International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Optimizing Speech Synthesis for Efficient Text-to-Speech Conversion with Enhanced Robustness and Resource Efficiency

Mukta Sandhu [1*]

Submitted: 28/12/2023 **Revised**: 07/02/2024 **Accepted**: 13/02/2024

Abstract: Speech synthesis is conversion of text to speech but it become very challenging when there is background noise. Additionally, it is very time-consuming, has high cost, and more power-consuming. To overcome these issues design a Butterfly-based Convolutional Neural System (BbCNS). Initially, the input text of certain users was collected and trained into the system and the preprocessing is utilized for removing the errors present in the dataset and preparing the text data for a specific context. Additionally, data normalization is employed to transfer the text into canonical and consistent form. Additionally, linguistic analysis is used to understand the content of the text and to identify the constituent morphemes of each word. Furthermore, Prosodic prominence prediction can be predicted from written language. Finally, the waveform is generated for converting text into speech. At last, the outcomes that are gained from the model that is designed are validated using other prevailing models with respect to accuracy, sensitivity, specificity, precision, and computation time.

Keywords: *Text-To-Speech Conversion, Linguistic Analysis, Prosody Prediction, Butterfly Optimization, Convolutional Neural Network, Waveform Generation.*

1. Introduction

Speech and text are the chief modes of human communication. An individual requires visual vision to access text information [1]. Moreover, speech becomes the primary communication technique in Human Intelligent Systems (HIS) [2]. Furthermore, Natural Language Processing (NLP) shows an important role in numerous fields that compact by the computation of linguistic natures [3-5]. Additionally, TTS is used for the conversion of text into audio and the TTS technology allows PC for speaking to a user also providing computed information [6]. As well TTS scheme collects inputs of texts, and the texts are pre-processed using computer algorithm. Finally, synthesizes the speech using mathematical models [7-9]. As an output, the TTS system typically produces sound data by generating waveforms in an audio layout [10]. It begins by analyzing the input text for extracting linguistic information from the input text [11]. The linguistic information is helpful for creating correct prosodic information. Meanwhile, it searches an acoustic inventory for the corresponding primitive waveform template string [12-14]. The basic process of TTS is shown in fig.1.

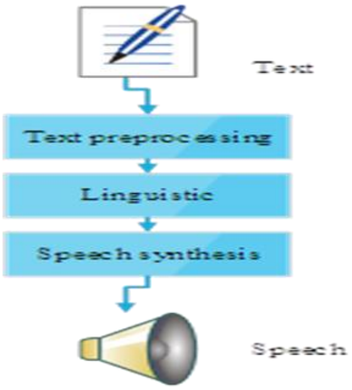

Fig.1 Basic text to speech conversion

Finally, it matches the prosody of the waveform template string to the prosodic information and generates the output synthetic speech [15]. Prosodic information generation is the most important part of determining the naturalness of synthetic speech in a TTS system [16]. Many methods for generating prosodic information have been proposed in the past [17]. The basic values of prosodic parameters are first assigned using a rule-based approach, and then some rules are applied to modify these values by the linguistic data gleaned from the input text [18]. The method's drawbacks include the difficulty in inferring adequate rules and the lengthy development period, Noisy environments, accents and multiple speakers may degrade results, like less accuracy and wrong interpretation, productivity, and time costs, interference noise in background , physical side effects, and large time consumption [19]. Finally, speech synthesis gains more processing power and low accuracy [20]. Several deep learning methods are proposed to convert text into speech. This prevailing methods haven't reach

[1] *Skill Department of Computer Science and Engineering, Shri Vishwarkarma Skill University, Palwal, India*
* *Corresponding Author Email: mukta.sandhu@gmail.com*

adequate accuracy and increase the processing power while converting text into speech, which motivated us to do this work

The key contribution of the developed model is detailed below,

- Initially, text was collected and trained to the system which is implemented using the python tool

- Moreover, design a Butterfly based Convolutional neural System (BbCNS) for improving the speech of the designed model

- Hereafter, Pre-processing is utilized for removing the errors present in the dataset and preparing the text data for a specific context.

- Furthermore, data normalization is employed to transfer the text into canonical and consistent form.

- Additionally, linguistic analysis is used to understand the content of the text and identify the constituent morphemes (and morphs) of each word.

- Furthermore, Prosodic prominence prediction can be predicted from written language.

- Finally, the sound waveform is generated using WaveNet for converting text into speech, and the attained outcomes are validated with prevailing models.

This paper is organized as - section two portrays the literature survey of TTS system, section three demonstrates the proposed methodology, section four discus the result and developed model, and section five describes the conclusion.

2. Literature survey

The various studies associated to converting text into speech with the deep learning methods are discussed below -

Chen, S.H., [21] have presented corpus based prosodic modeling scheme for TTS conversion. The output dependence of the prosodic information and the input linguistic information is modelled in this work using a four-layer RNN. Two synthesis model variations were discussed in this RNN. The first one is utilized to create an additional fuzzy-neural network that helps generate RNN prosody by inferring some fuzzy rules of attachment from high-level linguistic data. To lessen the burden on the RNN, the second one is utilized for supplementary statistical models of the prosodic parameter to eliminate some influencing aspects of linguistic features. It provides higher accuracy with lower sensitivity.

Rajbongshi, A., et.al, [22] have presented a Bangla character recognition and TTS conversion. In this work, camera-based assistive technology is used, people who are blind or illiterate can interpret papers with Bangla text through listening audio descriptions of the image subjects. It can assist Bangla speakers who are illiterate or have suffered a severe loss of vision. It provides higher specificity with lower precision.

Santra, S., et.al, [23] have presented a development of GUI for TTS synthesis by NLP method. In this work, used NLP for analysing and processing the text, a TTS synthesizer is created to turns the written word into spoken word. This technology uses Digital Signal Processing (DSP) for turning the managed text into a speech synthesis. At this time, a simple program that reads out user-inputted text as synthesized speech and can be stored as an mp3 file has been developed as a handy text-to-speech synthesizer. It provides a higher F-score with lower accuracy.

Nagdewani, S et.al, [24] have presented a dissimilar organization for TTS conversion that used in voice-based email scheme. The goal is to examine and contrast the various STT and TTS conversion techniques to identify the most effective approach that can be applied to both conversion processes. Consequently, a review study determined that the HMM statistical model is the best appropriate for both STT and TTS conversions. Finally, a model using ANN and HMM for TTS conversions and STT conversions was proposed. It provides lower computational time with a higher error rate.

Talman, A., et.al, [25] have presented a forecasting Prosodic Prominence (PP) model from text with word representations. Moreover, several distinct replicas, extending from feature-based classifiers to Neural Network (NN) scheme, are trained to predict the discretized PP. Additionally, benchmark dataset are described and executed for proving the efficiency. Through less precision, it offers a higher f-score. It provides a higher f-score with lower precision.

3. Proposed Methodology

To enhance the performance of TTS design and Butterfly based Convolutional neural System (BbCNS). It improves the performance of the TTS system by attaining better results. First, pre-processing and text normalization are done to get the text ready for audio conversion. The linguistic analysis and prosodic prediction are done sequentially to produce the waveform of the text message. The main aim to develop the system is it converts the text of the user to voice output. Thus the designed model analyses the input data and converts the text into audio using butterfly optimization and Convolutional Neural Network (CNN). Initially, the input text of certain users was collected and trained into the system and the preprocessing is utilized for removing the errors present in the dataset and preparing the text data for a specific context. The design of the developed technique is exposed in fig.2.

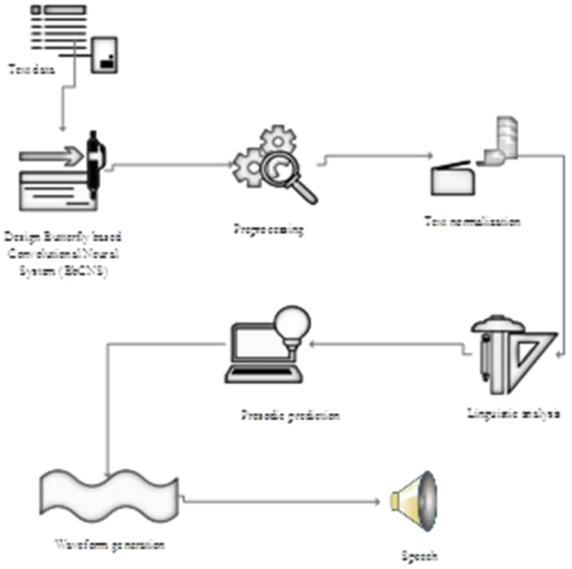

Fig.2 Proposed methodology

Additionally, data normalization is employed to transfer the text into canonical and consistent form. Additionally, linguistic analysis is used to understand the content of the text and to identify the constituent morphemes of each word. Furthermore, Prosodic prominence prediction can be predicted from written language. Finally, the sound waveform is generated for converting text into speech.

3.1. Process of Butterfly based Convolutional Neural Scheme

The main aim of the designed technique is enhance the performance of the TTS scheme by attaining better accuracy and less execution time. Generally, BbCNS is developed to convert the text into speech using input texts. Moreover, the fitness function of the Butterfly Optimization Algorithm (BOA) is updated to the developed CNN and it contains five layers. An unused metaheuristic algorithm inspired from nature and is built on how butterflies find food is called the BOA. In terms of biology, butterflies employ sense receptors to locate the origin of food. The utilization of these sensory receptors to detect aroma or scent, and these are dispersed throughout bodily parts of a butterfly, such as its legs, palps, and antennae. These sensors on the butterfly's skin are nerve cells Chemoreceptors are located on the body surface. BOA presumes that a butterfly can produce aroma with a certain level of intensity that is associated with butterfly fitness, and the fitness will change depending on the location. The purpose of using BOA fitness in the classification layer is it can convert the text into speech with better results. Initially, the text dataset is tested and trained into the system and the collected dataset is imported to the input layer. The preprocessing and text normalization is processed in the convolutional layer.

3.1.1. Preprocessing

In this stage, the errors, noises, and irregularities present in the dataset were removed and enhanced the data quality by removing noisy data. Preprocessing is utilized for removing the errors present in the dataset and preparing the text data for a specific context. Moreover, preprocessing minimize the noise because of various factors like error, and low quality. Furthermore, preprocessing cleans the lower casing, removes punctuations, removes frequent words, removes stop words, and stemming, removes rare words, and removes emojis. Additionally, preprocessing is processed using Eqn. (1),

$$p(r) = \sqrt{\frac{1}{M-1}\sum_{i=1}^{M}(g_i-\overline{g})}^{2} \qquad (1)$$

Let, g_i is denoted as the individual text of the dataset, and $\overline{g}$ is represented as a total number of texts present in the dataset. Moreover, $M$is represents as the mean and i is denoted as the constant.

3.1.2. Text normalization

The input texts are split into dissimilar corresponding text frames also standardizing the text. Moreover, text normalization converts abbreviations, numeric expressions, and acronyms into regular word forms. It is the process of transforming a single canonical form before processing or storing it. It minimizes the randomness by bringing it closer to the standard text. Moreover, minimize the number of different information and enhance efficiency. The process of text normalization is obtained using Eqn. (2),

$$T_n = \frac{j-min(j)}{max(j)-min(j)} \qquad (2)$$

Where, T_n is represented as normalized value, $j$is denoted as original value, $max(j)$ is considered as the extreme value of$j$, and $min(j)$ is considered as the least value of j.

3.1.3. Linguistic analysis

The linguistic analysis is used to understand the content of the text and to identify the constituent morphemes (and morphs) of each word. It was essential to first break down the text through a fundamental basic linguistic structure in which both voice and text surface realizations share. The speech synthesis method can then be driven by this structure to generate the required output acoustic sound. The voice signal is subject to numerous limiting relationships provided by the linguistic structure. Identifying each word's morphemes (and morphs) is the goal of word-level linguistic analysis. These units have a variety of other valuable qualities. The atomic, minimum syntactic building blocks of a language are known as morphemes, and they are extremely stable in the sense that neither new morphemes nor morpheme deletions occur frequently. These morphemes have a significant capacity for word formation, making it possible for a given morph lexicon to quickly encompass at least an order of magnitude more words. Furthermore, many linguistic events only occur inside the confines of morph borders, and lexical morphemes easily handle regularly inflected words and regularly compound terms.

3.1.4. Prosodic prediction

It employs individual or additional test patterns that correlated for same class speech sounds. Moreover, create pattern depiction of the features that are resulting from the text class. The input utterance is first spitted into syllable sections interspersed with pauses through forced aligner in the encoder through associated text's linguistic information each syllable has its prosodic-acoustic characteristics. The segment is then extracted. Then, a parametric description of a syllable's prosodic-acoustic features is generated. An analysis operation of prosody is used to estimate the segment according to the sensing behavior of butterfly optimization. Finally, some low-level linguistic features are encoded and sent to the decoder. The features of every prosodic-acoustic syllable segment are first reconstructed in the decoder by a prosody synthesis process that feeds the decoded low-level linguistic features. Finally, using the prosodic prediction, a CNN-based speech synthesizer generates the output speech. Additionally, the prosodic prediction is formulated using Eqn. (3),

$$\begin{aligned} P(S,H/D) &= P(H/S,D)P(S/D) \\ &= P(A,B,C/K,N,D)P(K,N/D) \\ &\approx P(A/K,N,D)P(B,C/K,D)B(t)P(N/K)P(K/D) \end{aligned} \quad (3)$$

Where, $P(A/K,N,D)$ is denoted as the syllable prosodic acoustic of the designed model that represents duration, contour, prosodic tags, and energy level of text. $P(B,C/K,D)$ is denoted as syllable junctions that describe the characteristics of inter-syllable acoustic and linguistic features. Furthermore, $P(N/K)$ is considered as a variation of prosodic state and neighbouring break type and $B(t)$ is represented as butterfly fitness. Additionally, $P(K/D)$ is represented as the dependence of break occurrences of the linguistic features.

3.1.5. Voice Synthesis

WaveNet is used to generate vocoder waveforms with 24 kHz-16 bit for every speaker. First, analyze the text, and then produce the speech waveform. This prosodic information is then converted into a waveform. Meanwhile, it searches an acoustic inventory for the appropriate primitive waveform template string. Finally, it matches the prosody of the waveform template string to the prosodic information and generates the output synthetic speech. A traditional TTS technique is fundamentally converts linguistic features that are extracted from the text toward acoustic features that helpful for speech synthesis waveforms by a sampling level of 16k Hz and 10-bit act quantization. Finally, the speech is produced as output and joined together to create the voice output.

4. Results And Discussion

Here design an optimization-based convolutional neural system for converting text into speech is proposed. 3.6 version of Python along with deep learning tool Keras, is used in the implementation of the proposed scheme. Here, various metrices of performance are analysed - computational time, specificity, sensitivity, f-score, accuracy, and precision The proposed approach is compared with other different prevailing methods, like RNN-PSM-TTS [21], NLP-DSP-TTS [22], and HMM-ANN-TTS [24] respectively.

4.1. Performance Metrics

For evaluation of the performance of system and conduction of the experiment few parameters have been used. Also calculate the attained results of this model with some metrics of performance. Their definitions and mathematical values are as follows -

- True Positive
- True Negative
- False Positive
- False Negative

4.1.1. Accuracy

Accuracy is defined as the ratio of precise predictions to a total number of proceedings in the dataset. And it is determined as the following Eqn. (4),

$$Accuracy = \frac{(T(P)+T(N))}{(T(P)+T(N)+F(P)+F(N))} \quad (4)$$

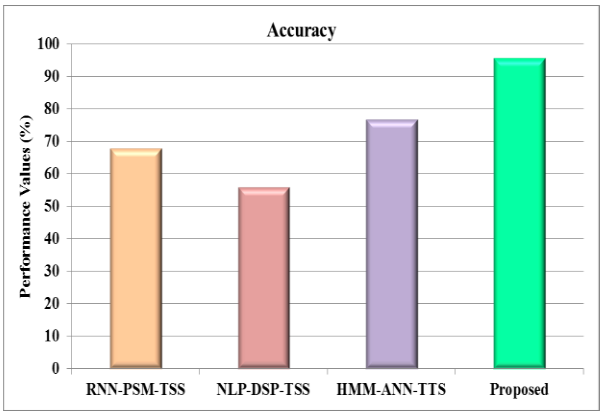

Fig.3 Accuracy Analysis

Fig.3 shows the accuracy examination of proposed model with other prevailing approaches. Performance of the proposed method provides 34.90%, 29.08%, and 34.76% greater accuracy while comparing other techniques such as RNN-PSM-TTS, NLP-DSP-TTS, and HMM-ANN-TTS respectively.

4.1.2. Precision

Precision is defined as the capability of the classifier to convert the text into speech without conditions. It is given by the Eqn. (5),

$$Pr\ e\ cision = {T(P)}\Big/\big(T(P) + F(P)\big) \quad (5)$$

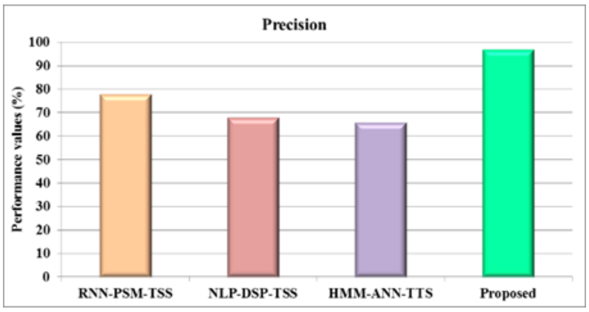

Fig.4 Precision analysis

Fig.4 shows the precision comparison of this proposed model with other prevailing approaches. Performance of proposed method provides 20.90%, 30.89%, and 42.67% greater precision while comparing other techniques such as RNN-PSM-TTS, NLP-DSP-TTS, and HMM-ANN-TTS respectively.

4.1.3. Sensitivity

Sensitivity is defined as calculation of the number of true positives that are synthesized accurately. It is given in the Eqn. (6),

$$Sensitivity = \frac{T(P)}{F(N)+F(P)} \quad (6)$$

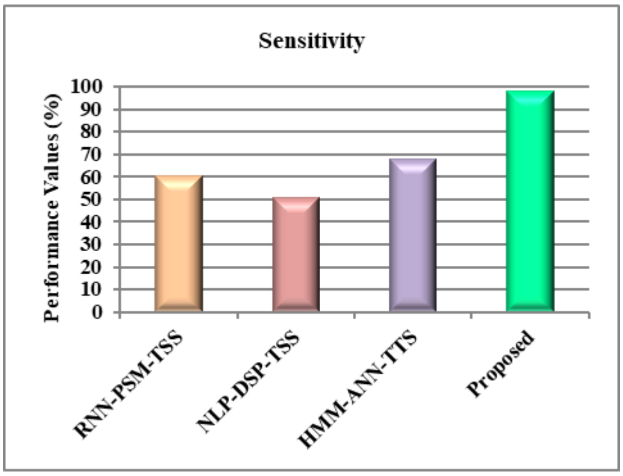

Fig.5 Sensitivity Analysis

Fig.5 shows the sensitivity comparison and it shows the proposed echnique provides 20.90%, 30.89%, and 42.67% greater sensitivity while comparing other techniques such as RNN-PSM-TTS, NLP-DSP-TTS, and HMM-ANN-TTS respectively.

4.1.4. Specificity

The TN rate is defined as the specificity. It is given by the Eqn. (7),

$$Specificity = \frac{F(N)}{F(P)+F(N)} \quad (7)$$

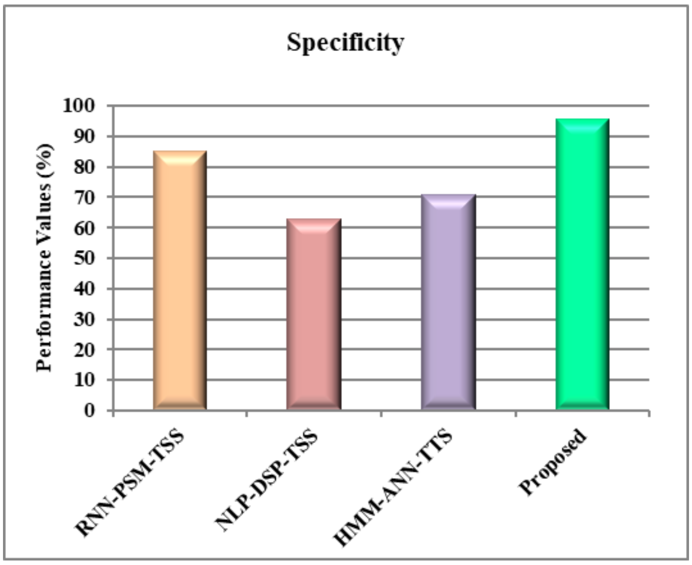

Fig.6 Specificity

Fig.6 indicates the specificity comparison. The attained results of the proposed method provide 29.78%, 27.56%, and 31.56% greater specificity while comparing other techniques such as RNN-PSM-TTS, NLP-DSP-TTS, and HMM-ANN-TTS respectively.

4.1.5. F-Score

It is defined as the harmonic mean of and precision and recall. Also it is the testing measure of accuracy of the text dataset. It is given by the Eqn. (8),

$$F - Score = \frac{2T(P)}{2T(P)+F(P)+F(N)} \quad (8)$$

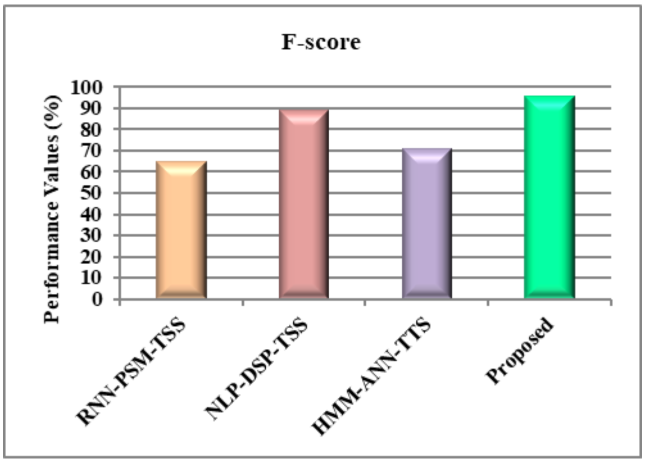

Fig.7 Analysis of F-score

Fig.7 shows the analysis of f-score of the proposed technique provide greater f-score 32.67%, 27.65%, and 26.78% while comparing other techniques such as RNN-PSM-TTS, NLP-DSP-TTS, and HMM-ANN-TTS respectively.

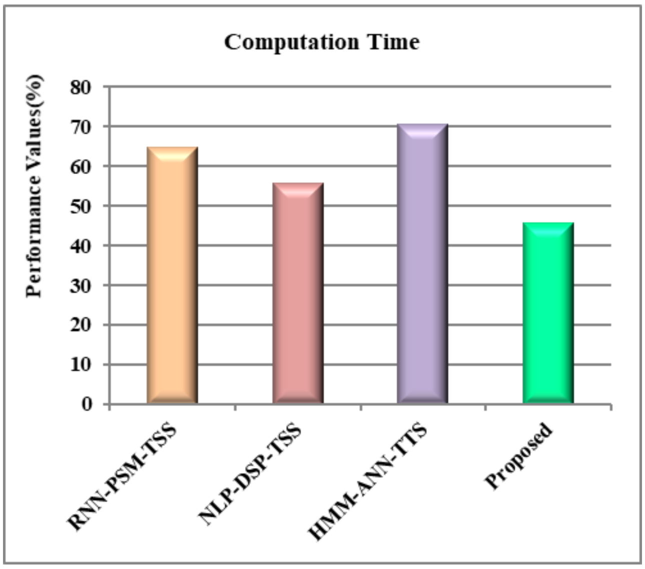

Fig.8 Computation Time Analysis

Fig.8 indicates the analysis of computation time. The performance of the proposed technique provides lower computational time 21.34%, 42.67%, and 32.45% in companion with the existing methods such as RNN-PSM-TTS, NLP-DSP-TTS, and HMM-ANN-TTS respectively.

5. Conclusion

Proposed and designed BbCNS for converting text messages into voice messaging services. Speech quality is also retained while speech synthesis is performed. Moreover, the designed model performs English text to English voice messaging. A prosody method is also proposed, by which a prosody code is retrieved from the source and used to modify the target TTS model by a prosody encoder. Furthermore, the linguistic analysis identifies the constituent morphemes of each work and predicts the prosody. The designed experiments proved the efficiency of voice conversion and also attained performance results of accuracy, sensitivity, precision, f-score, specificity, and computation time are analyzed. Finally, validate the experimental outcomes into other existing techniques and gained 95.67% accuracy, 96.89% precision, 97.9% sensitivity, 98.56% specificity, and 95.78% F-measure. While comparing other models designed a model to attain better outcomes and improve efficiency.

References

[1] ARUL, VH, and RAMALATHA MARIMUTHU. "Speech recognition using Taylor-gradient Descent political optimization based Deep residual network." Computer Speech & Language (2022): 101442.

[2] Agarwal, Gaurav, and Hari Om. "Performance of deer hunting optimization based deep learning algorithm for speech emotion recognition." Multimedia Tools and Applications 80.7 (2021): 9961-9992.

[3] Rathod, Vasundhara S., Ashish Tiwari, and Omprakash G. Kakde. "Wading corvus optimization based text generation using deep CNN and BiLSTM classifiers." Biomedical Signal Processing and Control 78 (2022): 103969.

[4] Gantayat, Harikrushna, Trilochan Panigrahi, and Pradyumna Patra. "An efficient direction-of-arrival estimation of multipath signals with impulsive noise using satin bowerbird optimization-based deep learning neural network." Expert Systems (2022): e13108.

[5] Koteswararao, Yannam Vasantha, and C. B. Rao. "Multichannel speech separation using hybrid GOMF and enthalpy-based deep neural networks." Multimedia Systems 27.2 (2021): 271-286.

[6] Bai, Zhongxin, Xiao-Lei Zhang, and Jingdong Chen. "Partial AUC optimization based deep speaker embeddings with class-center learning for text-independent speaker verification." ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020.

[7] Kothadiya, Deep, Nitin Pise, and Mangesh Bedekar. "Different Methods Review for Speech to Text and Text to Speech Conversion." International Journal of Computer Applications 975: 8887.

[8] Dong, Mingyu, Diqun Yan, and Rangding Wang. "Adversarial Privacy Protection on Speech Enhancement." arXiv preprint arXiv:2206.08170 (2022).

[9] Srivastava, Nidhi, and Sipi Dubey. "Moth Monarch Optimization-Based Deep Belief Network in Deception Detection System." Sādhanā 45.1 (2020): 1-14.

[10] Yuan, Nanqi, et al. "Laplacian Eigenmaps Feature Conversion and Particle Swarm Optimization-Based Deep Neural Network for Machine Condition Monitoring." Applied Sciences 8.12 (2018): 2611.

[11] Ren, Yi, et al. "Fastspeech: Fast, robust and controllable text to speech." Advances in Neural Information Processing Systems 32 (2019).

[12] Huang, Wen-Chin, et al. "Voice transformer network: Sequence-to-sequence voice conversion using transformer with text-to-speech pretraining." arXiv preprint arXiv:1912.06813 (2019).

[13] Zhang, Mingyang, et al. "Joint training framework for text-to-speech and voice conversion using multi-source tacotron and wavenet." arXiv preprint arXiv:1903.12389 (2019).

[14] Ren, Yi, et al. "Fastspeech 2: Fast and high-quality end-to-end text to speech." arXiv preprint arXiv:2006.04558 (2020).

[15] Raiyetunbi, Oladimeji Jude, and Ayeh Emmanuel. "An Interactive Cloud Based User Oriented, Dynamic and Intelligent Text-To-Speech Module." East African Scholars Journal of Engineering and Computer Sciences 3.1 (2020).

[16] SRMIST, Vadapalani, and U. G. Student. "Text-to-speech device for visually impaired people." International Journal of Pure and Applied Mathematics 119.15 (2018): 1061-1067.

[17] Chakladar, Debashis Das, Pradeep Kumar, Shubham Mandal, Partha Pratim Roy, Masakazu Iwamura and Byung-Gyu Kim "3D Avatar Approach for Continuous Sign Movement Using Speech/Text." Applied Sciences 11.8 (2021): 3439.

[18] Manikandan, K., Ayush Patidar, Pallav Walia and Aneek Barman Roy, "Hand gesture detection and conversion to speech and text." arXiv preprint arXiv:1811.11997 (2018).

[19] Anggraini, Nenny, Luh Kesuma Wardhani, Nashrul Hakiem, "Speech recognition application for the speech impaired using the android-based google cloud speech API." TELKOMNIKA (Telecommunication Computing Electronics and Control) 16.6 (2018): 2733-2739.

[20] Zerrouki, Taha, "Adapting espeak to Arabic language: Converting Arabic text to speech language using espeak." International Journal of Reasoning-Based Intelligent Systems 11.1 (2019): 76-89.

[21] Chen, S.H., 2000. A Corpus-Based Prosodic Modeling Method for Mandarin and Min-Nan Text-to-Speech Conversions. ISCSLP.

[22] Rajbongshi, A., Islam, M.I., Biswas, A.A., Rahman, M.M., Majumder, A. and Islam, M.E., 2020. Bangla optical character recognition and text-to-speech conversion using raspberry Pi. International Journal of Advanced Computer Science and Applications, 11(6).

[23] Santra, S., Bhowmick, S., Paul, A., Chatterjee, P. and Deyasi, A., 2018, May. Development of GUI for text-to-speech recognition using natural language processing. In 2018 2nd International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech) (pp. 1-4). IEEE.

[24] Nagdewani, S. and Jain, A., 2020. A REVIEW ON METHODS FOR SPEECH-TO-TEXT AND TEXT-TO-SPEECH CONVERSION.

[25] Talman, A., Suni, A., Celikkanat, H., Kakouros, S., Tiedemann, J. and Vainio, M., 2019. Predicting prosodic prominence from text with pre-trained contextualized word representations. arXiv preprint arXiv:1908.02262.