《人工智能导论》实验三设计说明

主讲人：马少平教授
助教：李祥圣

清华大学计算机系人工智能研究所

1. 任务简介

情感分析（sentiment analysis）是近年来国内外研究的热点，其任务是帮助用户快速获取、整理和分析相关评价信息，对带有情感色彩的主观性文本进行分析、处理、归纳和推理。

随着互联网技术的迅速发展和普及，对网络内容管理、监控和有害（或垃圾）信息过滤的需求越来越大，网络信息的主观倾向性分类受到越来越多的关注。这种分类与传统的文本分类不同，传统的文本分类所关注的是文本的客观内容（objective），而倾向性分类所研究的对象是文本的“主观因素”，即作者所表达出来的主观倾向性，分类的结果是对于一个特定的文本要得到它是否支持某种观点的信息。这种独特的文本分类任务又称为**情感分类（sentiment classification）**。

图 1：新闻中的情感投票例子

2. 实验数据

实验数据来源于新浪社会频道收集的 4570 篇中文新闻文章。每条新闻中中包含了不同数量用户的情感打分，共 8 种情感类别。该数据中每篇新闻文章至少有 6 个单词和 1 个用户评分。数据中已通过中文分词工具对文本进行分词处理。每行数据由时间戳，情感分布，文本三个部分组成，以 tab(\t)分割。其中，情感分布以及文本中的词以空格分割。最终，统一使用 2012 年 1 月至 2 月发布的 2,342 篇新闻文章来构建训练集，而使用 2012 年 3 月至 4 月发布的 2,228 篇新闻文章来创建测试集。

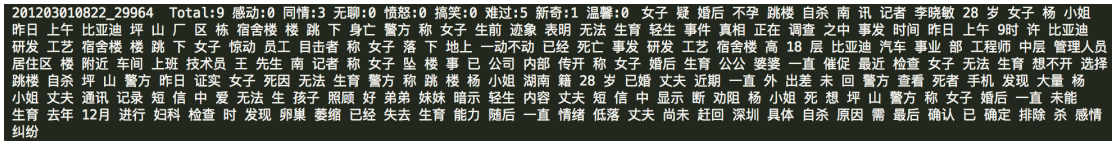

图 2：数据集样例

3. 数据处理

1） **文本表示方法：**

a) Bags-of-words，是信息检索领域常用的文档表示方法。对于一个文档，计算**每个词出现的频率**将其表示，忽略它的单词顺序和语法、句法等要素，将其仅仅看作是若干个词汇的集合，文档中每个单词的出现都是独立的，不依赖于其它单词是否出现。也就是说，文档中任意一个位置出现的任何单词，都不受该文档语意影响而独立选择的。

b) TF-IDF 特征表示，可减少高频停用词的影响。

c) word-embedding 方法表示，目前较常用的方法，可以利用在大语料上训练好的文本向量进行初始化。常用的有 Glove，word2vec 模型。预训练的词向量下载（参考）：https://github.com/Embedding/Chinese-Word-Vectors（在实验中不一定要用到这一步，可以不初始化词向量，直接在训练中学习调整）

2） **标签表示方法：**

a) 最大值转为单标签预测（分类问题）：
取图 2 中例子，标签分布为[0,3,0,0,0,5,1,0],最大值为 5，因此最终标签转为[0,0,0,0,0,1,0,0]。目标函数为交叉熵：

$$\mathcal{L} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m} y_{ij}\log\left(p_{ij}\right)$$

下标 i 代表第 i 次样本，下标j代表第 j 个类别的概率，y 是真实标签的分布，p是预测的标签分布。$p_{\mathrm{ij}} \in (0,1)$：$\sum_{j=1}^{m} p_{ij} = 1 \forall i,j$。n 是样本个数，m 是标签类别个数。

b) 归一化情感分布（回归问题）：
取图 2 中例子，标签分布为[0,3,0,0,0,5,1,0]，归一化分母为 3+5+1=9，因此最终标签转为[0,3/9,0,0,0,5/9,1/9,0]。目标函数为均方误差：

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(\tilde{Y}_i - Y_i\right)^2$$

4. 实验要求

本次实验要求**实现 CNN 与 RNN 两个模型**，并应用在情感分类任务上。RNN 可以是 LSTM，GRU 等类型。代码的语言不限，可借助深度学习的框架实现（theano，TensorFlow，keras 等）。**对比两模型的实验效果，并分析原因**。也可以实现其他模型作为对比模型（baseline），例如全连接神经网络（MLP），可适当加分。

1）**卷积神经网络（CNN）：**

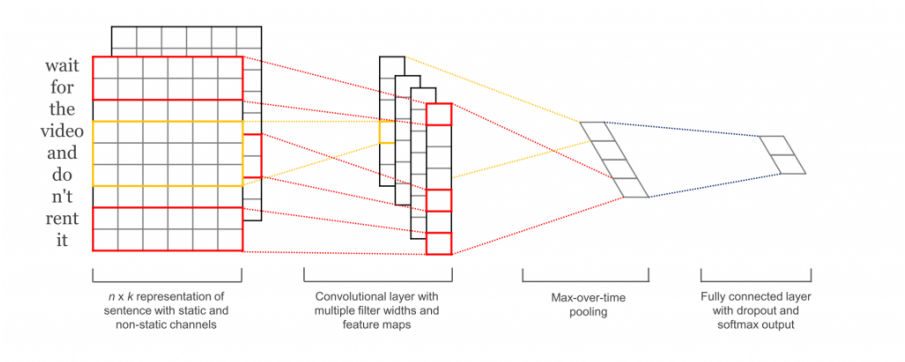

图 3：CNN 模型框架图

参考论文：Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014), 1746–1751.

2）**循环神经网络（RNN）：**

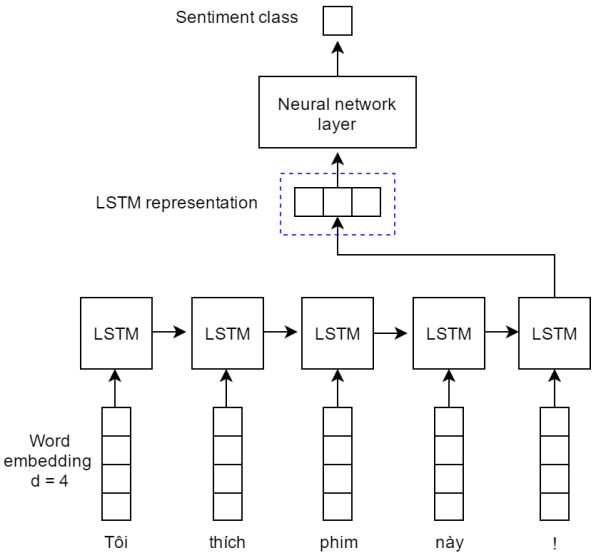

图 4：LSTM 的模型框架

3）**评价指标：**

a) **准确率**（Accuracy）：取情感标签中最大值为 ground truth，预测的最大概率标签为预测值，求整个测试集中的分类准确率。

b) **F-score**：计算 precision 以及 recall，最终由公式$F = \frac{2}{\frac{1}{\text{precision}}+\frac{1}{\text{recall}}}$得到。

Precision 和 recall 计算方式可见下图 5 所示。

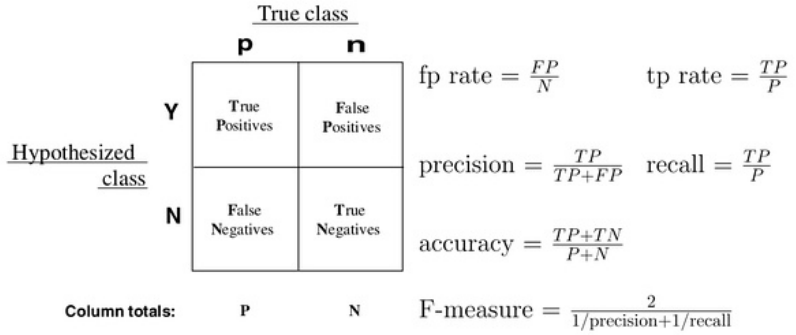

Fig. 1. Confusion matrix and common performance metrics calculated from it.

图 5：F-score 计算方式

对于一般的二分类问题，可以利用上述指标计算。但本实验是多分类问题，在综合考察模型性能的时候就会用到到宏平均和微平均。

- **宏平均（Macro-averaging）**，是先对每一个类都计算一个 F1 值，然后在对所有类求算术平均值。
- **微平均（Micro-averaging）**，是对数据集中的每一个类都计算 TP、FP 和 FN 值，再将多个类的这些值进行累加，从而计算 Precision、Recall 与 F1 值。

在 python 中，可以直接调用 sklearn 的工具进行计算，在实践中可能还会用到 weighted 平均计算方式，具体可以参考下面的链接：
https://www.jianshu.com/p/9e0caf109e88

报告中请说明自己计算的 F1 值是哪一种加权方式。

c) **相关系数**（Correlation Coefficient）：Accuracy 与 F-score 只考虑最大概率的预测值，而不考虑其他标签的值。因此，我们可以利用相关系数衡量预测分布与真实分布的接近程度。

$$r(X,Y) = \frac{Cov(X,Y)}{\sqrt{Var[X]\,Var[Y]}}$$

其中，Cov(X,Y)为 X 与 Y 的协方差，Var[X]为 X 的方差，Var[Y]为 Y 的方差。在 python 中，可以调用函数直接计算：

```
import numpy as np
from scipy.stats import pearsonr

x = np.array([0,0.2,0.3])
y = np.array([0.1,0.1,0.2])

print "Coef: ",pearsonr(x,y)
```

请在实验后计算以上三种评价指标，并如实地汇报在实验报告中。

5. 实验报告内容

1）模型的结构图，以及流程分析。
2）实验结果，准确率，F-score，相关系数三个指标的实验效果。
3）试简要地比较实验中使用的不同参数效果，并分析原因。
4）比较 baseline 模型与 CNN，RNN 模型的效果差异。（如果有实现）
5）问题思考
6）心得体会

6. 问题思考

1） 实验训练什么时候停止是最合适的？简要陈述你的实现方式，并试分析固定迭代次数与通过验证集调整等方法的优缺点。
2） 实验参数的初始化是怎么做的？不同的方法适合哪些地方？（现有的初始化方法为零均值初始化，高斯分布初始化，正交初始化等）
3） 过拟合是深度学习常见的问题，有什么方法可以方式训练过程陷入过拟合。
4） 试分析 CNN，RNN，全连接神经网络（MLP）三者的优缺点。

7. 评价方式

程序结果与代码：60%
实验报告：40% （baseline 不一定要实现，但实现可根据难度加分）

8. 提交方式

在网络学堂上提交，需要提交的必要材料如下：
1) 实验报告，以学号_姓名.pdf 命名；
2) 实验代码以及程序运行导引（README）。

9. 联络方式

助教：李祥圣
手机：13763361656 （微信同）
电邮：lixsh6@gmail.com