Comparison research on iot oriented image classification algorithms

Ke Du[1] and Kai-Yu Cai[2]

[1]*College of Computer, National University of Defense Technology, 410073ChangSha,Hunan,P.R.China*
[2]*College of Computer, National University of Defense Technology, 410073ChangSha,Hunan,P.R.China*

Abstract. Image classificationbelongs to the machine learning and computer visionfields,it aims to recognize and classify objects in the image contents. How to applyimage classification algorithms to large-scale data in the IoT framework is the focus of current research. Based on Anaconda, this article implementsk-NN,SVM, Softmax and Neural Networkalgorithms by Python, performs data normalization, random search, HOG andcolour histogram feature extractionto enhance the algorithms, experiments on them in CIFAR-10 datasets, then conducts comparison from three aspects of training time, test timeand classification accuracy.The experimental results show that: the vectorizedimplementation of the algorithms ismore efficient than the loopimplementation;Thetraining time of k-NN is the shortest,SVM andSoftmaxspend more time, and the training time ofNeural Network is the longest;The test time of SVM, Softmax and Neural Network are much shorter than of k-NN;Neural Network gets the highest classification accuracy, SVM and Softmaxget lower and approximateaccuracies, andk-NN gets the lowest accuracy. The effects of three algorithmimprovement methods are obvious.

1 Introduction

With the fast development ofIoT and mobile Internet,the volumes of image data from different fields such as social networks and sensor networks are growing exponentially. How to process the large-scale image data and recognize objects from the image contents has became an important issue.Image classification is the way to handle this problem. In general, image classification algorith msaredata-driven approaches. They usually get global descriptions of the imagesbymanualfeaturing or learning methods, use the learned classifiers to determine whether there is a certain object in the image or not.Image classification algorithms have been applied to many fields, such as face recognition, car number recognition, image searching and so on.

Anaconda is an open source scientific computation and analysis platformreleased by Continuum Company.It installedmany scientific computation libraries, such as NumPy,SciPy andMatplotlib.It also provides an interactive toolIPython Notebook, whichcombines experiment and document writing.Based on Anaconda, researchers and developerscould usePython to implementand analysis different algorithms, and write document by markdown.

Image classification andobject detection are active research areasin the computer visionfield. AfterAlexNet[1]was proposed in 2012, those areas are developingrapidly.In ILSVRC 2015, ResNet[2]wonbyits 3.57% top5 error. In the same year, YOLO[3]provides a real-time object detection with a mAP of 63.4 and a FPS of 45.Based on Anaconda, this paperimplements k-NN,SVM, Softmax and Neural Networkalgorithms and data normalization, random search, HOG and colour histogram feature extractionalgorithmimprovement methods, and conducts comparison on them by the training time, test timeand classification accuracy.Results show that the performance of the different algorithms vary a lot,improvement methodsimprovethe classification accuracy.Algorithms should be deployed in the different parts of the IoTframework according to theirperformance.

2 Comparison on algorithms

2.1. k-NN algorithm

k-NN is short for k-Nearest Neighbour.Instead of finding the single closest image in the training set as Nearest Neighbor algorithm, k-NN finds the top k closest images, and makes them vote on the label of the test image. In particular, when k = 1, it recover to the Nearest Neighbor algorithm. Intuitively, higher values of k have a smoothing effect that makes the classifier more resistant to outliersit is improved.

k-NNtreats image as vector, using L2 distance, which has the geometric interpretation of computing the euclidean distance between two vectors, to compare the difference between two images. The distance formula takes the form:

$$d_2(I_1,I_2) = \sqrt{\sum_p \left(I_1^p - I_2^p\right)^2} \quad (1)$$

The training process of k-NN is to store all training images in the memory, then the test process is to compare test image with stored images by L2 norm, finds the top k closest images,and makes them vote on the label of the test image.

2.2. SVM algorithm

Support Vector Machine algorithm uses score function:

$$f(x_i,W,b) = Wx_i + b \quad (2)$$

tocomputetheclass scores of imagefor different classes, then uses loss function:

$$L_i = \sum_{j\neq y_i} \max\left(0, w_j^T x_i - w_{y_i}^T x_i + \Delta\right) \quad (3)$$

Tocompute the loss value.SVM loss function is often called the hinge loss.Itmeasures how consistent the predictions on training data are with the ground truth labels. The loss value of entire dataset is composed of data loss and regularizationloss:

$$L = \frac{1}{N}\sum_i \sum_{j\neq y_i} \left[\max\left(0, f(x_i;W)_j - f(x_i;W)_{y_i} + \Delta\right)\right] + \lambda \sum_k \sum_l W_{kl}^2 \quad (4)$$

In the training process, SVMuses gradient descent method to find the optimal weights and biases, which make the loss value minimum,since making good predictions on the training set is equivalent to minimizing the loss.In the test process, algorithmusesthe optimal weights and biasesto classify test images.

2.3. Softmax algorithm

The score function ofSoftmax algorithm is the same with the SVM algorithm. However, it replaces the hinge loss with cross-entropy loss for its loss function:

$$L_i = -\log\left(\frac{e^{f_{y_i}}}{\sum_j e^{f_j}}\right) \quad (5)$$

The training and test processes of Softmax are similar with SVM. However, the core idea of Softmax is different.Unlike SVM, it is never fully happy with the scores it produces: the correct class could always have a higher probability and the incorrect classes always a lower probability and the loss would always get better.

2.4. Neural network algorithm

Neural Network models are often organized into distinct layers of neurons. For regular neural networks, the most common layer type is the fully-connected layer in which neurons between two adjacent layers are fully pairwise connected, but neurons within a single layer share no connections. With an appropriate loss function on the neuron's output, it can turn a single neuron into a linear classifier.

The training process consists of two parts: the forward pass and back propagation. In forward pass, itperforms matrix multiplication and activation function to obtain image classification scoreby the current weights and biases.Take the two-layer neuralnetworkmodel in our experimentfor example, the forward pass formula is:

$$s = W_2 max(0, W_1 x) \quad (6)$$

andare the matrix combined weightand biases , is the class score.

The backpropagationis a process of computing gradients of expressions through recursive application of chain rule in networks.Algorithm then uses gradient signalto perform parameters update, till the value of loss function get small enough. In the test process, algorithm obtainsclass score by forward pass. It have been proved that Neural Network algorithm can approximate any continuousfunction[4].

3 Algorithm improvement methods

3.1. Mean subtraction and normalization

Mean subtraction and normalization are both data preprocessing methods.Mean subtraction involves subtracting the mean across every individual feature in the data: .It has the geometric interpretation of centering the cloud of data around the origin along every dimension.

Data normalization refers to normalize the data dimensions: , so that they are of approximately the same scale.isthe average for all data samples, the is the standard deviation of all the sample data.

Since in image processing, the relative scales of pixels are already approximately equal (in range from 0 to 255), so it is not strictly necessary to perform this normalization step.

3.2 Random search

Random search is a hyperparameter optimizationmethod. Since hyperparameter tuning canbegenerally described as :

$$\lambda^{(*)} = \underset{\lambda\in\Lambda}{\operatorname{argmin}}\, \mathbb{E}_{x\sim\mathcal{G}_x}\left[\mathcal{L}\left(x; \mathcal{A}_\lambda\left(\mathcal{X}^{(\text{train})}\right)\right)\right] \quad (7)$$

represents hyperparameters, is a learning algorithm base on hyperparameter , is the loss function,samples are from a natural (grand truth) distribution , is the training set，a finite set of samples from.What we really need in practice is a way to choose so as to minimize generalization error. It has been proved that randomly chosen trials are more efficient for hyper-parameter optimization than trials on gridsearch[5].

3.3 HOG and colour histogram

HOG is short for Histogram of Oriented Gradients, this feature extraction method was proposed in 2005 by Dalai for pedestriandetection[6]. In our experiments, for each

image we will compute HOG as well as a colour histogram using the hue channel in HSV colour space. We form our final feature vector for each image by concatenating the HOG and colour histogram feature vectors.

HOG captures the texture of the image while ignoring colour information, and the colour histogram captures the colour of the input image while ignoring texture. As a result, we expect that using both together ought to work better than using either alone.

4 Experiments

4.1. Environments, datasets and measurements

The hardware environments are as follow: the CPU is Intel Core i5 2.8GHz, the capacity of memory is 8GB, and the capacity of disk is 500GB.The software environments are as follow: the operating system is OSX EI Capitan,we use Python2.7.11 and installed Anaconda 2.5.0 for Python 2.7 version.Additionally, since Anaconda installedmany thirdparty libraries by default, we list some main libraries and their versions:numpy1.10.4,scipy0.17.0, matplotlib 1.5.1,pandas 0.17.1, ipython 4.0.3, ipythonnotebook4.0.4.

We choose CIFAR-10 as our experimental dataset.CIFAR-10 is a subset of Tiny Images database, which contains 10categories.It contains 60000 images, 50000 for training, 10000 for test. Its image size is 32 x32 pixels.Although the image size is small, the amount of images for each category is large, so it is very suitable for training complex models such as deep learning model.Moreover, small image size means the request for computationalcapacity is not too high, which ease the pressure on our machine.

In experiments, algorithms are compared by three evaluation standards: classification accuracy， training time and test time.

The classification accuracy is defined as:

$$accuracy = N_{y_i=y_j}/M_{y_{test}} \quad (8)$$

Itevaluatesthe classification performance of the algorithms.Theis the amount of category labels corresponding to test set. Afteralgorithms finished classifying the test images, the number of prediction label which are consistent with real category is .

Use training time and test timeto measure the computational efficiency of the algorithms. isused to measure how long it takes algorithm to learn optimal parameters in the training process.isusedto measure how long it takes algorithm to classify the test image data in the test process.

4.2 Results

Since all algorithms need hyperparameter optimization, we divide CIFAR-10 dataset into three splits: the training set, validation set and test set. The training set contains 49000 images, the validation set and test set both contain 1000 images. k-NN algorithm divides training set into 5 subsets equally, uses cross validation to performhyperparameter optimization.Other algorithms perform hyperparameter optimizationon the validation set, and theirinitial hyper parameter tuning policy is grid search. Theiteration numbers of SVM, Softmaxand Neural Network algorithmsare set to 1500.Otherhyperparameters of algorithms are obtained in the tuning process.

4.2.1 Comparison on two kinds of implementations

Numpy library provides APIs that include efficient matrix computational operations and broadcasting mechanism, it can make algorithms implemented invectorizedcomputation.Comparedwith implementationbymultiple loops, vectorizedimplementationimproved the computation efficiency. The L2 distance computations in k-NN algorithm, and the loss functionsin SVM, Softmax and Neural Network algorithms can all be implemented in loops and vectorizedcomputations.Inexperiment, we use training timeand test time tocompareoperation efficiency of the twokinds ofimplementation.We count the time in seconds. Details are in the following table:

Table 1. T_{train} and T_{test} in Two Kinds of Implementation.

	Loops	Vectorize	Loops	Vectorize
k-NN	0.000066	0.000066	4288.6	9.1
SVM	65.5	6.6	0.47	0.044
Softmax	67.1	4.7	0.63	0.059
NN	687.3	66.3	0.60	0.065

Since the training process of k-NN algorithmsimply stores the training data into memory with no matrix computationaloperation,it is not strange that k-NN takes so little time to finish it training process.However,mostcomputational operationsofk-NN algorithm are in its test process.Unlike k-NN, Most computational operationsofthe SVM, Softmax and Neural Network algorithmsare in their training process for loss values and gradientscalculation, and in their test process, the computational cost is low.

According to the table, the loop implementation form of k-NN algorithm costs nearly 420 times than the vectorizedimplementation in . Moreover,the loop implementation forms ofthe SVM, Softmax and Neural Network algorithms cost about 10 times than the loop implementation forms in .

Therefore, it is obvious that theNumpyAPIcan greatly improve the efficiency of algorithm.Inthe follow experiments, the algorithms are implemented in vectorizedforms by default.

4.2.2 Comparison on algorithms

The computational efficiency and classification accuracy of the four algorithms are compared in the following table:

Table 2. Algorithms Comparison.

k-NN	0.000066	9.1	32.7%
SVM	5.4	0.0044	31.8%
Softmax	4.8	0.0059	31.8%
NN	59.6	0.065	35.5%

The of k-NN algorithm is much short than its .On the contrary, the of SVM, Softmax and Neural Network algorithmsare much short than their . Note that the ofNeural Network algorithm is about 10 times than of SVM and Softmax algorithms.

For now, the classification accuraciesoffour algorithms are on the same level, and the performance of Neural Network algorithm is slightly better. However, there was no significant difference amongthem.The reasonisthat the SVM, Softmax and Neural Network algorithmsdid not perform data preprocessing, feature extraction and hyperparameter tuning, so the potential capacities of these algorithms have not been further explored.

4.2.3 Data preprocessing method

In the experiment, input image data normalization wasperformed.Theaccuracy improvementsoffouralgorithms are in the following table:

Table 3. Effect of Data Normalization.

k-NN	0.000066	9.1	32.7%
SVM	6.5	0.0045	33.6%
Softmax	4.7	0.0058	34.5%
NN	66.3	0.065	52.3%

From the table above, After performing datanormalization, the performance of k-NN algorithm does not improve. This is because k-NN is essentially comparing the differenceforpixel. On the contrary, the classification accuracies of SVM, Softmax and Neural Network algorithmsareimproved.The accuracy of SVM improved by 1.8%, the accuracy ofSoftmax improved by 2.7%, and the accuracyofNeural Network algorithm improved by16.8%, which is much higher than the otheralgorithms.Therefore, the presentation capacity of deep learningmodelsuch as Neural Network algorithm has been proved.

4.2.4 Hyperparameter optimization method

The 32.7% classification accuracy of k-NN algorithm is obtained by cross validation optimization method, so in this experiment, we list k-NN just for comparison convenience, it dose not use random search method, though cross validation, the hyperparameter k of k-NN algorithm is set to be 5. But the SVM, Softmax and Neural Network algorithms all take random search to get better hyperparameters.

Table 4. Accuracy ofSearch Methods.

	Gird Search	Random Search
k-NN	32.7%	32.7%
SVM	33.6%	33.2%
Softmax	34.5%	35.3%
NN	52.3%	53.3%

Through random search, the accuracy of Softmaximproved by 0.8%, and the accuracy of Neural Network algorithm improved by1%.It shows that the random searchdid find better parameters in the parameter space.However, the classification accuracy of SVM algorithmdescends by 0.4%. It can be explain that in the grid search, theparameters are already close to the optimal parameters.

Corresponding to the accuracies, the hyperparameters of the algorithms are as follow: the learning rate of SVM is 5.0e-05, regularization strength is 9.0e+04, itsiterations are 1500. The learning rate of Softmax is 4.866548e-07, regularization strength is 2.393255e+04, its iterations are 1500. The learning rate of Neural Network is 8.944667e-04, regularization strength is 9.493759e-01, its iterations are 1500, the number of layers are 2, and the number of neurons in the hidden layer are 360.

In general, the random search method is simple to implementand more efficient than grid search in hyperparameter optimization process.

4.2.5 Feature extraction method

We form our final feature vector for each image by concatenating the HOG and colour histogram feature vectors. Since k-NN algorithm simply compares the raw pixels, we did not perform the feature extraction process on it.ForSVM, Softmax and Neural Network algorithms, take the feature vectorsas the input data.The results showsaccuracy improvements in the following table:

Table 5. Effect of Feature Extraction.

	Raw pixel	Features
SVM	33.2%	42.4%
Softmax	35.3%	41.8%
NN	53.3%	59.4%

Using feature vector as input data, the classification accuracy of SVM, Softmaxand Neural Networkare all improved. The accuracy of SVM improvedby 9.2%, the accuracy ofSoftmaximprovedby 6.5%, the accuracy of Neural Network algorithm improved by 6.1%. The capacity of our combined feature extraction method has been proved.

4.3. Conclusion

Based on Anaconda, this paper implemented k-NN, SVM, Softmax and Neural Network four classical algorithms by Python, thencompared and analysed theperformance of thealgorithms from two respects: the image classification accuracy and the computational resource cost.According to the experimental results we can get the following conclusion:

• Thevectorizedimplementations of algorithms are much more efficient than the loop implementations.

• Based on vectorizedimplementation, the training time length of k-NN algorithm is the shorter than that of any other algorithms. The training time length of Neural Network algorithm is the longest. The training time lengths of SVM and Softmax are approximately the same. However, the test time lengths of SVM, Softmax and Neural Network algorithms are much shorter than that of k-NN algorithm, which means the algorithms are suitable for the practical applications in the IoT framework except k-NN algorithm.

• Theclassification accuracy of Neural Network algorithm is the highest of all, which means it should be deployed to handle the most complex classification task. The classification accuracies of SVM and Softmax are lower and approximately the same, which means they should be deployed to handle some simple but real-time required classification tasks. And the classification accuracy of k-NN algorithm is the lowest.

• Thedata normalization, random search and feature extraction are all improvement tricks of the algorithm. After applying these tricks, the classification accuracy of algorithms have been improved by different levels. The random search gives a little improvement,the effect of data normalization is better, and the effect of HOG and colour histogram feature extraction is the best of all three tricks. All the tricks should be applied to algorithms in practical application.

• k-NN algorithm is not suitable for any practical applications in the IoT framework, since its low classification accuracy and high computational cost. The performances of SVM and Softmax algorithms are approximately the same, both on their classification accuracy and computational cost. They can be deployed to some real-time required tasks to handle some specific objects classification. The classification accuracy of Neural Network algorithm is the highest of all the four algorithms, which means it can handle some complex classification tasks. However, the computational cost of the Neural Network algorithm is huge, therefore, it should be deployed to the server and provide classification API to the remote devices.

In the future, we will implement the algorithms by using the TensorFlow framework, which is provided by Google, and deploy algorithms to different parts of the IoT system, measure the performance of the whole system and the algorithms respectively.

References

1. Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[C]//Advances in neural information processing systems. 2012: 1097-1105.
2. He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[J]. arXiv preprint arXiv:1512.03385, 2015.
3. Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real-time object detection[J]. arXiv preprint arXiv:1506.02640, 2015.
4. Cybenko G. Approximation by superpositions of a sigmoidal function[J]. Mathematics of control, signals and systems, 1989, 2(4): 303-314.
5. Bergstra J, Bengio Y. Random search for hyper-parameter optimization[J]. Journal of Machine Learning Research, 2012, 13(Feb): 281-305.
6. Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]//2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05). IEEE, 2005, 1: 886-893.