

JavaScript Powered Web Based Speech to Text Converter

M. Sai Ravindra¹, K. Tulasi Krishna Kumar²

¹MCA 2nd Year, Master of Computer Application, Sanketika Vidya Parishad Engineering College, Visakhapatnam, India

²Associate Professor and Placement Officer, Alwardas Group, Visakhapatnam, India

Abstract

The java script-powered web-based speech-to-text converter interface is used to access speech recognition, which involves the use of the Web Speech API, which allows web developers to incorporate speech recognition and synthesis capabilities into their web applications. The Web Speech API offers a selection of APIs for recording audio input, identifying speech, and speech to text conversion. We have developed a speech-to-text input method for use with web platforms. The system is provided as a JavaScript with dynamic HTML files, and CSS. Users can use voice commands to create content and articles instead of typing. Speech recognition entails listening to speech through a device's microphone and having a speech recognition service check proper grammar for any sentence or word phrases and use as a translator from one language to other. A result (or series of results) is returned as a text string when a word or phrase is correctly identified, and subsequent actions can be started as a result. The quality of the audio input, background noise, and language support are some of the variables that affect how accurately speech-to-text recognition performed using JavaScript. However, the technology has improved significantly over the years and can provide accurate and reliable results for many use cases. Typically, the device's built-in default speech recognition system—which is present in the majority of contemporary operating systems and may be used to provide voice commands—will be utilised for speech recognition. Think about Dictation on macOS, Siri on iOS, Cortana on Windows 10, Android Speech, etc.

Keywords: speech recognition, web api, ASR server, javascript libraries, speaker models, natural language processing, HMM model, ANN, DTW,LPC,MFCC, hybrid approach, neural networks, Real-time transcription.

1. Introduction

Speech is the most fundamental, widely used, and effective type of interactivity between individuals. These voice technologies are frequently used for a small but fascinating variety of tasks. With the use of these technologies, machines may reply to human voices accurately and dependably while also offering beneficial and worthwhile services. A microphone-captured acoustic signal is converted into a set of words using a process known as speech-to-text or voice recognition technology. It is possible to prepare documents using the captured data. Making a computer recognise other people's voice based on specific

words or phrases is known as speech recognition. Despite the fact that speaking is the simplest form of communication, there are still some concerns with speech recognition, including issues with fluency, pronunciation, broken words, and stuttering. While processing a speech, all of these must be addressed. One of the key terms in the documentation world is text summary. In addition to taking a lot of time to read and comprehend, lengthy documents are also time-consuming. So the speech to text will perform like user friendly to the user. It is offered as a JavaScript with a complex mechanism based on HTML dynamic documents, CSS, and Java script. Users can access the internet using standard PCs and browsers including Internet Explorer, Mozilla Firefox, and Safari. Hence, even non-technical web designers, editors, and bloggers can use our method with ease.

2. Speech Recognition System Classification

By characterising the type of voice utterance, speech recognition system classification may be divided into numerous categories. They can distinguish between different vocal and speaker models. The difficulties are briefly described below.

2.1. Speech Utterance Categories

Voice recognition systems are categorised based on the types of utterances they can identify. These systems use advanced algorithms and machine learning techniques to analyse the acoustic signals of spoken language and convert them into written text. They are used in a variety of applications, including virtual assistants, transcription services, and dictation software. Voice recognition systems work by first capturing the audio input using a microphone, then processing the speech signal to identify individual sounds and words. The system then matches the sounds and words to a database of known language patterns to determine the meaning of the speech. Finally, the system generates a text output that represents the spoken words. They are categorised as

2.1.1 Isolated Word

An isolated word recognizer typically needs silence (or the absence of an audio signal) on both sides of the sample window for each spoken word. It only accepts one word at once.

2.1.2 Linked Word

It is comparable to an isolated word but allows for the "running-together" of multiple utterances with a brief break in between them.

2.1.3 Continuous Speech

This feature enables people to speak normally while the computer determines the content in parallel.

2.1.4 Spontaneous Speech

It is the type of speech which is natural sounding and is not rehearsed.

2.2. Speaker Model Types

Based on speaker models speech recognition systems can be divided into two basic categories: speaker dependent and speaker independent.

2.2.1 Speaker Dependent Models

These systems were built with a particular speaker in mind. They are more accurate and simpler to develop, but they are less adaptable.

2.2.2 Speaker Independent Models

These systems were made to accommodate different speaker types. Although less precise and more challenging to create, these systems are exceedingly versatile.

2.3. Vocabulary Types

The speech recognition system's vocabulary size has an impact on the processing demands, accuracy, and the system's complexity. The many categories of vocabularies used in speech-to-text and voice recognition systems are as follows

2.3.1 Simple words with only one letter.

2.3.2 Two- or three-letter words with a medium vocabulary.

2.3.3 A large word list with more letters.

3. Terminology of Speech Recognition

Voice recognition is a technology that enables a gadget to pick up the words spoken by a human into a microphone. The system eventually outputs recognised words after these words have been subjected to voice recognition processing. Voice translation is crucial because it eliminates the language barrier in international commerce and cross-cultural interactions by enabling people from all over the world to converse in their native tongues. Achieving worldwide voice translation would be extremely significant from a scientific, cultural, and economic standpoint. Our project eliminates the language barrier. According to their capacity to comprehend the terms and word lists they have in various groups, speech recognition systems can be divided into several categories. Below are some terminologies used frequently in JavaScript-based Speech-to-Text (STT) Web Audio API, Speech recognition API, Machine learning, Natural language processing (NLP) Language model, Audio input, Text output, Accuracy, Latency, API key, Real-time transcription, Streaming recognition, Error handling, User interface (UI), Accessibility

4. Behaviour of Speech Recognition

JavaScript's speech-to-text functionality can behave differently depending on how it's implemented and which Web Speech API is being utilised. It's also vital to keep in mind that using JavaScript for speech-to-text conversion could have limitations in terms of precision, language support, and compatibility with various browsers and gadgets. To provide a seamless user experience, it is crucial to extensively test and optimise the speech-to-text capabilities. When using STT conversion, the system recognises words and phrases in audio input provided by a human or machine and transforms them into a text format that can be read. The communication between people, machines, and machines-to-people is made easier as a result. People communicating or interacting with one another across linguistic and dialectal boundaries is extremely beneficial. People speaking in various dialects and languages may not be able to understand one another in the absence of a STT conversion system. So, in such a situation, a STT converter may be helpful by translating the words uttered by a person with a distinct accent or dialect into text form that is

easily legible and understood by the other person. Natural language processing (NLP) is the study of how human language interacts with computers such that the latter can recognise, decipher, and even create human language. Translation is the transmission of meaning from one language (the source) to another language (the target). Essentially, speech synthesis has two main applications. The following describes the behaviour of JavaScript-based speech-to-text.

4.1 User Speaks Into Microphone

The Web Speech API is used by the browser to record the user's voice while they speak into the microphone.

4.2 Speech Recognition Starts

The browser starts the speech recognition process when the user clicks on the speech-to-text button or trigger. The audio input is converted to text using the Web Speech API.

4.3 Results of Recognition

The Web Speech API processes the spoken input and sends the text that has been identified back to the JavaScript code.

4.4 Display of Recognised Text

The identified HTML element, such as a text input box, text area, or paragraph, displays the recognised text. This enables the user to read through and, if necessary, change the text.

4.5 End of Recognition

After voice recognition is finished, the browser stops capturing audio input, and the trigger or button for speech-to-text is disabled until the user starts a new speech-to-text process.

5. Utilisation of Java Script and the Java Script Library for Speech-To-Text

When you click, the dynamic HTML application built in JavaScript is presented in the pane. a button on this page that records the voice of the user and converts it to text. The computer's microphone can record the user's voice once the recording device is turned on by pressing the button. The collected signals are uploaded to the ASR server as soon as the button is pressed. When making voice recordings, the button's bar transforms into a power level display. JavaScript, a scripting or programming language, is what I utilised for this project. It's a language that lets you generate dynamically updated material, manage multimedia, animate graphics, and pretty much everything else.

5.1. There are Different Types of Libraries Used in The Java Script That Was

5.1.1 Annyang

Annyang is a JavaScript speech recognition library that enables voice instructions to be issued to the website. On top of speech recognition web APIs, it was constructed. We'll provide an illustration of how annyang functions in the following section.

5.1.2 Artyom.Js

Artyom.js is a JavaScript speech synthesis and recognition package. On top of Web speech APIs, it was constructed. Further to voice commands, it also offers voice responses.

5.1.3 Mumble

To operate the website with voice commands, use the Mumble JavaScript Voice Recognition module. It is constructed using the Speech Recognition Web APIs. It operates in a manner akin to an annyang.

5.1.4 Julian.Js

For academics and developers working in the field of voice, Julius is a high-performance, compact large vocabulary continuous speech recognition (LVCSR) decoder. From a microcomputer to a cloud server, it can carry out real-time decoding on a variety of computers and gadgets.

6. Existing Model

Currently we are using 10 different techniques as well as 6 different models in speech to text conversion. These 10 techniques are 1.STT Conversions, 2. TTS and IVR Conversions, 3.STT Face recognition, 4.MFCC and HMM, 5.Automatic Speech Recognition, 6.HMM model and human machine interface, 7.Pattern Recognition, Neural Network and Artificial intelligence, 8.ANN and HMMSpeech Recognition Feature Extraction,MFCC and Dynamic Time Wrapping (DTW), 9.Machine Learning, ANN, ASR and Cuck Search Algorithm, 10.HMM, ANN and DWT.Remaining 6 models are 1.Linear Predictive Coding (LPC),2. Mel-Frequency Cestrum Co- efficient (MFCC), 3.Dynamic Time Wrapping, 4.Hidden Markov Model, 5.Neural Network, 6.Hybrid Approach.We have researched all of the approximately 10 different techniques and 6 different models in use, and we have listed descriptions of each methodology and model below. In our study, we found that there are a lot of issues, mostly with the short time statistic signal in ANN and HMM, and that there are certain drawbacks to the methodologies and models below that we have discussed in sections 6.1 and 6.2.Thus, the voice signal is represented by the introduction of digital signal processing techniques such feature extraction and feature matching. A number of techniques, including Liner Predictive Predictive Coding (LPC), Hidden Markov Model (HMM), Artificial Neural Network (ANN), and others, are assessed in an effort to find a simple and efficient technique for voice signal. The pre-processing or signal-filtering step is followed immediately by the extraction and matching phase. Mel Frequency Cepstral Coefficients (MFCCs), a non-parametric modelling methodology, are used as extraction techniques for the human auditory perception system. Techniques for features matching have been developed using Sakoe Chiba's Dynamic Time Warping (DTW), a nonlinear sequence alignment method.

Sl. No	Techniques Used	Description
1.	STT Conversions	They recommended that in order to transform audio messages to text for the user, the audio message should first be captured. The suggested design for a voice-based email system uses three modules: STT conversion and TTS conversion and IVR.
2.	TTS Conversions and IVR	The suggested system places a strong emphasis on giving consumers an intuitive interface. Technology called Interactive Voice Response is used by the system. An automated voice will instruct the user to perform certain tasks in order to access certain services in this system.
3.	STT and Face recognition	The article suggests creating a system that enables people who are blind, visually challenged, or both to use email services as effectively as a typical user. The system relies almost less on a mouse or

		keyboard and operates on STT operations. The user identity is also verified using face recognition.
4.	MFCC and HMM	STT system was proposed, employing HMM in place of the conventional MFCC. A novel method using HMM was proposed because the traditional MFCC methodology was less effective at extracting features from voice data. In comparison to the MFCC approach, the features sent to the HMM network produced higher feature recognition from the input audio. For a Speech-To-Text conversion system, HMM shown a significant improvement in the quality of feature extraction from the audio, leading to faster calculation and greater accuracy.
5.	Automatic Speech Recognition, HMM model and human machine interface	The study of STT deployment by HMM made recommendations for creating voice-based machine interaction systems. The system might be used to assist two different user types: <ul style="list-style-type: none"> •Individuals with disabilities who are unable to use a mouse or keyboard to access their email will benefit from using a Speech-to-Text conversion system. •Individuals who prefer to communicate in their original language, such as English, Telugu, or Hindi, because they do not comprehend English or are not proficient in it.
6.	Pattern Recognition, Neural Network and Artificial intelligence	A variety of speech representation and classification techniques were proposed. They also used a variety of feature extraction approaches in addition to database performance. They examined the numerous issues raised by ASR and suggested solutions. They focus on three different approaches to speech recognition: the AI Approach, the Pattern Recognition Method, and the Acoustic Phonetic Way.
7.	ANN and HMM	Several strategies can be combined to improve the suggested pace of STT conversion and produce text of higher quality. The goal is to create a continuous STT system that can accurately recognise the voice of many speakers, has a considerably greater vocabulary, and is speaker independent. The usage of ANN and HMM in tandem will be crucial for creating such a system.
8.	Speech Recognition, Feature Extraction, MFCC and Dynamic Time Wrapping (DTW)	A thorough analysis reveals that as the system's performance and dependability are impacted, some limits are introduced. The study also shows that English is the language for which STT is carried out to the greatest extent. With Hindi and other regional languages, less has been done. The study also looked at whether language had the highest rate of speech recognition: English or any other. The phonetic character of Indian languages is the cause of their poor recognition rate.
9.	Machine Learning, ANN, ASR and Cuck Search	An overview of the fundamental procedures carried out in a STT system, including the ASR architecture (Automated Speech Recognition). The use of machine learning in ASR, SVM, and ANN

	Algorithm	with the Cuckoo search method, along with ANN and back propagation classifier, is the main topic of this article. The fundamental STT system phases, including pre-processing, feature extraction, and classification, are investigated using machine learning. Traditional classifier findings can be further enhanced by combining them with other optimisation algorithms, according to the results generated. Hybridization of an algorithm is seen to be a better technique.
10.	HMM,ANN and DWT	They investigated numerous STT approaches. They came to the following conclusions after looking at various STT synthesis, TTS synthesis, and speech translation systems: <ul style="list-style-type: none"> •In STT, HMM functions as a better generator of text from speech despite its shortcomings due to their computational feasibility. •The system makes sure that the text it produces is smooth, rapid learning, and data-acquiring while also making sure that it is syntactically and grammatically correct.

Table 1- Various Techniques Speech to Text

By observing various methods we concluded that the field of speech-to-text conversion has a huge amount of room for development. We gathered from various research papers to finalize that the speech to text method will use in various purposes in future. A brief description of various research papers that were examined for this study is given above. The given above table 1 represents the summarization of various methods applied for Speech-To-Text. Going through these papers it was observed that there is an additional scope of work on the STT conversion method.

6.1. Various Models for Speech to Text

We'll discover a range of speech-to-text models, including those employed for various tasks. These six different model types each feature a unique speech recognition method, thus it is significant. The table will be shown below.

Linear Predictive Coding (LPC)	Mel-Frequency Cestrum Co-efficient (MFCC)	Dynamic Time Wrapping	Hidden Markov Model	Neural Network	Hybrid Approach
---------------------------------------	--	------------------------------	----------------------------	-----------------------	------------------------

Table 2- Various models for speech to text

6.2. Comparison between Various Models

METHOD	ADVANTAGE	DISADVANTAGE
Linear Predictive Coding (LPC)	<ul style="list-style-type: none"> • In order to extract features, LPC is a static approach. The idea behind LPC is that it has the ability to use voice samples from previous recordings as a linear combination. • Following the fragmentation of the voice signal into N frames, these framed windows are transformed to text. 	<ul style="list-style-type: none"> • Employs spectral analysis at a fixed resolution and an arbitrary frequency scale.
Mel-Frequency Cestrum Co-efficient (MFCC)	<ul style="list-style-type: none"> • The MFCC method is another one that extracts signal features using a filter bank. • The method uses the Discrete Fourier Transform, Windowing, and Framing stages for STT conversion. 	<ul style="list-style-type: none"> • The issue with MFCC is that it necessitates normalisation because the values in MFCC are not very effective when the environments are present.
Dynamic Time Wrapping (DTW)	<ul style="list-style-type: none"> • Dynamic programming is used to locate the analogy in two-time series occurrences that have varying speeds. Finding a workable match between the two feature vector sequences is its main goal. 	<ul style="list-style-type: none"> • The choice of the reference template for contrasting the time series events is a challenge.
Hidden Markov Model(HMM)	<ul style="list-style-type: none"> • HMM is a statistical model that is used for STT conversion, and it has its own structure and can learn on its own, which is why it is so effective. 	<ul style="list-style-type: none"> • This method uses a serial HMM and treats the voice signal as a static signal or short-term time static signal.
Neural Network (NT)	<ul style="list-style-type: none"> • A graph-based statistical model of a neural network is also available. • For the state transactions, neural networks use connection function values and connection strengths. 	<ul style="list-style-type: none"> • Parallel ANNs are used in this neural network architecture.
Hybrid Approach(HA)	<ul style="list-style-type: none"> • Speech frequencies are parallel, whereas syllable sequences and words are serial; the proposed hybrid approach is used for speech to text conversion. • This displays how both approaches are effective in a variety of settings. The HMM and NN approaches are combined in their execution. The Markov models can employ potential phoneme sequences or words and neural networks perform well when analysing probabilities from simultaneous voice input. 	<ul style="list-style-type: none"> • There is not much disadvantage identified in Hybrid approach as on date.

Table 3- The various models for speech-to-text conversion

We have determined that HMM offers the highest level of efficiency for STT conversion by analysing the various STT approaches. Moreover, neural network for STT offers the highest level of efficiency. The HMM model has the maximum accuracy in comparison to other models, hence we have developed the Hybrid technique for STT conversion that uses both of these technologies.

7. Browser Testing and Libraries Testing in STT

The two most crucial tests for speech to text recognition are those done on browsers and libraries. One test is done on browsers, while the other is done on libraries.

7.1. How Browser Test and Library Test Will Work?

These two tests are crucial since they determined if speech to text (STT) functions correctly or not. The only criteria used to judge if something was user-friendly or not and whether it would function best on a computer or on mobile devices was whether it would work in various browsers and libraries. The results of both tests are displayed in below.

7.2. Browser Compatibility Test

We gathered test results from several study articles, and we came to the conclusion that many different browsers took part in the speech recognition test for the speech to text approach. Because of how the browser functions on both computers and mobile devices, the test was conducted on both.

The browser compatibility test took place, and we gathered reports from the browser compatibility test in both pcs and mobiles. Many browsers participated, some for pcs and some for mobiles. The only intensity was how it will work and how user-friendly to users, and if users don't have pcs then how will they use it. Participating PC browsers: Chrome, Edge, Firefox, Safari, Internet Explorer, and Opera. Participated browsers for mobiles-Android Web view, Chrome for android, Firefox for android, opera for android, Safari on IOS, Samsung Internet.

7.3. Results of Browser Testing

Speech to text recognition was crucial in the browsers. The outcomes that will determine how easily and accurately the browser functions on pc and mobile devices.

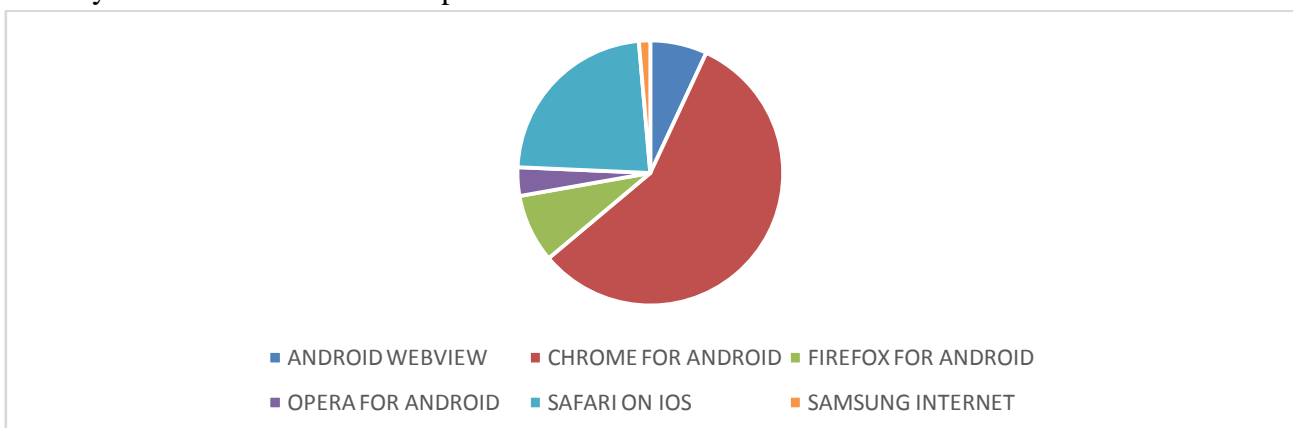


Figure 1: Results of browser testing for mobiles

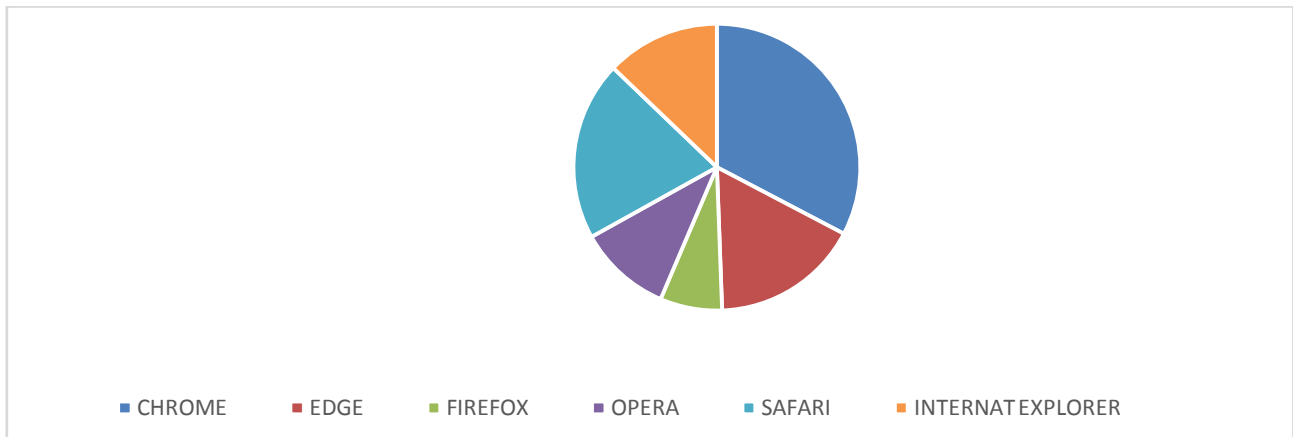


Figure 2: Results of browser testing for pc

As a consequence of the speech recognition testing, we came to the conclusion that Chrome provided the best and most accurate recognition when compared to other. Chrome performs most accurate in both pc and mobile browser test. This was fantastic because the majority of us use Chrome, making it a comfortable experience for all users.

7.4. Library Testing

The NPM trends carried out the library testing. Testing speech to text recognition was crucial in the libraries. But, for speech recognition, all of the participating open source java script libraries were used. Numerous open source java script libraries took part in this test.

Participated Libraries

- 7.4.1. Annyang
- 7.4.2. Artyom.Js
- 7.4.3. Mumble
- 7.4.4. Julius.Js
- 7.4.5. Voice-Commands.Js

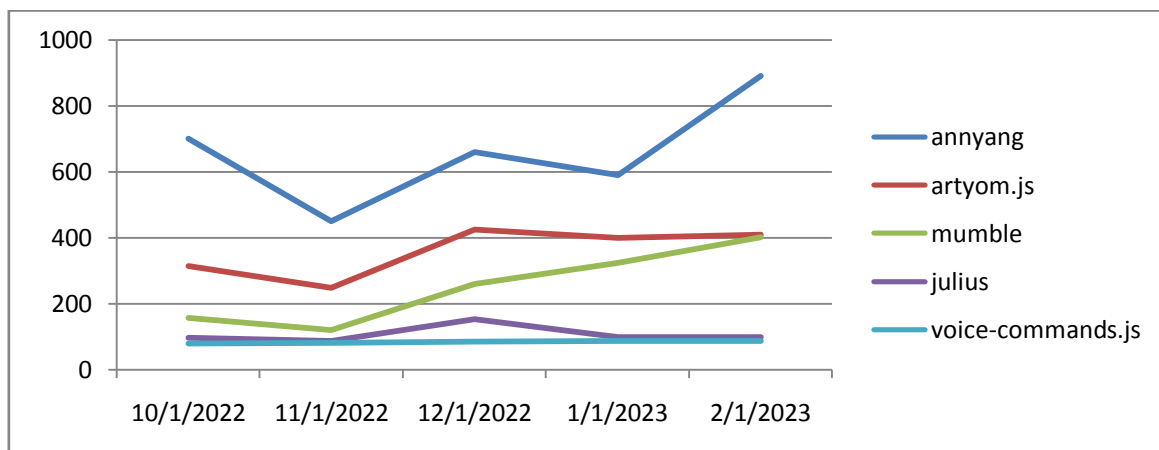


Figure 3: Line chart of library testing

7.5. Results of Libraries Testing

In the above results we concluded that Annyang was most downloaded speech recognition library in previous 6 months and arytom.js in the second place and others are in following places.

8. Proposed Model

The work for this study is based on the flowchart below. The models shown previously are composed of millions of parameters, from which the instruction corpus must be learned. When necessary, we employ extra information, including the fact that speech and text are closely related or that we are preparing to translate.

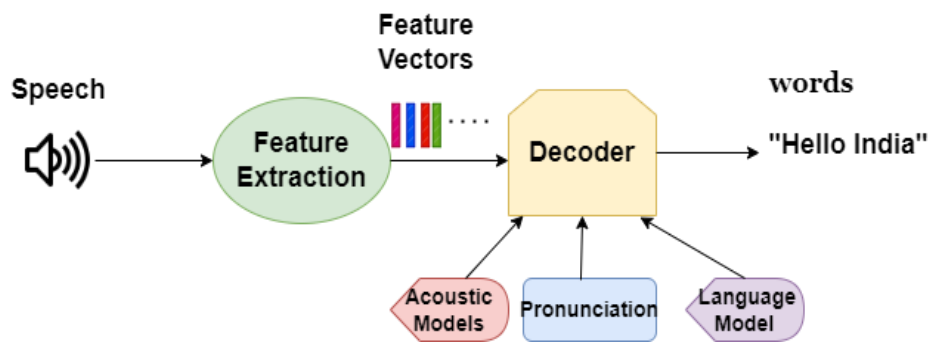


Figure 4: Proposed model of speech to text converter

In this proposed model of speech to text, we finally came to the conclusion that when a user speaks, the microphone receives the signal, the feature extraction removes the noise, the decoder is used to decode the voice, including acoustic model, pronunciation model, and language model of speech, and then it produces the output in the form of text, such as "Hello India" in the flow chart above. We need to create dynamic HTML, CSS, and Java script that resemble a website in order to represent the aforementioned programme. It has the ability to convert speech to text, record audio input, and recognise voice. Web pages can easily incorporate speech-to-text input. On the client side, voice-enabled websites can be accessed using standard browsers without any extra software. Users can access the internet using standard computers and browsers. So, our technology can be used with ease by non-technical users such as web designers, editors, bloggers, as well as physically blind or disabled people.

9. Conclusion

In conclusion, JavaScript-based speech-to-text technology has completely changed how we interact with the web. Web developers can now easily incorporate speech recognition functionality into their online applications with the aid of contemporary web APIs like the Web Speech API and the Speech Recognition API. People with impairments, such as those who are visually impaired or have mobility limitations, now find it simpler to access web information and engage with web apps thanks to this technology. We also included translation for this software, which easily translates practically all languages. Also, it has made it easier to develop cutting-edge voice-driven services and applications. Similar, dynamic HTML pages without the need for specialised browsers and add-on software, it enables users to see voice-activated web sites. The ease with which our technology may be

integrated into a web page is a benefit for web developers. Our publicly available ASR service has amassed a sizable amount of input voices in order to track actual human-machine spoken interactions in private contexts. Testing with libraries and browsers went well for us. However, there are still certain issues with voice recognition technology, including issues with accuracy and privacy. Particularly in noisy settings or with strong accents, speech recognition algorithms might not always accurately record speech. Users' lack of knowledge about what is being captured and how the data is used gives rise to privacy concerns. Developers must take the necessary precautions to preserve users' privacy and be open and honest about how they gather and use voice data. Ultimately, JavaScript-based speech-to-text technology has the potential to change how we interact with the web and provide products that are more inclusive and accessible. We may anticipate seeing even more cutting-edge and approachable speech-enabled applications in the future as technology advances.

10. Future Scope

Future Potential for Speech To Text Using Java Script:-There are numerous exciting potential for designing speech-to-text apps utilising JavaScript as the field of speech-to-text technology is continually expanding and innovating. The following are some potential future applications for JavaScript-based speech-to-text:

10.1 Real-Time Transcription

Real-time transcription is a key area of advancement for speech-to-text technology, and JavaScript may be used to create many useful applications for real-time transcription. Applications like real-time language translation, video and audio transcription, and live captioning for online events might all fall under this category.

10.2 Improved Accuracy

While speech-to-text technology's accuracy has considerably increased recently, there is still potential for improvement. It is feasible to develop speech-to-text systems that are more precise and effective by utilising machine learning and deep learning techniques in JavaScript.

10.3 Integration with other Technologies

Advanced speech-to-text apps can be created by combining JavaScript with other technologies like artificial intelligence (AI) and natural language processing (NLP). With chat bots, virtual assistants, and voice-activated home automation systems, for instance, speech-to-text technology can be used.

10.4 Accessibility

Accessible digital information for those with disabilities can be improved with speech-to-text technology. Web designers can make digital material more accessible to those with hearing impairments by including speech-to-text technologies into their web applications and websites.

Overall, JavaScript-based speech-to-text technology has a bright future, and there are many potential for developers to create cutting-edge, practical applications in this area.

References

1. Abookreferencefrom https://books.google.co.in/books?hl=en&lr=&id=iDHgboYRzmgC&oi=fnd&pg=PA1&dq=speech+recognition&OTS=jdcRDOYsjI&sig=wUPpRjJetBQAv0-68ZHrs8UiO9E&redir_esc=y#v=onepage&q=speech%20recognition&f=false

2. S.Biswas, R.Rautray, R. Dash and R. Dash, "Text Summarization: A Review," 2018 2nd International Conference on Data Science and Business Analytics (ICDSBA), Changsha, China, pp. 231-235, 2018.
3. Reference taken from IEEE platform <https://ieeexplore.ieee.org/abstract/document/4777877>
4. A website reference from <https://www.sciencedirect.com/science/article/abs/pii/S0167639307000404>
5. An article reference from <https://arxiv.org/abs/2002.01320>
6. K.M.Shivakumar, V. V. Jain and P. K. Priya, "A study on impact of language model in improving the accuracy of speech to text conversion system," 2017 International Conference on Communication and Signal Processing (ICCSP), Chennai, pp. 1148-1151, 2017.
7. Reference taken from IEEE platform <https://ieeexplore.ieee.org/abstract/document/222875>
8. Ayushi Trivedi, Navya Pant, Pinal Shah, Simran Sonik and Surrey Agrawal, "Speech to text and text to speech recognition systems-A review," IOSR Journal of Computer Engineering (IOSR-JCE), 2018.
9. Reference taken from IEEE platform <https://ieeexplore.ieee.org/abstract/document/4777877>
10. An article reference from <https://www.mdpi.com/2071-1050/13/22/12392>
11. An article reference from <https://dl.acm.org/doi/abs/10.1145/3180155.3180184>
12. Reference taken from IEEE platform <https://ieeexplore.ieee.org/abstract/document/115546>
13. An article reference from <https://portal.etsi.org/stq/cta/dsr/avios%20dsr%20paper.pdf>
14. An article reference from https://www.isca-speech.org/archive_open/odyssey/odys_101.html
15. Reference taken from IEEE platform <https://ieeexplore.ieee.org/abstract/document/8683550>
16. A website reference from <https://www.section.io/engineering-education/speech-recognition-in-javascript/>
17. Reference taken from IEEE platform <https://ieeexplore.ieee.org/abstract/document/8901594>
18. A website reference from <https://cloud.google.com/speech-to-text>
19. An article reference from <https://research.google/pubs/pub44268/>
20. An article reference from <https://betterprogramming.pub/perform-speech-recognition-in-your-javascript-applications-91367b0d0>
21. An article reference from <https://docs.ahcts.co/how-to-program/api-design-ebook-2012-03.pdf>
22. An article reference from <https://link.springer.com/article/10.1007/s11423-017-9556-8>
23. An article reference from <https://www.tandfonline.com/doi/abs/10.1080/14613800500169779>
24. An article reference from <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=97b24926b0b90d57d17fd89fd0b807fba2c89d43>
25. An article reference from https://link.springer.com/chapter/10.1007/978-81-322-3972-7_19
26. An article reference from <https://arxiv.org/abs/1003.4083>
27. An article reference from <https://academic.oup.com/jdsde/article/6/4/285/386068>
28. Reference taken from IEEE platform <https://ieeexplore.ieee.org/abstract/document/1457610>
29. An article reference from <https://www.ijedr.org/papers/IJEDR1404035.pdf>
30. An article reference from https://link.springer.com/chapter/10.1007/978-3-030-28374-2_51
31. Reference taken from IEEE platform <https://ieeexplore.ieee.org/abstract/document/9513968>
32. An article reference from <https://ojs.aaai.org/index.php/AAAI/article/view/6360>
33. An article reference from <https://aclanthology.org/2020.acl-main.350/>

34. A website reference from <https://www.sciencedirect.com/science/article/abs/pii/S0167639316301625>
35. Reference taken from IEEE platform <https://ieeexplore.ieee.org/abstract/document/6857341/>
36. A website reference from <https://www.tandfonline.com/doi/abs/10.1080/09588220701489374>