Available online at www.sciencedirect.com
ScienceDirect

The Journal of China Universities of Posts and Telecommunications

December 2016, 23(6): 1–7
www.sciencedirect.com/science/journal/10058885

http://jcupt.bupt.edu.cn

Progressive framework for deep neural networks: from linear to non-linear

Shao Jie[1] (✉), Zhao Zhicheng[1,2], Su Fei[1,2], Cai Anni[1]

1. School of Information and Communication Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China
2. Beijing Key Laboratory of Network System and Network Culture, Beijing University of Posts and Telecommunications, Beijing 100876, China

Abstract

We propose a novel progressive framework to optimize deep neural networks. The idea is to try to combine the stability of linear methods and the ability of learning complex and abstract internal representations of deep learning methods. We insert a linear loss layer between the input layer and the first hidden non-linear layer of a traditional deep model. The loss objective for optimization is a weighted sum of linear loss of the added new layer and non-linear loss of the last output layer. We modify the model structure of deep canonical correlation analysis (DCCA), i.e., adding a third semantic view to regularize text and image pairs and embedding the structure into our framework, for cross-modal retrieval tasks such as text-to-image search and image-to-text search. The experimental results show the performance of the modified model is better than similar state-of-art approaches on a dataset of National University of Singapore (NUS-WIDE). To validate the generalization ability of our framework, we apply our framework to RankNet, a ranking model optimized by stochastic gradient descent. Our method outperforms RankNet and converges more quickly, which indicates our progressive framework could provide a better and faster solution for deep neural networks.

Keywords framework, neural network, DCCA, semantic, RankNet

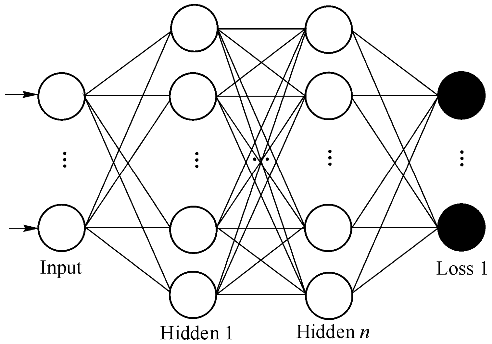
1 Introduction

Deep learning has attracted great attention recently and yielded state-of-the-art performances in multiple computer vision tasks. Deep models use a cascade of multiple layers of nonlinear processing units for feature extraction, and possess greater representation power than traditional shallow models. Compared with the unstable local optima of deep models, though traditional linear methods may not model non-linear statistical properties of real data well, they can provide a more stable output. In this paper, we combine the stability of linear methods and the ability of learning complex and abstract internal representations of deep learning methods, and propose a novel progressive framework.

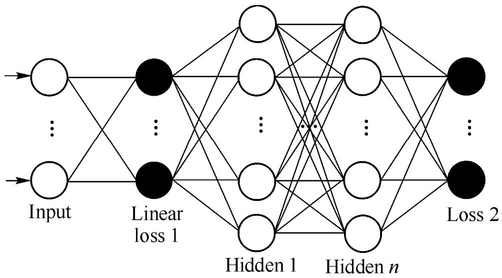
The end-to-end principle is a classic design principle in traditional neural networks. As shown in Fig. 1(a), traditional feedforward networks consist of a series of layers. The final layer produces the network's output. The proposed progressive framework for deep learning methods is shown in Fig. 1(b). A new loss layer with linear activation function is added after the input layer.

Based on the proposed framework, we conduct experiments for cross-modal retrieval task. The experimental results show the performance of our method is better than similar state-of-the-art approaches on publicly available dataset NUS-WIDE [1]. To validate whether our framework could be used for other applications, we conduct experiments on RankNet [2], a ranking model optimized by neural networks. Experimental results indicate that our method outperforms RankNet and converges more quickly.

Received date: 08-09-2016
Corresponding author: Shao Jie, E-mail: Shaojielyg@163.com
DOI: 10.1016/S1005-8885(16)60063-8



(a) Traditional framework



(b) The proposed framework

Fig. 1 Structure of traditional framework and our proposed framework

2 Related work

1) Cross-modal retrieval

Rasiwasia et al. [3] proposed correlation matching to map the features of images and texts into a common latent space using canonical correlation analysis (CCA). Akaho [4] used kernel method to improve linear CCA. Andrew et al. [5] built a deep architecture DCCA to learn complex nonlinear transformations of two data views such that the resulting representations are highly linear correlation. DCCA can be viewed as a nonlinear extension of traditional linear CCA. Though the representation power is improved, DCCA is easy to over-fitting, especially when the datasets are not big enough. Srivastava et al. [6] proposed to learn a good generative model of the joint space of image and text using deep belief network (DBN). DBN consists of multiple stacked restricted Boltzmann machine (RBM). Gaussian RBM [7] and replicated softmax RBM [8] are used to model the real-valued feature vectors for image and the discrete sparse word count vectors for text, respectively. Based on DBN, Feng et al. [9] proposed to learn a common space of image and text by correspondence autoencoder (Corr-AE). Corr-AE defines a novel optimal objective, which minimizes a linear combination of representation learning errors for each modality and correlation learning error between hidden representations of two modalities.

2) Learning to rank

Typical approaches for learning to rank can be divided into pair-wise methods [10,2] and list-wise methods [11–12]. Pair-wise methods predict the relative order of paired documents. List-wise methods minimize a permutational loss function and try to optimize the evaluation metric directly. RankNet, which employ cross entropy loss function with gradient descent algorithm to train a neural network model, is a feasible choice for validating our framework.

3 Framework

The proposed framework differs with the traditional one on three aspects:

1) Architecture

A loss layer is added after the input layer. Neurons in the new layer are activated by linear functions.

2) Loss function

During the forward phase, the optimization objective is a weighted sum of three parameters: linear loss of the first loss layer, non-linear loss of the last output layer and a regularization penalty, as shown in Eq. (1).

$$J = \lambda_1 L + \lambda_2 L^{\mathrm{N}} + \lambda_3 ||\theta||_{\mathrm{reg}} \tag{1}$$

L and L^{N} denote linear loss and non-linear loss respectively. It is necessary to choose appropriate values for λ_1 and λ_2. For choosing λ_1 and λ_2, we usually set one parameter to 1 and optimize the other one by performing grid search with predefined candidates. In our experiments, the parameters λ_1 and λ_2 are not very sensitive and we set them to 1.

3) Gradient computation

During the back-propagate phase, δ_1, which represents the error term for neurons in the first loss layer, is the sum of error back-propagated from the last output layer and error produced from its own output:

$$\delta_1 = \frac{\lambda_2 \partial L^{\mathrm{N}}}{\partial a_1} + \frac{\lambda_1 \partial L}{\partial a_1} \tag{2}$$

where a_1 is the output of the first loss layer. The former error term can be calculated from the error δ_2 produced from the output of the second loss layer through chain rule.

$$\delta_2 = \frac{\lambda_2 \partial L^{\mathrm{N}}}{\partial a_2} \tag{3}$$

where a_2 is the output of the second loss layer. The ways to calculate $\partial L^{\mathrm{N}} / \partial a_2$ and $\partial L / \partial a_1$ are the same. Given