Ingénierie des Systèmes d'Information
Vol. 28, No. 2, April, 2023, pp. 327-333
Journal homepage: http://iieta.org/journals/isi

Comparative Study of CNN Structures for Arabic Speech Recognition

Zoubir Talai*, Nada Kherici, Halima Bahi

LISCO, Badji Mokhtar University of Annaba, Annaba 23005, Algeria

Corresponding Author Email: zoubir.talai@univ-annaba.dz

https://doi.org/10.18280/isi.280208

Received: 5 January 2023
Accepted: 10 February 2023

Keywords:
convolutional neural network, Arabic speech recognition, AlexNet, GoogLeNet, ResNet

ABSTRACT

Speech recognition is an essential ability of human beings and is crucial for communication. Consequently, automatic speech recognition (ASR) is a major area of research that is increasingly using artificial intelligence techniques to replicate this human ability. Among these techniques, deep learning (DL) models attract much attention, in particular, convolutional neural networks (CNN) which are known due to their power to model spatial relationships. In this article, three CNN architectures that performed well in recognized competitions were implemented to compare their performance in Arabic speech recognition; these are the well-known models AlexNet, ResNet, and GoogLeNet. These models were compared based on a corpus composed of Arabic spoken digits collected from various sources, including messaging and social media applications, in addition to an online corpus. The architectures of AlexNet, ResNet, and GoogLeNet achieved respectively an accuracy of 86.19%, 83.46%, and 89.61%. The results show the superiority of GoogLeNet, and underline the potential of CNN architectures to model acoustic features of low-resource languages such as Arabic.

1. INTRODUCTION

The ultimate goal for an intelligent machine is to reproduce human behavior by first understanding its language, hence the importance of automatic speech recognition (ASR) systems. In this context, smart devices and systems are more efficient and reliable than ever. Home assistants like Alexa, Siri, or Google, or even self-driving cars like those made by Tesla, all rely on ASR. This is possible thanks to the advances made by researchers in the field of artificial intelligence (AI), as well as the explosion of data and better hardware. ASR systems convert the audio signal into words and commands that a computer can execute. Earlier in the 1900s, ASR systems relied on a hybrid approach, which combined a lexicon model with an acoustic model and a language model to convert a signal into a transcription [1, 2]. The traditional approach essentially suffers from limited performance, in addition to intensive time consumption and the requirement of expert phoneticians.

Thankfully, recent advances in deep learning (DL) have overcome these issues. Indeed, GPU parallel programming [3] and the huge amount of available data, have led scientists to rethink the construction of ASR systems. In particular, DL models allow the emergence of the so-called end-to-end ASR systems [4], among these models there is the convolutional neural network (CNN). Convolutional neural networks do not need as much external involvement as traditional approaches. Here the data is processed differently, instead of using a static approach to extract features, these models use convolution filters that change and evolve during training to best fit the data. CNNs have been widely used for image classification tasks [5], and some of them have become famous after reaching the top five among other competitors. This major success encouraged scientists to use CNN in ASR systems to detect and recognize words or parts of words (such as phonemes). Abdel-Hamid et al. [6] combined in a hybrid approach the Hidden Markov Models (HMM) and the Convolutional Neural Network to achieve speech recognition. They compared their results with a deep neural network (DNN) architecture, and they prove the superiority of the CNN. A similar approach was used [7], to perform phoneme recognition, the authors used the TIMIT speech corpus to train a model that provided decent results. Musaev et al. [8] considered sound files as pictures of spectrograms and applied a CNN model as image recognition architecture to classify Uzbek spoken digits. Chang and Morgan [9] introduced Gabor features as convolutional filters to enhance CNN recognition accuracy. They claimed that Gabor features are robust against noise present in the used datasets (Aurora 4, RATS) which are corrupted versions of the WSJ corpus. CNN was also used to develop an inquiry system for the airport using the Telegu language [10]. As recurrent neural networks (RNN) are known to handle temporal aspects [11], a long short-term memory, a variety of RNN, was used in conjunction with a CNN to perform continuous speech recognition. In this work, Passricha and Aggarwal [11] experimented with various weight-sharing methods, and pooling strategies to decrease the word error rate (WER). Haque et al. [12] suggested mimicking CNN for image recognition by converting the audio signal to a matrix of Mel frequency cepstral coefficients (MFCC). The proposed CNN model has been evaluated on the TIDIGITS corpus dataset and achieved 97.47% recognition rate. Gouda et al. [13] used image classification CNNs to classify a set of sound files into three different categories: A command, a silence, or an unknown word. For more details on CNN applications and their advantages [14].

As already seen, many works have dealt with CNN models for ASR, and much more works have used CNN as the basis

of speech recognizers. Herein, an important question arose: Which of the proposed models is most suitable for speech recognition? This paper aims to answer this question and to guide scientists in implementing future systems by comparing different CNN architectures. The comparison is performed between three winners of the ImageNet large-scale visual recognition competition (ILSCRC), namely, AlexNet the winner in 2012, GoogLeNet the winner in 2014, and ResNet the winner in 2015. The strengths and weaknesses of each architecture are discussed in the context of Arabic speech recognition.

While Arabic is the fifth most spoken language in the world, the development of Arabic speech recognizers is still modest due to the lack of resources [15]. CNN is expected to capture the language particularities and overcome the scarcity of resources. Moreover, the experiment outcomes would enrich the research in ASR for low-resource languages.

To conduct the comparison, a dataset of Arabic spoken digits was used. The files in this dataset were collected via messaging and social media apps [16], in addition to the corpus presented in the research [17].

The remainder of the paper is organized as follows. Section 2 describes the used architectures and their adaptation for signal processing. Section 3 outlines the difference between the models, pointing out their strengths and weaknesses. A description of the used dataset is presented in Section 4, and experimental results are detailed and discussed. Section 5 concludes the paper.

2. 1D CONVOLUTION NEURAL NETWORK

While 2D CNNs are mainly used for image classification, one-dimensional convolution networks are majorly used for signal processing problems such as automatic speech recognition, electrocardiogram monitoring, structural damage detection in civil infrastructure based on vibration, predictive maintenance for industrial machines, etc.

1D CNN operates the same way as 2D CNN, the first layer is dedicated to data input which generally consists of the raw signal. The next set of layers is a combination of convolution and pooling layers to extract features from the input signal. Finally, a fully connected network receives those features and classifies them into several classes. The last layer must have a neuron for each class.

A 1D convolution layer is mainly composed of several convolution filters with the same kernel size. Each kernel is initialized randomly and updated during model training. This kernel will move through the data by a certain "stride" to produce a convolved signal that can have the same size as the original or a reduced length depending on the padding.

A convolution is a mathematical function defined as research [18]:

$$s(t) = (x * w)(t) \tag{1}$$

The output of the operation is referred to as the feature map, herein, x is the input signal, and the w function is the kernel. In the machine learning context, x is usually a multidimensional array of data, and w is a multidimensional array of parameters that are adapted by the learning algorithm [18].

Convolutional layers are traditionally followed by pooling ones. The pooling function modifies the output of the layer at a certain location based on statistics of the nearby outputs. For example, the well-known max-pooling reports the maximum value within a rectangle neighborhood. The pooling function aims to make the computed representation invariant to small translations of the input. The size of the output can be calculated as follow:

$$outputSize = \frac{inputSize}{kernelSize} \tag{2}$$

In this work, different CNN architectures were used, and their performances were evaluated on speech recognition tasks, namely: AlexNet, ResNet, and GoogLeNet which are three popular architectures that proved their efficiency in image classification. Obviously, each network must be adapted for signal processing using 1D convolution and pooling layers.

2.1 AlexNet

AlexNet network was proposed by Krizhevsky et al. [19], the model is based on the work of Lecun et al. [20] presented and improved [21]. AlexNet is composed of three sets of convolution/pooling layers, the first two sets are simple, and the third one consists of three consecutive convolutions followed by a pooling layer, all of them are combined with three fully connected layers. AlexNet architecture was used to classify 1.2 million images into 1000 different classes.

For the purpose of this work, the original philosophy of each architecture is respected and adapted for signal processing. 1D convolution/pooling layers were combined with a dropout layer to avoid overfitting. Also, an extra set of convolution/pooling layers was added due to the size of the sound file. Figure 1 shows the used architecture.

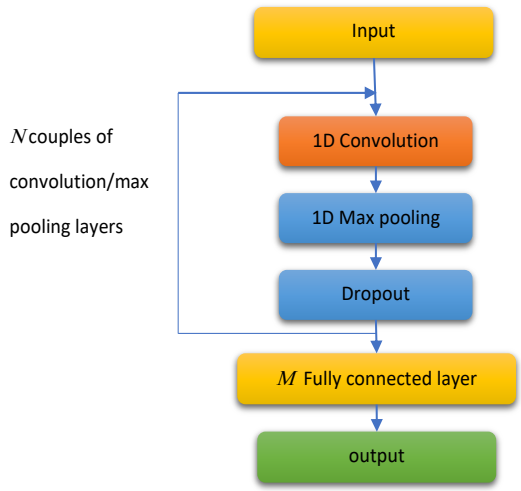


Figure 1. AlexNet architecture

While the input layer receives the raw speech, the output layer provides the recognized word label.

2.2 ResNet

Residual Network or ResNet was introduced by He et al. [22] to overcome the challenges of going deeper with neural networks. Indeed, the problem of vanishing gradient is faced once the network's architecture has more than a certain number of layers. To solve this issue, the authors proposed to sum the input and output at the end of each set of convolutions

layers [22]. This way, part of the input data is kept to be fed to the next layers. Figure 2 illustrates this principle.

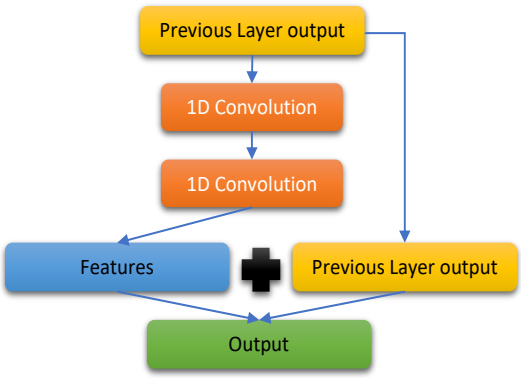


Figure 2. ResNet principle

2.3 GoogLeNet (Inception)

GoogLeNet model was presented during the competition ImageNet Large-Scale Visual Recognition Challenge held in 2014. GoogLeNet proposed an architecture with 22 layers with special blocks named 'Inception' blocks [23]. In each block, there are three different convolutions with one max-pooling layer that take the same output from the previous layer and concatenate their result to feed the next layer. The blocks used 1x1, 3x3, and 5x5 kernels for the convolutions and a 3x3 kernel for max pooling. This architecture was adapted to signal processing and speech recognition, which was done by replacing 2D convolutions and max-pooling with the appropriate 1D operations. Figure 3 shows an inception block.

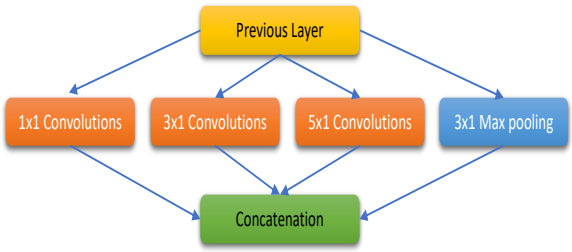


Figure 3. 1D inception block

3. COMPARATIVE STUDY

For the purpose of this study, the previous models were studied in depth and compared in the context of speech recognition of Arabic words.

For automatic speech recognition, the nature of a sound file and its composition are the keys to choosing adequate CNN architecture. As known, a spoken word is composed of phonemes, which can be considered sound elements. In Arabic, each phoneme produces a unique signal that corresponds to a particular letter. To achieve automatic speech recognition, first, the system must recognize each element separately, then, it should recognize the exact phoneme combination that produces a given word.

Each architecture presented in the previous section deals with signal processing differently. The sequential nature of AlexNet can be seen as its strength, and its simplicity of implementation as well. In fact, each couple of convolution/pooling layers extracts as many features as the user wants. The main problem faced using this architecture is when the network goes more profound. In this case, the same strength turns into a weakness that causes the vanishing gradient issue. At each set of convolution/pooling layers, parts of the original information are lost, and at a certain level, there is no relationship between the input and the output.

ResNet overcomes this weakness by adding the original information to the features extracted at the end of each set of convolution layers. This way, the network can go deeper by maintaining parts of input information and using different convolution combinations. Although the significant improvement in recognition rates, this architecture suffers from time consumption from adding more layers. This raises another question: "how can a deep model be trained in an acceptable time without losing input information?

GoogLeNet answers the question by implementing "Inception Blocks". These blocks process the same input information with different convolution/pooling operations, starting with a one-by-one (1x1) kernel that can deal with signal peaks. The next one is extended to three by one (3x1) to cover more information, and finally, a five-by-one (5x1) kernel is used to extract the phonemes. Along with these convolutions, max-pooling is used to summarize input information. All of the previous operations take the same input as shown in Figure 3. The result of each Inception block is the concatenation of all the outputs of those operations. These blocks allow different combinations of convolutions and pooling without the need to go deeper.

One particular strength of GoogLeNet architecture is that the network can have multiple auxiliary classifiers. These are added in the middle of the network to overcome the vanishing gradient issue. They are only used during the training of the model, and the loss computed at their end is weighted.

4. RESULTS AND DISCUSSION

4.1 Dataset

Table 1. Arabic digits pronunciation and syllables

Digit	Arabic transcription	Pronunciation	Syllables	Number of Syllables
0	صفر	Sĕfr	CVCC	1
1	واحد	wa-hĕd	CV-CVC	2
2	اثنين	ĕth-nāyn	CVC-CVCC	2
3	ثلاثة	tha-lăthah	CV-CV-CVC	3
4	أربعة	aar-ba-'aah	CVC-CV-CVC	3
5	خمسة	kham-sah	CVC-CVC	2
6	ستة	sĕt-tah	CVC-CVC	2
7	سبعة	sub-'aah	CVC-CVC	2
8	ثمانية	tha-mă-nyĕh	CV-CV-CVC	4
9	تسعة	Tĕsâh	CVC-CVC	2

Modern Standard Arabic has 34 phonemes: 28 consonants and six vowels [24], and the syllables in Arabic have six patterns: CV, CV-, CVC, CV-C, CVCC, and CV-CC, where V is a vowel, C is a consonant, and V- a long vowel. Arabic words can only start with a consonant. The Arabic digits are polysyllabic words except zero which is a monosyllable [25]. Table 1 presents the ten Arabic digits, their pronunciation, the number of syllables, and the types of syllables.

For the experimentation, two corpora were combined to cope with limitations related to the lack of labeled Arabic speech corpora. Spoken utterances of the ten Arabic digits were collected from 107 speakers; the recordings were sent via social media apps (WhatsApp, Facebook Messenger). The speakers were women, men, and children, speakers were aged from 4 to 64 years. The collected speech recordings were converted into wave-form files of two seconds duration and were labeled; it resulted in 1070 samples.

The second used corpus is described [17], it contains 9992 utterances of 20 words spoken by 50 native male Arabic speakers. We used only the digits utterances (4996) and discard any other words.

This amount of data is not sufficient for a deep learning model; hence we augmented the obtained dataset by using audio augmentation techniques such as:

- Pich changing.
- Adding Noise.
- Sound stretching.
- Time shifting.
- Harmonic-percussive source separation.
- Silence shifting.

Finally, the obtained corpora contain 54479 files.

4.2 CNN compared architectures

The following tables show the proposed architecture for each of the chosen models, according to the requirement related to speech processing as opposed to image processing. The dropout layers are inserted to overcome overfitting related to the lack of Arabic speech data. ReLU activation function was used in all of the convolution layers and hidden layers, and SoftMax was used in all the final layers to classify the input signal into one of the ten Arabic digits.

Table 2 shows the used architecture for AlexNet which is simple and forward, the kernel size is progressively increased as the decreasing of filters counts. The dropout layer was added to avoid overfitting.

Table 2. The proposed AlexNet model

AlexNet
Conv1D (8, 13)
Max-Pooling (3)
Dropout (0.3)
Conv1D (16, 11)
Max-Pooling (3)
Dropout (0.3)
Conv1D (32, 9)
Max-Pooling (3)
Dropout (0.3)
Conv1D (64, 7)
Max-Pooling (3)
Dropout (0.3)
Dense (256)
Dense (128)
Dense (10)

Table 3 summarizes the composition of the used ResNet model. It is considered that after three successive convolutions, a certain amount of input data will be lost, thus, the output is combined with the previous input as explained in the previous section.

Table 3. The proposed ResNet model

ResNet
Conv1D (1,64)
Conv1D (3,64)
Conv1D (1,256)
Sum(Previous input + output)
Conv1D(1,64)
Conv1D (3,64)
Conv1D (1,256)
Sum(Previous input + output)
Conv1D (1,128)
Conv1D (3,128)
Conv1D (1,512)
Sum(Previous input + output)
Conv1D (1,128)
Conv1D (3,128)
Conv1D (1,512)
Sum(Previous input + output)
Dense(512)
Dropout (0.3)
Dense(128)
Dropout (0.3)
Dense(10)

Table 4. The proposed GoogLeNet model

GoogLeNet
Inception (16,16,16,16,8,16)
Inception (16,16,16,16,8,16)
Average-pooling (5)
Conv1D (32,1)
Dense (10)
Inception (32,32,32,32,16,32)
Inception (32,32,32,32,16,32)
Dense (10)

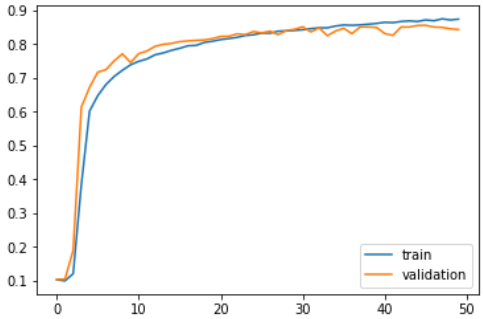
Table 5. Inception block architecture

Inception (a, b, c, d, e, f)			
Conv1D (a,1)	Conv1D (b,1)	Conv1D (d,1)	Max-Pooling (3)
	Conv1D (c,3)	Conv1D (e,5)	Conv1D (f,1)
Concatenation			

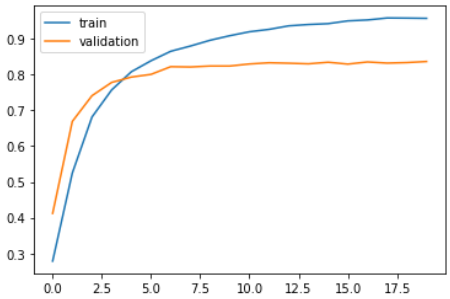
As shown in Tables 4 and 5, the used GoogLeNet model is composed of four (4) Inception blocks with one auxiliary classifier inserted midway that is noticeable in the sequence: Average pooling – Conv1D – Dense. Note that the number of extracted features, as well as the kernel size, have to be chosen meticulously so that the output size of each layer matches the input needed for the next one.

4.3 Results and discussion

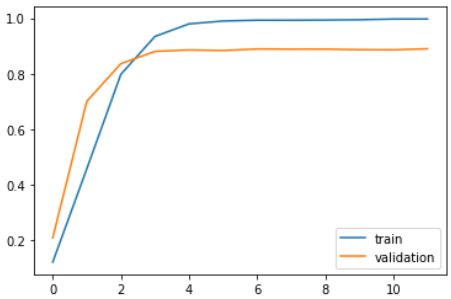
Figure 4 reports the accuracy progression for each used model, during the training and validation stages, according to the number of epochs.



(a)



(b)



(c)

Figure 4. Models' accuracy evolution: 4a: AlexNet; 4b: ResNet; 4c: GoogLeNet

To better appreciate the impact of the epochs' number on the training performances, Table 6 reports the results in terms of accuracy for the three models for the training and validation stages.

Table 6. Comparison of the three models in terms of accuracy (%)

	AlexNet	ResNet	GoogLeNet
Batch size	16	128	32
Epochs	50	20	12
Training accuracy	87.46	95.67	99.70
Validation accuracy	84.78	83.49	88.96

The results presented in Table 6 show the performance of each architecture. Considering the corrupted nature of the input data (transmission noise, quantification noise, etc.), the simplicity of implementation, and the arguably small network size, AlexNet performed well in terms of accuracy. As expected, it stops evolving at a certain accuracy even with 50 epochs. This can be explained by the fact that the sequential nature of this architecture affects its accuracy, in more detail, after each set of convolution/max-pooling layers a certain amount of data is lost, which negatively affects the backpropagation process, since the error impact on weights update is almost insignificant. Extending the network will produce worse results and shrinking it reduces its training ability to handle big datasets.

On the other hand, ResNet achieved better results after only 20 epochs, this is obviously due to adding parts of the original input data to every convolution's output. This allows the network expansion using thrice the convolution operations of AlexNet and improving recognition accuracy. The simplicity and expansion ability of this architecture make it a recommendable choice for ASR systems. Nevertheless, the training takes quite some time due to its size.

Finally, GoogLeNet outperforms both of the previous networks after only twelve epochs. It does that by solving each network's weaknesses. Firstly, the composition of Inception blocks addresses the loss of data caused by the sequential nature of AlexNet and ResNet. Different convolution operations are applied to the same data and their results are combined which makes feature extraction more efficient. Secondly, the auxiliary output layer solves the vanishing gradient issue by adding checkpoints at different locations in the network. These layers maintain the impact of error during backpropagation. This suggests that this model is the most suitable for automatic speech recognition tasks.

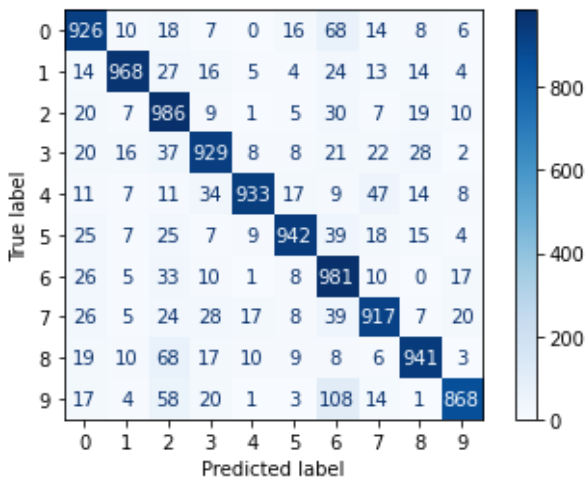
Finally, Table 7 reports the accuracy obtained during the test stage, for each of the ten digits.

Table 7. Model-wise accuracy (%) for each digit

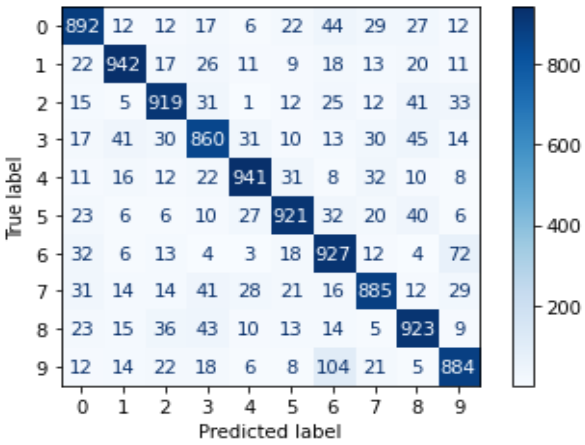
Spoken Digits	AlexNet	ResNet	GoogLeNet
0	86.30	83.13	91.61
1	88.89	86.50	90.08
2	90.13	84.00	89.40
3	85.15	78.83	90.65
4	85.52	86.25	91.93
5	86.34	84.42	90.93
6	89.92	84.97	91.38
7	84.05	81.12	91.29
8	86.25	84.60	85.61
9	79.34	80.80	83.18
Average	86.19	83.46	89.61

Table 7 confirms the superiority of GoogLeNet model, as it reaches an accuracy of about 89.61% outperforming the two other models.

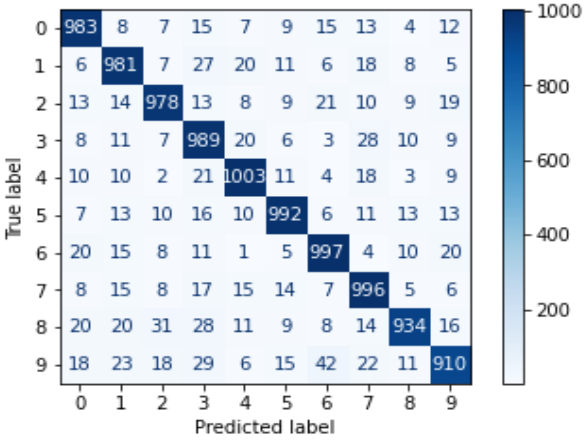
Figure 5 illustrates the confusion matrix of each architecture, and it can be noticed that all of the three models misclass the nine (9) as a six (6) due to their pronunciation in the Arabic language (6:"siteh" and 9:"tis3eh"), the same thing goes for zero and six (0:"sifr" and 6:"siteh").



(a)



(b)



(c)

Figure 5. Models' confusion matrix: 5a: AlexNet; 5b: ResNet; 5c: GoogLeNet

5. CONCLUSIONS

In this paper, the performances of three CNN architectures, namely: AlexNet, ResNet, and GoogLeNet were compared in the context of automatic Arabic speech recognition. The principle of each model was explained, and how each one of them deals with the weaknesses of the other two. Based on the results, it can be concluded that the simplicity of AlexNet may be considered a strength but it limits its performance especially considering its deep architecture. This can be overcome using ResNet which allows gaining performance by going deeper without facing the vanishing gradient problem. Due to its size, this model needs a significant amount of time to be trained. This differs from GoogLeNet, which is trained relatively quicker in addition to being robust in facing the vanishing gradient problem. Although, this model's implementation can be complex. Overall, the results presented in this paper should help researchers decide which model to consider for automatic speech recognition.

Concerning Arabic speech recognition, the obtained results suggest that GoogleNet outperforms significantly the two other models, and could be considered an interesting basis for Arabic speech recognizers.

REFERENCES

[1] Ghai, W., Singh, N. (2012). Literature review on automatic speech recognition. International Journal of Computer Applications, 41: 42-50. https://doi.org/10.5120/5565-7646

[2] Bahi, H., Sellami, M. (2001). Combination of vector quantization and hidden Markov models for Arabic speech recognition. Proceedings ACS/IEEE International Conference on Computer Systems and Applications, pp. 96-100. https://doi.org/10.1109/AICCSA.2001.933957

[3] Li, X.Q., Zhang, G.Y., Huang, H.H., Wang, Z.F., Zheng, W.M. (2016). Performance analysis of GPU-based convolutional neural networks. In 2016 45th International Conference on Parallel Processing (ICPP), pp. 67-76. http://dx.doi.org/10.1109/ICPP.2016.15

[4] Pouyanfar, S., Sadiq, S., Yan, Y.L., Tian, H.M., Tao, Y.D., Reyes, M.P., Shyu, M., Chen, S.C., Iyengar, S.S. (2018). A survey on deep learning: Algorithms, techniques, and applications. ACM Computing Surveys, 51(5): 1-36. https://doi.org/10.1145/3234150

[5] Li, Z.W., Liu, F., Yang, W.J., Peng, S.H., Zhou, J. (2021). A survey of convolutional neural networks: Analysis, applications, and prospects. IEEE Transactions on Neural Networks and Learning Systems, 33(12): 1-21. https://doi.org/10.1109/TNNLS.2021.3084827

[6] Abdel-Hamid, O., Mohamed, A., Jiang, H., Deng, L., Penn, G., Yu, D. (2014). Convolutional neural networks for speech recognition. IEEE/ACM Transactions on audio, speech, and language processing, 22(10): 1533-1545. https://doi.org/10.1109/TASLP.2014.2339736

[7] Glackin, C., Wall, J.A., Chollet, G., Dugan, N., Cannings, N. (2018). Convolutional neural networks for phoneme recognition. In 7th International Conference on Pattern Recognition Applications and Methods, pp. 190-195. http://dx.doi.org/10.5220/0006653001900195

[8] Musaev, M., Khujayorov, I., Ochilov, M. (2019). Image approach to speech recognition on CNN. In Proceedings of the 2019 3rd International Symposium on Computer Science and Intelligent Control, 57: 1-6. https://doi.org/10.1145/3386164.3389100

[9] Chang, S.Y., Morgan, N. (2014). Robust CNN-based speech recognition with Gabor filter kernels. Interspeech, pp. 1-5. https://www.icsi.berkeley.edu/pubs/speech/robustCNN14.pdf.

[10] Dimmita, N., Siddaiah, P. (2018). Speech recognition using convolutional neural networks. International Journal of Engineering & Technology, 7(4): 133-137. http://dx.doi.org/10.14419/ijet.v7i4.6.20449

[11] Passricha, V., Aggarwal, R.K. (2019). A hybrid of deep CNN and bidirectional LSTM for automatic speech recognition. Journal of Intelligent Systems, 29(1): 1261-1274. https://doi.org/10.1515/jisys-2018-0372

[12] Haque, M.A., Verma, A., Alex, J.S.R., Venkatesan, N. (2020). Experimental evaluation of CNN architecture for speech recognition. In First International Conference on Sustainable Technologies for Computational Intelligence, pp. 507-514. http://dx.doi.org/10.1007/978-981-15-0029-9_40

[13] Gouda, S.K., Kanetkar, S., Harrison, D., Warmuth, M.K. (2018). Speech recognition: Keyword spotting through image recognition. ArXiv:1803.03759. https://arxiv.org/pdf/1803.03759.pdf.

[14] Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M., Inman, D.J. (2021). 1D convolutional neural

networks and applications: A survey. Mechanical Systems and Signal Processing, 151: 107398. http://doi.org/10.1016/j.ymssp.2020.107398

[15] Dendani, B., Bahi, H., Sari, T. (2021). Self-supervised speech enhancement for Arabic speech recognition in real-world environments. Traitement du Signal, 38(2): 349-358. https://doi.org/10.18280/ts.380212

[16] Talai, Z., Bahi, H., Kherici, N. (2022). Remote spoken Arabic digits recognition using CNN. In Proceedings of the 11th International Conference on Robotics, Vision, Signal Processing and Power Applications, 829: 639-645. http://dx.doi.org/10.1007/978-981-16-8129-5_97

[17] Alalshekmubarak, A., Smith, L.S. (2014). On improving the classification capability of reservoir computing for Arabic speech recognition. In International Conference on Artificial Neural Networks, 8681: 225-232. https://doi.org/10.1007/978-3-319-11179-7_29

[18] Heaton, J., Goodfellow, I., Bengio, Y., Courville, A. (2018). Deep learning. Genetic Programming and Evolvable Machines, 19: 305-307. https://doi.org/10.1007/s10710-017-9314-z

[19] Krizhevsky, A., Sutskever, I., Hinton, G.E. (2012). Imagenet classification with deep convolutional neural networks. Communications of the ACM, 60: 84-90. https://doi.org/10.1145/3065386

[20] Lecun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D. (1989). Backpropagation applied to handwritten zip code recognition. Neural Computation, 1(4): 541-551. https://doi.org/10.1162/neco.1989.1.4.541

[21] Lecun, Y., Bottou, L., Bengio, Y., Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11): 2278-2324. https://doi.org/10.1109/5.726791

[22] He, K.M., Zhang, X.Y., Ren, S.Q., Sun, J. (2016). Deep residual learning for image recognition. In 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770-778. https://doi.org/10.1109/CVPR.2016.90

[23] Szegedy, C., Liu, W., Jia, Y.Q., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinvich, A. (2015). Going deeper with convolutions. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1-9. https://doi.org/10.1109/CVPR.2015.7298594

[24] El-Imam, Y.A. (2001). Synthesis of Arabic from short sound clusters. Computer Speech & Language, 15(4): 355-380. https://doi.org/10.1006/csla.2001.0172

[25] Alotaibi, Y.A. (2004). Spoken Arabic digits recognizer using recurrent neural networks. Proceedings of the Fourth IEEE International Symposium on Signal Processing and Information Technology, pp.195-199. https://doi.org/10.1109/ISSPIT.2004.1433720