

УДК 004.421

## ПРЕОБРАЗОВАНИЕ РЕЧИ В ТЕКСТ С ПОМОЩЬЮ JAVASCRIPT

Н.И. ЛАПКОВСКИЙ

(Представлено: Д.В. ПЯТКИН)

Статья посвящена работе с API Web Speech. Это очень мощный интерфейс браузера, который позволяет записывать человеческую речь и преобразовывать ее в текст. Также будет рассмотрен способ использовать его для обратного – считывания строк человеческим голосом.

**Конвертирование речи в текст.** Web Speech API фактически разделен на два совершенно независимых интерфейса: SpeechRecognition для понимания человеческого голоса и превращения его в текст и SpeechSynthesis для чтения строк вслух сгенерированным компьютером голосом.

API распознавания речи удивительно точен для бесплатной функции браузера. Он правильно распознал почти все мои разговоры и знал, какие слова объединяются в фразы, которые имеют смысл. Это также позволяет диктовать специальные символы, такие как точки останова, вопросительные знаки и новые строки. Пример представлен в листинге 1 [1].

Первое, что нужно сделать, это проверить, есть ли у пользователя доступ к API и показать соответствующее сообщение об ошибке. К сожалению, API преобразования речи в текст поддерживается только в Chrome и Firefox (с флажком), поэтому многие пользователи, вероятно, увидят это сообщение.

```
try {
  var SpeechRecognition = window.SpeechRecognition || window.webkitSpeechRecognition;
  var recognition = new SpeechRecognition();
}
catch(e) {
  console.error(e);
  $('#no-browser-support').show();
  $('#app').hide();
}
```

**Листинг 1. – Проверка наличия у пользователя доступа к API**

Переменная recognition даст нам доступ ко всем методам и свойствам API. Существуют различные доступные варианты, но мы только установим для параметра recognition.continuous значение true. Это позволит пользователям говорить с более длинными паузами между словами и фразами.

Прежде чем мы сможем использовать распознавание голоса, мы также должны настроить несколько обработчиков событий. Большинство из них просто слушают изменения в статусе распознавания. Пример представлен в листинге 2.

```
try {
  var SpeechRecognition = window.SpeechRecognition || window.webkitSpeechRecognition;
  var recognition = new SpeechRecognition();
}
catch(e) {
  console.error(e);
  $('#no-browser-support').show();
  $('#app').hide();
}
```

**Листинг 2 – Настройка обработчиков событий перед использованием распознавания голоса**

Однако есть специальное событие onresult, которое очень важно. Оно выполняется каждый раз, когда пользователь произносит слово или несколько слов в быстрой последовательности, давая нам доступ к текстовой транскрипции сказанного.

Когда мы записываем что-то с помощью обработчика onresult, мы сохраняем его в глобальной переменной и отображаем в текстовой области. Пример представлен в листинге 3.

```
recognition.onresult = function(event) {  
  
    // event это SpeechRecognitionEvent объект.  
    // Он хранит весь записанный диалог  
    // Нам нужен только текущий диалог/фраза  
    var current = event.resultIndex;  
  
    // Получаем текст сказанного.  
    var transcript = event.results[current][0].transcript;  
  
    // Добавляем текст к контенту нашего текстового документа.  
    noteContent += transcript;  
    noteTextarea.val(noteContent);  
}
```

### Листинг 3. – обработчик события recognition.onresult

Приведенный выше код немного упрощен. Существует очень странная ошибка на устройствах Android, из-за которой все повторяется дважды. Официального решения пока нет, но мне удалось решить проблему без каких-либо явных побочных эффектов. Имея в виду эту ошибку, код выглядит так [2]:

```
var mobileRepeatBug = (current == 1 && transcript ==  
event.results[0][0].transcript);  
  
if(!mobileRepeatBug) {  
    noteContent += transcript;  
    noteTextarea.val(noteContent);  
}
```

### Листинг 4. – Упрощённый код предыдущего листинга, решающий проблему с повторением выходного текста дважды на устройствах с ОС Android

Всё настроено, теперь можно начать использовать функцию распознавания голоса браузера. Чтобы начать, нужно просто вызвать метод start ():

```
$('#start-record-btn').on('click', function(e) {  
    recognition.start();  
});
```

### Листинг 5. – Вызов метода start () для начала распознавания голоса

При этом браузер попросит пользователей дать разрешение. Если разрешение будет дано, микрофон устройства будет активирован.

Браузер будет прослушивать некоторое время, и каждая распознанная фраза или слово будут транслироваться в текст. API прекратит прослушивание автоматически через пару секунд тишины или при остановке вручную.

```
$('#pause-record-btn').on('click', function(e) {  
    recognition.stop();  
});
```

### Листинг 6. – Обработка события нажатия кнопки остановки трансляции речи в текст

**Конвертирование текста в речь.** Синтез речи на самом деле очень прост. API доступен через объект speechSynthesis, и есть несколько методов для воспроизведения, приостановки и других вещей, связанных со звуком. Также есть несколько интересных функций, которые меняют высоту, скорость и даже голос читателя.

Все, что понадобится для демонстрации, это метод speak(). Он ожидает один аргумент — экземпляр класса SpeechSynthesisUtterance.

В листинге 7 представлен весь код, необходимый для считывания строки.

```
function readOutLoud(message) {  
    var speech = new SpeechSynthesisUtterance();  
  
    // Устанавливаем атрибуты текста и голоса.  
    speech.text = message;  
    speech.volume = 1;  
    speech.rate = 1;  
    speech.pitch = 1;  
  
    window.speechSynthesis.speak(speech);  
}
```

#### Листинг 7. – Чтение текста вслух голосом, синтезированным компьютером

**Заключение.** В статье был рассмотрен способ конвертирования человеческой речи в текст средствами API Web Speech. Так же был исследован способ трансляции текста в человеческую речь синтезированным компьютером голосом.

В эпоху, когда голосовые помощники стали популярнее, чем когда-либо, подобный API дает быстрый способ создания ботов, которые понимают и говорят на человеческом языке.

Добавление голосового управления к приложениям также может быть отличной формой улучшения доступности. Пользователи с нарушениями зрения могут пользоваться как пользовательскими интерфейсами «речь в текст», так и «текст в речь».

API синтеза речи и распознавания речи работают довольно хорошо и легко обрабатывают различные языки и акценты. К сожалению, на данный момент они имеют ограниченную поддержку браузеров, что ограничивает их использование в производстве.

#### ЛИТЕРАТУРА

1. Web Speech API – Web APIs | MDN [Электронный ресурс] – Режим доступа: [https://developer.mozilla.org/en-US/docs/Web/API/Web\\_Speech\\_API](https://developer.mozilla.org/en-US/docs/Web/API/Web_Speech_API) – Дата доступа: 24.09.2019.
2. HTML5 Speech Recognition API – codeburst [Электронный ресурс] – Режим доступа: <https://codeburst.io/html5-speech-recognition-api-670846a50e92> - Дата доступа: 24.09.2019.
3. Интерфейсы веб API | MDN [Электронный ресурс] – Режим доступа: <https://developer.mozilla.org/ru/docs/Web/API> – Дата доступа: 25.09.2019.
4. Интерфейсные события [Электронный ресурс] – Режим доступа: <https://learn.javascript.ru/event-details> – Дата доступа: 25.09.2019.