process. Syst.*, 1992, pp. 275–282.

[16] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition," 2014, *arXiv:1402.1128*.

[17] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, Oct. 1986.

[18] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.

[19] J. Martens and I. Sutskever, "Learning recurrent neural networks with hessian-free optimization," in *Proc. ICML*, 2011.

[20] M. Sundermeyer, R. Schlüter, and H. Ney, "LSTM neural networks for language modeling," in *Proc. 13th Annu. Conf. Int. Speech Commun. Assoc.*, 2012.

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

[22] P. Srivastava, "Essentials of deep learning: Introduction to long short term memory," Tech. Rep., Dec. 2017.

[23] J. T. Geiger, Z. Zhang, F. Weninger, B. Schuller, and G. Rigoll, "Robust speech recognition using long short-term memory recurrent neural networks for hybrid acoustic modelling," in *Proc. 15th Annu. Conf. Int. Speech Commun. Assoc.*, 2014, pp. 1–5.

[24] H. Sak, A. Senior, K. Rao, and F. Beaufays, "Fast and accurate recurrent neural network acoustic models for speech recognition," 2015, *arXiv:1507.06947*.

[25] M. Ravanelli, P. Brakel, M. Omologo, and Y. Bengio, "Light gated recurrent units for speech recognition," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 2, no. 2, pp. 92–102, Apr. 2018.

[26] R. Feng, W. Jiang, N. Yu, Y. Wu, and J. Yan, "Projected minimal gated recurrent unit for speech recognition," *IEEE Access*, vol. 8, pp. 215192–215201, 2020.

[27] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, and A. Y. Ng, "Deep speech: Scaling up end-to-end speech recognition," 2014, *arXiv:1412.5567*.

[28] J. Li, "Recent advances in end-to-end automatic speech recognition," 2021, *arXiv:2111.01690*.

[29] A.-R. Mohamed, G. Dahl, and G. Hinton, "Deep belief networks for phone recognition," in *Proc. Workshop Deep Learn. Speech Recognit. Rel. Appl. (NIPS)*, vol. 1, no. 9. Vancouver, BC, Canada, 2009, p. 39.

[30] A.-R. Mohamed and G. Hinton, "Phone recognition using restricted Boltzmann machines," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Mar. 2010, pp. 4354–4357.

[31] A.-R. Mohamed, T. N. Sainath, G. Dahl, B. Ramabhadran, G. E. Hinton, and M. A. Picheny, "Deep belief networks using discriminative features for phone recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2011, pp. 5060–5063.

[32] A.-R. Mohamed, G. E. Dahl, and G. Hinton, "Acoustic modeling using deep belief networks," *IEEE Trans. Audio, Speech, Language Process.*, vol. 20, no. 1, pp. 14–22, Jan. 2012.

[33] A. R. Mohamed, G. Hinton, and G. Penn, "Understanding how deep belief networks perform acoustic modelling," *Neural Netw.*, to be published.

[34] O. Abdel-Hamid, A.-R. Mohamed, H. Jiang, and G. Penn, "Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2012, pp. 4277–4280.

[35] O. Abdel-Hamid, L. Deng, and D. Yu, "Exploring convolutional neural network structures and optimization techniques for speech recognition," in *Proc. Interspeech*, vol. 11, 2013, pp. 5–73.

[36] T. N. Sainath, A.-R. Mohamed, B. Kingsbury, and B. Ramabhadran, "Deep convolutional neural networks for LVCSR," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 8614–8618.

[37] W. Chan and I. Lane, "Deep convolutional neural networks for acoustic modeling in low resource languages," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 2056–2060.

[38] Y. Yu, X. Si, C. Hu, and Z. Jianxun, "A review of recurrent neural networks: LSTM cells and network architectures," *Neural Comput.*, vol. 31, no. 7, pp. 1235–1270, Jul. 2019.

[39] Y. Tachioka and J. Ishii, "Long short-term memory recurrent-neural-network-based bandwidth extension for automatic speech recognition," *Acoust. Sci. Technol.*, vol. 37, no. 6, pp. 319–321, 2016.

[40] T. He and J. Droppo, "Exploiting LSTM structure in deep neural networks for speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2016, pp. 5445–5449.

[41] A. S. Mahfoudh, B. A. Wazir, and J. H. Chuah, "Spoken Arabic digits recognition using deep learning," in *Proc. IEEE Int. Conf. Autom. Control Intell. Syst. (I2CACIS)*, Jun. 2019, pp. 339–344.

[42] H. Palangi, L. Deng, Y. Shen, J. Gao, X. He, J. Chen, X. Song, and R. Ward, "Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 24, no. 4, pp. 694–707, Apr. 2016.

[43] F. A. Gers, J. Schmidhuber, F. Cummins, F. A. Gers, and F. Cummins, "Learning to forget: Continual prediction with LSTM," in *Proc. Int. Conf. Artif. Neural Netw. (ICANN)*, vol. 2, 1999, pp. 850–855.

[44] Q. Lyu and J. Zhu, "Revisit long short-term memory: An optimization perspective," in *Proc. Adv. Neural Inf. Process. Syst. Workshop Deep Learn. Represent. Learn.*, 2014, pp. 1–9.

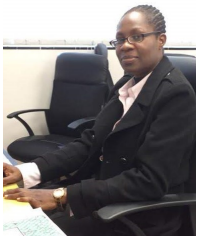
[45] F. A. Gers, N. N. Schraudolph, and J. Schmidhuber, "Learning precise timing with LSTM recurrent networks," *J. Mach. Learn. Res.*, vol. 3, no. 1, pp. 115–143, 2003.

[46] P. Protopapas and M. Glickman, "CS109B data science 2 lecture 10: Recurrent neural networks," Inst. Appl. Comput. Sci., Havard Univ., Cambridge, MA, USA, Tech. Rep., 2019.

[47] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, M. Hasan, B. C. Van Essen, A. A. S. Awwal, and V. K. Asari, "A state-of-the-art survey on deep learning theory and architectures," *Electronics*, vol. 8, no. 3, p. 292, Mar. 2019.

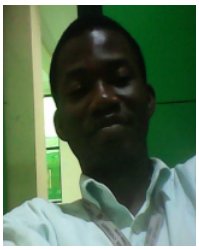
[48] Pannous.Github. (Dec. 2016). *Pannous/Tensorflow-Speech-Recognition*. Accessed: May 3, 2020. [Online]. Available: http://github.com/pannous/tensorflow-speech-recognition

[49] B. McFee, M. McVicar, C. Raffel, D. L. O. Nieto, J. Moore, and D. Ellis, "Librosa: V0.4.0.Zenodo, 2015," in *Proc. 14th Python Sci. Conf. (SCIPY)*, 2015.
[50] K. Greff, R. K. Srivastava, J. Koutnìk, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space Odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017.
[51] M. Boden, "A guide to recurrent neural networks and backpropagation," The Dallas Project, Tech. Rep., 2002.
[52] H. Jaeger, "Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the 'echo state network' approach," GMD-Forschungszentrum Informationstechnik Bonn, Tech. Rep., 2002, vol. 5, no. 1.
[53] D. Yu and L. Deng, "Recurrent neural networks and related models," in *Automatic Speech Recognition*. Springer, 2015, pp. 237–266.
[54] J. Dean, G. Corrado, R. Monga, and K. Chen, "Large scale distributed deep networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1223–1231.
[55] T. Parcollet, M. Morchid, G. Linares, and R. D. Mori, "Bidirectional quaternion long short-term memory recurrent neural networks for speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 8519–8523.



JANE ORUH received the B.Sc. degree in computer science from the Michael Okpara University of Agriculture, Nigeria, in 2005, and the M.Sc. degree in computer science from Ebonyi State University, Abakaliki, Nigeria, in 2013. She is currently pursuing the Ph.D. degree in computer science with the University of KwaZulu-Natal (UKZN), South Africa. Her research interests include biometric authentication systems, artificial intelligence, automatic speech recognition, speech signal processing, deep learning, and machine learning.

SERESTINA VIRIRI (Senior Member, IEEE) received the B.Sc. degree in mathematics and computer science, and the M.Sc. and Ph.D. degrees in computer science. He has been in academia, since 1998. He is currently a Full Professor of computer science with the School of Mathematics, Statistics and Computer Science, University of KwaZulu-Natal, South Africa. He is a Rated Researcher by the National Research Foundation (NRF), South Africa. He has published extensively in several artificial intelligence and computer vision-related accredited journals and international and national conference proceedings. His main research interests include artificial intelligence, computer vision, image processing, machine learning, medical image analysis, pattern recognition, and other image processing related fields, such as biometrics, medical imaging, and nuclear medicine. He serves as a reviewer for several machine learning and computer vision-related journals. He has also served on program committees for numerous international and national conferences.



ADEKANMI ADEGUN received the B.Tech., M.Sc., and Ph.D. degrees in computer science. He has close to ten years lecturing experience in Universities. He has also co-supervised M.Sc. and Ph.D. candidates in machine learning fields. He has published extensively in several artificial intelligence and computer vision-related accredited journals and international and national conference proceedings. His main research interests include artificial intelligence, computer vision, image processing, machine learning, medical image analysis, pattern recognition, and natural language processing. He currently serves as a reviewer for some machine learning and computer vision-related journals.