Journal of Theoretical and Applied Information Technology
15th April 2026. Vol.104. No.7
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

TEXT-TO-SPEECH (TTS) ANALYSIS SYSTEM WITH CLIENT-SIDE PROCESSING

TASKEOW SRISOD[1], PRACHYANUN NILSOOK[2], SASITORN ISSARO[3*], ORAPHAN AMNUAYSIN[3] THANANAN AREEPONG[3], ORAWAN SAEUNG[3], THANI JINTASUTTISAK[3], THAMASAN SUWANROJ[3]

[1]Techinnovation holdings group
[2]King Mongkut's University of Technology North Bangkok, (KMUTNB) Bangkok, Thailand
[3]Nakhon Si Thammarat Rajabhat University, 80280 Thailand

Email: [2]prachyanun.n@fte.kmutnb.ac.th, [*3]sasitorn_iss@nstru.ac.th, [3]Orapan_amn@nstru.ac.th, [3]thananan_are@nstru.ac.th, [3]orawan_ray@nstru.ac.th, [3]thani_jin@nstru.ac.th, [3]thamasan_suw@nstru.ac.th

ABSTRACT

In today's digital age, text-to-speech (TTS) technology has become a crucial tool for increasing accessibility and enhancing user experiences across platforms, from devices for the visually impaired to smart assistants. Developing this technology efficiently and quickly is essential. This article focuses on developing a text-to-speech analysis and synthesis system using client-side processing technology, an approach that enables TTS conversion to occur directly on a user's web browser, thereby reducing server load and increasing response speed. The work covers everything from the process of TTS, user interface (UI) development, to Web Speech API implementation. Furthermore, to ensure the quality of the synthesized voices, a systematic evaluation was conducted using the internationally-standardized Mean Opinion Score (MOS) for Thai voices from Microsoft client-side TTS voices, namely Pattara and Kanya, to measure clarity, naturalness, and fluidity. The results of this project not only serve as a prototype for an effective TTS system, but also provide valuable insights for the future development of synthetic voices that are more natural and closely approximate human speech.
Keywords: *Text to Speech (TTS), Client-Side Processing, Natural Language Processing (NLP)*

1. INTRODUCTION

Text-to-speech (TTS) technology plays a crucial role in the digital age, particularly in the context of intelligent assistants, online education, and providing access to information for the visually impaired. TTS systems convert text into understandable speech by utilizing natural language processing (NLP) techniques in conjunction with speech synthesis. The development of TTS has evolved through several iterations. It began with Rule-Based and Concatenative systems, which use human-defined rules and concatenate phonemes from real sound libraries. However, these approaches have limitations in terms of fluidity. Next came the era of Statistical Parametric Speech Synthesis (SPSS), which uses statistical models such as Hidden Markov Model (HMM) and Gaussian Mixture Models (GMM) to mitigate the limitations of traditional methods. However, while they offer increased flexibility, they lack naturalness [1]. The advent of Deep Learning has given rise to new models such as Tacotron, FastSpeech, and VITS, that can produce high-quality synthetic speech closely resembling human speech. In particular, models in the Transformer-based group that use self-attention and feed-forward blocks accelerate processing and improve synthesis efficiency [2], [3], [4], [5], [6], [7], [8]. Today, TTS systems have evolved to operate client-side, independent of servers or internet connections. This makes them ideal for applications requiring privacy and real-time response, such as screen reader systems. Research uses client-side TTS approaches on the user side, such as in the case of Ugan et al.'s design of speech-to-speech translation applications. Psychiatric treatment involving refugees uses client-side TTS with speech synthesis for those who cannot read, enhancing patient confidentiality in terms of information disclosure [9]. Andersen et al. investigated methods to enhance voice services for new AI speech recognition systems - Vosk and Whisper - by incorporating transcription functionality to reduce network load, as the systems operate on the client-side TTS [10]. In 2018, the potential of client-side processing in the context of

AI processing increased, especially through WebAssembly on modern browsers [11]. Usher and Whitty describe the role of client-side project managers who not only oversee projects but also apply design thinking concepts to solve complex problems involving distributed computing [12]. Motoo et al. proposed a network delay management technique for multiplayer FPS games, focusing on client-side processing to improve accuracy and fairness in gameplay [13]. An et al. presentation introduces a technique for controlling musical tone using a Variational Autoencoder (VAE) system. This addresses the problem of musical characteristics (such as pitch, loudness, and tempo) where adjustments often affect other parameters. VAE allows for direct and easily interpretable musical tone control, limiting the need for pre-defined musical detail data [14]. Pamisetty & Sri Rama Murty's presentation introduces a hybrid model that combines the advantages of statistics and neural networks to solve the problem of fine-tuning audio melody using a modular model. It simulates audio length and pitch values separately, making it ideal for tasks requiring high precision, such as automatic voiceovers that need to perfectly match audio length to the original video [15]. Despite the continuous advancements in TTS and client-side technologies, there remains a scarcity of studies on the voice quality of client-side TTS systems, particularly in the case of Thai. Therefore, despite the rapid development of TTS technology, particularly in client-side applications that address both privacy and real-time usability in the Thai context, there remains a lack of studies focusing on empirical voice quality assessment, specifically in terms of measuring the clarity, naturalness, and intelligibility of synthetic voices in real-world environments. This research aims to fill this gap by focusing on evaluating the quality of Microsoft's client-side TTS voices, namely Pattara and Kanya, using the MOS method to gain insights that can be used as guidelines for the development and improvement of Thai TTS systems in the future. This article consists of five sections: Section 1, Literature review, Section 2, Methodology, Section 3, Results, Section 4, Discussion, and Section 5, Conclusion.

2. LITERATURE REVIEW

2.1 Text-to-Speech

Text-to-Speech (TTS) is the process of automatically converting text into spoken form, a feature that is very important in voice technology applications such as intelligent assistants (e.g., Siri, Google Assistant), assistance for the visually impaired, and conversational AI [3] [5] [7]. The basic principles of TTS processing include text processing/analysis(frontend), phonetic transcription /grapheme-to-phoneme conversion (G2 P), prosody generation/modeling, and waveform synthesis (backend) [16] [17] [18] [19]. In terms of voice quality assessment, TTS tasks involve a voice quality assessment process. The synthesis process is illustrated in Table 1.

Table 1: TTS voice quality assessment.

Evaluation Method	Description	References
Mean Opinion Score (MOS)	This is a rating of the listener's overall satisfaction with the assessment of the synthesized speech, ranging from 1 (very poor) to 5 (very good).	[19], [20], [21], [22], [23], [24], [25]
Comparison Mean Opinion Score (CMOS)	This involves comparing two sets of sounds. The listener is asked to identify which sound is better and to rate the differences.	[7], [26], [27], [28], [29]
Intelligibility Test	This tests the listener's comprehension by requiring them to listen to a word or sentence from the TTS audio and type or select the correct missing word.	[28], [30], [32], [33], [36]
Objective Evaluation (Automatic Metrics)	This utilizes models or formulas such as PESQ, STOI, MCD, or MOSNet to automatically evaluate voice quality.	[19], [29], [34], [35]

Table 1 shows four methods for evaluating TTS voice quality: Mean Opinion Score (MOS). This is a rating of the listener's overall satisfaction with the assessment of the synthesized speech, with scores ranging from 1 (very poor) to 5 (very good). Method 2: Comparison Mean Opinion Score (CMOS). This compares two sets of voices, asking the listener to determine which is better and scoring the difference. Method 3: Intelligibility Test. This tests listener comprehension, for example, by requiring them to listen to a word or sentence in a TTS voice and then

typing or selecting the correct word. Method 4: Objective Evaluation (Automatic Metrics). This utilizes models or formulas such as PESQ, STOI, MCD, or MOSNet to automatically evaluate voice quality. Comparing the pros and cons of each evaluation method reveals that MOS (Mean Opinion Score) utilizes real human data, but it is time-consuming and requires a large number of participants. CMOS (Comparison MOS) offers a comprehensive comparative analysis, although it necessitates evaluating several sets of voice samples. Intelligibility tests can clearly measure comprehension, but they don't measure naturalness. Objective metrics are efficient and automated, although they may not always align with human perception.

This article presents the MOS assessment method. The MOS assessment questions are presented in a synthesis table of the MOS process, as shown in Table 2.

Table 2: Results of the synthesis of MOS assessment questions.

Assessment Topics	MOS Assessment Question Description	References
Clarity of Speech	Evaluates whether speech is clear, audible, and complete, with no background noise or interference that makes it difficult to understand.	[2], [4], [7], [18], [19], [23], [33], [37], [39], [40], [41], [42], [43], [44]
Naturalness	Evaluates whether the speech sounds like a real human voice, is natural in nature, and does not sound harsh or synthetic.	
Fluency	Evaluates whether the speech is continuous, has a smooth rhythm, and does not have inappropriate pauses or interruptions.	
Tone and Rhythm (Prosody)	Assesses the control of tone, pitch, and tempo appropriate to the mood and context of the sentence.	

According to Table 2, the MOS assessment questions involve four steps: Clarity, Naturalness, Fluency, and Tone and Rhythm. The equation for calculating the mean opinion score [23] [44] is as follows:

$$MOS = \frac{1}{N}\sum_{i=1}^{N} s_I \quad (1)$$

where s_i is the score of the i_{th} evaluator.

2.2 Client-Side Processing

Client-Side Processing refers to performing computations or data processing on the user's (client's) device instead of on the server. This enables faster system response times, conserves server resources, and enhances user privacy. It reduces processing latency by eliminating the need to wait for a response from the server, thereby saving server resources. It improves user data privacy and supports partial offline functionality [11].TTS User-Side: TTS is a sophisticated form of speech synthesis designed to convert any textual input into audible audio output. This technology serves as a crucial bridge between written and spoken communication, significantly increasing user accessibility and interaction. TTS was developed as an assistive technology, primarily intended to assist individuals with visual impairments or reading difficulties by enabling them to listen to written content [45] [46]. Distinction: This refers to the distinction between Client-Side vs. Server-Side Processing. In the widely used client-server architecture model, the "client" usually refers to the front-end user. The server handles data, business logic, and complex processing behind the scenes. The front-end consists of the graphical user interface (GUI) and all visual elements such as buttons, text fields, and images, that the user interacts with directly. This front-end manages user interactions, including displaying data and validating input. Complex requests are sent to the back-end. Client-side TTS refers to the entire speech synthesis process, from text analysis and phonetic transcription to melody generation and waveform synthesis, which is performed directly on the user's device, such as a Windows computer or an Android smartphone. A key feature of client-side TTS is its ability to operate without the need for a constant internet connection for core synthesis functions. In contrast, a server/client approach to TTS involves the core model and computationally-intensive tasks residing on a remote server (a cloud service). In this model, the client sends input text to the server, which then returns the synthesized speech. Examples include cloud-based TTS services such as Azure AI Speech, OpenAI's Audio API, Google Cloud TTS, and

Amazon Polly. These services typically require an active internet connection and may incur usage-based charges. This distinction is particularly important, as offline functionality and low latency are key advantages of client-side TTS processing [47][48][49].

2.3 Natural Language Processing

Natural Language Processing (NLP) is a branch of artificial intelligence that focuses on enabling computers to understand, process, and reproduce natural language as used by humans [50]. NLP is widely used in intelligent assistants (e.g., Siri, Alexa), including summarizing, language translation, and social media sentiment analysis. Early NLP research relied on human-written rules, such as POS tagging with rule-based systems. However, with the advent of big data, statistical NLP approaches such as Hidden Markov Models (HMMs) and Conditional Random Fields (CRFs) have become popular for learning from large samples of data [5 1]. Advances in deep learning have revolutionized NLP, particularly following the success of word embeddings such as Word2Vec [52] and GloVe [53], which enable words to be represented as semantic vectors. Subsequently, Recurrent Neural Networks (RNNs) and Long Short-Term Memory networks (LSTMs) were developed for handling sequential data, which is suitable for continuous languages, utilizing the Transformer architecture that revolutionized the entire NLP system and the industry [54]. This has become the foundation for Large Language Models (LLMs) such as BERT [55], which is an encoder-based model for tasks such as classification and question-answering; GPT [56], which is a decoder-based model used for text generation; and T5 [57], which is a unified framework for multiple NLP tasks. NLP applications include machine translation (e.g., Google Translate), and deep learning (based on Transformer), sentiment analysis (e.g., positive/negative/neutral); question-answering (e.g., ChatGPT), and BERT QA, together with information extraction (e.g., extracting specific information from articles, such as organization names, dates, times, and events).

Current literature is predominantly concerned with enhancing the accuracy and naturalness of TTS models. Despite these advances, there remains a notable gap in research addressing TTS system design for user-side processing, systemic analysis of synthesized speech results, and real-time operation within real-world environments. This research directly addresses these gaps by presenting a TTS system that integrates client-side analysis and processing, thereby enabling improved speed, flexibility, and data privacy.

3. METHODLOGY

3.1 Synthesizing the Text-to-Speech Process

The synthesizing process for text-to-speech analysis from national and international databases can be illustrated as shown in Table 3.

Table 3: Synthesis of Text-to-Speech Process.

Process	Description	References
Text Analysis	Analysis of source text, such as word tagging, POS tagging, and text normalization.	[31], [58], [59], [60], [61], [62]
Phoneme Embedding	Converting phonemes to vectors for use in models.	[7], [32], [64], [65], [66], [67]
Language Embedding	Converting language or accent data into a vector that the model understands.	[68], [69], [70], [71], [72], [73], [74]
Positional Encoding	Adding word position information in the sequence so that the Transformer can understand the data sequence.	[58], [75], [76], [77], [78], [79]
Encoder (Transformer)	Creating deep representations from embeddings with the Self-Attention mechanism.	[6],[41], [49], [80], [81], [82], [83], [84], [85], [86],
Variance Predictors	Predicting parameters Such as duration, pitch, and energy.	[7], [23], [87], [88], [89], [90]
Transformer Decoder	Decoding the encoder and variance parameters to create a sound spectrogram.	[6], [41], [49], [80], [81], [82], [83], [84], [85], [86]
Vocoder (Voice Encoder)	Converting spectrograms into audible waveforms,	[25], [27], [91], [92], [93], [94], [95]

Process	Description	References
	such as WaveNet and HiFi-GAN.	

Table 3 provides an overview of the steps involved in the Text-to-Speech analysis process: Step 1: Text Analysis, Step 2: Phoneme Embedding, Step 3: Language Embedding, Step 4: Positional Encoding, Step 5: Encoder (Transformer), Step 6: Variance Predictors, Step 7: Transformer Decoder, and Step 8: Vocoder (Voice Encoder). Then, the approach proceeds to the next step in the Text-to-Speech analysis process.

3.2 Development System

Development of a Text-to-Speech (TTS) system with Client-Side Processing. The design of the text-to-speech analysis was carried out in line with the synthesized steps. The details are as follows:

Step 1: Text Analysis (Linguistic Front-end): Linguistic analysis of the text is performed by breaking down the text into words (tokenization). Each word obtained through the tokenization process is then assigned a sentence type (POS) such as noun, verb, adjective, etc. Next, the "letters" are converted into "phonemes" (G2P) (Grapheme-to-Phoneme) sounds.

Step 2: Phoneme Embedding and Language Embedding: This involves representing phonemes with numeric vectors (Phoneme Embedding) in a machine-readable and processable format. Language Embedding is added, as this system can receive text and select the resulting audio in multiple languages.

Step 3: Adding Positions to the Vector (Positional Encoding): Since the Transformer architecture lacks a sequence structure, the Transformer doesn't know which word comes first or last. Therefore, it adds positional encoding to each token's vector to help the model understand the "order" of words in a sentence.

Step 4: Encoder (Transformer): The encoder in the Transformer architecture consists of the following internal processes:

1) Multi-Head Self-Attention; Self-attention is the process of assigning weights to words in the same sequence, determining how important each word is to the other. Multi-Head refers to dividing the embedding into multiple heads to learn the relationships between multiple perspectives simultaneously, enabling contextual understanding of the word. The main calculation method uses three vectors: the query keyword (Q), the key attribute (K), and the actual returned value (V) obtained by transforming the embedding with weight matrices. Attention: The self-attention weight is calculated [46], [95], [96] using the equation:

$$Attention\ (Q, k, V) = SoftMax\ (\frac{QK^T}{\sqrt{d_k}})\ V \qquad (2)$$

Where d_k is the size of the key vector (used for normalization).

The results of each head are then concatenated and passed through a linear layer.

2) Add & Norm: Add the results from the multi-head self-attention to the original input (Residual Connection) and perform layer normalization.

Add: makes the model learn more easily and more deeply without vanishing gradients. This allows the model to "remember" information from the original input even after multiple layers have been introduced.

Norm: This process adjusts the vector values to have a mean of 0 and a standard deviation of 1 at each layer, resulting in faster and more stable learning for the model.

3) Position-wise Feed-Forward Network processes the data of each position (token) by working on each token vector separately (i.e., not related to other tokens). The equation [97], [98] to calculate the position of each vector is as follows:

$$FFN\ (X) = max\ (0, xW_1 + b_1)\ W_2 + b_2 \qquad (3)$$

Where W1 and W2 are the weight matrices of layers 1 and 2

b1 and b2 are the bias vectors, and max (0,.) is the ReLU (Rectified Linear Unit) function.

4) Add & Normalize again. Take the results from the Feed-Forward and add them to the Feed-Forward input (which is the output of the previous step), and perform another layer normalization.

5) Send the results to the next layer. The data that has passed through one encoder layer becomes the input for the next encoder layer.

Step 5: Variance Predictors: Predict Duration (Pitch), Energy (Energy), and Energy (Duration) to determine how long each phoneme should be, and how much pitch (pitch) and energy (loudness) it should have.

Step 6: Transformer Decoder to Mel-Spectrogram: The model generates a mel-spectrogram (a graph representing the sound energy at each frequency) from the above data.

Step 7: Vocoder (Voice Encoder): Convert the mel-spectrogram to a waveform (representing real speech).

TTS analysis and workflow are shown in Figure 1-2

Journal of Theoretical and Applied Information Technology
15th April 2026. Vol.104. No.7
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

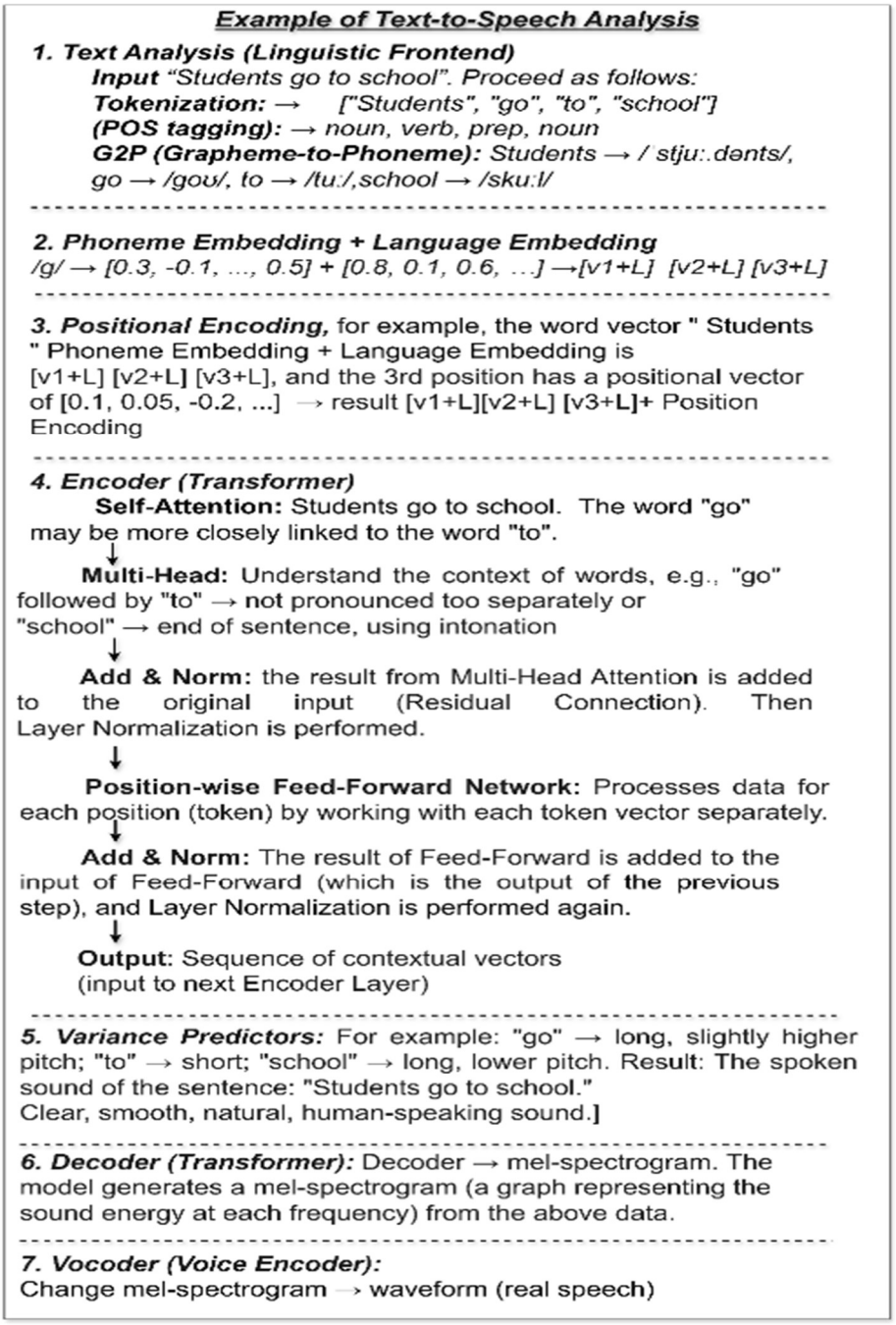

Figure 1: Example of Text-to-Speech Analysis

Journal of Theoretical and Applied Information Technology
15th April 2026. Vol.104. No.7
© Little Lion Scientific
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

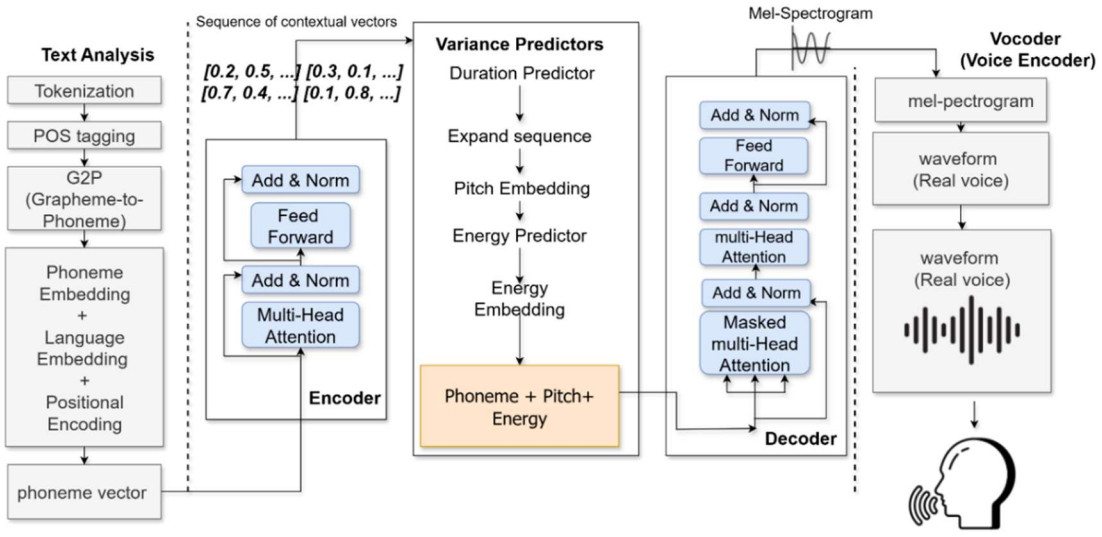

Figure 2: Working Process of Text-to-Speech (TTS)

4. RESULTS

The results of the Text-to-Speech analysis process can be summarized in the form of eight steps: Text Analysis, Phoneme Embedding, Language Embedding, Positional Encoding, Encoder (Transformer), Variance Predictors, Transformer Decoder, and Vocoder (Voice Encoder). The design and development of a TTS analysis system with client-side processing is as follows:

4.1 Creating a user interface (UI) with HTML, CSS, and JavaScript to facilitate user interaction.

4.2 Develop a TTS system using the Web Speech API to convert text to speech, leveraging large-scale AI language models to process questions and analyze content.

4.3 Use File Processing to handle PDF and text files using PDF.js and the FileReader API.

4.4 Use Server-Side Processing using PHP to handle requests and to communicate with the AI model's API.

4.5 The Text-to-Speech (TTS) analysis system is tested with regard to client-side processing as follows: Step 1: Prepare documents as PDF files to test the system. Step 2: Import the PDF file into the system so that the system can analyze the text and synthesize speech. Step 3: Set the questions for evaluating the overall sound quality using the MOS (Mean Opinion Score) method. This consists of 4 questions as follows: ***Clarity of speech (Clarity):*** Speech is clear, can be heard completely, with no noise disturbance. ***Naturalness:*** The voice sounds human, not harsh or synthetic. ***Fluency:*** The voice is continuous, with a smooth rhythm and no interruptions. ***Tone and Rhythm:*** The tone and rhythm are appropriate and consistent with the mood. Step 4: The 30 listeners rated the items on a 5-point Likert scale, where 5 - Excellent, 4 - Good, 3 - Fair, 2 - Poor, and 1 - Very Bad.

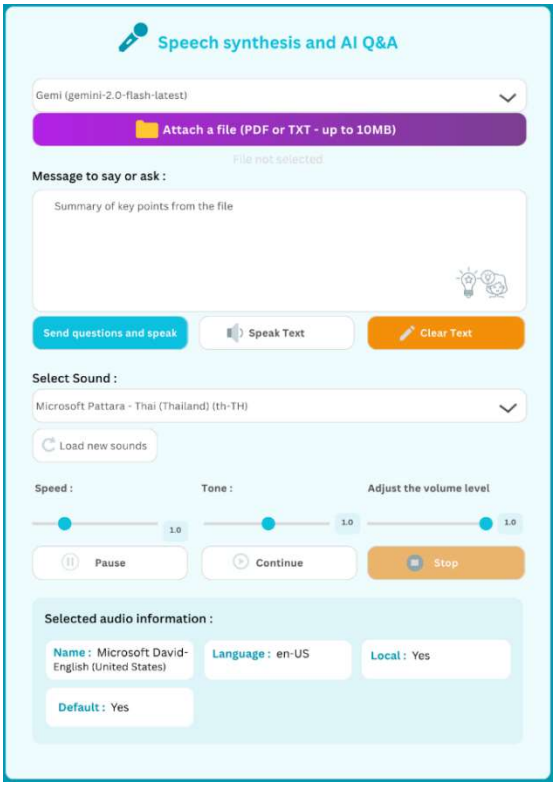

Figure 3: Text-to-Speech (TTS) system screen with client-side processing.

Table 4: Results of Text-to-Speech (TTS) system with Client-side Processing.

System / Language	Clarity	Naturalness	Intelligibility	Fluency	Average MOS (Total)
Microsoft Pattara (TH)	4.33 ± 0.80	3.33 ± 0.71	3.93 ± .14	3.93 ± 1.11	3.88 ± 0.79
Microsoft Kanya (TH)	4.10 ± 0.92	3.30 ± 0.88	3.90 ± 1.11	3.91 ± 1.17	3.81 ± 0.75

The results of system development are shown in Figure 3, and the results of using the Text-to-Speech (TTS) analysis system with client-side processing are as shown in Table 4. Table 4: Client-side TTS system evaluation, summarizing the Mean Opinion Score (MOS) and preliminary analysis of two different Microsoft Thai voices. The evaluation results revealed that Pattara's (male) voice performed slightly better than Kanya's in all aspects, particularly in clarity. The difference between the two voices was not significant, indicating that Microsoft's voice reproduction standards were relatively similar. Naturalness was lower in both systems compared to other aspects, suggesting a need for further development to create more human-like voices.

5. DISCUSSION

This article involves the design and development of a text-to-speech (TTS) analysis system with client-side processing. The process is divided into two aspects: synthesizing the text-to-speech process and designing and developing a text-to-speech system with client-side processing. The researchers used an eight-step text-to-speech process to design and develop a client-side text-to-speech system. This process involved text analysis, converting "letters" into "sounds or phonemes" (G2P). The next steps were Phoneme Embedding and Language Embedding, which represent phonemes and languages with numeric vectors. The positions were then added to the vector (Positional Encoding) to prepare for the Transformer model. This then went to the Encoder (Transformer), which performed various operations until the Variance Predictors process predicted the duration (Pitch), the energy (Energy), and the duration (Energy). The predictions were made with regard to the length of each phoneme, its pitch (Pitch), and the energy (Loudness). This then went to the Transformer and the Decoder to decode the sound. Finally, the Vocoder (Voice Encoder) converted the mel-spectrogram into a waveform, or actual speech. The web development system consisted of the following components. User Interface (UI) development, using the Web Speech API to convert text to speech, and a large-scale AI language model for question processing and content analysis, file management using PDF.js and the FileReader API, and server-side processing using PHP to handle requests and communicate with the AI model's API to display the results. This system operates on the client side and can function without a constant internet connection. Voice quality was assessed using MOS in all four dimensions: Clarity, Naturalness, Fluency, and Prosody. The evaluation results showed that Microsoft Pattara's (male) voice had a slightly higher MOS quality than that of Kanya's (female) in all dimensions. Specifically, in terms of Clarity, this study is consistent with work by Georgia Maniati [100], who used MOS assessment to assess naturalness by brainstorming with a group of people to elaborate on the design of a speech dataset with the aim of developing a model that better resonates with human raters in a speech-level TTS assessment task. Furthermore, Yamagishi et al. [101] indicated that clarity is an important factor affecting comprehension, especially in tone languages such as Thai. In addition, Pattara's (male) voice may have a vocal characteristic that listeners assess as being clearer and more solid, which may affect the MOS score for naturalness. The scores of both voices are not particularly high (around 3.3), which may be due to the limitations of user-side processing using a smaller TTS model than the server-side, in line with Shen Jonathan et al. [102], who proposed that the Tacotron 2 model combined with WaveNet provides smoother and more natural voices but is resource-intensive in terms of Intelligibility and Fluency. Both voices received similarly high scores, reflecting the effectiveness of the Microsoft Neural TTS system, which maintains high voice quality even when operating on the user side. This aligns with Andersen [10], who studied ways to improve voice services for new-generation AI speech recognition systems to reduce network load by operating on the client-side TTS. Furthermore, using client-side TTS helps maintain user confidentiality. This aligns with Ugan et al. [9], who designed a speech-to-speech translation application for psychiatric treatment in cases where refugees could not read, to maintain patient confidentiality when disclosing information.

6. CONCLUSION

The results of this study can be applied to selecting synthetic voices for educational, assistive technology, and accessibility applications. A Text-to-Speech (TTS) Analysis System with Client-side Processing is suitable for applications requiring confidentiality and speed. Client-side TTS can be applied because it operates without requiring an internet connection. This approach can be applied to other programming languages and operating systems, enhancing natural speech. This approach can be used to enhance speech clarity, naturalness, fluency, tone, and rhythm, enabling greater utilization. Although previous research, such as Tacotron 2 [103], has been able to generate highly natural-sounding speech, and the work of An et al. [14] has been able to efficiently control prosody through speech component separation and reduction of mutual information between speech features, both approaches remain model-oriented and heavily reliant on server-side processing. In contrast, this research presents a user-side processing-based TTS analysis system that reduces response time, enhances data privacy, and efficiently supports real-time applications.

REFERENCES

[1] A. J. Hunt and A. W. Black, "Unit Selection in a Concatenative Speech Synthesis System Using a Large Speech Database," in ICASSP, IEEE International Conference on Acoustics, Speech, and Signal Processing - Proceedings, IEEE, 1996, pp. 373–376. doi: 10.1109/icassp.1996.541110.

[2] Y. Wang et al., "Tacotron: Towards end-To-end Speech Synthesis," in Proceedings of the Annual Conference of the International Speech Communication Association, Interspeech, International Speech Communication Association, 2017, pp. 4006–4010. doi: 10.21437/Interspeech.2017-1452.

[3] Y. Ren et al., "FastSpeech: Fast, Robust, and Controllable Text to Speech," Nov. 2019, [Online]. Available: http://arxiv.org/abs/1905.09263

[4] J. Kim, J. Kong, and J. Son, "Conditional Variational Autoencoder with Adversarial Learning for End-to-End Text-to-Speech," Jun. 2021, [Online]. Available: http://arxiv.org/abs/2106.06103

[5] A. Vaswani et al., "Attention Is All You Need," 2023.

[6] N. Li, S. Liu, Y. Liu, S. Zhao, M. Liu, and M. Zhou, "Neural Speech Synthesis with Transformer Network," Jan. 2019, [Online]. Available: http://arxiv.org/abs/1809.08895

[7] J. Kim, S. Kim, J. Kong, and S. Yoon, "Glow TTS: A Generative Flow for Text-to-Speech via Monotonic Alignment Search," Oct. 2020, [Online]. Available: http://arxiv.org/abs/2005.11129

[8] W. Ping et al., "Deep Voice 3: Scaling Text-to-Speech with Convolutional Sequence Learning," Feb. 2018, [Online]. Available: http://arxiv.org/abs/1710.07654

[9] E. Y. Ugan, M. Mediani, O. Al Jawabra, A. Khader, Y. Liu, and A. Waibel, "Modular Design of a Front-End and Back-End Speech-to-Speech Translation Application for Psychiatric Treatment of Refugees," in 2023 IEEE Global Humanitarian Technology Conference, GHTC 2023, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 128–131.doi: 10.1109/GHTC56179.2023.10354809.

[10] E. P. Andersen et al., "AI-enabled Audio and Chat Collaboration Services," in Proceedings – IEEE Military Communications Conference MILCOM, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 728–733. doi: 10.1109/MILCOM61039.2024.10773645.

[11] B. Malle, N. Giuliani, P. Kieseberg, and A. Holzinger, "The Need for Speed of AI Applications: Performance Comparison of Native vs. Browser-based Algorithm Implementations," Feb. 2018, [Online]. Available: http://arxiv.org/abs/1802.03707

[12] Greg Usher and Stephen Jonathan Whitty, "Design Thinking Practitioner," 2018.

[13] T. Motoo, J. Kawasaki, T. Fujihashi, S. Saruwatari, and T. Watanabe, "Client-Side Network Delay Compensation for Online Shooting Games," IEEE Access, vol. 9, pp. 125678–125690, 2021, doi: 10.1109/ACCESS.2021.3111180.

[14] An,X., Soong, F. K., Yang, S., & Xie, L. (2021). Effective and direct control of neural TTS prosody by removing interactions between different attributes. *Neural Networks*, *143*, 250–260. https://doi.org/10.1016/j.neunet.2021.06.006

[15] Pamisetty, G., & Sri Rama Murty, K. (2023). Prosody-TTS: An End-to-End Speech Synthesis System with Prosody Control. *Circuits, Systems, and Signal Processing*, *42*(1), 361–384. https://doi.org/10.1007/s00034-022-02126-z

[16] S. Liao et al., "Fish-Speech: Leveraging Large Language Models for Advanced Multilingual Text-to-Speech Synthesis," Nov. 2024,

[Online]. Available: http://arxiv.org/abs/2411.01156
[17] X. Tan, T. Qin, F. Soong, and T.-Y. Liu, "A Survey on Neural Speech Synthesis," Jul. 2021, [Online]. Available: http://arxiv.org/abs/2106.15561
[18] S. K. Nithin and J. Prakash, "Emotional Speech Synthesis Using End-to-End neural TTS Models," in 18th International Computer Engineering Conference, ICENCO 2022, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 1–7. doi: 10.1109/ICENCO55801.2022.10032463.
[19] M. Flechl, S.-C. Yin, J. Park, and P. Skala, "End-to-end Speech Recognition Modeling from De-identified Data," Jul. 2022, [Online]. Available: http://arxiv.org/abs/2207.05469
[20] C. C. Lo et al., "MosNet: Deep Learning-based Objective Assessment for Voice Conversion," in Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, International Speech Communication Association, 2019, pp. 1541–1545. doi: 10.21437/Interspeech.2019-2003.
[21] D. Stanton, Y. Wang, and R. Skerry-Ryan, "Predicting Expressive Speaking Style from Text in End-to-End Speech Synthesis," Aug. 2018, [Online]. Available: http://arxiv.org/abs/1808.01410
[22] K. Shen et al., "NaturalSpeech 2: Latent Diffusion Models are Natural and Zero-Shot Speech and Singing Synthesizers," May 2023, [Online]. Available: http://arxiv.org/abs/2304.09116
[23] A. Mahaganapathy and K. Sarveswaran, "A Survey and Evaluation of Text-to-Speech Systems for the Tamil Language," Natural Language Processing Journal, vol. 12, p. 100171, Sep. 2025, doi: 10.1016/j.nlp.2025.100171.
[24] A. Kirkland, S. Mehta, H. Lameris, G. E. Henter, E. Szekely, and J. Gustafson, "Stuck in the MOS Pit: A Critical Analysis of MOS Test Methodology in TTS Evaluation," International Speech Communication Association, Aug. 2023, pp. 41–47. doi: 10.21437/ssw.2023-7.
[25] C. Miao et al., "EfficientTTS: An Efficient and High-Quality Text-to-Speech Architecture," 2021.
[26] Y. Kondo, H. Kameoka, K. Tanaka, and T. Kaneko, "Rethinking Mean Opinion Scores in Speech Quality Assessment: Aggregation through Quantized Distribution Fitting," Jun. 2025, [Online]. Available: http://arxiv.org/abs/2506.18307
[27] N. Martinez et al., "Evaluation of Peripartal Calcium Status, Energetic Profile, and Neutrophil Function in Dairy Cows at Low or High Risk of Developing Uterine Disease," J Dairy Sci, vol. 95, no. 12, pp. 7158–7172, Dec. 2012, doi: 10.3168/jds.2012-5812.
[28] J. Kong, J. Park, B. Kim, J. Kim, D. Kong, and S. Kim, "VITS2: Improving Quality and Efficiency of Single-Stage Text-to-Speech with Adversarial Learning and Architecture Design," Jul. 2023, [Online]. Available: http://arxiv.org/abs/2307.16430
[29] X. Zhang, Y. Wang, C. Wang, Z. Li, Z. Chen, and Z. Wu, "Advancing Zero-shot Text-to-Speech Intelligibility across Diverse Domains via Preference Alignment," Jun. 2025, [Online]. Available: http://arxiv.org/abs/2505.04113
[30] P. S. Varadhan et al., "Rethinking MUSHRA: Addressing Modern Challenges in Text-to-Speech Evaluation," May 2025, [Online]. Available:http://arxiv.org/abs/2411.12719
[31] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "An Algorithm for Intelligibility Prediction of Time-Frequency Weighted Noisy Speech," IEEE Trans Audio Speech Lang Process, vol. 19, no. 7, pp. 2125–2136, 2011, doi: 10.1109/TASL.2011.2114881.
[32] T. Raitio, R. Rasipuram, and D. Castellani, "Controllable Neural Text-to-Speech Synthesis Using Intuitive Prosodic Features," in Proceedings of the Annual Conference of the International Speech Communication Association, Interspeech, International Speech Communication Association, 2020, pp. 4432–4436. doi: 10.21437/Interspeech.2020-2861.
[33] D. Paul, M. P. Shifas, Y. Pantazis, and Y. Stylianou, "Enhancing Speech Intelligibility in Text-To-Speech Synthesis Using Speaking Style Conversion." [Online]. Available: https://dipjyoti92.github.io/TTS-Style-Transfer/
[34] T. Raitio, P. Petkov, J. Li, M. Shifas, A. Davis, and Y. Stylianou, "Vocal Effort Modeling in Neural TTS for Improving the Intelligibility of Synthetic Speech in Noise," Mar. 2022, [Online]. Available: http://arxiv.org/abs/2203.10637
[35] G. Mittag, B. Naderi, A. Chehadi, and S. Möller, "NISQA: A Deep CNN-Self-Attention Model for Multidimensional Speech Quality

Prediction with Crowdsourced Datasets," in Proceedings of the Annual Conference of the International Speech Communication Association, Interspeech, International Speech Communication Association, 2021, pp. 2818–2822. doi: 10.21437/Interspeech.2021-299.

[36] L. Finkelstein, J. Camp, and R. Clark, "Importance of Human Factors in Text-To-Speech Evaluations," International Speech Communication Association, Aug. 2023, pp. 27–33. doi: 10.21437/ssw.2023-5.

[37] T. Raitio, R. Rasipuram, and D. Castellani, "Controllable Neural Text-to-Speech Synthesis Using Intuitive Prosodic Features," in Proceedings of the Annual Conference of the International Speech Communication Association, Interspeech, International Speech Communication Association, 2020, pp. 4432–4436. doi: 10.21437/Interspeech.2020-2861.

[38] N. Zeghidour, O. Teboul, F. De, C. Quitry, and M. Tagliasacchi, "Leaf: A Learnable Frontend for Audio Classification."

[39] Y. Wang et al., "Tacotron: Towards End-To-End Speech Synthesis," in Proceedings of the Annual Conference of the International Speech Communication Association, Interspeech, International Speech Communication Association, 2017, pp. 4006–4010. doi: 10.21437/Interspeech.2017-1452.

[40] J. Ao et al., "SpeechT5: Unified-Modal Encoder-Decoder Pre-Training for Spoken Language Processing," Long Papers. [Online]. Available: https://github.com/microsoft/

[41] W. N. Hsu, B. Bolte, Y. H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units," IEEE/ACM Trans Audio Speech Lang Process, vol. 29, pp. 3451–3460, 2021, doi: 10.1109/TASLP.2021.3122291.

[42] R. J. Skerry-Ryan et al., "Towards End-to-End Prosody Transfer for Expressive Speech Synthesis with Tacotron," 2018. [Online]. Available: https://google.

[43] J.-X. Zhang et al., "Voice Conversion by Cascading Automatic Speech Recognition and Text-to-Speech Synthesis with Prosody Transfer," Sep. 2020, [Online]. Available: http://arxiv.org/abs/2009.01475

[44] B. Han et al., "VALL-E R: Robust and Efficient Zero-Shot Text-to-Speech Synthesis via Monotonic Alignment," Jun. 2024, [Online]. Available: http://arxiv.org/abs/2406.07855

[45] J. Kavitha, P. V. Chinmayee, N. Varun Reddy, M. Ramu, and L. Sharmila, "Enhancing Accessibility and Communication through Text to Speech Conversion," in 2024 IEEE Flagship International BIT Conference: Next Generation Applications in Green Energy Technology, BITCON 2024, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/BITCON63716.2024.10985405.

[46] Russ Garcia et al. (2025). Client-Side vs. Server-Side: What's the Difference. Retrieved 10 March 2025. From. https://www.indeed.com/career-advice/career-development/client-side-vs-server-side

[47] Fiveable. (2024). Client-Side and Server-Side Security Controls – Cybersecurity and Cryptography. Retrieved 10 March 2025. From.https://library.fiveable.me/cybersecurity-and-cryptography/unit-13/client-side-server-side-security-controls/study-guide/0d5i2MYckE578BAu

[48] T. Fadhilah Iskandar, M. Lubis, T. Fabrianti Kusumasari, and A. Ridho Lubis, "Comparison between Client-Side and Server-Side Rendering in the Web Development," in IOP Conference Series: Materials Science and Engineering, Institute of Physics Publishing, Jun. 2020. doi: 10.1088/1757-899X/801/1/012136.

[49] Carl, "Examensarbete 30 hp Comparisons of Server-Side Rendering and Client-side Rendering for Web Pages," 2023.

[50] V. Jain, "Server-Side Rendering vs. Client-Side Rendering: A Comprehensive Analysis," IJIRCT2501101 International Journal of Innovative Research and Creative Technology (www.ijirct.org), p. 1, 2021, doi: 10.5281/zenodo.14752604.

[51] J. Hirschberg and C. D. Manning, "Advances in Natural Language Processing," 2015. [Online]. Available: https://www.science.org

[52] C. Manning and H. Schütze, "Lexical Acquisition FSNLP, chapter 8," 1999.

[53] T. Mikolov, K. Chen, G. Corrado, and J.Dean, "Efficient Estimation of Word Representations in Vector Space," Sep. 2013, [Online]. Available: http://arxiv.org/abs/1301.3781

[54] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global Vectors for Word Representation." [Online]. Available: http://nlp.

[55] Y. Galphat, B. Vaswani, C. Gangwani, and S. Dhekale, "EmoSpeak: An Emotionally Intelligent Text-to-Speech System for

Visually Impaired," in International Conference on Advancements in Power, Communication and Intelligent Systems, APCI 2024, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/APCI61480.2024.10616666.

[56] N. J. Devlin, K. K. Shah, Y. Feng, B. Mulhern, and B. van Hout, "Valuing Health-related Quality of Life: An EQ-5D-5L Value Set for England," Health Economics (United Kingdom), vol. 27, no. 1, pp. 7–22, Jan. 2018, doi: 10.1002/hec.3564.

[57] K. S. Kalyan, "A Survey of GPT-3 Family Large Language Models Including ChatGPT and GPT-4," Natural Language Processing Journal, vol. 6, p. 100048, Mar. 2024, doi: 10.1016/j.nlp.2023.100048.

[58] C. Raffel et al., "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer," 2020. [Online]. Available: http://jmlr.org/papers/v21/20-074.html.

[59] J. Zheng, S. Ramasinghe, and S. Lucey, "Rethinking Positional Encoding," Oct. 2021, [Online]. Available: http://arxiv.org/abs/2107.02561

[60] N. Strelkovskii and N. Komendantova, "Integration of UN Sustainable Development Goals in National Hydrogen Strategies: A Text Analysis Approach," Int J Hydrogen Energy, vol. 102, pp. 1282–1294, Feb. 2025, doi: 10.1016/j.ijhydene.2025.01.134.

[61] P. Törnberg, "How to Use LLMs for Text Analysis," Jul. 2023, [Online]. Available: http://arxiv.org/abs/2307.13106

[62] S. Rathje, D. M. Mirea, I. Sucholutsky, R. Marjieh, C. E. Robertson, and J. J. Van Bavel, "GPT Is an Effective Tool for Multilingual Psychological Text Analysis," Proc Natl Acad Sci U S A, vol. 121, no. 34, Aug. 2024, doi: 10.1073/pnas.2308950121.

[63] K. Hansen and A. Świderska, "Integrating Open- and Closed-Ended Questions on Attitudes towards Outgroups with Different Methods of Text Analysis," Behav Res Methods, vol. 56, no. 5, pp. 4802–4822, Aug. 2024, doi: 10.3758/s13428-023-02218-x.

[64] T. Raitio, R. Rasipuram, and D. Castellani, "Controllable Neural Text-to-Speech Synthesis Using Intuitive Prosodic Features," in Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, International Speech Communication Association, 2020, pp. 4432–4436. doi: 10.21437/Interspeech.2020-2861.

[65] M. N. Sundararaman, A. Kumar, and J. Vepa, "Phoneme-BERT: Joint Language Modelling of Phoneme Sequence and ASR Transcript," Jun. 2021, [Online]. Available: http://arxiv.org/abs/2102.00804

[66] A. Santosa, A. Jarin, E. Nurfadhilah, M. T. Uliniansyah, T. Sampurno, and R. Fajri, "End-to-End Phoneme Recognition in Bahasa Indonesia with Pretrained Speech Embeddings and 1D-CNN Using CTC," in International Conference on Computer, Control, Informatics and its Applications, IC3INA, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 145–150. doi: 10.1109/IC3INA64086.2024.10732294.

[67] K. Fujita, A. Ando, and Y. Ijima, "Phoneme Duration Modeling Using Speech Rhythm-Based Speaker Embeddings for Multi-Speaker Speech Synthesis," in Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, International Speech Communication Association, 2021, pp. 3331–3335. doi: 10.21437/Interspeech.2021-826.

[68] Z. Wang, Y. Wang, R. Zeng, R. S. Srinivasan, and S. Ahrentzen, "Random Forest Based Hourly Building Energy Prediction," Energy Build, vol. 171, pp. 11–25, Jul. 2018, doi: 10.1016/j.enbuild.2018.04.008.

[69] L. Gutscher and M. Pucher, "Exploring Phonetic Features in Language Embeddings for Unseen Language Varieties of Austrian German." [Online]. Available: https://huggingface.co/TalTechNLP/

[70] Q. Huang, A. Jansen, J. Lee, R. Ganti, J. Y. Li, and D. P. W. Ellis, "MuLan: A Joint Embedding of Music Audio and Natural Language," Aug. 2022, [Online]. Available: http://arxiv.org/abs/2208.12415

[71] L. Wang, N. Yang, X. Huang, L. Yang, R. Majumder, and F. Wei, "Improving Text Embeddings with Large Language Models," May 2024, [Online]. Available: http://arxiv.org/abs/2401.00368

[72] Z. Nie et al., "When Text Embedding Meets Large Language Model: A Comprehensive Survey," Mar. 2025, [Online]. Available: http://arxiv.org/abs/2412.09165

[73] Z. Naeve, L. Mitchell, C. Reed, P. Campbell, T. Morgan, and V. Rogers, "Introducing Dynamic Token Embedding Sampling of Large Language Models for Improved

Inference Accuracy," Oct. 28, 2024. doi: 10.36227/techrxiv.173014793.37761346/v1.
[74] J. Kerr, C. M. Kim, K. Goldberg, A. Kanazawa, and M. Tancik, "LERF: Language Embedded Radiance Fields." [Online]. Available: https://lerf.io.
[75] X. Wang et al., "KEPLER: A Unified Model for Knowledge Embedding and Pre-trained Language Representation", doi: 10.1162/tacl.
[76] X. Chu, Z. Tian, B. Zhang, X. Wang, and C. Shen, "Conditional Positional Encodings for Vision Transformers," Feb. 2023, [Online]. Available: http://arxiv.org/abs/2102.10882
[77] T. Morita, "Positional Encoding Helps Recurrent Neural Networks Handle a Large Vocabulary," Nov. 2024, [Online]. Available: http://arxiv.org/abs/2402.00236
[78] A. Lopez-Avila, J. Du, A. Shimary, and Z. Li, "Positional Encoding Is Not the Aame as Context: A Study on Positional Encoding for Sequential Recommendation," Jan. 2025, [Online]. Available: http://arxiv.org/abs/2405.10436
[79] E. Apostolidis, G. Balaouras, V. Mezaris, and I. Patras, "Combining Global and Local Attention with Positional Encoding for Video Summarization," in Proceedings - 23rd IEEE International Symposium on Multimedia, ISM 2021, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 226–234. doi: 10.1109/ISM52913.2021.00045.
[80] Y. Li, S. Si, G. Li, C.-J. Hsieh, and S. Bengio, "Learnable Fourier Features for Multi-Dimensional Spatial Positional Encoding."
[81] H. A. Ahmad and T. A. Rashid, "Planning the Development of Text-to-Speech Synthesis Models and Datasets with Dynamic Deep Learning," Sep. 01, 2024, King Saud bin Abdulaziz University. doi: 10.1016/j.jksuci.2024.102131.
[82] Q. Fang, Y. Zhou, and Y. Feng, "DASpeech: Directed Acyclic Transformer for Fast and High-quality Speech-to-Speech Translation." [Online]. Available: https://github.com/ictnlp/DASpeech.
[83] D. Yang et al., "SimpleSpeech 2: Towards Simple and Efficient Text-to-Speech with Flow-based Scalar Latent Transformer Diffusion Models," IEEE Trans Audio Speech Lang Process, pp. 1–14, May 2025, doi: 10.1109/taslpro.2025.3574847.
[84] V. Bataev, S. Ghosh, V. Lavrukhin, and J. Li, "TTS-Transducer: End-to-End Speech Synthesis with Neural Transducer," Institute of Electrical and Electronics Engineers (IEEE), Mar. 2025, pp. 1–5. doi: 10.1109/icassp49660.2025.10890256.
[85] W.-C. Huang, T. Hayashi, Y.-C. Wu, H. Kameoka, and T. Toda, "Voice Transformer Network: Sequence-to-Sequence Voice Conversion Using Transformer with Text-to-Speech Pretraining," Dec. 2019, [Online]. Available: http://arxiv.org/abs/1912.06813
[86] X. Wang, H. Ming, L. He, and F. K. Soong, "s-Transformer: Segment-Transformer for Robust Neural Speech Synthesis," Nov. 2020, [Online]. Available: http://arxiv.org/abs/2011.08480
[87] H. Liu et al., "ViT-TTS: Visual Text-to-Speech with Scalable Diffusion Transformer," Apr. 2024, [Online]. Available: http://arxiv.org/abs/2305.12708
[88] S. Ogun, V. Colotte, and E. Vincent, "Stochastic Pitch Prediction Improves the Diversity and Naturalness of Speech in Glow-TTS," May 2023, [Online]. Available: http://arxiv.org/abs/2305.17724
[89] D. Stanton, Y. Wang, and R. Skerry-Ryan, "Predicting Expressive Speaking Style from Text in End-to-End Speech Synthesis," Aug. 2018, [Online]. Available: http://arxiv.org/abs/1808.01410
[90] D. Seong, H. Lee, and J. H. Chang, "TSP-TTS: Text-Based Style Predictor with Residual Vector Quantization for Expressive Text-to-Speech," in Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, International Speech Communication Association, 2024, pp. 1780–1784. doi: 10.21437/Interspeech.2024-1734.
[91] Y. Lee, J. Yang, and K. Jung, "Variance Flow: High-Quality and Controllable Text-to-Speech Using Variance Information Via Normalizing Flow," in ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 7477–7481. doi: 10.1109/ICASSP43922.2022.9747050.
[92] Y. Gu et al., "ByteSing: A Chinese Singing Voice Synthesis System Using Duration Allocated Encoder-Decoder Acoustic Models and WaveRNN Vocoders," in 2021 12th International Symposium on Chinese Spoken Language Processing, ISCSLP 2021, Institute of Electrical and Electronics Engineers Inc., Jan. 2021. doi: 10.1109/ISCSLP49672.2021.9362104.

[93] F. Bous and A. Roebel, "A Bottleneck Auto-Encoder for F0 Transformations on Speech and Singing Voice," Information (Switzerland), vol. 13, no. 3, Mar. 2022, doi: 10.3390/info13030102.

[94] Y. Kumar, A. Koul, and C. Singh, "A Deep Learning Approaches in Text-to-Speech System: a Systematic Review and Recent Research Perspective," Multimed Tools Appl, vol. 82, no. 10, pp. 15171–15197, Apr. 2023, doi: 10.1007/s11042-022-13943-4.

[95] R. Valle, K. Shih, R. Prenger, and B. Catanzaro, "Flowtron: An Autoregressive Flow-based Generative Network for Text-to-Speech Synthesis," Jul. 2020, [Online]. Available: http://arxiv.org/abs/2005.05957

[96] W.-N. Hsu, D. Harwath, C. Song, and J. Glass, "Text-Free Image-to-Speech Synthesis Using Learned Segmental Units," Dec. 2020, [Online]. Available: http://arxiv.org/abs/2012.15454

[97] Y. Deng, Z. Li, and Z. Song, "Attention Scheme Inspired Softmax Regression," Apr. 2023, [Online]. Available: http://arxiv.org/abs/2304.10411

[98] H. Saratchandran, J. Zheng, Y. Ji, W. Zhang, and S. Lucey, "Rethinking Attention: Polynomial Alternatives to Softmax in Transformers," May 2025, [Online]. Available: http://arxiv.org/abs/2410.18613

[99] Y. Qin et al., "FACT: FFN-Attention Co-optimized Transformer Architecture with Eager Correlation Prediction," in Proceedings - International Symposium on Computer Architecture, Institute of Electrical and Electronics Engineers Inc., Jun. 2023, pp. 301–314. doi: 10.1145/3579371.3589057.

[100] Z. Liu et al., "FFSplit: Split Feed-Forward Network for Optimizing Accuracy-Efficiency Trade-off in Language Model Inference," Jan. 2024, [Online]. Available: http://arxiv.org/abs/2401.04044

[101] G. Maniati et al., "SOMOS: The Samsung Open MOS Dataset for the Evaluation of Neural Text-to-Speech Synthesis," Aug. 2022, doi: 10.21437/Interspeech.2022-10922.

[102] Yamagishi, J., Veaux, C., King, S., & Renals, S. (2012). Speech synthesis technologies for individuals with vocal disabilities: Voice banking and reconstruction. In Acoustical Science and Technology (Vol. 33, Issue 1, pp. 1–5). https://doi.org/10.1250/ast.33.1

[103] Shen Jonathan, Pang Ruoming, and Weiss Ron J, ICASSP: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing: proceedings: April 15-20, 2018, Calgary Telus Convention Center, Calgary, Alberta, Canada. Institute of Electrical and Electronics Engineers, 2018.