Computers, Materials & Continua
DOI: 10.32604/cmc.2022.024590
Article

Tech Science Press

An Innovative Approach Utilizing Binary-View Transformer for Speech Recognition Task

Muhammad Babar Kamal[1], Arfat Ahmad Khan[2], Faizan Ahmed Khan[3], Malik Muhammad Ali Shahid[4], Chitapong Wechtaisong[2,*], Muhammad Daud Kamal[5], Muhammad Junaid Ali[6] and Peerapong Uthansakul[2]

[1]COMSATS University Islamabad, Islamabad Campus, 45550, Pakistan
[2]Suranaree University of Technology, Nakhon Ratchasima, 30000, Thailand
[3]COMSATS University Islamabad, Lahore Campus, 54000, Pakistan
[4]COMSATS University Islamabad, Vehari Campus, 61100, Pakistan
[5]National University of Sciences & Technology, Islamabad, 45550, Pakistan
[6]Virtual University of Pakistan, Islamabad Campus, 45550, Pakistan
*Corresponding Author: Chitapong Wechtaisong. Email: chitapong@g.sut.ac.th
Received: 23 October 2021; Accepted: 26 November 2021

Abstract: The deep learning advancements have greatly improved the performance of speech recognition systems, and most recent systems are based on the Recurrent Neural Network (RNN). Overall, the RNN works fine with the small sequence data, but suffers from the gradient vanishing problem in case of large sequence. The transformer networks have neutralized this issue and have shown state-of-the-art results on sequential or speech-related data. Generally, in speech recognition, the input audio is converted into an image using Mel-spectrogram to illustrate frequencies and intensities. The image is classified by the machine learning mechanism to generate a classification transcript. However, the audio frequency in the image has low resolution and causing inaccurate predictions. This paper presents a novel end-to-end binary view transformer-based architecture for speech recognition to cope with the frequency resolution problem. Firstly, the input audio signal is transformed into a 2D image using Mel-spectrogram. Secondly, the modified universal transformers utilize the multi-head attention to derive contextual information and derive different speech-related features. Moreover, a feed-forward neural network is also deployed for classification. The proposed system has generated robust results on Google's speech command dataset with an accuracy of 95.16% and with minimal loss. The binary-view transformer eradicates the eventuality of the over-fitting problem by deploying a multi-view mechanism to diversify the input data, and multi-head attention captures multiple contexts from the data's feature map.

Keywords: Convolution neural network; multi-head attention; multi-view; RNN; self-attention; speech recognition; transformer

1 Introduction

The recent surge of Artificial Intelligence (AI) in modern technology has resulted in the widespread adoption of Human-Computer-Interaction (HCI) applications. Big corporations in information technology like Google, Apple, Microsoft, and Amazon are relentlessly working to improve the applicability and dynamics of HCI applications using speech recognition algorithms. The importance of recognition systems underscores vast fields, including stakeholders from the domain related to entertainment applications, utility applications and critical lifesaving appliances. e.g., YouTube [1] and Facebook [2] use speech recognition systems for captioning. Various robust voice commands applications are proposed for devices that work in areas without internet services and critical mission's robots [3,4]. Moreover, robust design of micro-controller-based devices which works based on speech commands are also proposed in literature [5]. Apple Siri, Amazon Alexa, Microsoft Cortana, YouTube captions, and Google Assistant [6] deploy speech recognition systems which works based on these designs. Google and Microsoft [7], use deep neural networks-based algorithms that convert sound to text through speech recognition, process the text, and respond accordingly. Typically, deep learning algorithms processes the 1D data as audio is recorded and represented s a 1D waveform [8]. The waveform of the 1D signal is represented in the sinusoidal time domain. In [9], the authors studied the 2D representation of an audio signal called the spectrogram where the frequencies spectrum is derived from the time-frequency domain through Fourier transform.

Speech signals contains rich prominent features such as emotions and dialect. Studies have been conducted to compare the 1D audio waveform and 2D spectrogram, the spectrogram concluded that the 1D signal does not contain frequency information vital for better speech recognition [10]. Studies shows that 2D spectrogram performs better to extract features for speech recognition. Since a spectrogram focuses on all the frequencies, the recognition system cannot properly differentiate [11] between relevant frequencies and noise. Fusion of mel-scale with spectrogram reduces noise which shows performance improvement in speech recognition. The mel-scale discards noise and amplifies the desired spectrum of frequencies in the 2D spectrogram. The 2D transformation (mel-spectrogram) of audio signal deploy state-of-the-art image recognition algorithms in Neural Networks (N.N) for speech recognition to improve the precision of the system by imitating the human speech perception [12]. The N.N algorithms [13] process raw input data by correlating hidden patterns to recognize similar clusters in data and classify it by continuously learning and enhancing the recognition system. Recurrent NNs (RNNs) [14–16], Convolutional NNs (CNNs) [17] and Attention are commonly used to develop speech recognition systems. RNN captures sequential prediction of data using recurrence units to predict pattern for next likely scenario. RNN algorithms and their variants, i.e., Long-Short-Term-Memory (LSTM), and Gated-Recurrent-Unit (GRU) allow the machine to process sequential data models, such as speech recognition. LSTM has become popular in recurrent networks due to its success in solving the vanishing gradient problem by retaining the long-term dependencies of data.

However, the LSTM [18] fails to solve the vanishing gradient problem completely due to the complexity of the additional evaluation of memory cells. The RNN models are prone to over-fitting due to the difficulty of applying dropout algorithms with LSTM probabilistic units. The sequential nature of models is inconsistent with the parallelization of processing [19]. RNN models require more resources and time to train due to the linearized natures of layers and random weights initialization. Many researchers have used CNN for audio classification to analyze visual imagery of audio by convolving multiple filters on data to extract features for the neural network. Deep CNN convolves multiple layers of filters on image to extract distinct features having depth depending on the number of layers. Deep networks improve algorithm's ability by capturing unique properties using multiple

convolution layers to retrieve a higher level of features. The feature-map produced from this process enhances the recognition system accuracy.

However, these studies observe that the deeper layers of convolution tends to assimilate general/abstract level information from the input data [20]. The deep CNN model tends to over-fit when the labeled data for training is less. The deep networks of the convolution model are prone to gradient vanishing/exploding problems as the network deepens, causing less precision of the recognition model. Therefore, the researchers deploy attention mechanisms with an RNN model to obtain long-term dependencies by contextualizing the feature-map. The attention model uses probabilistic learning by giving weight to the important feature using the soft-max probabilistic function. Moreover, the attention-based models reduce the vanishing gradient problem by decreasing the number of features to process important and unique features for the recognition system [21]. In [22], the authors introduce one of the attention mechanism variations, self-attention, to compute the representation of the same sequence relating to different positioning. Self-attention allows input sequences to interact with all neighboring values and find contextual and positional attention within the same sequence. In [23], the authors observe the multi-view approach with a neural network algorithm to increase the efficiency of the architecture. The main objective of the paper is to improve existing speech recognition systems by building a precise method that can be implemented in any speech recognition application with a lightweight footprint.

In [24], the authors use Fourier transform to convert the waveform signal to alternative representations characterized by a sinusoidal function. The paper uses Infrared spectroscopy through Fourier transform for analysis of biological material. In [25], the Short-Time Fourier-Transform (STFT) is used to extract features from the signal of audio by slicing the signal into windows and performing Fourier transform on each window to obtain meaningful information. Actually, Deep Learning (DL) [26] models extract intricate structures in data, and back-propagation algorithms show which parameters are used for calculating each layer representation. In fact, DL allows the computation of multiple processing for the learning of data representation having many levels of abstractions. In [27], authors elaborate the feature extraction in speech categorizing speech recognition to three stages. At first, the audio signal is divided into small chunks; secondly, the phoneme is extracted and processed, and lastly, the speech is categorized on word level. Music detection is discussed in [28], where the authors use CNN with mel kernel to separate music content from speech and noise. The mel-scale is useful for focusing on a specific type of frequency and minimizing the effect of noisy and unrelated parts.

In [29], an attention model is used for audio tagging of Google Audio Set [30]. The authors investigate Multi Instance-Learning (MIL) problem for weakly labeled audio set classification by introducing the attention model for probabilistic learning, where attention is used with a fully connected neural network for multi-label classification on audio. Multi-head attention is used in [31], where authors elaborate the implication to extract information from multiple representation subspaces at various positions by the ability of multi-head to attend to different interpretations within the data jointly. The multi-head attention is useful for obtaining different contexts within the information which improve the efficiency of the model.

In this paper, we present a novel end-to-end binary view transformer-based architecture for speech recognition to cope with the frequency resolution problem. Firstly, the input audio signal is transformed into a 2D image using Mel-spectrogram. Secondly, the multi-view mechanism is used to enhance the frequency resolution in the image. In addition, the Modified universal transformers utilized the multi-head attention to derive contextual information and derive different speech-related

features. A feed-forward neural network is also deployed for classification. The proposed system is discussed in details in the Section 5. Moreover, the proposed system has generated robust results on Google's speech command dataset with an accuracy of 95.16% and with minimal loss. The binary-view transformer eradicates the eventuality of the over-fitting problem by deploying a multi-view mechanism to diversify the input data, and multi-head attention captures multiple contexts from the data's feature map.

The rest of the paper is organized as follows: The Section 2 contains the speech perception and recognition by using AI, and the proposed system is discussed in the Section 3. The Section 4 includes the experiment steps and testing. Furthermore, the Section 5 includes the experiment results and discussions. Finally, the Section 6 concludes the research work.

2 Speech Perception and Recognition Using AI

Perception is the ability to systematically receive information, identify essential data features and then interpret that information, while recognition is the system's ability to identify the classification of data. To build a system using AI for the speech recognition, we need to have input data that is in the form of an audio signal. After pre-processing, the audio signal progresses to the speech recognition system, and the systems output will be a classification transcript of the audio. A microphone records the audio signal with a bit depth of 16 (recorded signal in time domain having values of $2 * 16$). Audio is recorded at 16 kilohertz having a nitrous frequency of 8 kilohertz; the nitrous is a range of distinguished lower frequency, which is interpretable and differentiable by the brain for speech because most frequency changes happen at lower frequencies. The signal in the time domain is complicated to interpret, as the human ear can sense the intensity of frequency. Moreover, we use a pre-processing step to convert the signal into the frequency domain using Fourier transform, where the time-domain representation of the signal is transformed into a time-frequency domain.

The power spectral density of the audio signal for different bands of frequencies are shown in the Fig. 1, where the nitrous frequency range has most frequencies changes. We create a spectrogram by stacking periodogram adjacent to one another over time. The spectrogram is a colored 2D image representation of the audio.

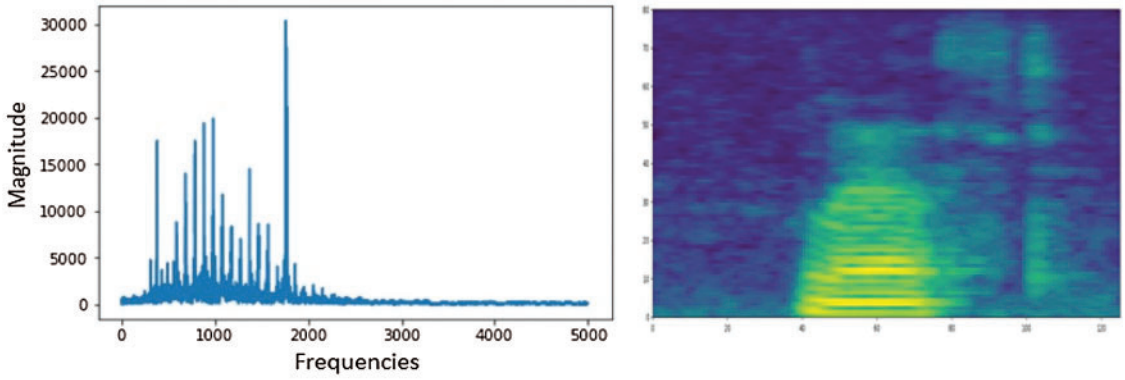

Figure 1: Periodogram of audio frequencies and the 2D representation of audio signal using spectrogram

For speech recognition, the human brain amplifies some frequencies, while nullifying or reducing the background noise by giving more importance to the lower band of frequencies. For example,

humans can tell the difference between 40 and 100 hertz, but are unable to differentiate between 10,000 and 12,000 hertz. This objective in computing is achieved through mel-scale; by applying mel-filter-bank on the frequencies, we can retrieve the lower frequencies efficiently, as shown in the Fig. 2.

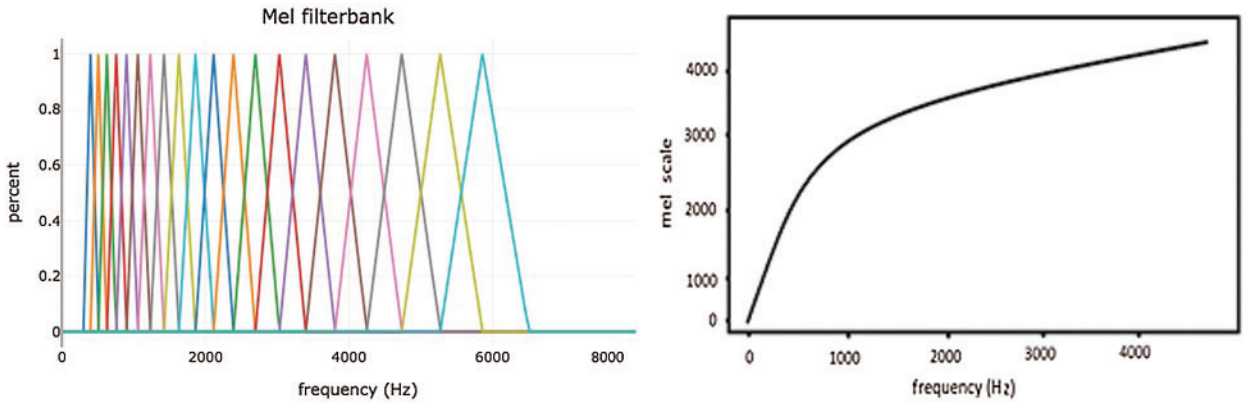

Figure 2: Mel filter-banks and the frequencies from linear to logarithmic

2.1 Convolutional Neural Network

In the field of machine learning, CNN is one of the most efficient algorithms for image recognition. Since the inception of CNN, the field of machine learning is revolutionized, and state-of-the-art results are produced. In CNN, different filters are convolved over the image to compute essential features by using the Eqs. (1) and (2), where filter B convolves over image A having k number of rows and columns. Convolution gives us a large pool of features in data that is passed to a N.N., which helps to classify them into different classes. Many variants of CNN are produced over the years that improve the performance of these models, e.g., Inception net [32], Resnet [33], and mobile net [34].

$$Conv_{(A,B)} = A * B \tag{1}$$

$$Conv|x, y| = \sum_{a=-k}^{K} \sum_{b=-k}^{K} A|a, b|B|x - a, y - b| \tag{2}$$

2.2 Recurrent Neural Network

RNN algorithms allow the machine to process temporal sequences of variable lengths [14]. This type of algorithm is useful in processing sequential data through sequential modelling, e.g., signal processing, NLP, and speech recognition. The RNN models produce hidden-vector as a function of the former states and the next input as shown in Eq. (3), where input vectors A are sequentially processed by recurrence *Function* having w parameters on each time-stamp to produce a new state for the model.

$$State_{(new)} = Function_{(w)}[State_{(old)}, A] \tag{3}$$

Recurrence models generate a sequential pattern of data that prevents parallelization for training data. The sequential nature increases the computation time of the model and limits longer sequences from processing, which causes the gradient vanishing/exploding problem.

2.3 Attention Mechanism

Attention is a deep learning mechanism that is mainly inspired by the natural perception ability of humans as humans receive information in raw form from the senses and transmit it to the brain [29]. The brain opts for the relevant and useful information by ignoring background noises; this process polishes the data, making it easier to perceive. Moreover, the attention is a weighted probabilistic vector with the soft-max function used in a neural network, which was introduced to improve the sequential models (LSTMs, RNNs) to capture essential features in context vectors as shown in Eq. (4), and *Attention_weight*is elaborated in the Eq. (5).

$$Context\_vector = \sum_{y}^{n} Attention\_weight_y * Hidden\_state_y \tag{4}$$

$$Attention\_weight_y = \frac{\exp(Hidden_{state\,y})}{\sum_{a=0}^{k} Hidden\_state_a} \tag{5}$$

The attention mechanism extracts model dependencies while the effect of distance between input and output sequences is negated, which improves the model performance. Self-attention [35] is a variation of the attention mechanism that allows the vectors to interact with each other to discover the important features, so that more attention can be given. Applying attention to sequential models improves the accuracy, and state of the art results are achieved.

2.4 Transformer

In transformer architecture, instead of sequential representation, the positional information (input data) is embedded in positional vector with input vectors that permit parallelization. Transformer architecture consists of two parts, i.e., encoder layers and decoder layers. In a transformer, attention mechanism is used for content-based memory retrieval, where decoder attends to content that is encoded and decides which information needs to be extracted based on affinity or its position. The input data is processed by the transformer in the form of pixel blocks [36] i.e., each row of image are embedded, and the positional information of data are encoded by using positional encoder into the input embedding, which is subsequently passed to transformer for processing.

The positional information is extracted from the data by using positional encoder E, which is added to the input embedding. The input embedding and positional encoder have same dimension d, so that both can be summed. The positional encoding is extracted using sine function by using the Eq. (6), and cosine function in Eq. (7) alternatively. The Eq. (6) is used for odd values, and Eq. (7) is used for even value as shown in the Eq. (8) for n length input sequence. The sine and cosine functions are used to create a unique pattern of values for each position.

$$E_p = sine\left(\frac{p}{f_i}\right) \tag{6}$$

$$E_p = cosine\left(\frac{p}{f_i}\right) \tag{7}$$

where p is the position to encode, and f_i are the frequencies of i numbers up to $d/2$ as shown in equation Eq. (9).

$$E_p = \begin{cases} sine\left(\frac{p}{f_1}\right) \\ cosine\left(\frac{p}{f_1}\right) \\ sine\left(\frac{p}{f_2}\right) \\ cosine\left(\frac{p}{f_2}\right) \\ \vdots \\ sine\left(\frac{p}{f_{\frac{d}{2}}}\right) \\ sine\left(\frac{p}{f_{\frac{d}{2}}}\right) \end{cases} \tag{8}$$

$$f_i => \frac{1}{\lambda_i} := 10000^{\frac{2i}{d}} \tag{9}$$

In Transformer encoder layer, the embedded input $X = \{x_1, x_2, x_3, \ldots x_n\}$ is fed into three fully connected layers to create three embeddings, namely keys, query and value; these embeddings are commonly used in the search retrieval system. During the search retrieval, the query is mapped against some keys that are associated with search candidates; this presents best match searches (values). To compute the attention value of input embedding x_1 against x_2 as shown in the Fig. 3, transformer self-attention; Firstly, the Q, K, and V are randomly initialized having same dimension as the input embedding. The input x_1 is matrix-multiplied with Q to produce Query embedding Q_e, and embedding x_2 is matrix-multiplied with K to produce Key embedding K_e, then the resultant matrixes dot product (weighted score matrix Z) is calculated. The scores Z are then scaled-down as shown in Eq. (10) for a stable gradient, where dK_e is the dimension of keys embedding.

$$Z^s = \frac{Z}{\sqrt{dK_e}} \tag{10}$$

The softmax function in Eq. (11) is applied to $Z^s = \{z_1, z_2, z_3, \ldots z_n\}$ to calculate attention weights, giving probability values between zero and one. The Fig. 3 input embedding is a matrix which is multiplied with the V_e to produce value embedding.

$$Softmax(z_i) = \frac{\exp(z_i)}{\sum_i \exp(z_i)} \tag{11}$$

$$Attention\ (K, Q, V) = Softmax\left(\frac{Q_e K_e^t}{\sqrt{dK_e}}\right) V_e \tag{12}$$

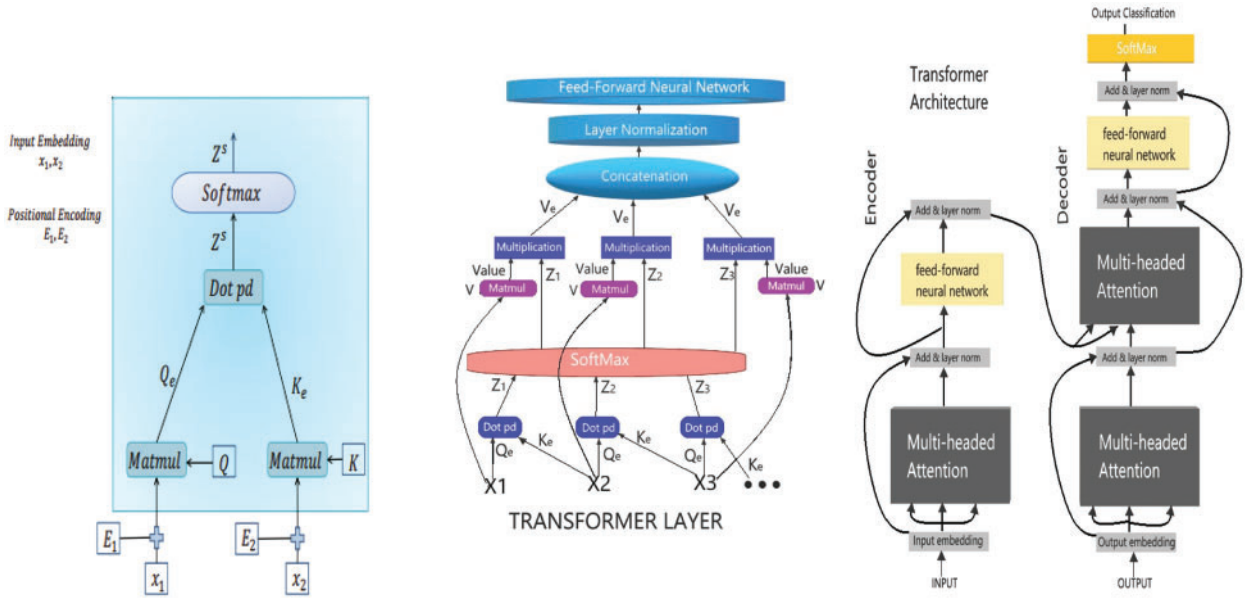

Figure 3: Transformer: Encoding layer self-attention and the transformer layer and model

The Z^s multiplies with the V_e. This process is repeated for all the inputs neighborhood, and V_e are then concatenated. The functionality is elaborated in Eq. (12), where $K_e{}^t$ is the transpose of keys embedding. The self-attention produces weighted embeddings for each input.

3 Proposed System and Architecture

The architecture proposed in this paper is a novel binary-view transformer architecture, an end-to-end model for a speech recognition system, which is inspired by the human perception of speech and is articulated by carefully studying human physiology. The architecture consists of two primary modules i.e., (i) Pre-processing and feature extraction module and (ii) Classification module.

Three convolution layers are applied for the feature extraction on both inputs. The filter size is 3×3, and numbers of filters are 32, 64, and 1, respectively. After each layer of convolution, batch normalization is implemented with an activation function. Both of the inputs are then concatenated to add the extracted features in multi-view i.e., binary view model.

Our system incorporates a modified universal transformer [19,22], where multi-head self-attention is used with four heads capturing four different contexts at the same time. The depth of the transformer is six, i.e., six encoding and six decoding transformer layers are implemented. The transformer is tuned to 25 percent dropout after each layer, and a high-performance gaussian-error linear unit [37] activation function of the neural network is used. The adaptive computations time algorithm is then used with the aim of allowing the neural network to determine computation steps for getting inputs and computing outputs. The resultant vectors then proceed to global average pooling [38], where mean value for every feature map is computed, and soft-max then determines its probabilities. Lastly, the feature map is passed to a dense layer, i.e., fully-connected layer; of 128 nodes and subsequently another dense layer of nodes equal to desire classes where the input is classified to its respective class. It is important to noted that the classification is vital by considering the fact that the internet data traffic is increasing with every passing day [39–42]. The working of system is shown in Algorithm 1.

4 Training and Experimentation

For training the proposed model, we use Google's data-set of speech command [43,44] created by Google Brain, which has speech audio files in WAV format, having a total of 105,829 command utterance files. The data-set audio files have a length of 1 s, divided into 35 classes of words. The audio was recorded in a 16-bits mono channel, and the command files are collected from 2618 different speakers having a range of dialects.

The tool used for training the architecture is Google cloud service for machine learning, namely Google-colab, which uses a jupyter-notebook environment, and Tesla Graphics Processing Unit (GPU) K80 is provided by Google having 12 GB of GPU.

We trained different architecture for speech recognition of 35 classes, which includes our binary-view transformer model and the models introduced by paper [3], i.e., LSTM and attention-based recurrent convolutional architectures. We also experimented with well-known convolutions architectures of resnet and inception net, where we modify our model by replacing the Transformer with Resnet (Fig. 4), proposed architecture (Fig. 5) and Inception net (Fig. 6). We then compute and Recurrent compare their results. We then compute and Recurrent compare their results.

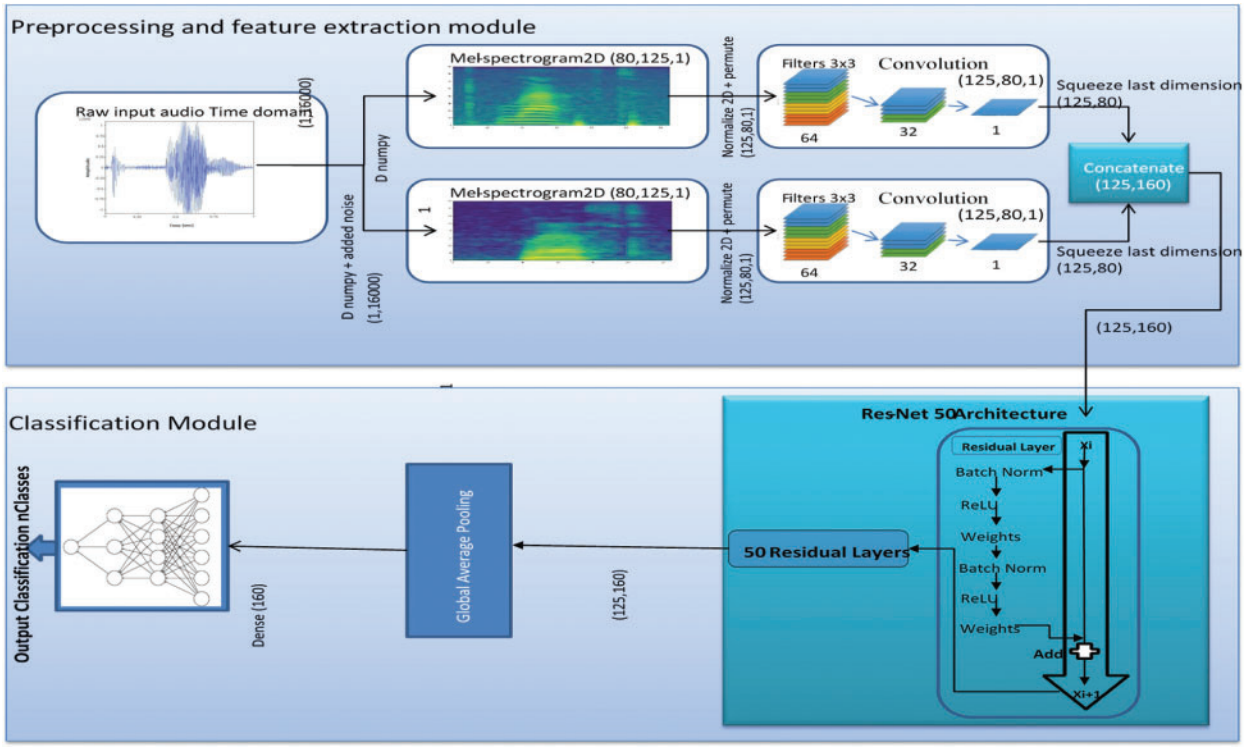

Figure 4: Binary-view ResNet

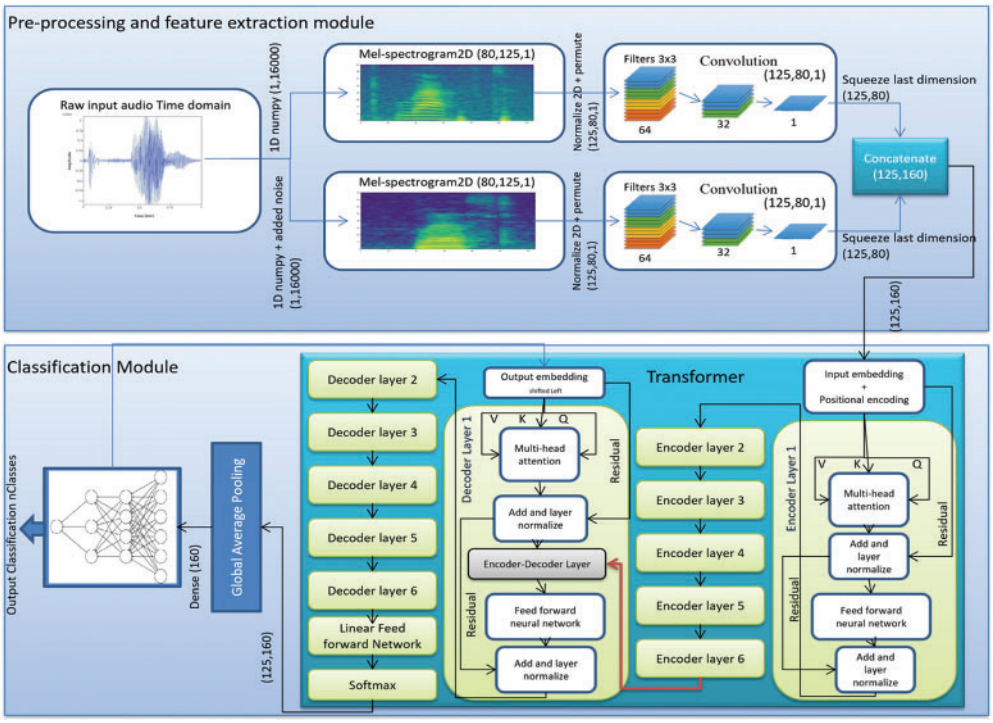

Figure 5: System architecture of binary-view convolution the transformer, which captures multiple feature contexts of each class, making recognition more rob

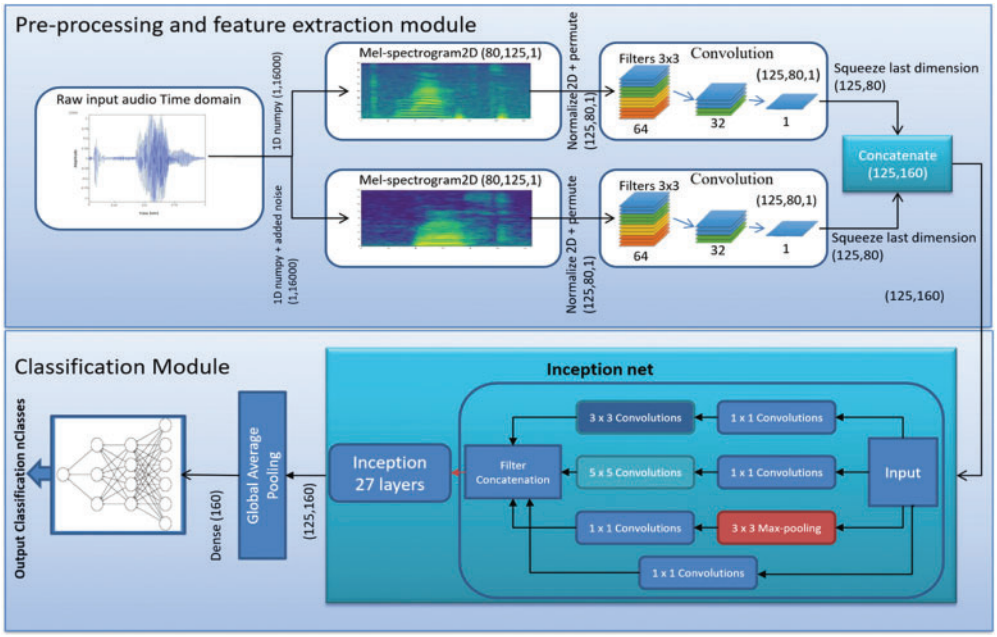

Figure 6: Binary-view inception net

5 Results and Discussion

Initially, the experiments of paper [3] were replicated, including the LSTM model and attention with a recurrent convolutional network. The purpose of the experimentation results is to compare and demonstrate the efficiency and shortcomings of different neural network models for the speech recognition. In terms of accuracy, we improved the validation accuracy by gradually decreasing the learning rate over epochs and increasing filters in convolutions and introducing batch normalization. The LSTM model [3] accuracy is recorded up to 93.9, and the attention based recurrent convolutional network [3] accuracy is 94.2. The transformer architecture without multiple inputs gives 94.24% accuracy. The binary-view resnet model, binary view inception net model, binary-view convolutional model, and binary-view transformer model were executed on the dataset, where validation accuracy was 94.91, 94.74, 95.05, and 95.16, respectively as shown in Fig. 7. Moreover, the proposed transformer model produced state-of-the-art as well as a minimalistic number of parameters, i.e., 375,787. The Fig. 8 shows the comparison of training and validation accuracies.

Algorithm 1: Speech Command Classification Using Binary-View Transformer

Input: $audio\_numpy1d_{(16000)}, Nclasses;$
Output: $ClassificationIntoNclasses$
Binary_View_Transformer =
$Speech\_Classification(Binary\_View(audio\_numpy1d_{(16000)}), Nclasses);$
`Binary_View` $(audio\_numpy1d_{(16000)})$:
$A_{(1,16000)} = Reshape(1,-1)(audio\_numpy1d_{(16000)});$
$B_{(1,16000)} = AdditiveNoise()(A_{(1,16000)});$
$ConcatenateAB(A_{(1,16000)}, B_{(1,16000)})$:
View_A:
$A_{(80,125,1)} = Mel\_Spectrogram(mels = 80, windows = 125)(A_{(1,16000)});$
$A_{(125,80,1)} = Norm2D(permute(2,1,3))(A_{(80,125,1)});$
$A_{(80,125,64)} = Conv2D(64,(3,3))(A_{(125,80,1)});$
$A_{(125,80,32)} = Conv2D(32,(3,3))(A_{(125,80,64)});$
$A_{(125,80,1)} = Conv2D(1,(3,3))(A_{(125,80,32)});$
$A_{(125,80)} = squeez\_last\_dimension()(A_{(125,80,1)});$
return $A_{(125,80)}$;
View_B:
$B_{(80,125,1)} = Mel\_Spectrogram(mels = 80, windows = 125)(B_{(1,16000)});$
$B_{(125,80,1)} = Norm2D(permute(2,1,3))(B_{(80,125,1)});$
$B_{(80,125,64)} = Conv2D(64,(3,3))(B_{(125,80,1)});$
$B_{(125,80,32)} = Conv2D(32,(3,3))(B_{(125,80,64)});$
$B_{(125,80,1)} = Conv2D(1,(3,3))(B_{(125,80,32)});$
$B_{(125,80)} = squeez\_last\_dimension()(B_{(125,80,1)});$
return $B_{(125,80)}$;
$AB_{(125,160)} = concatenate(View\_A, View\_B)$
return $AB_{(125,160)}$;
`Speech_Classification` $(AB_{(125,160)}, Nclasses)$:
$Transformer_{(125,160)} = Transformer(Input\_AB)$
Transformer:
$Transformer(depth = 6, multi\_head = 4)(Input\_AB);$
return $Transformer_{(125,160)}$;
$GAP\_AB_{(1,160)} = GlobalAveragePooling2D()(Transformer_{(125,160)});$
$NN\_AB_{(128)} = Dense(128)(GAP\_AB_{(1,160)});$
$AB\_Nclasses = Dense(Nclasses)(NN\_AB_{(128)});$
return AB_Nclasses;

In terms of loss, the binary-view transformer model validation loss is comparatively less, which is 0.191. Single input transformer model produces 0.227 loss. The binary-view resnet model, binary-view inception net model, binary-view convolutional model losses, and attention based recurrent convolutional network were 0.194, 0.192, 0.21, and 0.237, respectively as can be seen in Fig. 9. The decline of loss exhibits a better performance of the architecture and lower chance of the model being over-fitting with the aim of eradicating the gradient vanishing/exploding problem.

Table 1: Results comparison of the proposed approach with existing studies

Model	Validation loss	Validation accuracy	Training accuracy	Validation precision	Validation recall	Validation F1-score	Training precision	Training recall	Training F1-score
LSTM model [3]	0.242	82.11	98.84	-	-	-	-	-	-
Attention based recurrent convolution network [3]	0.237	84.35	98.68	-	-	-	-	-	-

(Continued)

Table 1: Continued

Model	Validation loss	Validation accuracy	Training accuracy	Validation precision	Validation recall	Validation F1-score	Training precision	Training recall	Training F1-score
Transformer	0.227	93.91	94.51	87.2	88.89	87.10	92.10	91.89	92.33
Binary-view inception Net model (Ours)	0.192	94.2	99.48	90.93	89.10	91.10	97.45	96.20	97
Binary-view ResNet Model (Ours)	0.194	94.24	**99.55**	91.10	90.23	91.93	96.7	95.8	97.12
Binary-view inception Net model (Ours)	0.21	**94.74**	99.48	92.14	92	92.5	**98.4**	**98**	**98.2**
Binary-view Transformer model (Ours)	**0.191**	95.05	99.05	**93.34**	**92**	**93.4**	97.6	96.8	97.2

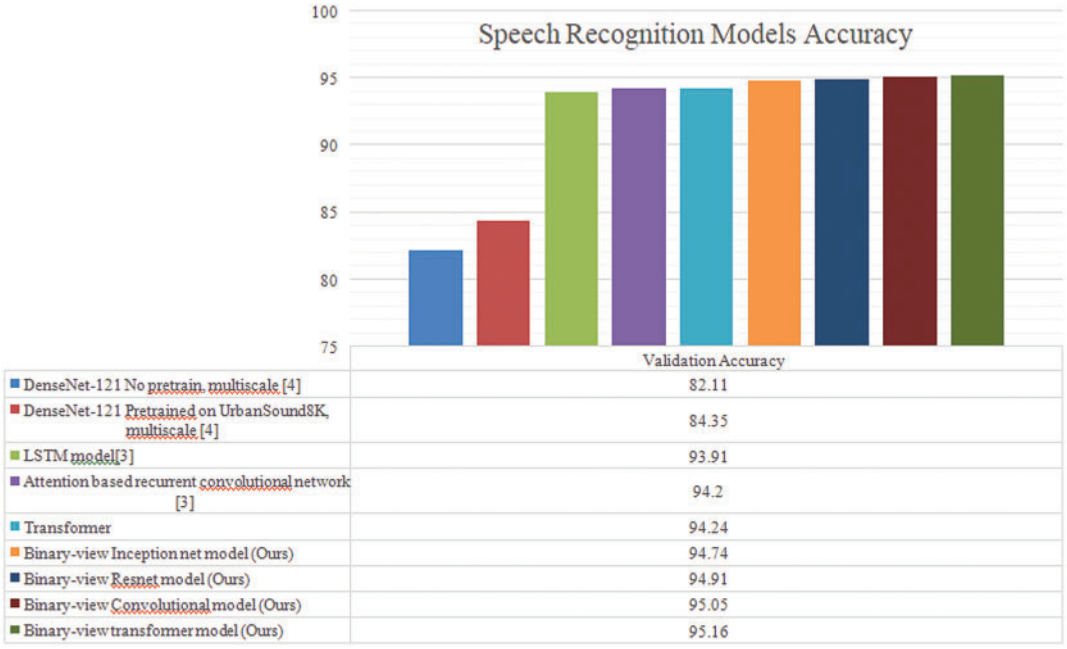

Figure 7: Validation accuracies of speech recognition models

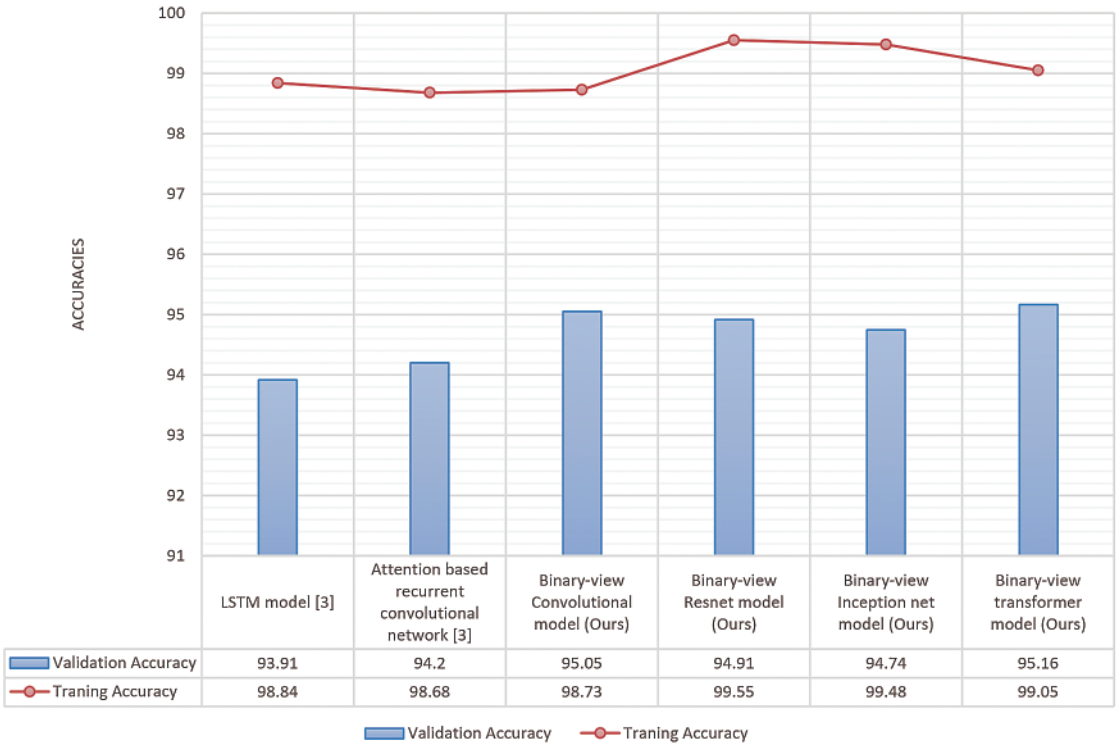

Figure 8: Training and validation accuracies of implemented models

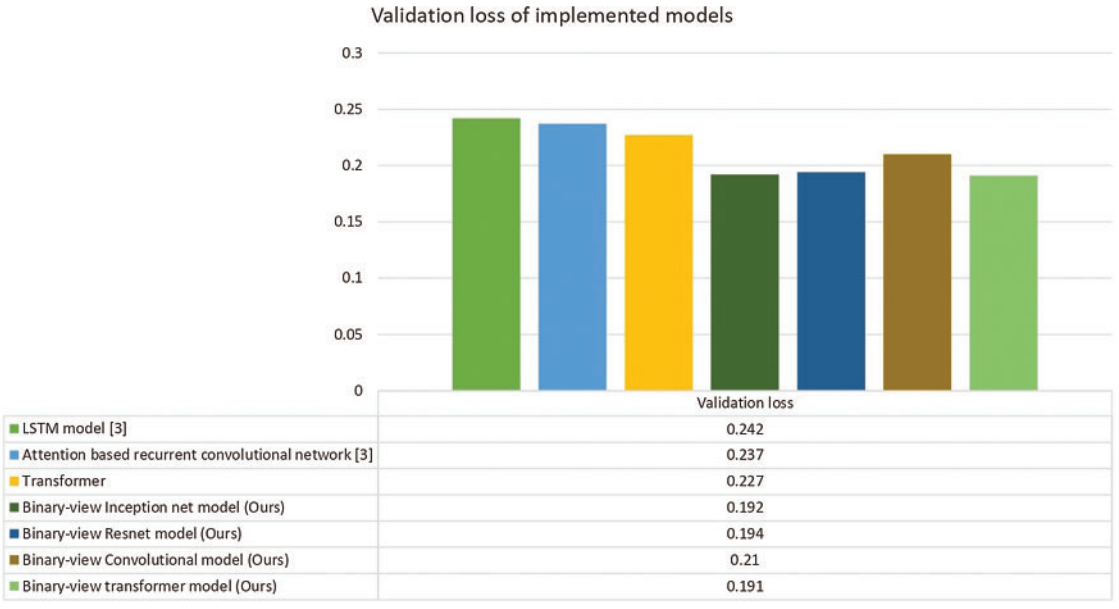

Figure 9: Validation loss of speech recognition model

6 Conclusions

This research aimed to improve the speech recognition system. We analyzed human physiology for speech perception. This research aimed to improve the speech recognition system. In addition, Binary-view transformer architecture produced state of the art results on Google's speech command dataset [43,44]. Three aspects of recognition models, i.e., validation accuracy, precision, and loss, were considered to determine the efficiency of binary-view transformer architecture. By introducing a binary-view mechanism, similar data from different sources were processed, and the attention mechanism within the transformer increases efficiency, where the best validation accuracy of 95.16 was achieved. The proposed model decreased the eventuality of gradient vanishing/exploding problem by processing long-term dependencies. Whereas the confusion matrix showed better precision of the binary-view transformer architecture compared to other models since the transformer used a multi-head attention mechanism, which catches more contexts of the same data, which helped in improving model precision and the probability of model over-fitting diminish. Better precision on Google's speech command dataset showed that our model performed better on different dialects because over 2000 speaker's speech was precisely recognized. As shown in Tab. 1, our model exhibited less loss of 0.191 compared to 0.237, 0.194, 0.192, and 0.21 of the attention based recurrent convolutional networks [3], binary-view resnet model, binary-view inception net model and binary-view convolutional model, respectively. The binary-view transformer architecture has a lightweight footprint of 375,787 trainable parameters, which can be run locally on small systems.

Funding Statement: This research was supported by Suranaree University of Technology, Thailand, Grant Number: BRO7-709-62-12-03.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] L. Kaushik, A. Sangwan and J. H. L. Hansen, "Automatic sentiment extraction from YouTube videos," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, Olomouch, Czech Republic, pp. 239–244, 2013.

[2] M. Aylett, Y. Vazquez-Alvarez and L. Baillie, "Evaluating speech synthesis in a mobile context: Audio presentation of Facebook, Twitter and RSS," in *Proc. of the ITI 2013 35th Int. Conf. on Information Technology Interfaces*, Cavtat, Croatia, pp. 167–172, 2013.

[3] D. C. D. Andrade, S. Leo, M. Viana and C. Bernkopf, "A neural attention model for speech command recognition," 2008. [Online]. Available: arXiv preprint arXiv:1808.08929.

[4] B. McMahan and D. Rao, "Listening to the world improves speech command recognition," in *Proc. of the 32nd AAAI Conf. on Artificial Intelligence*, New Orleans, LA, USA, pp. 209–215, 2018.

[5] Y. Zhang, N. Suda, L. Lai and V. Chandra, "Hello edge: Keyword spotting on microcontrollers," 2017. [Online]. Available: arXiv preprint arXiv:1711.07128.

[6] M. H. Hoy, "Alexa, Siri, Cortana, and More: An introduction to voice assistants," *Medical Reference Services Quarterly*, vol. 37, no. 1, pp. 81–88, 2018.

[7] G. López, L. Quesada and L. A. Guerrero, "Alexa *vs.* siri *vs.* cortana *vs.* google assistant: A comparison of speech-based natural user interfaces," in *Advances in Intelligent Systems and Computing*, New York, USA: Springer, pp. 241–250, 2017.

[8] M. Müller, D. Ellis, A. Klapuri and G. Richard, "Signal processing for music analysis," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, pp. 1088–1110, 2011.

[9] P. Yenigalla, A. Kumar, S. Tripathi, C. Singh, S. Kar *et al.,* "Speech emotion recognition using spectrogram & phoneme embedding," in *Interspeech*, Hyberabad, India, pp. 3688–3692, 2018.
[10] S. Dieleman and B. Schrauwen, "End-to-end learning for music audio," in *2014 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, Italy, pp. 6964–6968, 2014.
[11] J. Shen, R. Pang, R. Weiss, M. Schuster, N. Jaitly *et al.,* "Naturaltts synthesis by conditioning wavenet on mel spectrogram predictions," in *2018 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Alberta, Canada, pp. 4779–4783, 2018.
[12] M. R. Schomers and F. Pulvermüller, "Is the sensorimotor cortex relevant for speech perception and understanding? An integrative review," *Frontiers in Human Neuroscience*, vol. 10, pp. 435, 2016.
[13] G. Dede and M. HüsnüSazlı, "Speech recognition with artificial neural networks," *Digital Signal Processing*, vol. 20, no. 3, pp. 763–768, 2010.
[14] W. Feng, N. Guan, Y. Li, X. Zhang and Z. Luo, "Audio visual speech recognition with multimodal recurrent neural networks," in *2017 Int. Joint Conf. on Neural Networks (IJCNN)*, Ak, USA, pp. 681–688, 2017.
[15] J. Gonzalez and W. Yu, "Non-linear system modeling using lstm neural networks," *IFAC-PapersOnLine*, vol. 51, no. 13, pp. 485–489, 2018.
[16] R. Dey and F. M. Salemt, "Gate-variants of gated recurrent unit (gru) neural networks," in *2017 IEEE 60th Int. Midwest Symp. on Circuits and Systems (MWSCAS)*, MA, USA, pp. 1597–1600, 2017.
[17] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen *et al.,* "CNN architectures for large-scale audio classification," in *2017 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, LA, USA, pp. 131–135, 2017.
[18] Z. Cao, Y. Zhu, Z. Sun, M. Wang, Y. Zheng *et al.,* "Improving prediction accuracy in lstm network model for aircraft testing flight data," in *2018 IEEE Int. Conf. on Smart Cloud (SmartCloud)*, NY, USA, pp. 7–12, 2018.
[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones *et al.,* "Attention is all you need," in *Proc. of Advances in Neural Information Processing Systems 30 (NIPS 2017)*, CA, USA, pp. 5998–6008, 2017.
[20] M. Ghafoor, S. A. Tariq, T. Zia, I. A. Taj, A. Abbas *et al.,* "Fingerprint identification with shallow multifeature view classifier," *IEEE Transactions on Cybernetics*, vol. 51, no. 9, pp. 4515–4527, 2019.
[21] H. Fan and J. Zhou, "Stacked latent attention for multimodal reasoning," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, PR, USA, pp. 1072–1080, 2018.
[22] M. Dehghani, S. Gouws, O. Vinyals, J. Uszkoreit and Ł. Kaiser, "Universal transformers," 2018. [Online]. Available: arXiv preprint arXiv:1807.03819.
[23] M. Junaid, M. Ghafoor, A. Hassan, S. Khalid, S. A. Tariq *et al.,* "Multi-feature view-based shallow convolutional neural network for road segmentation," *IEEE Access*, vol. 8, pp. 36612–36623, 2020.
[24] M. J. Baker, J. Trevisan, P. Bassan, R. Bhargava, H. J. Butler *et al.,* "Using Fourier transform ir spectroscopy to analyze biological materials," *Nature Protocols*, vol. 9, no. 8, pp. 1771, 2014.
[25] M. Krawczyk and T. Gerkmann, "Stft phase reconstruction in voiced speech for an improved single channel speech enhancement," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 12, pp. 1931–1940, 2014.
[26] Y. L. Cun, Y. Bengio and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
[27] N. Dave, "Feature extraction methods lpc, plp and mfcc in speech recognition," *International Journal for Advance Research in Engineering and Technology*, vol. 1, no. 6, pp. 1–4, 2013.
[28] B. Jang, W. Heo, J. Kim and O. Kwon, "Music detection from broadcast contents using convolutional neural networks with a mel-scale kernel," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 11, no. 1, pp. 11, 2019.
[29] Q. Kong, Y. Xu, W. Wang and M. D. Plumbley, "Audio set classification with attention model: A probabilistic perspective," in *2018 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, Calgary, Canada, pp. 316–320, 2018.
[30] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence *et al.,* "Audio set: An ontology and human-labeled dataset for audio events," in *2017 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, LA, USA, pp. 776–780, 2017.

[31] J. Li, Z. Tu, B. Yang, M. R. Lyu and T. Zhang, "Multi-head attention with disagreement regularization," 2018. [Online]. Available: arXiv preprint arXiv:1810.10183.
[32] C. Szegedy, S. Ioffe, V. Vanhoucke and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Thirty-First AAAI Conf. on Artificial Intelligence*, California, USA, pp. 4278–4284, 2007.
[33] Z. Wu, C. Shen and A. Van Den Hengel, "Wider or deeper: Revisiting the resnet model for visual recognition," *Pattern Recognition*, vol. 90, pp. 119–133, 2019.
[34] Z. Qin, Z. Zhang, X. Chen, C. Wang and Y. Peng, "Fd-mobilenet: Improved mobilenet with a fast down sampling strategy," in *2018 25th IEEE Int. Conf. on Image Processing (ICIP)*, Athens, Greece, pp. 1363–1367, 2018.
[35] A. A. Khan, C. Wechtaisong, F. A. Khan and N. Ahmad, "A Cost-efficient environment monitoring robotic vehivle for smart industries," *CMC-Computers, Materials & Continua*, vol. 7, pp. 473–487, 2022.
[36] N. Parmar, A. Vaswani, J. Uszkoreit, Ł. Kaiser, N. Shazeer *et al.,* "Image transformer," 2018. [Online]. Available: arXiv preprint arXiv:1802.05751.
[37] D. Hendrycks and K. Gimpel, "Gaussian error linear units (gelus)," 2016. [Online]. Available: arXiv preprint arXiv:1606.08415.
[38] Z. Li, S. Wang, R. Fan, G. Cao and Y. Zhang, "Teeth category classification via seven-layer deep convolutional neural network with max pooling and global average pooling," *International Journal of Imaging Systems and Technology*, vol. 29, no. 4, pp. 577–583, 2019.
[39] A. A. Khan, P. Uthansakul, P. Duangmanee and M. Uthansakul, "Energy efficient design of massive MIMO by considering the effects of nonlinear amplifiers," *Energies*, vol. 11, pp. 1045, 2018.
[40] P. Uthansakul and A. A. Khan, "Enhancing the energy efficiency of mmWave massive MIMO by modifying the RF circuit configuration," *Energies*, vol. 12, pp. 4356, 2019.
[41] P. Uthansakul and A. A. Khan, "On the energy efficiency of millimeter wave massive MIMO based on hybrid architecture," *Energies*, vol. 12, pp. 2227, 2019.
[42] A. A. Khan, P. Uthansakul and M. Uthansakul, "Energy efficient design of massive MIMO by incorporating with mutual coupling," *International Journal on Communication Antenna and Propagation*, vol. 7, pp. 198–207, 2017.
[43] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," 2018. [Online]. Available: arXivpreprint. arXiv:1804.03209.
[44] S. N. Atluri and S. Shen, "Global weak forms, weighted residuals, finite elements, boundary elements & local weak forms," in *The Meshless Local Petrov-Galerkin (MLPG) Method*, 1st ed., vol. 1. Henderson, NV, USA: Tech Science Press, pp. 15–64, 2004.