Iterative SVM for Tiny Image Classification

CHEN Xinxi, QIAN Xin

Abstract

Given the classification task on semi-supervised sub-dataset of CIFAR-10, several prevailing algorithms were examined. Starting from a simple tuned polynomial Kernel SVM on labeled data with 50.0% accuracy, we then move on to add GIST descriptor before training SVM to get a 59.0% accuracy. Our third and final model incorporates Bag-of-Words with a L2 linear SVM classifiers, either with or without self-training. The final accuracy on public data set ranges from 68.00% to 70.67%.

Index Terms

Semi-supervised, SVM, GIST, K-means, Bag of Visual Words.

I. Introduction

On receiving the assignment description, we are amazed to see how ”tiny” a 32x32 pixels image could be. Even with our eyes, sometimes it is hard to reliably identify the exact object in it. However, intuition told us that we should feel confident that our trained computer could do better (not only as accurate as human, but both efficient and acceptably accurate) than us (which is accurate but way too inefficient!).

Provided with 3000 labeled images, 31800 unlabeled images and 1200 test images, a so-called ”semi-supervised data set”. We considered first supervised learning on simple SVM, followed with adding more techniques to a more effective SVM,

- Pre-processing: Normalization and Whitening
- Feature Extraction

Instead of discarding the ”free” unlabelled data, we then considered unsupervised learning, specifically, K-means clustering or i.e. Bag of Visual Words, to make a more general model.

Results turns out such a combination of supervised learning and unsupervised learning is a good choice.

CHEN Xinxi, QIAN Xin
May 05, 2015

II. Method

A. Support Vector Machine(SVM)

In our data set, the number of features is n=32*32*3=3072, while the valid training set is m=300, i.e. n is large relative to m. Either use logistic regrression, or SVM without a kernel (”the linear kernel”). Also there are indeed many SVM libraris with multi-class classification options. Our choice is LIBLINEAR. After tuning parameters, our first attempt results in 50.00%accuracy.

B. Feature Extraction on Gist

With starter code, we visualized several images to get some new ideas and discovered that nearly all of images has two parts, the foreground object and the background scene. And there is a strong correlation between scene category and the object category. e.g. Birds are likely to fly on the sky while automobiles are likely to drive on the highways.

Searching through the Internet, Gist is a good model of the scene recognition of scenes that bypasses the segmentation and the processing of individual objects or regions.[1]

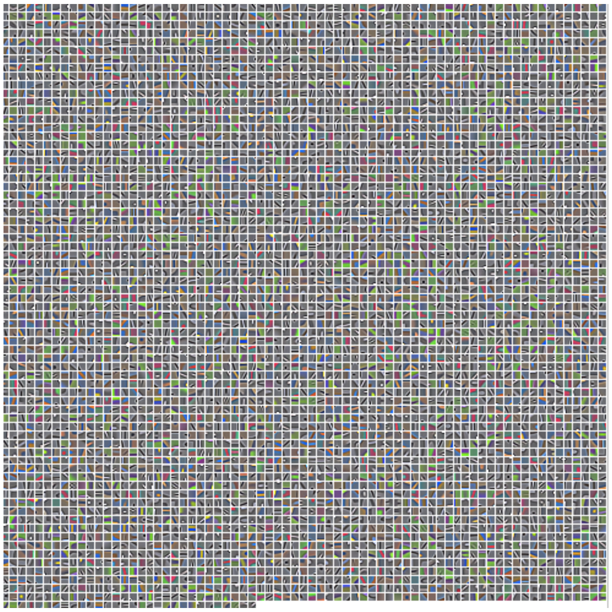
Applying Gist before our simple LIBLINEAR SVM, every image obtains a 512-dimension vector as its feature descriptor. This time our accuracy grows up to 59.00%.

We move on to ask the next question. Instead of Gist scene recognition, what about other more expressive feature extraction techniques?

C. Bag of Visual Words

We got this idea from our lecture slides. During the lecture, Bag of Words was introduced as techniques for Information Retrieval, e.g. Spam Email Detection. After thorough research, we are excited to learn that there is an analogy in Computer Vision field called ”Bag of Visual Words”. Details of BOW[4] are omitted here.

In short, our implementation chose patch size w=6, with normalization(zero mean, unit variance) and whitening. The feature mapping is learned using soft K-means clustering with K=1600, 2000 and 5000. Further pooling was also used to reduce the dimensionality of the resulting feature space.



K-means, 5000 centroids with whitening

D. Pre-processing of Image Data

We discovered the urgent need of pre-processing by playing a small trick on searching our sub-CIFAR10 data set among the original super CIFAR10 data set. Amazing result shows that not a single one data set is found in the original CIFAR10. Therefore it is valid to assume that conversion was made from original CIFAR10 set to get our sub-CIFAR10 set, e.g. adjusting the brightness, mirroring the image, etc.

E. Self-training

After obtaining the 68% accuracy, we were still wondering what else could be done. Our answer is, fully utilize the unlabeled data. Anyway, those unlabeled data is cheap but of low quality. We have used them once before, to build the visual word ”dictionary”. What about twice?

Here we labelled/predict the unlabeled data using the our SVM model. This magic turns the unlabeled data to roughly labeled data. We than repeat to train a second SVM model. Since there could be much mis-labeled noise, we trained the second SVM model with C=1 to reduce overfitting.

III. Results

Best performance: K-means clustering of 6,000,000 patches from labeled data and unlabeled data with K=5,000 + Normalization + Whitening + Self-training = 70.67% on Kaggle public data set.

TABLE I
RESULT ACCURACY

Method description	Leaderboard Accuracy	Training Cost
pure linear SVM	50.00%	10 mins
GIST + linear SVM	59.33%	15 mins
K-means of 400,000 patches from labeled data with K=1600	68.00%	25 mins
K-means of 500,000 patches from labeled data and unlabeled data with K=1600	65.33%	1hrs
6,000,000 patches from labeled and unlabeled, K=5000, C=10	70.00%	3hrs
6,000,000 patches from labeled and unlabeled, K=5000, C=1	70.00%	1.5hrs
6,000,000 patches from labeled and unlabeled, K=5000, C=10 then self-training C=1	70.67%	overnight

IV. CONCLUSIONS

We made our model effective mainly through,

- Pre-processing(Normalization and Whitening), which make the data input largely unaffected by position, transformation, mirroring, etc. i.e. the input data are more general
- Feature Extraction 1: a larger K in K-means means more features/visual words, a large ”dictionary” and a larger number of Patches (from both labeled data and unlabeled data) means more representative the centroids are
- Feature Extraction 2: The number of patches should match the number of training set. 400,000 patches extracted from 3000 data or 6,000,000 patches extracted from 31800+3000 data results in good. But 500,000 patches for 31800+3000 data are not enough to fully conclude all the features in this large set of images.
- An adequate C on training SVM, which means acceptable training convergence and less overfitting, preferably, preferablly around 1 to 10

V. FURTHER THOUGHTS

A. Drawbacks of BOW

Recall in lecture 8, the text-base BOW model ignores word. Same for visual word, the model ignores the positions of visual words in image and classifies simply based on a histogram of the frequency of visual words.

B. Other BOW representations

We are expecting to test other BOW representations, such as SIFT, RGB-D Kernel Descriptors or Sparse auto-encoders and compare their performance.

C. More precise on C

We have tested on C=1, 10 and 100. What about other value? We could validate the parameter C by separating the 3000 labeled training data into 1800 training data and 1200 validation data. Based on the accuracy, choose the optimized C.

REFERENCES

[1] Modeling the shape of the scene: a holistic representation of the spatial envelope http://people.csail.mit.edu/torralba/code/spatialenvelope/

[2] Scene recognition with bag of words; CS 143: Introduction to Computer Vision http://cs.brown.edu/courses/cs143/proj3/

[3] Bag of visual words model: recognizing object categories

[4] Discriminative Training for Object Recognition Using Image Patches

[5] An Analysis of Single-Layer Networks in Unsupervised Feature Learning
http://ai.stanford.edu/ ang/papers/nipsdlufl10-AnalysisSingleLayerUnsupervisedFeatureLearning.pdf

[6] Analysis of Single-Layer Networks Presented by Hourieh Fakourfar

[7] The CIFAR-10 dataset http://www.cs.toronto.edu/ kriz/cifar.html