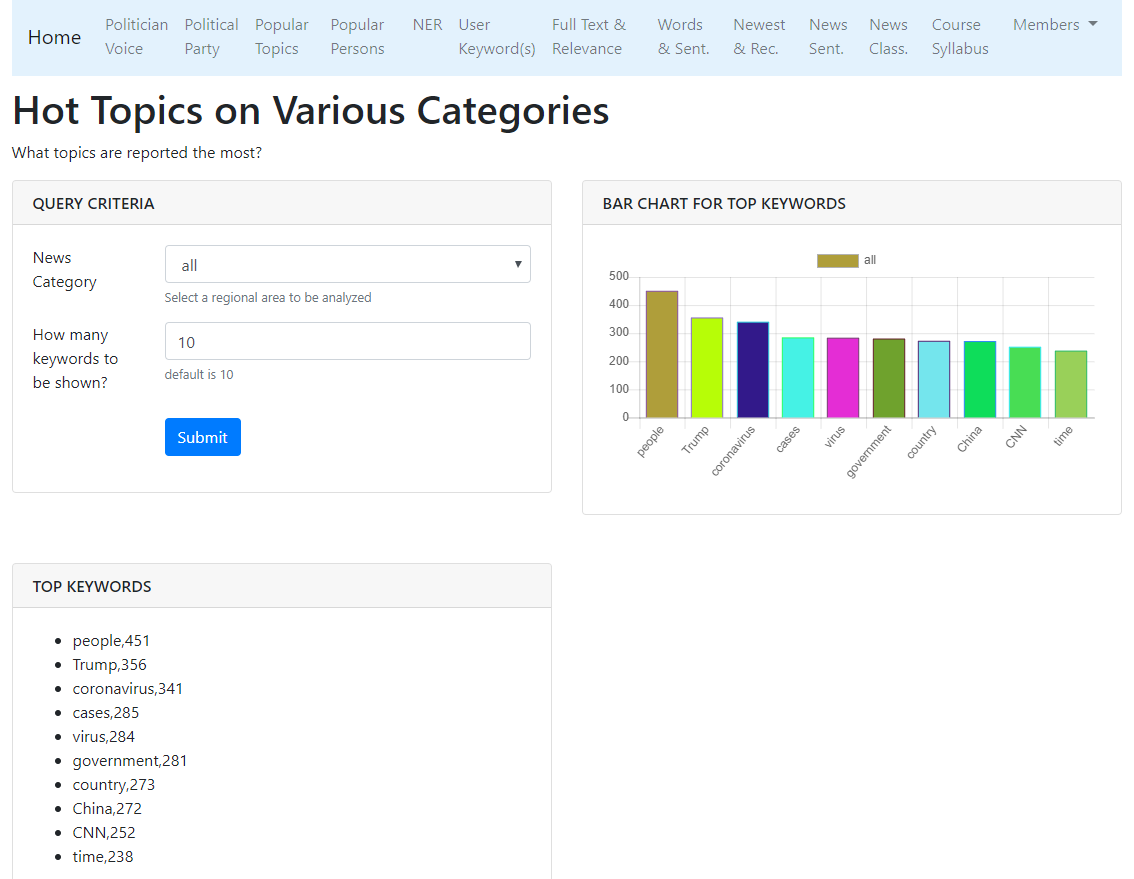
Top Keywords Display and Charting

We use Ajax in this tutorial. However, you can use any other techniques to implement the website.

There are many other chart libraries you can use, such as [D3.js](https://d3js.org/) and [Highcharts](http://www.highcharts.com/). We use Char.js and draw a bar chart for top keywords.





Development tools

In this class, we use “VS Code.” However, you can use PyCharm, Sublime, Atom, Notepad++, vi, vim, or any other favorite editor.

Create a new Django project

Step 1: Create a folder named “website\_news\_analysis\_v1”

mkdir website\_news\_analysis\_v1

Step 2: Go into the folder “website\_news\_analysis\_v1”

cd website\_news\_analysis\_v1

Step 3: Create a project configures folder named website\_configs

django-admin startproject website\_configs .

A dot here

Runserver

python manage.py runserver 8000

Create an APP

Step 1: Please make sure if you are in the folder of “website\_news\_analysis\_v1.” If not, please go into the folder “website\_news\_analysis\_v1”

Step 2: Create an APP named “app\_top\_keyword”

django-admin startapp app\_top\_keyword

settings.py

settings.py

|  |
| --- |
| import os  ALLOWED\_HOSTS = [] # or '\*', 'localhost', '127.0.0.1',"your public ip"  INSTALLED\_APPS = [  …..  'app\_top\_keyword',  ]  TEMPLATES = [  {  'BACKEND': 'django.template.backends.django.DjangoTemplates',  'DIRS': [os.path.join(BASE\_DIR, 'templates')],  'APP\_DIRS': True,  'OPTIONS': …..  ……  ……  },  ] |

Notice 1: installed app:

Please make sure the app name should be in the installed app list.

INSTALLED\_APPS = [

…..

'app\_top\_keyword',

]

Django will find your templates (e.g., home.html) in the folder of “app\_top\_keyword/templates/”

Notice 2: templates path:

Django could not find the base.html. Why did it happen? Because we did not specify the search path for templates in setting.py, Django cannot find our navbar.html. How to specify the search path in the setting.py?

TEMPLATES = [

{

'BACKEND': 'django.template.backends.django.DjangoTemplates',

'DIRS': [os.path.join(BASE\_DIR, 'templates')],

'APP\_DIRS': True,

'OPTIONS': …..

……

……

},

]

Inside the TEMPLATES =[ … ], please make sure if the following line exists:

'DIRS': [os.path.join(BASE\_DIR, 'templates')],

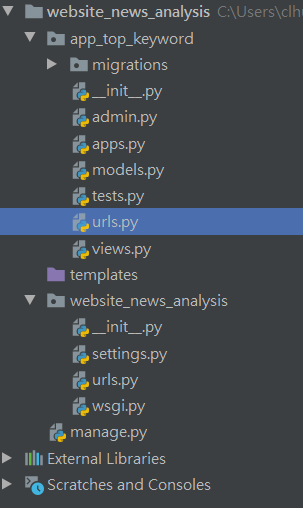
Django will also find your HTML files in the following folder:

websit\_news\_analysis\_v1/templates

Create “app\_top\_keywrod/urls.py”

In folder app\_top\_keyword, create a python file named “urls.py”

app\_top\_keywrod/urls.py



This is the urls.py you created.

app\_top\_keywrod/urls.py

|  |
| --- |
| from django.urls import path  from app\_top\_keyword import views  # Declare a namespace for this APP  app\_name = 'app\_top\_keyword'  urlpatterns = [  # For home  path('', views.home, name='home'),  # For Ajax  # path('api\_get\_cate\_topword/', views.api\_get\_cate\_topword),  ] |

'api\_get\_cate\_topword/'

Notice there is a slash symbol at the end of path string.

Declare a namespace for this APP. The name of namespace is 'app\_top\_keyword.' We will use the namespace in the future’s integrated website.

Modify “website\_configs/urls.py”

website\_configs/urls.py

|  |
| --- |
| from django.urls import path  from django.urls import include  urlpatterns = [  # top keywords  path('topword/', include('app\_top\_keyword.urls')),  ] |

views.py

|  |
| --- |
| from django.shortcuts import render  from django.http import JsonResponse  import pandas as pd  # render渲染網頁  def home(request):  return render(request, 'app\_top\_keyword/home.html')  print("app\_top\_keywords--類別熱門關鍵字載入成功!") |

完整的views.py

|  |
| --- |
| from django.shortcuts import render  from django.http import JsonResponse  import pandas as pd  # render渲染網頁  def home(request):  return render(request, 'app\_top\_keyword/home.html')  # read df  df\_topkey = pd.read\_csv('app\_top\_keyword/dataset/cna\_news\_topkey\_with\_category\_via\_token\_pos.csv', sep=',')  # prepare data  data={}  for idx, row in df\_topkey.iterrows():  data[row['category']] = eval(row['top\_keys'])  # We don't use it anymore, so delete it to save memory.  del df\_topkey  # POST: csrf\_exempt should be used  # 指定這一支程式忽略csrf驗證  from django.views.decorators.csrf import csrf\_exempt  @csrf\_exempt  def api\_get\_cate\_topword(request):  cate = request.POST.get('news\_category')  #cate = request.GET['news\_category'] # this command also works.  topk = request.POST.get('topk')  topk = int(topk)  print(cate, topk)  chart\_data, wf\_pairs = get\_category\_topword(cate, topk)  response = {'chart\_data': chart\_data,  'wf\_pairs': wf\_pairs,  }  print(response)  return JsonResponse(response)  def get\_category\_topword(cate, topk=10):  wf\_pairs = data[cate][0:topk]  words = [w for w, f in wf\_pairs]  freqs = [f for w, f in wf\_pairs]  chart\_data = {  "category": cate,  "labels": words,  "values": freqs}  return chart\_data, wf\_pairs  print("app\_top\_keywords--類別熱門關鍵字載入成功!") |

HTML (Template)

In the app folder, create a HTML file named “home.html”

app\_top\_keyword/templates/app\_top\_keyword/home.html

home.html

|  |
| --- |
| <!DOCTYPE html>  <html lang="zh-TW">  <head>  <title>輿情分析平台</title>  <meta charset="utf-8" />  <meta name="viewport" content="width=device-width, initial-scale=1" />  <!-- Bootstrap 5 CSS -->  <link  href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"  rel="stylesheet"  />  </head>  <body>  <div class="container">  <div class="row">  <!-- Navigation Bar -->  <div class="col-lg-12 mb-2">  <nav  class="navbar navbar-expand-lg navbar-light"  style="background-color: #e3f2fd"  >  <div class="container-fluid">  <a class="navbar-brand" href="#">輿情大數據</a>  <button  class="navbar-toggler"  type="button"  data-bs-toggle="collapse"  data-bs-target="#navbarSupportedContent"  aria-controls="navbarSupportedContent"  aria-expanded="false"  aria-label="Toggle navigation"  >  <span class="navbar-toggler-icon"></span>  </button>  <div class="collapse navbar-collapse" id="navbarSupportedContent">  <ul class="navbar-nav me-auto mb-2 mb-lg-0">  <li class="nav-item">  <a class="nav-link" href="#">政治人物聲量排行榜</a>  </li>  <li class="nav-item">  <a class="nav-link" href="#">政黨聲量排行榜</a>  </li>  <li class="nav-item">  <a class="nav-link" href="{% url 'app\_top\_keyword:home' %}"  >熱門關鍵詞分析</a  >  </li>  <li class="nav-item">  <a class="nav-link" href="#">昨日熱門關鍵字</a>  </li>  <li class="nav-item">  <a class="nav-link" href="#">熱門人物排行分析</a>  </li>  <li class="nav-item">  <a class="nav-link" href="#">NER熱門分析</a>  </li>  <li class="nav-item">  <a class="nav-link" href="#">你的關鍵詞熱門度分析</a>  </li>  <li class="nav-item">  <a class="nav-link" href="#">全文檢索與關聯新聞分析</a>  </li>  <li class="nav-item">  <a class="nav-link" href="#">你的關鍵詞情緒分析</a>  </li>  <li class="nav-item">  <a class="nav-link" href="#">今日新聞瀏覽與新聞推薦</a>  </li>  <li class="nav-item">  <a class="nav-link" href="#">新聞或文章情緒分類</a>  </li>  <li class="nav-item">  <a class="nav-link" href="#">新聞分類新聞分類</a>  </li>  <li class="nav-item">  <a class="nav-link" href="#">課程介紹網頁</a>  </li>  <li class="nav-item dropdown">  <a  class="nav-link dropdown-toggle"  href="#"  id="navbarDropdown"  role="button"  data-bs-toggle="dropdown"  aria-expanded="false"  >  會員獨享功能  </a>  <ul class="dropdown-menu" aria-labelledby="navbarDropdown">  <li>  <a class="dropdown-item" href="#"  >長時間(超過3個月)監控分析(收費標準)</a  >  </li>  <li>  <a class="dropdown-item" href="#"  >自訂分析功能(收費標準)</a  >  </li>  <li>  <a class="dropdown-item" href="#"  >更多資料庫分析(收費標準)</a  >  </li>  <li><hr class="dropdown-divider" /></li>  <li>  <a class="dropdown-item" href="#"  >為你訂製輿情分析任務(報價)</a  >  </li>  </ul>  </li>  </ul>  </div>  </div>  </nav>  </div>  <!-- End of Navigation Bar -->  <!-- Main Content -->  <div class="col-lg-12">  <h1>各類新聞最熱門的關鍵詞</h1>  <p>熱門度分析:可以了解新聞關注那些重要的東東</p>  </div>  <!-- 新聞類別選單------------------------------->  <div class="col-lg-6 mb-2">  <div class="card">  <div class="card-header">  <h3 class="h6 text-uppercase mb-0">  熱門關鍵字瀏覽與繪圖(資料週期:資料截止時間的前4周)  </h3>  </div>  <div class="card-body">  <!--新聞類別選單 form group-->  <div class="form-group row">  <label class="col-sm-3 form-control-label">新聞類別</label>  <div class="col-md-9">  <select  id="cate-selected"  name="news\_category"  class="form-control"  >  <!--<option>請選擇</option>-->  <option>全部</option>  <option>政治</option>  <option>科技</option>  <option>娛樂</option>  <option>運動</option>  <option>生活</option>  <option>產經</option>  <option>證卷</option>  <option>國際</option>  <option>社會</option>  <option>文化</option>  <option>兩岸</option>  </select>  <small class="form-text text-muted">請選擇新聞類別 </small>  </div>  </div>  <!--form group-->  <!--熱門詞多少個?form group-->  <div class="form-group row">  <label class="col-md-3 form-control-label">多少個熱門詞?</label>  <div class="col-md-9">  <input  id="topk-selected"  name="topk"  value="10"  class="form-control form-control-success"  />  <small class="form-text text-muted">內定值為10 </small>  </div>  </div>  <!--form group-->  <!--submit按鈕form group-->  <div class="form-group row">  <div class="col-md-9 ml-auto">  <button type="button" id="btn-ok" class="btn btn-primary">  查詢  </button>  </div>  </div>  <!--form group-->  </div>  <!--card body-->  </div>  <!--column-->  </div>  <!-- 區塊結束 -->  <!-- 繪圖區塊--------------------------------------------------------->  <div class="col-lg-6 mb-5">  <div class="card">  <div class="card-header">  <h3 class="h6 text-uppercase mb-0">熱門關鍵字繪圖</h3>  </div>  <div class="card-body">  <canvas id="mychart"></canvas>  </div>  </div>  </div>  <!-- 區塊結束 -->  <!-- 熱門關鍵字區塊----------------------------------------------------->  <div class="col-lg-6 mb-5">  <div class="card">  <div class="card-header">  <h3 class="h6 text-uppercase mb-0">熱門關鍵字</h3>  </div>  <div class="card-body">  <ul id="topkeys"></ul>  </div>  </div>  </div>  <!-- 區塊結束 -->  </div>  </div> <!-- container -->  <!-- Bootstrap 5 JS Bundle with Popper -->  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>  <!-- Additional Scripts -->  <!-- chartjs圖js-->  <script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.7.3/Chart.min.js"></script>  <!-- 程式碼區 -->  <script>  // Show default top words bar chart when the page is loaded.  // Invoke callAjax() after the function is defined, unless callAjax() is defined with "hoisting"  callAjax();  //\*新聞類別選單select被選中值有改變時，執行以下事件  //$('#cate-selected').on('change', callAjax() );  $("#cate-selected").on("change", function () {  callAjax();  }); //event function  //\*\*按鈕事件  $("#btn-ok").on("click", function () {  callAjax();  }); //event function  // \*\* draw chart using Ajax 畫圖  // There are two ways to define a function:  // one is expression, and the other is definition with "hoisting"  // function with hoisting: function callAjax() {}  // normal function expression: let callAjax = function() {}  // Define callAjax function with hoisting  // callAjax()這樣定義可以在被定義前就被使用 跟我們在Java裡面的函數用法一樣!  //let callAjax = function() {  function callAjax() {  let cate = $("#cate-selected").val();  //console.log(cate);  let topk = $("#topk-selected").val();  //console.log(topk);  $.ajax({  type: "POST",  //url: "/topword/api\_get\_cate\_topword/",  url: "http://127.0.0.1:8000/topword/api\_get\_cate\_topword/",  //url: "http://163.18.23.21:8000/topword/api\_get\_cate\_topword/",  //url: "api\_get\_cate\_topword/", //Not recommended!  data: {  news\_category: cate,  topk: topk,  },  success: function (received) {  //console.log(received);  let chart\_data = received.chart\_data;  let wf\_pairs = received.wf\_pairs;  console.log(wf\_pairs);  showTopKeys(wf\_pairs);  showChart(chart\_data);  }, //success function  }); //ajax  } //callAjax  //\* 顯示關鍵詞資料函數  function showTopKeys(items) {  //先清除前一次的資料  $("#topkeys").empty();  //將內容加上li標籤附加起來，顯示在顯示區"topkeys"  for (let i = 0; i < items.length; i++) {  let item\_li = "<li>" + items[i] + "</li>";  $("#topkeys").append(item\_li);  }  } //function  //\*\*繪圖函數showChart()  function showChart(chart\_data) {  // 畫圖需要的數據資料  let values = chart\_data.values;  let labels = chart\_data.labels;  let category = chart\_data.category;  //第1個變數: 餵給chart的資料  let data = {  labels: labels,  datasets: [  {  label: category,  data: values,  backgroundColor: randomColors(values.length),  borderColor: randomColors(values.length),  borderWidth: 1,  },  ],  };  //第2個變數: chart的選項 指定y坐標軸從零開始顯示  let options = {  scales: {  yAxes: [  {  ticks: {  beginAtZero: true,  },  },  ],  },  };  //取得在前面html區域欲顯示的圖代號  let canvas\_mychrat = document.getElementById("mychart");  //\*\*先清除前一個圖 再繪新圖  // 可以印出barchart物件是否存在  // console.log(window.barchart);  //先清除前一個圖 再繪新圖 if 有以下兩種寫法皆可  // if (window.barchart) //若存在則為true  // if (typeof (barchart) != "undefined"){  if (window.barchart) {  barchart.destroy();  }  //\*\*繪圖(產生一個圖物件變數名稱為barchart)  // 必須全域變數--注意:前面不要有let, var, const等修飾詞  // 理由: 我們要讓它存在於網頁全域變數，  // 這樣我們才方便判斷是否有前一次的圖，如果存在有，要刪除之，否則，很多張圖會疊在一起  barchart = new Chart(canvas\_mychrat, {  type: "bar",  data: data,  options: options,  });  //\*\* 產生隨機顏色  function randomColors(num\_colors) {  let colors = [];  for (i = 0; i < num\_colors; i++) {  let r = Math.floor(Math.random() \* 255);  let g = Math.floor(Math.random() \* 255);  let b = Math.floor(Math.random() \* 255);  let rgb = `rgba(${r},${g},${b},0.5)`; // (red, green, blue, alfa) alfa透明度  colors.push(rgb);  }  return colors;  }  } //show chart function  // document就是這個網頁HTML所有的元素  // window就是這個網頁的全域變數global variables:有一大堆，我們自己定義的有callAjax, showChart, barchart等  // 把document, window印出來看看就能理解它們是甚麼  //console.log(document);  //console.log(window);  </script>  </body>  </html> |

Url link

The hypertext reference for 熱門關鍵詞分析 is as follows:

<li class="nav-item">

<a class="nav-link" href="{% url 'app\_top\_keyword:home' %}">熱門關鍵詞分析</a>

</li>

app\_top\_keyword Namespace名稱空間

home 定義在urls.py path中的name

Run server

python manage.py runserver 8000

<http://localhost:8000>/topword

What does your website look like?

Write your JS code step by step

Exercise Step by Step

|  |
| --- |
| <!-- chartjs圖js-->  <script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.7.3/Chart.min.js"></script>  <!-- 程式碼區 -->  <script>  // Exercise#0: display user selection  let cate = $('#cate-selected').val();  console.log(cate);  let topk = $('#topk-selected').val();  console.log(topk);    // Exercise#1: Event handling  //\*\*按鈕事件  $('#btn-ok').on('click', function () {  console.log("按下按鈕");  alert("按下按鈕");  //call\_ajax();  }); //event function  //\*新聞類別選單select被選中值有改變時，執行以下事件  $('#cate-selected').on('change', function () {    let cate = $('#cate-selected').val();  console.log(cate);  let topk = $('#topk-selected').val();  console.log(topk);  //call\_ajax();  }); //event function  // Exercise#2: Define a function  // Please paste showTopKeys function here!  //\* 顯示關鍵詞資料函數  function showTopKeys(items) {  //先清除前一次的資料  $('#topkeys').empty();  //將內容加上li標籤附加起來，顯示在顯示區"topkeys"  for (let i = 0; i < items.length; i++) {  let item\_li = "<li>" + items[i] + "</li>";  $('#topkeys').append(item\_li);  }  } //function  // Exercise#3: Call function showTopKeys()  ~~let wf\_pairs = [["高科大", 25], ["資管系", 20]]; //define a JavaScript array~~  ~~showTopKeys(wf\_pairs);~~  // Call function when btn\_ok is clicked  $('#btn-ok').on('click', function () {    let wf\_pairs = [ ["大數據",45] ,["高科大", 18], ["資管系", 12],]; //define a JavaScript array  showTopKeys(wf\_pairs);  }); //event function  // Exercise#4: Define “call\_ajax” function to perform Ajax  // Call ajax function when page is loaded and button is clicked.  // See what the data received from backend API looks like.  // Display word frequency pairs.  function call\_ajax() {  let cate = $('#cate-selected').val();  let topk = $('#topk-selected').val();  $.ajax({  type: "POST",  url: "/topword/api\_get\_cate\_topword/",  //url: "http://127.0.0.1:8000/topword/api\_get\_cate\_topword/",  //url: "http://163.18.23.21:8000/topword/api\_get\_cate\_topword/",  //url: "api\_get\_cate\_topword/", //Not recommended!  data: { "news\_category": cate, "topk": topk },  success: function (received) {  console.log(received);  let chart\_data = received.chart\_data;  let wf\_pairs = received.wf\_pairs;  console.log(wf\_pairs);  showTopKeys(wf\_pairs);  //showChart(chart\_data);  } //success function  }); //ajax  } //call\_ajax  // Exercise#5: Define barchart function and call it in the $.ajax block.  //\*\*繪圖函數showChart()  // Please paste below showChart function here!  function showChart(chart\_data) {  程式碼在後面…  }  </script> |

Display keyword and frequency

|  |
| --- |
| //\* 顯示關鍵詞資料函數  function showTopKeys(items) {  //先清除前一次的資料  $('#topkeys').empty();  //將內容加上li標籤附加起來，顯示在顯示區"topkeys"  for (let i = 0; i < items.length; i++) {  let item\_li = "<li>" + items[i] + "</li>";  $('#topkeys').append(item\_li);  }  } //function |

Show chart function

|  |
| --- |
| //\*\*繪圖函數showChart()  function showChart(chart\_data) {  // 畫圖需要的數據資料  let values = chart\_data.values;  let labels = chart\_data.labels;  let category = chart\_data.category;  //第1個變數: 餵給chart的資料data  let data = {  labels: labels,  datasets: [{  label: category,  data: values,  backgroundColor: randomColors(values.length),  borderColor: randomColors(values.length),  borderWidth: 1,  }],  };  //第2個變數: chart的選項 指定y坐標軸從零開始顯示  let options = {  scales: {  yAxes: [{  ticks: {  beginAtZero: true  }  }]  },  };  //取得在前面html區域欲顯示的圖代號  let canvas\_mychrat = document.getElementById("mychart");  //\*\*先清除前一個圖 再繪新圖  // 可以印出barchart物件是否存在  // console.log(window.barchart);  //先清除前一個圖 再繪新圖 if 有以下兩種寫法皆可  // if (window.barchart) //若存在則為true  // if (typeof (barchart) != "undefined"){  if (window.barchart) {  barchart.destroy();  }  //\*\*繪圖(產生一個圖物件變數名稱為barchart)  // 必須全域變數--注意:前面不要有let, var, const等修飾詞  // 理由: 我們要讓它存在於網頁全域變數，  // 這樣我們才方便判斷是否有前一次的圖，如果存在有，要刪除之，否則，很多張圖會疊在一起  barchart = new Chart(canvas\_mychrat, {  type: 'bar',  data: data,  options: options,  });  //\*\* 產生隨機顏色  function randomColors(num\_colors) {  let colors = [];  for (i = 0; i < num\_colors; i++) {  let r = Math.floor(Math.random() \* 255);  let g = Math.floor(Math.random() \* 255);  let b = Math.floor(Math.random() \* 255);  let rgb = `rgba(${r},${g},${b},0.5)` // (red, green, blue, alfa) alfa透明度  colors.push(rgb);  }  return colors;  }  } //show chart function |

* RGBA Colors

RGBA color values are an extension of RGB color values with an alpha channel - which specifies the opacity of the object.

An RGBA color is specified with the rgba() function, which has the following syntax:

rgba(red, green, blue, alpha)

The alpha parameter is a number between 0.0 (fully transparent) and 1.0 (fully opaque不透明).

Make your api\_get\_cate\_topword API work!

Put your dataset into the project

You should count your keyword of CNA news in advance and save the top keywords of all categories in the file “cna\_news\_topkey\_with\_category\_via\_token\_pos.csv”

You can use it in your Django website. Copy your top keywords csv file into folder “dataset” as follows:

app\_top\_keyword/dataset/cna\_news\_topkey\_with\_category\_via\_token\_pos.csv

Enable (unmark) your api path

app\_top\_keywrod/urls.py

|  |
| --- |
| from django.urls import path  from app\_top\_keyword import views  # Declare a namespace for this APP  app\_name = 'app\_top\_keyword'  urlpatterns = [  # For home  path('', views.home, name='home'),  # For Ajax  path('api\_get\_cate\_topword/', views.api\_get\_cate\_topword),  ] |

Complete your views.py

|  |
| --- |
| from django.shortcuts import render  from django.http import JsonResponse  import pandas as pd  # render渲染網頁  def home(request):  return render(request, 'app\_top\_keyword/home.html')  # read df  df\_topkey = pd.read\_csv('app\_top\_keyword/dataset/cna\_news\_topkey\_with\_category\_via\_token\_pos.csv', sep=',')  # prepare data  data={}  for idx, row in df\_topkey.iterrows():  data[row['category']] = eval(row['top\_keys'])  # We don't use it anymore, so delete it to save memory.  del df\_topkey  # POST: csrf\_exempt should be used  # 指定這一支程式忽略csrf驗證  from django.views.decorators.csrf import csrf\_exempt  @csrf\_exempt  def api\_get\_cate\_topword(request):  cate = request.POST.get('news\_category')  #cate = request.POST['news\_category'] # this command also works.  topk = request.POST.get('topk')  topk = int(topk)  print(cate, topk)  chart\_data, wf\_pairs = get\_category\_topword(cate, topk)  response = {  'chart\_data': chart\_data,  'wf\_pairs': wf\_pairs,  }  #print(response)  return JsonResponse(response)  def get\_category\_topword(cate, topk=10):  wf\_pairs = data[cate][0:topk]  words = [w for w, f in wf\_pairs]  freqs = [f for w, f in wf\_pairs]  chart\_data = {  "category": cate,  "labels": words,  "values": freqs}  return chart\_data, wf\_pairs  print("app\_top\_keywords--類別熱門關鍵字載入成功!") |

Better Structure

base.html

Please create the navbar.html in the folder of “websit\_news\_analysis\_v1/templates. That is the full path is as follows:

websit\_news\_analysis\_v1/templates/base.html

You have defined the search path of templates file, so Django will automatically find your html files in the websit\_news\_analysis\_v1/templates folder!

home.html

|  |
| --- |
| {% extends 'base.html' %}  {% block title %}輿情分析平台 - 熱門關鍵詞分析{% endblock %}  {% block content %}  <div class="col-lg-12">  <h1>各類新聞最熱門的關鍵詞</h1>  <p>熱門度分析:可以了解新聞關注那些重要的東東</p>  </div>  <!-- 新聞類別選單------------------------------->  <div class="col-lg-6 mb-2">  <div class="card">  <div class="card-header">  <h3 class="h6 text-uppercase mb-0">熱門關鍵字瀏覽與繪圖(資料週期:資料截止時間的前4周)</h3>  </div>  <div class="card-body">  <!--新聞類別選單 form group-->  <div class="form-group row">  <label class="col-sm-3 form-control-label">新聞類別</label>  <div class="col-md-9">  <select id="cate-selected" name="news\_category" class="form-control">  <!--<option>請選擇</option>-->  <option>全部</option>  <option>政治</option>  <option>科技</option>  <option>娛樂</option>  <option>運動</option>  <option>生活</option>  <option>產經</option>  <option>證卷</option>  <option>國際</option>  <option>社會</option>  <option>文化</option>  <option>兩岸</option>  </select>  <small class="form-text text-muted">請選擇新聞類別  </small>  </div>  </div>  <!--form group-->  <!--熱門詞多少個?form group-->  <div class="form-group row">  <label class="col-md-3 form-control-label">多少個熱門詞?</label>  <div class="col-md-9">  <input id="topk-selected" name="topk" value="10"  class="form-control form-control-success">  <small class="form-text text-muted">內定值為10  </small>  </div>  </div>  <!--form group-->  <!--submit按鈕form group-->  <div class="form-group row">  <div class="col-md-9 ml-auto">  <button type="button" id="btn-ok" class="btn btn-primary">查詢</button>  </div>  </div>  <!--form group-->  </div>  <!--card body-->  </div>  <!--column-->  </div><!-- 區塊結束 -->  <!-- 繪圖區塊--------------------------------------------------------->  <div class="col-lg-6 mb-5">  <div class="card">  <div class="card-header">  <h3 class="h6 text-uppercase mb-0">熱門關鍵字繪圖</h3>  </div>  <div class="card-body">  <canvas id="mychart"></canvas>  </div>  </div>  </div><!-- 區塊結束 -->  <!-- 熱門關鍵字區塊----------------------------------------------------->  <div class="col-lg-6 mb-5">  <div class="card">  <div class="card-header">  <h3 class="h6 text-uppercase mb-0">熱門關鍵字</h3>  </div>  <div class="card-body">  <ul id="topkeys"></ul>  </div>  </div>  </div><!-- 區塊結束 -->  {% endblock %}  {% block extra\_js %}  <!-- chartjs圖js-->  <script src="https://cdnjs.cloudflare.com/ajax/libs/Chart.js/2.7.3/Chart.min.js"></script>  <!-- 程式碼區 -->  <script>  // Show default top words bar chart when the page is loaded.  // Invoke callAjax() after the function is defined, unless callAjax() is defined with "hoisting"  callAjax();  //\*新聞類別選單select被選中值有改變時，執行以下事件  //$('#cate-selected').on('change', callAjax() );  $('#cate-selected').on('change', function () {  callAjax();  }); //event function  //\*\*按鈕事件  $('#btn-ok').on('click', function () {  callAjax();  }); //event function  // \*\* draw chart using Ajax 畫圖  // There are two ways to define a function:  // one is expression, and the other is definition with "hoisting"  // function with hoisting: function callAjax() {}  // normal function expression: let callAjax = function() {}  // Define callAjax function with hoisting  // callAjax()這樣定義可以在被定義前就被使用 跟我們在Java裡面的函數用法一樣!  //let callAjax = function() {  function callAjax() {  let cate = $('#cate-selected').val();  //console.log(cate);  let topk = $('#topk-selected').val();  //console.log(topk);  $.ajax({  type: "POST",  //url: "/topword/api\_get\_cate\_topword/",  url: "http://127.0.0.1:8000/topword/api\_get\_cate\_topword/",  //url: "http://163.18.23.21:8000/topword/api\_get\_cate\_topword/",  //url: "api\_get\_cate\_topword/", //Not recommended!  data: {  "news\_category": cate,  "topk": topk  },  success: function (received) {  //console.log(received);  let chart\_data = received.chart\_data;  let wf\_pairs = received.wf\_pairs;  console.log(wf\_pairs)  showTopKeys(wf\_pairs);  showChart(chart\_data);  } //success function  }); //ajax  } //callAjax  //\* 顯示關鍵詞資料函數  function showTopKeys(items) {  //先清除前一次的資料  $('#topkeys').empty();  //將內容加上li標籤附加起來，顯示在顯示區"topkeys"  for (let i = 0; i < items.length; i++) {  let item\_li = "<li>" + items[i] + "</li>";  $('#topkeys').append(item\_li);  }  } //function  //\*\*繪圖函數showChart()  function showChart(chart\_data) {  // 畫圖需要的數據資料  let values = chart\_data.values;  let labels = chart\_data.labels;  let category = chart\_data.category;  //第1個變數: 餵給chart的資料  let data = {  labels: labels,  datasets: [{  label: category,  data: values,  backgroundColor: randomColors(values.length),  borderColor: randomColors(values.length),  borderWidth: 1,  }],  };  //第2個變數: chart的選項 指定y坐標軸從零開始顯示  let options = {  scales: {  yAxes: [{  ticks: {  beginAtZero: true  }  }]  },  };  //取得在前面html區域欲顯示的圖代號  let canvas\_mychrat = document.getElementById("mychart");  //\*\*先清除前一個圖 再繪新圖  // 可以印出barchart物件是否存在  // console.log(window.barchart);  //先清除前一個圖 再繪新圖 if 有以下兩種寫法皆可  // if (window.barchart) //若存在則為true  // if (typeof (barchart) != "undefined"){  if (window.barchart) {  barchart.destroy();  }  //\*\*繪圖(產生一個圖物件變數名稱為barchart)  // 必須全域變數--注意:前面不要有let, var, const等修飾詞  // 理由: 我們要讓它存在於網頁全域變數，  // 這樣我們才方便判斷是否有前一次的圖，如果存在有，要刪除之，否則，很多張圖會疊在一起  barchart = new Chart(canvas\_mychrat, {  type: 'bar',  data: data,  options: options,  });  //\*\* 產生隨機顏色  function randomColors(num\_colors) {  let colors = [];  for (i = 0; i < num\_colors; i++) {  let r = Math.floor(Math.random() \* 255);  let g = Math.floor(Math.random() \* 255);  let b = Math.floor(Math.random() \* 255);  let rgb = `rgba(${r},${g},${b},0.5)` // (red, green, blue, alfa) alfa透明度  colors.push(rgb);  }  return colors;  }  } //show chart function  // document就是這個網頁HTML所有的元素  // window就是這個網頁的全域變數global variables:有一大堆，我們自己定義的有callAjax, showChart, barchart等  // 把document, window印出來看看就能理解它們是甚麼  //console.log(document);  //console.log(window);  </script>  {% endblock %} |

base.html

|  |
| --- |
| <!DOCTYPE html>  {% load static %}  <html lang="zh-TW">  <head>  <title>  {% block title %}  輿情分析平台  {% endblock %}  </title>  <meta charset="utf-8" />  <meta name="viewport" content="width=device-width, initial-scale=1" />  <!-- Bootstrap 5 CSS -->  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet" />  {% block extra\_css %}  {% endblock %}  </head>  <body>  <div class="container">  <div class="row">  <!-- Navigation Bar -->  <div class="col-lg-12 mb-2 mt-2">  <nav class="navbar navbar-expand-lg navbar-light" style="background-color: #e3f2fd">  <div class="container-fluid">  <a class="navbar-brand" href="/">輿情大數據</a>  <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation"><span class="navbar-toggler-icon"></span></button>  <div class="collapse navbar-collapse" id="navbarSupportedContent">  <ul class="navbar-nav me-auto mb-2 mb-lg-0">  <!-- 基礎熱門分析 -->  <div class="btn-group">  <button type="button" class="btn dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">熱門分析</button>  <div class="dropdown-menu">  <a class="dropdown-item" href="{% url 'app\_top\_keyword:home' %}">熱門關鍵詞分析</a>  <a class="dropdown-item" href="#">熱門人物排行分析</a>  <a class="dropdown-item" href="#">命名實體熱門分析</a>  <a class="dropdown-item" href="#" style="color: green">昨日誰最大</a>  </div>  </div>  <!-- 進階自訂分析 -->  <div class="btn-group">  <button type="button" class="btn dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">進階查詢</button>  <div class="dropdown-menu">  <a class="dropdown-item" href="#">自訂關鍵詞熱門度分析</a>  <a class="dropdown-item" href="#">自訂全文檢索與關聯分析</a>  <a class="dropdown-item" href="#">自訂關鍵詞之情緒分析</a>  </div>  </div>  <!-- 特色分析 -->  <div class="btn-group">  <button type="button" class="btn dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">特色分析</button>  <div class="dropdown-menu">  <a class="nav-link" href="#" style="color: green">時中聲量我最大</a>  <div class="dropdown-divider"></div>  <a class="nav-link" href="#" style="color: green">台北市長選舉聲量觀測站</a>  <div class="dropdown-divider"></div>  <a class="nav-link" href="#">政黨聲量排行榜</a>  <a class="nav-link" href="#">政治人物聲量排行榜</a>  <div class="dropdown-divider"></div>  <a class="nav-link" href="#" style="color: green">事件人物聲量相關分析</a>  </div>  </div>  <!-- 新聞推薦系統 -->  <div class="btn-group">  <button type="button" class="btn dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">新聞推薦系統</button>  <div class="dropdown-menu">  <a class="dropdown-item" href="#">Bert/Qwen新聞推薦-新聞查找相似新聞</a>  </div>  </div>  <!-- NLP應用 -->  <div class="btn-group">  <button type="button" class="btn dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">自然語言理解</button>  <div class="dropdown-menu">  <a class="dropdown-item" href="#">語言模型Bert/Qwen情緒分類</a>  <a class="dropdown-item" href="#">語言模型Bert/Qwen新聞分類</a>  </div>  </div>  <!-- 使用資料庫 -->  <div class="btn-group">  <button type="button" class="btn dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">使用DB</button>  <div class="dropdown-menu">  <a class="dropdown-item" href="#">熱門人物資料庫查詢</a>  <a class="dropdown-item" href="#">昨日新聞資料庫全文檢索</a>  </div>  </div>  <!-- 其他特色應用 -->  <div class="btn-group">  <button type="button" class="btn dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">特色應用</button>  <div class="dropdown-menu">  <a class="dropdown-item" href="#">高雄市市長選舉人氣分析</a>  <a class="dropdown-item" href="#">珊珊市長人氣</a>  <a class="dropdown-item" href="#">阿邁市長人氣</a>  <div class="dropdown-divider"></div>  <a class="dropdown-item" href="#">疫情發燒監視</a>  <a class="dropdown-item" href="#">蘋果發燒站</a>  <div class="dropdown-divider"></div>  <a class="dropdown-item" href="#">人氣美食排行榜</a>  <a class="dropdown-item" href="#">財經議題排行榜</a>  <a class="dropdown-item" href="#">股市新聞監視站</a>  <a class="dropdown-item" href="#">科技新聞熱門議題分析</a>  <a class="dropdown-item" href="#">人力銀行職缺大熱門</a>  <div class="dropdown-divider"></div>  <a class="dropdown-item" href="#">批踢踢酸民大吐槽</a>  <a class="dropdown-item" href="#">批踢踢鄉民來八卦</a>  <a class="dropdown-item" href="#">滴卡大學生關心議題</a>  <div class="dropdown-divider"></div>  <a class="dropdown-item" href="#">新聞媒體政黨傾向調查分析</a>  <a class="dropdown-item" href="#">自訂競爭大PK</a>  </div>  </div>  <!-- 會員訂閱 -->  <div class="btn-group">  <button type="button" class="btn dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">付費訂閱</button>  <div class="dropdown-menu">  <a class="dropdown-item" href="#">長時間(超過3個月)監測分析(收費表)</a>  <a class="dropdown-item" href="#">更多資料庫分析(收費表)</a>  <a class="dropdown-item" href="#">為你訂製輿情分析任務(報價)</a>  </div>  </div>  <!-- 關於 -->  <div class="btn-group">  <button type="button" class="btn dropdown-toggle" data-bs-toggle="dropdown" aria-expanded="false">關於</button>  <div class="dropdown-menu">  <a class="dropdown-item" href="#">課程介紹網頁</a>  <a class="dropdown-item" href="#">如何使用輿情API</a>  </div>  </div>  </ul>  </div>  </div>  </nav>  </div>  <!-- End of Navigation Bar -->  <!-- Main Content -->  {% block content %}  {% endblock %}  </div>  </div>  <!-- Bootstrap 5 JS Bundle with Popper -->  <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>  <!-- Additional Scripts -->  {% block extra\_js %}  {% endblock %}  </body>  </html> |

How to define url in Ajax?

你的API程式開放的IP? 127.0.0.1

啟動網頁時要使用http://127.0.0.1:8000/topword/

* Call local api service. 呼叫本機端localhost的API程式

url: "/topword/api\_get\_cate\_topword/",

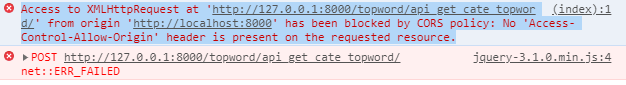
* 呼叫http://127.0.0.1的API程式

url: "http://127.0.0.1:8000/topword/api\_get\_cate\_topword/",

Why do your page (localhost:8000/topword/) be blocked by CORS policy?

Do not load your page using “localhost:8000/topword/”

Or you your page will be blocked by CORS policy.



Access to XMLHttpRequest at 'http://127.0.0.1:8000/topword/api\_get\_cate\_topword/' from origin 'http://localhost:8000' has been blocked by CORS policy: No 'Access-Control-Allow-Origin' header is present on the requested resource.

你的API服務不接收來自於跨站的請求!! 因為，我們的Djanog網站目前還沒設定允許CORS，日後會再介紹!