

二维卷积神经网络层

Conv2D(input, #张量输入
Filter,#卷积核参数
Strides,#步长
Padding,#卷积方式（边缘填充）
Use_cudnn_on_gpu=None,#使用 N 卡 GPU
Data_format=None,#数据格式，于步长参数配合，决定移动方式
Name=None#用于 tensorboard 图形显示时使用)

激活（函数）层

Activation(activation,#激活函数名字
**kwargs#可以有 input_shape，当作为网络的第一层时),

最大池化层

MaxPooling2D(pool_size,#池化窗口大小
Strides,#步长
Padding,#边缘填充
data_format,#'channels_first','channels_last'
name# tensorboard
),

随机丢弃

Dropout(Rate,#丢弃率
Noise_shape=None,#一维整数张量，表示将与输入相乘的二进制丢失掩码的形状相同。
Seed=None,#用作随机种子的 python 整数
**kwargs 其他参数
),

全连接层

Dense(units,#输出空间的维数
Activation,
Use_bias,#字面意思
Kernel_initialize,#核初始化
Bias_initialize,#B 初始化
Kernel_regularizer,#核的正则化函数
Bias_regularizer,#B 的正则化函数
Activity_regularizer,#当前层的激活函数
Kernel_constraint,#核的矩阵约束
Bias_constraint,#B 的矩阵约束)

配置用于训练的模型

compile(optimizer,#优化器
loss,#目标函数

```
metrics, #评价标准
loss_weights=None,#
sample——weights=None,#
weights_metrics,#权重评价
target_tensors=None.
Distribute=None,
**kwargs), 、
```

Evaluate——返回测试模式下模型的损失值和指标值

为了模型训练固定的 epochs 的迭代

```
fit(x=None,#配置训练的输入数据
    y=None,#配置训练的标注数据
    batch_size=None,#配置批大小
    epoch=1,#epoch 数量
    verbose=1,#训练信息的展示方式
    callbacks=None,#断点续训
    validation_split,#划分验证集占训练集的比例
    validation_data,#配置验证集
    shuffle=True,#随机打乱数据
    class_weights=None,#
    sample_weights,
    initial_epoch,#新的训练周期是从指定的 epoch 开始训练的。
    steps_per_epoch,#配置每个 epoch 训练的步数
    validation_steps,
    validation_freq
    max_queue_size,
    workers=1,
    use_multiprocessing,**kwargs)
```