Project Report
Machine Learning

Kaggle

Google Landmark Recognition Challenge

Marcel Rohde

Thorben Wiese

Contents

List of Figures

1 Introduction

Kaggle is a platform owned by the Kaggle Inc. which is owned by Alphabet Inc. (Google) providing multiple so called challenges in the field of data science, predictive modeling and data analysis. The Kaggle challenge aims at solving so far unsolved tasks or finding a better solution for already solved tasks in a crowdsourcing fashion. Some challenges can be solved for monetary prices, others are hosted for knowledge or training purposes. In order to solve a challenge one must register and submit a solution to the platform to get a score for the submitted solution [4].

The Google Landmark Recognition Challenge aims at detecting different landmarks in images, such as the Eiffel Tower or the Leaning Tower of Pisa [1].

We chose the Google Landmark Recognition Challenge because we are interested in image processing tasks and the challenge provides a lot of training data. We also wanted to try out Convolutional (Deep) Neural Networks which we previously discussed in the lecture.

The data provided with the challenge contains mainly of two CSV files. The file for training (train.csv) providing the IDs, URLs and Landmark IDs and the file for testing (test.csv) providing IDs and URLs [2].

Our goal for this part of the project is to prepare the image data and to use different classification methods in order to get good results in detecting landmarks in images.

The evaluation is described in [3]. We need to predict one landmark per image and a corresponding confidence score for it. The overall score will be calculated with the Global Average Precision (GAP) score

$$GAP = \frac{1}{M} \sum_{i=1}^{N} P(i) rel(i)$$

where:

- N is the total number of predictions returned by the system, across all queries

- M is the total number of queries with at least one landmark from the training set visible in it

- $P(i)$ is the precision at rank i

- $rel(i)$ denotes the relevance of prediction i: it's 1 if the i-th prediction is correct, and 0 otherwise

2 Predictive Modeling Steps

Image to feature vector (feature extraction)

Normalization

3 Training and Test Data

We chose to create a subset of the original training data for computation time and performance reasons. The original training data has about 1.2 million images which takes too long to process for a one week assignment. We therefore selected the seven most relevant Landmark IDs according to our PCA analysis (IDs: 2061, 6051, 6599, 9633, 9779, 60, 51) and chose 2000 images containing those IDs for our training data. The script for downloading this data is attached to this report (download-images.py).

The test data will be chosen accordingly with a size of approximately 1000 images.

4 Methods

Our task in this challenge is to predict whether an image contains a specific landmark or none at all. We had several approaches for such a classification task in the lecture and we chose to use three methods that we discussed previously in class.

4.1 KNN

KNN stands for k-Nearest-Neighbor and describes a typical algorithm that can be used for classification and regression. For classification, the algorithm assigns a data point to a class according to the majority of its k neighbors where k is a pre-defined constant.

4.2 SVM

One very well known method for classification is the Support Vector Machine (SVM) which can be used for a variety of different classification and regression tasks. It usually has a linear output, however, using the so called Kernel Trick allows it to map its inputs into higher dimensional space resulting in non-linear classification of data points.

4.3 Logistic Regression

The Logistic Regression is a special form of regression using the logistic loss as the loss function explained in [5]

$$\ell(X, f(X), Y) = \log_2(1 + exp(-Yf(X))).$$

5 Results

Plots

5 Results

6 Method Comparison

The results shown in chapter 5 show that ...

7 Conclusion

We introduced three well-known methods for image classification in this project and compared their results for solving the Google Landmark Recognition Challenge.

The results are ...

However, a deep convolutional neural network with a much longer training time is probably the best choice for approaching a task like this.

Bibliography

[1] INC., Kaggle: *Google Landmark Recognition Challenge.* `https://www.kaggle.com/c/landmark-recognition-challenge.` Version: June 2018

[2] INC., Kaggle: *Google Landmark Recognition Challenge Data.* `https://www.kaggle.com/c/landmark-recognition-challenge/data.` Version: June 2018

[3] INC., Kaggle: *Google Landmark Recognition Challenge Evaluation.* `https://www.kaggle.com/c/landmark-recognition-challenge#evaluation.` Version: July 2018

[4] INC., Kaggle: *Kaggle.* `https://www.kaggle.com.` Version: June 2018

[5] LUXBURG, Ulrike von: *Lecture Slides Machine Learning*