TU Chemnitz

Faculty of Natural Sciences
Institute of Physics

Bachelor Thesis

In the course of degree in computational science (B.Sc.)

To obtain the academic degree Bachelor of Science

Topic:	Investigation of strategies for image classification on small training data sets
Author:	Björn Hempel <bjoern@hempel.li> Matriculation number 025038
Version from:	February 11, 2020
First assessor:	Prof. Dr. Angela Thränhardt
Second assessor:	Dr. David Urbansky

Abstract

Neural networks are extraordinarily good at finding patterns in data. For this purpose, these networks must be trained with known data sets and adapted accordingly. Data sets are usually very expensive to obtain and should therefore be used with care and good quality. The training of the network takes place under many different parameters and process techniques. Care must be taken to use the best possible model with its best possible parameters. In this thesis, common modern methods of image classification will be presented and compared with each other. The main goal of the work is to find optimal parameters and techniques for the classification, which also allow to create an optimal model with little training data.

Contents

1 Introduction

In this thesis different techniques of image classification are compared. Variable parameters during training will have a decisive influence on the accuracy of the model and are compared here in detail. Not always only the accuracy is a decisive factor. Also the required computing time, which is necessary to determine the model, should not be disregarded and should be included in the evaluation. I assume that a small learning rate combined with many learning epochs and correspondingly more computing time required will achieve better results than a few learning epochs combined with a high learning rate (slow adaptation vs. fast adaptation). I also assume that a high quality and a larger amount of data will have a decidedly positive influence on the result. New and more complex convolutional neural networks are more successful in model accuracy than older and smaller models.

2 Background

2.1 Image Classification

Classifications are a process of identifying to which class an unobserved object belongs. A number of predefined classes can be specified and, based on their properties, an attempt can be made to classify unknown and previously unobserved objects. The procedure for image classification is similar. The previously mentioned objects are now simply images.

Figure 1: Is it a dog or a cat?[1]

For a long time, the automatic recognition of objects, people and scenes in images by computers was considered impossible. The complexity seemed too great to be programmatically taught to an algorithm. Until a few decades ago, attempts were made to achieve image classification by manually developed algorithms. Automated classification based on given and pre-classified images and the automated creation of models was a new step into a new approach. The neural networks developed in this process played a huge role and dramatically changed the way of approach! In the meantime, image recognition has become a widespread application area of machine learning. So-called "Convolutional Neural Networks[2]" or "ConvNets" are often used for images.

The image classification algorithm takes an image as input and classifies it into one of the output categories. Deep Learning has revolutionized the field of image classification and has achieved great results. Various Deep Learning networks, such as ResNet, DenseNet, Inception, etc. have been developed as high-precision networks for image classification. At the same time, image data sets were created to capture tagged image data. These are now primarily used to train existing networks and to organize annual challenges that compete with the model accuracies already known and developed. ImageNet is such a large data set with more than 11 million images and over 11,000 categories. Once a network has been trained with ImageNet data, it can be generalized with other data sets by simple re-compilation or optimization. In this transfer learning approach, a network is initialized with weights that come from a previously trained network. This previously initialized network is now simply adapted for a new image classification task.

[1] Source: `https://towardsdatascience.com/image-classifier-cats-vs-dogs-with-convolutional-neural-networks-cnns-and-google-colabs-4e9af21ae7a8`

[2] Convolutional neural network, Wikipedia contributors, January 31, 2020, `https://en.wikipedia.org/wiki/Convolutional_neural_network`

The underlying work here is mainly concerned with supervised learning, in which a mathematical model is trained based on existing known data sets. The goal of the trained model is to make best possible predictions even for unknown images. These known data sets are usually created manually (ontologist), automatically determined based on known facts or determined in a semi-automatic process.

2.1.1 Deductive approach

Since the late 1960s, attempts have been made to classify images with self-written algorithms. This part of Computer Vision deals with techniques such as image creation, image processing and image segmentation. In the field of image processing, well-known techniques such as edge detection, feature detectors, edge linking, contrast enhancement, etc. are used[3]. Common to all techniques is the use of the deductive approach. With the deductive approach, one creates rules (feature detectors) which are supposed to predict the desired result. These rules are given and described and thus allow later classification of unknown objects. Since the model and its algorithm are sufficiently well known, this procedure is called white-box procedure.

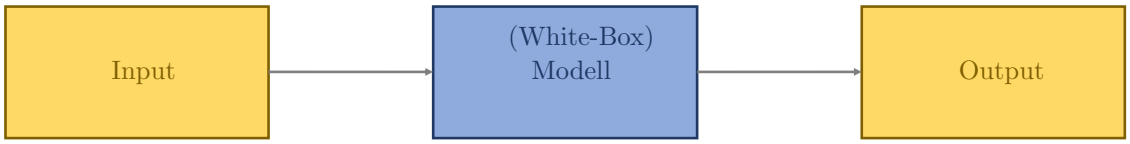

Figure 2: Deductive approach

2.1.2 Inductive approach

The inductive approach, on the other hand, takes a different approach to classifying images. The goal is not to specify a rule, but to learn a rule (model) automatically from already known individual objects. A model is usually a complex function and a mathematical representation of a space (VC dimension[4]), in which individual objects with their properties can be mapped and separated. The model is adapted piece by piece to the known objects in such a way that the input value corresponds to the output value or corresponds to a large extent (backpropagation). The goal is to create a function with this model, which is able to classify unknown objects in the best possible way. Because the space of this model is mostly far away from the imagination and the possibility of explanation, this procedure is also called black box procedure. The procedure described here is mostly used for any kind of supervised learning and is a part of machine learning.

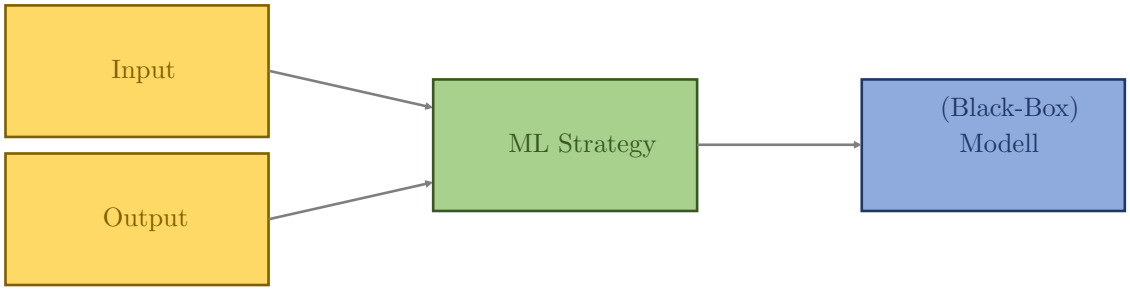

Figure 3: Inductive approach

[3]Szeliski, R.: Computer Vision: Algorithms and Applications, Springer Science Business Media, 10 (2010)

[4]Vapnik–Chervonenkis dimension, Wikipedia contributors, January 31, 2020, `https://en.wikipedia.org/wiki/Vapnik%E2%80%93Chervonenkis_dimension`

2.1.3 Balanced training data set

Neural networks have made enormous progress in the field of pattern recognition in recent years. A decisive factor is that the data for learning must be of high quality and easy for the network to process. Wrongly classified or irrelevant data could cause the network to learn something wrong. This also applies to non-existent or unsuitable pre-processing.[5]

With the beginning of a classification project the question is what exactly you want to classify and how extensive the classification should be. Assuming you want to identify different classes of food, this could be classes like pizza, burgers, donuts and lasagna (etc.). For these classes you now need a large number of images. Ideally, this data should reflect reality as well as possible. A large variation is advantageous (balanced data set): different viewing angles, size, position, colour brightness, variations, number, etc. Images of e.g. only one colour brightness or only one viewing angle should be avoided. If the data are not balanced, they must be corrected accordingly: e.g. by adding further data, image processing or by removing data that causes an imbalance. Furthermore, the selected classes should be clearly optically separable from each other. If two classes are visually very similar and not really distinguishable even by a human, consideration should be given to combining them (e.g. "burger" and "veggie burger"):

Figure 4: Example pictures of a burger class

Figure 5: Example pictures of a donut class

Figure 6: Example pictures of a pizza class

Accessing data is often not that easy. Every data source has its own special features. One way to access data would be an automatic crawling of image databases, search engines or reviews in which images appear. A certain amount of creativity is advantageous:

- Google
- Bing
- Flickr
- TripAdvisor
- etc.

[5] Douwe Osinga. *Deep Learning Kochbuch: Praxisrezepte für einen schnellen Einstieg.* O'Reilly Verlag, 2019, pp. 19–26. ISBN: 9783960090977.

Probably the most expensive way to obtain data is to search and classify them manually, e.g. by an ontologist. The ontologist evaluates and searches for different images and manually classifies them in the appropriate classes. A combined variant is also possible and probably preferable: automatic crawling and manual sorting out of incorrect, unfavorable or irrelevant images.

2.1.4 Training, test and evaluation data set

Before starting the training of balanced images, they must be divided into a training, a test and possibly a validation data set. This is necessary because neural networks will not generalize to some extent, but will learn by heart (overfitting[6]). The idea is to train with a training data set, while the validation data set is used to monitor the general validity of the network and its parameters. Based on the results, adjustments are made at runtime. Since the adjustment of the parameters is carried out using the test data, there is also an independent test data set, which carries out a renewed check of the model for previously uninvolved data. This ensures that hyperparameters are not inadvertently optimized for the validation data set only.[7] The use of the test data set is optional and simulates the model under real conditions. If the number of data is limited, this data record can also be added to the training data record, for example. In this thesis, the test data set is not used and all evaluations refer to the validation data set.

An optimal division of the training and validation data set depends on the existing classification problem and the amount of data available. In this paper a ratio of 80 percent training data and 20 percent validation data is used, unless otherwise stated.

The question of why 80 percent training data and 20 percent validation data are used remains to be clarified. Is there a paper or a study on this? Or is another test required?

2.1.5 Methods of machine learning

Different machine learning systems can be classified according to the type and procedure of monitoring the training. A distinction is made between the type of data we have or the data we need to determine ourselves.

2.1.5.1 Supervised learning

Supervised learning refers to machine learning with known training data sets (see also chapter "inductive approach"). The learning process in turn refers to the ability of an artificial intelligence to reproduce regularities and patterns. The results are known by laws of nature or expert knowledge and are used to teach the system by creating a training set containing the desired solutions. This is also called labelled data. The learning algorithm now tries to find a hypothesis epoch by epoch, which allows the most accurate predictions on unknown data. A hypothesis in this case is an image that assigns the assumed output value (the predicted class) to each input value (the image itself). This work makes extensive use of supervised learning.

In supervised learning, an input vector is fed to a classification function (usually an artificial neural network). The input vector generates an output vector using the classification function, which produces this neural network in its current state[8]. This value is compared with the value that it should actually output. The comparison of the nominal and actual state provides information on how and in what form changes must be made to the network in order to further approximate the actual state and minimize the error. For artificial neural networks without a hidden layer (single-layer perceptron[9]), the delta rule[10] for correction can be applied. For networks with one or more

[6] "Overfitting", Wikipedia contributors, January 31, 2020, `https://en.wikipedia.org/wiki/Overfitting`

[7] Osinga, *Deep Learning Kochbuch: Praxisrezepte für einen schnellen Einstieg*.

[8] The neural network consists of many (usually millions) parameters, which can be adjusted during the learning process to minimize the error.

[9] "Perceptron", Wikipedia contributors, February 2, 2020, `https://en.wikipedia.org/wiki/Perceptron`

[10] "Least mean squares" filter also known as "delta rule", Wikipedia contributors, February 2, 2020, `https://en.wikipedia.org/wiki/Least_mean_squares_filter`

hidden layers backpropagation[11] is used to minimize the error. Backpropagation is a generalization of the delta rule.

The neural network is only one algorithm from the category of supervised learning algorithms. For completeness here is a list of further algorithms:

- k-nearest neighbors[12]
- Linear regression[13]
- Logistic regression[14]
- Support-vector machine[15]
- Random forest[16]
- etc.

2.1.5.2 Unsupervised learning

In unsupervised learning, one tries to gain knowledge of patterns even without labelled data. Suppose you have several pictures of burgers, pizza and donuts, which are unsorted in a data set. Unsupervised learning now tries to find similarities in order to cluster these images. In the best case you get three unnamed groups A, B and C at the end. Analysts will take a closer look at these groups afterwards and draw a conclusion if possible: Group A is burgers, group B is pizzas, etc.

The following unsupervised learning algorithms can be used for clustering:

- k-means[17]
- Hierarchical clustering[18]
- Expectation–maximization [19]
- etc.

One technique, hierarchical clustering, is used later to facilitate the introduction of hierarchies. For the general analysis, the finding of optimal parameters for learning models, this kind of learning is not used in this thesis.

2.1.5.3 Reinforcement learning

Reinforcement learning[20] is a type of machine learning in which an agent independently learns the best possible strategy for achieving a goal. To achieve the goal, actions are necessary which produce rewards at certain points in time. These rewards can also be negative (punishment). Based on these rewards, the aim is to achieve the best possible reward value. This type of learning is not relevant for the classification of images, which is why it will not be discussed further here.

[11] "Backpropagation", Wikipedia contributors, February 2, 2020, `https://en.wikipedia.org/wiki/Backpropagation`

[12] "k-nearest neighbors algorithm", Wikipedia contributors, February 2, 2020, `https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm`

[13] "Linear regression", Wikipedia contributors, February 2, 2020, `https://en.wikipedia.org/wiki/Linear_regression`

[14] "Logistic regression", Wikipedia contributors, February 2, 2020, `https://en.wikipedia.org/wiki/Logistic_regression`

[15] "Support-vector machine", Wikipedia contributors, February 2, 2020, `https://en.wikipedia.org/wiki/Support-vector_machine`

[16] "Random forest", Wikipedia contributors, February 2, 2020, `https://en.wikipedia.org/wiki/Random_forest`

[17] "k-means clustering", Wikipedia contributors, February 2, 2020, `https://en.wikipedia.org/wiki/K-means_clustering`

[18] "Hierarchical clustering", Wikipedia contributors, February 2, 2020, `https://en.wikipedia.org/wiki/Hierarchical_clustering`

[19] "Expectation–maximization algorithm", Wikipedia contributors, February 2, 2020, `Expectation\OT1\textendashmaximizationalgorithm`

[20] "Reinforcement learning", Wikipedia contributors, January 31, 2020, `https://en.wikipedia.org/wiki/Reinforcement_learning`

2.1.6 Classification Metrics and confusion matrix

Choosing the right metric is crucial in evaluating machine learning models. Metrics are used to monitor and measure the performance of a model during training and testing. Some important metrics are explained below.

2.1.6.1 Confusion Matrix

The Confusion Matrix is a special quadratic matrix in the field of machine learning that allows the visualization of the performance of a predictive model. Each row of the matrix represents the actual class, while each column indicates the number or a numerical value as a percentage of the predicted class (or vice versa)[21]:

		predicted			
		$\boldsymbol{class_1}$	$\boldsymbol{class_2}$	$\boldsymbol{\ldots}$	$\boldsymbol{class_n}$
actual	$\boldsymbol{class_1}$	TP	FN		
	$\boldsymbol{class_2}$	FP	TN		
	$\boldsymbol{\ldots}$				
	$\boldsymbol{class_n}$				

Table 1: Confusion matrix

Finally, the Confusion Matrix has the following structure, where the number of elements in the class $C_{i,P}$ was predicted although ($\cong$) it should have been class $C_{j,A}$:

$$\boldsymbol{M}_{confusion} = \begin{bmatrix} \#C_{1,P} \cong C_{1,A} & \ldots & \#(C_{n,P} \cong C_{1,A}) \\ \vdots & \ddots & \vdots \\ \#(C_{1,P} \cong C_{n,A}) & \ldots & \#(C_{n,P} \cong C_{n,A}) \end{bmatrix} = (a_{nn}) \tag{1}$$

2.1.6.2 Accuracy

Top-1 accuracy is probably the most important accuracy. It tells you the percentage of the model's best prediction of the data in the validation set that matches the expected class.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} = \frac{\sum_{i,j=1}^{n} a_{ij}}{\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}} = \frac{Correct_{all}}{CorrectPossible_{all}} \tag{2}$$

The **Top 5 Accuracy** is another accuracy specification. However, not only the best hit is included here, but also the next four. As soon as the correct class can be found within the first five predicted classes, this prediction is also true:

$$Accuracy_{top-5} = \frac{CorrectWithinTheBest5Classes_{all}}{CorrectPossible_{all}} \tag{3}$$

The accuracy of the entire model is a good indication of the performance of the model. However, a problem occurs in extreme cases where assumptions can no longer be made reliably. For example, if you are working with an unbalanced dataset.[22] Example: Suppose we have a model that always

[21] "Confusion matrix", Wikipedia contributors, February 5, 2020, `https://en.wikipedia.org/wiki/Confusion_matrix`

[22] Aurélien Géron. "Praxiseinstieg Machine Learning mit Scikit-Learn und TensorFlow: Konzepte, Tools und Techniken für intelligente Systeme". In: O'Reilly Verlag, 2017. Chap. Konfusionsmatrix, pp. 86–88. ISBN: 9783960090618.

predicts the class $class_1$. The class $class_1$ consists of 9990 elements and of the other classes $class_2$ to $class_n$ we have exactly 10. Then the Confusion Matrix looks like this:

		predicted			
		$\boldsymbol{class_1}$	$\boldsymbol{class_2}$	$\boldsymbol{\ldots}$	$\boldsymbol{class_n}$
actual	$\boldsymbol{class_1}$	$TP = 9990$	$FN = 0$		
	$\boldsymbol{class_2}$	$FP = 10$	$TN = 0$		
	$\boldsymbol{\ldots}$				
	$\boldsymbol{class_n}$				

Table 2: Confusion matrix example

The model accuracy in this case is 99.9%, although it is a bad model:

$$Accuracy = 99,9\% \tag{4}$$

Therefore there are additional performance metrics like Precision, Recall and F-Measure.

2.1.6.3 Precision

Precision[23] expresses how reliable the statement of a prediction of a class is:

$$Precision = \frac{Correct}{Actual} = \frac{TP}{TP + FP} \tag{5}$$

Or more precisely for class c:

$$Precision_{@c} = \frac{a_{cc}}{\sum_{i=1}^{n} a_{ic}} \tag{6}$$

2.1.6.4 Recall

Recall[24] is the accuracy of a class. This means how well the class could be predicted:

$$Recall = \frac{Correct}{CorrectPossible} = \frac{TP}{TP + FN} \tag{7}$$

Or more precisely for class c:

$$Recall_{@c} = \frac{a_{cc}}{\sum_{i=1}^{n} a_{ci}} \tag{8}$$

2.1.6.5 F-Measure

F-Measure[25] combines precision and recall, with the parameter β representing the weighting:

$$F_\beta = (1 + \beta^2) \cdot \frac{Precision \cdot Recall}{\beta^2 \cdot Precision + Recall} = \frac{(1 + \beta^2) \cdot TP}{(1 + \beta^2) \cdot TP + \beta^2 \cdot FN + FP} \tag{9}$$

[23] "Precision and recall", Wikipedia contributors, February 7, 2020, `https://en.wikipedia.org/wiki/Precision_and_recall`

[24] "Precision and recall", Wikipedia contributors, February 7, 2020, `https://en.wikipedia.org/wiki/Precision_and_recall`

[25] "F1 score", Wikipedia contributors, February 7, 2020, `https://en.wikipedia.org/wiki/F1_score`

The higher β the more importance is placed on precision instead of recall. This is important if you put more importance on the quality of the prediction than on the accuracy of prediction. For example, when predicting diseases: assigning class "ill" to healthy people is just as fatal as assigning class "healthy" to sick people (although case two would be even more fatal than case one). With a beta value of $\beta = 0,5$ we get an equal distribution of both values and is called F1 score:

$$F_1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} = \frac{2 \cdot TP}{2 \cdot TP + FN + FP} \tag{10}$$

2.1.6.6 Loss function

Since this is a classification problem and not a regression problem, the prediction is based on softmax regression. After each prediction, a vector λ of the size n is returned, where n corresponds to the number of classes to be distinguished.[26] Each individual $\hat{p}$ value corresponds to the probability that it is class $class_n$:

$$\lambda = \begin{pmatrix} \hat{p}_1 \\ \hat{p}_2 \\ \vdots \\ \hat{p}_n \end{pmatrix} \quad \Big| \quad \sum_{i=1}^{n} \hat{p_i} = 1 \tag{11}$$

The expected value of the parameter function and thus of the current class is returned as a one hot vector. The value 1 corresponds to the expected class. All other classes return 0. This is also called one hot encoding:

$$g(\vartheta_2) = \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix} \tag{12}$$

The loss function[27] assigns a loss to each prediction, which results from the comparison with the true value or parameter. For this purpose, the distance between the predicted class and the true class is calculated (if they are equal, the distance is 0). If the adaptation of the model (learning process) improves the prediction of all predicted classes to their true classes, the value of the loss function is also reduced. A typical loss function is e.g. for an r-dimensional space:

$$L_r(\vartheta, \lambda) := \|\lambda - g(\vartheta)\|^r \tag{13}$$

λ represents the estimated value and $g(\vartheta)$ the parameter function which returns the actual value for ϑ. The average loss on the entire data set with n elements is thus:

$$\hat{L} = \frac{1}{k} \sum_{i=1}^{k} L_r(\vartheta_i, \lambda_i) \tag{14}$$

2.2 Machine Learning

Machine learning is a generic term for the artificial generation of knowledge from experience. It follows the approach of inductive learning (see also chapter "inductive approach").

[26] Aurélien Géron. "Praxiseinstieg Machine Learning mit Scikit-Learn und TensorFlow: Konzepte, Tools und Techniken für intelligente Systeme". In: O'Reilly Verlag, 2017. Chap. Entscheidungsgrenzen, pp. 138–140. ISBN: 9783960090618.

[27] "Verlustfunktion (Statistik)", Wikipedia contributors, February 7, 2020, `https://de.wikipedia.org/wiki/Verlustfunktion_(Statistik)`

2.2.1 Artificial neural network

Artificial neural networks provide functions that are able to separate highly complex data in multidimensional space. For large and highly complex tasks, such as the classification of billions of images, speech and text recognition, neural networks usually perform better than other machine learning methods. The significant increase in computational capacity since the 1990s allows the training of large neural networks within a reasonable period of time. Artificial neural networks are the core component of deep learning.

Neural networks process an input vector $\bar{x}$ and convert it into a new output vector $\hat{\bar{x}}$. They are networks of many artificial neurons connected in series and parallel. An artificial neuron in turn converts a vector into a scalar by scaling and summing the inputs $\bar{x}$ with the changeable parameters $\bar{\omega}$ and correcting them with a bias b (the bias is also a changeable variable). The activation function ensures that the first degree polynomial (linear regression model) becomes a nonlinear function[28]:

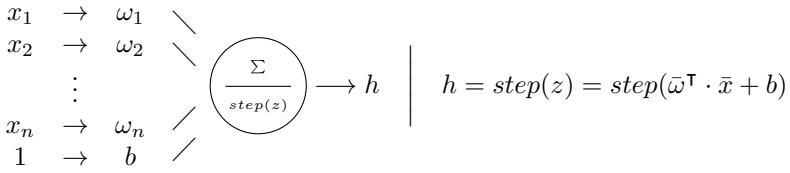

Figure 7: The construction of an artificial neuron.

The artificial neural network is composed of many layers connected in series, which again contain neurons connected in parallel:

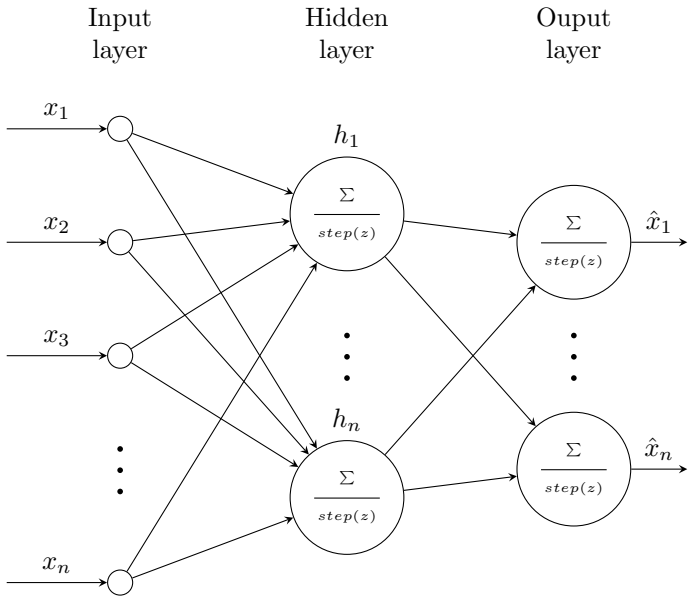

Figure 8: The construction of a simple neural network.

A neural network is able to classify complex inputs. But how exactly can this be imagined? Let us consider a simple classification function, where $\bar{x}$ represent the coordinates of the individual class points and $\bar{w}$ and b are learnable parameters:

$$f(\bar{x}, \bar{\omega}, b) = sgn(\bar{\omega}^\intercal \cdot \bar{x} + b) \tag{15}$$

With this linear function the following problem can be easily classified:

[28] "Activation functions, their types and uses", https://www.ai-united.de/, February 8, 2020, `https://www.ai-united.de/aktivierungsfunktionen-ihre-arten-und-verwendungsmoeglichkeiten/`

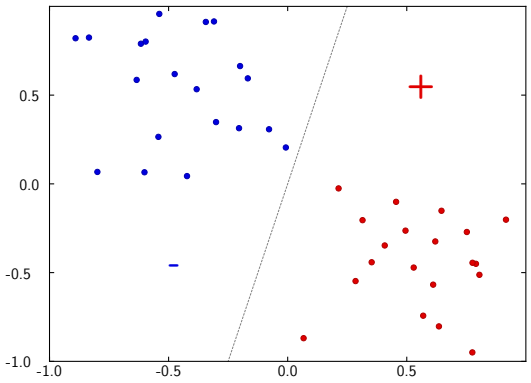

Figure 9: Simple class shattering

The function corresponds to a neural network without a hidden layer and contains only one input and one output layer with one artificial neuron without activation function. The dimension that this function can separate is 2 and is called VC dimension[29]. This function can separate exactly 2 classes. But what about nonlinear problems? In this case let's look at the following classifications:

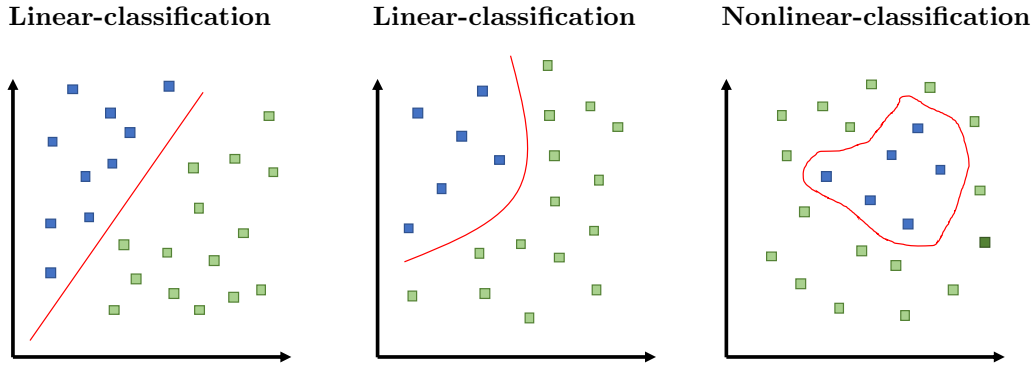

Figure 10: Linear vs. nonlinear classification

For the second problem we can still adjust the function. For the third nonlinear problem the classification space is no longer sufficient and requires a different algorithm. And this is where the neural networks come into play. A tool to visualize the separation of data and to test the functionality of the individual layers is `https://playground.tensorflow.org`[30]. In the simplest case, a nonlinear problem can be solved by adding a hidden layer with three additional neurons:

[29] Vapnik–Chervonenkis dimension: `https://en.wikipedia.org/wiki/Convolutional_neural_network`
[30] Neural Network Right Here in Your Browser: `https://playground.tensorflow.org`

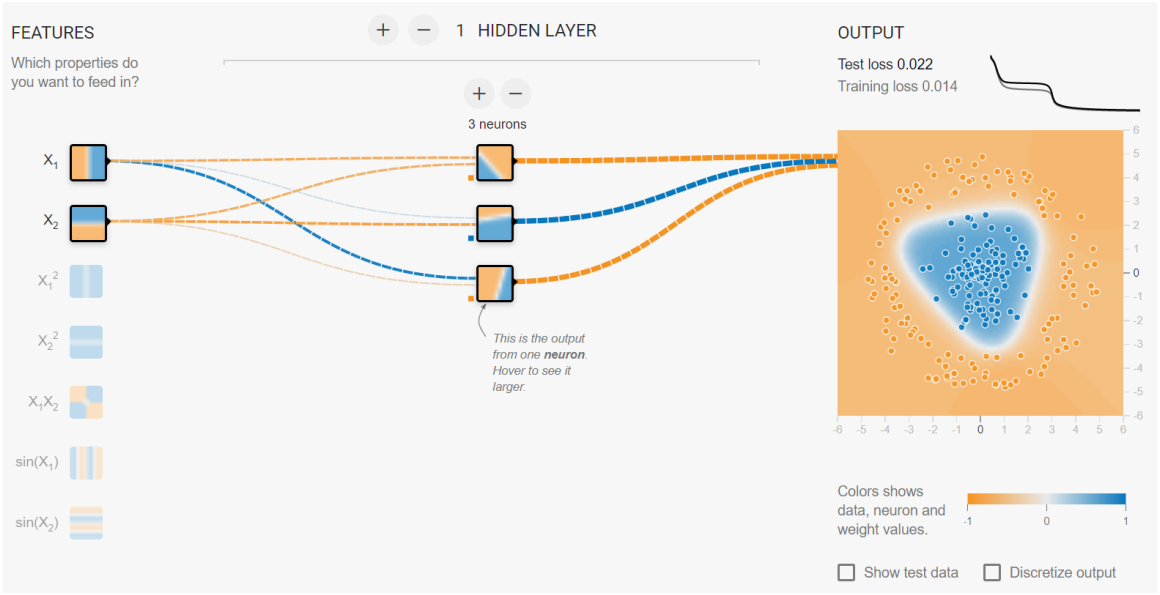

Figure 11: Simple neuronal network with one hidden layer[31]

2.2.2 Convolutional Neuronal Network

A neural network processes a vector and returns a new vector. The problem with input data such as images is that at first view they cannot be successfully described as vectors to be trained with a normal neural network. You need an algorithm that can handle matrix-like inputs and that is able to recognize patterns. In the past the principle of the convolutional layer was developed. A convolutional layer receives a matrix input, transforms it and returns an output value (in this case another matrix). This output value is then passed on to the next layer. A convolutional layer contains a set n of square matrices (usually 3x3 or 5x5 matrices). These matrices are called filters. Each filter[32] is now calculated from top left to bottom right over the pixels of the image using a scalar product, which creates a new image. The so-called Feature Map. With a number of n filters, n feature maps are created at the end and highlight the features defined in the filters in the newly calculated image. This process is also called convolution.[33]

Neural networks that make use of convolutional layers are called convolutional neural networks (in short CNN) and have made a decisive contribution to the progress of image classification and also in other areas like speech recognition. In addition to the Convolutional Layers, a Convolutional Neural Network has other special layers that differ from normal neural networks: For example the Pooling Layer. In a pooling layer, unnecessary information is discarded and feature maps are reduced in size. This process reduces memory requirements and increases calculation speed. The convolutional layer and the pooling layer usually alternate until a large vector is created at the end instead of a nxn matrix of the input image, which can be further processed by a normal neural network and finally ends in the already described one hot vector.

Idea? `https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53`

[31] Source: `http://playground.tensorflow.org/`

[32] Filters, which can recognize edges, corners, squares, etc. and in deeper layers things like eyes, ears, hair, etc.

[33] deeplizard. *Convolutional Neural Networks (CNNs) explained*. Youtube. 2017. URL: `https://www.youtube.com/watch?v=YRhxdVk{\_}sIs`.

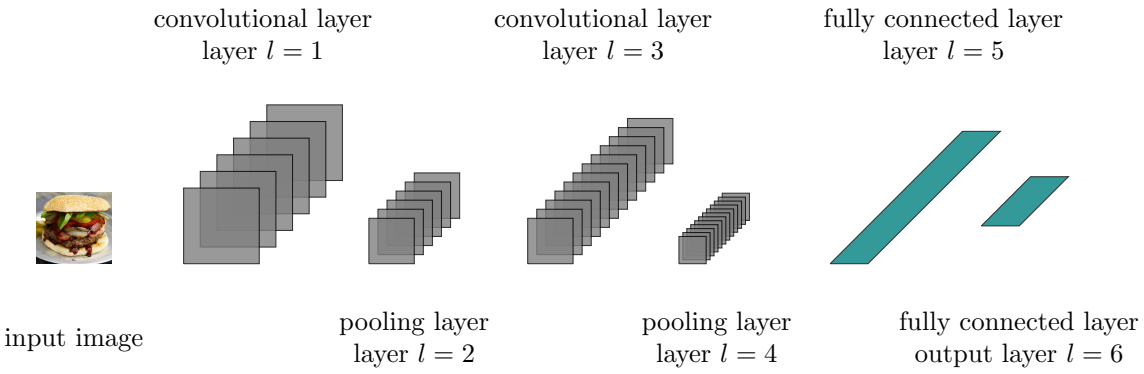

Figure 12: Architecture of a traditional convolutional neural network.

A big advantage of convolutional neural networks should not remain unmentioned: They require relatively little preprocessing compared to other image classification algorithms. This means that the network independently learns the filters that are normally developed by hand in conventional algorithms, if trained with adequate training. This property of these networks is a great advantage because they can be automated and change independently when the input data changes and do not require human intervention.

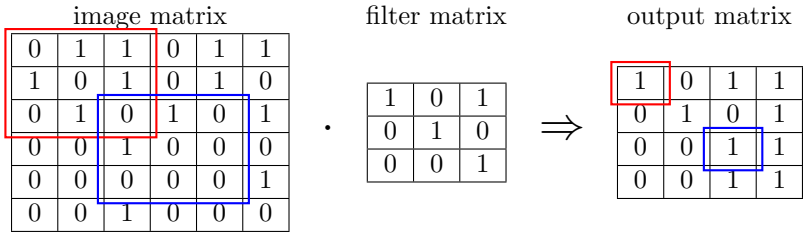

Figure 13: Simple convolution

2.2.3 Transfer Learning

Convolutional neural networks are great and have made a significant contribution to the classification of images. With the foundation of the research database ImageNet in 2006, annual competitions are organized to compare developed neural networks. ImageNet is an image database with more than 14 million images. A CNN called AlexNet in 2012 got a top-5 error of 15.3% and is currently increasing steadily every year. But the architecture of a CNN has a problem. All convolutional layers are randomly initialized from the beginning and do not yet contain any patterns. For it to work reliably, it needs to be trained with many images. If one would develop and use a CNN from scratch, all convolutional layers have to be trained in advance.

The convolutional layers extract features such as edges, squares, circles, etc. These are present in almost every image and the question arises whether you can reuse them to reduce the training effort. The idea of Transfer Learning is to use an already pre-trained CNN and just adapt the neural network at the end of the Convolutional neural network to the own problem:

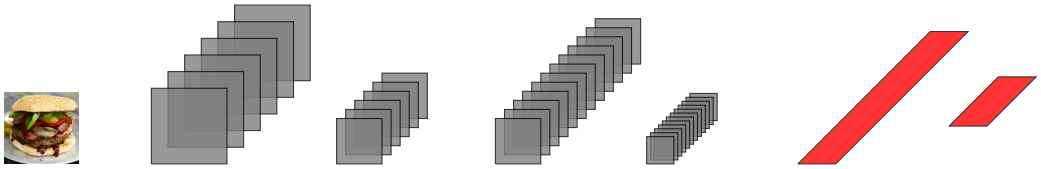

Figure 14: The green area (the neural network) was replaced by a network (red) adapted to the new problem (see Figure 12: Architecture of a traditional convolutional neural network).

The advantage of a pre-trained network can be seen in the chapter “Use of the transfer learning

approach" of this thesis.

2.2.4 Overview of current and known convolutional neural networks

Last but not least, here are a few current and well-known convolutional neural networks. They differ mainly in the following metrics, whereby in combination each network has its advantages and disadvantages:

- the top-1 accuracy (based on the ImageNet image dataset)
- the computing operations which are required for a single forward pass (G-Ops)
- the model size (for comparison: the model size of InceptionV3 is about 180 MB)

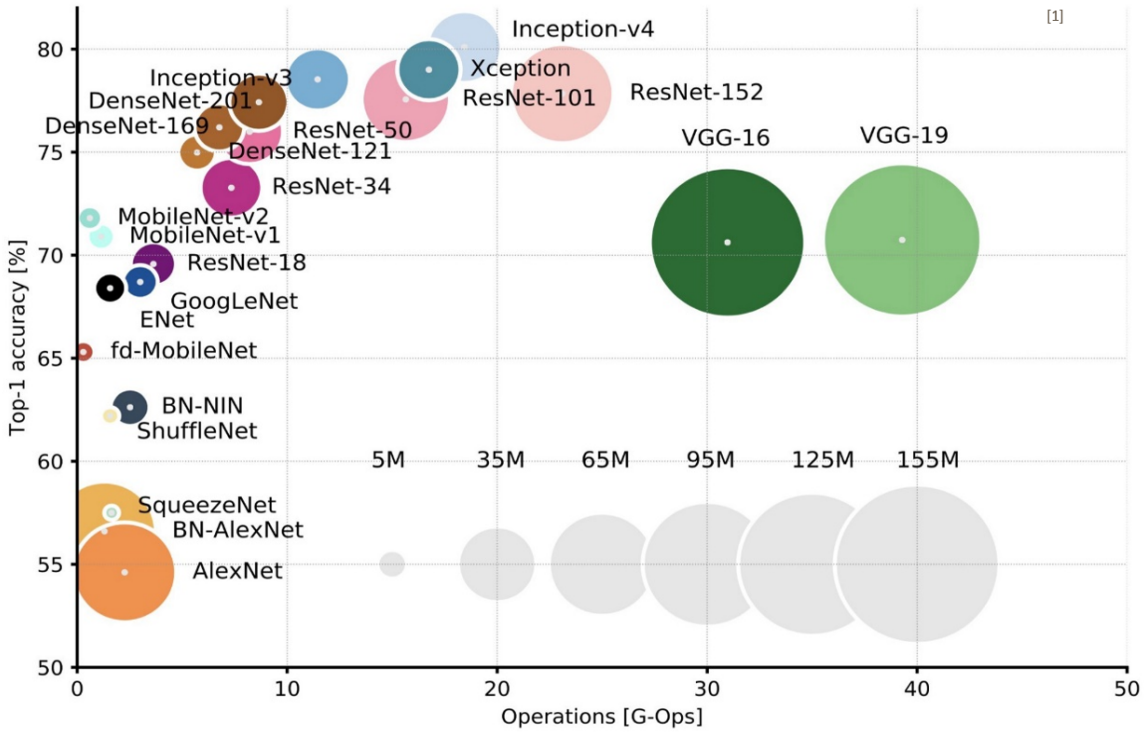

Figure 15: Overview of current and known convolutional neural networks.[34]

2.3 Further definitions

2.3.1 Learning epoch

A learning epoch is understood to be the one time training run with all training data sets. Usually one training run is not sufficient, which is why further learning epochs follow. As soon as the performance of the network does not increase anymore with further epochs, the training is terminated.

2.3.2 Learning rate

The learning rate is a tuning parameter in an optimization algorithm. It tells us how large the step size is for each iteration with which the parameters move closer to the minimum of a loss function. It should not be too high (minimum is not found) and not too small (very slow learning). The learning rate is often indicated by the character η or α.

[34]Source: `https://towardsdatascience.com/neural-network-architectures-156e5bad51ba`

2.3.3 Batch Size

The Batch Size defines the number of images that are transmitted simultaneously over the network before the parameters in the network are customized. The advantage is that only the data necessary for the current batch needs to be kept in memory. The disadvantage is the increased computing effort and an inaccurate estimation of the gradient, because the change only applies to the current batch and not to all images[35].

2.3.4 Data Augmentation

Data Augmentation is the artificial augmentation of a data set. This is primarily used when only little data is available. Existing images are rotated, mirrored, color adjusted, cropped, etc. Data Augmentation can improve model accuracy during training. Is there an example that proves this quickly?

3 Insufficient amount of data

If you give a person a donut and explain to him that it is a donut, then after some repetition he is able to classify this donut in the future. With Machine Learning this problem is a bit more complex. As with most machine learning methods, a large amount of data is required. How much is not properly documented. Especially when you are dealing with many classes to be predicted, experience shows that the amount of data increases. Some opinions in forums and blog articles say (hypothesis) that there must be at least 1000 pictures per class.[36,37,38,39]

Depending on the number of classes to be trained, you will quickly arrive at the required data set, which consists of several gigabytes of data. With Transfer Learning it is possible to reduce this number a little bit, but the problem of the large amount of data remains. A paper from Microsoft in 2001 showed at that time that simple algorithms with enough data gave similar results as complex algorithms based on less data. The researchers referred to data which should classify language constructs:

"*We have shown that for a prototypical natural language classification task, the performance of learners can benefit significantly from much larger training sets.*"[40]

Another article only a few years later also addresses this issue. This referred to data that learn from texts and that usually only small or medium sized data sets are available. To improve the efficiency also in this case, it is a good idea to improve the algorithms and methods: Examples!

"...."[41]

[35] "What is batch size in neural network?", itdxer on stackexchange.com, February 9, 2020, `https://stats.stackexchange.com/questions/153531/what-is-batch-size-in-neural-network`

[36] "Deep Learning for Image Classification with Less Data", https://towardsdatascience.com, February 2, 2020, `https://towardsdatascience.com/deep-learning-for-image-classification-with-less-data-90e5df0a7b8e`

[37] "How many images do you need to train a neural network?", https://petewarden.com, February 2, 2020, `https://petewarden.com/2017/12/14/how-many-images-do-you-need-to-train-a-neural-network/`

[38] "What is the minimum sample size required to train a Deep Learning model - CNN?", https://www.researchgate.net, February 2, 2020, `https://www.researchgate.net/post/What_is_the_minimum_sample_size_required_to_train_a_Deep_Learning_model-CNN`

[39] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems*. https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf, 2012, pp. 1097–1105.

[40] Michele Banko and Eric Brill. "Scaling to very very large corpora for natural language disambiguation". In: *Proceedings of the 39th annual meeting on association for computational linguistics*. Association for Computational Linguistics. https://www.aclweb.org/anthology/P01-1005.pdf, 2001, pp. 26–33.

[41] Alon Halevy, Peter Norvig, and Fernando Pereira. "The unreasonable effectiveness of data". In: *IEEE Intelligent Systems* 24.2 (2009), pp. 8–12.

4 Related work

There are a number of studies on image classification on very large datasets.[42,43,44] Most of the time this huge number of classification classes and datasets is not desired. Furthermore, there are many investigations on small data sets. What they are missing is the fact that the most common tuning parameters can be found in one overview. Data is expensive to obtain and usually not easy to get (subsubsection 2.1.3: Balanced training data set). And in contrast, image classifications are appearing in more and more areas of our life and are also being used directly in more and more companies that have not made use of them so far and are now considering introducing their own implementations. There are e.g. companies which try to classify products based on different data. Is it a good idea to implement your own implementation or is the step to the software tools of company giants like Microsoft, Google and Co. unavoidable? A person sees an article and classifies it: From the text, the description or an image. Sometimes only an image remains, because e.g. texts have not been maintained properly or only return cryptic values. Even this classification is often not a challenge for humans. In this case he also recognizes the product X on the basis of the still existing image.

This thesis deals with the image classification part. Since these are companies with limited resources, the question is: With which tools, techniques and tricks is it possible to carry out a successful image classification. Do the conditions mentioned in section 3: Insufficient amount of data have to be fulfilled or are successful classifications already possible with less? Is it possible to get the most out of model creation by adjusting certain tuning parameters? For example, is a cluster analysis and the associated categorical breakdown of the classification a successful approach? These and other questions are to be clarified with this thesis. And I dare to say at this point that it is possible to reach a good goal even with fewer requirements.

5 Validation process

This is the part where I explain my approach.

5.1 Preamble

In the following, the best possible accuracy is to be achieved by testing various parameters. A learning set with the following properties was used:

- 14865 images
- classified within 50 classes
- different number of images per class (unbalanced)

With the exception of the model tests, all tests were based on the following parameters (whereby one value of the parameters varied depending on the chapter):

- model: resnet18
- learning rate: 0,001 (decreases every 7 epochs to 10% of the previous value)
- batch size: 48
- epochs: 21 (learning rate from epoch 15 to 21: 0,00001)

[42] Jia Deng et al. "What does classifying more than 10,000 image categories tell us?" In: *European conference on computer vision*. Springer. http://vision.stanford.edu/pdf/DengBergLiFei-Fei_ECCV2010.pdf, 2010, pp. 71–84.

[43] Yi Sun, Xiaogang Wang, and Xiaoou Tang. "Deep learning face representation from predicting 10,000 classes". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. https://www.cv-foundation.org/openaccess/content_cvpr_2014/papers/Sun_Deep_Learning_Face_2014_CVPR_paper.pdf, 2014, pp. 1891–1898.

[44] Krizhevsky, Sutskever, and Hinton, "Imagenet classification with deep convolutional neural networks".

- image size: 224x224 pixels
- the entire training and validation set (14865 images)

Different models were tried out in chapter ≪**??**[45]≫ with the same parameters as above:

- ResNet18
- ResNet50
- ResNet152
- AlexNet
- VGG
- SqueezeNet
- DenseNet
- Inception v3

5.2 Working environment

Explain in this part of the thesis the frameworks, environments and hardware used, etc.

5.3 Splitting and preparing the data

5.3.1 Situation

We have 14866 images differently distributed in 50 classes (unbalanced). We would like to divide these into 80% training and 20% validation images.

5.3.2 Unbalanced

The unbalanced dispersion data set is divided exactly in the same ratio:

- 2953 images for the training
- 11913 images for validation

For training with different training elements, the validation dataset of 2953 images is retained for a comparable result. The number of training elements deviating from the total data set results from this:

$$n_{train} = k \cdot 500; k \in 1 \dots 26 \tag{16}$$

5.3.3 Balanced

...

5.4 Performance

...

[45] see on page **??** chapter **??** ≪**??**≫

5.5 Accuracy and evaluations

...

5.5.1 Influence of number of trained images on accuracy

...

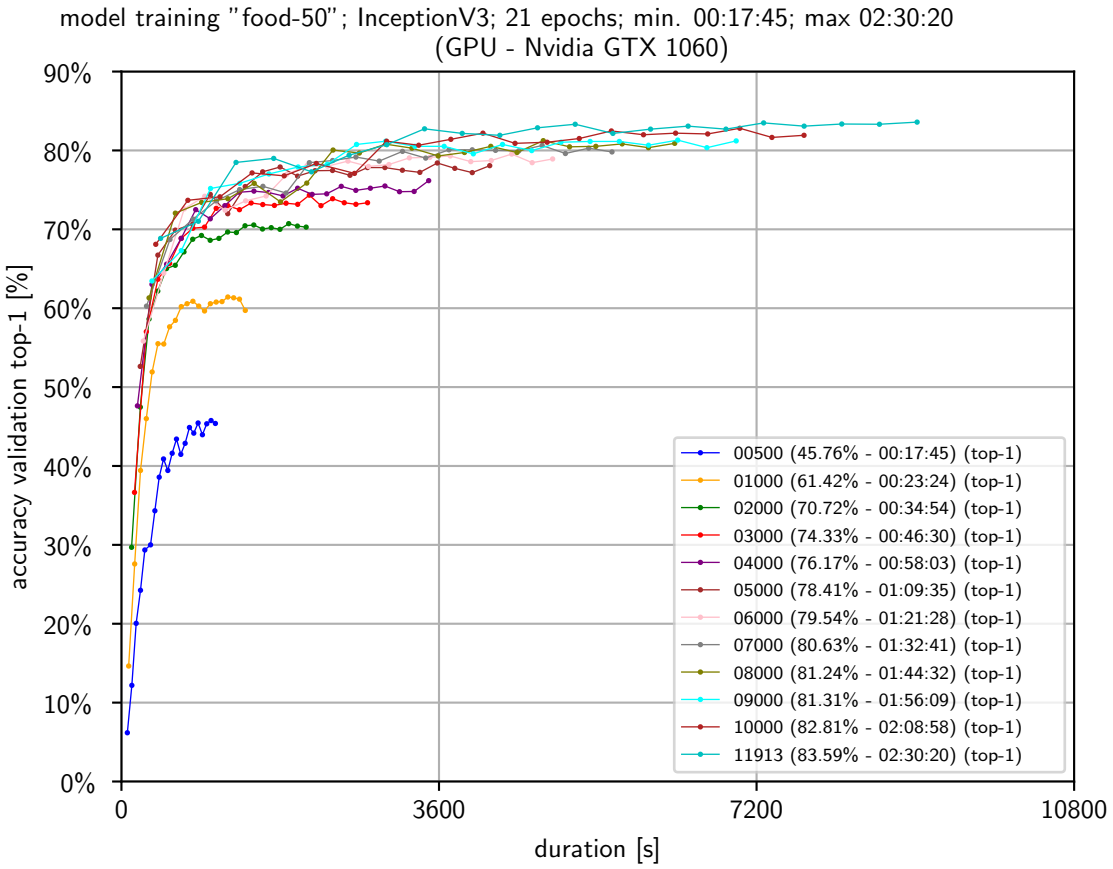

Figure 16: Overview of influence of number of trained images on accuracy

...

5.5.2 Comparison of different CNN models

...

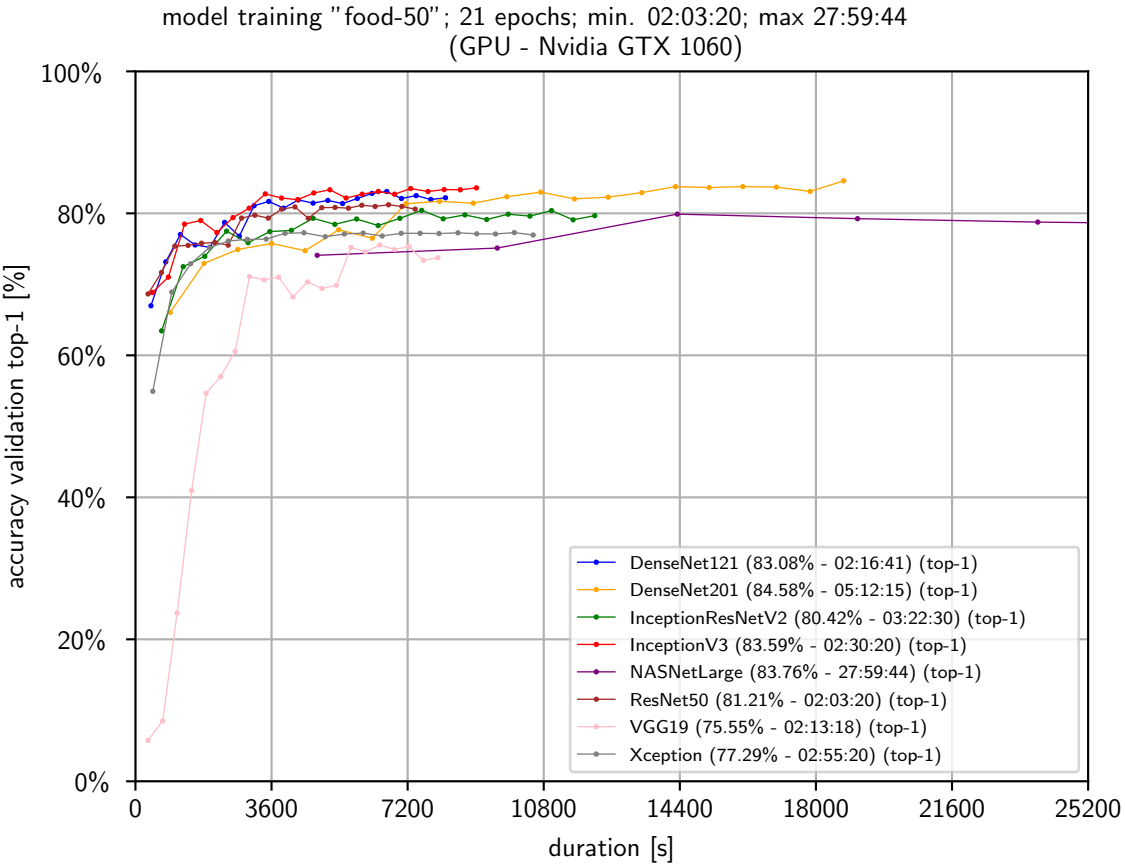

Figure 17: Overview of known transfer learning models

...

5.5.3 Use of the transfer learning approach

...

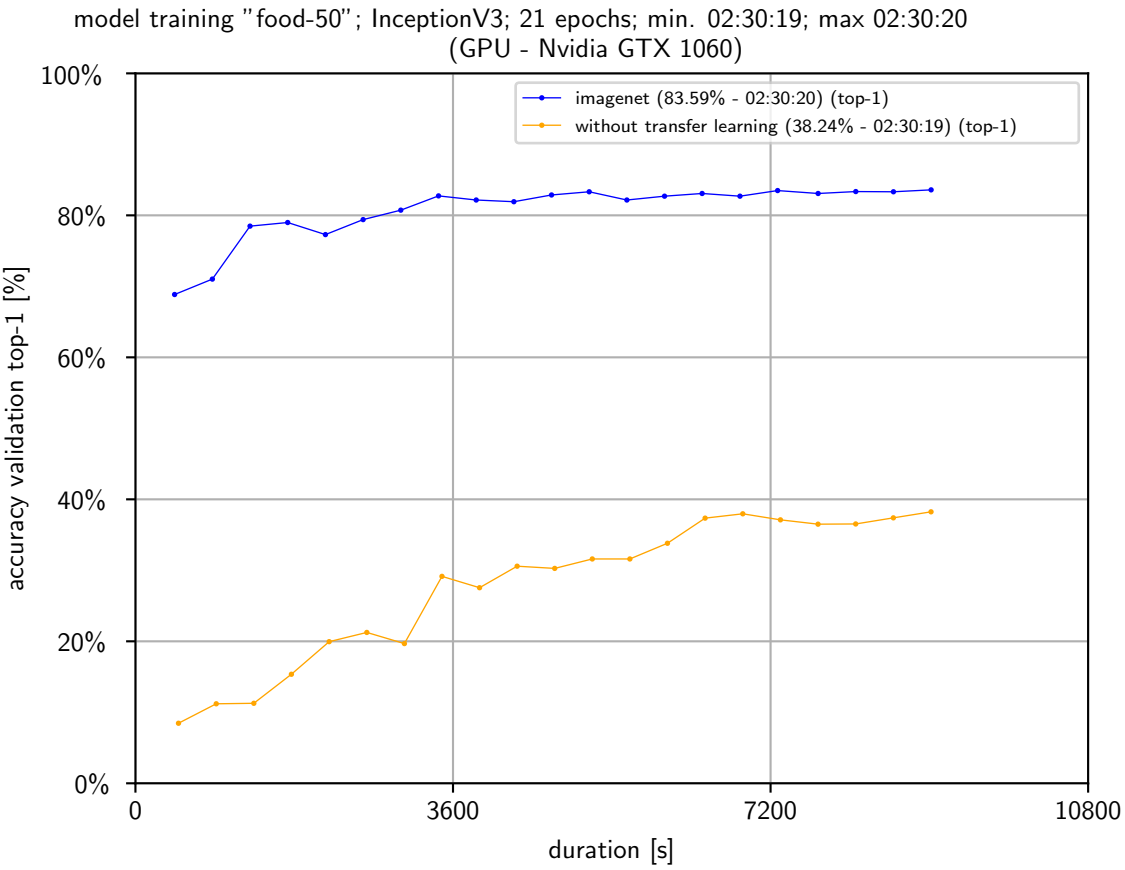

Figure 18: Overview of use of the transfer learning approach

...

5.5.4 Influence of different error optimizers

5.5.4.1 Comparison Optimizer

...

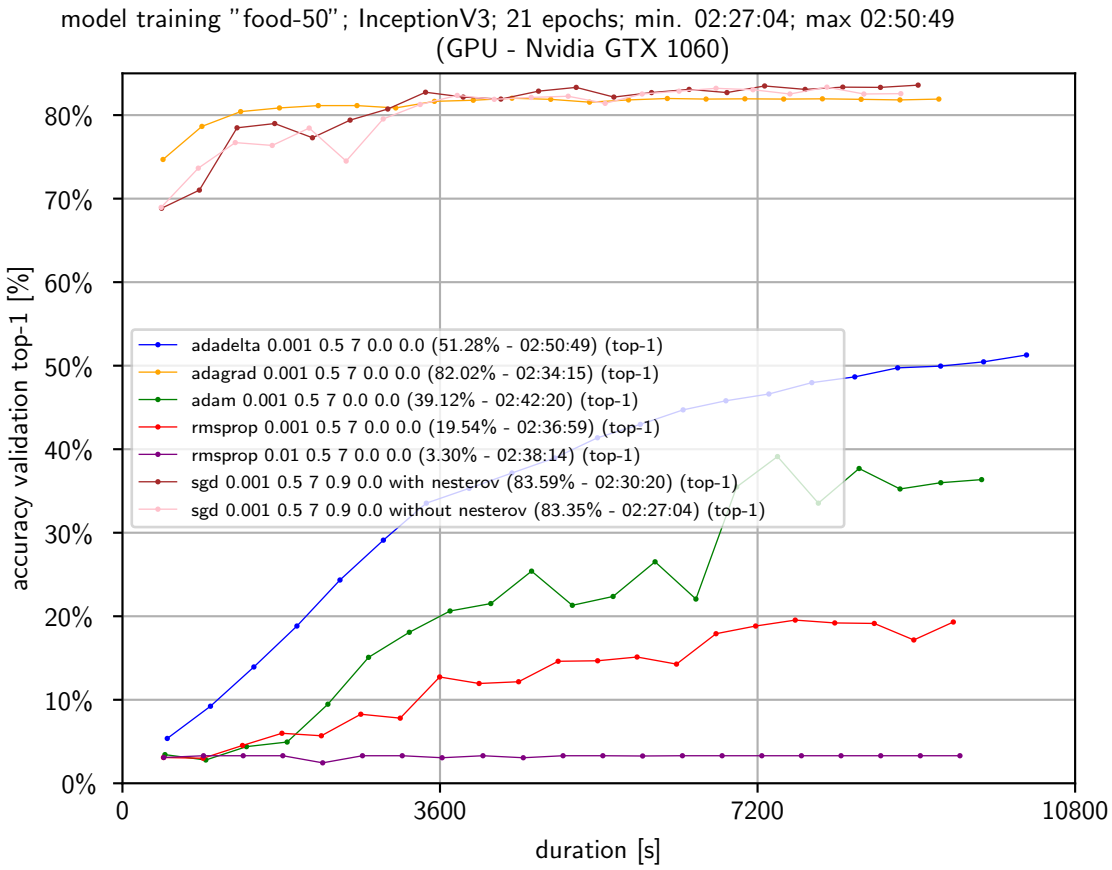

Figure 19: Overview of best optimizer

...

5.5.4.2 Influence of the momentum and the Nesterov momentum

...

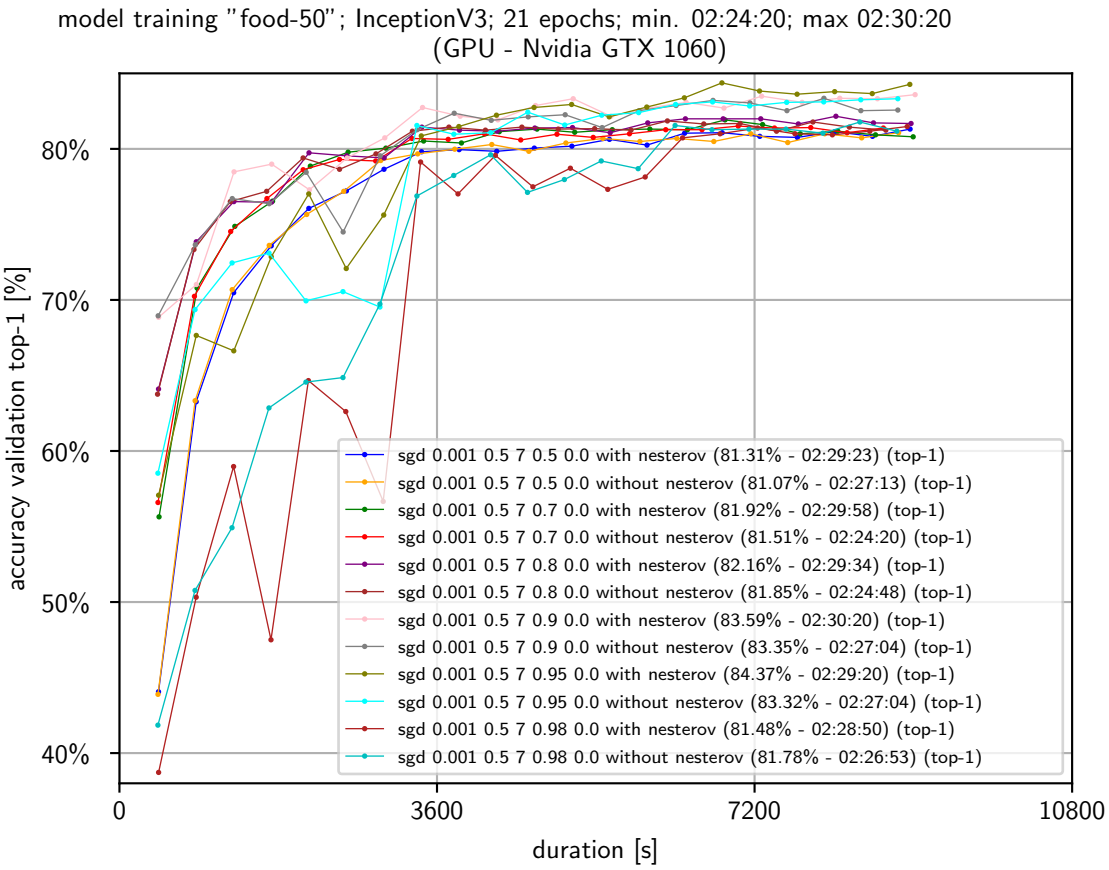

Figure 20: Overview momentum vs nesterov momentum

...

5.5.5 Influence of the number of trained layers on the accuracy

...

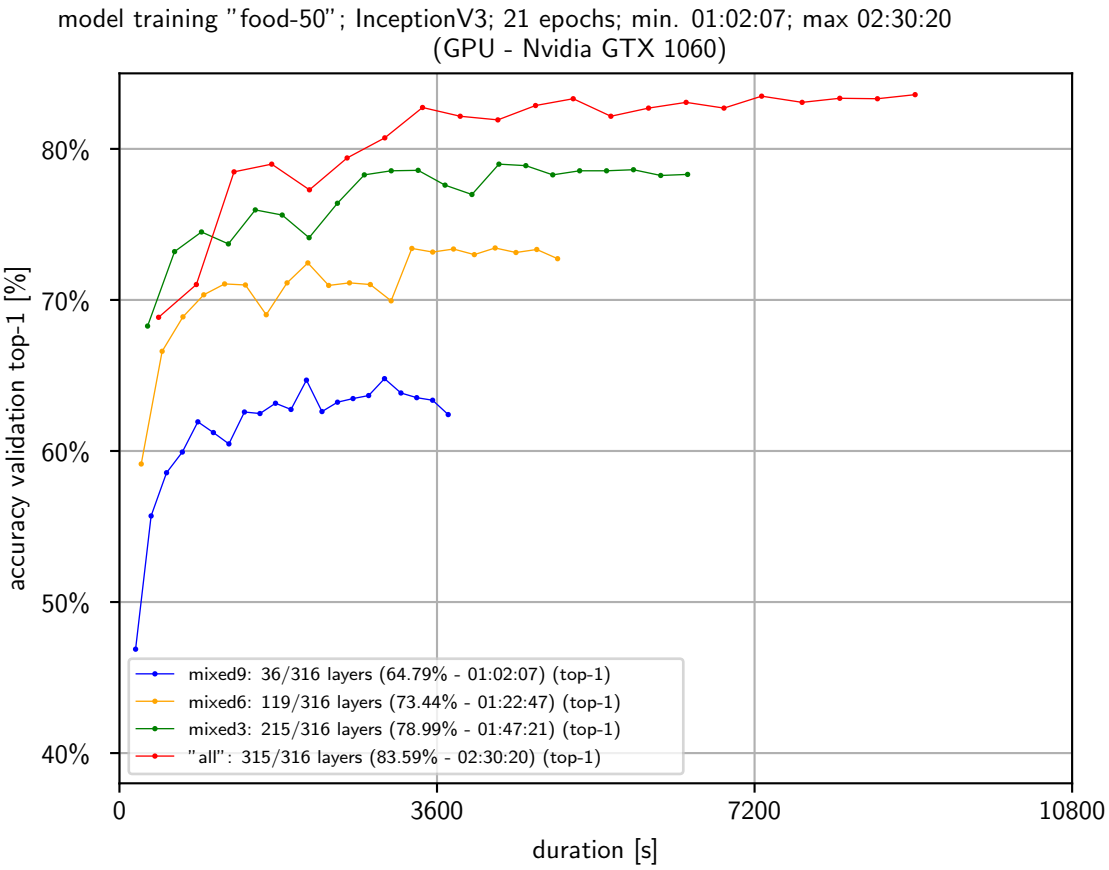

Figure 21: Overview of influence of the number of trained layers

...

5.5.6 Influence of a dynamic learning rate on accuracy (scheduling)

...

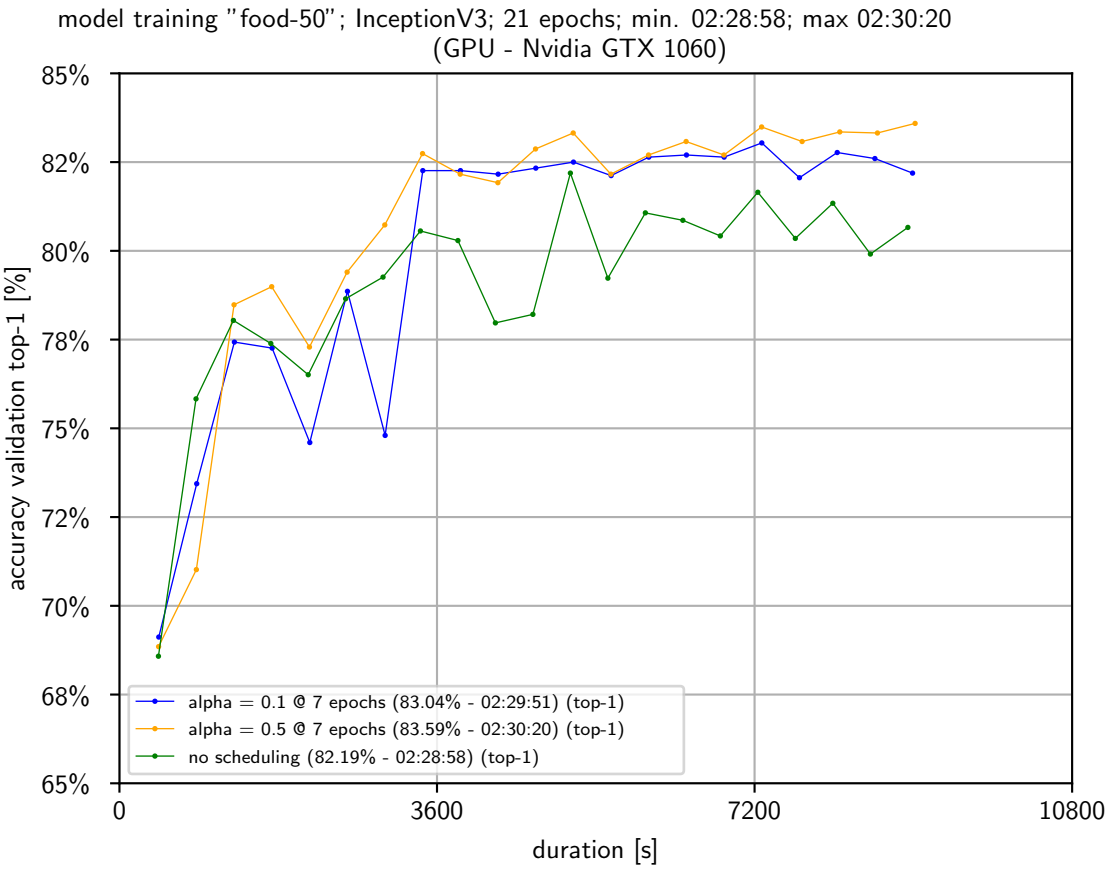

Figure 22: Overview of a dynamic learning rate on accuracy

...

5.5.7 Different batch sizes

...

5.5.8 Different image sizes

...

5.5.9 Different number of learned epochs

...

6 Optimization process

This chapter contains ideas, approaches and evaluations of more complex ideas, which do not fit into the range of simple parameter changes.

6.1 Preamble

...

6.2 Data augmentation

...

F:/data/raw/food-50-augment/_other/chocolate-chips-cookies-american-cookies.jpg

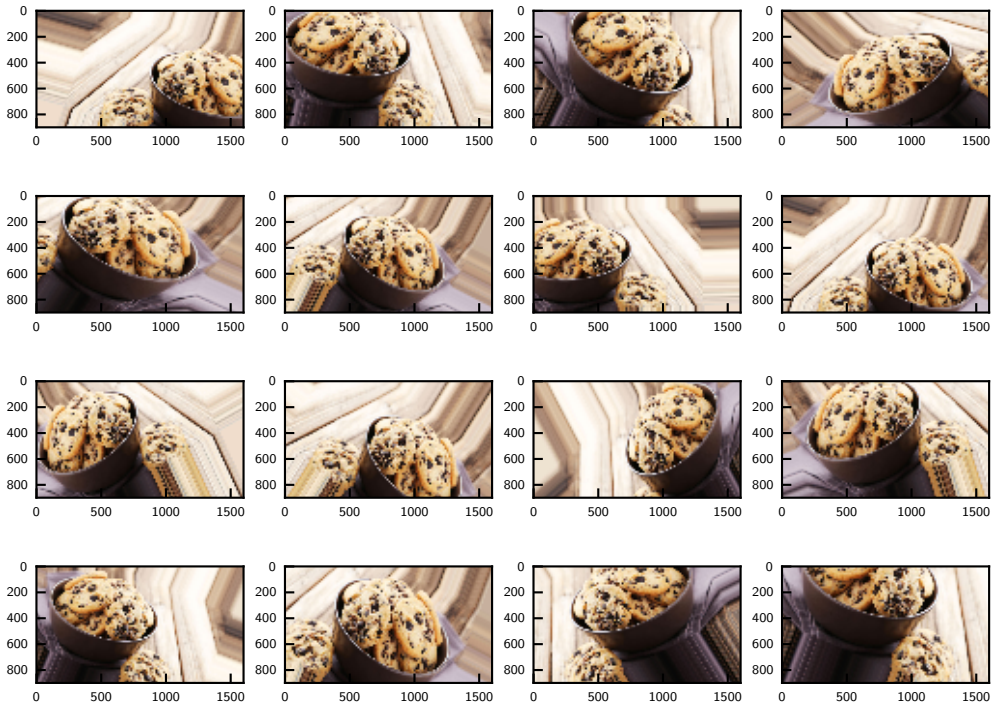

Figure 23: Data Augmentation

6.3 Enrichment of the data set from other data sources

...

6.4 Analyses with multidimensional scaling

...

6.5 Hierarchical classification

By using a single model for all classes, previous classifiers have been trained to minimize the loss of the class output vector. Each class used so far has the same rank in both training and classification. The prediction of "Pizza" costs the same as the prediction of "Martini".

The human ability to classify objects does not only work on one level. Categories will naturally overlap and have a hierarchical structure. For example, a human will classify a picture under "pizza", "tuna pizza" or even "fast food", which is correct from this point of view. Depending on the classification, there will only be a "loss of information". However, a person will not mistake a "pizza" as a "Martini", which is more likely to be classified as a "drink" or "cocktail".[46]

[46] Eleanor Rosch et al. "Basic objects in natural categories". In: *Cognitive psychology: Key readings* 448 (2004).

6.6 Binary classifiers

...

6.7 Evaluation

...

6.8 Use of the model across programming languages

...

7 Summary and outlook

What's the outcome? What else is possible? How can this work be continued? In here!

List of figures

List of Tables

List of literature

Banko, Michele and Eric Brill. "Scaling to very very large corpora for natural language disambiguation". In: *Proceedings of the 39th annual meeting on association for computational linguistics*. Association for Computational Linguistics. https://www.aclweb.org/anthology/P01-1005.pdf, 2001, pp. 26–33.

deeplizard. *Convolutional Neural Networks (CNNs) explained*. Youtube. 2017. URL: `https://www.youtube.com/watch?v=YRhxdVk{\_}sIs`.

Deng, Jia et al. "What does classifying more than 10,000 image categories tell us?" In: *European conference on computer vision*. Springer. http://vision.stanford.edu/pdf/DengBergLiFeiFei_ECCV2010.pdf, 2010, pp. 71–84.

Géron, Aurélien. "Praxiseinstieg Machine Learning mit Scikit-Learn und TensorFlow: Konzepte, Tools und Techniken für intelligente Systeme". In: O'Reilly Verlag, 2017. Chap. Konfusionsmatrix, pp. 86–88. ISBN: 9783960090618.

— "Praxiseinstieg Machine Learning mit Scikit-Learn und TensorFlow: Konzepte, Tools und Techniken für intelligente Systeme". In: O'Reilly Verlag, 2017. Chap. Entscheidungsgrenzen, pp. 138–140. ISBN: 9783960090618.

— *Praxiseinstieg Machine Learning mit Scikit-Learn und TensorFlow: Konzepte, Tools und Techniken für intelligente Systeme*. O'Reilly Verlag, 2017, pp. 8–14. ISBN: 9783960090618.

Halevy, Alon, Peter Norvig, and Fernando Pereira. "The unreasonable effectiveness of data". In: *IEEE Intelligent Systems* 24.2 (2009), pp. 8–12.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems*. https://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf, 2012, pp. 1097–1105.

Osinga, Douwe. *Deep Learning Kochbuch: Praxisrezepte für einen schnellen Einstieg*. O'Reilly Verlag, 2019, pp. 19–26. ISBN: 9783960090977.

Rosch, Eleanor et al. "Basic objects in natural categories". In: *Cognitive psychology: Key readings* 448 (2004).

Sun, Yi, Xiaogang Wang, and Xiaoou Tang. "Deep learning face representation from predicting 10,000 classes". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. https://www.cv-foundation.org/openaccess/content_cvpr_2014/papers/Sun_Deep_Learning_Face_2014_CVPR_ 2014, pp. 1891–1898.

List of links

- Deep learning unbalanced training data?
  - https://towardsdatascience.com/deep-learning-unbalanced-training-data-solve-it-like-this-6c528e9efea6
- Data Augmentation
  - https://machinelearningmastery.com/how-to-configure-image-data-augmentation-when-training-deep-learning-neural-networks/
- Stop Feeding Garbage To Your Model! — The 6 biggest mistakes with datasets and how to avoid them.
  - https://hackernoon.com/stop-feeding-garbage-to-your-model-the-6-biggest-mistakes-with-datasets-and-how-to-avoid-them-3cb7532ad3b7

Declaration

I hereby declare that the work presented in this thesis is solely my work and that to the best of my knowledge this work is original, except where indicated by references to other authors. No part of this work has been submitted for any other degree or diploma.

Signature :

Place, Date :