Indonesian Journal of Data and Science

Volume 6 Issue 2
ISSN 2715-9936
https://doi.org/10.56705/ijodas.v6i3.327

Research Article

Comparative Analysis of Speech-to-Text APIs for Supporting Communication of the Deaf Community

Anik Nur Handayani [1], Hariyono[2], Ahmad Munjin Nasih[3], Rochmawati[4], Imanuel Hitipeuw[4], Harits Ar Rosyid[5], Jevri Tri Ardiansah[7], Rafli Indar Praja[8], Ahmad Nurdiansyah[9], Desi Fatkhi Azizah[10]

[1] Universitas Negeri Malang, Malang, Indonesia, aniknur.ft@um.ac.id
[2] Universitas Negeri Malang, Malang, Indonesia, hariyono.fis@um.ac.id
[3] Universitas Negeri Malang, Malang, Indonesia, munjin.nasih.fs@um.ac.id
[4] Universitas Negeri Malang, Malang, Indonesia, rochmawati.fip@um.ac.id
[5] Universitas Negeri Malang, Malang, Indonesia, imanuel.hitipeuw.fip@um.ac.id
[6] Universitas Negeri Malang, Malang, Indonesia, harits.ar.ft@um.ac.id
[7] Universitas Negeri Malang, Malang, Indonesia, jevri.ardiansah.ft@um.ac.id
[8] Universitas Negeri Malang, Malang, Indonesia, rafli.indra.2005366@students.um.ac.id
[9] Universitas Negeri Malang, Malang, Indonesia, ahmad.nurdiansyah.2105366@students.um.ac.id
[10] Universitas Negeri Malang, Malang, Indonesia, desi.fatkhi.2405349@students.um.ac.id

Correspondence should be addressed to Anik Nur Handayani; aniknur.ft@um.ac.id

Received 11 October 2025; Accepted 30 December 2025; Published 31 December 2025

Abstract:

Hearing impairment can have a profound impact on the mental and emotional state of sufferers, as well as hinder communication and delay in accessing information directly that relies on interpreters. Advances in assistive technology, especially speech recognition systems that are able to convert spoken language into written text (speech-to-text). However, its implementation faces various challenges related to the level of accuracy of each speech-to-text Application Programming Interface (API), thus requiring an appropriate deep learning model. This study serves to analyze and compare the performance of speech-to-text API services (Deepgram API, Google API and Whisper AI) based on Word Error Rate (WER) and Words Per Minute (WPM), to determine the most optimal API in a web-based real-time transcription system using the JavaScript programming language and Glitch.com. The three API services were tested by calculating their error rates and transcription speeds, then evaluated to see how low the error accuracy rate was and how high the transcription speed was. On average, Whisper AI had a WER of 0% across all word categories, but its speed was lower than the other two APIs. Deepgram API displayed the best balance between accuracy and speed, with an average WER of 13.78% and 67 WPM. Google API performed stably, but its WER value was slightly higher than Deepgram API. In conclusion, based on the results, Deepgram API was deemed the most optimal for live transcription, as it is capable of producing fast and error-free transcriptions, significantly increasing the accessibility of information for the deaf community.

Keywords: Speech-to-Text, API, Word Error Rate (WER), Word Per Minute (WPM), Deaf Community

1. Introduction

Hearing is an essential aspect of human life that is often overlooked until it is impaired or lost. Hearing loss has been shown to significantly impact the mental and emotional well-being of those affected [1], [2]. Along with the development of assistive technology, speech recognition optimization has made a significant contribution to increasing information accessibility for deaf people [3], [4]. Speech recognition system is an interdisciplinary field involving language processing, signal processing and AI (Artificial Intelligence) which consists of speech where the sound signal is continuous with a sequence of phonemes and fundamental modes, so it plays an important role for deaf users [5]. Speech-to-text technology automatically converts speech to text, bridging the communication gap between individuals with hearing impairments and the general public. Implementing this technology plays a crucial role in supporting the deaf community [6], [7], [8], [9]. The speech-to-text system is developed with the JavaScript programming language and the Glitch.com platform that allows users to create, edit, and run projects from a browser without the need for additional software.

Various APIs such as Deepgram API, Google API, and Whisper AI have been developed in terms of accuracy, speed, and different resource requirements [10], [11]. Performance differences between APIs often pose a challenge in determining the best solution for real-time communication. Factors such as WER, and WPM need to be thoroughly analyzed to ensure their suitability for use by the deaf community [12], [13]. While many speech-to-text APIs are available on the market, there is still a lack of comprehensive performance comparison information on APIs suitable for use by the deaf community. This lack of comparative studies often leads to trial-and-error API selection, which can hinder effective system implementation [14]. To address this, testing is carried out using the deep learning approach found within each API. This method has proven to be a powerful tool for automatically exploiting large-scale training data to create complex analytical systems, such as those used in ASR (Automatic Speech Recognition) [15], [16], [17], [18]. The detailed specifications and architecture of the deep learning models used are not publicly available, but there are indications that speech recognition can work by storing various data in each API [19], [20]. This is proven by the large number of vocabulary datasets owned by the Deepgram API which already support Indonesian, so it has a wide vocabulary coverage.

The main objective of this research is to analyze the performance comparison of three speech-to-text APIs (Google API, Deepgram API, Whisper AI) based on WER and WPM values. In addition, this research aims to achieve high values in WPM and low values in WER based on testing of each APIs [21], [22]. The research question is focused on the performance of the API that is most superior in terms of accuracy and speed for the real-time communication needs of the deaf community.

The scope of this research focuses on three speech-to-text APIs, namely Google API, Deepgram API, and Whisper AI, testing using Indonesian with varying word counts of 10, 15, 20, 25, 30, and 35 words. The variation in the number of words was chosen to obtain more varied error rate results and transcription speeds from each trial of each API. The scope of testing includes two main indicators, namely WER, and WPM, which are analyzed under controlled test conditions. The limitations of this research are not testing against environmental factors such as extreme noise or significant network signal interference, and does not include testing with dialogue-shaped sentences. This limitation is intended to maintain the consistency of the results, because the system's capabilities are still limited and not yet adequate.

2. Method

This research began with a systematic and procedural design based on the steps in a flowchart adapted to the Deaf Converter tool development process. This design can help determine the effectiveness of experiments conducted to compare the Deepgram API, Google API, and Whisper AI for use in speech-to-text tools. This research method consists of six main stages, as illustrated in **Figure 1**.

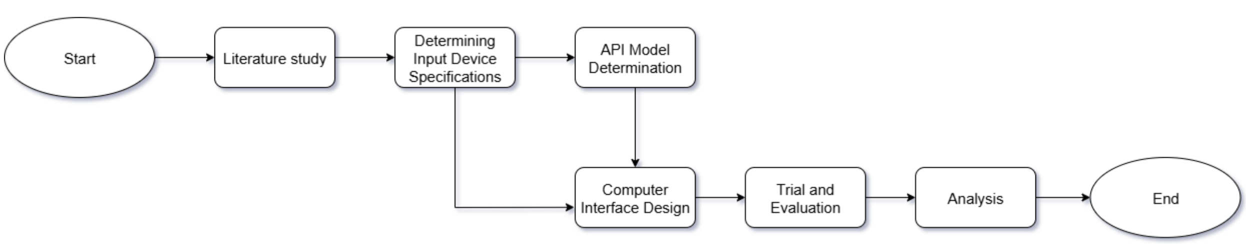

Figure 1. Research Flowchart

Literature Study

At this stage, a comprehensive literature review is conducted by searching various scientific journals, papers, scientific works, and articles on similar topics. This literature review is necessary to avoid plagiarism in titles and serve as a proper reference base for the research. It also ensures the originality of the topic so that the research can make a clear contribution to the development of science [23]. Literature can be obtained through reputable databases such as Scopus, Google Scholar, and ScienceDirect [24]. Further review of previous research was also conducted to ensure alignment with the theoretical basis that will be used in this research.

Determining Input Device Specifications

The next step is to determine the minimum specifications of the input devices, both in terms of software and hardware, required to support the speech recognition trial process in this study. The software is in the form of a Microsoft Edge browser used to open the speech-to-text web and the hardware used is a Fifine A6 microphone and a PC with a minimum specification of 2GB Random Access Memory (RAM) and 128 GB storage type Solid State Drive (SSD) and a Central Processing Unit (CPU) used Intel® Core™ i3. Determining these specifications aims to ensure that the system can run optimally, stably, and according to testing needs, while minimizing potential technical obstacles during implementation [25]. Regular checks of software and hardware devices are required to ensure good performance and effective use.

API Model Determination

At this stage, an efficient API model is determined. API selection is performed by determining the overall WER and WPM values of the identified APIs. The WER and WPM values can be calculated using the following equations:

$$WER = \frac{S + D + I}{N} * 100\% \quad \text{(i)}$$

$$WPM = \frac{N}{T} \times 60 \quad \text{(ii)}$$

This formula is used to calculate the errors in the transcription results, where S is substitution (the number of words that are incorrectly transcribed), D is deletion (the number of words that are missed from the transcription process), I is insertion (the number of incorrect words that are output by the transcription process) [26], [27]. And for WPM, N is the number of words produced in the transcription and T is the Speech Output in seconds or output speed when using a speech-to-text tool [28]. This calculation is used to assess how fast the system can convert speech to text, which is an indicator of the real-time performance of speech-to-text technology.

Computer Interface Design

This stage is the implementation stage after the best API method has been selected and found. Using the Microsoft Edge browser and the Glitch.com website as an IDE (Integrated Development Environment), the interface is designed with user-friendliness in mind. Glitch.com supports both JavaScript and HTML programming languages, so a combination of the two is used as the basis for designing the speech recognition website, which can be accessed from the project page.

The program will have two options that users can select on the right side of the program: Transcribe, which is used to run the program, and the badge option, which is a feature of the program's main menu. The appearance and output of the transcription results can be seen in **Figure 2**.

Figure 2. Program Interface Prototype

Trial and Evaluation

The trial and evaluation phase is the empirical data collection phase. This phase begins with system design. This phase begins with the creation of a project key, program creation, debugging (program improvements), training a custom dataset, and conducting simulation tests on each API, namely the Deepgram API, Google API, and Whisper AI. After testing, performance evaluation is conducted based on parameters such as accuracy, processing speed, and network latency. This approach aligns with empirical practice, where the importance of emphasizing the analysis of system effectiveness and reliability [29], [30]. This can ensure consistent performance and minimize potential errors in transcription results.

Analysis

The final stage is analysis, where the entire research process is carried out by identifying test results, namely the WER value as an indicator of transcription accuracy, the WPM value as a measure of processing speed, and network latency, which represents the delay time in sending and receiving data. These three parameters were chosen because they can provide a comprehensive picture of system performance, in terms of speed, time efficiency, and connection stability.

Test results from each APIs were analyzed to identify performance and determine the most optimal API for the real-time transcription system, based on a balance between accuracy, and processing speed [9]. Therefore, the selected API implementation is expected to make a significant contribution to improving information accessibility for the deaf community.

System Operational Layout Design

This Next, in this study, the operational layout of the system was designed as shown in **Figure 3**. The process begins with the tutor providing input into the speech-to-text processing program running on a PC. The transcription results are then displayed on the output media in the form of an LCD layer, so that the resulting information can be observed visually. This design is expected to facilitate access to information for the deaf community, by minimizing communication barriers through direct and real-time text presentation.

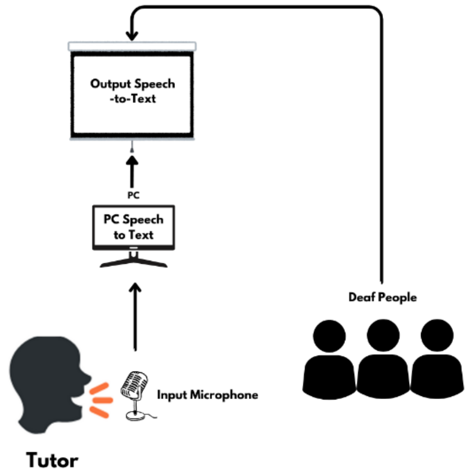

Figure 3. System Operational Layout

3. Result and Discussion

Result

This study compares the effectiveness of APIs used in speech-to-text technology between Deepgram API, Google API, and Whisper AI by testing the WER and WPM values from each experiment that will be calculated to find the best value between low WER and high WPM. The experiment was carried out using a Fifine A6 microphone as an input device. Based on the values in **Table 1** below that have been calculated from each experiment between the three APIs, the Whisper AI is consistent with a WER value of 0% in all categories, indicating no errors in transcription. Deepgram API also shows high accuracy with the majority recording a WER value of 0% although it increased by 12% in the 25-word category before returning low. Google API has a relatively stable WER value level of 0-10% in the initial category, but occasionally reaches a value of 6.67% in the 30-word category and 5.71% in the 35-word category, thus indicating that the WER value of Google API is greater than Deepgram API and Whisper AI.

In terms of WPM, Deepgram API and Google API show higher values than Whisper AI which is proven by a WPM value of 50 in the 10-word category and increases consistently to 79-80 WPM in the 35-word category. Whisper API has a lower value in the 10-word category of 36-40 WPM and experiences an increase in value as the number of words increases to 73-74 WPM in the 35-word category.

Table 1. Calculation results

		Deepgram API		**Google API**		**Whisper AI**	
		WER	WPM	WER	WPM	WER	WPM
10	Attempt 1	0,00%	50	0,00%	50	0,00%	40
	Attempt 2	10,00%	50	10,00%	50	0,00%	36
	Attempt 3	0,00%	53	0,00%	53	0,00%	38
	Attempt 4	0,00%	53	0,00%	53	0,00%	38
	Attempt 5	0,00%	50	0,00%	50	0,00%	37
15	Attempt 1	0,00%	60	0,00%	58	0,00%	51
	Attempt 2	0,00%	60	0,00%	58	0,00%	50
	Attempt 3	6,67%	60	0,00%	57	0,00%	50
		Deepgram API		**Google API**		**Whisper AI**	
		WER	WPM	WER	WER	WPM	WER
	Attempt 4	0,00%	59	0,00%	57	0,00%	50
	Attempt 5	0,00%	66	6,67%	57	0,00%	50
20	Attempt 1	0,00%	66	0,00%	66	0,00%	58
	Attempt 2	5,00%	65	5,00%	65	0,00%	59

		Deepgram API		Google API		Whisper AI	
	Attempt 3	0,00%	66	5,00%	66	0,00%	59
	Attempt 4	5,00%	65	5,00%	65	0,00%	59
	Attempt 5	0,00%	66	5,00%	66	0,00%	59
25	Attempt 1	12,00%	73	8,00%	73	0,00%	68
	Attempt 2	4,00%	74	8,00%	74	0,00%	68
	Attempt 3	4,00%	73	0,00%	73	0,00%	68
	Attempt 4	12,00%	73	8,00%	73	0,00%	68
	Attempt 5	4,00%	73	4,00%	73	0,00%	68
30	Attempt 1	3,33%	74	6,67%	74	0,00%	72
	Attempt 2	6,67%	74	6,67%	74	0,00%	72
	Attempt 3	0,00%	75	0,00%	74	0,00%	73
	Attempt 4	3,33%	74	6,67%	74	0,00%	72
	Attempt 5	6,67%	74	6,67%	74	0,00%	72
35	Attempt 1	0,00%	79	2,86%	79	0,00%	74
	Attempt 2	0,00%	80	0,00%	79	0,00%	74
	Attempt 3	0,00%	80	5,71%	79	0,00%	74
	Attempt 4	0,00%	80	2,86%	79	0,00%	74
	Attempt 5	0,00%	80	0,00%	79	0,00%	73

Discussion

This section will discuss the performance test results of three speech-to-text API services: Deepgram API, Google API, and Whisper AI, using two key indicators: WER and WPM values. The analysis focuses on how the systems maintain transcription accuracy and handle word count variations. These findings will be interpreted considering the needs of web-based real-time transcription.

Based on the test results that can be visualized with the graph in **Figure 5**, Whisper AI shows a high level of accuracy with a value of 0% in the overall word count category, forming a flat and stable trend line without fluctuation. However, the WPM trend experiences an increasing trend in processing speed as the number of words increases. In contrast, Deepgram API and Google API show a WER trend with little fluctuation: Both maintain a low error rate, with a value of mostly 5% with a slight spike in the middle category. The WPM trend of both experiences a steady increase and continues to increase until it approaches the maximum value of the 35 word category. The consistent increase in WPM in these two APIs shows that they are still able to handle larger inputs while maintaining a stable transcription speed.

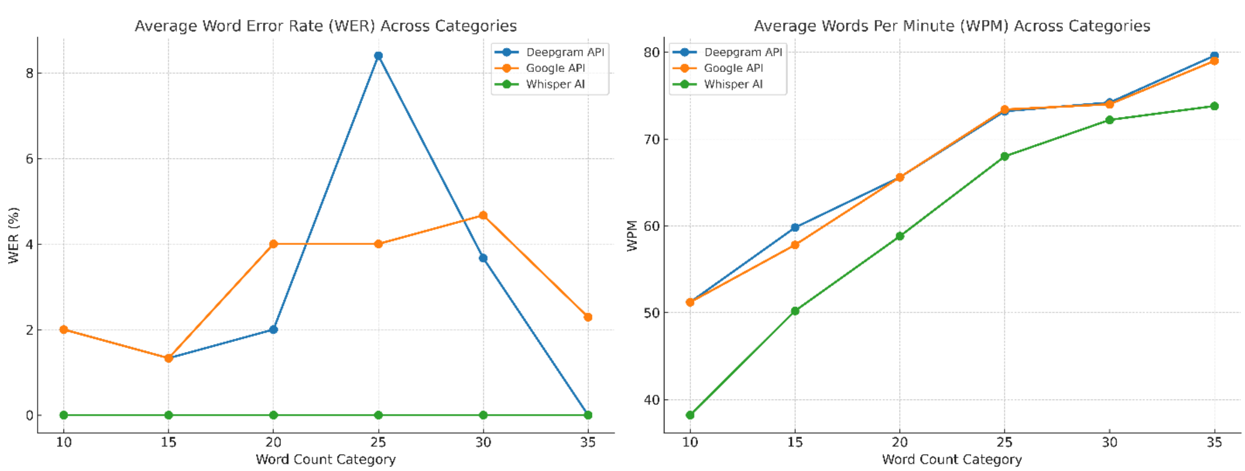

Figure 4. WER and WPM graphs of each API

As can be seen in **Figure 5** above, the combination of a flat WER trend on Whisper AI with a continuously increasing WPM trend on all systems illustrates a trade-off, where Whisper AI relies on the optimal side of accuracy stability, while Deepgram API and Google API maintain a balance between speed and accuracy on various workloads.

In addition, when the overall results are averaged across each API as shown in **Table 2**. The lowest Avg. WER is at 0% obtained by Whisper AI using pre-recorded audio, but has the lowest Avg. WPM among the other baselines with a speed as high as 60 WPM. The highest Avg. WPM is obtained by Deepgram API and Google API with a speed as high as 67 WPM and for the lowest Avg. WER is 13.78% owned by Deepgram API.

Table 2. Overall average results

Parameter	Deepgram API	Google API	Whisper AI
Avg. WER	13,78%	17,13%	0%
Avg. WPM	67	67	60
Noted	Live Transcription	Live Transcription	Pre- Recorded

Based on the data, the Whisper AI system is not a good solution for live transcription needs for the deaf community due to its low Avg. WPM and its reliance on pre-recorded audio to achieve high accuracy results, Live Transcription Deepgram API has a lower Avg. WER than Google API, namely Deepgram API shows the most promising results among the three baseline candidates with high WPM speeds accompanied by low WER.

4. Conclusion

This study obtained the results of an analysis of three comparisons of speech-to-text API services, namely Deepgram API, Google API, and Whisper AI, with the main indicators of WER and WPM. The results of the analysis show that Whisper AI has a WER value of 0% with high accuracy, but this cannot be used for direct communication and can only be used in a pre-recorded manner. On the other hand, Deepgram API and Google API show a balance between speed and accuracy, but Deepgram API has a lower accuracy value WER than Google API at 13.78% and a relatively high transcription speed value of 67 WPM, so Deepgram is a more promising solution to be used in assisting live transcription communication for the deaf community. This research can form the basis for developing more efficient and sophisticated assistive technology. Future research should include features such as automatic punctuation and customizable vocabulary to improve transcription quality. Regular updates to the Deepgram API model to the latest version are necessary to significantly improve the accuracy and smoothness of the transcription process. Furthermore, integrating Open AI with the Deepgram API's deep learning algorithms can enhance system performance and improve accuracy and intelligence.

5. Acknowledgments

The author expresses his deepest appreciation to all parties who provided support, input, and contributions during the preparation of this research. He also expresses his gratitude to Malang State University for its support and facilities, particularly through funding this research, which enabled the successful implementation of all activities.

References:

[1] M. Shezi and A. Ade-Ibijola, "Deaf Chat: A Speech-to-Text Communication Aid for Hearing Deficiency," *Advances in Science, Technology and Engineering Systems Journal*, vol. 5, no. 5, pp. 826–833, 2020, doi: 10.25046/aj0505100.

[2] D. M. P. Jayakody *et al.*, "Is There an Association Between Untreated Hearing Loss and Psychosocial Outcomes," *Front Aging Neurosci*, vol. 14, May 2022, doi: 10.3389/fnagi.2022.868673.

[3] P. Patel, S. Pampaniya, A. Ghosh, R. Raj, D. Karuppaih, and S. Kandasamy, "Enhancing Accessibility Through Machine Learning: A Review on Visual and Hearing Impairment Technologies," *IEEE Access*, vol. 13, pp. 33286–33307, 2025, doi: 10.1109/ACCESS.2025.3539081.

[4] P. A. Rodríguez-Correa, A. Valencia-Arias, O. N. Patiño-Toro, Y. Oblitas Díaz, and R. Teodori De la Puente, "Benefits and development of assistive technologies for Deaf people's communication: A systematic review," *Front Educ (Lausanne)*, vol. 8, Apr. 2023, doi: 10.3389/feduc.2023.1121597.

[5] L. A. Kumar, D. K. Renuka, S. L. Rose, M. C. Shunmuga priya, and I. M. Wartana, "Deep learning based assistive technology on audio visual speech recognition for hearing impaired," *International Journal of Cognitive Computing in Engineering*, vol. 3, pp. 24–30, Jun. 2022, doi: 10.1016/j.ijcce.2022.01.003.

[6] P. A. Rodríguez-Correa, A. Valencia-Arias, O. N. Patiño-Toro, Y. Oblitas Díaz, and R. Teodori De la Puente, "Benefits and development of assistive technologies for Deaf people's communication: A systematic review," *Front Educ (Lausanne)*, vol. 8, Apr. 2023, doi: 10.3389/feduc.2023.1121597.

[7] A. Apriyanto, A. Intes, S. Rachmawati, S. Hanim, and A. K. Alhamdani, "Supporting Inclusivity Through an Automatic Transcription Application to Improve Hearing Skills for the Deaf," *Journal International of Lingua and Technology*, vol. 3, no. 2, pp. 425–440, Aug. 2024, doi: 10.55849/jiltech.v3i2.672.

[8] K. Kuhn, V. Kersken, and G. Zimmermann, "Communication Access Real-Time Translation Through Collaborative Correction of Automatic Speech Recognition," in *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, New York, NY, USA: ACM, Apr. 2025, pp. 1–8. doi: 10.1145/3706599.3719984.

[9] R. Whetten, M. T. Imtiaz, and C. Kennington, "Evaluating Automatic Speech Recognition in an Incremental Setting," Feb. 2023 http://arxiv.org/abs/2302.12049.

[10] A. A. H. Al-Shateri, R. A. Rashid, A. Aburaya, and N. A. Muhammad, "Luna: A Benchmark Project in the Convergence of Artificial Intelligence and Internet of Things for Home Automation," in *2024 IEEE International Conference on Advanced Telecommunication and Networking Technologies (ATNT)*, IEEE, Sep. 2024, pp. 1–4. doi: 10.1109/ATNT61688.2024.10719226.

[11] G. Arop, "Integration Of A Speech Recognition System Into Fulafia FMIS," *SSRN Electronic Journal*, 2025, doi: 10.2139/ssrn.5096981.

[12] K. Hux, J. A. Brown, S. Wallace, K. Knollman-Porter, A. Saylor, and E. Lapp, "Effect of Text-to-Speech Rate on Reading Comprehension by Adults With Aphasia," *Am J Speech Lang Pathol*, vol. 29, no. 1, pp. 168–184, Feb. 2020, doi: 10.1044/2019_AJSLP-19-00047.

[13] R. Yakubovskyi and Y. Morozov, "Speech Models Training Technologies Comparison Using Word Error Rate," *Advances in Cyber-Physical Systems*, vol. 8, no. 1, pp. 74–80, May 2023, doi: 10.23939/acps2023.01.074.

[14] L. N. Yeganeh, N. S. Fenty, Y. Chen, A. Simpson, and M. Hatami, "The Future of Education: A Multi-Layered Metaverse Classroom Model for Immersive and Inclusive Learning," *Future Internet*, vol. 17, no. 2, p. 63, Feb. 2025, doi: 10.3390/fi17020063.

[15] M. Telmem, N. Laaidi, and H. Satori, "The impact of MFCC, spectrogram, and Mel-Spectrogram on deep learning models for Amazigh speech recognition system," *Int J Speech Technol*, vol. 28, no. 1, pp. 299–312, Mar. 2025, doi: 10.1007/s10772-025-10183-3.

[16] Y. O. Sharrab, H. Attar, M. A. H. Eljinini, Y. Al-Omary, and W. E. Al-Momani, "Advancements in Speech Recognition: A Systematic Review of Deep Learning Transformer Models, Trends, Innovations, and Future Directions," *IEEE Access*, vol. 13, pp. 46925–46940, 2025, doi: 10.1109/ACCESS.2025.3550855.

[17] A. B. P. Utama, A. P. Wibawa, A. N. Handayani, and M. Y. Chuttur, "Exploring the Role of Deep Learning in Forecasting for Sustainable Development Goals: A Systematic Literature Review," *International Journal of Robotics and Control Systems*, vol. 4, no. 1, pp. 365–400, Mar. 2024, doi: 10.31763/ijrcs.v4i1.1328.

[18] A. B. P. Utama, A. P. Wibawa, A. N. Handayani, W. S. G. Irianto, Aripriharta, and A. Nyoto, "Improving Time-Series Forecasting Performance Using Imputation Techniques in Deep Learning," in *2024 International Conference on Smart Computing, IoT and Machine Learning (SIML)*, IEEE, Jun. 2024, pp. 232–238. doi: 10.1109/SIML61815.2024.10578273.

[19] A. Karibayeva, V. Karyukin, B. Abduali, and D. Amirova, "Speech Recognition and Synthesis Models and Platforms for the Kazakh Language," Jul. 28, 2025. doi: 10.20944/preprints 202507.2282.v1.

[20] K. Ko, S. Kim, and H. Kwon, "Selective Audio Perturbations for Targeting Specific Phrases in Speech Recognition Systems," *International Journal of Computational Intelligence Systems*, vol. 18, no. 1, p. 103, May 2025, doi: 10.1007/s44196-025-00844-1.

[21] E. Pusateri *et al.*, "Retrieval Augmented Correction of Named Entity Speech Recognition Errors," in *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, Apr. 2025, pp. 1–5. doi: 10.1109/ICASSP49660.2025.10888936.

[22] S. D. Stavisky, "Restoring Speech Using Brain–Computer Interfaces," *Annu Rev Biomed Eng*, vol. 27, no. 1, pp. 29–54, May 2025, doi: 10.1146/annurev-bioeng-110122-012818.

[23] M. L. Rethlefsen *et al.*, "PRISMA-S: an extension to the PRISMA statement for reporting literature searches in systematic reviews," *Journal of the Medical Library Association*, vol. 109, no. 2, Jul. 2021, doi: 10.5195/jmla.2021.962.

[24] B. Hammer, E. Virgili, and F. Bilotta, "Evidence-based literature review: De-duplication a cornerstone for quality," *World J Methodol*, vol. 13, no. 5, pp. 390–398, Dec. 2023, doi: 10.5662/wjm.v13.i5.390.

[25] A.-L. Georgescu, A. Pappalardo, H. Cucu, and M. Blott, "Performance vs. hardware requirements in state-of-the-art automatic speech recognition," *EURASIP J Audio Speech Music Process*, vol. 2021, no. 1, p. 28, Dec. 2021, doi: 10.1186/s13636-021-00217-4.

[26] A. Ferraro, A. Galli, V. La Gatta, and M. Postiglione, "Benchmarking open source and paid services for speech to text: an analysis of quality and input variety," *Front Big Data*, vol. 6, Sep. 2023, doi: 10.3389/fdata.2023.1210559.

[27] E. Kumalija and Y. Nakamoto, "Performance evaluation of automatic speech recognition systems on integrated noise-network distorted speech," *Frontiers in Signal Processing*, vol. 2, Sep. 2022, doi: 10.3389/frsip.2022.999457.

[28] T. Olatunji *et al.*, "A multi-country study comparing typed to automatic speech recognition-based medical documentation speeds among Low- and Middle-Income Country Trained Clinicians," May 13, 2025. doi: 10.1101/2025.05.11.25327386.

[29] L. Yang *et al.*, "Quality Assessment in Systematic Literature Reviews: A Software Engineering Perspective," *Inf Softw Technol*, vol. 130, p. 106397, Feb. 2021, doi: 10.1016/j.infsof.2020.106397.

[30] S. Vegas, P. Riofrio, E. Marcos, and N. Juristo, "On (Mis)perceptions of testing effectiveness: an empirical study," Feb. 2024, doi: 10.1007/s10664-020-09805-y.