

2203628

by Chenyang Li

Submission date: 30-Mar-2023 04:03PM (UTC+0100)

Submission ID: 201044388

File name: 40779_Chenyang_Li_2203628_495066_1652531040.docx (94.16K)

Word count: 2560

Character count: 15442

Report

1 Abstract

Today's world is developing faster and faster, and mobile phones have become a tool that can control the world's information from the previous one that can only talk, the way people receive images has become increasingly handy and effective. As a result, with the increasing number of image data on the Internet, image data contains more and more information, and the storage of image data becomes increasingly difficult. Therefore, preserving information in images becomes increasingly important. The information in the pictures is also important to us. This information not only helps us obtain insights from people, but also provides us with a large amount of data on human behavior.

Image classification has been widely used in many fields, such as object identification, face recognition, image updating, etc. This method is of great importance to other areas of study. Firstly, it is necessary to manage the classification of the images effectively. Moreover, it's hard to sort out the picture data from a computer with the same degree of agility as people do. The problem is not just about the irregularity of the picture, but also about the size of the picture and the way it is presented. Image classification is one of the most challenging tasks in recent years.

In this paper, svm algorithm in sklearn framework and tensorflow framework are used to write an mlp algorithm for image classification data set experiments, and the experimental results are better mlp algorithm.

2 Introduction

Swain and Ballard proposed a method based on color histograms to analyze the differences in color distribution between two images by comparing the differences in color histograms. However, due to their insensitivity to other characteristics such as shape, the effect is not very ideal. In the 1980s, Pawlak. Z et al^[1]. proposed the rough set theory, which interprets knowledge by introducing equivalence relations in algebra, and obtains better classification results by processing some knowledge.

In addition, classification based on machine learning methods mainly consists of Bayesian methods and neural network methods.

The Bayesian classification method mainly includes: First, we calculate the class distribution for each training group and treat it as the probability distribution for each class. Then, we can estimate the probability of which category the feature sample belongs by applying Bayesian theorem in probability theory and statistical methods. Bayesian methods can also be divided into Bayesian belief networks and naive Bayesian methods.

However, Bayesian belief networks have been widely used in this method.

However, its training process is complex and selection is difficult.

The main issues to be addressed. In recent years, deep learning models have made significant breakthroughs in extracting high-level feature representations from images. In the 1960s, in the research of neurobiologists, the processing of information by the human brain's visual system was hierarchical. It has been widely used in various areas, and it has become one of the hot spots in machine learning. Deep learning is a kind of neural network based on neural networks, which is based on neural networks. Lots of hidden layers, like the V1 and V2 regions of the brain, can form high-level feature representations from pixels to targets, which traditional shallow learning structures do not possess. However, constructing multi layered network structures cannot efficiently complete training. Until 2006, Hinton et al^[2]. proposed using unsupervised greedy layer by layer training algorithms to solve the training problem of depth structures, which attracted people's attention to deep learning. Subsequently, deep learning models have been improved and optimized, such as convolutional depth confidence networks, denoising automatic encoders, and deconvolution networks

In this paper, we combine the theory of machine learning with

convolutional neural networks to produce multi-ML models for image classification.

Sklearn is a robust, third-party, Python language library, which includes all aspects of the pre-process and the training model. User learning in this area can cut down on the time required for coding and coding, so that we can devote more time to analysing the data, making adjustments to the model, and making adjustments to the overparameters. An Estimator is a model that is used to predict or regression data. Generally, there are two ways to estimate the estimation: Using input data and tags, the training time depends on the parameters, the size of the datasets, and the features of the data. But in various cases, the criterion of evaluation of the model is not only based on the precision ratio, but also on the other indexes, particularly in the case of unbalanced classification. Precision is not a good assessment of the model, so don't be deceived by the marks when evaluating the model. The Transformer can deal with standard, dimension reducing and characteristic selecting. Similar to the use of estimator.

Tensorflow is an open-source machine learning framework that lets you rapidly construct a neural network, which can be used to train, assess, and store data rapidly. This method can be applied to the study of Machine Learning and Deep Neural Networks, as well as the rapid development of

certain Machine Learning Algorithms. Portability and Cross Platform: You can run the same code and model on a server, PCS, and a mobile device.

4 Related work

In the practical application of image recognition technology of machine learning, it is mainly the comprehensive application of multiple technologies. Image recognition technology is an important way to interpret images. In the aspect of specific technology application, the original signal sequence is transformed by the corresponding algorithm to form the results that can be recognized. Image recognition technologies of machine learning include pair classification and clustering algorithm principle image gray segmentation, such as finite mixture model and artificial neural network. By taking advantage of the image recognition technology of machine learning, the quality and efficiency of image recognition can be improved, which can promote the development of social production.

Deep learning detector realizes feature extraction through convolutional neural network. Compared with other types of neural networks, convolutional neural network has the advantages of small parameter number, strong scalability, ability to independently complete the learning of target features, and the extracted features have strong robustness and

anti-interference. With the deepening of research, there are two main mainstream target detection algorithms at present. The first is called two-stage algorithm model, represented by R-CNN, Fast R-CNN, Faster R-CNN, etc., such algorithms first need to screen out potential target candidate regions from input images. The subsequent convolutional neural network was used to optimize and adjust the candidate regions, and the classification and boundary frame prediction information were obtained. The accuracy of this algorithm is generally higher, but its reasoning speed is slow. The other is the one-stage algorithm model based on regression idea, such as SSD, FCOS, YOLO series, etc. This kind of algorithm does not need to screen candidate regions in advance, but to divide the final obtained feature map into different regions, and directly output a certain amount of prediction box size and classification information of the region in each region. The accuracy of this kind of algorithm is generally slightly lower than that of the two-stage algorithm, but due to the lack of the screening and optimization process of prediction box, its detection speed is faster and its real-time performance is higher.

4 Methods

Support Vector Machine (SVM) is a guiding classification method. The specific research content includes: Assuming that there is some similarity between two different types of sample points, by constructing a segmentation hyperplane that can segment two different types of sample

points, and selecting a segmentation hyperplane with strong generalization performance, the similarity between the two sample points is maximized. Assuming that there is a certain correspondence between the classification results of the dataset and the classification results of the training sample set, by constructing a segmentation hyperplane that can segment two different types of sample points, and selecting a segmentation hyperplane with strong generalization performance, the similarity between the two sample points is maximized, so that two different types of sample points can be classified. SVM algorithm is based on this principle to achieve classification problems.

Support Vector Machine (SVM) is a machine learning method based on SVM. This is a binary pattern that maps the characteristic values of a sample to a certain point. Support vector machines aim to draw a "best" distinction between two types of data, in order to better distinguish them when new data appears in the future. SVM is suitable for processing small sample, non-linear, high-dimensional data. SVM was first presented by Vladimir Vapnik and Chervonenkis in 1963, and the current SVM (SVM) was published by Corinna Cortes and Vapnik in 1995 in 1993^[3]. Before the rise of deep neural networks (2012), SVM was recognized as the most successful and optimal method in the past decade. However, due to the high computational complexity and the limitations of kernel functions and

polynomial time complexity, it is difficult to generalize in practical applications.

Neural networks in deep learning can also be seen as a special form of support vector machines. In deep learning, there is a set of implicit nonlinear relationships between inputs and outputs, which are composed of fully connected layers. At the beginning of neural networks, there was only one layer of neurons between an input layer and an output layer, but in later development, a hidden layer was gradually added. At first, neural networks were composed of several simple multi-objective programming, and later gradually developed into neural networks with more hidden layers and more complex structures. There are two very important hidden layer structures: forward propagation and reverse propagation. Backpropagation is defined by the difference between the sum of the weights of each neuron in the network and the sum of the weights of neurons between the input and output layers.

Multilevel perceptron (MLP), also known as ANN (Interaction Networks), can contain multiple hidden layers other than the output layer. The simplest multi objective programming has only one hidden level, known as the so-called three levels.

The output of the first layer is a series of features, while the output of the second layer is a linear combination of these features. MLP maximizes the objective function by using a minimum loss function. For example, it aims

to maximize output error and minimize it. This structure leads to a large number of optimization problems, including gradient descent, Newton iterative method, and fast gradient descent algorithm. The MLP structure enables neural networks to learn at a very fast speed, and even with a large number of samples, it can achieve good generalization performance. The biggest advantage of MLP is that it can learn the relationships between hidden levels, such as forward and backward propagation. Due to its powerful generalization performance, MLP has always been one of the most popular and potential machine learning technologies in the field of machine learning.

However, the disadvantage of MLP is that its model requires high training data requirements, and its model complexity is high. As the dataset continues to grow, this model will become slower and slower.

5 data introduction

This paper mainly uses the Fashion-MNist dataset, which in total contains a dataset of 60,000 28x28 grayscale images in 10 Fashion categories, and a test set of 10,000 images, with 10 fashion labels as follows:

0↵	T-shirt↵
1↵	Trouser↵
2↵	Pullover↵
3↵	Dress↵
4↵	Coat↵
5↵	Sandal↵
6↵	Shirt↵
7↵	Sneaker↵
8↵	Bag↵
9↵	Ankle boot↵

6 data preprocessing



Data normalization is performed for each column of data. Since the maximum value of all positions is 255 and the minimum value is 0, the normalization operation of each position is processed using the python lambda function. Here are some of the codes::

```
for i in range(784):  
    train_df[i] = train_df[i].apply(lambda x:x/255)  
    test_df[i] = test_df[i].apply(lambda x:x/255)
```

7 Model Construction

The main methods used in model construction are: Firstly, SVM model method is used in sklearn; firstly, relevant function installation package is imported; secondly, model training is conducted. The code is shown as follows:

```
from sklearn.svm import SVC  
  
model = SVC(kernel='linear')  
model.fit(train_df.drop('label',1), train_df[['label']])
```

Secondly, test the test set. The following is the code for the test::

```
test_predict = model.predict(test_df.drop('label', 1))  
Finally, the accuracy of the svm model effect was evaluated, as shown below:  
from sklearn.metrics import accuracy_score  
print("accuracy:", accuracy_score(y_test, test_predict))  
accuracy: 0.8464
```

Another approach is to use tensorflow to build the mlp model by first writing a data loader based on the data situation. The main code definition is as follows:

```
class FashionMnistLoader():  
    def __init__(self):  
        mnist = tf.keras.datasets.mnist  
        (self.train_data, self.train_label), (self.test_data, self.test_label) =  
        keras.datasets.fashion_mnist.load_data()
```

```

        self.train_data = np.expand_dims(
            self.train_data.astype(
                np.float32) / 255.0,
            axis=-1) # [60000, 28, 28, 1]
        self.test_data = np.expand_dims(
            self.test_data.astype(np.float32) / 255.0,axis=-1) # [10000, 28, 28,
1]

        self.train_label = self.train_label.astype(np.int32) # [60000]
        self.test_label = self.test_label.astype(np.int32) # [10000]
        self.num_train_data, self.num_test_data = self.train_data.shape[0],
self.test_data.shape[0]

    def get_batch(self, batch_size):
        index = np.random.randint(0, np.shape(self.train_data)[0], batch_size)
        return self.train_data[index, :], self.train_label[index]

```

Secondly, tensorflow is used to construct the model structure, which is shown as follows:

```

class MLP(tf.keras.Model):
    def __init__(self):
        super().__init__()
        self.flatten = tf.keras.layers.Flatten()
        self.dense1 = tf.keras.layers.Dense(units=100, activation=tf.nn.relu)
        self.dense2 = tf.keras.layers.Dense(units=10)

    def call(self, inputs):
        # [batch_size, 28, 28, 1]
        x = self.flatten(inputs) # [batch_size, 784]
        x = self.dense1(x) # [batch_size, 100]
        x = self.dense2(x) # [batch_size, 10]
        output = tf.nn.softmax(x)
        return output

Set related parameters
import numpy as np

num_epochs = 5
batch_size = 50
learning_rate = 0.001
model = MLP()
data_loader = FashionMnistLoader()
optimizer = tf.keras.optimizers.Adam(learning_rate=learning_rate)

```

Model training code

```
num_batches = int(data_loader.num_train_data // batch_size * num_epochs)
for batch_index in range(num_batches):
    X, y = data_loader.get_batch(batch_size)
    with tf.GradientTape() as tape:
        y_pred = model(X)
        loss = tf.keras.losses.sparse_categorical_crossentropy(
            y_true=y, y_pred=y_pred)
        loss = tf.reduce_mean(loss)
        print("batch %d: loss %f" % (batch_index, loss.numpy()))
    grads = tape.gradient(loss, model.variables)
    optimizer.apply_gradients(grads_and_vars=zip(grads, model.variables))

sparse_categorical_accuracy = tf.keras.metrics.SparseCategoricalAccuracy()
num_batches = int(data_loader.num_test_data // batch_size)
for batch_index in range(num_batches):
    start_index, end_index = batch_index * \
        batch_size, (batch_index + 1) * batch_size
    y_pred = model.predict(data_loader.test_data[start_index: end_index])
    sparse_categorical_accuracy.update_state(
        y_true=data_loader.test_label[start_index: end_index], y_pred=y_pred)
print("test accuracy: %f" % sparse_categorical_accuracy.result())
test accuracy: 0.871500
```

8 Conclusion

This article first introduces the main content of two recognition image classification methods, and then introduces the work of image related researchers. Secondly, it introduces the svm and mlp algorithms, and then predicts the training steps of data sets and data processing models. Finally, it forecasts the experimental results. The experimental results in this paper show that traditional methods have obvious limitations in recognizing image classification, and they also have some problems. Therefore, deep learning based methods may provide new solutions to these problems.

Deep learning has made some progress in image classification, the most famous of which are convolutional neural networks (CNN), cyclic neural networks (RNN), short-term and short-term memory networks (LSTM), and fully connected networks (FCN). In image classification, these methods use neural networks to achieve automatic target recognition or classification. Although these methods have made some progress in image classification, there are still many challenges. Firstly, the scale and rotation of the target region in the image vary greatly, which makes image classification very difficult. In addition, in some applications, there may be occlusion between objects in the image and the background, which will make target recognition or classification more difficult. Finally, due to changes in some environmental and lighting conditions, the targets in the image may change.

Reference

- [1] Pawlak, Z. (1982) Rough Sets. *International Journal of Computer and Information Science*, 11, 341-356.
- [2] Hinton, G.E., & Salakhutdinov, R. (2006). Reducing the Dimensionality of Data with Neural Networks. *Science*, 313, 504 - 507.
- [3] Cortes, C., & Vapnik, V.N. (1995). Support-Vector Networks. *Machine Learning*, 20, 273-297.
- [4] 高锦 .(2010). 基于 SVM 的图像分类 (硕士学位论文, 西北大学).<https://kns.cnki.net/kcms2/article/abstract?v=6uDTENTNErAYYpqmMZ51IYnuZ3JdHiXTaf17ab1uQ2vK2DigBW8M8ujqP0O3KitwQBlnZhPoZ4nALfk6XtGPdx3J5VegjWsVfDvAyBcesPy4-gK8qlaqpge=&uniplatform=NZKPT&language=CHS>



7025code.ipynb

FINAL GRADE

46

/100

GENERAL COMMENTS

Instructor

Abstract - A little to much justification for the task here. Please include some summary findings and key results, here too.

Your style lacks clarity at times with some repetition in places. Try to be more structured in your approach.

Regarding refs, you need to correct your formatting issues in the References section and please use English exclusively. Avoid online refs and use established texts and relevant high quality journal publications please.

You provide some code for the Fashion MINST dataset and MLP and state accuracy as a single value for each.. Please justify use of parameters clearly and explain the design.

You haven't completed the 2nd part using the CIFAR10 dataset. A shame as you have already designed and implemented both classifiers and it is just a little extra work to train these using the CIFAR10 dataset.

Additionally you have not analysed and evaluated the results - use confusion matrices as we did in class, calculate F1 score, recall and precision both for overall results and for each class. and discuss the results.

Heed the above comments and you will be able to pass similar assignments.

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9



Comment 1

Capitalize headings

PAGE 10



Comment 2

Again, capitalize

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

CHENYANG LI

0

RESEARCH-INFORMED LITERATURE

KNOWLEDGE AND UNDERSTANDING

ANALYSIS

PRACTICAL APPLICATION AND DEP.

SKILLS FOR PROFESSIONAL PRAC.