Project Elective

Map Analytics - Endterm Report

Nilay Kamat

May 2024

Contents

Chapter 1

Extraction of Numerical Data

1.1 Introduction

The task proposed in the project was to retrieve the numerical data present in each map image and evaluate it for the states in the USA. The extraction of the numerical data is being done with the help of OCR techniques.

1.2 Problem Statement

Given any map image containing certain data about the USA and represented using Choropleth or Isocline maps, our task is to extract the numerical data that was used to create the thematic map. The challenges include working with different types of data depictions - Colour Bar or Discrete Legends, leveraging OCR techniques to obtain decent results of the numerical data and to extract the data in a format such that an analysis can be made using the same.

1.3 Approach

1.3.1 Super Resolution(ESRGAN) followed by PaddleOCR

Due to the drawbacks faced when using Tesseract OCR, we decided to enhance the images provided and then apply an OCR model to these enhanced images. We performed enhancing using ESRGAN(Enhanced Super-Resolution Generative Adversarial Networks), a method for single image super-resolution that aims to generate realistic textures and details while enhancing the perceptual quality.

Followed by image enhancement using ESRGAN, we leveraged the PaddleOCR model to obtain OCR results from the map image on the relevant numerical data. Similar to the code used for the TesseractOCR model, we used a Jupyter Notebook/Python script to obtain output of the average value, the unit, the colour associated with the value and the map number(present in the image title). There were slight changes that had to be made in the code due to the change in the model. The values in the legends are given in the format (a−b) followed by some units. Hence, using some post-processing of the value detected using the OCR model, we stored the average value and the unit along with the colour corresponding to this obtained average value.

1.3.2 Code explanation

In the pipeline of the application, the OCR step is preceded by the super-resolution and annotation+classification step. Hence, we obtain the bounding boxes of the **title**, **map** and **legend/colour bar** along with the type of map image we are dealing with for all the map images given as input to the application. Using this information(stored in a CSV format), we prepare 2 sub-images(cropped images) of the original map image for the title and the legend/-colour bar information. Then, we perform OCR on these 2 cropped images using PaddleOCR and prepare the following information:

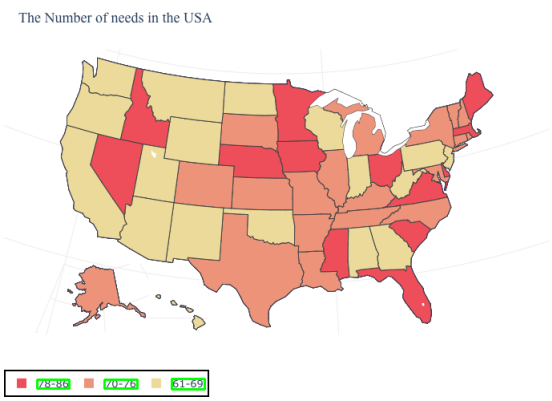
1. Map Name
2. Map Type
3. Map Title
4. RGB colour of **a** value(stored as R, G, B)
5. Value(if range, average considered)
6. Unit

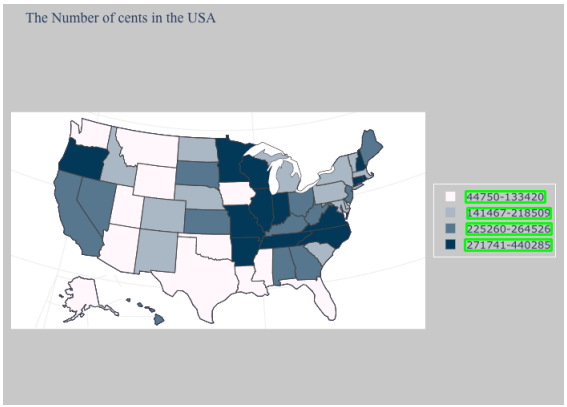
This information is then stored in CSV format and passed to the next step in the pipeline - **Colour Matching/Association**. There were quite a few changes to this section of the application to cater to multiple image input, faults in initial thought process and logic and in order to obtain an output useful and easily accessable by the next sections of the application.

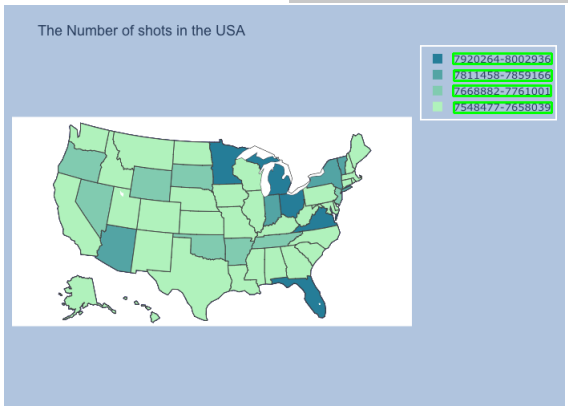
1.4 Results

We used a couple of map images from the MapQA dataset to verify the working of our script and to check if the output obtained aligned with the expectations and requirements. Images obtained on using PaddleOCR model after enhancing the images using ESRGAN were as follows:

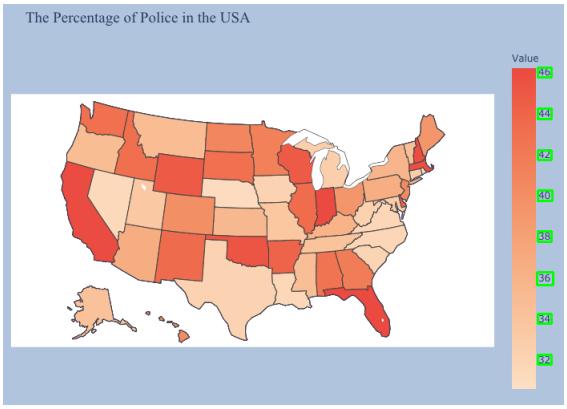
1. Discrete Legends

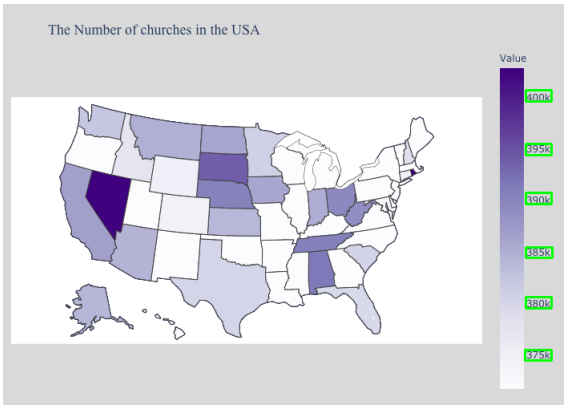






2. Colour Bar





Formatted output of 2 map images

1. ('map_16.png', 'discrete', 'Health Insurance Coverage of Adults 19-24 — KFF', (236, 218, 154), 3.7, %)
2. ('map_16.png', 'discrete', 'Health Insurance Coverage of Adults 19-24 — KFF', (236, 171, 132), 5.4, %)
3. ('map_16.png', 'discrete', 'Health Insurance Coverage of Adults 19-24 — KFF', (237, 124, 111), 7.35, %)
4. ('map_16.png', 'discrete', 'Health Insurance Coverage of Adults 19-24 — KFF', (238, 77, 90), 9.5, %)
5. ('map_51.png', 'continuous', 'Health Insurance Coverage of Low Income Adults 19-64 (under 200% FPL)', (84, 39, 143), 200, k)
6. ('map_51.png', 'continuous', 'Health Insurance Coverage of Low Income Adults 19-64 (under 200% FPL)', (122, 114, 180), 150, k)
7. ('map_51.png', 'continuous', 'Health Insurance Coverage of Low Income Adults 19-64 (under 200% FPL)', (174, 172, 210), 100, k)
8. ('map_51.png', 'continuous', 'Health Insurance Coverage of Low Income Adults 19-64 (under 200% FPL)', (224, 224, 238), 50, k)

As can be seen, the CSV file stores all the required numerical data present in map images, be it a continuous or discrete map image.

The codes and results can be found at the following repository :- **Map Analytics GitHub - OCR**.

Chapter 2

Colour Association/Matching

2.1 Objective

The task is to retrieve the numerical value corresponding to the states present in the USA using the output of the segmentation(which provides the colour representation of the state) and the extracted numerical data(using OCR techniques).

2.2 Approach

We will be using the output obtained from the modified segmentation task along with the extracted numerical data to perform an analysis on the map image. Our input sources provide information regarding the RGB value of colour of the states in the map and the extracted values present in the image along with their corresponding value. There are certain different approaches that we experimented with in obtaining the similarity between the RGB values provided in the inputs in order to make an association/matching of sorts to evaluate the value.

In order to make an association, the logic/work flow used was to obtain the state's(under observation) colour as given in the output of the segmentation task. Using this colour representation and the colour representations along with their values present in the extracted values(via OCR), an association/matching could be made using some form of similarity metric. We used 2 different similarity metrics for either of the cases of Discrete Legends and Colour bar(this information about the map image can be extracted from the OCR data).

There were quite a few changes to this section of the application as well to cater to multiple

image input, faults in initial thought process and logic and in order to obtain an output useful and easily accessable by any next sections of the application. There were also many issues that were found during testing the dataset we had and checking evaluations and accuracies.

2.2.1 Input Files

We have 2 input files - from segmentation of the map image and from the extraction of numerical data.

OCR data examples(CSV)

1. ('map_16.png', 'discrete', 'Health Insurance Coverage of Adults 19-24 — KFF', (236, 218, 154), 3.7, %)
2. ('map_16.png', 'discrete', 'Health Insurance Coverage of Adults 19-24 — KFF', (236, 171, 132), 5.4, %)

Segmentation data examples(CSV)

1. ('map_17.png', 'Massachusetts', 1, (158.55921052631578, 585.9934210526316), (0, 150, 570, 1, 166, 604), (141, 113, 163))
2. ('map_17.png', 'New Mexico', 2, (285.4919093851133, 247.98312528895053), (1, 249, 213, 2, 321, 283), (249, 221, 218))
3. ('map_17.png', 'Oklahoma', 3, (275.2837675350701, 340.0048096192385), (2, 256, 285, 3, 299, 372), (249, 221, 218))
4. ('map_17.png', 'Kentucky', 4, (244.01780626780626, 464.2877492877493), (3, 225, 427, 4, 261, 495), (195, 167, 190))
5. ('map_17.png', 'Lousiana', 5, (334.58353658536583, 400.8469512195122), (4, 309, 379, 5, 358, 432), (249, 221, 218))

We coded a Python script to use the 2 input CSV files and make a similarity comparison for each state present in segmentation task output. Choosing the colour with the most similarity, we then assigned the colour's corresponding value to that state, writing it an output CSV file that shall serve as the output for the Colour Matching task.

2.3 Colour Similarity Metrics

After attempting to use multiple colour similarity metrics such as **Euclidean Distance**, **Weighted Euclidean Distance** and **Piecewise Linear Interpolation**, we decided to use separate colour metrics for the 2 different types of images.

2.3.1 Discrete Legends

For map images with Discrete Legends, since the colour can only be associated with one value and not a range of values, the metric of **Euclidean distance** to calculate similarity is justified. Hence, we leveraged the concept of Euclidean distance and calculated the similarity of the state's colour and the colours present in the legends. The smallest distance corresponded to the most similar colour and thus, we assigned that value to the corresponding state.

2.3.2 Colour Bar

For map images with Colour, since the colour can lie within a range of values, the metric of Euclidean distance to calculate similarity is not justified and cannot be used. Hence, we leveraged the concept of **Piecewise Linear Interpolation** and calculated the similarity of the state's colour and the colours present in ranges in the colour bar.

For each range present in the colour bar, we evaluated the parameter α using the following formula:

$$\alpha = \frac{(state\_colour - C\_2)}{(C\_1 - C\_2)}$$

α was calculated as 3 components - R,G and B. Hence, the above formula was used 3 times for each component of the colour. α is then the average of the all the non-zero components.

If the value of α lies between 0 and 1, then we know that the colour lies in that range and thus, we evaluate the colour as follows:

$$\text{Value} = \alpha * (C\_1 - C\_2) + C\_2$$

Improvement: There were some map images where the corresponding colour was observed to be outside the values given at the edges of the colour bar. In this case, the logic used above was failing. Thus, we decided to add a delta factor while checking if the colour lies in the range. This delta factor was added only to the first and last range to correct any errors that could

arise. The delta value has been set at 0.5 to account for even large difference in the actual extreme values in the colour bar and the value we obtain via the OCR implementation.

2.4 Results

The results obtained are stored as a CSV file containing the state name and columns for each map image. The values of the states corresponding to each map image was duly filled in the respective element position.

Output(Colour Matching Analysis)

State_Name	Health Insurance Coverage of Adults 19-24 KFF	HIC of Men 19-24 KFF
Alabama	9.5%	91150.0u
Alaska	5.4%	33750.0u
Arizona	7.35%	91150.0u
Arkansas	9.5%	185200.0u
California	5.4%	33750.0u
Colorado	9.5%	33750.0u
Connecticut	9.5%	33750.0u
Delaware	0	33750.0u

As can be seen from the output obtained, the colour matching is giving results as expected. Each map image's title is stored as a column and all the corresponding values for the states are obtained in the respective elemental positions. This concludes the Colour Matching section of the application.

The codes and results can be found at the following repository :- **Map Analytics GitHub - Colour Matching**.