

案例 2

```
from sklearn.model_selection import train_test_split
from keras.models import Sequential
from keras.layers import Dense, LSTM, Embedding, SpatialDropout1D
import matplotlib.pyplot as plt
plt.rcParams[' font.sans-serif' ] = [ ' SimHei' ]
plt.rcParams[' axes.unicode_minus' ] = False
import seaborn as sns
from sklearn.metrics import confusion_matrix, classification_report
from sklearn import svm
import pandas as pd
# 测试集效果检验
from sklearn import metrics
from sklearn import svm
from utils import load_corpus, stopwords
from sklearn.feature_extraction.text import TfidfVectorizer
from utils import processing
labels_train=[]
sentences_train=[]
TRAIN_PATH = ""
TEST_PATH = ""
# 分别加载训练集和测试集
train_data = load_corpus(TRAIN_PATH)
test_data = load_corpus(TEST_PATH)
df_train = pd.DataFrame(train_data, columns=["words", "label"])
df_test = pd.DataFrame(test_data, columns=["words", "label"])
df_train.head()
vectorizer=TfidfVectorizer(token_pattern='[? \w+\ ]? ', stop_words=stopwords)
X_train = vectorizer.fit_transform(df_train["words"])
y_train = df_train["label"]
X_test = vectorizer.transform(df_test["words"])
y_test = df_test["label"]
clf = svm.SVC(C=0.5, kernel=' rbf' , degree=3, coef0=0.0, shrinking=True, probability=False, tol=0.001, cache
_size=200, class_weight=None, verbose=False, max_iter=- 1, decision_function_shape=' ovr' , random_state=
None)
clf.fit(X_train, y_train)

# 在测试集上用模型预测结果
y_pred = clf.predict(X_test)
print(metrics.classification_report(y_test, y_pred))
print("准确率:", metrics.accuracy_score(y_test, y_pred))
```