

# 开题报告

## 一．项目背景

自 2006 年，Geoffrey Hinton 提出了深度学习后，同时伴随着大数据的出现以及计算机硬件的飞速发展，深度学习成为近十年来人工智能领域取得的最重要的突破之一。它在语音识别、自然语言处理、计算机视觉、图像与视频分析、多媒体等诸多领域都取得了巨大成功。在 2012 年，Hinton 的研究小组采用深度学习赢得了 ImageNet 图像分类的比赛。ImageNet 是当今计算机视觉领域最具影响力的比赛之一。随后谷歌和百度分别沿用 Hinton 在 ImageNet 竞赛中用的深度学习模型，发现图像搜索的准确率得到了大幅度的提高。

本项目来源于 kaggle 竞赛，是一个图像分类问题。通过训练集的学习让计算机能够分别识别猫和狗的图像。

## 二．问题描述

猫狗大战目来源于 kaggle 竞赛，是一个图像二元分类问题，即判断一张图片的分类是猫还是狗，1 表示分类结果是狗，0 表示分类结果是猫。需要根据 kaggle 提供的训练集图片来训练自己的深度学习模型来对测试集的图片进行预测。

## 三．数据或输入

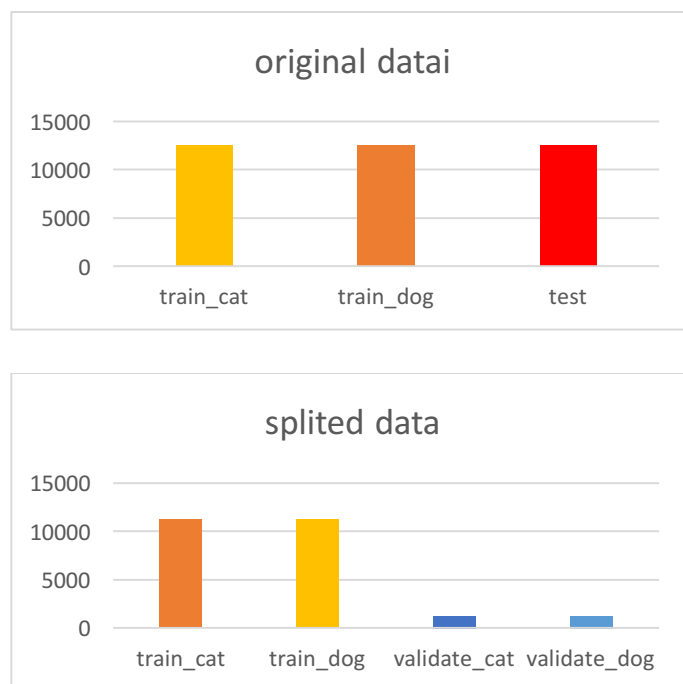
kaggle 一共提供了两个数据集：train 和 test（可以从 <https://www.kaggle.com/c/dogs-vs-cats-redux-kernels-edition/data> 进行数据下载）。其中训练集包含 25000 个样本，cat 和 dog 各 12500 个，测试集有图片 12500 张，但并未标注猫与狗，用于最后的模型测试。通过观察 train 和 test 的中图片，发现图片中不会同时存在猫和狗，但是可能同时会有两个猫或者两条狗。而且每张图片的清晰度和图片的大小不同，猫或狗占整张图片的比例也有所差别。



接下来，我会把 train 文件夹按照 10%，90%的比例分成两部分：

- 训练数据集(包含两个文件夹)
  - dog（包含需要参与训练的 11250 张狗的图片）
  - cat（包含需要参与训练的 11250 张猫的图片）
- 验证数据集(包含两个文件夹)
  - dog（包含需要参与验证的 1250 张狗的图片）
  - cat（包含需要参与验证的 1250 张猫的图片）

下面是数据分布图：



## 四．评估标准

kaggle 一共举行过两次猫狗大战的比赛，第一次使用正确率作为评估标准，第二次使用的是 log 损失函数。因为现在深度学习发展十分的迅速，而深度学习尤其适合处理图像方面的问题，如果依旧是使用正确率作为评估标准，那么大多数选手的模型都是 99% 的正确率，不能明显地区分开。如果使用 log 损失函数，不仅仅需要分类正确，还需要对结果有一个较高的可信度，这样就能明显地区分各个模型的分类效果，尤其是 Top 模型的分类效果。log 损失函数具体如下所示：

$$LogLoss = -\frac{1}{n} \sum_{i=1}^n [y_i \log \hat{y}_i + (1 - y_i) \log (1 - \hat{y}_i)]$$

- n 表示测试集样本数量
- $\hat{y}_i$  表示该图片被预测为狗的概率
- $y_i$  表示如果图片为 dog，则为 1，反之为 0

## 五．基准模型

关于基准模型，根据 ILSVRC 历年的 top 错误率排行榜，我参考了前三名：VGG、GoogleNet 和 ResNet。本次毕业项目采用经典的 ResNet 模型，ResNet 在 2015 年提出以来，曾在 ImageNet 比赛 classification 任务上获得第一名，因为它“简单与实用”并存，之后很多方法都建立在 ResNet50 或者 ResNet101 的基础上完成的，检测，分割，识别等领域都有着广泛的运用。所以最终选择 ResNet50 做为基准模型。

除了 ResNet50 外，我还试了比 ResNet50 更深的 InceptionV2，但是在训练集上得到的 accuracy (85%左右) 并没有 ResNet50 的好，其中一个可能的原因是数据集的数量不够大。

通过本项目的最低要求为 kaggle 排行榜前 10%，也就是在 Public Leaderboard 上的 logloss 要低于 0.06127，所以我的最终目标是 logloss 要低于 0.06127。

## 六. 项目设计

### 1. 划分数据集

Kaggle 一共提供了两个数据集 train 和 test，所以需要把 train 数据按照一定比例划分出训练集和验证集，划分后文件结构如下：

```
data/
  train/
    dogs/
      dog001.jpg
      dog002.jpg
      ...
    cats/
      cat001.jpg
      cat002.jpg
      ...
  validation/
    dogs/
      dog001.jpg
      dog002.jpg
      ...
    cats/
      cat001.jpg
      cat002.jpg
      ...
```

在本次毕业项目中，划分比例是 10%。

### 2. 数据预处理

其中包含两部分：(1) ResNet50 所需要的图片大小是 (224, 224, 3)，而训练集里的图片大小不相同，所以需要把图片进行 resize；(2) 考虑当前的数据集不是很大，所以需要通过 augment 图像增广技术来扩充数据集。这里使用 Keras.preprocessing.image.ImageDataGenerator 方法来 做完成上面两部分预处理。

### 3. 构造模型

本项目的模型结构是 ResNet50+3 层 Dense layer，项目中直接使用了 keras 中的预训练模型 ResNet50，模型的 weights 也是 keras 中训练的结果。Dense layer 的 weights 是通过 kaggle 下载的数据训练出来的。

### 4. 训练、优化模型

首次是 batch size 和 training epochs 的设定。

- 太大的 batch size 需要的足够大的内存空间，太小的 batch size 可能会导致 overfit，在一开始训练的时候 batch size=512，结果导致 out of memory，所以在最后的训练中 batch size 设定为 256。
- 太大的 training epochs 也可能会引起 overfit，但是太小可能会导致训练不充分，得不到最好的模型。我在训练中 epochs 设为 200，并且保存每个 epoch 产生的 weights 结果，最后通过观察 tensorboard 中的图表结果，选择 accuracy 基本稳定 loss 最低的结果。

使用 SGD 做为梯度步长更新的方法，为了防止 Learning rate 设置过大，同时设置了 Decay 和 Momentum。Momentum 使模型快速收敛，Decay 提高 SGD 寻优能力。

对于过拟合的处理本项目中用到了两种方法：

- 增加数据量, 通过 augment 图像增广技术来扩充数据集
- 使用 Dropout

#### 5. 测试数据

使用保存的模型，对 test 中的数据进行预测，

## 七 . 参考文献

- [1] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei - Fei. Imagenet: A large - scale hierarchical image database. In IEEE Int' l Conf. Computer Vision and Pattern Recognition, 2009.
- [2] A. Krizhevsky, L. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In Proc. Neural Information Processing Systems, 2012.
- [3] kaggle 猫狗大战手记,  
<http://moverzp.com/2018/01/21/kaggle%E7%8C%AB%E7%8B%97%E5%A4%A7%E6%88%98%E6%89%8B%E8%AE%B0/>
- [4] Deep Residual Learning for Image Recognition,  
<https://arxiv.org/pdf/1512.03385.pdf>
- [5] Dogs vs. Cats Redux: Kernels Edition