Improving Japanese English pronunciation with speech recognition and feedback system

Kanta Igarashi¹,∗ and Ian Wilson¹,∗∗

¹CLR Phonetics Lab, University of Aizu, Tsuruga, Ikki-machi, Aizuwakamatsu, Fukushima-ken, Japan

Abstract. For Japanese people, communicating with English speakers from abroad has become more common because of internationalization, and there are many people who want to improve their English-speaking skills. However, there are few environments where we can speak English outside of the classroom, so Japanese students rarely have a chance to study English pronunciation. Even if students do have a chance to take an English pronunciation class, teachers do not have enough time to individually teach each student pronunciation in a big class. Because of that, computers and smartphones may be one good type of tool to solve this problem. In this research, we develop a web-based application to help Japanese learners with their English pronunciation.

1 Introduction

According to Dizon [1], intelligent assistants such as Siri or Alexa, which use speech recognition, could be helpful tools for language learners, especially Japanese second-language (L2) English learners, who usually do not have many opportunities to use English outside of the classroom. Daniels and Iwago [2] have compared Siri to Google Speech Recognition (GSR) and found that GSR is more accurate for transcribing Japanese undergraduates’ L2 English speech. However, although intelligent assistants can recognize speech, they do not advise users about their pronunciation. So, the current research is focused on creating a web-based feedback system that would be helpful for language learners. The feedback system advises users from the point of view of articulation and correctness based on typical errors in real test results that we collected. In this research, the web application with feedback system to evaluate users’ English pronunciation uses Mozilla Developer Network (MDN)’s Web Speech API [3]. In order to make the range of possible feedback more manageable, the words and phrases used for this web application are limited. They are chosen from the “Wolf Story” [4], a phonetic passage adapted from an Aesop fable, because of the wide range of English phonemes and different combinations of phonemes. It is easy for teachers and students to access the system, because the web application is hosted on the University of Aizu’s CLR Phonetics Lab website at http://clrlab1.u-aizu.ac.jp/acoustics.html.

2 Speech Recognition Interface

There are many speech recognition applications, so it is possible to choose an interface according to one’s individual requirements. For example, Google Speech Recognition (GSR), IBM Cloud, Siri, Alexa, and Speech Recognition API (MDN) are all popular. As previously stated, GSR is better for L2 English learners than Siri and Alexa. So, we compared GSR, IBM, and MDN’s API. Table 1 shows a list of these APIs’ advantages and disadvantages.

<table>
<thead>
<tr>
<th></th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>GSR</td>
<td>● High accuracy</td>
<td>● Need account, many source files, must pay according to use</td>
</tr>
<tr>
<td>IBM</td>
<td></td>
<td>● Need account, must pay according to use</td>
</tr>
<tr>
<td>MDN</td>
<td>● Only need Javascript</td>
<td>● Accuracy is lower</td>
</tr>
<tr>
<td></td>
<td>● Free</td>
<td>● Only works with Chrome</td>
</tr>
</tbody>
</table>

3 Method

The web application created in this research consists of HTML, CSS and JavaScript. First, the HTML and CSS functions are “recognition start button”, “sample mp3 files control”, etc. Javascript controls the speech recognition and feedback system. The Web Speech API is an interface that Mozilla Developer Network (MDN) provides and this API can recognize any language. In this web application, we selected American English as the language setting because much of the English education in Japanese primary and secondary schools seems to focus on American English. In addition, the application needs users to install and use Google Chrome because MDN’s Web Speech API works on Google Chrome only (this API does not work on iOS).
4 Web Application Details

This web application uses Web Speech API [3]. A part of the web application is shown in Figure 1. The application includes a feedback system and a lecture page. The feedback system must give advice to users, so we created an advice dataset corresponding to words from test participants that were commonly misunderstood by the speech recognition system. In case the recognized word is not in the dataset, the feedback given is very general (“please try again and speak more clearly”).

Figure 1. Part of the “word practice” page of the resultant web application

The lecture section of the application contains three pages. First, an explanation of what the International Phonetic Alphabet (IPA) is. Second and third, lessons teaching how to pronounce vowels and consonants, respectively. For each type of sound, static MRI images of articulators producing the sounds can be seen. Although this study’s purpose is a complete web application with feedback system, we also made a lecture page to help with the study of English pronunciation. This application is mainly for Japanese L2 English learners, so the lecture page is written in Japanese. According to Ono [5], the way of effective pronunciation teaching is that the student repeats the teacher’s English pronunciation and the teacher explains, in words and illustrations, how to pronounce.

4.1 Feedback System

The web application can return advice for a word from a list of common errors by Japanese speakers. Figures 2, 3, and 4 show examples of feedback. In case the recognized word is not in the list, the application returns common advice. For example, if the word is missing “th” sound, the application returns appropriate advice such as “Put your tongue tip between your teeth to make “th” sound.”

Figure 2. Example showing feedback given to the user for “th” sound

4.1.1 Common Pronunciation Errors

There are many common errors that Japanese speakers make when speaking English as a second language. Table 2 and Figure 5 show the consonants and vowels, respectively, that exist in Japanese and English. In Table 2, consonants only used in English are shown in blue and consonants only used in Japanese are shown in red. Also, consonants used in both English and Japanese are shown in black. Common pronunciation errors by Japanese speakers of English usually occur in vowels and consonants that are not found in Japanese (i.e., the ones shown in blue). In Figure 5, the vowels underlined in pink are those found in North American English but not in Japanese.

Figure 3. Example showing feedback given to the user for a vowel sound

Figure 4. Example showing feedback given to the user for “i” sound

4.1.2 Data Collection

In order to build a feedback system, we collected real examples of the Web Speech API’s transcriptions of Japanese speakers’ English utterances. In order to collect test samples, we created a website that recognizes users pronunciation and sends data by email to us. Participants were mainly students of the University of Aizu. Anyone could access the website easily because it is hosted on the University of Aizu’s CLR Phonetics Lab website at http://clrlab1.u-aizu.ac.jp/acoustics.html. The data collected can be seen in Table 3. The target word is shown
Table 2. International Phonetic Alphabet chart of consonants showing differences between English and Japanese. Red phonemes are found only in Japanese, blue ones are found only in English, and black ones are found in both languages.

<table>
<thead>
<tr>
<th></th>
<th>Bilabial</th>
<th>Labiodental</th>
<th>Dental</th>
<th>Alveolar</th>
<th>Postalveolar</th>
<th>Palatal</th>
<th>Velar</th>
<th>Glottal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plosive</td>
<td>p</td>
<td>b</td>
<td>t</td>
<td>d</td>
<td>k</td>
<td>g</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Nasal</td>
<td>m</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Trill</td>
<td>r</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tap or Flap</td>
<td>f</td>
<td>v</td>
<td>ð</td>
<td>s</td>
<td>z</td>
<td>ñ</td>
<td>h</td>
<td></td>
</tr>
<tr>
<td>Lateral fricative</td>
<td>f</td>
<td>v</td>
<td>θ</td>
<td>s</td>
<td>z</td>
<td>ñ</td>
<td>h</td>
<td></td>
</tr>
<tr>
<td>Approximant</td>
<td>w</td>
<td></td>
<td>ñ</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lateral approximant</td>
<td>l</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Affricate</td>
<td>ts, dz</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4.2 Lecture Page

In the lecture page, learners are taught how to pronounce English vowels and consonants with MRI pictures. Figure 6 shows the top page of the web application. This webpage has four tabs. In “About IPA”, “Vowels” and “Consonants” pages, it explains how to read the IPA chart and how to make vowel and consonant sounds. In the “Word Practice” page, users can practice pronunciation and get feedback. I referred to [6] in making the “Vowels” and “Consonants” pages.

5 Source Code Description

5.1 HTML Source Code

The following is HTML source code. This code is for showing what is found in Figure 1.

```html
<div class="box_2">
  <div class="practice-place">
    <div class="example">
      <div class="practice-right">1. there</div>
      <button class="rec-btn" id="p1">REC</button>
    </div>
  </div>
  <div class="resultDiv" id="practice-result1">result here</div>
  <div class="btn-box">
    <div class="audio-btn">
      <audio src="there.mp3" controls>
      </audio>
    </div>
  </div>
</div>
```

The above code is for the left side of Figure 1. This display example word and REC button in example class, recognized word place in resultDiv class and example sound in audio.

```javascript
The clickFunc code above is for displaying the results of recognition and feedback.
```

4.2 Lecture Page

In the lecture page, learners are taught how to pronounce English vowels and consonants with MRI pictures. Figure 6 shows the top page of the web application. This webpage has four tabs. In “About IPA”, “Vowels” and “Consonants” pages, it explains how to read the IPA chart and how to make vowel and consonant sounds. In the “Word Practice” page, users can practice pronunciation and get feedback. I referred to [6] in making the “Vowels” and “Consonants” pages.

5 Source Code Description

5.1 HTML Source Code

The following is HTML source code. This code is for showing what is found in Figure 1.

```html
<div class="box_2">
  <div class="practice-place">
    <div class="example">
      <div class="practice-right">1. there</div>
      <button class="rec-btn" id="p1">REC</button>
    </div>
  </div>
  <div class="resultDiv" id="practice-result1">result here</div>
  <div class="btn-box">
    <div class="audio-btn">
      <audio src="there.mp3" controls>
      </audio>
    </div>
  </div>
</div>
```

The above code is for the left side of Figure 1. This display example word and REC button in example class, recognized word place in resultDiv class and example sound in audio.

```javascript
The clickFunc code above is for displaying the results of recognition and feedback.
```
<table>
<thead>
<tr>
<th>Sample</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
<th>K</th>
<th>L (native)</th>
<th>M</th>
</tr>
</thead>
<tbody>
<tr>
<td>there</td>
<td>there</td>
<td>yeah</td>
<td>yeah</td>
<td>there</td>
<td>sarah</td>
<td>date</td>
<td>there</td>
<td>there</td>
<td>dale</td>
<td>zelle</td>
<td>sarah</td>
<td>there</td>
<td>there</td>
</tr>
<tr>
<td>poor</td>
<td>cooler</td>
<td>frac</td>
<td>poor</td>
<td>poor</td>
<td>poor</td>
<td>poor</td>
<td>pearl</td>
<td>cool</td>
<td>cool</td>
<td>hua</td>
<td>torah</td>
<td>poor</td>
<td>poor</td>
</tr>
<tr>
<td>flocks</td>
<td>flat</td>
<td>flock</td>
<td>trunks</td>
<td>crocs</td>
<td>lyrics</td>
<td>clocks</td>
<td>trucks</td>
<td>rocks</td>
<td>crooks</td>
<td>clocks</td>
<td>flocks</td>
<td>phlox</td>
<td>phlox</td>
</tr>
<tr>
<td>dark</td>
<td>dark</td>
<td>DAC</td>
<td>dak</td>
<td>dark</td>
<td>doc</td>
<td>doc</td>
<td>dink</td>
<td>dark</td>
<td>dog</td>
<td>dogs</td>
<td>dark</td>
<td>dark</td>
<td>dogs</td>
</tr>
<tr>
<td>plan</td>
<td>brown</td>
<td>put on</td>
<td>prawn</td>
<td>rap</td>
<td>brown</td>
<td>plum</td>
<td>put on</td>
<td>brown</td>
<td>prom</td>
<td>plan</td>
<td>slime</td>
<td></td>
<td></td>
</tr>
<tr>
<td>run</td>
<td>noun</td>
<td>land</td>
<td>run</td>
<td>long</td>
<td>ren</td>
<td>long</td>
<td>la</td>
<td>wrong</td>
<td>wrong</td>
<td>run</td>
<td>wrong</td>
<td></td>
<td></td>
</tr>
<tr>
<td>village</td>
<td>ferrets</td>
<td>Dodge</td>
<td>petr czech</td>
<td>greenwich</td>
<td>spirits</td>
<td>we teach</td>
<td>pH</td>
<td>pH</td>
<td>spinach</td>
<td>philles</td>
<td>village</td>
<td>village</td>
<td>village</td>
</tr>
<tr>
<td>little</td>
<td>little</td>
<td>retina</td>
<td>little</td>
<td>little</td>
<td>critter</td>
<td>I don't</td>
<td>little</td>
<td>later</td>
<td>little</td>
<td>little</td>
<td>little</td>
<td>little</td>
<td>beetle</td>
</tr>
<tr>
<td>fun</td>
<td>song</td>
<td>fan</td>
<td>front</td>
<td>fun</td>
<td>fang</td>
<td>phone</td>
<td>farm</td>
<td>farm</td>
<td>farm</td>
<td>farm</td>
<td>phone</td>
<td>fun</td>
<td>bomb</td>
</tr>
<tr>
<td>third</td>
<td>todd</td>
<td>turn</td>
<td>sad</td>
<td>third</td>
<td>salt</td>
<td>surd</td>
<td>third</td>
<td>sad</td>
<td>sad</td>
<td>surag</td>
<td>shirt</td>
<td>third</td>
<td>todd</td>
</tr>
<tr>
<td>wolf</td>
<td>hello</td>
<td>all right</td>
<td>lowe's</td>
<td>oops</td>
<td>bro</td>
<td>both</td>
<td>wolf</td>
<td>wall</td>
<td>earth</td>
<td>wall</td>
<td>earth</td>
<td>wall</td>
<td>wall</td>
</tr>
<tr>
<td>threaten</td>
<td>threatened</td>
<td>Toyota</td>
<td>threatened</td>
<td>threatened</td>
<td>samaritan</td>
<td>threatened</td>
<td>creighton</td>
<td>slaton</td>
<td>sheronan</td>
<td>veteran</td>
<td>threatened</td>
<td>310</td>
<td></td>
</tr>
<tr>
<td>sheep</td>
<td>sheep</td>
<td>sweet</td>
<td>sheep</td>
<td>sheep</td>
<td>sheep</td>
<td>seat</td>
<td>see</td>
<td>sheep</td>
<td>sheep</td>
<td>sheep</td>
<td>sheep</td>
<td>sheep</td>
<td>sheep</td>
</tr>
<tr>
<td>louder</td>
<td>louder</td>
<td>ladder</td>
<td>louder</td>
<td>browser</td>
<td>rhoda</td>
<td>louder</td>
<td>yoda</td>
<td>louder</td>
<td>louder</td>
<td>louder</td>
<td>louder</td>
<td>louder</td>
<td>louder</td>
</tr>
<tr>
<td>exactly</td>
<td>exactly</td>
<td>exactly</td>
<td>exactly</td>
<td>exactly</td>
<td>echoes act 3</td>
<td>duck tree</td>
<td>exactly</td>
<td>exactly</td>
<td>you can suck 3</td>
<td>exactly</td>
<td>exactly</td>
<td>exactly</td>
<td>exactly</td>
</tr>
<tr>
<td>who</td>
<td>who</td>
<td>who</td>
<td>who</td>
<td>who</td>
<td>who</td>
<td>who</td>
<td>who</td>
<td>who</td>
<td>who</td>
<td>who</td>
<td>who</td>
<td>who</td>
<td>who</td>
</tr>
<tr>
<td>bother</td>
<td>bother</td>
<td>brother</td>
<td>bozeman</td>
<td>brother</td>
<td>weather</td>
<td>butter</td>
<td>bother</td>
<td>buzzer</td>
<td>father</td>
<td>squirrel</td>
<td>sound</td>
<td>bouncer</td>
<td>bother</td>
</tr>
<tr>
<td>fool</td>
<td>who</td>
<td>woo</td>
<td>food</td>
<td>full</td>
<td>food</td>
<td>coup</td>
<td>whole food</td>
<td>food</td>
<td>who</td>
<td>who</td>
<td>tune</td>
<td>cool</td>
<td>pool</td>
</tr>
<tr>
<td>cousins</td>
<td>cousins</td>
<td>cousins</td>
<td>cousins</td>
<td>cousins</td>
<td>consonance</td>
<td>cousin</td>
<td>cozy</td>
<td>cousins</td>
<td>cousin</td>
<td>cousins</td>
<td>cousin</td>
<td>cousins</td>
<td>cousin</td>
</tr>
<tr>
<td>fist</td>
<td>quit</td>
<td>quest</td>
<td>tryst</td>
<td>fist</td>
<td>fist</td>
<td>fixed</td>
<td>first</td>
<td>feast</td>
<td>fist</td>
<td>wrist</td>
<td>cost</td>
<td>fist</td>
<td>best</td>
</tr>
</tbody>
</table>

| # OF ERRORS | 11 | 18 | 13 | 5 | 16 | 19 | 12 | 16 | 14 | 12 | 13 | 2 | 11 |

Table 5: Results of MDN Web Speech API speech recognition for 11 Participants including one native speaker
Figure 6. The Vowels page of the web application

```javascript
var checkFunc = function(clickNum){
  var result=resultDiv[clickNum].innerHTML;
  var advice="";
  var listFlag = 0;
  result = result.toLowerCase();
  if(result.indexOf(sample[clickNum]) != -1)
    return 'correct!';
  The checkFunc code above makes and returns advice
  according to recognized word. If the recognized word
  is not collected, go to branch of switch.
  switch (clickNum) {
    case 0:
      for(let i = 0;i<misThere.length;i++)
        {
          if(result.indexOf(misThere[i][0]) != -1)
            {
              advice = misThere[i][1];
              listFlag = 1;
            }
        }
    break;
    case 1:
      ........

  6 Future Work

  We believe that this web application should be developed
  further, because the feedback system in the web applica-
  tion is based on too little result data. In the future, we
  must collect more recognized word data and improve the
  feedback system.

  References

```