数据预处理步骤

根据获取到的中文数据，我们这边要做情感分析，首先需要做数据预处理

数据预处理的步骤

首先读取相关的数据表格：

原数据一共为：118633条，

后经过处理，剩余数据为：103620条

处理过程，首先基于原始的数据做去掉标点符号和机械压缩

```
def preprocess_word(word):
    word1 = str(word)
    word1 = re.sub(r'#\w+#', '', word1)
    word1 = re.sub(r'【.*?】', '', word1)
    word1 = re.sub(r'@[\w]+', '', word1)
    word1 = re.sub(r'[a-zA-Z]', '', word1)
    word1 = re.sub(r'\.\d+', '', word1)
    return word1
```

在做好这边之后去掉一些表情包，防止影响情感判断

```
def emjio_tihuan(x):
    x1 = str(x)
    x2 = re.sub('(\[.*?\])', "", x1)
    x3 = re.sub(r'@[\w\u2E80-\u9FFF]+:?|\[\w+\]', '', x2)
    x4 = re.sub(r'\n', '', x3)
    return x4
```

接着我们使用jieba进行分词处理，仅保留名词 动词和形容词，并且去判断这个分词是否为中文，它是否不在停用词文本里面，以及它的长度是否大于等于2，以此来保留有效的分词信息

```
def get_cut_words(content_series):
    try:
        # 对文本进行分词和词性标注
        words = pseg.cut(content_series)
        # 保存名词和形容词的列表
        nouns_and_adjs = []
        # 逐一检查每个词语的词性，并将名词和形容词保存到列表中
        for word, flag in words:
            #判断是否为名词或者形容词或者动词
            if flag in ['Ag','a','ad','an','Ng','n','v']:
                if word not in stop_words and len(word) >= 2 and
is_all_chinese(word) == True:
                    # 如果是名词或形容词，就将其保存到列表中
                    nouns_and_adjs.append(word)
        if len(nouns_and_adjs) != 0:
            return ' '.join(nouns_and_adjs)
        else:
            return np.NAN
    except:
```

```
        return np.NAN
```

后，我们再使用百度的飞浆情感分析模型来进行情感判断，获取这个情感分类的类别以及它的准确性

```
def emotion_analysis(text):
    # 进行情感分析
    results = sentiment_analysis(text)
    for result in results:
        label = result['label']
        score = result['score']
        return label,score
```

这个是飞浆的模型介绍：https://aistudio.baidu.com/projectdetail/3696243?contributionType=1

在处理好上面的事情后，我们开始进行数据处理和情感分类

```
df = pd.read_excel('评论表.xlsx')
# 初始化情感分析任务
sentiment_analysis = Taskflow("sentiment_analysis")
print('原数据总数:',len(df))
df['评论内容'] = df['评论内容'].apply(preprocess_word)
df['评论内容'] = df['评论内容'].apply(emjio_tihuan)
df = df.dropna(subset=['评论内容'], axis=0)
df['fenci'] = df['评论内容'].apply(get_cut_words)
df = df.dropna(subset=['fenci'], axis=0)
print('清洗过后数据总数:',len(df))
list_label = []
list_score = []
for d in df['fenci']:
    label,score = emotion_analysis(d)
    list_label.append(label)
    list_score.append(score)
df['情感类别'] = list_label
df['准确率'] = list_score
df.to_csv('new_data.csv',index=False,encoding='utf-8-sig')
```