Intelligent Automation & Soft Computing
DOI: 10.32604/iasc.2023.034069
Article

Tech Science Press

Deep Learning Driven Arabic Text to Speech Synthesizer for Visually Challenged People

Mrim M. Alnfiai[1,2], Nabil Almalki[1,3], Fahd N. Al-Wesabi[4,*], Mesfer Alduhayyem[5], Anwer Mustafa Hilal[6] and Manar Ahmed Hamza[6]

[1]King Salman Center for Disability Research, Riyadh, 13369, Saudi Arabia
[2]Department of Information Technology, College of Computers and Information Technology, Taif University, P.O. Box 11099, Taif, 21944, Saudi Arabia
[3]Department of Special Education, College of Education, King Saud University, Riyadh, 12372, Saudi Arabia
[4]Department of Computer Science, College of Science & Arts at Muhayel, King Khaled University, Abha, 62217, Saudi Arabia
[5]Department of Computer Science, College of Sciences and Humanities-Aflaj, Prince Sattam bin Abdulaziz University, Al-Aflaj, 16733, Saudi Arabia
[6]Department of Computer and Self Development, Preparatory Year Deanship, Prince Sattam bin Abdulaziz University, AlKharj, 16242, Saudi Arabia
*Corresponding Author: Fahd N. Al-Wesabi. Email: falwesabi@kku.edu.sa
Received: 05 July 2022; Accepted: 14 October 2022

Abstract: Text-To-Speech (TTS) is a speech processing tool that is highly helpful for visually-challenged people. The TTS tool is applied to transform the texts into human-like sounds. However, it is highly challenging to accomplish the TTS outcomes for the non-diacritized text of the Arabic language since it has multiple unique features and rules. Some special characters like gemination and diacritic signs that correspondingly indicate consonant doubling and short vowels greatly impact the precise pronunciation of the Arabic language. But, such signs are not frequently used in the texts written in the Arabic language since its speakers and readers can guess them from the context itself. In this background, the current research article introduces an Optimal Deep Learning-driven Arab Text-to-Speech Synthesizer (ODLD-ATSS) model to help the visually-challenged people in the Kingdom of Saudi Arabia. The prime aim of the presented ODLD-ATSS model is to convert the text into speech signals for visually-challenged people. To attain this, the presented ODLD-ATSS model initially designs a Gated Recurrent Unit (GRU)-based prediction model for diacritic and gemination signs. Besides, the Buckwalter code is utilized to capture, store and display the Arabic texts. To improve the TSS performance of the GRU method, the Aquila Optimization Algorithm (AOA) is used, which shows the novelty of the work. To illustrate the enhanced performance of the proposed ODLD-ATSS model, further experimental analyses were conducted. The proposed model achieved a maximum accuracy of 96.35%, and the experimental outcomes infer the improved performance of the proposed ODLD-ATSS model over other DL-based TSS models.

Keywords: Saudi Arabia; visually challenged people; deep learning; Aquila optimizer; gated recurrent unit

1 Introduction

The term 'Visually Impaired (VI)' denotes individuals with non-recoverable low vision or no vision [1]. Per a survey conducted earlier, 89% of visually-impaired persons live in less-developed nations, whereas nearly half are women. Braille is a language that was particularly developed for those who lost their vision. It has numerous compilations of six-dot patterns [2]. In the Braille language, the natural language symbols about the sounds are indicated by the activation or deactivation of the dots. Primarily, the Braille language is written and read on slates or paper with raised dots. It was introduced by Perkin Brailler, a Braille typewriter, who specifically created this writing pattern for visually-impaired people. With the arrival of technology, multiple mechanisms and touchscreen gadgets have been developed. Their applications have expanded in terms of smart magnifiers, talkback services, navigators, screen readers and obstacle detection & avoidance mechanisms for visually-impaired people [3]. The Braille language helps them to interact with their counterparts and others globally. It helps enhance natural language communication and conversational technology for educational objectives [4]. Most vision-impaired people hesitate to leverage their smartphones due to usability and accessibility problems. For instance, they find it challenging to identify the location of a place on smartphones. Whenever vision-impaired persons execute a task, a feedback mechanism should be made available to them to provide the result [5,6].

Conventional Machine Learning (ML) techniques are broadly utilized in several research fields. But, such methodologies necessitate profound knowledge in the specific field and utilize the pre-defined characteristics for assessment [7,8]. So, it becomes essential to manually execute an extensive range of feature extraction works in the existing techniques. Deep Learning (DL), one of the ML techniques, is a sub-set of Artificial Intelligence (AI) techniques. It is characterized by a scenario in which computers start learning to think based on the structures found in the human brain [9]. The DL techniques can examine the unstructured data, videos and images in several ways, whereas ML techniques cannot easily perform these tasks [10]. The ML and the DL techniques are applied in multiple industrial domains, whereas language plays a significant role in the day-to-day life of human beings. Language is of prime importance, whether it may be a passion, speech, a coding system or sign language, i.e., to convey meaning via touch. It expresses one's experiences, thoughts, reactions, emotions and intentions. The Text-To-Speech (TTS) synthesizer converts the language data, stored in the form of text, to a speech format. Recently, it has been primarily utilized in audio-reading gadgets for vision-impaired persons [11].

TTS has become one of the major applications of the Natural Language Processing (NLP) technique. Various researchers have worked on speech synthesis processes in literature since the significance of novel applications has increased, such as the information retrieval services over the telephone like banking services, announcements at public locations such as train stations and reading manuscripts to collect the data [12]. Many research works have been conducted earlier in two languages, English and French, in the domain of TTS. However, other languages like Arabic are yet to be explored in detail. Sufficient space exists for the growth and development of research works in this arena. So, the development of an Arabic Text-To-Speech system is still in the nascent stage. Hence, this project's scope is limited in terms of providing the guidelines for developing an Arabic speech synthesis technique and changing the methodology to overcome the difficulties experienced by the authors in this domain. This is done to help the researchers build highly-promising Voice-to-Text applications for the Arabic language [13].

In the study conducted earlier [14], a new structure was devised for signer-independent sign language detection with the help of multiple DL architectures. This method had hand-shaped feature representations, Deep Recurrent Neural Network (RNN) and semantic segmentation. An advanced semantic segmentation technique, i.e., DeepLabv3+, was trained to utilize pixel-labelled hand images to extract hand regions from every frame of the input video. Then, the extracted hand regions were scaled and cropped to a fixed size to alleviate the hand-scale variations. The hand-shaped features were attained with the help of a single-layer Convolutional Self-Organizing Map (CSOM) instead of a pre-trained Deep Convolutional

Neural Network (CNN). In literature [15], a prototype was developed for a text-to-speech synthesizer for Tigrigna Language.

In literature [16], the Arabic version of the data was constructed as a part of MS COCO and Flickr caption data sets. In addition, a generative merger method was introduced to caption the images in the Arabic language based on CNN and deep RNN methods. The experimental results inferred that when using a large corpus, the merged methods can attain outstanding outcomes in the case of Arabic image captioning. The researchers [17] investigated and reviewed different DL structures and modelling choices for recognising Arabic handwritten texts. Moreover, the imbalanced dataset issue was overcome by offering the model to the DL mechanism. To face this problem, a new adaptive data-augmentation method was presented to promote class diversity. Every word was allocated weight in the database lexicon. The authors [18] aim to automatically detect the Arabic Sign Language (ArSL) alphabet using an image-related method. To be specific, several visual descriptors were analyzed to construct an accurate ArSL alphabet recognizer. The derived visual descriptors were conveyed to the One-*vs*.-All Support Vector Machine (SVM) method.

The current study introduces an Optimal Deep Learning-driven Arab Text-to-Speech Synthesizer (ODLD-ATSS) model to help the visually-challenged people in the Kingdom of Saudi Arabia. The prime objective of the presented ODLD-ATSS model is to convert the text into speech signals for the visually-challenged people. To attain this, the presented ODLD-ATSS model initially designs a Gated Recurrent Unit (GRU)-based prediction model for diacritic and gemination signs. Besides, the Buckwalter code is also utilized to capture, store and display the Arabic text. In order to enhance the TSS performance of the GRU model, the Aquila Optimization Algorithm (AOA) is used. Numerous experimental analyses were conducted to establish the enhanced performance of the proposed ODLD-ATSS model.

2 The Proposed Model

In this study, a novel ODLD-ATSS technique has been proposed for the TSS process so that it can be used by the visually-challenged people in the Kingdom of Saudi Arabia. The presented ODLD-ATSS model aims to convert text into speech signals for visually-impaired people. The overall working process of the proposed model is shown in Fig. 1.

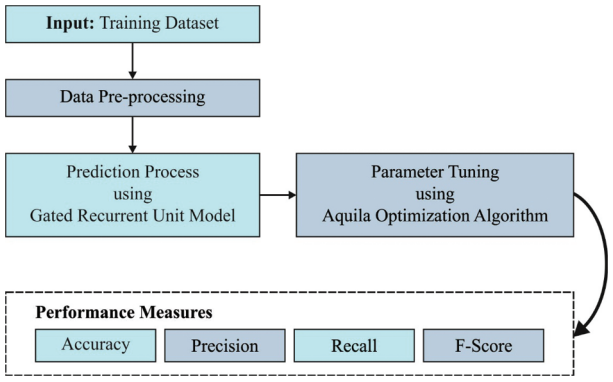

Figure 1: Overall process of the ODLD-ATSS model

2.1 GRU-Based Diacritic and Gemination Sign Prediction

The presented ODLD-ATSS model initially designs a GRU-based prediction model for diacritic and gemination signs. The RNN characterizes the Neural Network (NN) through multiple recurrent layers in the form of hidden layers. It diverges from others in the reception of an input dataset. The RNN method

is mainly applied in sequence datasets that are closely connected with time series datasets, for instance, stock prices, language, weather and so on. So, the previous dataset creates an impact on the outcomes of the model [19]. The RNN model is a well-developed model that can process the time series datasets easily; it employs the recurrent bias as well as the weights by passing the dataset via a cyclic vector. It attains the X input vector and generates an output vector, y. It has a Fully Connected (FC) structure that is classified based on the unlimited length of the input and output values. The shape and the style are generated for the network by modifying its architecture. However, the existing RNN suffers from long-term dependency issues [19]. The weight gets deviated towards infinity or gets converged towards 0 as the time lag progresses. As a result, the Long Short-Term Memory (LSTM) model is devised to overcome the long-term dependency issue of the existing RNN model. It is identified as a different architecture from RNN and is classified as the occurrence of a cell state. The LSTM model contains the forget gate, output gate and the input gate. The application of the LSTM model brings a remarkable capability to remember the long-term dependencies, as it takes a long period to train the module due to its complex architecture. Consequently, the GRU model is developed to accelerate the trained technique. It is a type of RNN structure that has a gate model based on a simple structure and the LSTM model.

The GRU model is a variant of the LSTM model whereas the latter contains three gate functions with respect to RNN such as the output gate, input gate and the forget gate to the output value, control input and the memory respectively. There are two gates present in the GRU mechanism such as the upgrade gate and the reset gate. Fig. 2 shows a specific architecture in which σ represents the gating function. This technique effectively reduces the calculation amount and the possibility of vanishing the gradient explosions. A specific function model is shown herewith [20]:

$$Z_t = \sigma(W_z \cdot [h_{t-1'} x_t] + b_z), \tag{1}$$

$$r_t = \sigma(W_r \cdot [h_{t-1'} x_t] + b_r), \tag{2}$$

$$h'_t = tanh(W \cdot [r_t * h_{t-1}, x_t] + b_h), \tag{3}$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * h'_t, \tag{4}$$

$$y_t = \sigma(W_0 \cdot h_t), \tag{5}$$

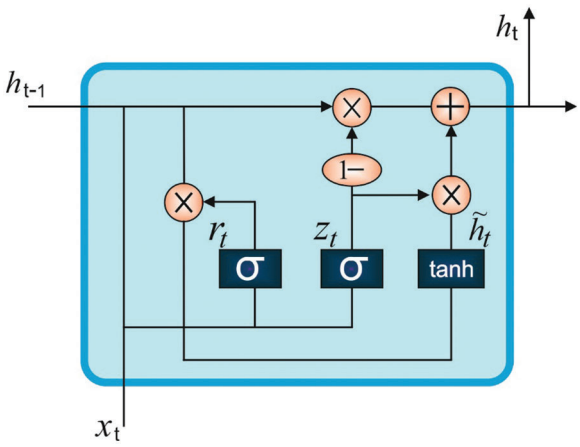

Figure 2: Structure of the GRU model

In this expression, Z_t and r_t signify the reset gate and the upgrade gate respectively, W_z, W_r, W, and W symbolize the weight variables of the input dataset, h_{t-1} specifies the output of the preceding layer and x_t represents the input of the existing layer. b_z, b_r, and b_h correspond to biases, σ indicates the sigmoid function and *tanh* is exploited to change the value flow across the network. The output value, next to σ and *tanh* functions, can be controlled between $(0,\ 1)$ and $(-1, 1)$. Then, the final output is attained whereas the loss values are evaluated based on the loss function given below.

$$E_t = \frac{1}{2}(y_d - y_t^0)^2, \tag{6}$$

$$E = \sum_{t=1}^{T} E_{t'} \tag{7}$$

Now, E_t embodies the loss of the instance at t time, y_d specifies the real label data, y_t^0 denotes the output value of the preliminary iteration and E indicates the loss of the instance at time t.

The Backpropagation (BP) methodology is exploited to learn the network. Therefore, the partial derivatives of the loss function must be evaluated for the variable. After the evaluation of the partial derivatives, the loss convergence is iteratively defined and the variable is upgraded.

2.2 AOA Based Hyperparameter Optimization

In order to improve the TSS performance of the GRU model, the AOA technique is used as a hyperparameter optimizer [21]. Like other birds, Aquila has a dark brown colour complexion. The back of its head has an additional golden brown colour. These birds have agility and speed. Furthermore, it has sharp claws and strong legs that assist in capturing the target. Aquila is famous for attacking the adult deer. It constructs large nests in mountains or in high altitudes. Aquila is a skilled hunter and its intelligence is equal to that of the human beings' intelligence. Similar to the population-based algorithms, the AOA methodology starts with a population of the candidate solutions. The technique stochastically initiates with an upper limit and a lower limit [21]. Every iteration almost defines the optimal solution as given below.

$$\mathrm{X} = \begin{bmatrix} X_{1,1} & \ldots & \ldots & X_{1,n} \\ X_{2,1} & \ldots & \ldots & X_{2,\mathrm{n}} \\ \vdots & \vdots & \vdots & \vdots \\ X_{\mathrm{m},1} & X_{\mathrm{m,n}} & \ldots & X_{\mathrm{m,n}} \end{bmatrix} \tag{8}$$

$$X_{ij} = rand \times \left(UB_j - LB_j\right) + LB_j,\ i = 1, 2 \ldots\ldots,\ mj = 1, 2, \ldots\ldots\ n \tag{9}$$

In Eq. (7), m denotes the overall number of the candidate solutions available and n represents the size of the dimension. *rand* corresponds to an arbitrary value and the *j-th* lower limit is denoted by LB_j. UBj denotes the *j-th* upper limit. In general, the AO process simulates the Aquila behaviour in the hunting procedure as shown below.

- Step 1: Increased exploration ()

In step 1, the Aquila explores from the sky to define a searching region and identifies the location of the prey. Then, it recognizes the areas of the prey and selects the finest area for hunting.

$$X_1(t+1) = X_{best}(t)\ \left(1 - \frac{t}{T}\right) + (X_M(t) - X_{best}(t) \times rand) \tag{10}$$

$$X_M(t) = \frac{1}{N}\sum_{i=1}^{N} X_i(t), N = 1, 2, \ldots \ dim \tag{11}$$

Here, the solution for iteration t is denoted by $X_1(t+1)$. It produces a preliminary searching technique X_1, whereas $X_{best}(t)$ denotes the optimally-attained solution during the t^{th} iteration. This value defines the assessed point of the target. The variable that manages the augmented exploration through the iteration count is denoted by $\left(1 - \frac{t}{T}\right)$. The mean point value of the existing solution, connected during t iteration, is denoted by X_M, in which *rand* corresponds to an arbitrary number. The population size is N. The dimension size is *dim*.

- Step 2: Limited exploration ()

Next, the target is established at a high altitude. Here, the Aquila encircles in the cloud, reaches the location and gets ready to attack the target. This process is arithmetically expressed through the following equations.

$$X_2(t+1) = X_{best}(t) \times Levy(D) + X_R(t) + (y - x) \times rand \tag{12}$$

$$\text{Levy}\ (D)\ = s \times \frac{u \times \sigma}{|v|}\frac{1}{\beta} \tag{13}$$

$$\sigma = \left(\frac{\Gamma(1+\beta) \times sine\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}} \right) \tag{14}$$

$$y = r \times \cos\ (\theta) \tag{15}$$

$$x = r \times \sin\ (\theta) \tag{16}$$

$$r = r_1 + U + D_1 \tag{17}$$

$$\theta = -\omega \times D_1 + \theta_1 \tag{18}$$

$$\theta_1 = \frac{3 \times \pi}{2} \tag{19}$$

In these expressions, the conclusion of iteration t, created by the next technique phase, is denoted by $X_2(t+1)$. The allocation function of the Levy Flight is denoted by Levy (D). The dimensional space is represented by $X_R(t)$, which is an arbitrary solution in the range of 1 to N. s represents a constant value in the range of 0.01. u and v indicate the arbitrary values between 0 and 1. σ denotes a constant value in the range of 1.5. χ and y are applied to define the spiral shapes. r_1 is a value that is chosen between 1 and 20 and is applied to fix the search cycle count. U denotes a parameter that is multiplied by 0.00565. D_1 indicates an integer from 1 to the maximum searching space parameter (dim) value. ω denotes that the parameter has constant smaller values multiplied by 0.005.

- Step 3: Increased exploitation ()

In this step, the Aquila is present at the location of exploitation, viz., gets closer to the prey and makes a pre-emptive attack as shown below.

$$X_3(t+1) = (X_{best}(t) - X_R(t)) \times \alpha - rand + ((UB - LB) \times rand + LB \times \delta) \quad (20)$$

In this exploitation tuning parameter set, the small values $(0, 1)$ are denoted by α and δ. UB and LB specify the upper and lower limits, respectively.

- Step 4: Limited exploitation ()

Now, the Aquila approaches the nearby prey. Then, the Aquila attacks the target on the ground i.e., its final position, by walking on the ground to catch the prey. The behaviour of the Aquila is modelled as given herewith.

$$X_4(t+1) = Q_f \times X_{best}(t) - (G_1 \times X(t) \times rand) - G_2 \times Levy(D) + rand \times G_1 \quad (21)$$

$$Q_f = r\frac{2 \times r(rand - 1)}{(1 - T)^2} \quad (22)$$

$$G_1 = 2 \times rand - 1 \quad (23)$$

$$G_2 = 2 \times \left(1 - \frac{t}{T}\right) \quad (24)$$

In Eq. (24), the iteration solution that is produced by the last searching technique (X_4) is $X_4(t+1)$. Qf denotes the quality functions that are utilized to balance the searching technique. Each type of movement applied for tracking the prey is denoted by G_1. G_2, which reduces from [2, 0]. It shows the flight inclination of the Aquila employed to follow the target from its primary to the final spot. The present solution at *the* $t-th$ iteration is denoted by (T). An arbitrary point in the range of [0, 1] is denoted by *rand*. The present iteration is denoted by *t*. The maximal iteration count is denoted by *T*.

2.3 Buckwalter Code Based Transcription Model

At last, the Buckwalter code is utilized to capture, store and display the Arabic text. It is predominantly applied in the Arabic transcription process to capture, store and display Arabic text. Being a stringent transcription method, it adheres to the spelling agreements of the language. Further, it substitutes one-to-one mapping and is completely reversible to contain every data that is required for good pronunciation.

3 Experimental Validation

The current section presents the overall analytical results of the proposed ODLD-ATSS model under distinct aspects. The parameter settings are given herewith; learning rate: 0.01, dropout: 0.5, batch size: 5, epoch count: 50 and activation: ReLU. The proposed model was simulated in Python.

Table 1 and Fig. 3 provide the detailed $accu_y$ examination results achieved by the proposed ODLD-ATSS model under validation and testing datasets of gemination. The experimental values imply that the proposed ODLD-ATSS model achieved enhanced $accu_y$ values. For instance, on the validation set, the proposed ODLD-ATSS model obtained a maximum $accu_y$ of 99.61%, whereas the All-Dense, All-LSTM, All-bidirectional LSTM (BLSTM) and the hybrid models attained the least $accu_y$ values such as 98.60%, 98.70%, 99.34% and 99.04% respectively. On the contrary, on the test dataset, the proposed ODLD-ATSS approach gained a maximum $accu_y$ of 99.76%, whereas the All-Dense, All-LSTM, All-BLSTM and the hybrid approaches acquired the least $accu_y$ values such as 98.61%, 98.71%, 98.81% and 99.00% correspondingly.

Table 1: $Accu_y$ analytical results of the ODLD-ATSS model on the selected models of gemination

Accuracy (%)		
Models	Validation set	Test set
ODLD-ATSS	99.61	99.76
All-Dense	98.60	98.61
All-LSTM	98.70	98.71
All-BLSTM	99.34	98.81
Hybrid model	99.04	99.00

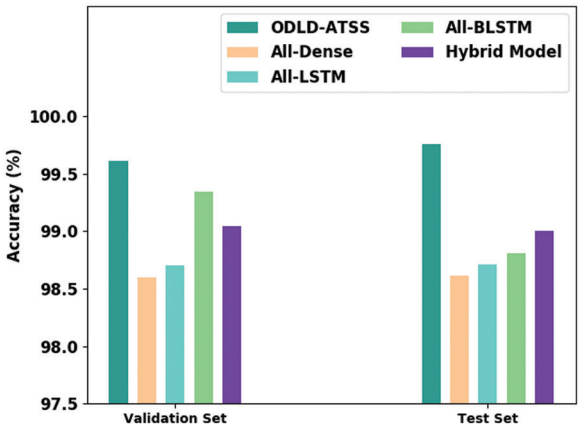

Figure 3: Comparative $Accu_y$ analytical results of the ODLD-ATSS model on the selected models of gemination

Table 2 provides the overall results achieved by the proposed ODLD-ATSS model on germination signs with the DNN model. The results imply that the proposed ODLD-ATSS model produced enhanced outcomes for both non-geminated and geminated classes. For instance, the presented ODLD-ATSS model categorized the non-geminated classes with a $prec_n$ of 99.71%, $reca_l$ of 99.71%, and an $F1_{score}$ of 99.76%. At the same time, the proposed ODLD-ATSS model categorized the germination classes with a $prec_n$ of 99.70%, $reca_l$ of 99.67% and an $F1_{score}$ of 99.74%.

Table 2: Overall gemination sign classification results of the proposed ODLD-ATSS model

Gemination classes	Precision	Recall	F1 score
DNN model			
Non geminated	98.39	99.22	98.48
Geminated	97.99	89.33	94.01
ODLD-ATSS			
Non geminated	99.71	99.71	99.76
Geminated	99.70	99.67	99.74

Table 3 reports the overall gemination prediction outcomes accomplished by the proposed ODLD-ATSS model. The results imply that the proposed ODLD-ATSS model attained the effectual prediction outcomes for gemination on the validation dataset and the testing dataset. For instance, on the validation dataset, the proposed ODLD-ATSS model offered an increased $accu_y$ of 98.58%, whereas the All-Dense, All-LSTM, All-BLSTM and the hybrid models produced the least $accu_y$ values such as 81.14%, 86.14%, 91.08% and 91% correspondingly. Moreover, on test set, the proposed ODLD-ATSS algorithm accomplished an increased $accu_y$ of 98.54%, whereas the All-Dense, All-LSTM, All-BLSTM and the hybrid methods achieved the least $accu_y$ values such as 76.00%, 80.54%, 83.98% and 84.02% respectively.

Table 3: Predicted gemination results of the proposed ODLD-ATSS model

Accuracy (%)		
Models	Actual gemination	Predicted gemination
Validation Set		
ODLD-ATSS	98.79	98.58
All-Dense	83.12	81.14
All-LSTM	87.72	86.14
All-BLSTM	91.32	91.08
Hybrid Model	92.10	91.00
Test Set		
ODLD-ATSS	98.66	98.54
All-Dense	77.73	76.00
All-LSTM	81.95	80.54
All-BLSTM	84.75	83.98
Hybrid Model	86.44	84.02

Table 4 and Fig. 4 provide the detailed final diacritization $accu_y$ analysis results achieved by the proposed ODLD-ATSS model and other existing models under different kinds of phonemes. The results infer that the proposed ODLD-ATSS model achieved enhanced $accu_y$ values under each phoneme. For instance, in terms of plosives, the proposed ODLD-ATSS model obtained a maximum $accu_y$ of 99.42%, whereas the All-Dense, All-LSTM, All-BLSTM, hybrid 1, and hybrid 2 models produced the least $accu_y$ values such as 97.97%, 99%, 98.94%, 98.71% and 98.72% respectively. In terms of Fricatives, the proposed ODLD-ATSS method attained a maximum $accu_y$ of 99.61%, whereas the All-Dense, All-LSTM, All-BLSTM, hybrid 1 and hybrid 2 models accomplished the least $accu_y$ values, such as 99.06%, 98.73%, 99.01%, 98.61% and 98.75% correspondingly. In addition, in terms of Affricates, the presented ODLD-ATSS model reached an optimal $accu_y$ of 100%, whereas the All-Dense, All-LSTM, All-BLSTM, hybrid 1, and hybrid 2 models reported the least $accu_y$ values such as 99.83%, 99.70%, 99.79%, 99.74% and 99.69% correspondingly.

Table 5 provides the overall classification results accomplished by the proposed ODLD-ATSS model under distinct diacritic classes. Fig. 5 shows the brief results of the proposed ODLD-ATSS model under distinct diacritic sign classes in terms of $prec_n$. The figure infers that the proposed ODLD-ATSS model achieved an enhanced performance under each class. The proposed ODLD-ATSS model obtained the following $prec_n$ values such as 97.56% under fatha class, 96.39% under dhamma class, 97.09% under

Kasra class, 96.41% under Fathaten class, 96.58% under Dhammaten class, 97.77% under Kasraten class, 97.57% under sukun class and 97.13% under others' class respectively.

Table 4: Overall diacritization accuracy results of the proposed ODLD-ATSS model under diverse phonemes

Methods	Plosives	Fricatives	Affricates	Nasals	Trills	Laterals	Semi-vowels	Average
ODLD-ATSS	99.42	99.61	100	98.91	100	99.6	99.15	**99.53**
All-Dense	97.97	99.06	99.83	97.89	97.81	99.05	94.67	**98.04**
All-LSTM	99.00	98.73	99.70	97.90	100.00	98.71	95.60	**98.52**
All-BLSTM	98.94	99.01	99.79	96.79	100.00	98.94	96.63	**98.59**
Hybrid 1	98.71	98.61	99.74	97.70	99.70	98.85	95.80	**98.44**
Hybrid 2	98.72	98.75	99.69	98.91	98.85	98.67	96.86	**98.64**

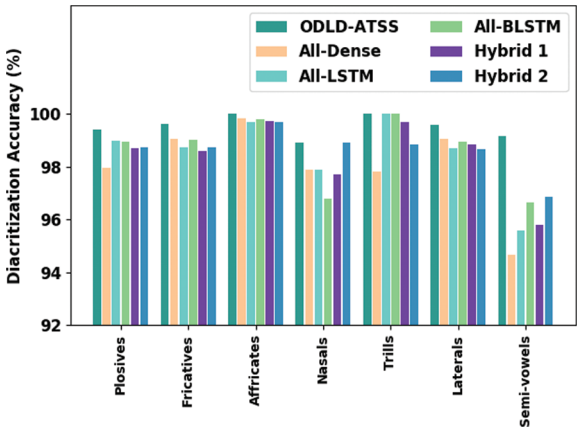

Figure 4: Comparative diacritization accuracy results of the proposed ODLD-ATSS model under diverse phonemes

Table 5: Comparative diacritic sign class outcomes of the proposed ODLD-ATSS model

Diacritic sign class	Precision	Recall	F1 score
Fatha	97.56	96.22	96.37
Dhamma	96.39	97.32	96.85
Kasra	97.09	97.03	96.55
Fathaten	96.41	96.78	96.17
Dhammaten	96.58	96.2	96.28
Kasraten	97.77	97.05	97.65
Sukun	97.57	97.35	97.44
Others	97.13	96.41	96.71
Average	**97.06**	**96.80**	**96.75**

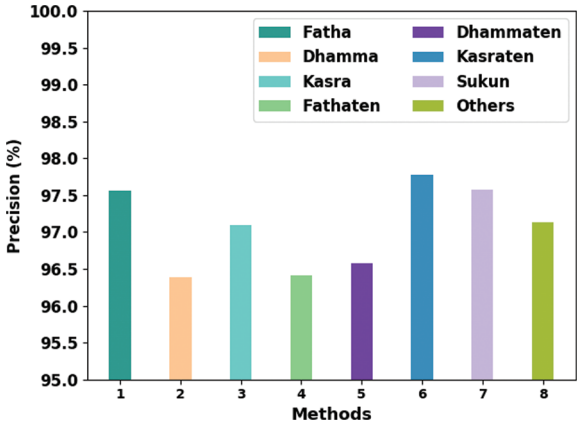

Figure 5: Comparative $prec_n$ analysis results under different diacritic sign classes

Fig. 6 portrays the comprehensive analytical results attained by the proposed ODLD-ATSS algorithm under different diacritic sign classes in terms of $reca_l$. The figure denotes that the proposed ODLD-ATSS model accomplished an improved performance under all the classes. The proposed ODLD-ATSS model gained the following $reca_l$ values such as 96.22% under fatha class, 97.32% under dhamma class, 97.03% under Kasra class, 96.78% under Fathaten class, 96.2% under Dhammaten class, 97.05% under Kasraten class, 97.35% under sukun class and 96.41% under others' class correspondingly.

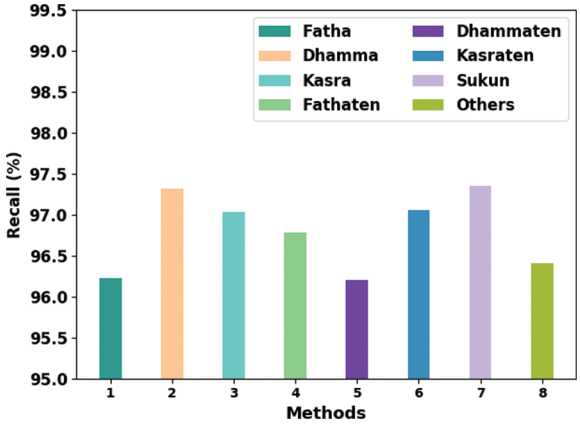

Figure 6: Comparative $reca_l$ analysis results under different diacritic sign classes

Fig. 7 provides the detailed illustration of the results achieved by the proposed ODLD-ATSS model under different diacritic sign classes in terms of $F1_{score}$. The figure infers that the proposed ODLD-ATSS model achieved an enhanced performance under each class. The proposed ODLD-ATSS model acquired the following $F1_{score}$ values such as 96.37% under the fatha class, 96.85% under the dhamma class, 96.55% under the Kasra class, 96.17% under Fathaten class, 96.28% under Dhammaten class, 97.65% under Kasraten class, 97.44% under sukun class and 96.71% under others' class respectively.

Table 6 and Fig. 8 exhibit the comparative $accu_y$ analysis results achieved by the proposed ODLD-ATSS model and other diacritization models. The experimental values imply that the rule-based model and the DNN model produced the least $accu_y$ values, such as 77.36% and 79.36%, respectively. At the same time, the SVM and the LSTM models reported slightly enhanced $accu_y$ values, such as 81.53% and 82.16% respectively. Likewise, the BLSTM model accomplished a reasonable $accu_y$ of 85.35%.

However, the proposed ODLD-ATSS model achieved a maximum $accu_y$ of 96.35%. Therefore, the proposed ODLD-ATSS model can be considered as an effectual tool for activity recognition.

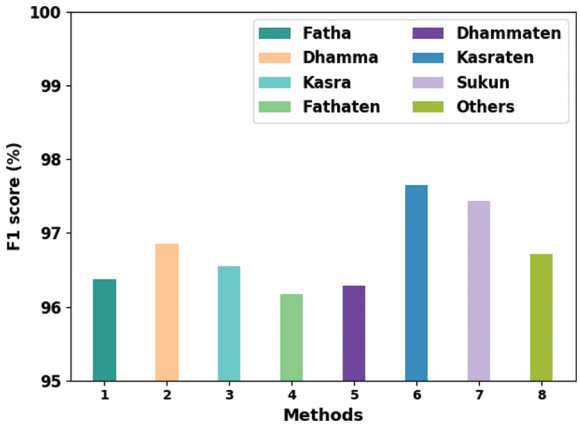

Figure 7: Comparative $F1_{score}$ analysis results under different diacritic sign classes

Table 6: Overall accuracy values of the proposed ODLD-ATSS model and other existing models

Methods	Accuracy score (%)
ODLD-ATSS	96.35
SVM	81.53
Rule-Based	77.36
DNN	79.36
LSTM	82.16
BLSTM	85.35

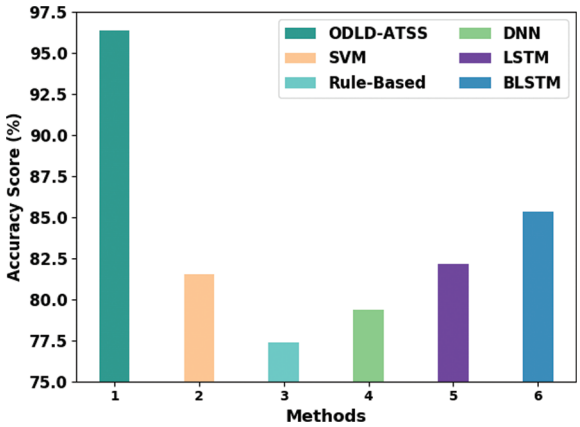

Figure 8: Comparative $accu_y$ analysis results of the proposed ODLD-ATSS model

4 Conclusion

In the current study, the proposed ODLD-ATSS technique has been developed for the TSS process to be used by the visually-challenged people in the Kingdom of Saudi Arabia. The prime objective of the presented ODLD-ATSS model is to convert the text into speech signals for the visually-challenged people. To attain this, the presented ODLD-ATSS model initially designs a GRU-based prediction model for diacritic and gemination signs. Besides, the Buckwalter code is also utilized to capture, store and display the Arabic text. In order to enhance the TSS performance of the GRU method, the AOA technique is used. To establish the enhanced performance of the proposed ODLD-ATSS algorithm, various experimental analyses were conducted. The experimental outcomes confirmed the improved performance of the ODLD-ATSS model over other DL-based TSS models. The proposed ODLD-ATSS model achieved a maximum accuracy of 96.35%. In the future, the performance of the proposed ODLD-ATSS model can be enhanced using hybrid metaheuristic algorithms. Besides, the presented model can also be extended for the object detection process in real-time navigation techniques.

Funding Statement: The authors extend their appreciation to the King Salman center for Disability Research for funding this work through Research Group no KSRG-2022-030.

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

References

[1] R. Olwan, "The ratification and implementation of the Marrakesh Treaty for visually impaired persons in the Arab Gulf States," *The Journal of World Intellectual Property*, vol. 20, no. 5–6, pp. 178–205, 2017.

[2] R. Sarkar and S. Das, "Analysis of different braille devices for implementing a cost-effective and portable braille system for the visually impaired people," *International Journal of Computer Applications*, vol. 60, no. 9, pp. 1–5, 2012.

[3] M. Awad, J. El Haddad, E. Khneisser, T. Mahmoud, E. Yaacoub *et al.,* "Intelligent eye: A mobile application for assisting blind people," in *IEEE Middle East and North Africa Communications Conf. (MENACOMM)*, Jounieh, pp. 1–6, 2018.

[4] F. N. Al-Wesabi, "A hybrid intelligent approach for content authentication and tampering detection of Arabic text transmitted via internet," *Computers, Materials & Continua*, vol. 66, no. 1, pp. 195–211, 2021.

[5] N. B. Ibrahim, M. M. Selim and H. H. Zayed, "An automatic arabic sign language recognition system (ArSLRS)," *Journal of King Saud University-Computer and Information Sciences*, vol. 30, no. 4, pp. 470–477, 2018.

[6] M. Al Duhayyim, H. M. Alshahrani, F. N. Al-Wesabi, M. A. Al-Hagery, A. M. Hilal *et al.,* "Intelligent machine learning based EEG signal classification model," *Computers, Materials & Continua*, vol. 71, no. 1, pp. 1821–1835, 2022.

[7] F. N. Al-Wesabi, "Proposing high-smart approach for content authentication and tampering detection of Arabic text transmitted via internet," *IEICE Transactions on Information and Systems*, vol. E103.D, no. 10, pp. 2104–2112, 2020.

[8] A. L. Valvo, D. Croce, D. Garlisi, F. Giuliano, L. Giarré *et al.,* "A navigation and augmented reality system for visually impaired people," *Sensors*, vol. 21, no. 9, pp. 3061, 2021.

[9] H. Luqman and S. A. Mahmoud, "Automatic translation of Arabic text-to-Arabic sign language," *Universal Access in the Information Society*, vol. 18, no. 4, pp. 939–951, 2019.

[10] R. Jafri and M. Khan, "User-centered design of a depth data based obstacle detection and avoidance system for the visually impaired," *Human-Centric Computing and Information Sciences*, vol. 8, no. 1, pp. 1–30, 2018.

[11] Y. S. Su, C. H. Chou, Y. L. Chu and Z. Y. Yang, "A finger-worn device for exploring chinese printed text with using cnn algorithm on a micro IoT processor," *IEEE Access*, vol. 7, pp. 116529–116541, 2019.

[12] M. Brour and A. Benabbou, "ATLASLang MTS 1: Arabic text language into arabic sign language machine translation system," *Procedia Computer Science*, vol. 148, no. 6, pp. 236–245, 2019.

[13] T. Kanan, O. Sadaqa, A. Aldajeh, H. Alshwabka, S. AlZu'bi *et al.,* "A review of natural language processing and machine learning tools used to analyze arabic social media," in *IEEE Jordan Int. Joint Conf. on Electrical Engineering and Information Technology (JEEIT)*, Amman, Jordan, pp. 622–628, 2019.

[14] S. Aly and W. Aly, "DeepArSLR: A novel signer-independent deep learning framework for isolated Arabic sign language gestures recognition," *IEEE Access*, vol. 8, pp. 83199–83212, 2020.

[15] M. Araya and M. Alehegn, "Text to speech synthesizer for tigrigna linguistic using concatenative based approach with LSTM model," *Indian Journal of Science and Technology*, vol. 15, no. 1, pp. 19–27, 2022.

[16] O. A. Berkhemer, P. S. Fransen, D. Beumer, L. A. V. D. Berg, H. F. Lingsma *et al.,* "A randomized trial of intraarterial treatment for acute ischemic stroke," *The New England Journal of Medicine*, vol. 372, no. 1, pp. 11–20, 2015.

[17] M. Eltay, A. Zidouri and I. Ahmad, "Exploring deep learning approaches to recognize handwritten arabic texts," *IEEE Access*, vol. 8, pp. 89882–89889, 2020.

[18] R. Alzohairi, R. Alghonaim, W. Alshehri and S. Aloqeely, "Image based arabic sign language recognition system," *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 3, pp. 185–194, 2018.

[19] J. X. Chen, D. M. Jiang and Y. N. Zhang, "A hierarchical bidirectional GRU model with attention for EEG-based emotion classification," *IEEE Access*, vol. 7, pp. 118530–118540, 2019.

[20] H. M. Lynn, S. B. Pan and P. Kim, "A deep bidirectional GRU network model for biometric electrocardiogram classification based on recurrent neural networks," *IEEE Access*, vol. 7, pp. 145395–145405, 2019.

[21] I. H. Ali, Z. Mnasri and Z. Lachiri, "DNN-based grapheme-to-phoneme conversion for Arabic text-to-speech synthesis," *International Journal of Speech Technology*, vol. 23, no. 3, pp. 569–584, 2020.