the
jaltcalljournal

ISSN 1832-4215
Vol. 11, No.2 Pages 177–187
©2015 JALT CALL SIG

# Using Web Speech technology with language learning applications

Paul Daniels

*Kochi University of Technology*
*daniels@kochi-tech.ac.jp*

Column

## Computerized recognition and assessment of speech

Electronic communication has rapidly evolved over the last few decades, from using chat commands via Telnet and USENET, to making video calls via Skype and FaceTime. We can now simply speak to our mobile device when drafting an email or calling a friend instead of entering text with a keyboard. While computer-mediated communication has traditionally focused on human-to-human communication, a new paradigm may emerge that includes human conversations with artificial intelligence, given that talking *"to"* a computer is becoming as common as talking *"through"* a computer. It appears that it will not be long before we speak to a computer as if it was a human rather than just a machine. In 2011 Google introduced their voice search, and more recently, in 2014, a group of researchers claimed that a computer had for the first time passed the Turing test (University of Reading, 2014), a "test of a machine's ability to exhibit intelligent behavior equivalent to, or indistinguishable from, that of a human" (Wikipedia, 2015).

## Brief history of speech technology

Human-to-computer interaction relies on sophisticated computerized speech recognition algorithms. Research on speech recognition dates back to the 1930 when AT&T's Bell Labs began using computers to transcribe human speech, but it wasn't until the 1990s that it was available to the masses. In the late 1990s Dragon Naturally Speaking released a popular speech recognition software package to consumers, developed by Dragon Systems. Dragon was later purchased by Nuance who offers speech recognition applications for Windows and for mobile devices. More recently Mac OS X started shipping with Dictation, which allows users to speak text instead of typing. For additional information, a comprehensive list of speech recognition software can be found at wikipedia.org searching for: "List_of_speech_recognition_software". In the early days of speech recognition we were able to use voice commands to dial a phone number or take notes using speech-to-text software. Speech recognition has also been popular in the customer service sector or with information retrieval systems.

But it involves more than just speech recognition for a computer to exhibit intelligent language. The 'recognized' speech needs to be parsed and a reply must be generated and passed on to the user. These tasks are typically carried out using natural language processing, or using a simpler method that employs a textual database of top matching keywords to generate a computerized reply. Typically chatterbots or Artificial Conversational Entities can appear to hold intelligent conversations using a keyword matching technique. Popular chatbots include cleverbot.com, jabberwacky.com, and A.L.I.C.E. If the chatterbot is asked "What is your name?", then the reply will be selected from one of the records in the knowledgebase. Example code from a chatterbot using a keyword matching technique might look something like this:

```
record KnowledgeBase[] = {
  {"WHAT IS YOUR NAME",
   {"MY NAME IS CHATTERBOT.",
    "YOU CAN CALL ME CHATTERBOT.",
    "WHY DO YOU WANT TO KNOW MY NAME?"}
  },
```

The notion of speaking to a computer that is able to process your speech and provide functional feedback leads to enormous possibilities for language instruction. Computerized scoring of spoken language is perhaps the most promising and most developed technologies that integrates speech recognition and language learning. ETS's SpeechRater engine and Pearson's Versant are two automated speaking tests that make use of computerized scoring of spontaneous spoken responses. While large companies such as ETS and Pearson have been developing innovative speech recognition tools for language learning for some time now, it was not until recently that applications could be developed on a smaller scale, in fact even by language instructors themselves. With the arrival of cloud-based computing and Google's Web Speech API, anyone with an Internet connection and Chrome browser is able to take advantage of powerful speech synthesis (text-to speech or TTS), and speech recognition (speech-to-text or STT) tools. Since Google's Web Speech tool is cloud-based, speech recognition and speech synthesis are more predictable and reliable than with client-based applications since the speech processing isn't dependent on the local hardware and the operation system of the client. This is especially advantageous when working with mobile devices with limited hardware resources.

## Google's Web Speech API

Web Speech API is a JavaScript function that allows you to add speech recognition to any html web page. This API works in Chrome version 25 and later. Additional Web Speech API information can be found at https://dvcs.w3.org/hg/speech-api/raw-file/tip/speechapi.html.

The JavaScript Web Speech API uses the webkitSpeechRecognition object to transcribe speech.

## WebRTC

In addition to Google's Web Speech API, the WebRTC project provides a set of API tools to provide Real-Time Communications within browsers and mobile apps. WebRTC is a free, open project supported by Google, Mozilla and Opera and includes a powerful tool called getUserMedia, which allows the Chrome, Firefox or Opera web browsers to use a computer's camera and microphone to capture video and audio. More information about WebRTC can be found at http://www.webrtc.org/ and demos of getUserMedia in use can be found at: http://simpl.info/index.html.

## HTML5 audio capture

The first example here involves capturing and saving audio using only the web browser. Traditionally Adobe Flash was a popular choice for web-based audio capture. Now using HTML5 audio capture together with JavaScript, it is possible to capture a live audio stream from the device microphone using the getUserMedia() API. Firefox, Chrome and Chrome for Android all support this API, but unfortunately it is not supported on Apple iOS devices.

HTML audio capture enables us to create activities to capture audio and video from the user via the browser. If we simply want to capture live audio from the microphone and save it as an audio file, all from within a webpage, a JavaScript library named 'recorder.js' can be used.

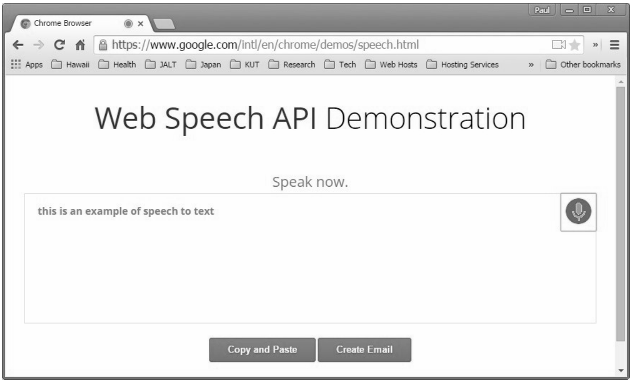
A demo of the HTML5 audio capture process can be found here: http://webaudiodemos.appspot.com/AudioRecorder/index.html.

A useful tutorial on capturing audio & Video in HTML5 can be found here: http://www.html5rocks.com/en/tutorials/getusermedia/intro/.

Finally, a tutorial on how to capture and convert audio to a MP3 file can be found here: https://nusofthq.com/blog/recording-mp3-using-only-html5-and-javascript-recordmp3-js/. In these examples a JavaScript library called 'Recorder.js' uses getUserMedia to capture the audio and save it as a .wav file. The .wav audio is not compressed so longer audio files can be very large. The audio can be compressed to an mp3 file, again in real time from within the browser using a separate JavaScript library called 'libmp3lame.js' The important element is that all of the tasks- capturing, converting and saving of the media are performed within the browser in real time.

## Using Google's Speech API for language learning activities

The following section describes Google's Web Speech API and introduces language learning activities that make use of the API. A demo of the Web Speech API is online at: https://www.google.com/intl/en/chrome/demos/speech.html



Web Speech API Demonstration

Speak now.

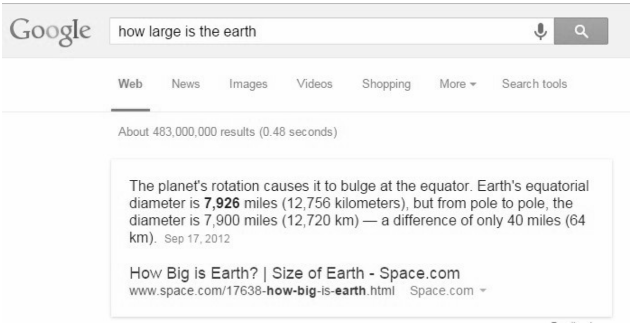
this is an example of speech to text

Figure 1. Google's Web Speech API demonstration

The demo shown in figure 1 uses a JavaScript Web Speech API specified by W3C to add speech recognition to the web page. There is a detailed introduction to this API at HTML-5Rocks, http://updates.html5rocks.com/2013/01/Voice-Driven-Web-Apps-Introduction-to-the-Web-Speech-API. Although this page is intended as a demo, language learners can use it to practice speaking phrases and sentences. Google's search page, illustrated in figure 2, allows users to search by speaking key words or by asking a question. A simple but engaging language learning activity can be created by collating a list of questions that have spoken answers by Google, and have learners ask these questions orally in order to complete a task by listening to the spoken replies generated by the search engine.

Below is a list of example search questions that Google provides oral answers for.

- What is the population of America?
- Who is the prime minister of the United Kingdom?
- What is the largest city in Australia?
- What is the diameter of the Earth?
- How large is the Earth?



how large is the earth

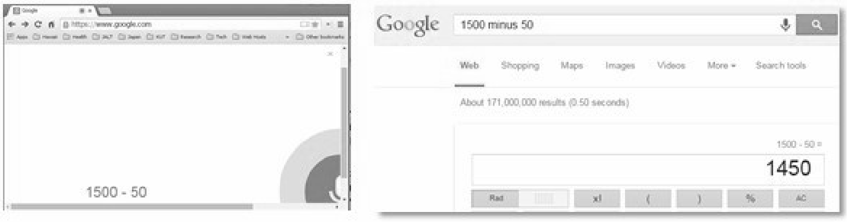
About 483,000,000 results (0.48 seconds)

The planet's rotation causes it to bulge at the equator. Earth's equatorial diameter is **7,926** miles (12,756 kilometers), but from pole to pole, the diameter is 7,900 miles (12,720 km) — a difference of only 40 miles (64 km). Sep 17, 2012

How Big is Earth? | Size of Earth - Space.com
www.space.com/17638-**how-big**-is-**earth**.html Space.com

Figure 2. Google's voice search with the Web Speech technology.

Tasks such as speaking numbers works well with Google search as shown in figure 3. Alternatively, this type of activity can be completed using Siri on an Apple iOS device.



1500 - 50

1500 minus 50

About 171,000,000 results (0.50 seconds)

1500 - 50 =

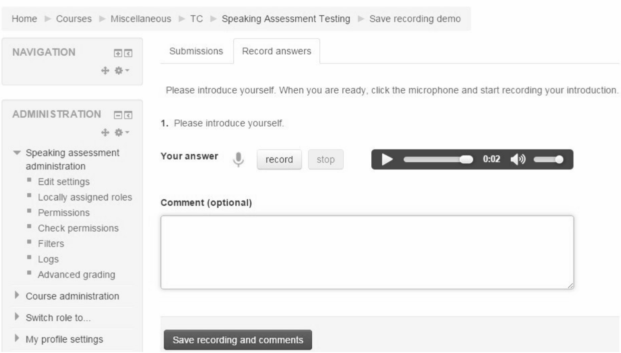
1450

Figure 3. Speaking numbers with Google voice search.

## Using the getUserMedia API and the Web Speech API with language learning applications

Google's Web Speech API can be used to add speech recognition and speech synthesis to language learning activities. Recently a number of language learning activities are being developed that utilize Google's Speech API. Several of these applications are described in the following section, including a simple web-based audio recorder, a voice shadowing application with speech recognition, a computerized oral assessment application and an app that learners can use to practice speaking skills with a  with a computer bot.

## Web-based recording



Home ▶ Courses ▶ Miscellaneous ▶ TC ▶ Speaking Assessment Testing ▶ Save recording demo

Please introduce yourself. When you are ready, click the microphone and start recording your introduction.

1. Please introduce yourself.

Your answer

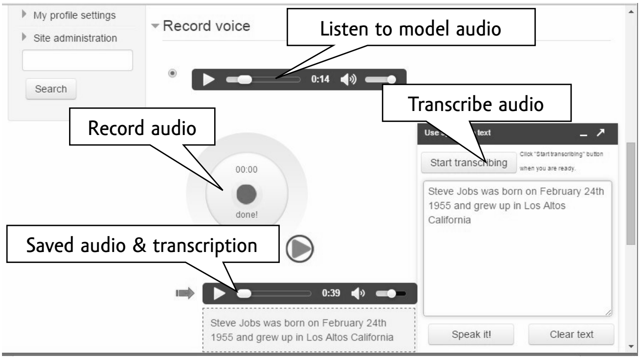
Comment (optional)

Figure 4. Simple web-based audio recording activity in Moodle

Web-based recording can be used in course management systems such as Moodle to easily capture and save audio input from learners. Figure 4 illustrates an activity where students are prompted to record a short self-introduction which is then posted to a shared course page. While this same task could be accomplished using a Flash-based audio recorder, this HTML5 solution is simpler in that it doesn't require any browser plugins. Audio capture and conversion is accomplished using the *record.js* and *libmp3lame.js* JavaScript libraries. Capturing audio from within the browser can be useful for simple tasks such as saving student speaking assignments, but a more practicable step would be to process the captured audio in order to provide specific feedback to the learner. A simple example would be to have the learner's recorded speech automatically transcribed and submitted along with their audio recording. A simple example of this process is demonstrated in the next section.

## VoiceShadow activity

The VoiceShadow activity, developed as a Moodle plugin, makes use of a number of Web Speech features to provide feedback to the learner while engaged in online speaking tasks. The task involves the learner simultaneously listening to a target audio file and recording or 'shadowing' the target language. This shadowing technique attempts to improve a learner's speaking confidence though a repeated listening shadowing process (Kumai & Daniels, 2013). The student-recorded speech samples along with the transcriptions can then be uploaded to a shared course page for teacher, peer and self-assessment. Using the Web Speech API, learners have the option to view a live transcription of their speech that appears in a pop-up window shown in figure 5.



Listen to model audio

Transcribe audio

Record audio

Saved audio & transcription

Click "Start transcribing" button when you are ready.

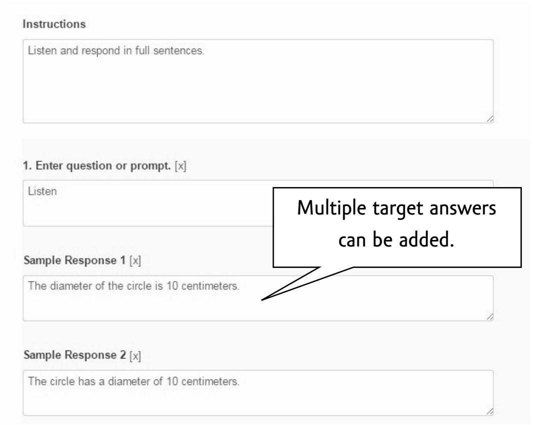
Steve Jobs was born on February 24th 1955 and grew up in Los Altos California

Steve Jobs was born on February 24th 1955 and grew up in Los Altos California

Figure 5. VoiceShadow activity

## Computerized assessment of speech

Automated speech assessment is another exciting area where the Web Speech API can be employed. Figures 6–8 illustrate a computerized speech assessment tool that was developed for Moodle. Using this tool, tasks can be created to capture, transcribe and analyze speaking samples from students. In effect, this activity enables teachers to administer computerized speaking assessments. Depending on the assessment algorithm, a speaking score can be automatically generated by comparing the transcribed text to the model answers. This automated assessment is typically more beneficial for closed ended questions that have a limited number of responses, for example, if a learner is asked to respond to a question while viewing an illustration that governs the possible answers. For example, if a student is asked to respond to the question "What is the diameter of the circle?" Possible answers may include "The diameter of the circle is 10 centimeters." or "The circle has a diameter of 10 centimeters." Figure 6 shows the target answers in the activity edit mode. Figures 7 and 8 illustrate the language input process and the transcription analysis. The speech assessment plugin requires Moodle 2.X and can be downloaded at: https://github.com/spnova/sassessment.



Instructions

Listen and respond in full sentences.

1. Enter question or prompt. [x]

Listen

Multiple target answers can be added.

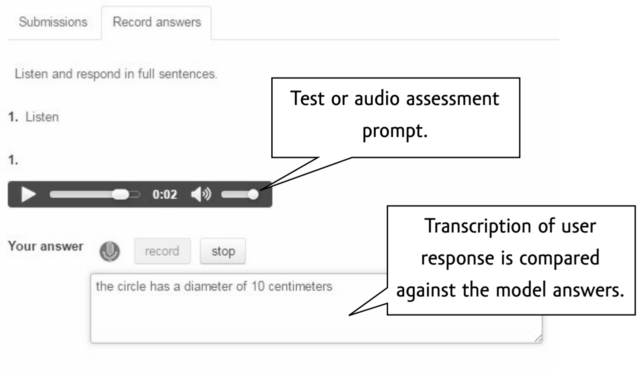
Sample Response 1 [x]

The diameter of the circle is 10 centimeters.

Sample Response 2 [x]

The circle has a diameter of 10 centimeters.

Figure 6. Adding speech analysis items in speech assessment tool.



Listen and respond in full sentences.

1. Listen

1.

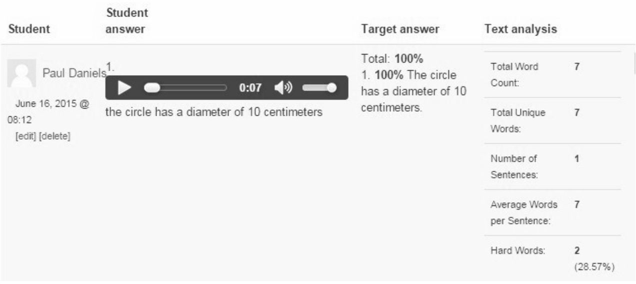
Test or audio assessment prompt.

Your answer

the circle has a diameter of 10 centimeters

Transcription of user response is compared against the model answers.

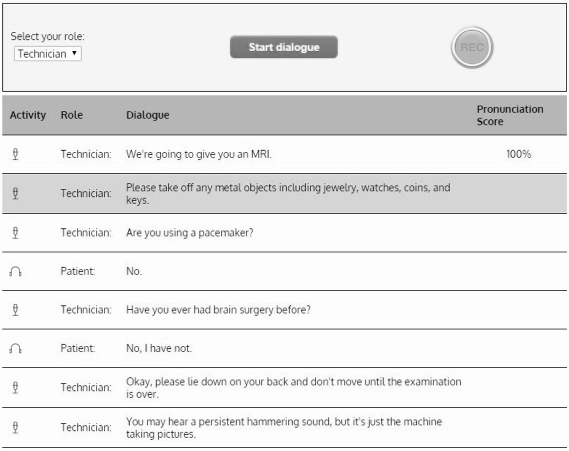
Figure 7. Recording responses in speech assessment tool.



| Student | Student answer | Target answer | Text analysis | |
|---|---|---|---|---|
| Paul Daniels<br>June 16, 2015 @ 08:12<br>[edit] [delete] | 1.<br>0:07<br>the circle has a diameter of 10 centimeters | Total: **100%**<br>1. **100%** The circle has a diameter of 10 centimeters. | Total Word Count: | 7 |
| | | | Total Unique Words: | 7 |
| | | | Number of Sentences: | 1 |
| | | | Average Words per Sentence: | 7 |
| | | | Hard Words: | 2<br>(28.57%) |

Figure 8. Analysis of transcription in speech assessment tool.

The Apps 4 EFL website developed an innovative web app that incorporates Google Web Speech API called Vocalyzer. This HTML5 web application encourages language learners to participate in a spoken dialogue. The learner listens to part of a dialogue and is then prompted to give a spoken reply, as shown in figure 9. If the transcription of the spoken reply matches the target answer, then the conversation continues. The student is also given a score that is calculated using a text comparison of the target language and the student language transcribed by Google. A demonstration of this web app can be found at: http://www.apps4efl.com/.

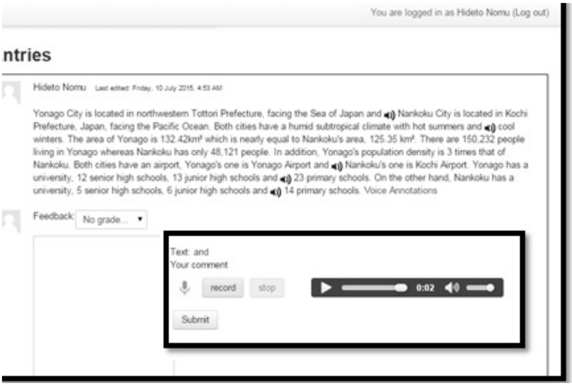


| Activity | Role | Dialogue | Pronunciation Score |
|---|---|---|---|
| 🎙 | Technician: | We're going to give you an MRI. | 100% |
| 🎙 | Technician: | Please take off any metal objects including jewelry, watches, coins, and keys. | |
| 🎙 | Technician: | Are you using a pacemaker? | |
| 🎧 | Patient: | No. | |
| 🎙 | Technician: | Have you ever had brain surgery before? | |
| 🎧 | Patient: | No, I have not. | |
| 🎙 | Technician: | Okay, please lie down on your back and don't move until the examination is over. | |
| 🎙 | Technician: | You may hear a persistent hammering sound, but it's just the machine taking pictures. | |

Figure 9. Vocalyzer web application

## Voice annotations

WebRTC can also be useful to insert voice annotations to text using the *getUserMedia* function. An example of how this can be used with text assignments submitted by students is illustrated in figure 10. This particular example illustrated a modified Moodle.org journal assignment. After a student has submitted a text assignment, the instructor can view the text online, highlight any part of the text and record a voice comment which is then automatically linked to the highlighted text.



You are logged in as Hideto Nomu (Log out)

ntries

Hideto Nomu Last edited: Friday, 10 July 2015, 4:53 AM

Yonago City is located in northwestern Tottori Prefecture, facing the Sea of Japan and 🔊 Nankoku City is located in Kochi Prefecture, Japan, facing the Pacific Ocean. Both cities have a humid subtropical climate with hot summers and 🔊 cool winters. The area of Yonago is 132.42km² which is nearly equal to Nankoku's area, 125.35 km². There are 150,232 people living in Yonago whereas Nankoku has only 48,121 people. In addition, Yonago's population density is 3 times that of Nankoku. Both cities have an airport, Yonago's one is Yonago Airport and 🔊 Nankoku's one is Kochi Airport. Yonago has a university, 12 senior high schools, 13 junior high schools and 🔊 23 primary schools. On the other hand, Nankoku has a university, 5 senior high schools, 6 junior high schools and 🔊 14 primary schools. Voice Annotations

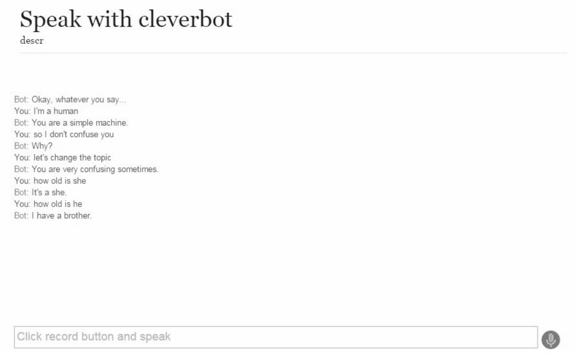
Feedback: No grade... ▾

Text: and
Your comment

Figure 10. Using WebRTC to add online voice annotations

## Chatting with a bot

The last example, in figure 11, illustrates the potential of using computerized bots for conversation practice. While a conversation with a bot may not be particularly engaging, it is an example of what the future may hold as far as human to computer interaction. This application uses speech recognition to convert spoken utterances into text which is then sent to a Chabot named CleverBot, who parses the text, compares it to a database of possible replies and sends back a text reply which is then converted to speech using text-to-speech technology. The entire process happens automatically so it feels as if you are holding a conversation with a real person. While the linguistic capabilities of Chatbots has been improving with native speakers, a Chatbot that is able to adapt language to accommodate the needs of a language learner is still a far way off (Coniam, 2008).



Speak with cleverbot
descr

Bot: Okay, whatever you say...
You: I'm a human
Bot: You are a simple machine.
You: so I don't confuse you
Bot: Why?
You: let's change the topic
Bot: You are very confusing sometimes.
You: how old is she
Bot: It's a she.
You: how old is he
Bot: I have a brother.

Figure 11: Web Speech technology being used to converse with Cleverbot.

## Web Speech API notes

### *1. Transcribing audio:*

Using the HTML5 Speech Recognition API, JavaScript has access to a browser's audio stream which is converted to text using Google's speech recognition engine and returned to the browser as raw text.

**Tools**: webkitSpeechRecognition API
**More information:**

- http://updates.html5rocks.com/2013/01/Voice-Driven-Web-Apps-Introduction-to-the-Web-Speech-API

### *2. Capturing audio:*

The 'recorder.js' Javascript libray can be used to capture audio from any input device. The audio stream is saved as a .wav file using getUserMedia. The .wav file can then be converted to an .mp3 file in real-time within the browser using 'libmp3lame.js'.

**Tools:** getUserMedia API, recorder.js and libmp3lame.js JavaScript libraries
**More information**:

- http://typedarray.org/from-microphone-to-wav-with-getusermedia-and-web-audio/
- http://www.smartjava.org/content/record-audio-using-webrtc-chrome-and-speech-recognition-websockets
- http://stackoverflow.com/questions/16810450/html5-getusermedia-record-audio-save-to-web-server-after-certain-time
- https://github.com/mattdiamond/Recorderjs

### *3. Capturing & transcribing audio:*

Audio capture is performed using Recorder.js as outlined in the previous example. The audio is then transcribed using Google's webkitSpeechRecognition API. The trick is that a python proxy is required to convert the captured wav audio file to flac – mono 22Hz, which is the format that Google's speech recognition engines requires. The transcribed text reply from Google's server then needs to be parsed.

**Tools:** *speech_recognition* module written in Python.
**More information:**

- https://github.com/Uberi/speech_recognition

**Other options**:

It is also possible to allow the browser to continually listen to audio input in real time, waiting for key phrases or commands to trigger an event. Here are two examples of this technique:

- https://www.talater.com/annyang/
- http://jqueryhouse.com/5-voice-control-javascript-libraries-for-developers

**Additional notes**

Web-based speech recognition tools are rapidly evolving. Sample code and APIs found on the Internet often become outdated and no longer operate.

If you wish to both save the recorded speech as an audio file and provide a transcription, then there are limitations. Audio must first be saved and converted and then sent to Google's server for transcription. The current API processes audio files differently from live speech input into the microphone.

Time limitations may exist with capturing and transcribing text. The technology outlined in the article typically works best with shorter spoken utterances. It can also take a long time to capture, process and convert raw audio using the JavaScript libraries.

Web pages using the Web Speech technology hosted on HTTPS web servers will not repeatedly ask for permission to access the local microphone, whereas HTTP hosted pages will repeatedly ask for access permission, therefore use HTTPS is possible.

## References

Coniam, D. (2008). Evaluating the language resources of chatbots for their potential in English as a second language. *ReCALL, 20* (2), 98–116.

Kumai, N. & Daniels, P. (2013). Moodle module and app development for shadowing Practice on Mobile Devices. 言語文化社会, *11*, 115–130.

Wikipedia. (2015, July 9). Turing test. Retrieved June 10, 2015, from https://en.wikipedia.org/w/index.php?title=Turing_test&oldid=670598902

University of Reading. (2014, June 8). Turing Test success marks milestone in computing history. Retrieved July 16, 2015, from http://www.reading.ac.uk/news-and-events/releases/PR583836.aspx