Project Report - Visualization Week 4

Workload Division

- Fill in the names and email addresses of your group members and describe how you divide works among team members;
  - Team members:

    李炳楠 libn@shanghaitech.edu.cn

    秦一帆 qinyf1@shanghaitech.edu.cn

    陈溯汀 chenst@shanghaitech.edu.cn
  - Workload division:

    李炳楠：writing vis.html and beautifying the `home page.html` and `handler.html`

    秦一帆：beautifying the image which shows the location of flights, adding labels to the flights and adding arrows to shows its direction.

    陈溯汀：origin work of data processing and draw picture. Writing the `demo.py` file.

Preliminary Comment

- Please cite any online or offline resources you consulted in this project;
- online resource:
  1. https://matplotlib.org/3.3.3/api/_as_gen/matplotlib.pyplot.legend.html?highlight=legend#matplotlib.pyplot.legend
  2. https://matplotlib.org/3.3.3/tutorials/index.html
  3. https://matplotlib.org/3.3.3/api/_as_gen/matplotlib.pyplot.html
- offline resources:
  1. Eric Matthes (July, 2016). *Python Crash Course: A Hands-On, Project-Based Introduction to Programming*. No Starch Press, Page 285-295
- Please describe the difficulties you encountered in this project;

  adding legend to the picture

  showing the direction of flights

  refresh the page with newest data

Advanced: `matplotlib` and Flask

- How to create a plot in `matplotlib`?

By use the `matplotlib.pyplot.figure()`, we can create a plot in `matplotlib`. Or by using `matplotlib.pyplot.subplot()` or `matplotlib.pyplot.subplots()`, we can create a subplot in `matplotlib`.

- How to draw line graph (折线图), histogram (直方图), bar chart (条形图) and pie chart (饼状图)?

  By using `matplotlib.pyplot.plot()`, we can create a line graph. And the list passing into the parameter of this function is the points of the line graph.(not finished)

- How to change the legend, x-axis label and y-axis label of a graph?

  `plt.legend()` means showing the legend of data. And `plt.set_label()` can change the legend.

  ```
  plt.xlabel('longitude')
  plt.ylabel('latitude')
  plt.axis("equal")
  ```

  means setting label of x or y axis to 'longitude' and 'latitude' and setting the length of the axis equal.

- How to save the plot as a image?

  The plot can be saved to a image file by using the function `plt.savefig("filepath")` and the image will be saved into the file. Following is an example.

  ```
  if __name__ == "__main__":
      plt.savefig('./static/image1.jpeg')
  else:
      plt.savefig('./web_server/static/image1.jpeg')
  ```

  means saving the plot as a image to the path.

- How to serve image (or any static file) with Flask?

  We can create a file holder named 'static' so that flask can get the images inside the folder and serve the file to users as a static file. Flask get the file by using routes, for example, `src=/static/image.jpeg` enables the Flask to get the image named 'image.jpeg' under the 'static' folder.

- How to add a route (that handles new URLs) to Flask?

  We can add a new route to Flask by using `@web_server.route("new_route")`. Then Flask can handle the new URL. Following is an example.

  ```
  @web_server.route("/nothing")
  def nothing():
      return "nothing here! :("
  ```

- How to render a HTML template with Flask with parameters?

We can render a HTML template by using {{}} in the HTML file. And the word inside the braces can be served as a index of the parameter. For example, the 1.html file has {{error}} in it. And in python we can use function `render_template(1.html, error="bad value")` and the {{error}} will be changed into "bad value". Following is another example. In `home page.html`:

```
<head>
    <title>{{title}}</title>
</head>
...
<p>
    <br/>
    <br/>
    <br/>
    {{handler}}
    <br/>
    <br/>
    <br/>
</p>
```

And in the python file:

```
return render_template("handler.html",title="Succeeded",handler="传入参数成功")
```

It can change {{title}} and {{handler}} into "Succeeded" and "传入参数成功"

Implementation

- Describe the overall workflow of this part of this project, including answers to the following bulletin points.
    - When do you update your graph? When the new data comes, or when the user request comes?

      When the user request comes, we update our graph.
    - How do you store the data used for rendering the graph?

      The data used for rendering the graph is stored under the path `./data_source/flight_data.csv`.
    - How do you store the graph after being rendered by `matplotlib`?

      The graphs are stored under the path `./web_server/static` as the static files for the html.
    - How is the image served to the user?

      By using html templates.

After the user request comes, the function under `@web_server.route('/vis', methods=['GET'])` will be run. The program will first get the data from the file `flight_data.csv`. Then, it will process the code and draw the graphs. The graphs will be saved as images under path `./web_server/static`. Then we can present the image to the users by using the template. The images will be rewrite after every request.