Machine Learning Engineer Nanodegree

Capstone Proposal

Jakob Schmollgruber
2018-04-01

Proposal

Domain Background

A few years ago I read an article about hand written digit recognition. I was amazed. It was the first time I have heard about Machine Learning. Now some years later I'd like to train my own Machine Learning algorithm to classify hand written mathematical symbols (HASYv2 dataset), especially because I'm a mathematics student. Handwriting recognition (HWR) belongs to the domain of image processing. One of the first researchers who was active in this field was Sheloa Guberman 1962. The business application capabilities of this topic are incredible. The best results in this domain have been achieved on the MNIST database. It is probably the most famous dataset and therefore the most influential data set in this research area. In this capstone project we won't use this dataset but it's essential to mention it. Furthermore, I suggest to the interested reader to look at Street View Imagery.

Problem Statement

Handwriting recognition is a classical classification problem. In our case we have a photo with a mathematical symbol and want to know which it is. In this capstone project will have to handle 369 classes (symbols). With the aid of 168233 labeled training data we will use a supervised learning approach to solve this problem. I will use several learners like the

- Deep Neural Networks
- Convolutional Neural Networks
- Support Vector Machines (maybe)

and compare the prediction accuracy. We are faced with challenges because

- Number of training data per class varies

- Mathematical symbols look complex and sometimes similar
- Large amount of classes.

Datasets and Inputs

For this project we will use data from the free available HASYv2 dataset [1]. It contains 168233 labeled images of 369 handwritten mathematical symbols. The images have a 32 × 32 px resolution. So there are $32 \times 32 = 1024$ features in $[0, 255]$. We will use One Hot encoded labels to train the learners. It's important to remark that the number of training data per class varies as mentioned above. In addition, the HASYv2 dataset is designated to be trained by a 10-fold cross validation.

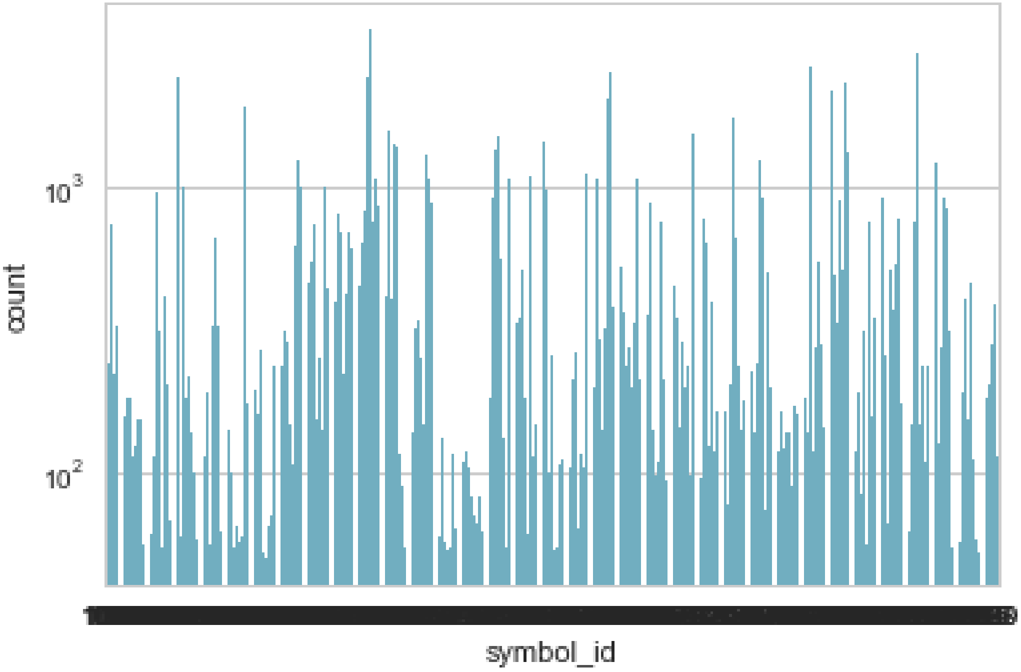

Abbildung 1: Here we can see the frequency of training data per class

Solution Statement

As already mentioned we will use supervised training algorithms to classify the symbols. We will use labeled training data to fit our learners. To be able to guarantee replicability we will train the algorithms with set RandomSates.

- We will train the Deep Neural Networks (DNN) with some $(1, 2, 3, 4, 5)$ hidden, fully connected layers. The DNN will have 1024 input neurons and 369 output neurons. We will use the sigmoid as activation function.
- Certainly the CNN will have 1024 input neurons and 369 output neurons too. But the architecture of the hidden layer differs. We will use some sequence of convolutional and pooling layers.
- Will use the Grid Search to find hyperparameter for the SVM.

Towards the training process recognition proceeds fast. One has to put a picture into the classifier. Each of the 369 output neurons will return a value in [0,1]. We will take the neuron with the highest output value and choose the symbol corresponding to the neuron.

Benchmark Model

In the capstone project we will choose two different benchmark approaches . On the one hand we will compare the presented learners among themselves. On the other hand I will compare my results with domain related results[1].

Evaluation Metrics

In this project we will use the accuracy as evaluation metric. Let y be the true label and $\tilde{y}$ the predicted value. Then accuracy is defined as follows

$$\text{accuracy}(y, \tilde{y}) = \frac{1}{n} \sum_{n=0}^{n-1} 1_{\tilde{y}=y}$$

A classifier always predicting the truth (existence condition) leads to an accuracy of 1 on any test set. A classifier always being wrong will deliver a accuracy of 0 on any test set. In addition, we will investigate if some symbols are interchanged by the classifier (Confusion Matrix) a lot.

Project Design

- **Programming**: Python 3
- **Libraries**: pandas, sklearn, keras, ggplot, numpy, pyplot
- **Data**: HASYv2 dataset

- **Sequence of Work**:
  - Import and preprocess data
  - Get an overview
  - Generate dummy variables
  - Split Data (k-fold cross-validation)
  - Define Neural Network Architecture
  - Train the DNN and CNN
  - Validate
  - Test
  - Generate visualization, diagrams, Confusion Matrix
  - Clean code

Let’ s describe the data structure in detail. Here we can see the head of the csv file in detail.

```
   path                          symbol_id    latex    user_id
0  hasy-data/v2-00000.png        31            A       50
1  hasy-data/v2-00001.png        31            A       10
2  hasy-data/v2-00002.png        31            A       43
3  hasy-data/v2-00003.png        31            A       43
4  hasy-data/v2-00004.png        31            A       4435
```

Each row of the csv represents a hand drawn mathematical symbol. The path feature shows where the corresponding picture is saved. In the words of relational algebra the features symbol_id and latex are functional dependent. Both label the symbols. The feature user_id represents the creator of the symbol - mentioned not to be reliable [1].

Literatur

[1] Thoma Martin, The HASYv2 dataset, arXiv:1701.08380v1, 2017

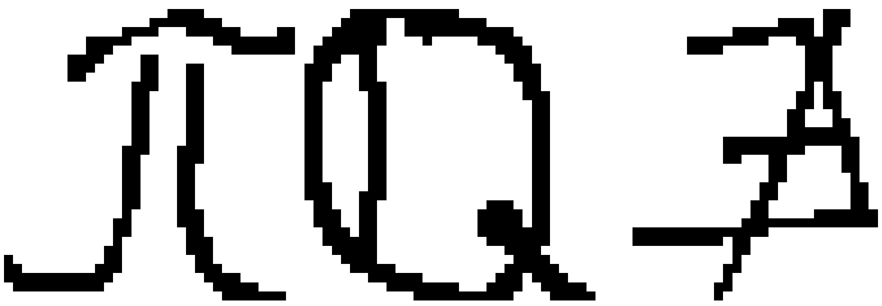

Abbildung 2: Mathematical Symbols from HASYv2