

一种基于模拟退火算法改进的卷积神经网络

满凤环,陈秀宏,何佳佳

(江南大学 数字媒体学院, 江苏 无锡 214122)

摘 要: 卷积神经网络(CNN)是一种深度学习技术,其目标是构建一个接近人类的智能来执行任何知识的任务.使用基于交叉熵损失函数的模拟退火算法对 dropout 正则化,即有些单元值被抑制的 CNN 进行优化训练,而在测试阶段所使用的模型平均方法则同时考虑了保留概率和池化区域内单元值所占概率.在 MNIST 手写数据库和 CMU-PIE 的部分图像库的实验结果表明,在同一 CNN 结构和相同迭代次数时,此方法均优于其他方法,不仅能够得到更好的识别率而且更能防止过拟合的问题.

关键词: 卷积神经网络;模拟退火算法;交叉熵;图像识别

中图分类号: TP183

文献标识码: A

文章编号: 1000-7180(2017)09-0058-05

An Improved Convolutional Neural Network Based on Simulated Annealing Algorithm

MAN Feng-huan, CHEN Xiu-hong, HE Jia-jia

(School of Digital Media, Jiangnan University, Wuxi 214122, China)

Abstract: CNN is a deep learning method, where its aim is to the goal of artificial intelligence that could perform any intellectual tasks. This paper proposes to optimize CNN with dropout regularization that it can some units became zero using simulated annealing algorithm based on cross entropy loss function, also proposes model averaging method that it combines retaining probability and probability of each unit within pooling region at test time. We do experiments on MNIST handwritten database and part of CMU - PIE database. Under the same structure and the same number of iterations, the method is superior to other methods. It can get a better recognition rate and can be better to overcome the over-fitting problem.

Key words: convolution neural networks; simulated annealing algorithm; cross entropy; image recognition

1 引言

深度学习中的一个主要模型就是卷积神经网络(CNN),CNN 包括卷积层和池化层,其中卷积层中不同的卷积核可用来提取不同的图像特征,池化层用来降低特征的维数.现代启发式算法主要解决科学、工程等方面的优化问题,将其用于优化 DL 的研究成果还很少,You 和 Pu^[1]将遗传算法与 CNN 相结合,通过遗传算法的重组和突变来改变 CNN 的特性,实现对 CNN 的卷积层中权值和偏置的调节. Rosa^[2]利用和声搜索对 CNN 进行微调.

本文将模拟退火算法(simulated annealing, SA)用于优化 CNN,主要目的是用 SA 算法随机地寻找 CNN 最后一层关于损失函数的最优解,然后利用该解更新前面层的权值和偏置,这里 CNN 结构中的损失函数采用交叉熵函数.另外,CNN 结构还使用了 dropout 正则化方法,以便能有效地减少过度拟合的影响.

2 相关知识

2.1 卷积神经网络

文献[3]提出的 CNN 结构是一种基于局部感受

域、权值共享和 subsampling(二次抽样)等传统多层感知器(MLP)的人工神经网络,其基本结构包括卷积层和池化层,如图1所示,其中C1和C3层是卷积层,S2和S4是池化层,用于分类的F5和F6是全连接层。

卷积层中每个神经元的输入与前一层的局部接受域相连,并提取该局部的特征。对同一输入图像卷积核可以是不同的,而对不同输入图像卷积核则是相同的,所以权值是共享的。卷积层的每一个特征图是由上一层特征图通过相应的卷积核卷积加上相对应的偏置,通过一个非线性激活函数而得到。例如,输入 x 、权值 W^k 和偏置 b^k ,经激活函数sigmoid激活后的第 k 个特征图 a^k 为

$$a^k = \text{sigm}(W^k x + b^k) \quad (1)$$

采用sigmoid非线性函数可能会使梯度消失,而且它是指数运算,在反向传播求导时涉及除法,计算量相对较大。因此本文采用Relu(rectified linear units)^[4]激活函数,减少计算量。Relu能使一部分神经元的输出为0,这样可得到稀疏的网络,且能减少参数的相互依存关系,缓解过拟合问题的发生。ReLU激活函数为

$$a^k = \max(W^k x + b^k, 0) \quad (2)$$

在池化层,需要计算特征图一个区域上某个特定特征的平均值(或最大值)。这些池化不仅可以降低特征维度(相比使用所有提取到的特征),而且保持了空间的不变性,同时还会改善结果(不容易过拟合)。例如图1中C1层的每个特征图大小是 24×24 ,经过S2池化层后大小变为 12×12 。

经过几次卷积和池化交替操作,之后连接的全连接层,分类器是一层或者几层的全连接层组成,输出的一个神经元代表一类,以MNIST数据集为例,输出层包含十个神经元对应的数据库的十类。

2.2 模拟退火算法

模拟退火算法(SA)是一种随机搜索全局最优解的方法,具有渐近收敛、并行、鲁棒等特性,并具有灵活、容易实现以及运算速度较快等优点^[5]。Kirkpatrick^[6]等将退火思想引入到组合优化领域。由于在搜索过程中引进了随机因素,以一定概率接受一个比当前解要差的解,而该概率随时间推移逐渐降低,因此该方法的实施有可能跳出局部最优解而最终求得全局最优解,这里的概率定义为

$$p = \exp\left(\frac{-\Delta E}{kT}\right) \quad (3)$$

式中, ΔE 能量差; k 是波尔兹曼常数; T 是控制退火过程的温度。SA优化的标准过程可描述如下。

第一步:由一个产生函数从当前解产生一个新解 S' 。通常采用由当前解经过简单的变换而产生新解的方法。

第二步:初始化一个温度。如果温度设置过高,需要长时间才能达到收敛,而过小的话会导致系统错失全局最佳。

第三步:计算与新解所对应的目标函数差。

第四步:判断新解是否被接受。

第五步:降低温度。在搜索过程中,温度也随之降低。

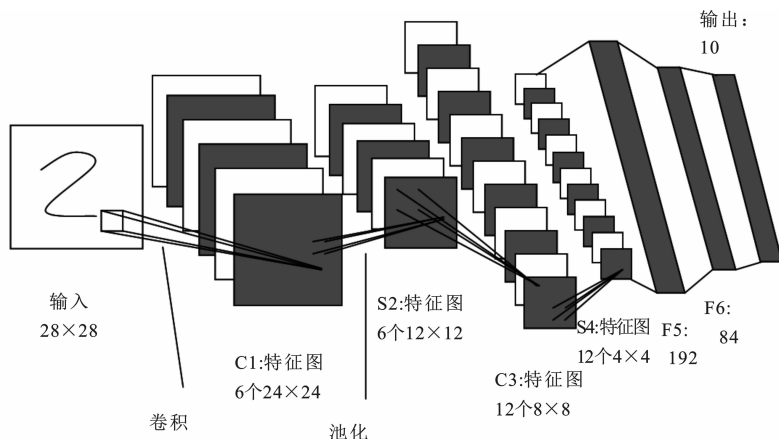


图1 LeCun等人提出的CNN结构

第六步:停止或者进入下一轮迭代。当终止条件满足时就停止;当不满足时则重复第一步到第六步。

3 结合SA的CNN模型

本文提出的CNN结构参考了简单的CNN

(LeNet-5),如图1所示。首先使用了以下两种结构进行简单的验证,一种结构表示为i-6c-2s-12c-2s即C1层有6个特征图,C3层有12个;另一种为i-8c-2s-16c-2s,所有卷积层卷积核的大小是 5×5 ,池化区域的大小是 2×2 。此结构用于MNIST手写数字

数据集的识别.

以下将 SA 用于训练 CNN 得到满足条件的最优解. 目标函数为

$$Q = -\frac{1}{N} \sum_{n=1}^N (y^n \log(o^n) + (1 - y^n) \log(1 - o^n)) \quad (4)$$

或者训练集上的标准错误. 其中, y 表示期望的输出; o 代表实际的输出; N 表示一次训练的样本数. 训练过程终止准则有两种方式: 一是达到设置的最大迭代次数; 二是当损失函数小于某一常数. 满足任意一个准则意味着达到了最优的状态.

训练 CNN 可得到卷积核和全连接层的权值和偏置, 而最后一层得到的权值和偏置经过激活后再用于计算损失函数. 将最后一层的权值和偏置看作一个解向量 S , 再用 SA 算法进行优化, 这里随机产生的新解是在原解上随机加 ΔS . 判断新解是否被接受, 最常用的接受准则是 Metropolis 准则: 若 S' 的目标函数值小于原解 S 的目标函数值, 则接受 S' 并作为新的当前解 S ; 否则, 以概率 p 接受 S' 作为新的当前解 S . 而当新解 S' 被判定为舍弃时, 则在当前解 S 不变. 算法过程如下:

```

初始化和设置 CNN, 如 i-6c-2s-12c-2s;
前向传播得到权值  $W$  和偏置  $b$ , 并计算损失函数  $Q$ ;
while 不满足终止条件 do
 $S' = S + \Delta S, Q(S')$ ;
If  $Q(S') \leq Q(S)$ 
 $S \leftarrow S'$ ;
else if  $p > \text{rand}(1, 1)$ 
 $S \leftarrow S'$ ;
end
end
降低温度:  $T = c \times T$ ;
BP 更新其他层的权值和偏置;
end

```

算法中 ΔS 是一个重要的因素, 适当地选取其值可显著地降低错误识别率. 例如, 使用前面提到的两种结构迭代一次, 如果 $\Delta S = 0.000\ 01 \times \text{rand}$, 则为 3.06%, 而如果 $\Delta S = 0.000\ 05 \times \text{rand}$, 则错误率为 3.33%.

训练 CNN 结构的方法是使用随机梯度下降的 BP 算法更新其他层的权值和偏置. 在每次迭代中, 优化损失函数并求其最小值, 第 i 次迭代后第 k 个权值矩阵的第 $i+1$ 次更新准则为

$$\Delta w_k^{(i+1)} = m \cdot \Delta w_k^{(i)} + \lambda \cdot E\left(\frac{\partial Q}{\partial w_k^{(i)}}\right) \quad (5)$$

$$w_k^{(i+1)} = w_k^{(i)} + \Delta w_k^{(i+1)} \quad (6)$$

式中, m 是动量参数; λ 表示学习率; $E\left(\frac{\partial Q}{\partial w_k^{(i)}}\right)$ 是样本第 i 次迭代时损失函数 Q 关于第 k 个权值的偏导数的平均值.

训练阶段, 本文 CNN 中的池化层和全连接层使用 dropout 正则化来抑制一部分激活单元. 池化传统的方法有求平均和求最大值, 求平均是以同样的贡献考虑池化区域的所有激活值, 这样做可能淡化大的激活值的作用; 求最大值则只取每个池化区域的最大值, 忽略了区域内其他的单元值的作用. 使用 dropout 可以避免以上缺点, 因为 dropout 可能会使有些神经元的值被抑制, 池化得到的值就有了随机性, 不再总是池化区域内所有值的平均或者其中的最大值. 训练时每次 dropout 后输入都是不同的, 可以在相同时间和相同数据的情况下训练更多的模型.

此外, 训练阶段使用 dropout 后, 测试阶段需要一种模型平均的方法以便能得到接近所有模型的平均值, 文献[7]提出了模型预测平均方法, 得到第 $l+1$ 层第 k 个特征图中第 j 个池化区域池化的结果, 表达式为

$$a_j^{l+1,k} = \sum_{i=0}^n q_{j,i} a_{j,i}^{l,k} = \sum_{i=1}^n q_{j,i} a_{j,i}^{l,k}, i = 1, \dots, n \quad (7)$$

式中, n 是一个池化区域的大小; $i=0$ 是池化区域全被 dropout 的情况; $q_{j,i} = q(1-q)^{n-i}$ 表示 $a_{j,i}^{l,k}$ 内第 i 个数据被保留的概率. 文献[8]提出的随机池 (stochastic pooling) 正则化方法, 作者在训练阶段用一种随机取值的方法代了传统的池化方法, 而在测试阶段, 计算在池化区域内每个单元值 $a_{j,i}^{l,k}$ 所占的概率 $p_{j,i}$:

$$p_{j,i} = \frac{a_{j,i}^{l,k}}{\sum_{i=1}^n a_{j,i}^{l,k}} \quad (8)$$

从而测试阶段池化的结果 $a_{j,i}^{l+1,k}$ 的值为

$$a_j^{l+1,k} = \sum_{i=1}^n p_{j,i} a_{j,i}^{l,k} \quad (9)$$

本文同时考虑特征图上单元值所占概率和被保留的概率, 提出了新的模型平均方法

$$a_j^{l+1,k} = \sum_{i=1}^n qp_{j,i} a_{j,i}^{l,k} \quad (10)$$

式中, q 表示被保留的概率. 将这个值作为池化区域内考虑所有可能模型平均的近似值.

4 实验结果与分析

实验中 CNN 结构中参数动量 m 为 0.9,学习率为 0.1,batch size(一次进入 CNN 的图像数)为 50.实验环境:操作系统 WIN7,处理器 Intel®Xeon®CPU E5-4607 v2@2.60 GHz,2 600 MHz,6 个内核,12 个逻辑,RAM32GB,MATLAB R2012a.模拟实验的原始代码参考来自 Vedaldi A 等人^[9]的深度学习工具箱.分别在 MNIST 数据集和 CMU-PIE 数据库上进行实验.

4.1 MNIST 数据集实验

MNIST 数据集^[10]包括 60 000 幅训练图像和 10 000 幅测试图像,每幅图像为 28×28 ,包含了 0 到 9 其中的一个数字.实验中没有任何的预处理,只是将像素值规范到 $[0,1]$ 范围内.

实验中将本文方法与已有的 Bouvrie CNN^[11]、AlexNet CNN^[12] 和 L. M. Rere CNN^[13] 三种方法比较.由实验结果(表 1、2)可知,无论是 i-6c-2s-12c-2s 结构还是 i-8c-2s-16c-2s 结构,本文方法每次迭代的结果绝大部分好于文献[13]中的结果.

表 1 MNIST 数据集上不同方法同一 CNN(i-6c-2s-12c-2s)迭代 10 次的错误率 %

迭代次数	Bouvrie. CNN ^[11]	AlexNet. CNN ^[12]	L. M. Rere CNN ^[13]	本文 CNN
1	9.25	4.05	3.27	3.06
2	6.93	2.49	2.80	2.36
3	5.39	2.40	2.00	2.04
4	4.68	2.00	1.94	1.83
5	3.89	1.87	1.79	1.74
6	3.50	1.84	1.76	1.56
7	3.42	1.80	1.55	1.52
8	3.17	1.62	1.52	1.29
9	3.00	1.58	1.43	1.27
10	2.71	1.59	1.41	1.18

表 2 MNIST 数据集上不同方法同一 CNN(i-8c-2s-16c-2s)迭代 10 次的错误率 %

迭代次数	Bouvrie. CNN ^[11]	AlexNet. CNN ^[12]	L. M. Rere CNN ^[13]	本文 CNN
1	9.64	3.42	3.29	3.02
2	7.17	2.33	2.03	2.15
3	5.54	2.30	1.97	1.78
4	4.62	2.15	1.72	1.57
5	4.06	1.87	1.56	1.48
6	3.52	1.59	1.50	1.44
7	3.10	1.46	1.45	1.36
8	2.89	1.44	1.35	1.28
9	2.70	1.37	1.31	1.21
10	2.48	1.29	1.30	1.14

为进一步比较本文方法与 L. M. Rere CNN 的有效性,对两种 CNN 结构分别迭代 120 次,实验结果如图 2、3 所示,使用本文方法的错误率几乎每次迭代都低于 L. M. Rere 的结果.

另一方面,对比两种方法在结构 i-8c-2s-16c-2s 下的训练时间,计算迭代时间平均值. L. M. Rere CNN 训练一次 227.01 s,本文方法 225.51 s,快了 1.5 s,如果是迭代百次千次的实验,时间优势还是很明显的.

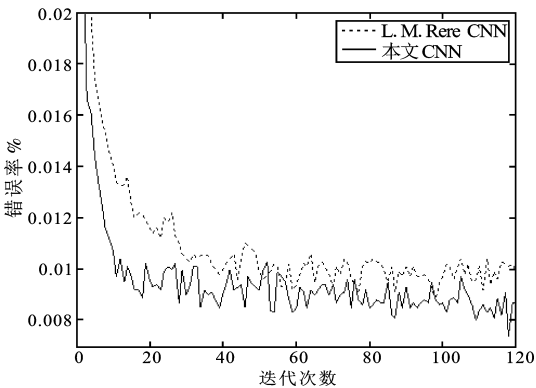


图 2 MNIST 上迭代 120 次的错误率(i-6c-2s-12c-2s)

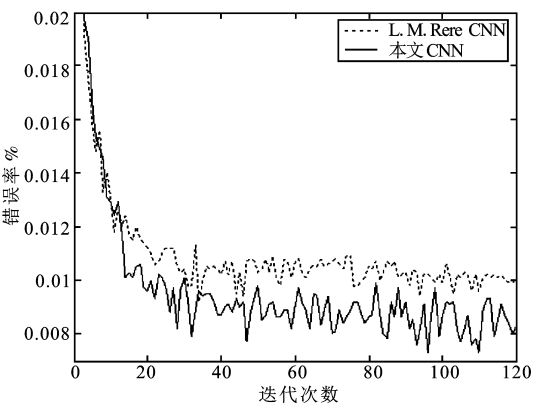


图 3 MNIST 上迭代 120 次的错误率(i-8c-2s-16c-2s)

CNN 结构中使用前面提到 dropout 正则化后的模型平均方法,在 i-20c-2s-40s 结构训练 300 次后结果如表 3 所示,采用本文方法错误率约为 0.44%,比 L. M. Rere CNN 降低了 0.21%,比文献[7]也降了 0.08%.

表 3 MNIST 上不同方法同一 CNN (i-20c-2s-40c-2s)的错误率

方法	错误率 %
Bouvrie. CNN	1.20
AlexNet. CNN	0.90
L. M. Rere CNN	0.65
Prob. weighted pooling ^[7]	0.52
本文 CNN	0.44

4.2 CMU-PIE 数据集实验

CMU-PIE 数据集包含 68 位志愿者的多姿态、光照和表情的面部图像,部分样本如图 4 所示,表 4 记录了迭代 50 次后的平均结果,结果表明,本文方法训练的 CNN 有着更好的性能,能够得到更好的识别率,并能更好地防止过度拟合.



图 4 CMU-PIE 数据集部分示例
表 4 CMU-PIE 上不同方法在 CNN
(i-20c-2s-40c-2s) 结构下的错误率

方法	错误率 %
Bouvier, CNN	21.45
AlexNet, CNN	11.50
L. M. Rere CNN	5.00
Prob. weighted pooling ^[7]	4.15
本文 CNN	2.60

5 结束语

本文提出的使用改进的模拟退火算法来优化 CNN 结构的解,提高了 CNN 的识别性能;使用 dropout 正则化方法后新的模型平均方法也可以更好地防止过度拟合问题. 在 MNIST 数据集和 CMU-PIE 数据集上的实验结果说明了上述方法的有效性. 训练和识别的时间也少于其他类似方法. 这里使用的数据集规模还不是很大,今后的研究工作和对与大规模数据集,应如何将以上思想应用到其它深度学习方法中.

参考文献:

[1] Zhining Y, Yunming P. The genetic convolutional neural network model based on random sample[J]. International Journal of u-and e-Service, Science and Technology, 2015, 8(11): 317-326.
[2] Rosa G, Papa J, Marana A, et al. Fine-tuning convo-

lutional neural networks using harmony search[C]//Iberoamerican Congress on Pattern Recognition. Springer International Publishing. ACM, 2015: 683-690.
[3] Le Cun Y, Kavukcuoglu K, Farabet C. Convolutional networks and applications in vision [C]//ISCAS. IEEE, 2010: 253-256.
[4] Nair V, Hinton G E. Rectified linear units improve restricted boltzmann machines[C]//Proceedings of the 27th International Conference on Machine Learning (ICML-10). IEEE, 2010: 807-814.
[5] El-Ghazali T. Metaheuristics: from design to implementation[J]. onh Wiley and Sons Inc. Chichester, 2009(9):10-11.
[6] Kirkpatrick S. Optimization by simulated annealing; Quantitative studies[J]. Journal of statistical physics, 1984, 34(5/6): 975-986.
[7] Wu H, Gu X. Towards dropout training for convolutional neural networks[J]. Neural Networks, 2015, 71(2015):1-10.
[8] Zeiler M D, Fergus R. Stochastic pooling for regularization of deep convolutional neural networks[J]. arXiv preprint, 2013.
[9] Vedaldi A, Lenc K. Matconvnet: Convolutional neural networks for matlab [C]//Proceedings of the 23rd ACM international conference on Multimedia. ACM, 2015: 689-692.
[10] Niu X X, Suen C Y. A novel hybrid CNN-SVM classifier for recognizing handwritten digits [J]. Pattern Recognition, 2012, 45(4): 1318-1325.
[11] Bouvier J. Notes on convolutional neural networks [R]. MIT CBCL Tech Report, 2006 : 38-44.
[12] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks [C]//Advances in neural information processing systems. IEEE, 2012: 1097-1105.
[13] Rere L M, Fanany M I, Arymurthy A M. Metaheuristic algorithms for convolution neural network[J]. Computational Intelligence and Neuroscience, 2016 (1):1-13.

作者简介:

满凤环 女, (1990-), 硕士研究生. 研究方向为模式识别、图像处理. E-mail: nora_nico@163. com.
陈秀宏 男, (1964-), 教授, 硕士生导师. 研究方向为图像处理、模式识别.
何佳佳 女, (1991-), 硕士研究生. 研究方向为模式识别、图像处理.