Crawler and data visualization by python

Yun Zhang
Bartow Hanvos Kent School Ningbo Campus
Email: sophiazhangbarsow@outlook

***Abstract* – Have you used smart technology to identify objects in pictures? That is accomplished by the use of Object Detection Algorithm, which is an advanced technology to recognize the information on image and transfer it to readable text. Traditional methods such as AI, Artificial Intelligence including Machine Learning, algorithm exist few shortcomings, including low resistance of noise and disturbance, leading to mis-recognition. However, following will introduce or display a hybrid method of Object Detection Algorithm by which information can be filtered automatically and even can achieve 75,8% accuracy.**

I. Introduction

The question of our Kaggle competition is Kuzushiji recognition which is a cursive writing style had been used in Japan for over a thousand years. There are very few readers of Kuzushiji today, who are only 0.01% of modern Japanese natives. The object is using Machine Learning to automatically recognize these historical texts and transcribe them into modern Japanese characters. The final model is not only a great contribution to the machine learning community, but also a great help for making millions of documents more accessible and leading to new discoveries of Japanese history and culture.

This report mainly explain the utilization of EDA(Exploratory Data Analysis) for preprocessing the data, and try to discover underneath features. The primary EDA is calculating the number of words in the Kuzushiji context and presents them in an ascent order of frequency.

OCR(Optical character recognition) has been rooted in many aspects of people's lives, including scanners, photographs to find related goods, and cameras in the parking lot entries etc. But in terms of achieving advanced OCR technology, scientists have developed neural network to make detection more precisely.

Neural network contains three layers -- input layer, hidden layers, and output layer -- and for CNN（Convolution Neural Networks）, the input which holds neurons that have pixels of the picture and biases. Neurons are interconnected by weights, another parameter to adjust for a more accurate result. To update the parameters of the neural network, people use loss function, which is used to describe the differences between predicted values and underground truth values. Therefore, adjust bias and weights, two important parts in network, to get a small result of the loss function, and the smaller is, the better the performance is.

One of the method we use is called Exploratory Data Analysis, EDA in short, which is an approach to analyze data sets to summarize their main characteristics, often with visual methods. Unlike a regular statistical model which only demonstrate single ambiguous relationship of data, EDA primarily is for seeing what the data can tell people beyond the formal modeling or hypothesis task. For recognizing Kuzushiji , EDA can be a approachable method to calculate the most common words in the context and display the number of each word on a graph from the highest frequency to the lowest. However, in terms of maximize the preciseness of the result, words that only appear a few times can be ignored.

II. Materials and Methodology

The content of the method EDA is as following:

1. New df_train
2. missing data
3. char stats
4. Graph of Top 200 words
5. Graph of cumulative frequency of the words in Kuzushiji context

A. EDA

Following image is one page of a book in Kuzushiji writing style. We visualize the training dataset as following, according to the location points, width and length showed in the dataset, we drawing red box around the characters.

In order to find more perspectives, we read the image and the training data, input them in a data set called "df_train", a data frame that can be read and operated by the computer, and assign a Unicode for each Kuzushiji character and add them into a chart. Code as following:

```
Unicode of Kuzushiji character

PATH = '../input/kuzushiji-recognition/'
df_train = pd.read_csv(PATH+'train.csv')
df_test = os.listdir(PATH+'test_images/')
unicode_map = {codepoint: char for codepoint, char in
pd.read_csv(PATH+'unicode_translation.csv').values}
print ("TRAIN: ", df_train.shape)
print ("TEST: ", len(df_test))
df_train.head()
```

And here is a chart of Unicode with its corresponding Kuzushiji character:


```
TRAIN:  (3881, 2)
TEST:  4150
```

As a result, there are totally 4212 distinct Kuzushiji characters, but in the training set only 3881 characters with labels, and 274 without labels. Therefore, codes to drop these characters are needed in order to pursue the convenience and preciseness of the result. Codes as following:

Drop characters without labels

```
#df_train.dropna(inplace=True)
df_train.reset_index(inplace=True, drop=True)
print ("TRAIN: ", df_train.shape)
```

Secondly, create a dictionary, {} marks, called “Chars” and add each label with U as capital -- like “U + 306F” “U + 304C” -- into the dictionary, so that we can count the number of meaningful characters in each image.
Codes as following:

Chars dictionary

```
chars = {}
for i in range (df_train.shape[0]):
    try:
        a = [x for x in df_train.labels.values[i].split(' ') if x.startswith('U')]
        n_a = int(len(a))
        for j in a:
            if j not in chars: chars[j]=1
            else:
                chars[j]+=1

        a = " ".join(a)

    except AttributeError:
        a = None
        n_a = 0

    df_train.loc[i,'chars'] = a
    df_train.loc[i,'n_chars'] = n_a
    df_train.head()
```

The result is a chart with Unicode of each characters in Chars dictionary and the number of such Unicode in one image. For the result below, there are totally 67 characters in image id “100241706_00004_2”, and 80 in image id “100241706_00005_1” etc.

	image_id	labels	chars	n_chars
0	100241706_00004_2	U+306F 1231 3465 133 53 U+304C 275 1652 84 69 ...	U+306F U+304C U+3044 U+3051 U+306B U+306B U+30...	67.0
1	100241706_00005_1	U+306F 1087 2018 103 65 U+304B 1456 1832 40 73...	U+306F U+304B U+304B U+3044 U+306B U+306B U+30...	80.0
2	100241706_00005_2	U+306F 572 1376 125 57 U+306E 1551 2080 69 68 ...	U+306F U+306E U+3078 U+304C U+306B U+3081 U+30...	78.0
3	100241706_00006_1	U+3082 1455 3009 65 44 U+516B 1654 1528 141 75...	U+3082 U+516B U+309E U+306B U+308B U+304B U+30...	72.0
4	100241706_00007_2	U+309D 1201 2949 27 33 U+309D 1196 1539 27 36 ...	U+309D U+309D U+309D U+309D U+3078 U+309D U+25...	167.0

After importing a statistic graph for visualizing the number of each character, the next step is to count the number of each character.

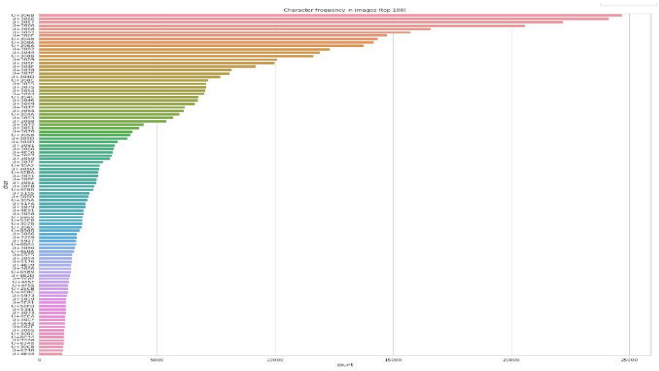
Count the number of Kuzushiji characters

```
chars = pd.DataFrame(list(chars.items()), columns=['char', 'count'])
chars['jp_char'] = chars['char'].map(unicode_map)
print (" >> Chars dataframe <<")
print("Number of chars: ",chars.shape[0])
chars.to_csv("chars_freq.csv",index=False)chars.head()
```

Result as following, character “U+306F” appears 14759 times, and character “U+304C” appears 6740 times, etc.

	char	count	jp_char
0	U+306F	14759	は
1	U+304C	6740	が
2	U+3044	11903	い
3	U+3051	4224	け
4	U+306B	24685	に

Therefore, by ordering these data from the high frequency to low, the top 100 characters can be acquired. As the graph below, the most common Kuzushiji character appears up to 14700 times, leaving a tremendous difference with the least common character.



By contrasts, 2273 out of 4212 total characters only appear less than 10 times, which are called rare chars, and word like “U + 5541” only appears one time throughout the entire book, a incredible number.
Code is following:

Counting rare characters

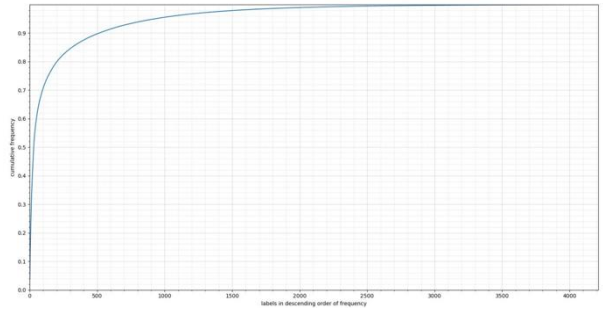
```
print ('Total chars', chars.shape[0])
print ('< 10 freq', chars[chars['count'] <= 10].shape[0])
rare = chars[chars['count'] <= 10]
print (rare.shape)
rare.head()
```

Following is the result:

	char	count	jp_char
72	U+5541	1	啁
89	U+51E0	7	几
90	U+8A1B	7	訛
135	U+92F3	2	鋳
142	U+5C41	4	屁

As for the cumulative frequency, it begins with accumulating

the frequency of each character and adding them up. For the following result of the cumulative frequency, the X-axis presents the labels in descending order of the frequency, where one unit is 100. The Y-axis is the cumulative frequency from 0.0 to 1 regarding to the content of the entire data set. Therefore, as the graph demonstrates, the top 200 common characters constitute 80% of the whole context, and top 500 occupy 90%.



Based on above data analysis, we team uses a two-stage approach: Detection + Classification. Out-of-the-box object detection methods, such as Faster-RCNN, SSD, can solve these two problems together. However, our experiments showed bad results. Therefore, we firstly detect the kuzushiji characters with Faster-RCNN/RetinaNet detector and classify the proposals later with ResNet50/Xception.

III. Discussion and Evaluation

EDA is a direct visualization of the total number of each Kuzushiji character in the entire context. Users can clearly look at the graph and be informed which character is the most common word in the book and which is the least. It is an analytical method by which the neural network can be created to recognize the features of the Kuzushiji characters.

By participating in this Kaggle competition, though my report paid more attention on previous data exploration rather than Neural Network architectures, I learned the basic function of machine learning which is significantly helpful to me. Before this course, I did not have any knowledge of neural network and its broad application in our daily life. However, through the two-month study, I have gained the basic understanding of what neural network is and how it works. I am not ignorant any more. After this class, when I use the scanner on the phone, I know how a complex neural network is under the screen and how does the information is transformed and then the phone can recognize the image. I start to be interested in the study of machine learning. I may apply the knowledge that I learned now into my job, so that my work is more comprehensive and more professional.

IV. Reference

Jesucristo. “Kuzushiji Recognition Complete Guide.” *Kaggle*. Kaggle, 22 Oct. 2019. Web. 28 Oct. 2019.

[https://www.kaggle.com/jesucristo/kuzushiji-recognition-complete-guide/notebook]