International Journal Software Engineering and Computer Science (IJSECS)
5 (3), 2025, 1117-1125
Published Online December 2025 in IJSECS (http://www.journal.lembagakita.org/index.php/ijsecs)
P-ISSN: 2776-4869, E-ISSN: 2776-3242. DOI: https://doi.org/10.35870/ijsecs.v5i3.5459.

RESEARCH ARTICLE Open Access

Implementation of AI-Based Natural Language Processing (NLP) for Automatic Meeting Minutes and Summarization Using Voice-to-Text on Mobile Applications

Muhammad Umar Hamidi Yusuf *
Informatics Engineering Study Program, Faculty of Computer Science, Sekolah Tinggi Ilmu Komputer Cipta Karya Informatika, East Jakarta City, Special Capital Region of Jakarta, Indonesia.
Corresponding Email: mhmmdumar943@gmail.com.

Dadang Iskandar Mulyana
Informatics Engineering Study Program, Faculty of Computer Science, Sekolah Tinggi Ilmu Komputer Cipta Karya Informatika, East Jakarta City, Special Capital Region of Jakarta, Indonesia.
Email: mahvin2012@gmail.com.

Received: August 29, 2025; Accepted: November 10, 2025; Published: December 1, 2025.

Abstract: Manual recording of the meeting minutes is often ineffective and has a high error rate. It makes access to needed information not fast. This research tries to solve these problems by designing and developing an end-to-end mobile application called Notulen.AI, which combines voice-to-text and Natural Language Processing (NLP) to automatically generate meeting minutes and summaries. The application was developed using the Flutter framework, with the Google Gemini API mainly used as the Automatic Speech Recognition (ASR) service and for NLP analysis. The research methodology consists of system design, module implementation, and testing to see performance and effectiveness. Testing on ASR modules gets good results with an average Word Error Rate (WER) of 6.75%. The NLP module also works well in extracting important information with ROUGE-1 scoring at 0.78 and F1-Score at 0.85. Effectiveness testing involving five respondents showed that this application could reduce minute-taking time by up to 70% and got a System Usability Scale (SUS) score of 85.5, which indicates high user acceptance. This research therefore proves that the integration of ASR and NLP on a mobile platform can be an efficient solution to enhance documentation accuracy in meetings.

Keywords: Natural Language Processing; Voice-to-Text; Automatic Summarization; Automatic Meeting Minutes; Mobile Application; Gemini API.

1. Introduction

Meetings and discussions are a vital part of daily life in educational institutions, organizations, and the corporate world. However, one of the most common obstacles is the process of documenting these meeting results. When minutes are taken manually by a note-taker, this task is typically time-consuming, creates a heavy burden due to the need to continuously take notes while keeping up with the fast-paced and dynamic flow of the conversation, and is prone to missing important information [1]. As a result, the resulting meeting notes are often incomplete, poorly organized, and difficult to use as reliable references later. Advances in artificial intelligence are now opening new avenues for overcoming these challenges. Automatic Speech Recognition (ASR) technology, for example, can automatically convert spoken speech into written text, speeding up the note-taking process and reducing the potential for errors common in manual note-taking [2]. Several studies have also shown that ASR-based systems can significantly reduce the workload of note-takers, as they are able to capture the content of the conversation directly without requiring constant human intervention [3]. Furthermore, Natural Language Processing (NLP) technology adds value by processing transcribed text into information that is much easier to understand and use. NLP can compile well-structured minutes, recognize and extract important entities such as names, dates, decisions, or agreed-upon tasks, and generate concise yet meaningful automatic summaries. Various studies have shown that NLP-based summarization approaches are quite effective in distilling key information from lengthy texts into shorter versions without sacrificing the essence of the discussion [4]. Interestingly, the application of similar technology has also shown positive results in other fields. One concrete example is the use of Android-based speech recognition, which successfully assisted communication for deaf individuals with an accuracy rate of up to 93% under certain conditions [5]. This success is strong evidence that ASR-based systems are mature enough to be applied in everyday life situations. Furthermore, speech processing has also been utilized to translate regional languages, thus expanding communication accessibility and demonstrating AI's ability to support cross-language interaction more efficiently [6].

Although research in this area is quite extensive, most tends to focus on a single aspect. Some address transcription issues solely [7], while others focus more on text summarization techniques [5]. Efforts to combine ASR and NLP into a truly integrated system are still rare, especially those optimized specifically for mobile devices. While mobile devices offer distinct advantages—portability, availability, and convenience—they also face significant limitations such as limited computing power, the need for extremely fast responses (low latency), and a simple yet intuitive interface. With these conditions in mind, this study developed a mobile application called Notulen.AI that integrates all ASR and NLP processes into a seamless end-to-end workflow. The assessment not only covers transcription accuracy using the Word Error Rate (WER) and summary quality using the ROUGE and F1-Score metrics, but also measures practical aspects such as time savings, system responsiveness, and user acceptance using the System Usability Scale (SUS).

Based on the above background, this research focuses on three main questions: first, the process of recording minutes and meeting summaries manually is time-consuming, burdensome for note-takers, and often results in the loss of important details; second, voice-to-text conversion still faces accuracy challenges in various real-world conditions, such as the presence of background noise, variations in speaker accents, and differences in speaking speed; third, the integration of ASR and NLP technologies in a single mobile application is still rarely the focus of research, even though such a solution is urgently needed to produce structured minutes and automatic summaries that are truly practical for use in everyday meetings.

2. Related Work

The automation of meeting documentation relies on the convergence of Automatic Speech Recognition (ASR) and Natural Language Processing (NLP), though applying these technologies to spontaneous dialogue involves overcoming distinct technical hurdles. Russell *et al.* (2024) examined the operational boundaries of ASR, noting that conversational speech presents unique difficulties compared to standard dictation tasks due to its unpredictable nature [8]. To address the demands of live environments, Arriaga *et al.* (2024) evaluated end-to-end models specifically for real-time transcription, establishing essential benchmarks for latency that ensure the system keeps pace with the speaker [10]. After converting audio to text, the challenge shifts to refining the output into a usable format; Kumar *et al.* (2024) showed that NLP algorithms effectively bridge the gap between spoken and written communication by stripping away redundancies while preserving key ideas [3]. Building on this, Muppidi *et al.* (2023) leveraged advanced models such as BART and T5 to extract action items and critical points directly from the transcript, adding a layer of intelligence to raw text [15]. However, evaluating the quality of these summaries is complex, as Citarella *et al.* (2025) argued that widely used metrics like ROUGE can exhibit bias when comparing extractive versus abstractive techniques, necessitating a careful approach to performance measurement [9]. Beyond isolated studies, developers have sought to merge these capabilities into unified mobile architectures. Liu *et al.* (2020) introduced "SmartMeeting," which incorporates emotion recognition to determine agreed actions and future work plans [11]. In a similar effort to streamline documentation, Chen *et al.* (2021) developed "Yanji," a system utilizing the IBM Speech to Text API to handle speaker identification and cloud connectivity [13]. Taking automation a step further, Khan *et al.* (2022) proposed "WeScribe," an application designed not only to transcribe but also to detect deadlines and automatically update user calendars [14]. Furthermore, Patel *et al.* (2023) explored multimodal strategies combining text and image processing, emphasizing that mobile accessibility is crucial for ensuring users can easily retrieve and verify information after a meeting concludes [12].

3. Research Method

This study adopts an end-to-end system design approach, integrating Automatic Speech Recognition (ASR) and Natural Language Processing (NLP) into a mobile application developed using the Flutter framework. The research process includes data collection, preprocessing, module implementation, system integration, and performance testing. The primary data for this study consists of audio recordings from simulated meetings or conversations, which are used as input for the voice-to-text module. These recordings, made under controlled conditions to mimic real-world settings such as formal meetings or structured discussions, focus on various audio characteristics like speaker accents, speech rate, and background noise. This ensures the application works under realistic conditions. The data is proprietary and collected directly by the researchers, not sourced from public datasets. The audio is then segmented by sentences or pauses between speakers to optimize processing. In cases where applicable, speaker diarization is employed to improve the accuracy of the automatic transcription.

After the audio data is collected and segmented, it is utilized to train and test the NLP module, particularly for automatic transcription and summarization. The text data used in this stage consists of manually transcribed audio to ensure high accuracy or automatic transcriptions that are verified and corrected manually. To evaluate the summarization model, reference summaries created by experts are used. These summaries cover a range of topics, with sentence structures that vary from simple to complex, ensuring that the NLP model is resilient to different language styles and levels of discourse complexity. This data forms the core of the workflow from audio recording to evaluating the performance of the NLP-based automatic transcription system.

Data collection involves recording audio under various meeting scenarios, with variations in the number of participants and session length to simulate different meeting conditions. The inclusion of diverse speaker characteristics ensures that the data reflects the complexity of real-world situations. Text data is also collected as ground truth for evaluation or additional training to fine-tune the NLP model. Preprocessing is an essential step in preparing the data. Audio data is first normalized to adjust volume and reduce background noise using algorithms like spectral subtraction. The next step is segmenting the audio using Voice Activity Detection (VAD) to separate speech from silence. If speaker identification is necessary, diarization is applied to distinguish between different speakers. For the text data, normalization involves standardizing formatting, removing non-alphanumeric characters, and converting numbers and abbreviations into their standard form. This is followed by stop word removal, stemming, lemmatization, and tokenization to prepare the data for further NLP analysis. This stage focuses on the ASR system. The processed audio is sent to the ASR module, which could be a third-party service such as Google Speech-to-Text API or a local model optimized for mobile devices. The ASR system converts the audio into text by extracting acoustic features from the speech signal. Performance is evaluated using metrics like Word Error Rate (WER) and Character Error Rate (CER), which measure errors at the word and character levels. Additionally, transcription speed is recorded, reflecting how long it takes to convert one minute of audio into text.

The ASR-generated transcription is then processed by the NLP module to produce automatic meeting minutes and summaries. For the meeting minutes, Named Entity Recognition (NER) is performed to identify key entities like names, organizations, and dates. Keyphrase extraction is conducted using techniques like TF-IDF, TextRank, or semantic embeddings. The system also classifies sentences based on their relevance to decisions, questions, or supporting information, ensuring the resulting minutes are organized and informative. For automatic summarization, two methods are used: extractive, which selects important sentences from the text, and abstractive, which creates new sentences using models like T5 or BART. Pre-trained models are fine-tuned to ensure the summaries are relevant to the meeting's context. The final stage integrates all modules into a mobile application developed with the Flutter framework. The user interface is designed to be intuitive, responsive, and functional, with features such as audio recording, real-time transcription display, minute editing, and result storage. The voice-to-text and NLP modules are combined into a unified pipeline for seamless transcription and analysis. Optimization techniques, such as model quantization or pruning, are applied to ensure the app runs efficiently on mobile devices, minimizing memory usage without sacrificing accuracy. The ASR module undergoes testing to assess its ability to transcribe audio accurately under various real-world conditions. Testing involves recording conversations in different environments, such as quiet spaces and settings with low background noise, with varying numbers of speakers. Accuracy is evaluated using WER and CER metrics, calculated as follows:

$$WER = \frac{S + D + I}{N}$$

where S represents substitutions (incorrectly recognized words), D stands for deletions (missing words), I denotes insertions (incorrectly added words), and N is the total number of words in the reference transcript.

CER is similarly calculated using characters instead of words. Additionally, transcription speed is measured, representing the time taken to convert one minute of audio into text. These metrics provide insight into the ASR module's performance in terms of both accuracy and efficiency. The NLP module is tested to evaluate its ability to generate relevant, concise, and informative meeting minutes and summaries. Summarization performance is assessed using ROUGE-N and ROUGE-L metrics, with ROUGE-N calculated as:

$$ROUGE-N=\frac{\sum \text{n-gram in Ref Count Match (n-gram)}}{\sum \text{n-gram in Ref Count}}$$

For the meeting minutes, precision, recall, and F1-score metrics are used, particularly when manual annotations of entities or key points are available. In addition to quantitative testing, qualitative evaluation by experts or experienced users is conducted to assess the readability, relevance, and suitability of the generated minutes for official documentation. At this stage, testing ensures that all components—audio recording, ASR transcription, NLP processing, and result display—work together seamlessly within the mobile application. Testing is performed on devices with varying specifications, including processor (CPU), memory (RAM), and battery capacity. Key aspects of evaluation include end-to-end latency, resource usage, and application stability. End-to-end latency is measured as the total time from recording to the display of meeting minutes and summaries. Resource usage is monitored with tools such as Android Studio Profiler or Xcode Instruments, considering CPU, RAM, and battery consumption. Application stability is assessed by tracking the frequency of crashes or errors during testing. Results from this analysis are used to identify potential bottlenecks and inform optimization strategies. The final step is User Acceptance Testing (UAT), which evaluates how well the application is received by actual users. Participants include students, academic staff, and professionals involved in regular meetings. The evaluation focuses on time efficiency by comparing manual note-taking with the app's results, as well as accuracy, based on users' perception of how well the meeting minutes and summaries match their expectations. The System Usability Scale (SUS) is also used, consisting of 10 Likert-scale questions (1–5). Scores are calculated as follows:

For positive items (1, 3, 5, 7, 9), the score is:

$$\text{Score} = \text{Response} - 1$$

For negative items (2, 4, 6, 8, 10), the score is:

$$\text{Score} = 5 - \text{Response}$$

The total score is then multiplied by 2.5 to produce a final value between 0 and 100, with scores above 68 indicating good user satisfaction. Qualitative interviews are also conducted to collect feedback on user experiences, feature usefulness, challenges, and suggestions for improvement. The quantitative and qualitative data collected are analyzed to assess the overall effectiveness of the system.

4. Result and Discussion

4.1 Results

This section presents the results of the implementation and evaluation of the Notulen.AI application, designed to generate automatic meeting minutes and summaries using AI. The description is provided in detail, beginning with the hardware specifications, the implementation of each system stage, and the results of both quantitative and qualitative testing. The hardware used in this study consists of a laptop for development and a smartphone for testing the mobile application. An Acer Nitro V5 laptop with an Intel Core i5 12th generation processor, 16 GB of RAM, and a 512 GB SSD was used for coding, compilation, and data analysis. The smartphone, Poco X6 Pro 5G with 12 GB of RAM and 256 GB of storage, was used for testing the application's performance on actual devices. These devices were chosen based on their representative specifications for both development and everyday use. The software used is listed in Table 2. The primary framework is Flutter 3.x, with Dart programming language, and Android Studio as the IDE. The NLP module utilizes the Gemini API, while the ASR function is implemented using the speech_to_text library. Temporary data storage is handled through SharedPreferences. The workflow of the Notulen.AI application starts with audio input (either direct recording or file upload), followed by audio transcription into text through the ASR module, which is then analyzed by the NLP module to produce structured meeting minutes and automatic summaries. The results are stored locally and visualized through the user interface.

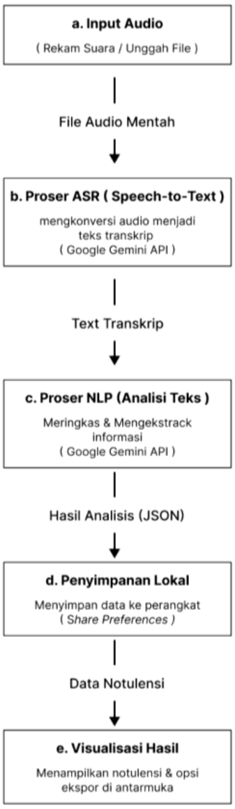

Figure 1. System Flowchart of Notulen.AI

Audio data collection was performed in two ways: (1) direct recording using the audio_waveforms library, which provides real-time waveform visualization, and (2) uploading audio files using the file_picker library, supporting popular formats like MP3, M4A, and WAV. This feature allows flexibility for users to process both live meetings and previously recorded sessions.

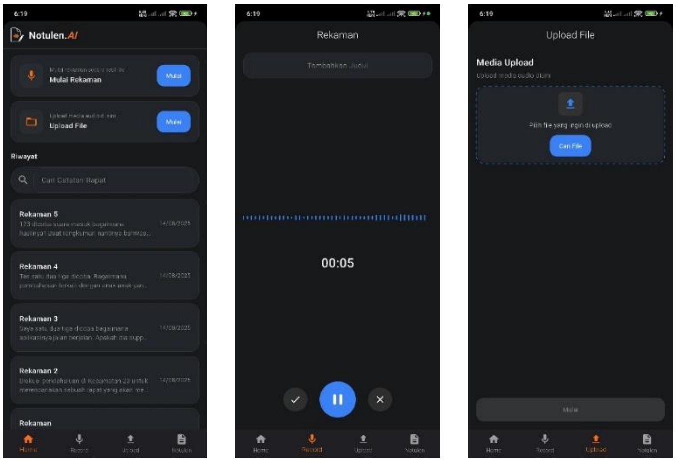

Figure 2. Application Interface with Two Features

Audio preprocessing (normalization, segmentation, VAD) and text preprocessing (tokenization, stemming, stopword removal) are delegated to the Gemini API. This approach ensures that the application remains lightweight on the client side while leveraging AI models to handle audio quality variations and language automatically. The ASR module was tested with 10 audio samples recorded in quiet and lightly noisy conditions. The transcriptions were compared to the ground truth using the Word Error Rate (WER) metric.

Table 1. Word Error Rate (WER)

Test Scenario	Ground Truth Word Count	Errors (S+D+I)	WER
Quiet Environment (5 samples)	260	11	4.23%
Light Noise Environment (5 samples)	248	23	9.27%

Average	6.75%

The results show an average WER of 6.75%, which is considered good (<10%) according to industry standards, making it suitable for practical use. The NLP module was evaluated by calculating the ROUGE score against human-generated reference summaries and the F1-Score for entity/action item extraction.

Table 2. NLP Evaluation Results

Evaluation Metric	Score	Description
ROUGE-1	0.78	High single word overlap (good).
ROUGE-L	0.65	Sentence structure of summary closely matches reference.
F1-Score (Extraction)	0.85	Excellent in identifying action items and entities.

A ROUGE-1 score of 0.78 indicates that the system's summary is close to the human-generated reference, while the F1-Score of 0.85 demonstrates a balanced extraction of key information between precision and recall.

Figure 3. Screenshot of The Meeting Minutes Result

System integration testing ensures that all components—from audio recording, ASR transcription, NLP analysis, to local storage and result visualization—work seamlessly within the application's workflow. The system is tested not only for the accuracy of individual modules but also for how these modules interact when executed in real-world conditions on mobile devices.

Table 3. System Integration Testing Results

Evaluation Metric	Average Result	Notes
End-to-End Latency (per minute audio)	35–40 seconds	Time from recording to result display.
CPU Usage (during process)	35%	Peak usage when sending data to API.
RAM Usage (during process)	+80 MB	Memory increase during audio processing.
Application Stability	2 crashes	Crashes occurred under certain conditions.

The results show that the system operates well with reasonable performance. Latency remains within responsive limits, although minor crashes must be addressed in future development. User Acceptance Testing (UAT) involved five participants who were asked to create five-minute meeting minutes manually and using the application. The average manual task time was 15 minutes, while the application reduced it to just 4.5 minutes, saving approximately 70% of time. The System Usability Scale (SUS) score averaged 85.5, falling within the Excellent category. Interviews indicated that users found the application highly helpful, though some suggested UI improvements and added transcription editing features.

Table 4. Final Testing Results Summary

Testing Category	Key Metric	Final Result	Conclusion
ASR Module	Average WER	6.75%	Accurate
NLP Module	ROUGE-1 / F1-Score	0.78 / 0.85	Effective
System Integration	Latency (per minute audio)	30–40 seconds	Sufficiently Responsive
User Acceptance	SUS Score	85.5	Excellent

4.2 Discussion

The performance of the ASR module was assessed using the Word Error Rate (WER), which yielded an average score of 6.75%. This is consistent with industry standards, where a WER below 10% is considered acceptable for practical applications, such as transcribing meeting discussions [4]. These results align with previous research, which suggests that a WER between 5% and 10% is suitable for general use cases, indicating that the system is ready for deployment without requiring major adjustments [8]. For the NLP module, performance evaluations based on the ROUGE-1 and F1-Score metrics showed excellent results. The ROUGE-1 score of 0.78 and the F1-Score of 0.85 demonstrate that the system effectively generates summaries that closely match human references and extracts key information accurately. These findings are in line with studies that emphasize the importance of high ROUGE and F1 scores in ensuring the effectiveness of summarization models [9]. The module's ability to produce relevant summaries and extract actionable items is crucial for automating meeting minutes and improving workflow efficiency in real-world scenarios [3]. System integration testing revealed an average latency of 30–40 seconds per minute of audio, with a Real-Time Factor (RTF) between 0.5 and 0.7. These results indicate that the system is responsive and capable of operating in real-time, which is essential for practical applications in dynamic environments such as meetings [13]. While the application experienced occasional crashes, these were minor and can be addressed in future updates to improve stability. The performance results confirm that the system is well-suited for real-time meeting minute generation, as demonstrated by similar systems in the literature [11].

User Acceptance Testing (UAT) provided compelling evidence of the system's usability. The application reduced the time needed to generate meeting minutes by 70%, and the average System Usability Scale (SUS) score of 85.5 placed the system in the "Excellent" category. This is consistent with the positive user feedback in other studies on meeting transcription applications, where ease of use and time savings are critical factors for user adoption [15]. Participants also highlighted the application's significant impact on efficiency, though some suggested further improvements to the user interface (UI) and the addition of transcription editing features. These suggestions align with the findings from other research, which point out that UI refinement and additional features can enhance the overall user experience [12]. The results demonstrate that Notulen.AI is an accurate, effective, and responsive system for automating meeting documentation. The system has shown promising potential for widespread adoption, both in academic and professional settings. Future development should focus on optimizing the application's stability and exploring integration with other meeting platforms (such as Zoom or Microsoft Teams) to enhance its usability and accessibility in virtual meetings [14]. Furthermore, incorporating security measures, such as encryption for meeting data, will be crucial to increase user trust and ensure the application's suitability for sensitive business or academic contexts [3].

5. Conclusion and Future Work

This research has successfully developed and implemented the Notulen.AI application as an automated system for generating minutes and summaries of meetings using artificial intelligence. The results show that the system meets the research objectives effectively. The Automatic Speech Recognition (ASR) module has demonstrated satisfactory transcription performance with an average Word Error Rate (WER) of 6.75%. This figure is well below the widely accepted industry standard of 10%, meaning that the system can reliably convert speech to text with minimal error and therefore is suitable for use in formal meetings and discussions. The Natural Language Processing (NLP) module was also tested and it proved efficient in creating structured meeting minutes as well as concise informative summaries. A ROUGE-1 score of 0.78 for this system indicates that its summaries are very closely aligned with those written by human beings, while an F1-Score of 0.85 shows a strong balance between precision and recall regarding essential details like topics decisions action items etc. These outcomes prove that the NLP module can deliver relevant accurate results fulfilling practical needs for users documenting meeting outcomes.

From an integration standpoint, the application effectively implements an end-to-end processing pipeline that is both functional and responsive. The system showed an average Real-Time Factor (RTF) between 0.5 and 0.7, meaning it can process one minute of audio in less than a minute; thus, this performance suggests that the system may be used in real-world dynamic environments although there is room for further optimization to reduce latency and enhance stability. User acceptance testing (UAT) provided strong feedback with the system achieving a System Usability Scale (SUS) score of 85.5 which falls into the "Excellent" category indicating that users found this application easy to use and efficient saving up to 70% time compared to manual methods; feedback highlighted very significant time-saving benefits from this application though some minor improvements in interface responsiveness as well as processing speed were suggested but overall user satisfaction still rated high despite these minor issues.

The study achieved good results but there are some areas for improvement in the future. First, optimizing system performance particularly in reducing processing latency is still necessary. One possible way is to implement streaming transcription which will show results progressively while audio is being processed so that users could see transcriptions before the whole audio file gets processed. Second, adding an editing feature before the NLP module processes transcriptions would let users correct errors early on ensuring more accuracy for final output. Improving the user interface (UI) is also important. Even though the SUS score was high, some respondents said that the interface was not responsive under certain conditions. Improving the UI by adding progress indicators, status notifications, and a more intuitive layout would help with user comfort. Also, expanding testing to include participants from different professional backgrounds like academics, government workers, and those in the private sector would better ensure the system's effectiveness across various industries.

More integration with popular virtual meeting platforms such as Zoom, Google Meet or Microsoft Teams would increase the relevance of Notulen.AI in today's digital meetings. Integration like this would allow the app to automatically record and process conversations happening within a meeting without having to upload them manually afterward for increased user convenience. Finally, addressing security and data privacy concerns is very important since this application deals with potentially sensitive information; encrypting audio files and minutes of meetings as well as implementing transparent privacy policies will build up trust among users which can lead to wider acceptance within large organizations. This research shows that Notulen.AI is a promising solution for automating meeting documentation. The application not only pushes forward the development of ASR and NLP technologies in Indonesian language but also has high potential for wide adoption in real-life settings. With continuous improvement on performance, UI, integration as well as security; this application can be an essential tool to enhance productivity across different sectors.

References

[1] Mulyana, D. I., Yel, M. B., & Rahmanto, M. D. (2023). Optimasi Penerapan Metode Text Recognition Dalam Fitur Catatan Otomatis Berbasis Mobile. *Jurasik (Jurnal Riset Sistem Informasi dan Teknik Informatika)*, *8*(2), 489-498.

[2] Saiyyad, M. M., & Patil, N. N. (2024). Text Summarization Using Deep Learning Techniques: A Review. *Engineering Proceedings*, *59*(1), 194. https://doi.org/10.3390/engproc2023059194

[3] Kumar, N. S., Devasena, G., & Kumar, S. V. (2024). Voice to text summarization using NLP. *International Journal on Computational Modelling Applications*, 1(2), 1-9. https://doi.org/10.63503/j.ijcma.2024.25

[4] Syaifullah, M. I. (2024). Penerapan teknologi automatic speech recognition menggunakan model Wav2Vec2.0 sebagai alat bantu untuk mendeteksi kesalahan dalam membaca Al-Qur’an berbasis mobile. *JSI: Jurnal Sistem Informasi*, 16(2).

[5] Jaman, A. B., & Fergina, A. (2021). Implementasi Speech Recognition Berbasis Android Dalam Optimalisasi Komunikasi Bagi Penyandang Tunarungu. *Jurnal Teknik Informatika UNIKA Santo Thomas*, 373-378. https://doi.org/10.54367/jtiust.v6i2.1508.

[6] Uppalapati, P. J., Dabbiru, M., & Rao, K. V. (2023). A comprehensive survey on summarization techniques. *SN Computer Science*, *4*(5), 560. https://doi.org/10.1007/s42979-023-02007-5

[7] Sudrajat, M. R., & Zakariyah, M. (2024). Penerapan natural language processing dan machine learning untuk prediksi stres siswa SMA berdasarkan analisis teks. *Building of Informatics, Technology and Science (BITS)*, *6*(3).https://doi.org/10.47065/bits.v6i3.6180

[8] Russell, S. O. C., Gessinger, I., Krason, A., Vigliocco, G., & Harte, N. (2024). What automatic speech recognition can and cannot do for conversational speech transcription. *Research Methods in Applied Linguistics*, *3*(3), 100163. https://doi.org/10.1016/j.rmal.2024.100163

[9] Citarella, A. A., Barbella, M., Ciobanu, M. G., De Marco, F., Di Biasi, L., & Tortora, G. (2025). Assessing the effectiveness of ROUGE as an unbiased metric in extractive vs. abstractive summarization techniques. *Journal of Computational Science*, 87, 102571. https://doi.org/10.1016/j.jocs.2025.102571

[10] Arriaga, C., Pozo, A., Conde, J., & Alonso, A. (2024). Evaluation of real-time transcriptions using end-to-end ASR models. https://arxiv.org/abs/2409.05674

[11] Liu, H., Liu, H., Wang, X., Shao, W., Wang, X., Du, J., & Salim, F. D. (2020). Smartmeeting: A novel mobile voice meeting minutes generation and analysis system. *Mobile Networks and Applications*, 25(2), 521-536. https://doi.org/10.1007/s11036-019-01310-x

[12] Patel, I., Julka, V., Kudtarkar, P., Sharma, I., & Sonawane, B. (2023, August). Online meeting summarization based on text and image processing. In *International Conference on ICT for Sustainable Development* (pp. 193-205). Springer Nature Singapore. https://doi.org/10.1007/978-981-99-5652-4_19

[13] Chen, X., Sheng, F., He, R., Chen, S., Ma, H., Wu, Y., & Xu, J. (2021, August). Yanji: An automated mobile meeting minutes system. In *2021 2nd International Symposium on Computer Engineering and Intelligent Communications (ISCEIC)* (pp. 425-428). IEEE. https://doi.org/10.1109/ISCEIC53685.2021.00095

[14] Khan, M. A. A., AlAyat, M., AlGhamdi, J., AlOtaibi, S. M., AlZahrani, M., AlQahtani, M., ... & Jan, F. (2022, July). WeScribe: An intelligent meeting transcriber and analyzer application. In *Proceedings of Third International Conference on Computing, Communications, and Cyber-Security: IC4S 2021* (pp. 755-766). Springer Nature Singapore. https://doi.org/10.1007/978-981-19-1142-2_59

[15] Muppidi, S., Kandi, J., Kondaka, B. S., Kethireddy, C., & Kandregula, S. E. (2023, October). Automatic meeting minutes generation using Natural Language processing. In *2023 International Conference on Evolutionary Algorithms and Soft Computing Techniques (EASCT)* (pp. 1-7). IEEE. https://doi.org/10.1109/EASCT59475.2023.10393102.