The Revolution will be Spoken

Adding Speech to Your Web App

Scott Davis
scott.davis@thoughtworks.com
@scottdavis99

Social profiles

@scottdavis99
scott.davis@thoughtworks.com

Scott Davis

Web architect & developer advocate

Scott Davis is a Principal Engineer with ThoughtWorks, where he focuses on leading-edge, emerging, and non-traditional aspects of web development. Scott specifically works on serverless web apps, mobile web apps (Responsive PWAs), HTML5-based SmartTV apps, Conversational UIs (like Siri and Alexa), and building IoT solutions using web technologies. Scott's focus on innovative web development has led him to his accessibility advocacy work, which includes educating developers on accessible web design and speaking about the importance of web accessibility for people with disabilities. Scott speaks at software conferences around the world (**It's Spelled "Accessibility," Not "Disability"**), explaining why accessibility should be just as important as mobile design strategy was 10 years ago.

2020 Trends In Computing

Why Web Design Is More Than Visuals

WEB DESIGN

The visual part of web design matters, sometimes even to the extent that a considerably disproportionate portion of the overall attention is focused on it, instead of content.

MYTH:
User Experience (UX) == Visual Design only

5 Digital Technology Trends for 2020

Trends

November 21, 2019 | Contributor: Manasi Sakpal

CIOs must understand how people interact with and experience digital technology along with understanding the technology itself.

Trend No. 1: Multiexperience

Multiexperience refers to the various devices and apps with which users interact on their digital journeys. This includes creating fit-for-purpose apps based on touchpoint-specific modalities while at the same time ensuring a consistent and unified user experience (UX) across web, mobile, **wearables**, conversational and immersive touchpoints.

New devices and new modes of interaction — from natural-language-based chat and voice to gestures used in 3D or virtual environments — coexist with the ever-popular web browser and mobile apps. Any combination of these new touchpoints can be used by customers along their journey.

CIOs should take the lead in promoting an IT and CX vision for multiexperience design and development that engages business partners in meaningful collaboration for the overall digital experience.

“Development teams should master **mobile app** design, development and architecture because ‘mobile’ is typically the gateway to multiexperience,” Sun says.

Trend No. 2: Interfaceless machines

Manufacturers across industries are abandoning on-machine controls or traditional interface models in favor of apps that run on their customers' mobile devices. Larger screens on mobile devices, high resolutions and rich device **APIs** allow for device control experiences far beyond what can be achieved with on-machine interfaces.

Interfaceless machines offer CIOs an entirely new perspective on digital product management. CIOs can also consider the possibility of extending digital product management capabilities to the digital extensions of the organization's main products.

Read more: Is Your Product Development Ready for Zero-Touch User Interfaces?

Trend No. 3: Agent interfaces

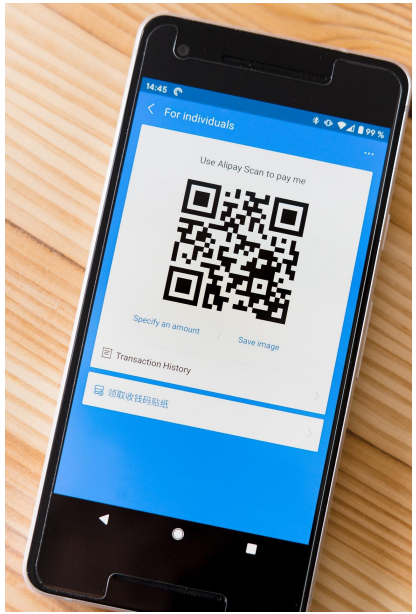
Agent interfaces represent a new paradigm of human-computer interaction and have broad implications that will greatly influence how enterprises interact with customers, offer services and provide tools to employees. Conversational UIs (or chatbots) are good examples. One of the most common applications of conversational UI is virtual assistants like Amazon Alexa.

With chatbots, the number of steps required to provide a command are reduced to a simple conversation. Agent interfaces employ **artificial intelligence** (AI) to predict what users intend to do using data from past interactions on a master interface.

"CIOs will need to build the skills required as agent interfaces become more mainstream," says Sun.

Trend No. 4: Facial recognition payment

Facial recognition payment is a digital experience trend emerging mainly in China, although it is quickly spreading elsewhere. It will increase the use of QR code payments and further diminish the use of bank cards and cash. This technology requires a high degree of confidence and trust in the provider. The trend is slowly gaining popularity outside China with Apple's Face ID with Apple Pay, with the trust existing between the user and the ecosystem provider (such as a bank).



10. 50% of all searches will be voice searches by 2020.

(Source: TheWebMaster)

- The Chinese market for voice search devices **accounted for 52%** of global smart speaker growth in 2018.
- The global voice-based smart speaker market could be worth **$30 billion by 2024**.
- The global voice search devices market **grew** 187% in Q2 of 2018.

Come next year every 5 out of 10 people will start searches with a search assistant. The smart speaker market is already growing steadily and should reach the $30 billion mark in 5 years' time. In fact, voice search stats 2018/2019 show that market growth is close to 200%!

Written Chinese

From Wikipedia, the free encyclopedia

Written Chinese (Chinese: 中文; pinyin: *zhōngwén*) comprises Chinese characters used to represent the Chinese language. Chinese characters do not constitute an alphabet or a compact syllabary. Rather, the writing system is roughly logosyllabic; that is, a character generally represents one syllable of spoken Chinese and may be a word on its own or a part of a polysyllabic word. The characters themselves are often composed of parts that may represent physical objects, abstract notions,[1] or pronunciation.[2] Literacy requires the memorization of a great number of characters: educated Chinese know about 4,000.[3][4] The large number of Chinese characters has in part led to the adoption of Western alphabets or other complementary systems as auxiliary means of representing Chinese.[5]

Chinese characters

Scripts

Precursors
Oracle-bone
Bronze
Seal (bird-worm · large · small)
Clerical · Regular · Semi-cursive · Cursive · Flat brush
Simplified characters

Type styles

Imitation Song · Ming · Sans-serif

Trend No. 5: Inclusive design

Inclusive design is the principle that the best way to serve the needs of the broad community is to consider the special **needs of all possible communities**.

Designers need to think about all potential users of the products and services that they design. The data sources used in their design efforts need to reflect all potential user segments and avoid datasets that are too narrow or non inclusive.

Top 14 general voice search statistics

1. By 2020, 50% of all searches across the internet will be voice-based.
2. By 2020, 30% of all searches will be done using a device without a screen.
3. In the US, house penetration for smart speakers was 13% in 2018.
4. In the US, house penetration for smart speakers is predicted to rise to 55% by 2022.
5. Voice search queries are longer than as compared with regular text-based searches.
6. Voice search queries tend to be three-to-five keywords in length.
7. 40% of the adults now use mobile voice search at least once daily.
8. 20% of the adults use mobile voice search at least once monthly.
9. Surprisingly, 9% of 55-64 surprisingly also use mobile voice search.
10. 20% of the searches on a mobile device are voice-based.
11. 25% of the queries on Android devices are voice-based.
12. 60% of smartphone users had tried voice search at least once in the past 12 months.
13. 55% of teenagers are using voice search daily basis.
14. Voice-based searches using a mobile phone are 3 times more likely to be location-specific.

Introducing Voice Control on Mac and iOS — Apple

1,158,308 views • Jun 3, 2019

Speech Synthesis (Text-to-Speech)

Speech synthesis

From Wikipedia, the free encyclopedia

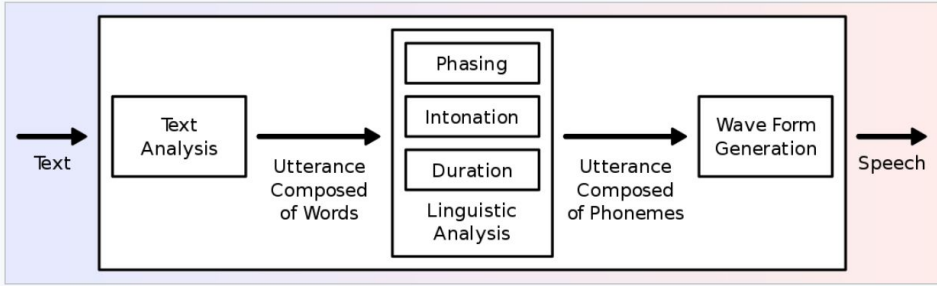
See also: Speech-generating device

Speech synthesis is the artificial production of human speech. A computer system used for this purpose is called a **speech computer** or **speech synthesizer**, and can be implemented in software or hardware products. A **text-to-speech** (**TTS**) system converts normal language text into speech; other systems render symbolic linguistic representations like phonetic transcriptions into speech.[1]

Synthesized speech can be created by concatenating pieces of recorded speech that are stored in a database. Systems differ in the size of the stored speech units; a system that stores phones or diphones provides the largest output range, but may lack clarity. For specific usage domains, the storage of entire words or sentences allows for high-quality output. Alternatively, a synthesizer can incorporate a model of the vocal tract and other human voice characteristics to create a completely "synthetic" voice output.[2]

The quality of a speech synthesizer is judged by its similarity to the human voice and by its ability to be understood clearly. An intelligible text-to-speech program allows people with visual impairments or reading disabilities to listen to written words on a home computer. Many computer operating systems have included speech synthesizers since the early 1990s.

A text-to-speech system (or "engine") is composed of two parts:[3] a front-end and a back-end. The front-end has two major tasks. First, it converts raw text containing symbols like numbers and abbreviations into the equivalent of written-out words. This process is often called *text normalization*, *pre-processing*, or *tokenization*. The front-end then assigns phonetic transcriptions to each word, and divides and marks the text into prosodic units, like phrases, clauses, and sentences. The process of assigning



Overview of a typical TTS system

Phoneme

From Wikipedia, the free encyclopedia

In phonology and linguistics, a **phoneme** /ˈfoʊniːm/ is a unit of sound that distinguishes one word from another in a particular language.

> **This article contains IPA phonetic symbols.** Without proper rendering support, you may see question marks, boxes, or other symbols instead of Unicode characters. For an introductory guide on IPA symbols, see Help:IPA.

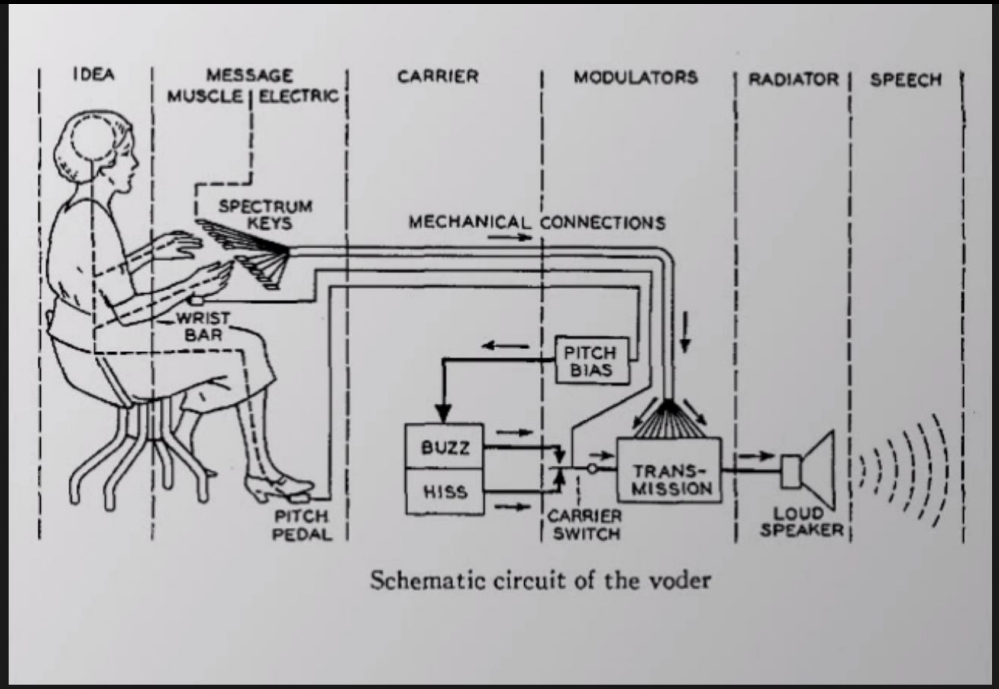
For example, in most dialects of English, with the notable exception of the west midlands and the north-west of England,[1] the sound patterns /sɪn/ (*sin*) and /sɪŋ/ (*sing*) are two separate words that are distinguished by the substitution of one phoneme, /n/, for another phoneme, /ŋ/. Two words like this that differ in meaning through the contrast of a single phoneme form a *minimal pair*. If, in another language, any two sequences differing only by pronunciation of the final sounds [n] or [ŋ] are perceived as being the same in meaning, then these two sounds are interpreted as variants of a single phoneme in that language.

Phonemes that are established by the use of minimal pairs, such as *tap* vs *tab* or *pat* vs *bat*, are written between slashes: /p/, /b/. To show pronunciation, linguists use square brackets: [pʰ] (indicating an aspirated *p* in *pat*).

Within linguistics, there are differing views as to exactly what phonemes are and how a given language

This is the under-the-hood view of the keyboard for the Voder (Voice Operating Demonstrator), the first electronic device capable of generating continuous human speech. It accomplishes this feat through a series of keys that generate the syllables, plosives, and affricatives normally produced by the human larynx and shaped by the throat and tongue. This week's film is a picture montage paired with the audio from the demonstration of the Voder at the 1939 World's Fair.

The Voder was created by one [Homer Dudley] at Bell Laboratories. He did so in conjunction with the Vocoder, which analyzes human-generated speech for encrypted transfer and re-synthesizes it on the other end. [Dudley] spent over 40 years researching speech at Bell Laboratories. His development of



Schematic circuit of the voder

In this film, the Voder is first demonstrated with a flat, robotic rendition of the phrase “she saw me”. The operator then runs through the various possible inflections to show the flavor that the foot pedal provides. Inside the Voder is a group of band pass filters in parallel that span the frequency range of human speech. Excitations are received from either the noise generator or the relaxation oscillator, and selection between the two is made from the wrist bar. The pitch is controlled with the foot pedal. The band pass outputs are fed to ten gain pots under the operators fingers. Three additional keys manipulate the excitations to produce the consonant stop sounds like /t/, /d/, /p/, /b/, /k/, and /g/.

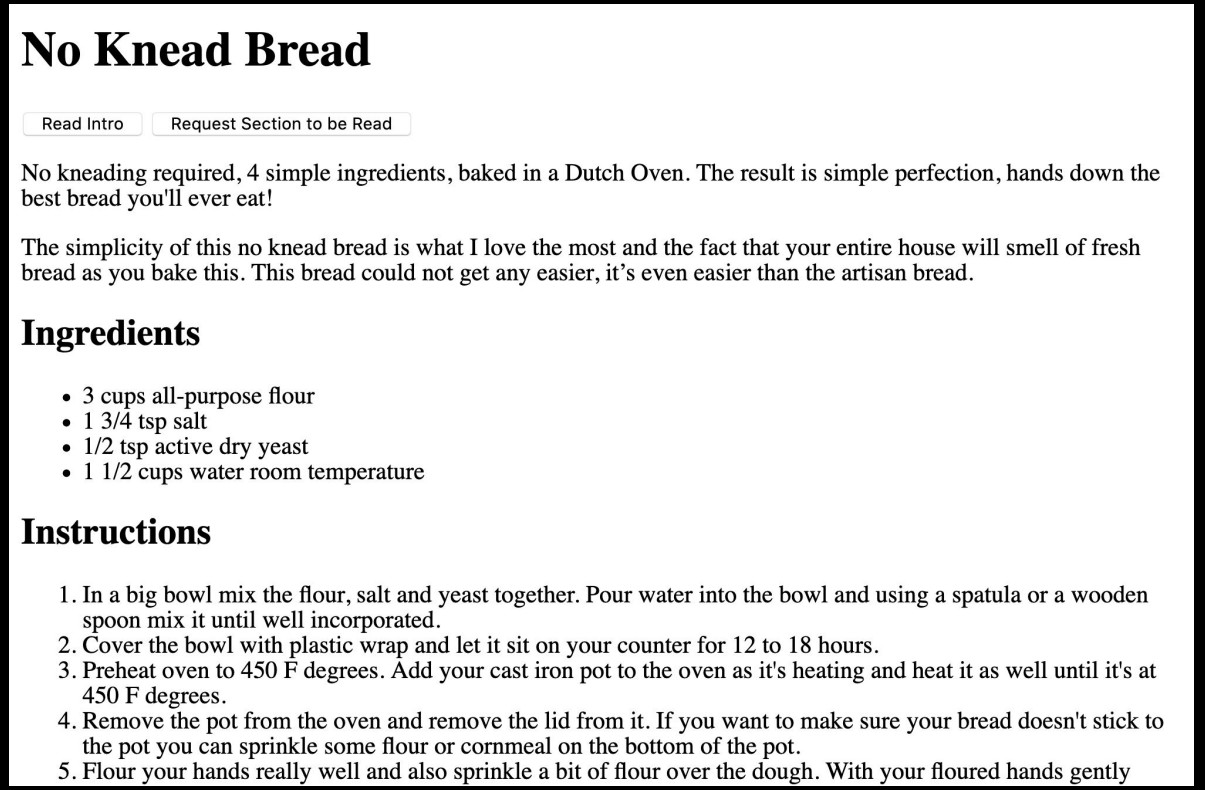
YouTube Search SIGN IN

ToneSpectra.com - Bell Lab's 'Voder' -An Early Vocoder Machine

5,964 views • Aug 21, 2010

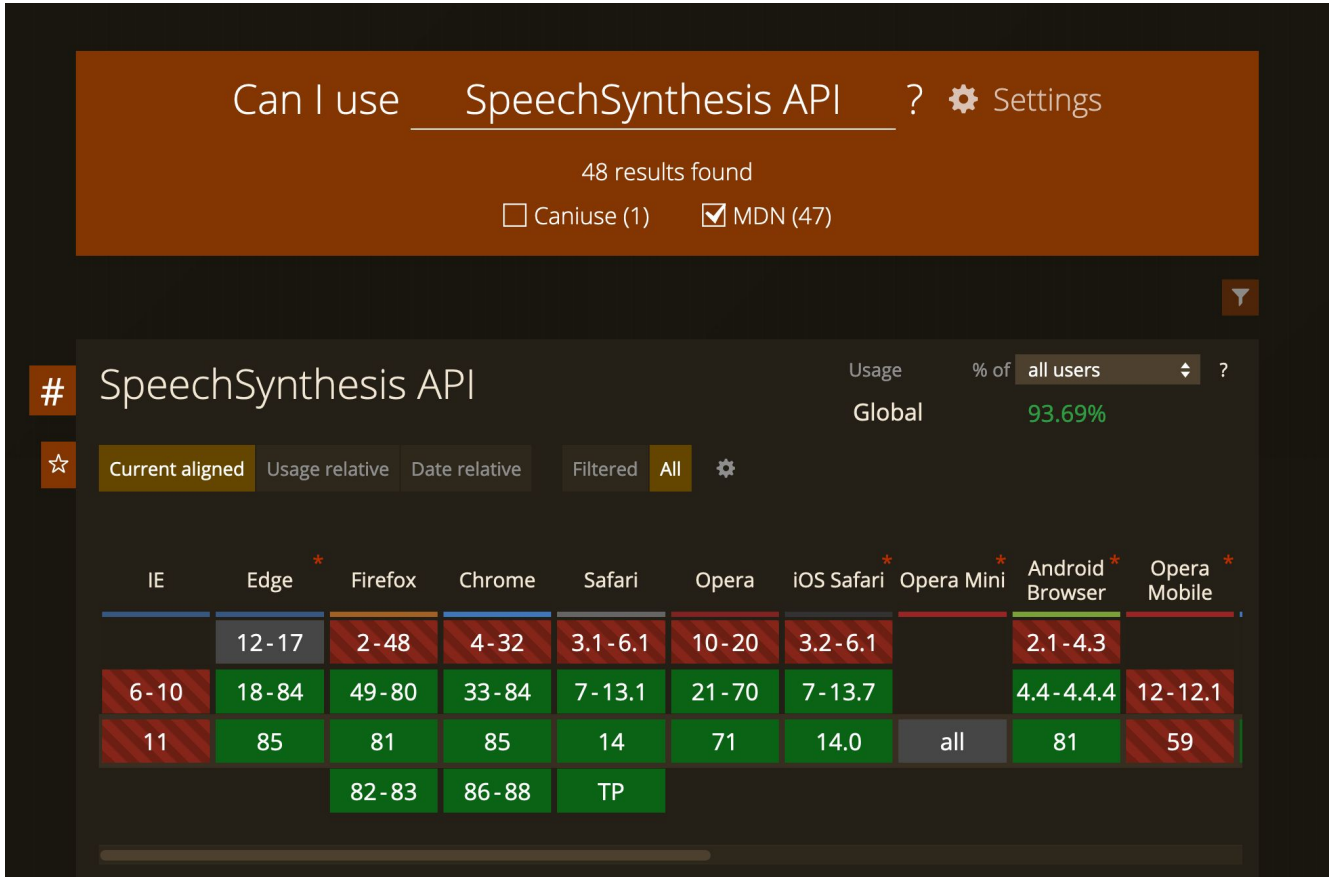
58 0 SHARE SAVE

Demo: SpeechSynthesis API



```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Recipe: No Knead Bread</title>

    <script>
      window.addEventListener('load', init);

      function init(){
        let b = document.querySelector('button');
        b.addEventListener('click', readIntro);
      }

      function readIntro(event){
        let paragraphs = document.querySelectorAll('p');
        for(let p of paragraphs){
          let msg = new SpeechSynthesisUtterance(p.innerText);
          window.speechSynthesis.speak(msg);
        }
      }
    </script>
  </head>

  <body>
    <h1>No Knead Bread</h1>
    <button>Read</button>

    <p>No kneading required, 4 simple ingredients, baked in a Dutch Oven.
    The result is simple perfection, hands down the best bread you'll ever
    eat!</p>
```

https://caniuse.com



Demo: https://thirstyhead.com/synthia/

Speech Synthesis Markup Language

From Wikipedia, the free encyclopedia

Speech Synthesis Markup Language (**SSML**) is an XML-based markup language for speech synthesis applications. It is a recommendation of the W3C's voice browser working group. SSML is often embedded in VoiceXML scripts to drive interactive telephony systems. However, it also may be used alone, such as for creating audio books. For desktop applications, other markup languages are popular, including Apple's embedded speech commands, and Microsoft's SAPI Text to speech (TTS) markup, also an XML language. It is also used to produce sounds via Azure Cognitive Services' Text to Speech API or when writing third-party skills for Google Assistant or Amazon Alexa.

SSML is based on the Java Speech Markup Language (JSML) developed by Sun Microsystems, although the current recommendation was developed mostly by speech synthesis vendors. It covers virtually all aspects of synthesis, although some areas have been left unspecified, so each vendor accepts a different variant of the language. Also, in the absence of markup, the synthesizer is expected to do its own interpretation of the text. So SSML is not a strict standard in the sense of C, or even HTML.

```
<!-- ?xml version="1.0"? -->
<speak xmlns="http://www.w3.org/2001/10/synthesis"
       xmlns:dc="http://purl.org/dc/elements/1.1/"
       version="1.0">
  <metadata>
    <dc:title xml:lang="en">Telephone Menu: Level 1</dc:title>
  </metadata>

  <p>
    <s xml:lang="en-US">
      <voice name="David" gender="male" age="25">
        For English, press <emphasis>one</emphasis>.
      </voice>
    </s>
    <s xml:lang="es-MX">
      <voice name="Miguel" gender="male" age="25">
        Para español, oprima el <emphasis>dos</emphasis>.
      </voice>
    </s>
  </p>

</speak>
```

MARKETS ▼ AMZN $3140.00 -4.88 0.16%

Amazon's Alexa Can Now Respond With Emotions

CONTRIBUTOR

RTTNews.com — RTTNews

PUBLISHED

NOV 28, 2019 8:32AM EST

(RTTNews) - Amazon said its voice assistant Alexa can now express emotions of either a happy/excited or a disappointed/empathetic tone in the US.

In a blog post, the technology and e-commerce giant said it introduced two new Alexa capabilities and new speaking styles for a more natural and intuitive voice experience.

Customers can use the newly published SSML tags to get started with Alexa emotions. They simply need to wrap Alexa's response with the appropriate SSML tag -'excited' or 'disappointed', and the level of intensity with which the emotion should be applied to the response.

Amazon's early customer feedback indicates that overall satisfaction with the voice experience were up 30 percent when Alexa responded with emotions. Further, while conducting 'blind listening' tests, the news style was perceived to be 31 percent more natural than Alexa's standard voice and the music style was perceived to be 84 percent more natural.

Prosody (linguistics)

From Wikipedia, the free encyclopedia

"Prosodic" redirects here. For other uses, see Prosody (disambiguation).

In linguistics, **prosody** is concerned with those elements of speech that are not individual phonetic segments (vowels and consonants) but are properties of syllables and larger units of speech, including linguistic functions such as intonation, tone, stress, and rhythm. Such elements are known as **suprasegmentals**.

Prosody may reflect various features of the speaker or the utterance: the emotional state of the speaker; the form of the utterance (statement, question, or command); the presence of irony or sarcasm; emphasis, contrast, and focus. It may otherwise reflect other elements of language that may not be encoded by grammar or by choice of vocabulary.

Part of a series on

Phonetics

[fəˈnɛtɪks]

Part of the Linguistics Series

Subdisciplines

Acoustic · Articulatory · Auditory

Articulation

Places of articulation

Labial · Dental · Alveolar · Postalveolar · Palatal · Velar · Uvular · Laryngeal

Manners of articulation

Consonant · Plosive · Affricate · Fricative ·

Generating Speech from SSML Documents

PDF | RSS

You can use Amazon Polly to generate speech from either plain text or from documents marked up with Speech Synthesis Markup Language (SSML). Using SSML-enhanced text gives you additional control over how Amazon Polly generates speech from the text you provide.

For example, you can include a long pause within your text, or change the speech rate or pitch. Other options include:

- emphasizing specific words or phrases
- using phonetic pronunciation
- including breathing sounds
- whispering
- using the Newscaster speaking style.

For complete details on the SSML tags supported by Amazon Polly and how to use them, see Supported SSML Tags

DECEMBER 23, 2019 • Leave a comment

Creating Voice Skills For Google Assistant And Amazon Alexa

15 min read

UI, Accessibility, User Experience, Interfaces

Share on Twitter or LinkedIn

Saved for offline reading

QUICK SUMMARY ↬ *Voice assistants are hopping out of emerging tech and into everyday life. As a front end developer, you already have the skills to build one, so let's dive into the platforms.*

OVER THE PAST DECADE, THERE HAS BEEN A SEISMIC SHIFT TOWARDS CONVERSATIONAL interfaces. As people reach 'peak screen' and even begin to scale back their device usage with digital wellbeing features being baked into most operating systems.

To combat screen fatigue, voice assistants have entered the market to become a preferred option for quickly retrieving information. A well-repeated stat states that 50% of searches will be done by voice in year 2020. Also, as adoption rises,

Speech Synthesis Markup Language (SSML)

Send feedback

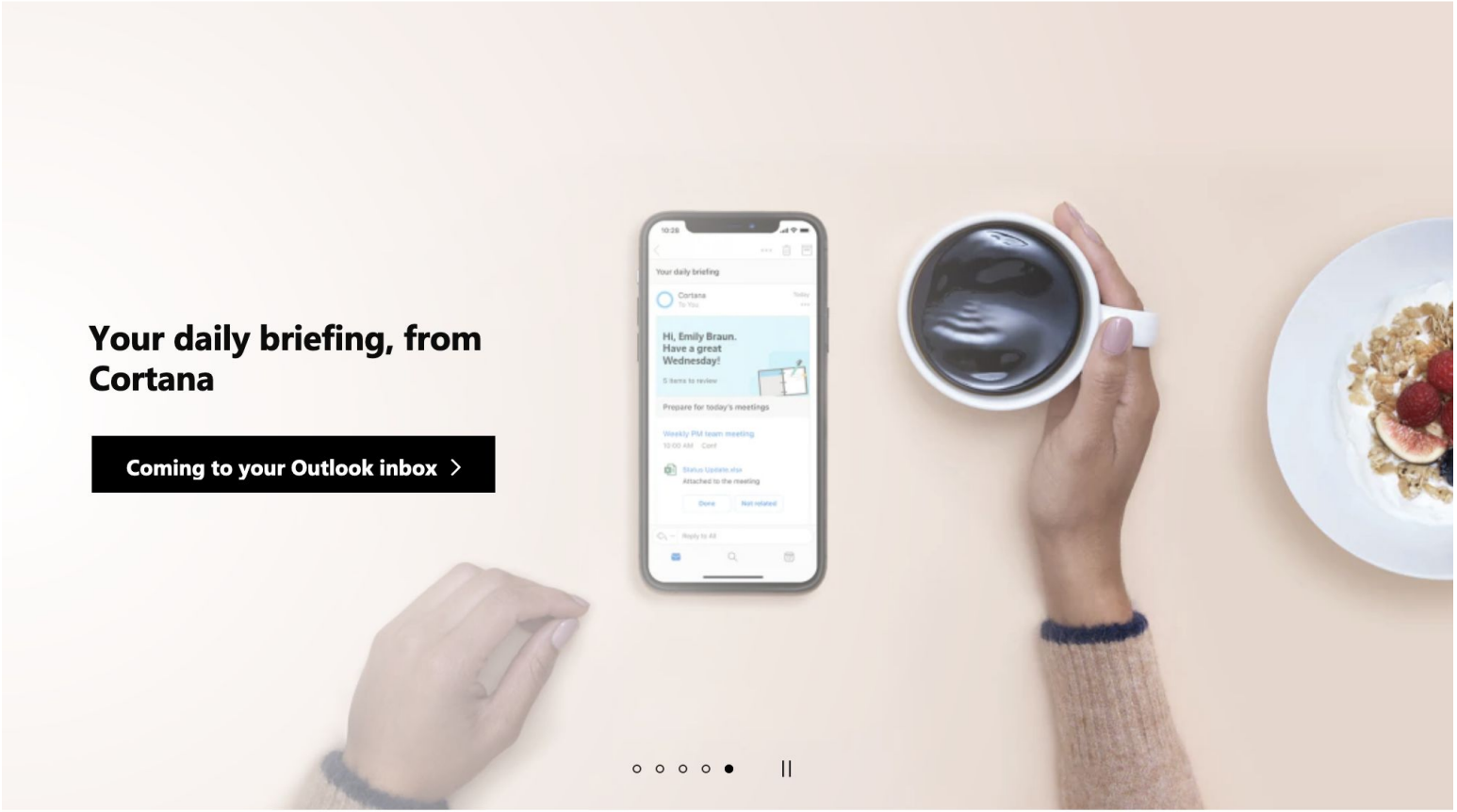
You can send Speech Synthesis Markup Language (SSML) in your Text-to-Speech request to allow for more customization in your audio response by providing details on pauses, and audio formatting for acronyms, dates, times, abbreviations, or text that should be censored. See the Speech-to-Text SSML tutorial for more information and code samples.

Note: SSML characters count toward character limits. See the quotas & limits page for more information.

The following shows an example of SSML markup and the Text-to-Speech synthesizes the text:

```
<speak>
  Here are <say-as interpret-as="characters">SSML</say-as> samples.
  I can pause <break time="3s"/>.
  I can play a sound
  <audio src="https://www.example.com/MY_MP3_FILE.mp3">didn't get your MP3 audio file</au
  I can speak in cardinals. Your number is <say-as interpret-as="cardinal">10</say-as>.
  Or I can speak in ordinals. You are <say-as interpret-as="ordinal">10</say-as> in line.
  Or I can even speak in digits. The digits for ten are <say-as interpret-as="characters"
  I can also substitute phrases, like the <sub alias="World Wide Web Consortium">W3C</sub
  Finally, I can speak a paragraph with two sentences.
  <p><s>This is sentence one.</s><s>This is sentence two.</s></p>
</speak>
```

Here is the synthesized text for the example SSML document:


Your daily briefing, from Cortana
Coming to your Outlook inbox
Your daily briefing
Hi, Emily Braun. Have a great Wednesday!
Prepare for today's meetings

Adding audio to Cortana skills

Whether you are developing a Cortana skill for a headless device (a device without a screen, such as the Harman Kardon Invoke speaker), or a device with a screen (such as a PC or mobile device), you can use a variety of audio features to enhance a user's experience.

In addition to Cortana's built-in text-to-speech technology, you can:

- Use Speech Synthesis Markup Language (SSML) to customize speech and embed short audio clips.
- Use an audio card to stream audio.

In Create your first Cortana skill and Building conversations you learned how to use the basic functionality of Cortana's text-to-speech technology. In this module you'll see how to extend the **Mixtape** skill developed in Building conversations to include additional audio elements.

Step 1 - Customize how Cortana speaks

In this step, you will use SSML to customize and enhance the **Mixtape** skill created in Building conversations. You can adjust a variety of speech attributes, including the speed at which Cortana speaks.

To increase the speed at which Cortana speaks, revise the **MixtapeIntent** method in the **BasicLuisDialog.cs** module of your **Mixtape** skill as follows:

Speech Recognition (Speech-to-Text)

DESIGNING THE INVISIBLE

For many, embarking on a voice UI (VUI) project can be a bit like entering the Unknown. Find out more about the lessons learned by William Merrill when designing for voice.

Read article →

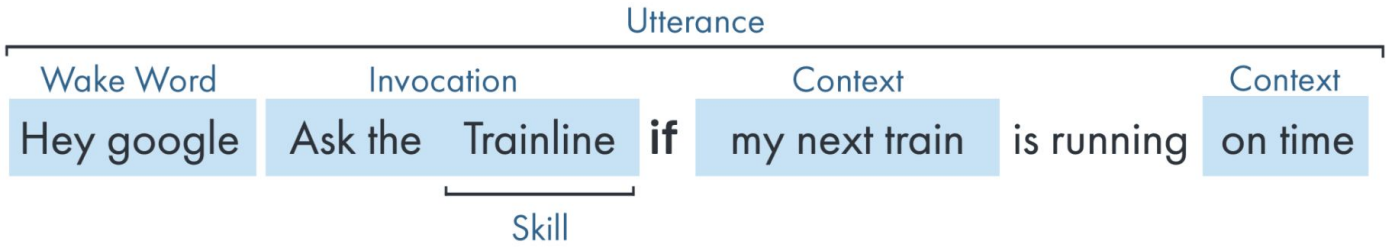
What Is A Conversational Interface?

A Conversational Interface (sometimes shortened to CUI, is any interface in a human language. It is tipped to be a more natural interface for the general public than the Graphic User Interface GUI, which front end developers are accustomed to building. A GUI requires humans to learn its specific syntaxes of the interface (think buttons, sliders, and drop-downs).

This key difference in using human language makes CUI more natural for people; it requires little knowledge and puts the burden of understanding on the device.

Commonly CUIs comes in two guises: Chatbots and Voice Assistants. Both have seen a massive rise in uptake over the last decade thanks to advances in Natural Language Processing (NLP).

UNDERSTANDING VOICE JARGON



(*Large preview*)

Keyword	Meaning
Skill/Action	A voice application, which can fulfill a series of intents
Intent	Intended action for the skill to fulfill, what the user wants the skill to do in response to what they say.
Utterance	The sentence a user says, or utters.
Wake Word	The word or phrase used to start a voice assistant listening, e.g. ‘Hey google’, ‘Alexa’ or ‘Hey Siri’

How an Alexa Skill Works

An Alexa skill has both an interaction model—or voice user interface—and application logic. When a customer speaks, Alexa processes the speech in the context of your interaction model to determine the customer request. Alexa then sends the request to your skill application logic, which acts on it. You provide your application logic as a back-end cloud service hosted by Alexa, AWS, or another server.

Next-generation Google Assistant runs in real time on your phone

By Simon Hill
May 7, 2019

Google just showed off some impressive advances that it's bringing to Google Assistant in the near future. By shrinking down speech-recognition and language-understanding models, it has found a way to squeeze 100GB of models from the cloud down to less than half a gigabyte that can run locally on your phone.

The demo at Google I/O showed off the Google Assistant running on a phone and performing tasks with almost no latency. You can ask Google Assistant to launch apps, reply to texts, find photos, compose emails, and more without any delay and without the need for a network connection.

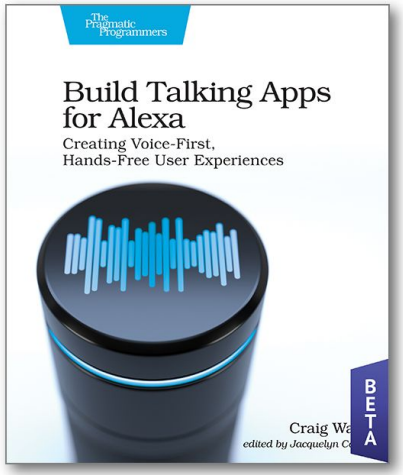
DeepSpeech 0.6: Mozilla's Speech-to-Text Engine Gets Fast, Lean, and Ubiquitous

By Reuben Morais

Posted on December 5, 2019 in Audio, Featured Article, and Speech

The Machine Learning team at Mozilla continues work on DeepSpeech, an automatic speech recognition (ASR) engine which aims to make speech recognition technology and trained models openly available to developers. DeepSpeech is a deep learning-based ASR engine with a simple API. We also provide pre-trained English models.

Our latest release, version v0.6, offers the highest quality, most feature-packed model so far. In this overview, we'll show how DeepSpeech can transform your applications by enabling client-side, low-latency, and privacy-preserving speech recognition capabilities.



TWEET THIS

Build Talking Apps for Alexa

Creating Voice-First, Hands-Free User Experiences

by Craig Walls

Voice recognition is here at last. Alexa and other voice assistants have now become widespread and mainstream. Is your app ready for voice interaction? Learn how to develop your own voice applications for Amazon Alexa. Start with techniques for building conversational user interfaces and dialog management. Integrate with existing applications and visual interfaces to complement voice-first applications. The future of human-computer interaction is voice, and we'll help you get ready for it.

How To Build A Custom Amazon Alexa Skill, Step-By-Step: My Favorite Chess Player

Uros Ralevic Follow
Jul 24, 2018 · 17 min read

```
{
    "interactionModel": {
        "languageModel": {
            "invocationName": "chess players",
            "intents": [
                {
                    "name": "AMAZON.FallbackIntent",
                    "samples": []
                },
                {
                    "name": "AMAZON.CancelIntent",
                    "samples": []
                },
                {
                    "name": "AMAZON.HelpIntent",
                    "samples": []
                },
                {
                    "name": "AMAZON.StopIntent",
                    "samples": []
                },
                {
                    "name": "playerBio",
                    "slots": [
                        {
                            "name": "player",
                            "type": "playerNames"
                        }
                    ],
                    "samples": [
                        "{player}",
                        "who is {player}",
                        "tell me about {player}"
                    ]
                },
                {
                    "name": "AMAZON.NoIntent",
                    "samples": []
                }
```

The events are encoded in json format by the skill interface in accordance with the interaction model. The responses sent by the AWS service to the skill interface are also encoded in json format.

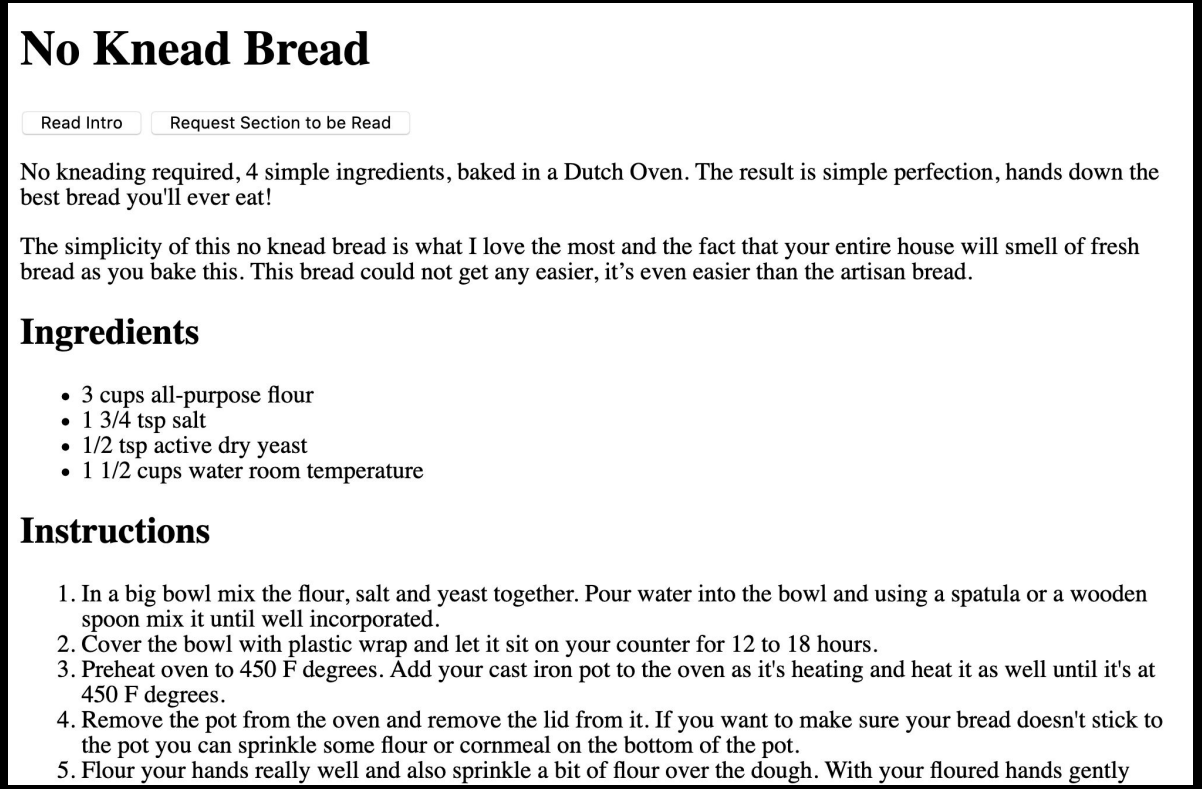
For our simple custom skill we'll configure the Lambda function using Python, which is supported on the AWS services.

The response of our Lambda function is going to be in the following format:

```
{
    "body": {
        "version": "1.0",
        "response": {
            "outputSpeech": {
                "type": "PlainText",
                "text": ""
            },
            "card": {
                "type": "Simple",
                "title": "",
                "content": ""
            },
            "reprompt": {
                "outputSpeech": {
                    "type": "PlainText",
                    "text": ""
                }
            },
            "shouldEndSession": value
        }
    }
```

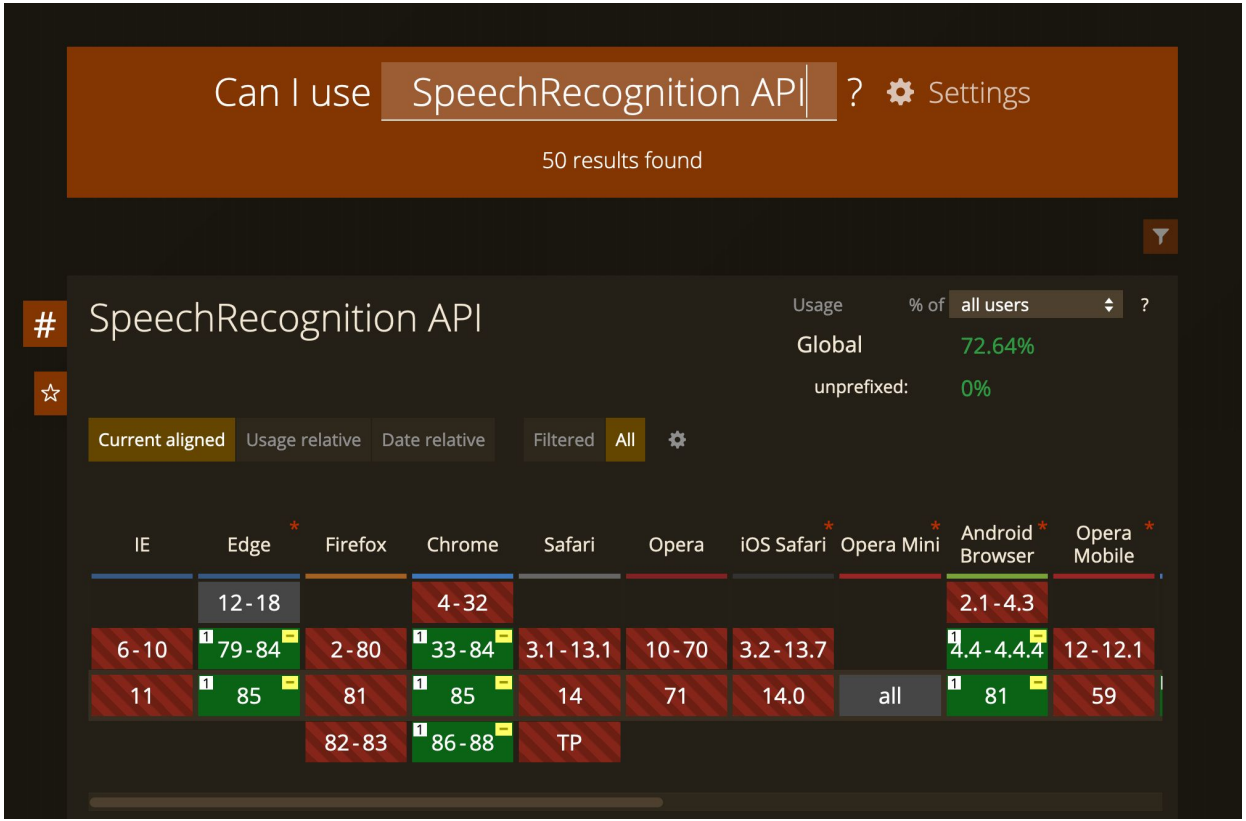
```
{
  "outputSpeech": {
    "type": "SSML",
    "ssml": "<speak>Garry Kimovich Kasparov (April 3, 1963) is a Russian chess grandmaster, former world chess champion, writer, and political activist, who many consider to be the greatest chess player of all time.</speak>"
  },
  "shouldEndSession": false,
  "reprompt": {
    "outputSpeech": {
      "type": "SSML",
      "ssml": "<speak>Would you like to learn about another chess player?<break time='500ms'/>Just say, "Tell me about (the chess player's name)."</speak>"
    }
  }
}
```

Demo: SpeechRecognition API



```
47      ////////////////////////
48      // Speech Recognition
49      ////////////////////////
50      let recognition = new webkitSpeechRecognition();
51
52      function startListening(){
53        recognition.start();
54      }
55
56      recognition.addEventListener('result', (event) => {
57        let results = event.results[0][0].transcript;
58        let recognitionOutput = document.querySelector('#recognition-output');
59        recognitionOutput.innerHTML = `Recognition results: ${results}`;
60
61        if(results === 'ingredients'){
62          readIngredients();
63        }else if(results === 'instructions'){
64          readInstructions();
65        }
66      });
67
68      recognition.addEventListener('speechend', (event) => {
69        recognition.stop();
70      });
```

https://caniuse.com



Demo: https://littlepath.org/zoom-cc/

Demo: Zoom Closed Captioning

To add closed captions to your Zoom call:

1. As the host of a Zoom meeting, click the **CC (Closed Caption)** button in your Zoom toolbar
2. Click the **Copy the API token** button under **"Use a 3rd party CC service"**.
3. Paste it into the **"Zoom CC API Token"** field on this page.
4. Press the **Start CC** button on this page to start closed captions.

You should see the transcript appear on this page as well as in your Zoom call.

NOTE: If you are the host of a Zoom call and you don't see a "CC (Closed Caption)" button in your Zoom toolbar, follow Zoom's instructions to enable closed captioning: https://support.zoom.us/hc/en-us/articles/207279736-Using-closed-captioning

Zoom CC API Token

https://wmcc.zoom.us/closedcaption?id=97679531206&ns=U2NvdHQgRGF2aXMnIFpvb20gTWVldGluZw&expire=86400&sparam

Language

en-US (en-US for American English, es-MX for Mexican Spanish, etc.)

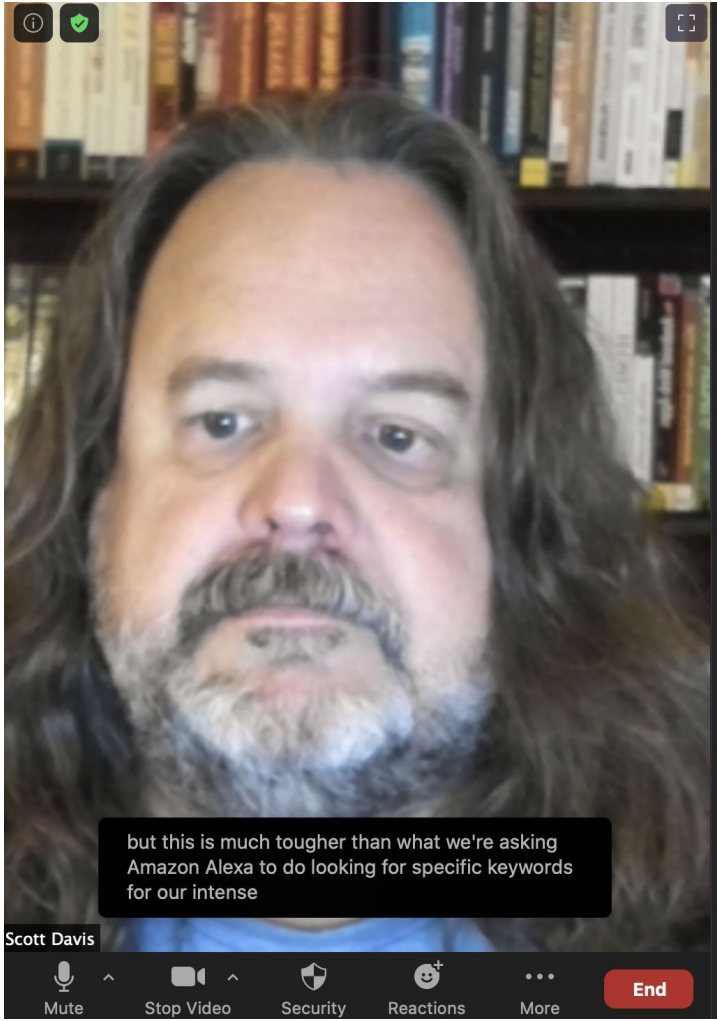
Stop CC

hi my name is Scott Davis

I'm demonstrating closed captioning using the Google cloud services

V Club services use the speech recognition API stand Eyes by the World Wide Web Consortium

as you can see the speech recognition isn't perfect but this is much tougher than what we're asking Amazon Alexa to do looking for specific keywords for our intense



Transcript

Search

16:10:32

hi my name is Scott Davis

16:10:38

I'm demonstrating closed captioning using the Google cloud services

16:10:48

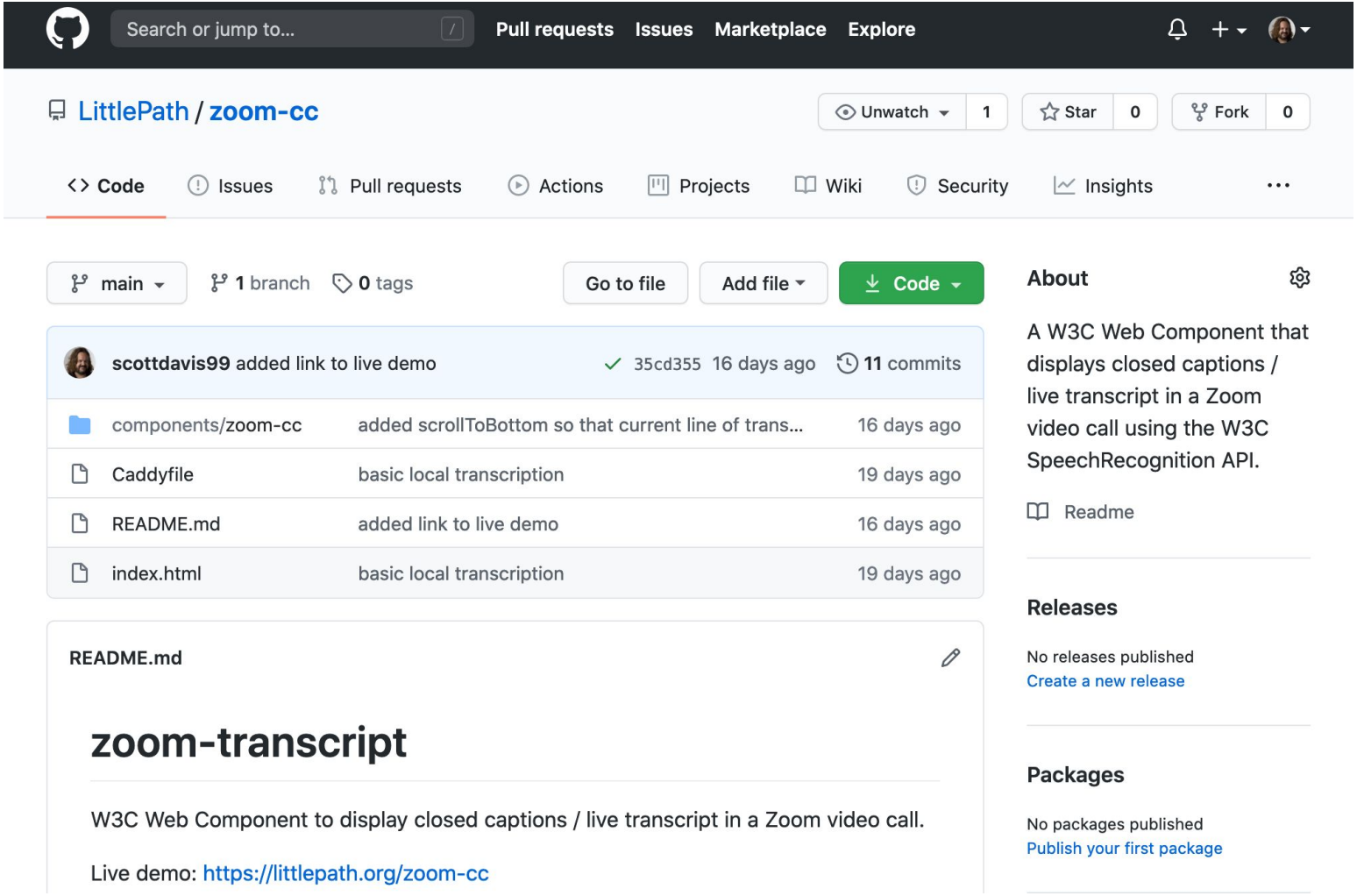
V Club services use the speech recognition API stand Eyes by the World Wide Web Consortium

16:14:51

as you can see the speech recognition isn't perfect but this is much tougher than what we're asking Amazon Alexa to do looking for specific keywords for our intense

Save Transcript

Powered by Otter.ai


Search or jump to...
Pull requests
Issues
Marketplace
Explore
LittlePath / zoom-cc
Unwatch 1
Star 0
Fork 0
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
main
1 branch
0 tags
Go to file
Add file
Code
scottdavis99 added link to live demo
35cd355 16 days ago
11 commits
components/zoom-cc
added scrollToBottom so that current line of trans...
16 days ago
Caddyfile
basic local transcription
19 days ago
README.md
added link to live demo
16 days ago
index.html
basic local transcription
19 days ago
README.md
zoom-transcript
W3C Web Component to display closed captions / live transcript in a Zoom video call.
Live demo: https://littlepath.org/zoom-cc
About
A W3C Web Component that displays closed captions / live transcript in a Zoom video call using the W3C SpeechRecognition API.
Readme
Releases
No releases published
Create a new release
Packages
No packages published
Publish your first package

```
111       start(){
112         let startStopButton = this.shadowRoot.querySelector('#start-stop');
113         const SpeechRecognition = window.SpeechRecognition || window.webkitSpeechR
114
115         if(typeof SpeechRecognition === "undefined"){
116           startStopButton.disabled = true;
117           this.write(`This browser doesn't support the <a href="https://developer.
118         }else{
119           startStopButton.innerHTML = 'Stop CC';
120           startStopButton.classList.add('stop');
121
122           this.recognition = new SpeechRecognition();
123           this.recognition.continuous = true;
124           this.recognition.interimResults = false;
125           this.recognition.onresult = (event) => {
126             const result = event.results[event.resultIndex][0].transcript;
127             this.write(result);
128             this.postToZoom(result);
129             this.scrollToBottom();
130           }
131           this.recognition.onerror = (event) => {
132             console.log(event);
```

```
156       async postToZoom(message){
157         let zoomURL = this.shadowRoot.querySelector('#zoomURL').value;
158         let zoomLanguage = this.shadowRoot.querySelector('#zoomLanguage').value;
159         zoomURL += `&lang=${zoomLanguage}`;
160         zoomURL += `&seq=${this.sequence++}`;
161
162         const options = {
163           method: 'POST',
164           mode: 'no-cors',
165           body: message,
166           headers: {
167             'Content-Type': 'plain/text',
168             'Content-Length': message.length
169           }
170         };
171
172         const response = await fetch(zoomURL, options);
173         return response;
174       }
175
```

Getting Alexa to Respond to Sign Language Using Your Webcam and TensorFlow.js

TensorFlow Follow
Aug 8, 2018 · 11 min read

By Abhishek Singh

A few months ago, while lying in bed one night, a thought flashed through my head — “If voice is the future of computing interfaces, what about those who cannot hear or speak?”. I don’t know what exactly triggered this thought, I myself can speak and hear and have no one close to me who is either deaf or mute, nor do I own a voice assistant. Perhaps it was the countless articles popping up on the proliferance of voice assistants, or the competition between large companies to become your voice activated home assistant of choice, or simply seeing these devices more frequently on the counter tops of friends’ homes. As the question refused to fade from memory, I knew it was an itch I needed to scratch.

While I could have simply released the code, I instead chose to post a video demonstrating the system, since I feel a lot of machine learning projects lack a visual element, making it difficult for people to relate to and understand them. Also I hoped this approach would help switch the focus away from the tech element of the project and instead shine light on the human element — that it's not about the underlying tech but the capabilities that such tech provides us as humans.

Top highlight

Now that the video is out, this blog post takes a look at the underlying tech

Welcome!

People with speech impairments often find that speech recognition systems don't reliably understand them.

Project Euphonia, a Google Research project, is focused on improving how such systems recognize impaired speech... including from people with conditions such as Down syndrome, stroke, traumatic brain injury, cerebral palsy or ALS.

Improvements to speech recognition depend upon analyzing impaired speech. **If you have a voice that is difficult to understand due to a condition (but not just because of an accent), you can help!** Please fill out this form to contribute to this research effort by recording a set of phrases. We provide cash gift cards to those who complete recording sets.

Record phrases for Google Research

Yes, that's Robert Downey, Jr. talking about speech recognition and speech synthesis and machine learning and Google's Project Euphonia and…
Trust me -- it's good!

Conclusion

MYTH:
User Experience (UX) == Visual Design only

5 Digital Technology Trends for 2020

Trends

November 21, 2019 | Contributor: Manasi Sakpal

CIOs must understand how people interact with and experience digital technology along with understanding the technology itself.

Introducing Voice Control on Mac and iOS — Apple

1,158,308 views • Jun 3, 2019

The Revolution will be Spoken

Adding Speech to Your Web App

Scott Davis
scott.davis@thoughtworks.com
@scottdavis99