

Баруткин Илья Дмитриевич,
обучающийся 8 класса МБОУ «Школа №40 с углубленным изучением
отдельных предметов» городского округа город Уфа Республики
Башкортостан

Создание электронного ассистента как части прототипа «Умного дома»

Цель - создать «электронного ассистента»

Объект- создание электронного ассистента

Задачи:

1. Подобрать имеющиеся электронные устройства, необходимые для этого, изучить их.
2. Изучить материал, затрагивающий тему звука, его распознавание и синтез. Найти программное обеспечение, позволяющее это выполнить качественно и быстро.
3. Используя объектно-ориентированные возможности языка C++, создать программу для взаимодействия с ними. Разработать, используя информацию о форматах файлов, небольшой класс для определения некоторых характеристик WAVE PCM файлов.
4. Задать алгоритмы поведения, научить выявлять смысловые объекты из предложения.
5. Используя нейронные сети, создать пример собственной системы распознавания речи.
6. Создать простой web-сервис управления
7. Предотвратить возможные ошибки.

main.php

actions.php

test.php

simple_train.php

mainpoint.php

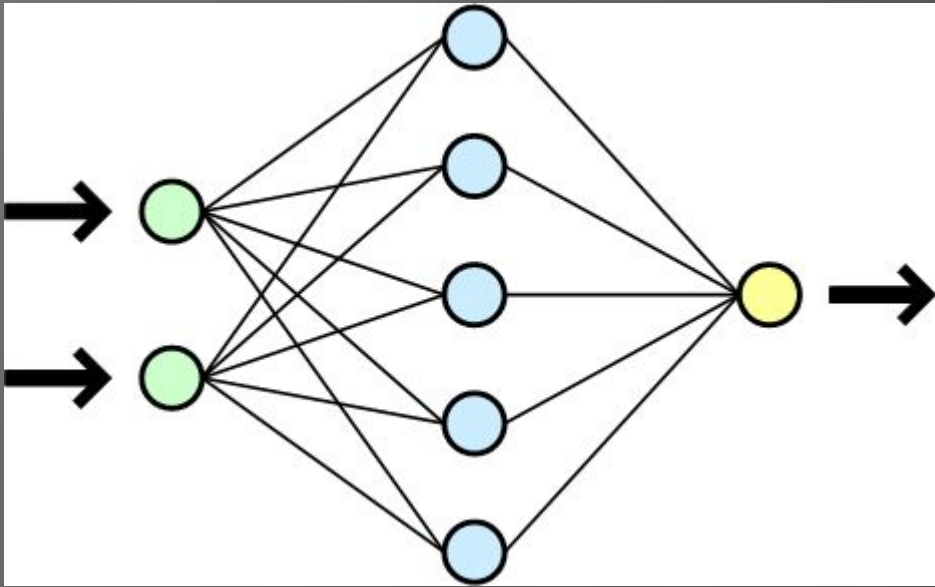
index.php

```

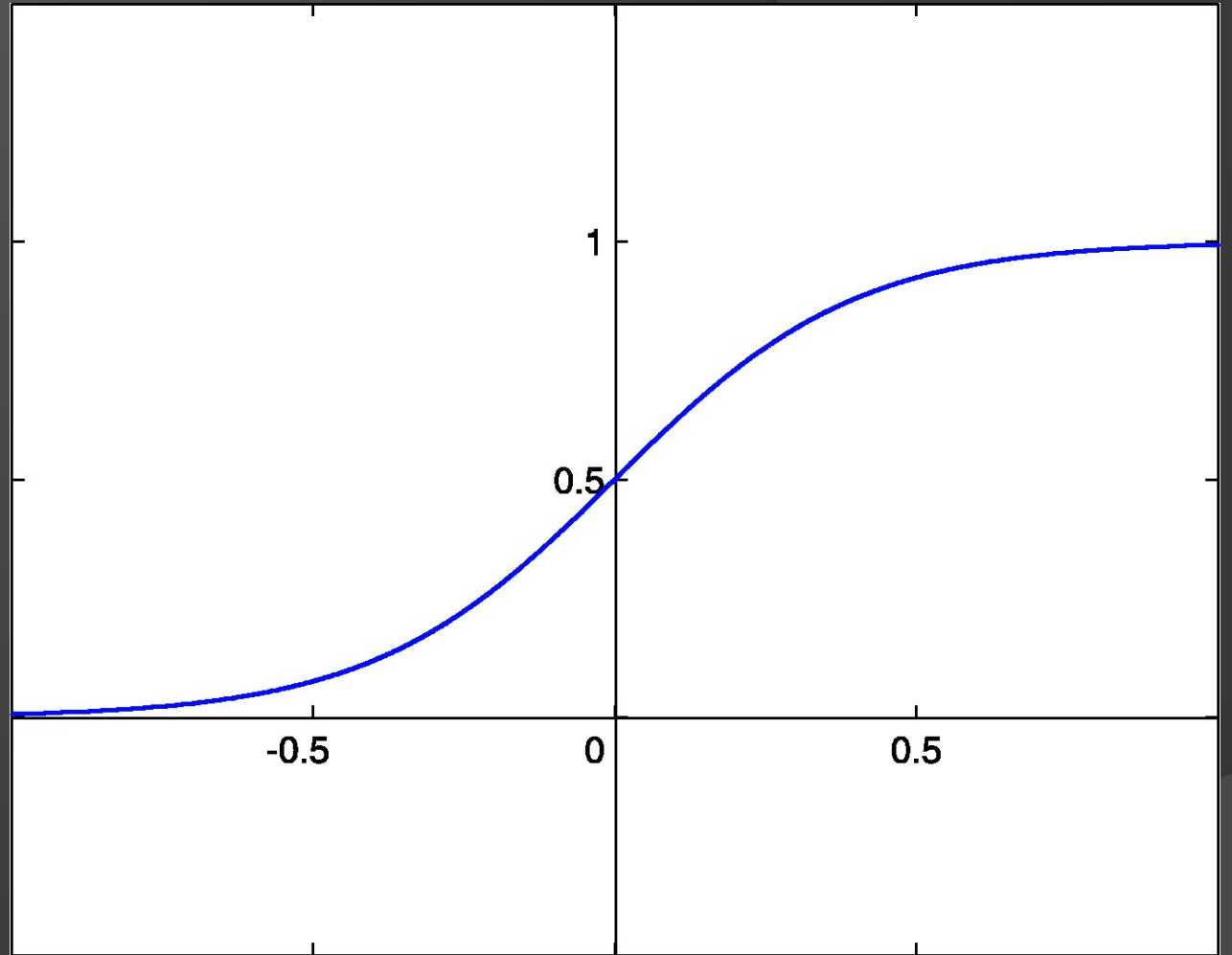
1  <?php
2  function get1_speech(){
3      exec("arecord -d1 -r 16000 test.wav");//-D plughw:1,0 test.wav"); --buffer-time=1000000
4      exec("./waver > ./res");
5      $res = file_get_contents("res");
6      return $res+0;
7  }
8  function saytotext(){
9      // exec("echo 4 > /sys/class/gpio/export");
10     // exec("echo 'out' > /sys/class/gpio/gpio4/direction");
11     echo "start\n";
12     system("arecord -r 16000 -d 3 current.wav");//-D plughw:1,0 current.wav"); -B --buffer-time=1000000 -f dat
13     echo "ok\n";
14     exec("flac -f current.wav");
15     //////////////////////////////////////
16     echo "Script started\n";
17     exec("wget -q -U \"Mozilla/5.0\" --post-file current.flac --header=\"Content-Type: audio/x-flac; rate=16000\" -O - \"http://www.google.com/speech-api/v2/re
18     $result = file_get_contents("recognize");
19     $result = str_replace("{\"result\":[]}", "", $result);//remove empty result(if it is)
20     $json_array = json_decode($result, true);
21     $voice_cmd = $json_array["result"][0]["alternative"][0]["transcript"];
22     // exec("echo 1 >/sys/class/gpio/gpio4/value");
23     //exec("echo 4 >/sys/class/gpio/unexport");
24     return $voice_cmd;
25 }
26
27 function word_to_number($word){
28     $word = explode(" ", $word);
29     $numbs = array("ноль", "один", "два", "три", "четыре", "пять", "шесть", "семь", "восемь", "девять");
30     $stens = array("десять", "одиннадцать", "двенадцать", "тринадцать", "четырнадцать", "пятнадцать", "шестнадцать", "семнадцать", "восемнадцать", "девятнадцать", "двадц
31     $hands = array("сто", "двести", "триста", "четыреста", "пятьсот", "шестьсот", "семьсот", "восемьсот", "девятьсот");

```

```
main.php actions.php test.php simple_train.php mainpoint.php index.php
26     <input type="hidden" name="cmd" value="tospeak" />
27     <input type="submit" value="Сказать" />
28 </form>
29 </div>
30 </div>
31
32 <div class="col-sm-12">
33     <div class="main-text" >
34         <h4>Температура процессора: </h4>
35         <iframe src="/client.php?cmd=cputemp&desc=null" > <a href="/client.php?cmd=cputemp&desc=null"> нажмите сюда </iframe>
36     </div>
37 </div>
38
39 <div class="col-sm-12">
40     <div class="main-text" >
41         <h4>Фото: </h4>
42         
43     </div>
44 </div>
45
46 <div class="col-sm-12">
47     <div class="main-text" >
48         <h4>Уровень громкости: </h4>
49         <iframe style="border:0;width:100%;max-height:300px;height:300px;" src="/client.php?cmd=volume&desc=null" > <a href="/client.php?cmd=volume&desc=null" > нажмите сюда </iframe>
50     </div>
51 </div>
52 </div>
53 <div class="col-sm-12">
54     <div class="main-text" >
55 <form action="/client.php" method="GET">
56     <input type="hidden" name="cmd" value="shutdown" />
```



$$\text{OUT} = f\left(\sum_{i=1}^n w_i x_i + w_0 x_0\right)$$





File Edit View Selection Find Packages Help

File Edit View Selection Find Packages Help

main.php

actions.php

test.php

```

1 <?php
2 $num_input = 72;
3 $num_output = 1;
4 $num_layers = 3;
5 $num_neurons_hidden = 10;
6 $desired_error = 0.001;
7 $max_epochs = 500000;
8 $epochs_between_reports = 1000;
9 //////////////////////////////////////////////////
10 echo "Need you for training? : ";
11 $line = trim(fgets(STDIN));
12 if(false !== strpos($line,"no") ) goto start;
13 while(1){
14
15 exec("sh script.sh");
16 exec("cat obch.txt > file2");
17 $f2 = explode("\n",file_get_contents("file2"));
18
19 foreach($f2 as $el){
20 //echo $el."\n";
21 if(isset(explode(" ",$el)[1])) $arr1[] = explode(" ",$el)[1];
22 else echo "";
23 }
24 $c = 0;
25 file_put_contents("./net.dat","\n",FILE_APPEND);
26 foreach($arr1 as $el){$c++;
27 if($c == 73) break;
28 file_put_contents("./net.dat",((int)($el/40) > 20 ) ? "1 " : "0 ",FILE_APPEND);
29 }
30 echo "HELLO OR NO :";
31 $line = trim(fgets(STDIN));

```

main.php

actions.php

test.php

```

21 if(isset(explode(" ",$el)[1])) $arr1[] = explode(" ",$el)[1];
22 else echo "";
23 }
24 $c = 0;
25 file_put_contents("./net.dat","\n",FILE_APPEND);
26 foreach($arr1 as $el){$c++;
27 if($c == 73) break;
28 file_put_contents("./net.dat",((int)($el/40) > 20 ) ? "1 " :
29 }
30 echo "HELLO OR NO :";
31 $line = trim(fgets(STDIN));
32 if(false !== strpos($line, "y")) file_put_contents("./net.dat",
33 elseif(false !== strpos($line, "n")) file_put_contents("./net.
34 elseif(false !== strpos($line, "go")) break;
35 }
36 //////////////////////////////////////////////////
37 start:
38 $ann = fann_create_standard($num_layers, $num_input, $num_neurons_h
39
40 if ($ann) {
41 fann_set_activation_function_hidden($ann, FANN_SIGMOID_SYMMETRI
42 fann_set_activation_function_output($ann, FANN_SIGMOID_SYMMETRI
43
44 $filename = dirname(__FILE__) . "/net.dat";
45 if (fann_train_on_file($ann, $filename, $max_epochs, $epochs_be
46 fann_save($ann, dirname(__FILE__) . "/net.net");
47
48 fann_destroy($ann);
49 }
50 ?>
51

```

Исследования, проведенные во время работы:

- 1) Были проведены исследования звука, распознавания и синтеза речи, принципа работы Raspberry Pi, основ электроники и нейронных сетей.
- 2) Изучение объектно-ориентированного и процедурного программирования в C, C++, PHP, что начиналось ранее, но позволило понять эту тему глубже.
- 3) Работа с сетью.
- 4) Программирование звука.
- 5) Изучение Linux и Unix-like операционных систем, которым я занялся тоже долго до этого.

Выводы:

- ◎ Создан «ассистент», хорошее развлечение и простой способ получения нужной информации, не отвлекаясь от других важных дел.
- ◎ Создан прототип собственной системы распознавания речи.