# 文件路径

DAM模型路径：

“./models-release-1.5/PaddleNLP/dialogue\_model\_toolkit/deep\_attention\_matching”

数据保存路径：

“./models-release-1.5/PaddleNLP/dialogue\_model\_toolkit/deep\_attention\_matching/data/data.pkl”

训练模型保存路径：

“./models-release-1.5/PaddleNLP/dialogue\_model\_toolkit/deep\_attention\_matching/saved\_models”

预训练模型保存路径：

“./models-release-1.5/PaddleNLP/dialogue\_model\_toolkit/deep\_attention\_matching/model\_files”

# 参数说明

Config文件路径：

“./models-release-1.5/PaddleNLP/dialogue\_model\_toolkit/deep\_attention\_matching/config.py”

* do\_train

是否进行训练

* do\_test

是否进行测试

* num\_scan\_data

训练epoch数量，论文中默认为2

* data\_path

data.pkl所在路径

* save\_path

Train时生成的模型文件保存路径

* model\_path

预训练的模型保存位置

* use\_cuda

是否使用GPU加速（如果使用GPU需要安装gpu版的paddle）

* ext\_eval

数据为中文时设置为store\_false

# Train

1. 设置config中do\_train为True
2. 运行DAM模型路径下的main.py
3. 训练好的模型文件以及结果保存在save\_path中设置的路径下，包括三个部分：
   1. “Step\_”+序号的文件夹为模型文件
   2. “score\_”+序号的txt文件为使用当前step的模型为train dataset中每一行数据标注的label，用于检验模型的train accuracy
   3. “result\_”+序号的txt文件中为当前step模型的各项评估指标数据

# Evaluation

1. 设置config中do\_test为True
2. 设置config中model\_path为需要检验的模型文件夹路径（例如“./saved\_models/step\_400”）
3. 运行DAM模型路径下的main.py
4. evaluation结果保存在save\_path中设置的路径下，包括两个部分：
   1. “score\_”+序号的txt文件为使用当前step的模型为train dataset中每一行数据标注的label，用于检验模型的train accuracy
   2. “result\_”+序号的txt文件中为当前step模型的各项评估指标数据

# Predict

1. 参考主文件夹中Predict\_with\_DAM.ipynb文件里Embedding部分代码，将需要predict的数据打包为predict.pkl，并保存在save\_path中设置的路径下
2. 运行DAM模型路径下的predict.py
3. 运行主文件夹中的Predict\_with\_DAM.ipynb的Predict部分代码，得到模型预测出的正确回答