沈龙飞

猫狗大战

机器学习工程师纳米学位毕业项目

# 定义

## 1.1项目概述

这是一个Kaggle上的竞赛项目，目标是训练一个深度学习模型，在给定的图像中分辨猫和狗。属于图像分类问题。

本项目中使用的模型是**卷积神经网络**（Convolutional Neural Network, **CNN**）是一种前馈神经网络，它的人工神经元可以响应一部分覆盖范围内的周围单元，对于大型图像处理有出色表现。

卷积神经网络由一个或多个卷积层和顶端的全连通层（对应经典的神经网络）组成，同时也包括关联权重和池化层（pooling layer）。这一结构使得卷积神经网络能够利用输入数据的二维结构。与其他深度学习结构相比，卷积神经网络在图像和语音识别方面能够给出更好的结果。这一模型也可以使用反向传播算法进行训练。相比较其他深度、前馈神经网络，卷积神经网络需要考量的参数更少，使之成为一种颇具吸引力的深度学习结构[。](https://zh.wikipedia.org/wiki/卷积神经网络#cite_note-)

项目选择的数据集是Kaggle上竞赛提供的数据，训练集包括12500张被标记为猫和12500标记为猫的图片。测试机是包含12500未标记的图片。对于每一张测试机中的图像，模型需要预测出是狗图像的概率（1 代表狗，0 代表猫）。

## 问题陈述

Kaggle竞赛数据是从真实数据中采集到的包含猫或狗的图像。图像的分辨率参差不齐，

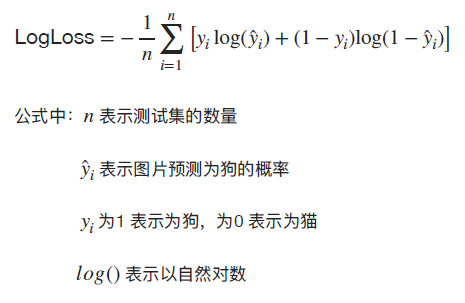
图片中猫和狗的主体也有很大的差异。同时猫和狗的不同品种造成的本身的形体差异。在加上拍摄角度与所处角度环境的关系。照成图片数据中的异常噪音非常大。

在图像分类问题中，需要出现两个过程：体征提取和分类。在特征提取环节，模型需要提取边缘，轮廓，色彩组合等的不同抽象级别特征。在分类部分，需要对提取出的特征进行准确分类。以得出分类结果。

## 评价指标

在kaggle的比赛中提出了标准的评价公式，本项目也采用相同的评价指标。

公式如下：

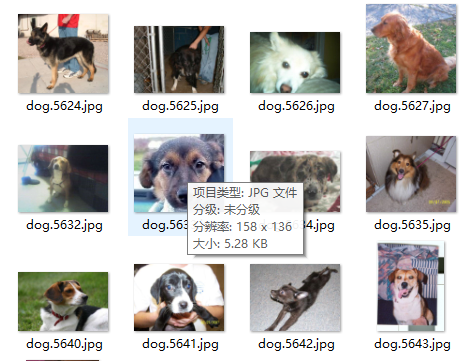


# 分析

## 2.1 数据可视化

竞赛数据集中有两个子集，一个训练数据集，一个是测试数据集，训练数据集中的文件名称即为标签。说明了图片数据猫还是狗。





训练集中共有25000张图片，猫狗各占一半，因此最后训练的时候不需要调整类别权重。

同时猫狗形态各异，要额外注意欠拟合和过拟合现象。

## 算法和技术

### 深度学习简介

**深度学习**（deep learning）是机器学习的分支，是一种试图使用包含复杂结构或由多重非线性变换构成的多个处理层对数据进行高层抽象的算法。

深度学习是机器学习领域中对模式（声音、图像等）进行建模的一种方法，它也是一种基于统计概率的模型。在对各种模型进行建模之后，便可以对各种模式进行识别了，例如待建模的模式是声音的话，那么这种识别便可以理解为语音识别。

深度学习是从很早之前的人工神经网络模型发展而来。这种模型一般采用计算机科学中的图模型来直观的表达，而深度学习的“深度”便是指的是图模型的层数以及每一层的节点上核量，相对之前的神经网络而言，有了很大的提升。

为了进行某种模式的识别，通常的做法首先是以某种方法，提取这个模式中的特征，这个特征有时候是人工设计或制定的，有时候是在给定相对较多的数据前提下，有计算机自己在模型中总结得出的。深度学习提出了一种让计算机自动学习出模式特征的方法，并将特征学习融入到了简历模型过程中，从而减少了不少认为设计特征的不完备性。而目前以深度学习为核心的某些机器学习应用，在满足特定条件的应用场景下，已经达到了超越现有是废的识别或分类性能。

由于深度学习能够自动学习，在学习时要额外注意损失函数的结果，是否存在过拟合的想象。再者，在模型训练过程中，需要大量的数据进行总结归纳，会造成训练耗时边长。

我们先了解一下神经元。神经元的结构如下图，也叫做感知机。



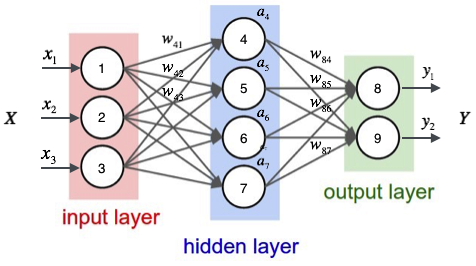
一个感知机包含如下部分：

输入权值（inputs），一个感知机可以接受多个输入（）在每个输入上又一个权值 ，此外还有一个偏置项 ,也就是上图的

激活函数感知器的激活函数可以有很多选择，比如我们可以选择下面的阶跃函数 来作为激活函数：

输出，感知机的输出有如下公式来计算：

深度学习使用的是神经网络模型，神经网络其实是按照一定规则连接起来的多个神经元。



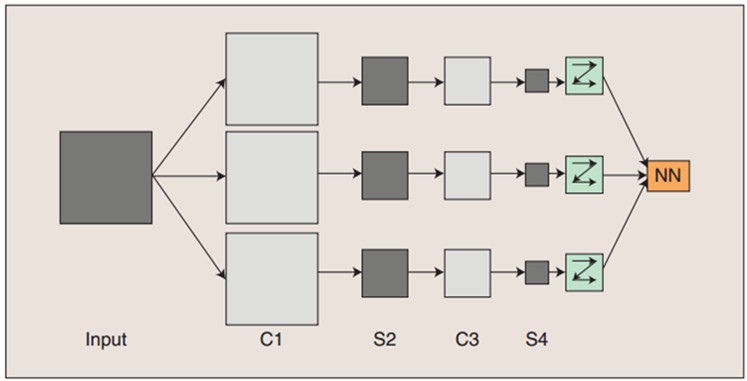
如上图所示：

* 神经元按照层来布局，最左侧叫输入层（input layer），负责接收输入数据；最右边的层叫输出层（output layer），我们可以从这层获取神经网络的输出数据。输入层和输出层之间的层叫做隐藏层（hidden layer），因为他对外部调用者来说是不可见的。
* 同一层之间的不同神经元没有连接。
* 第N层每个神经元和低N-1 层所有的神经元相连（Full Connected），第N-1 层神经元的输出就是低N层神经元的额输入
* 每一个连接都有一个权值

### 卷积神经网络

卷积神经网络（Convolutional Neural Network, CNN）是一种[前馈神经网络](https://zh.wikipedia.org/wiki/前馈神经网络)，它的人工神经元可以响应一部分覆盖范围内的周围单元，对于大型图像处理有出色表现。

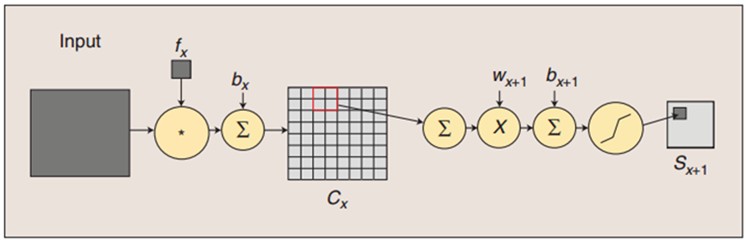
卷积神经网络由一个或多个卷积层和顶端的全连通层（对应经典的神经网络）组成，同时也包括关联权重和[池化](https://zh.wikipedia.org/w/index.php?title=池化&action=edit&redlink=1)层（pooling layer）。这一结构使得卷积神经网络能够利用输入数据的二维结构。与其他深度学习结构相比，卷积神经网络在图像和[语音识别](https://zh.wikipedia.org/wiki/语音识别)方面能够给出更好的结果。这一模型也可以使用[反向传播算法](https://zh.wikipedia.org/wiki/反向传播算法)进行训练。相比较其他深度、前馈神经网络，卷积神经网络需要考量的参数更少，使之成为一种颇具吸引力的深度学习结构。



*图：卷积神经网络的概念示范：输入图像通过和三个可训练的滤波器和可加偏置进行卷积，滤波过程如图一，卷积后在C1层产生三个特征映射图，然后特征映射图中每组的四个像素再进行求和，加权值，加偏置，通过一个Sigmoid函数得到三个S2层的特征映射图。这些映射图再进过滤波得到C3层。这个层级结构再和S2一样产生S4。最终，这些像素值被光栅化，并连接成一个向量输入到传统的神经网络，得到输出。*

### 卷积层的基本原理

卷积层（Convolutional layer），卷积神经网络中每层卷积层由若干卷积单元组成，每个卷积单元的参数都是通过[反向传播算法](https://zh.wikipedia.org/wiki/反向传播算法)最佳化得到的。卷积运算的目的是提取输入的不同特征，第一层卷积层可能只能提取一些低级的特征如边缘、线条和角等层级，更多层的网路能从低级特征中迭代提取更复杂的特征。



*图：卷积和子采样过程：卷积过程包括：用一个可训练的滤波器fx去卷积一个输入的图像（第一阶段是输入的图像，后面的阶段就是卷积特征map了），然后加一个偏置bx，得到卷积层Cx。子采样过程包括：每邻域四个像素求和变为一个像素，然后通过标量Wx+1加权，再增加偏置bx+1，然后通过一个sigmoid激活函数，产生一个大概缩小四倍的特征映射图Sx+1。*

通过对CNN和卷积的基本了解，大致可以认为在整个模型构建过程中，可以对结果产生影响的方面包括：

- 卷积的层数

- 滤波器的大小

- 隐藏层的数据

### 2.2.4 技术实现

项目实现CNN使用的是Facebook 开源的PyTorch。由于使用CNN处理分辨率全彩色图片所需要的计算量非常大，项目使用了GPU进行计算加速。在安装PyTorch的同时还安装了CUDA和cuDNN。

在图片处理的CNN问题中，主要包含的步骤如下：

1. 读取图片
2. 标准化图片
3. 卷积和池化（可以出现多次卷积和池化）
4. 归一化（隐藏层）
5. 输出结果

整个神经网络中由于面对具体问题的不一样可能会在不同的步骤呈现不同的特点。在猫狗问题中，我们面对的主要挑战来自，图片像素尺寸，以及猫狗动作的不一致性非常高。还包含又猫和狗的品种的不一致产生的数据噪音问题。

用深度学习中的术语来描述，则是说图片中提取的特征需要尽可能多，同时又要避免产生过拟合。

在深度学习的整个流程中，特征提取主要是在图片标准化，卷积和池化环境产生。换言之，这几个环节提取的特征数量，对后续的结果输出影响会很大。

## 基准指标

依据本题的考核标准。需要获得kaggle比赛前10%的成绩，在Kaggle的竞赛页面中，截止目前提交成绩人数为1314名。前10%则意味着分数要超过131名，在页面中可以看到第131名的得分为0.06127

基准模型指标得分需要小于0.06127

# 具体方法

## 3.1 迁移学习

### 3.1.1 迁移学习介绍

迁移学习，简单的说，就是能让现有的模型算法稍加调整即可应用于一个新的领域和功能的一项技术。这个概念目前在机器学习中其实比较少见，但其实它的潜力可以相当巨大。杨强教授在CCF-GAIR上的演讲中曾提到一个愿景——利用迁移学习，即使是自身没有条件获得大量训练数据的小公司也可以按照自己的需要应用大公司训练出来的模型，从而普及AI的应用。

本题中从零开始训练一个卷积神经网络，需要进行细致的网格结果设计，并调整大量的参数进行优化。耗费时间和精力非常惊人。因此我们采用迁移学习，通过微调其中参数以得到较好的结果。

### 3.1.2 模型选择

依据之前得出的论断，我们需要的是对特征尽可能多的提取，以保证在后面的全连接层中可以得到较好的分类结果。

而在PyTorch中提供有多种模型，最符合上面筛选条件的是ResNet模型。

两种主要的迁移学习方法如下：

**微调convnet**：我们使用预训练的网络进行参数初始化而不是参数随机初始化。之后的训练就和平常的训练一样了。

**convnet作为特征提取器**：这里，我们将冻结所有网络层的权重，除了最后一层全连接层。最后一层是一个参数随机初始化的新层，并且只有这一层被训练。

### 3.1.3 数据预处理

由于我们使用的迁移学习，所有数据预处理必须要能符合Resnet101的模型的输入结构。

我们这里对图片进行了随机切,然后resize 成224的大小.同时以0.5的概率翻转图像.

为了方便加载数据,我们将图片分为训练集和验证集,并且,训练集和测试机下都有猫和狗的二级目录.

猫狗训练集分别是11250张,测试机猫和狗分别是1250.

### 3.1.4 模型fine-tune

我们使用预先训练好的模型,这次我们在初期调试的时候,只在最后一层增加一个fc层.让最终的结果输出未两个类别.

## 3.2 实现

### 3.2.1 模型训练

在之前我们对图片进行了预处理中,我们已经做了简单的数据增强行为.主要是随机翻转,同时还有数据标准化.

优化器使用的SGD,学习率每七个epoch 下降0.1.

# 4. 结果

## 4.1 模型评估与验证

程序在批量读取测试集中的图片,并根据模型的进行相应的预处理,在初步使用resnet101的过程中,我们之前对比了vgg16 和vgg19 等两个模型.

最终几个网站的得分为:

vgg16 得分为:0.12744

vgg19 得分为:0.19743

resnet50 得分为:0.71907

resent101 得分为:0.04898

resnet101的得分,目前处于kaggle 猫狗项目的52 和 53名之间.已经满足目前的得分范围.

# 5.结论

猫狗大战是一个充满挑战的竞赛,使用有限的数据样本,发掘尽可能多的特征.CNN是一个非常灵活的模型,围绕分类以及特征表示,有非常多的方法可以运用,

在整个迁移学习的过程中,尝试过很多中不同的方法,比如结构简单训练速度也很快的VGG模型,但是具体到实际问题的时候,发现存在很多问题.在之前的开发过程中,大致认为是特征提取数量不够,因为原始数据中对图片特征的干扰非常之多.

文献参考:

[1] Deep Residual Learning for Image Recognition

[Kaiming He](https://arxiv.org/find/cs/1/au:+He_K/0/1/0/all/0/1), [Xiangyu Zhang](https://arxiv.org/find/cs/1/au:+Zhang_X/0/1/0/all/0/1), [Shaoqing Ren](https://arxiv.org/find/cs/1/au:+Ren_S/0/1/0/all/0/1), [Jian Sun](https://arxiv.org/find/cs/1/au:+Sun_J/0/1/0/all/0/1)

[2] Very Deep Convolutional Networks for Large-Scale Image Recognition

[Karen Simonyan](https://arxiv.org/find/cs/1/au:+Simonyan_K/0/1/0/all/0/1), [Andrew Zisserman](https://arxiv.org/find/cs/1/au:+Zisserman_A/0/1/0/all/0/1)

[3] THE INTEGRATION OF GEOGRAPHIC DATA WITH REMOTELY SENSED IMAGERY TO IMPROVE CLASSIFICATION IN AN URBAN AREA