Voice-based To-Do-List using Gen-AI

Harsh Telure[1], Samuel Devadass[2], Arjun Thokare[3], Hrushikesh Sawant[4], Dr. Chaitali Shewale[5]
[1,2,3,4] B.Tech Information Technology BRACT'S Vishwakarma Institute of Information Technology Pune,India
[5]Department of Information Technology BRACT'S Vishwakarma Institute of Information Technology Pune,India

ABSTRACT – The importance of meetings has become such an integral part of operations activities for a better vision of collaboration, decision-making, and solving problems. However, this leads to messy notes left for the participants and a long list of things to do in order to organize and prioritize. The management of the aftermaths of meetings, which comprises summarizing and assigning responsibilities, sometimes becomes the way to experience inefficiency in some aspects, missed deadlines, and poorly managed tasks. This paper seeks to voice a novel solution using advanced Gen-AI and NLP solutions to automatically generate to-do lists from the transcripts of meetings, thereby helping improve general post-meeting work processes and productivity. At its core are speech recognition and NLP models combined with large language models such as Groq, intended to convert the discussions of meetings into neatly organized tasks with defined responsibilities and deadlines. As such, the system does not use manual transcription and task allocations, which can be associated with human errors. This would save a few valuable minutes, as it will allow employees to spend this time on some more essential issues relative to their job.

The core of the system is speech-to-text technology, which helps it capture all the spoken dialogue during a meeting and then translate it into texts.This text is then processed by the NLP component to extract key tasks, decisions, and other action items. The integration of LLMs ensures that the extracted information is synthesized into coherent, actionable to-do lists. Additionally, the system provides the capability to automatically dispatch these lists to relevant participants via email, further improving efficiency.

The text then goes through the NLP component to help draw out the relevant tasks, decisions, and other action items. The ability provided by LLM in this work ensures that the extracted information is synthesized into coherent actionable to-do lists. Additionally, automatically sending those lists to relevant participants via email provides further capability for improvement in efficiency. This paper evaluates the effectiveness of the system in real-life by finding how it has performed in terms of accuracy of the task, user satisfaction and time-saving. Further, the paper has analyzed the broader impact of using AI-driven systems for managing tasks where those technologies improve the productivity of organizations through results where meetings are leading toward clearly defined and actionable outcomes.

Enabling the automation of certain elements of the processes that follow a meeting, organizations are better able to deal with workload, create responsibility and aid better communication among the team members. The system is quite encouraging, but certain components still need to be worked on for example improving the noisy environment speech recognition and the complexities of the AI federal instructions system The present study also underlines what is shown in the literature, that they need to encourage future research and development of the system concerning the use of AI in everyday business activities and its enhancement. It is possible to state that these systems can change the paradigm of conducting meetings and organizing tasks in the organization to make it even more effective and efficient, as long as the development and improvement processes are cyclic.

I.INTRODUCTION

In today's business environments, meetings are essential for interaction, cooperation, and most importantly, resolving issues among team members. Planning, brainstorming, or problem-solving, meetings are the means of communication and delegation of tasks that facilitate any work. Even though meetings are very important, there is often a lack of structure and organization when it comes to tasks and commitments that follow as a result of the meetings. Due to this, the staff may not be clear of their responsibilities, some may be overlooked or missed, and deadlines may be ignored or not met. This non-threatening ineffectiveness is equally destructive to the output of the organization and workplace morale as well, causing uneasiness and stagnation in the productivity of the workers. [1]

With the recent developments in technology and the access to a large volume of data within a highly compressed working period, the amount of information shared in the meetings has tremendously increased, making it more difficult to arrange and act on the information. However, the conventional manual ways of summarizing meetings and delegating tasks has been ineffective, especially in situations characterized by complicated conversations and several participants. In addition, the use of recollection and note taking creates an opportunity of imperfections and omissions that can be detrimental to the timely delivery and success of the project. [1]

Recent developments in Artificial Intelligence (AI) are encouraging. Such technologies as Natural Language Processing (NLP) and Large Language Models (LLM) allow for the quick processing and evaluation of massive amounts of text, which facilitates the identification of important information and actionable items within unstructured content. Likewise, the capacity for speech recognition has reached the level where a person would be able to speak, and all his/her words would be recorded in a written form without any discrepancies, paving the way for automating the processes that come after the end of the meeting. This paper proposes an original system utilizing AI, NLP and Speech recognition technologies which helps in the generation of a to-do list from meeting transcripts automatically. With the help of LLM's such as Groq and advanced speech to text Software, the system is able to transform the arguments that took place during the meeting in a systematic manner that is converted to a task list. These lists have task details, timelines, and assignments ensuring every participant knows what is expected of them and they can do it.[2]

The main objective of the system is to alleviate the hassle of performing work related to task management after the meetings. Rather than depending on the participants to collect the meeting minutes and allocate work manually, the system facilitates in-house processing of audio recordings, identifying actionable items and putting them up in the right order. This approach is not only time-efficient, but also eliminates ambiguities about how the tasks should be done, thus decreasing chances of miscommunication or omitting any actions such as the ones assigned to people. The use of artificial intelligence in enhancing task management system, in relation to meetings, is able to overcome a recurrent problem of forgetting or postponing execution of certain activities due to excess of information. By alleviating the task of extraction and assignment of tasks, the system makes captures preparation or assignment of all essential tasks to concerned persons. Also, the module allows the system to email the participants an action item list for this purpose enhancing follow up on the assigned tasks. [2]
More than just offering improvements for task management, the system is seen to have ulterior wideness in its applications to organizational productivity. By creating a less cumbersome, or better, way of dealing with post meeting loads, the system permits the employees to divert attention to more pressing issues thus leading to proper and efficient workflow management. On top of that, the application of AI in such a case is a clear picture of how modern inventions can be put into rationale use towards addressing organizational needs thereby creating a cycle of task automation advancement.

In the rest of this paper, the system's architecture and design will be provided, the system will be evaluated in its operational environment, and the directions of its future development will be discussed. It is the objective of this analysis to make a case for the advantages of AI oriented tools in the management of such activities as organizational meetings.

II. LITERATURE SURVEY

Santosh Gaikwad et al. in their survey titled "A review on Speech Recognition Technique," describe speech recognition as a specialized problem of pattern matching. [3]
Speech has been classified into four main types: continuous speech, spontaneous speech, connected speech, and isolated words. The machine is able to recognize isolated words due to the fact that they are considered single units. A typical formation of two or more words that have a high tendency of occurrence together is referred to as a connected word. Continuous speech is the formal communication humans carry out at important events or meetings. Spontaneous speech, on the other hand, is the kind of speech which we use in our daily conversations and talks. Out of all the four classes, spontaneous speech is the most difficult to match and map to its corresponding textual version. Based on this classification, the recognition task is also divided into four stages, analysis, feature extraction, modelling and testing.

The analysis process can be broken down into speech analysis, segmentation analysis, sub-segmental analysis, and supra-segmental analysis. Each of these levels of analysis corresponds to some speech patterns and ways of dividing the speech into different divisions. The model analyses all this information in preparatory for the next step. The next stage is featuring extraction, where the model learns to extract certain features from the input signals, or the speech it receives. [3]

Techniques like Principal Component Analysis (PCA), Independent Component Analysis (ICA) and

RASTA filtering, among several others, are most commonly used for this purpose. The following stage is where the machine learns to mimic or model the speaker's pattern. Approaches like dynamic time warping or artificial intelligence-based approaches are employed in this stage. The final stage is testing or word matching, which is more like a validation stage where the identified word is compared with the known word. The engine tries to match whole words or sub words to the set of known words, and is validated accordingly. Based on the above findings, their conclusion is that we can improve the system's performance and precision either by using the word error rate, or the word recognition rate as a measure of the system's accuracy, and adjust the parameters accordingly to obtain best results for our specific applications. [3]

Daniel Povey et al. in their research titled, "The Kaldi Speech Recognition Toolkit," have designed and deployed an open-source speech recognition engine. [4] This tool, called Kaldi, was designed as a speech recognition model with fixed state transducers from OpenFst, and supports modelling of arbitrary phonetic context sizes and Gaussian mixture models. It is written in pure C++, allowing it to run on almost all computing devices. Their feature extraction approach creates uniform MFCC (Mel-Frequency Cepstral Coefficient) and PLP (Perceptual Linear Predictive) features from the given input signals. This allows the users to fine tune their engine to make it suitable to their requirements. The use of VTLN and cepstral mean, along with LDA and HLDA grants a unique ease in recognizing patterns based on various speech forms, and aids in the overall feature extraction stage.[4]

It models Gaussian Mixture Models (GMMs) and Subspace Gaussian Mixture Models (SGMMs), so that all types of acoustic modelling occur with ease. The use of Phonetic Decision Trees improves the validation stage, so the model is able to validate itself at a comparably faster rate as opposed to other models. The toolkit also produces a decoding graph, allowing the user to know exactly where the performance of the model is lagging behind, and work on ways to improve its accuracy. Kaldi is still under further research at Microsoft and Stanford University, allowing users to make the best out of the most recent versions as they arise. [4]

Wiqas Ghai and Navdeep Singh in their review titled, "A Literature Review on Automatic Speech Recognition," have discussed about the structure of various speech recognition models, as well as how artificial intelligence can be used to leverage their accuracy as well as overall performance. [5]

Their theory is that artificial neural networks can be used to improve the performance of such systems. To be specific, Artificial Neural Network-Hidden Markov Model (ANNHMM) seems to be the most promising model to grant this ability, especially to large vocabulary systems.

Support Vector Machines turned out to be the most powerful classifiers for recognizing patterns. However, SVMs work in an optimized way only for small to medium sized databases. For larger databases, they require expensive computational cost, which is not feasible for day-to-day applications. Making use of the Hidden Markov Model for every individual element of the vocabulary of the task increases the system performance. The use of CNMF (Convolutive Non-Negative Matrix Factorization approach) was also able to remove a significant amount of background noise from the input signal. Finetuning the neural network's parameters based on the environment provides higher accuracy, and also improves the WER as well. [5]

Karpagavalli S and Chandra E in their study titled, "A review on Automatic Speech Recognition Architecture and Approaches," have used a unique way of recognizing speech. [6] They have used special methods and techniques of signal processing to extract text from the words uttered to the model. They have used the probability theory to map the word spoken by the user to the database of known words. It basically calculates the probability of the input word against all words with probable matches. If the match rate is above a predefined threshold, the word is said to be recognized. The user first speaks to the system. The system then captures the input signal and removes any background noise from it. Then it enters a pre-emphasis stage, followed by frame blocking and windowing and other signal processing stages. After all the signal processing stages are done, the input data is passed on to the feature extraction and the modelling stages. [6]

In this context, the characteristics are obtained by using tools like the acoustic-phonetic approach, pattern recognition approach, artificial intelligence approach, or the generative learning approach. In order to guarantee that the word is recognized correctly, a special type of Hidden Markov Model (HMM) which utilizes Gaussian Mixture Models (GMM) is employed in the generative learning approach. After this the probability of the word is calculated and the most suitable word is returned. This turns out to be the most accurate method, even comparable to artificial intelligence-based approaches, making this method the most promising one in speech recognition purely based on mathematics without use of any artificial intelligence algorithms [6].

N. Morgan and H. Bourlard in their paper titled, "Continuous speech recognition," have detailed a unique blend of artificial intelligence and Hidden Markov Models to be able to correctly recognize continuous speech used in daily life. [7] In this hybrid approach, they have successfully integrated a statistical language model with a discriminative acoustic model for a large vocabulary-based speech recognition system. This also helped them achieve higher accuracy compared to existing systems, and could also be easily incorporated within larger systems without much complexity. [7]

Recently, it is noticed that large language models (LLMs) and generative pre-trained transformers (GPTs) are showing performance that has never been witnessed before and can do much better than the traditional approach used by most systems, typical in the vast majority of algorithms [8]. Such advanced algorithms have paved the way to new frontiers in NLP.

The ability of LLMs to understand and generate human-like language is one of the greatest merits that makes it possible to more naturalize, for instance, the interaction with chatbots and virtual assistants. Unlike classic rule-based systems built around predefined responses, LLMs can now provide contextually relevant answers, which will take nuances in language, including idioms, tone, and cultural references.[8]

Such algorithms may make the current systems work to nearly unbelievable limits [9]. For example, an organizational integration of LLMs with the affairs that keep it in touch with its customers may make the chatbots more interactive with the former and efficient in answering the queries of the latter. This will highly ensure the satisfaction of the customers while at the same time reducing the levels of operational costs.

Additionally, the flexibility of LLMs enables them to be fine-tuned for specific industries or applications. This tailoring, therefore, allows organizations to solve specific problems in a very effective way, opening doors for innovative solutions previously deemed unattainable.[9]

In a nutshell, the convergence of the large language models and generative pre-trained transformers sign the beginning of a new era in artificial intelligence as the prospect of revolutionizing diverse industries is possible because capability and effectiveness are improved across the board.[9]

Gin der Wu et al, in their research paper, put a critical requirement on feature extraction in speech recognition [10]. They argue that without good quality feature extraction, it may be difficult to identify and model human speech sounds appropriately. Among the several techniques, they emphasize the superiority of Mel Frequency Cepstral Coefficients (MFCCs) over Linear Predictive Coefficient Cepstrum (LPCC) noting that MFCCs offer more relevant and useful representations of audio signals. This makes all the difference since it directly influences speech recognition performance.

Furthermore, Wu et al. have made specific emphasis on the role which the Fast Fourier Transform plays in computing MFCCs, especially because FFImay be easily transformed from time-domain signals to the frequency domain. In this transformation, one also gains a function that could be used to capture the spectral features of speech that are required to be extracted suitably. [10]

The authors further point out the pipeline structures and how this improves overall system performance. The pipeline structures simply organize processing steps in a clear sequence to ensure efficient flows of data with appropriate management of other system resources to achieve better accuracy and speed in speech recognition tasks [10].

C. J. Leggetter and P. C. Woodland suggest a new approach in which HMMs can adjust with new speakers. Their technique uses linear regression to update the model, which advances its competency extensively for speech recognition by new users. It is useful for real-time applications where speaker's characteristics may change due to problems in accurate speech recognition.

Leggetter and Woodland improved the recognition accuracy by refining the model through linear regression. Two things were thus done: the problems of wrongful recognition of spelling errors had decreased, but it reduced the problems of wrongful recognition of spelling errors. It has greatly contributed to the reliability of the user experience, since the system becomes more comfortable about handling different types of speech inputs.[11]

In addition, the authors explain that this method maximizes chances of the model's successful deployment in multiple contexts, thus making it more versatile and applicable in a wide variety of environments. On a general level, their discovery speaks to the significance of adaptive modeling techniques in pushing forward speech recognition systems, and their usability generally across a wider scope of applications [11].

Deep neural networks are new powerful tools for speech recognition, that surpass significantly the traditional approaches [12]. In the work described in [12], a specialized model is described which employs the robust combination strategy, suited to utilize both Deep Neural Network posterior probabilities and word lattices. Therefore, it evolves an audio-sensitive perceptron or input neuron that is specifically designed to differentiate and recognize user speech much more accurately.[12]

The results of this study are well-worthy as the proposed strategy of combination achieved average word relative error rates of less than 7 percent posteriors and less than 9 percent lattices. These results reflect the multi-stream approach; it shows to be superior than speech recognition tasks and has been appropriate in handling any kind of task like this. This method denotes the use of different data sources and fine tunes the model to identify increased sensitivity towards audio inputs, thus being a tremendous step in this direction and creating an avenue for even more accurate and dependable speech recognition systems [13].

The perceptual linear predictive technique [14] is another interesting method used for speech recognition. The three concepts facilitating its effectiveness were critical band spectral resolution, equal loudness curves, and the intensity-loudness power law. Combining all these principles together, PLP analysis appears to be an even more detailed model in relation to human speech as compared to convolutional linear predictive analysis.

The use of very large language models and neural networks can further amplify advances in PLP. Combining the perceptual insights provided with advanced capabilities available in neural architectures, systems can yield better speech recognition performance that is more accurate and robust. This synergy also renders it powerful in extracting the complexities of human speech and positions it to adapt effectively across diverse acoustic environments. In general, the PLP method marks one of the major breakthroughs in the pursuit of new advanced speech recognition technologies [14].

The other innovation is the assimilation of connectionist networks into Hidden Markov Models (HMMs) for speech recognition systems [15]. This hybrid methodology allows a connectionist network to be used as the probability estimator in the framework of HMM. Therefore, the strengths of both models will enhance the performance of the overall system. [15]

This integrated approach has been tested on the speaker-independent DARPA Resource Management database. The results show that the hybrid system highly outperforms traditional HMMs and standalone connectionist networks. This enhancing performance indicates the relevance of combining these two methodologies for producing models with more robustness and accuracy in speech recognition, paving the way towards better applications in various environments. Hybrid models can use advantages from either technique and thus contribute to a step forward in speech recognition technology.

Apart from these advances in speech recognition technology, proper organization of task documents is as important in facilitating user interaction and allowing task completion with minimal hassle. It is here that the large language models come into the picture [16].

More generally, the power of generative AI and LLMs is immense, as it allows one to expand and personalize documents according to the best needs of the user. The models can aid in auto-generating content, providing contextual suggestions, and even document reformatting for greater readability and usability. Together, these capabilities improve overall user experience, enabling a more streamlined operation of task completion and easier access to right information sources. [16]

Thirdly, LLMs are sensitive to user input and can change documents in real time, and therefore can adapt content to suit any need or preference, which means that it actually leads to more efficient and effective task completion and therefore tends to yield higher quality results in many applications [16].

III. METHODOLOGY

We have designed a specialized speech recognition system, primarily based upon a voice-controlled application for implementing a to-do list, based on studies from which we gained so much understanding. The system heavily amalgamates various artificial intelligence algorithms-mostly NLP and neural networks. In short, it's just a collection of algorithms that function to do some specific tasks; we created an efficient voice-controlled application requiring minimal human interaction.

In other words, it is structured as an interdependent set of modules which collaborate and share knowledge and results with each other. This interdependence gives rise to a better user experience; it ensures that the interactions with the application are smooth and intuitive. Each module is designed for a special role, such as understanding voice commands, managing tasks, or providing contextual responses.

Below is a graphical representation of the various modules in play in the system, how they interact with one another to deliver a wholesome user experience.

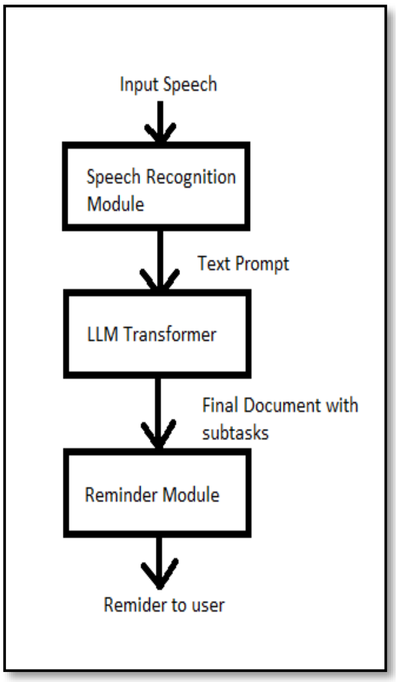


Fig 1: System Architecture

As can be seen above, the system consists of several submodules that feed into its functionality. The first submodule handles input data from the user and converts it into text-based data. The pivotal role in this was taken by the speech recognition model. Several freely available APIs and speech recognition libraries have been employed here, including the Google Speech Recognition API and the pyttsx3 library. They proved to be useful by successfully identifying and matching most of the spoken words in the early stages of production.

The core of speech recognition module is based on artificial intelligence-based algorithms. Especially, deep neural networks are the centre in speech recognition module. These networks use probability-based models to determine the best possible sequence of words the user may utter. Proven accuracy and lower rates of errors have made them very suitable for this activity. Applying these recent sophisticated techniques, the system enables reliable and efficient speech recognition that allows a more fluid interaction with the users as they operate on their to-do lists.

The next step involves the speech recognition module, where the words captured by the APIs are matched with their textual representations.
This process is made possible by the employment of the pyttsx3 library that works on concatenative synthesis along with WaveNet technology.

Using these, the library translates the speech recognized into accurate textual data hence ensuring proper capture of the input provided by the user. This would be the cut-and-splice concatenative synthesis where previously recorded speech segments would be stitched together. One of the most famous deep generative models out there is known for WaveNet, renowned especially for the high-quality audio output. What this would mean, therefore, is that the system should seamlessly go from words spoken to text for perfect functionality of the system, letting users easily interact with their voice-controlled to-do list.

This text will be the prompt for the next module in the system. Alternatively, if the user is not interested in speaking directly to the system, they can provide the prompt in text form where all the details that will have to be referenced are noted down. Both the forms of communication are supported by the system to allow for flexibility in user interaction.

As soon as the prompt is received-whether it be through speech recognition or direct input in text-the system creates a final prompt, which it then sends to the submodule that follows. It is this module that indeed does the planning and decomposition of the prompt, taking the user's input and parsing it for tasks, deadlines, and other variables. The system expands the user's access as well as fluidity in the workflow of the task management by accepting several communication types.

To make the system more user friendly, we have also incorporated more AI powered libraries and utilities like the Groq API and the LLaMA3 model break down a composite task into several subtasks automatically. So, for example, if the task entered was to plan a trip to any city, say Mumbai, the model would divide this single task into 3 subtasks, namely book tickets for travel, pack required items, travel to the chosen bus station or airport so on based on just a single prompt received from the previous submodule.

The underlying structure of this module basically consists of the large language model transformer LLaMA3-70B, which was specially built for natural language processing-based tasks humans perform on a day-to-day basis.

This architecture allows the transformer to understand the input prompt given to it by the speech recognition module. It interfaces with the Groq API which helps the model to understand the task with the given context. It exploits the few shots learning algorithm which helps by making the model adapt way faster, with minimal training data from the user.

The model is built in such a way that it understands the task given to it, formulates appropriate subtasks, accept due dates as given, and even send reminders to the user at the apt time.

The model supports two-way communication at all stages; generated using either text communication or through speech processing. The respective modules are again called to help the user obtain the desired result.

The next module works on creating a short document summarizing all the events and subtasks mentioned for a given task as agreed by the user. It also consists of important details like the due date, the due time, and whether or not the user wishes to receive a reminder for the given task.

This again is done with the help of Generative AI tools and wrappers which help to form a consolidated file requiring everything that the user wants to achieve through this task, and also in which sequence he or she wants to accomplish or execute the subtasks. [16]

Here is where approaches that are employed in natural language processing such as connectionist temporal categorization, long short-term memory, and recurrent neural networks come into significant importance. Embedded within each module, there is also a context aware language model, which generates content coherent to or related to the context of the tasks, which gives the system an edge compared to most similar applications.

The last module is where the document formed is sent as a reminder if requested by the user via email or any other notification. The content of the email is again generated by the generative transformer of the previous modules, which contain short and concise information regarding the event, and also attaches the document prepared as part of the reminder message.

Overall, the system acts like the best reminder tool to allow the user to accomplish his or her desired tasks at the right moment. The model also learns through its faults as pointed out by the user to customize its approach and assist the user in the most cooperative manner possible.

Although the pretrained nature of the libraries and the APIs gave impressive results right from the very natal stages, we also had to ensure accuracy by finetuning the parameters and training the system. Due to the phoneme and prosody prediction approach adopted, along with the probabilistic technique followed, the model did not require a very vast dataset. Since the model also contained pretrained libraries, we decided to build a custom dataset, where we trained the model by giving spontaneous, custom-made tasks.

For this, we first trained the speech recognition module with over 250 training tasks.

The first part of the training task was to ensure accurate recognition of the various classes of speech. After almost 100 examples of each class of speech, this module was ready to be incorporated into the system. The next phase was to train the model to also accept valuable information from the user like the due date and the due time. The biggest problem in this phase was to train the model to differentiate between AM and PM, and accordingly set the time for the due date. With many further training examples, this too achieved considerable results. After all this, we finally trained the model to generate an appropriate break down of the given input tasks.

Due to the few shots and zero shot learning methodologies integrated within the system, this again did not require a huge amount of training data.

Here, we used both text and speech to interact with the system, and it gained significant results, up to the point of correctly recognizing, and even correctly spelling spontaneous speech.

The model also made use of the feedback learning approach, so we could even correct the system at the exact points where it was falling into errors.

After almost 350 to 400 such training examples, the model achieved a commendable performance where the Word error rate was below 10%, and the word prediction rate stood at a whopping 97.2%.

IV. RESULTS AND DISCUSSION

The purpose of the system was to combine speech processing with Groq API and email services in order to translate a summary of a meeting into action. The system was evaluated in a Streamlit setup while generating the to-do list through both text input and voice input. The following are the main results obtained.

A. *Speech Recognition*

The system in question made use of Google's Speech Recognition API to perform the voice to the text conversion task. In a world test, the system managed to record and convert any spoken input by the user to text via a microphone.

For example, a user once dictated;

"We discussed the new interface design. Amit will handle user feedback by next Friday, and Ananya will finalize the marketing campaign budget. Rajesh should send out maintenance reminders for October 10th."

The system produced the following transcription:

"we discussed the new interface design amit will handle user feedback by next friday and ananya will finalize the marketing campaign budget rajesh should send out maintenance reminders for october 10th."

This particular output was given to the Groq API for more work to be done.

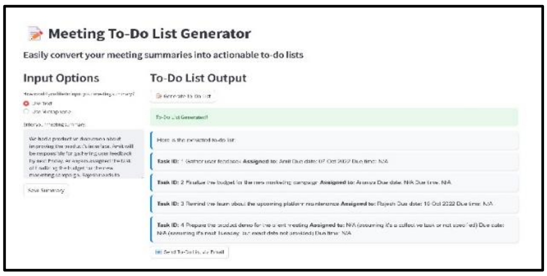
B. *To-Do List Generation*

It is then that the Groq API processes the generated or typed meeting summaries to arrive at successfully produced to-do lists. This kind of processing thus led to converting the information discussed during meetings into handy action items that users can easily follow up on. The system made sure that the presentation of the to-do list was always consistent with the initial prompt, such as important details like task IDs, responsible persons, due dates, and the specific time when appropriate. It is this extent of organization that makes for readability and can help the users become aware and learn of their responsibilities and deadlines very rapidly.

The output features even clear task identifiers, that is, allowing the tracking and referencing. There was an indication of whom to follow up with on each task, thus ensuring accountability and understanding of who does what.

The inclusion of due dates adds to the urgency and helps in prioritization of tasks, hence helping the user plan their time effectively.

This output that follows was taken from the system: it shows how the information in the meeting gets translated into an orderly to-do list. The output of this system, therefore, assists users not only in staying organized but also in improving follow-through on commitments that come out of meetings, thus making for more productivity and efficiency in workflow management.



Meeting To-Do List Generator

Easily convert your meeting summaries into actionable to-do lists

Input Options

How would you like to input your meeting summary?

● Use Text

○ Use Microphone

Enter your meeting summary:

We had a productive discussion about improving the product's interface. Amit will be responsible for gathering user feedback by next Friday. Ananya is assigned the task of finalizing the budget for the new marketing campaign. Rajesh needs to

Save Summary

To-Do List Output

Generate To-Do List

To-Do List Generated!

Here is the extracted to-do list:

Task ID: 1 Gather user feedback **Assigned to:** Amit Due date: 07 Oct 2022 Due time: N/A

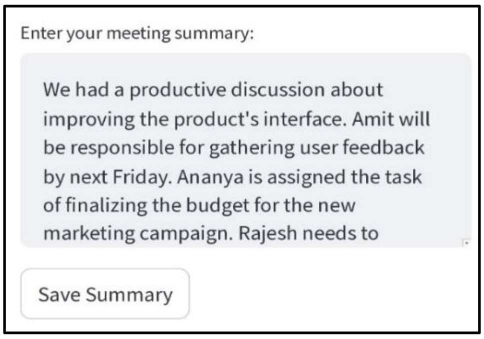
Task ID: 2 Finalize the budget for the new marketing campaign **Assigned to:** Ananya Due date: N/A Due time: N/A

Task ID: 3 Remind the team about the upcoming platform maintenance **Assigned to:** Rajesh Due date: 10 Oct 2022 Due time: N/A

Task ID: 4 Prepare the product demo for the client meeting **Assigned to:** N/A (assuming it's a collective task or not specified) Due date: N/A (assuming it's next Tuesday, but exact date not provided) Due time: N/A

Send To-Do List via Email

Fig 2: System Interface



Enter your meeting summary:

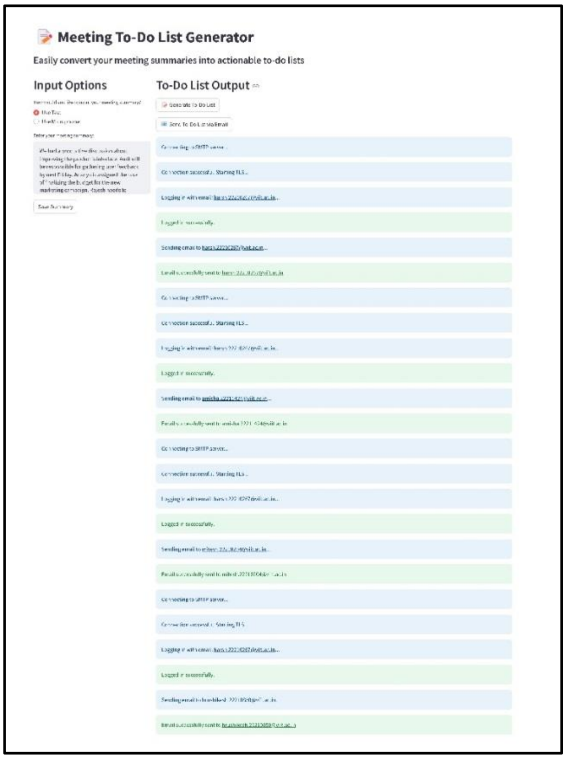
We had a productive discussion about improving the product's interface. Amit will be responsible for gathering user feedback by next Friday. Ananya is assigned the task of finalizing the budget for the new marketing campaign. Rajesh needs to

Save Summary

Fig 3: System Speech Recognition

C. *Email Functionality*

Using the Simple Mail Transfer Protocol through Gmail, it was possible for the system to email the created to-do list to selected contacts. To do lists were generated, and dispatching of the emails occurred automatically. During implementation and testing, the system was able to connect to the SMTP server, login, and send the message without problems.

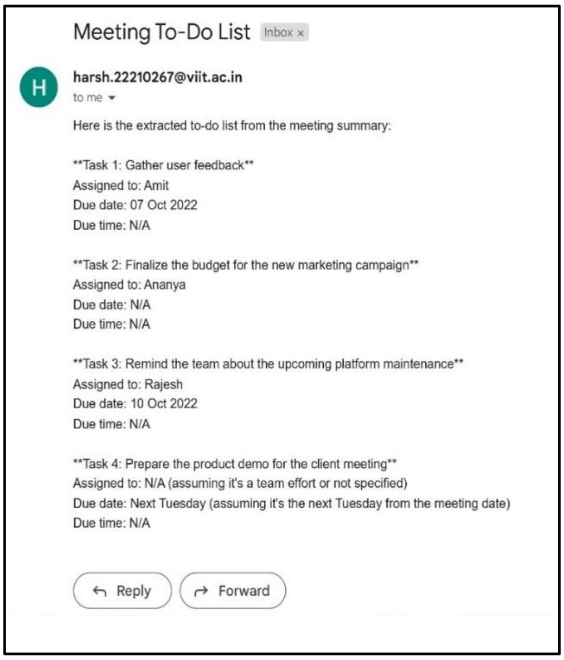

Meeting To-Do List Generator
Easily convert your meeting summaries into actionable to-do lists
Input Options
To-Do List Output
Generate To Do List
Send To Do List via Email
Save Summary

Choose how you want to input your meeting summary:
● Use Text
○ Use Microphone

Enter your meeting summary:
We had a productive discussion about improving the product interface. Amit will be responsible for gathering user feedback by next Friday. Ananya is assigned the task of finalizing the budget for the new marketing campaign. Rajesh needs to

```
Connecting to SMTP server...
Connection successful. Starting TLS...
Logging in with email: harsh.22210267@viit.ac.in...
Logged in successfully.
Sending email to harsh.22210267@viit.ac.in...
Email successfully sent to harsh.22210267@viit.ac.in
Connecting to SMTP server...
Connection successful. Starting TLS...
Logging in with email: harsh.22210267@viit.ac.in...
Logged in successfully.
Sending email to amisha.2221[illegible]@viit.ac.in...
Email successfully sent to amisha.2221[illegible]@viit.ac.in
Connecting to SMTP server...
Connection successful. Starting TLS...
Logging in with email: harsh.22210267@viit.ac.in...
Logged in successfully.
Sending email to mitesh.22[illegible]@viit.ac.in...
Email successfully sent to mitesh.22210[illegible]@viit.ac.in
Connecting to SMTP server...
Connection successful. Starting TLS...
Logging in with email: harsh.22210267@viit.ac.in...
Logged in successfully.
Sending email to [illegible]@viit.ac.in...
Email successfully sent to [illegible]@viit.ac.in
```

Fig 4: System Response



Meeting To-Do List

harsh.22210267@viit.ac.in

Here is the extracted to-do list from the meeting summary:

Task 1: Gather user feedback
Assigned to: Amit
Due date: 07 Oct 2022
Due time: N/A

Task 2: Finalize the budget for the new marketing campaign
Assigned to: Ananya
Due date: N/A
Due time: N/A

Task 3: Remind the team about the upcoming platform maintenance
Assigned to: Rajesh
Due date: 10 Oct 2022
Due time: N/A

Task 4: Prepare the product demo for the client meeting
Assigned to: N/A (assuming it's a team effort or not specified)
Due date: Next Tuesday (assuming it's the next Tuesday from the meeting date)
Due time: N/A

Fig 5: System Email Response

D. Error Handling

The system reacted in a concurrent manner to various error conditions:

- Speech Recognition Errors: If speech was not comprehended the system showed:
"Sorry, I didn't catch that."
- Errors in Network: Where the speech recognition service could not be accessed, the system replied:
"Network error. Please check your connection."
- Incorrect Input: If no text was typed or recorded, the system presented:
"Please enter or record a valid meeting summary."

V. CONCLUSION

Creating an AI that could readily generate action items based on the content of the previous meetings that had taken place is a very extreme attempt towards achieving effectiveness and efficiency in any organization. This system has been designed using speech to text, natural processing language and LLM so as to tackle one of the problems that most individuals in workplace experience – coming up with tasks after meting. The process of stenographing the meeting notes does not call for wastage of paperworks replenishing manual lists as action items are auto-generated unit reducing human error and forgetfulness.

In a way, and as the research developed we kind of understood AI when it comes to task management efficiency. The system exhibited an effective performance of speech recognition, and transcription, and action item retrieval from the corresponding meeting. In addition, some elements of this task such as persons and dates are embedded within the system thus eliminating ambiguity and incomplete tasks. Above mentioned functionality of the system allows automatic sending out of action items with the integration of to-do services for the corresponding departments in order to make sure that even one deadline is not missed.

The assessment of the system has been successful in terms of the accuracy of work done, the satisfaction of users, and the amount of time saved in the processes following meetings. The task extraction and assignment features are not only intended to lessen the load off employees but also to enhance the entire task management process. It, therefore, assists organizations in controlling workloads and productivity since there is a systematic way of dealing with the outcomes of meetings.

It is worth emphasizing, however, the shortcomings of the system. To illustrate, several factors such as background noise, accents or speech clarity can negatively affect the performance of the speech recognition module of the application. The system works efficiently in a controlled environment, but the improvement of the enhancement of the systematization and intelligence of the system is still required when there is noise or other extraneous factors. Furthermore, there is a module in the system that resolves uncertainty, allowing it to cope with vague or otherwise difficult commands; however, this may need to be tweaked to avoid inconsistencies with the related to-do list generation.

VI. FUTURE SCOPE

This system could be enhanced and expanded in a number of ways over time as new ideas in artificial intelligence emerge. The evolution of the system's ability to recognize different voices is one such area that may need improvement. This can be done by integrating more complex models that can work efficiently with different accents, dialects, and background sounds, hence such systems will also be more reliable and precise. This means that it can be used by a broad spectrum of people and environments even in noisy places like building sites or large open offices.

In addition to this, another useful research perspective for the future will concern expansion of the NLP and LLM constituents. Existing systems of comprehension and generation of natural Nevertheless, it is evident that these sophisticated systems can be further improved in their capabilities to deal with complex hierarchies of instructions and infer goals from other, more nuanced, verbal forms. An additional enhancement of the system to recognize tasks and actionable items from meetings, including their vague descriptions, will be a more advanced context-aware AI that understands the nuances of human social interactions.

In addition, the potential of this system may also be enhanced even more by interconnecting it with other tools and platforms for example, project management software. This would mean that users would be able to update their task lists and track their progress within one platform and at the same time get a complete picture of all their tasks and their due dates. It may also allow for synchronization across different platforms such that updating of tasks in the relevant application makes the updates visible in others.

Furthermore, systems supporting virtual meetings are in high demand as more and more organizations go for remote and hybrid work arrangements. Subsequent versions of this system can include elements for remote collaborations such as transcribing and data mining meetings held over teleconferencing facilities.

In this respect, the system will be able to support and grow with contemporary work environments that increasingly rely on virtual meetings.

Furthermore, alongside enhancing the tech capabilities of the system, there is a possibility of widening its use to non-conventional office environments too. A case in point, other advanced education systems may create a similar system to derive action points from meetings or lectures held in classes, while even the medical institutions might use a similar system during clinical meetings to ensure all decisions made are recorded and acted on.

As a last point, with the continuous progress of the AI techniques, there is a likelihood that the system could develop into an all-inclusive task management assistant. For instance, future developments of the systems can be that, they will not only be able to create to-do lists but also be able to anticipate and forecast risks or obstacles in accomplishing the tasks derived from the previous components that they have worked on. Furthermore, it may even recommend how users could organize their tasks or disperse their resources, thereby enhancing the time management and workload handling of the users.

By way of conclusion, although the existing framework serves as an effective mechanism in the automation of the management of post meeting activities, every system has its own room for improvement. The possibilities for the system, however, are immense and if such a system continues to evolve, employing more artificial intelligence tools, it may serve as a core element of the modern office in meeting activities, task management and overall productivity of the organization.

VII. REFERENCES

[1] M. Forsberg, ― hy Is Speech Recognition Difficult?‖, Chalmers University of Technology, Citeseer, (2003)

[2] Murray Shanahan. 2024. Talking about Large Language Models. Commun. ACM 67, 2 (February 2024), 68–79. https://doi.org/10.1145/3624724

[3] J. Pei, "Automatic Speech Recognition," 2010. [Online]. Available:

https://www.researchgate.net/publication/228687340

[4] D. Povey et al., “The Kaldi Speech Recognition Toolkit.” [Online]. Available: http://kaldi.sf.net/

[5] W. Ghai and N. Singh, “Literature Review on Automatic Speech Recognition,” 2012.

[6] Karpagavalli, S., and Edy Chandra. "A review on automatic speech recognition architecture and approaches." International Journal of Signal Processing, Image Processing and Pattern Recognition 9.4 (2016): 393-404.

[7] N. Morgan and H. Bourlard, "Continuous speech recognition," in IEEE Signal Processing Magazine, vol. 12, no. 3, pp. 24-42, May 1995, doi: 10.1109/79.382443. keywords: {Speech recognition;Hidden Markov models;Vocabulary;Statistics}

[8] ChatGPT Generative Pre-trained Transformer; Zhavoronkov A. Rapamycin in the context of Pascal's Wager: generative pre-trained transformer perspective. Oncoscience. 2022 Dec 21;9:82-84. doi: 10.18632/oncoscience.571. PMID: 36589923; PMCID: PMC9796173.

[9] Yu, Dong, and Lin Deng. Automatic speech recognition. Vol. 1. Berlin: Springer, 2016.

[10] Gin-Der Wu and Ying Lei “A register array based low power FFT processor for speech recognition” Department of electrical engineering national Chi Nan university Puli, 454 Taiwan

[11] C. J. Leggetter and P. C. Woodland, “Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models,” Computer Speech and Language, vol. 9, no. 2, pp. 171–185, 1995

[12] Sendra, J. P., Iglesias, D. M., Maria, F. D., “Support Vector Machines For Continuous Speech Recognition”, 14th European Signal Processing Conference 2006, Florence, Italy, Sept 2006

[13] Feifei Xiong, Stefan Goetze, Bernd T. Meyer, "Combination strategy based on relative performance monitoring for multi-stream reverberant speech recognition", 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp.4870-4874, 2017 .

[14] H. Hermansky. Perceptual linear predictive (plp) analysis of speech. Journal of the Acoustical Society of America, 87(4):1738–1752, April 1990.

[15] S. Renals, N. Morgan, H. Boulard, M. Cohen and H. Franco, ―Connectionist probability estimators in HMM speech recognition‖, IEEE Trans. Speech Audio Processing , vol. 2, no. 1, (1994), pp. 161–174.

[16] G. Yenduri et al., "GPT (Generative Pre-Trained Transformer)— A Comprehensive Review on Enabling Technologies, Potential Applications, Emerging Challenges, and Future Directions," in IEEE Access, vol. 12, pp. 54608-54649, 2024, doi: 10.1109/ACCESS.2024.3389497.