

The Final Project of “introduction to Statistical Learning and Machine Learning”

Yanwei Fu

December 2, 2018

Abstract

(1) This is the final project of our course. The project is released on Dec 25th, 2018. The deadline is 5:00pm, Feb. 7th, 2018. Please send the report to sunqiang85@gmail.com. The late submission is also acceptable; however, you will be penalized 10% of total scores for EVERY TWO DAYS' delay (by 5:00pm of that day).

(2) Note that if you are not satisfied with the initial report, the updated report will also be acceptable given the necessary score penalty of late submission.

(3) OK! That's all. Please let me know if you have any additional doubts of this project. Enjoy!

Note that:

(a) If the size of training instances are too large, you may want to apply some sampling techniques to extract a small portion of training data.

(b) If you think the dimension of features is too high, you may also want to use some techniques to do feature dimension reduction, such as PCA, KPCA, ISOMAP.

(c) The referring papers are listed as an introduction to the context of problems. It's not necessarily to exactly implement these papers, which actually is not an easy task.

(d) **For all the projects, we DO care about the performance on each dataset with the correct evaluation settings.**

Merry Christmas and Happy New Year!

1 Introduction

1.1 Collaboration Policy

You are allowed to work in a group with at most one collaborators. you will be graded on the creativity of your solutions, and the clarity with which you are able to explain them. If your solution does not live up to your expectations, then you should explain why and provide some ideas on how to improve it. You are free to use any third-party ideas or codes that you wish as long as it is publicly available. You must provide references to any work that is not your own in the write-up.

1.2 Writing Policy

Final project (20%) is finished by one team. Each team should have up to 2 students; and will solve a real-world Big-Data problem. The final report should be written in English. The main components of the report will cover

1. Introduction to background and potential applications (2%);

2. Review of the state-of-the-art (3%);
3. Algorithms and critical codes in a nutshell (10%);
4. Experimental analysis and discussion of proposed methodology (5%).

Please refer to our latex example: http://yanweifu.github.io/courses/SLML/chap5/IEEE_TAC_2016.zip .

1.3 Submitting Policy

The paper must be in NIPS format (downloadable from ¹) and it must be double-blind. That is, you are not allowed to write your name on it etc. For more info, please read: NIPS revieweng and double blind policy.

Package your code and a copy of the write-up pdf document into a zip or tar.gz file called finalProject-*your-student-id1_student-id2_student-id3.[zip|tar.gz]. Also include functions and scripts that you had used. Send to TA. In the submission email, you should well explain the authours and co-workers of this project. The TAs can know the names of your works.

1.4 Evaluation of Final Projects

We will review the papers **anonymously (double blind)**. Specifically, when reading and evaluating the submitted paper, we donot know who you are. By virtue of such a way, we donot bring any prior bias into the whole evaluation process.

We will review your work on the following NIPS criteria:

Overview: you should briefly summarize the main content of this paper, as well as the Pros and Cons (advantages and disadvantage) in general. This part aims at showing that you had read and at least understand this paper.

Quality: Is the paper technically sound? Are claims well-supported by theoretical analysis or experimental results? Is this a complete piece of work, or merely a position paper? Are the authors careful (and honest) about evaluating both the strengths and weaknesses of the work?

Clarity: Is the paper clearly written? Is it well-organized? (If not, feel free to make suggestions to improve the manuscript.) Does it adequately inform the reader? (A superbly written paper provides enough information for the expert reader to reproduce its results.)

Originality: Are the problems or approaches new? Is this a novel combination of familiar techniques? Is it clear how this work differs from previous contributions? Is related work adequately referenced?

Significance: Are the results important? Are other people (practitioners or researchers) likely to use these ideas or build on them? Does the paper address a difficult problem in a better way than previous research? Does it advance the state of the art in a demonstrable way? Does it provide unique data, unique conclusions on existing data, or a unique theoretical or pragmatic approach?

¹<https://nips.cc/Conferences/2016/PaperInformation/StyleFiles>

1.4.1 Minimum Requirements

For all the projects listed below, in general you should devise your own machine learning algorithms which target at each specific problem of each project. You should compare with the machine learning algorithms taught in this course/mini-projects, which, include but not limited to, *linear regression/classification, K-NN/NN, logistic regression, linear/RBF kernel SVM, Neural network* as well as *tree-based methods*. Thus, the minimum requirements, as you can image, just apply and compare with these methods; and explain the advantage and disadvantage of using these methods for the project problem. Note that, your algorithms can be derived from one of these machine learning algorithms; and feel free to use any machine learning package you like.

2 Potential Projects

2.1 One-shot Learner

2.1.1 Introduction to this project

The success of recent machine learning (especially the deep learning) greatly relies on the training process on hundreds or thousands of labelled training instances of each class. However in practice, it might be extremely expensive or infeasible to obtain many labelled data, e.g. for objects in dangerous environment with limited access. On the other hand, human can recognize an object category easily with only a few shots of training examples. Inspired by such an ability of humans, one-shot learning aims at building classifiers from a few or even a single example. One-shot learning means that only one or very few training examples are available to train classifiers. Of course, the major obstacle of learning good classifiers in one-shot learning setting is the lack of enough training data.

Existing one-shot learning approaches can be divided into two groups: the direct supervised learning based approaches and the transfer learning based approaches.

- **Direct Supervised Learning-based Approaches:** Early approaches do not assume that there exist a set of auxiliary classes which are related and/or have ample training samples whereby transferable knowledge can be extracted to compensate for the lack of training samples. Instead, the target classes are used to trained a standard classifier using supervised learning. The simplest method is to employ nonparametric models such as kNN which are not restricted by the number of training samples. However, without any learning, the distance metric used for kNN is often inaccurate. To overcome this problem, metric embedding can be learned and then used for kNN classification. Other approaches attempt to synthesize more training samples to augment the small training dataset [11]. However, without knowledge transfer from other classes, the performance of direct supervised learning based approaches is typically weak. Importantly, these models cannot meet the requirement of lifelong learning, that is, when new unseen classes are added, the learned classifier should still be able to recognize the seen existing classes.
- **Transfer Learning-based One-shot Recognition:** This category of approaches follow a similar setting to zero-shot learning, that is, they assume that an auxiliary set of training data from different classes exist. They explore the paradigm of learning to learn [15] or meta-learning [9] and aim to transfer knowledge from the auxiliary dataset to the target dataset with one or few examples per class. These approaches differ in (i) what knowledge is transferred and (ii) how the knowledge is represented. Specifically, the knowledge can be extracted and shared in the form of model prior in a generative model [4], features [8], and semantic attributes [7, 12, 14]. Many of these approaches take a similar strategy as the existing zero-shot learning approaches and transfer knowledge via a shared embedding space. Embedding space can typically be formulated using neural networks (*e.g.*, siamese network [10]), discriminative classifiers (*e.g.*, Support Vector Regressors (SVR) [3, 12]),

or kernel embedding [8] methods. Particularly, one of most common embedding ways is semantic embedding which is normally explored by projecting the visual features and semantic entities into a common *new* space. Such projections can take various forms with corresponding loss functions, such as SJE [2], WSABIE [16], ALE [1], DeVISE [5], and CCA [6].

We use mini-imagenet dataset for this task. And the dataset of this project downloaded from our webpage².

As for the evaluation metrics, the basic experimental protocol is on “Transfer Learning-based One-shot Recognition”, which should be strictly following Sec. 2 in [13]. USING WRONG EVALUATION METRICS WILL LEAD TO LOWER SCORE. Also note that, your work should be compared against several baselines. We do care the performance and accuracy reported.

You can also try the experimental protocol on “Direct Supervised Learning-based Approaches”. Successful approaches on this aspect are very encouraging. Note that, still you need to compare several baselines.

2.1.2 Submission and Evaluation

Note that, you should not copy any sentences from any paper. Remember the definition of plagiarism.

2.2 Image Captioning

Given an image, the task of image captioning is trying to give a sentence to describe this image. In this task, we use the Flickr30K as the testbed for image captioning tasks.

You can find the Flickr30k dataset here³: there two files and two folders in it. The folder “images” is the original images of Flickr30k The folder “resnet101_fea” contains the fc_fea and conv_fea of Flickr30k. There are full-layer features and convolutional features of images respectively. The file “dic_flickr30k.json” is the prepared vocabulary of Flickr30k and the training and testing split information. The file “cap_flickr30k.json” is the captions of images.

2.2.1 Questions and Evaluation

As for this task, we ask several questions to inspire your work:

1. Can you exploit a model that can generate a sentence given an image? The generated sentence must be related to the given image.
2. Can you add Attention mechanism in you captioning model?

Minimum Requirements:

1. The captioning generation model is a requirement in Question-1.
2. The Attention mechanism is an extra point column.

Notes and Evaluation:

1. You can also extract any image features if you want; we provide some good features which we use to submit a work to top-tie conference.
2. You can evaluate the quality of generated sentenced by using the online script: <https://github.com/tylin/coco-caption>. Note that they are the only objective metrics that we trust. We mainly test your sentence by the CIDEr metric.

²http://www.sdspeople.fudan.edu.cn/fuyanwei/course/projects/final_project/mini-imagenet.tar.gz

³http://www.sdspeople.fudan.edu.cn/fuyanwei/course/projects/final_project/DATASET_Flickr30k

3. We know there are lots of open source codes and models for image captioning, and we know most of them. So we expect your model; and NO CHEATING here. If your model is modified from another model, please discuss the similarity and difference between your model and the referring model. The “*Originality*” is one evaluation metric.

2.3 Learning to Score Figure Skating Sport Videos

We have one project of learning to predict the scores of figure skating sport video. Please read our arxiv paper: Learning to Score Figure Skating Sport Videos (<https://arxiv.org/pdf/1802.02774v3.pdf>). We have the Fis-V dataset is download-able from here ⁴. We upload the training and testing videos on the website. Please read our paper how to evaluate the performance of the model.

Note that, if you have achieved significantly better results over the values that we reported in our arxiv manuscript, we would also like to give you higher final scores of the course (most probably “A”).

2.4 Other projects.

You can also try other projects. However, please let TA and me know first in order to get the approval. Note that if the project is too easy, it would affect your scores of final projects.

References

- [1] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for attribute-based classification. In *CVPR*, 2013.
- [2] Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In *CVPR*, 2015.
- [3] Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. Describing objects by their attributes. In *CVPR*, 2009.
- [4] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE TPAMI*, 2006.
- [5] Andrea Frome, Greg S. Corrado, Jon Shlens, Samy Bengio, Jeffrey Dean, Marc’Aurelio Ranzato, and Tomas Mikolov. DeViSE: A deep visual-semantic embedding model. In *NIPS*, 2013.
- [6] Yanwei Fu, Timothy M. Hospedales, Tao Xiang, Zhengyong Fu, and Shaogang Gong. Transductive multi-view embedding for zero-shot recognition and annotation. In *ECCV*, 2014.
- [7] Yanwei Fu, Timothy M. Hospedales, Tao Xiang, and Shaogang Gong. Learning multi-modal latent attributes. *IEEE TPAMI*, 2013.
- [8] T. Hertz, A.B. Hillel, and D. Weinshall. Learning a kernel function for classification with small training samples. In *ICML*, 2016.
- [9] Ricardo J Vilalta and Youssef. Drissi. A perspective view and survey of meta-learning. *Artificial intelligence review*, 2002.

⁴http://www.sdspeople.fudan.edu.cn/fuyanwei/course/projects/final_project/figure_skating

- [10] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML – Deep Learning Workshok*, 2015.
- [11] Brenden M. Lake and Ruslan Salakhutdinov. One-shot learning by inverting a compositional causal process. In *NIPS*, 2013.
- [12] Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE TPAMI*, 2013.
- [13] S. Ravi and H. Larochelle. Optimization as a model for few-shot learning. In *ICLR*. 2017.
- [14] Marcus Rohrbach, Sandra Ebert, and Bernt Schiele. Transfer learning in a transductive setting. In *NIPS*, 2013.
- [15] S. Thrun. *Learning To Learn: Introduction*. Kluwer Academic Publishers, 1996.
- [16] Jason Weston, Samy Bengio, and Nicolas Usunier. Wsabie: Scaling up to large vocabulary image annotation. In *IJCAI*, 2011.