Creation of a text recogniser based on OpenCV and Keras for parsing and analysis of ISDA CSA bank agreement

Big Data, Machine Learning and Applications to Economics and Finance

Maria Dmitrieva - 22.12.2019

Abstract:

The analysis of bank circulation is an extremely important but routine process. The ISDA (CSA) plays a significant role within the framework of risk management in the financial markets - it contains a large number of metrics, within which OTC derivatives are traded. This work will present a simple text recogniser, which will select the necessary metrics using machine learning and put them in a common table.

Table of contents:

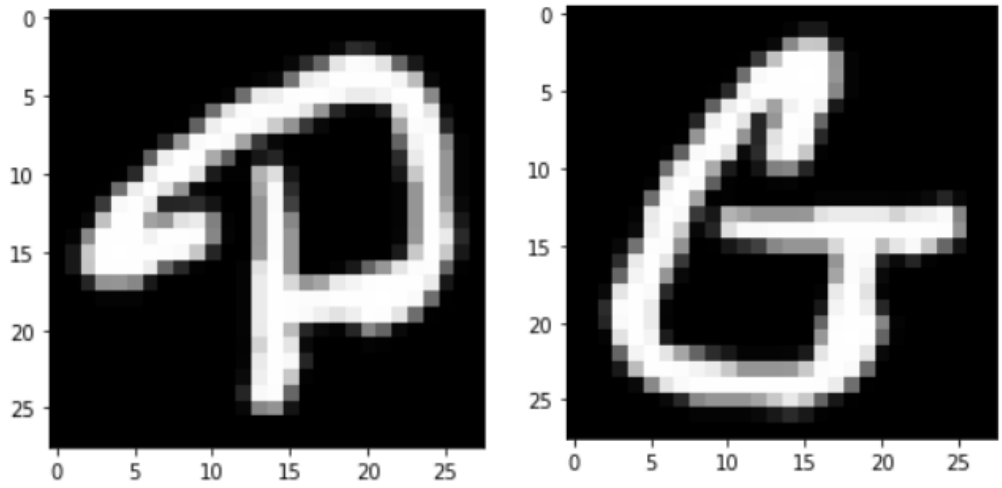
Introduction

My project is a reflection of my work experience. As part of the automation of PNL settlements for transactions of over-the-counter derivatives, one often has to turn to ISDA (CSA) agreements - they contain such parameters as initial margin, thresholds, minimum transfer amount, and so on. Since documents are signed live, only scans of these documents are usually available, which complicates data aggregation. In this work, I will try to aggregate the main metrics of this ISDA, I am going to use machine learning to recognise individual letters, highlight common patterns and the information I need.

Data Description

EMNIST Dataset

Since bank documents are commercial secrets and their use and publication outside the bank is punishable as part of compliance, I propose to use the public library EMNIST as training dataset. It is available on Kaggle as well as in the archive and article [1]. The original size is 28*28 pixels.



Picture 1 Part of EMNIST dataset (original and with rotation)

There are several procedures that required to process data for test sample. First, it is the rotation of all dataset on 90 degrees to make it appropriate for future preparations. Second, in the set up of the neural network, I also implement Keras Convolution2d() - this is a set up, which helps to add filters on each step of convolutional neural network. Method `glorot_uniform` helps draws values from a uniform distribution *W* with positive and negative bounds. We use cross entropy loss function, and

in these case Keras Convolution2d() and during training, the values in the filters are optimised with

$$W \sim \mathcal{U}\left(\frac{6}{n_{in}+n_{out}}, \frac{-6}{n_{in}+n_{out}}\right),$$

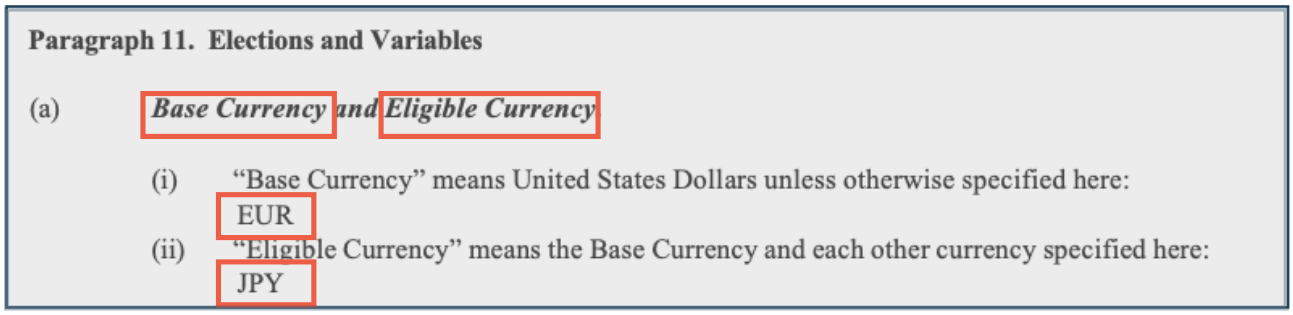
back propagation with respect to a loss function. [2]

Document dataset

In the final version of my work, I preferred to concentrate on document printed information nor on the handwritten type, because when both types of letters used, it is hard to make recognition of proper quality and it requires a more sophisticated dataset. In this case, my sample of documents consists of 3 ISDA MASTER agreement documents, which were prepared for neural network. The preparation includes several steps:

1. Import pdf document and extract required parameters by grid.

ISDA is a typical structured agreement, so we can use exact coordinates for required parameters:



Paragraph 11. Elections and Variables

(a) ***Base Currency*** ***and*** ***Eligible Currency***

(i) "Base Currency" means United States Dollars unless otherwise specified here:
EUR

(ii) "Eligible Currency" means the Base Currency and each other currency specified here:
JPY

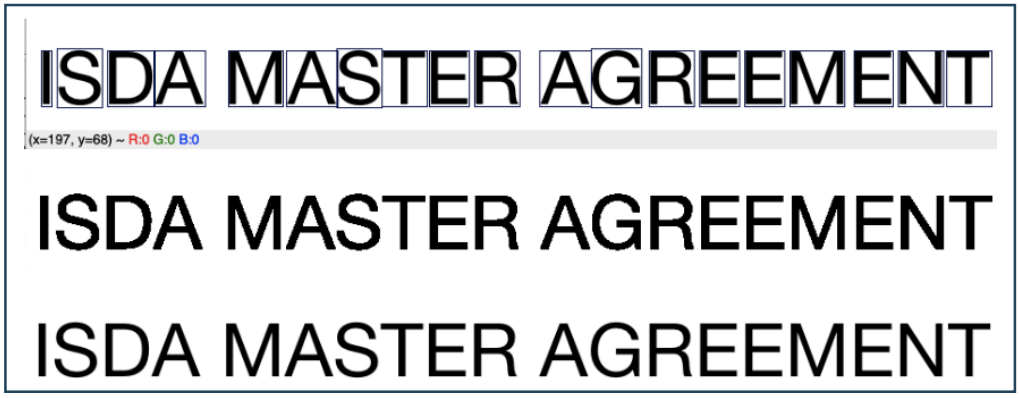
Picture 2 Part of the document sample with grid

2. Standardisation of the cropped fragments: make the image black and white, add a Gaussian outline using the library Open CV2.

3. Separate words to letters and capture counters by cv2.RETR_TREE

4. Scaling letters to 28*28 px (As in EMNIST)

As result, we get 22 images for each document for recognition.



Picture 3 Type text with good contours. The top picture – contours, the middle picture – erode sample, the bottom picture – original image.

Neural network training

In this work, there are three sub-experiments:

1. One-hour training network
2. 10 hours training network
3. Pytessaract neural network.

Keras uses a multilayer system of nonlinear filters to extract features with transformations. Each subsequent layer receives the output of the previous layer at the input. The attributes of hierarchically organised higher level attributes are derived from lower level attributes. I used the following parameters for this network. Kernel size is 3 - this is value specifying the length of the 1D convolution window. Also, there are no padding. As you can see, this is a classic convolutional network that highlights certain features of the image (the number of filters 32 and 64), to the "output" of which is connected the "linear" MLP network, which forms the final result. To read the database, we will use the idx2numpy library. We will prepare data for training and validation. These neural networks have equal parameters nor the estimation time. [4]

The control recognition is made by Pytessaract neural network library. This is a ready-made library for text recognition. All functions of machine learning and OCD are included in the functions of this library. It is built on the basis Google Tesseract Open Source OCR Engine and there is opportunity to include custom alphabet for dataset.

Result and discussion

There are results for all models:

A) One-hour training:(full version: result_1_hour.csv) (partial)

result_1_hour

date	name	Base Currency	Eligible Currency	Delivery Amount	Return Amount
03 05 2011	Smaug International	KZT	XAU	1750 000 00	875007000 00
03 05 2011	Iron Bank	EUR	JPY	500 000 00	5000W0 00
13 04 2010	Bank Gringotts	USD	CHF	2 500 000 00	175007000 00

B)10-hours training:(full version: result_10_hour.csv) (partial)

result_10_hours

date	name	Base Currency	Eligible Currency	Delivery Amount	Return Amount
03 05 20I1	Smaug International	KZT	XAU	1750 0U0 00	875U070U0 00
03 05 20I1	Iron Bank	EUR	JPY	500 0U0 00	5003W0 00
13 04 2010	Bank Gringotts	USD	CHF	2 500 0U0 00	175U070U0 00

C)Pytessaract training:(full version: result_pytessaract.csv)(partial)

result_pytessaract

date	name	Base Currency	Eligible Currency	Delivery Amount	Return Amount
03.05.2011	Smaug International	KZT	XAU	1,750,000.00	8,500,000.00
03.05.2011	Iron Bank	EUR	JPY	500,000.00	500,000.00
13.04.2010	Bank Gringotts	USD	CHF	2,500,000.00	1,500,000.00

As we can see, both models A and B are good enough in text recognition but there are some difficulties in recognition punctuation marks such as commas and dots. There is also misrecognition of zeros.

According to a personal assessment of these three results in comparison with the original documents, the accuracy of the estimate for Pytessaract is close to 97%, for model A this figure is 77% and for model B 70%. These values were defined as the ratio of incorrectly recognised characters to all recognised characters.

Based on these results, we can conclude that this model based on EMNIST does not yet have sufficient accuracy for serial data recognition.

As for models A and B, we can also notice that despite the longer training time, model B showed worse results compared to model A. In this situation, the phenomenon of overtraining may take place. Overtraining is an undesirable factor that occurs during solving learning from examples when the probability of errors in the training algorithm at the facility is higher than the average error in the learning sample. Overtraining occurs when using overly complex models.[3]

Conclusion

This model is an extremely interesting educational project for text recognition. As an improvement of the existing algorithm, it is supposed to change the dataset for teaching from handwritten to printed, including various fonts (Times New Roman, Arial etc.), as well as punctuation marks. We can also use a larger database of contracts as a training dataset to get rid of the parameter grid (all the same, contracts sometimes differ) and replace it with recognition of the key parameter + number. (example: if was found: «Eligible currency» - save «number»).

References

[1] Gregory Cohen, Saeed Afshar, Jonathan Tapson, André van Schaik (2017) *EMNIST: an extension of MNIST to handwritten letters //* 2017 International Joint Conference on Neural Networks (IJCNN)
[2] https://jacobgil.github.io/deeplearning/filter-visualizations

[3] Everitt B.S., Skrondal A. (2010), *Cambridge Dictionary of Statistics*, Cambridge University Press.

[4] habr.ru