**《信息检索与Web挖掘》**

实验报告

小组成员：陈泽濠 徐成 刘广鹏

指导老师：董守斌

2017.06.07

基于神经网络的重复问题检测

一、实验目的

* 了解海量数据分析的基本原理及方法；
* 掌握基础的自然语言处理技术；
* 掌握实验评估的基本方法。

二、实验内容与要求

问答网站中需要对相似问题进行合并来优化用户体验，而目前的合并工作都是通过人工识别或者字符完全匹配进行，效率低下。问答网站Quora最近发布了一个公开数据集（https://data.quora.com/First-Quora-Dataset-Release-Question-Pairs），数据集中每一行包括两个部分，根据一定规则抓取的相似问题以及两个问题是否重复的人工标注结果，共有400000多行数据。请下载相关数据，并设计算法实现重复问题的识别，以十折交叉验证下的F1值作为算法的评估指标。

三、总体框架

计算TF-IDF

生成word2vec

词向量空间

三层神经网络

交叉验证

F1值

图3-1 总体框架

四、实验过程

4.1 数据信息

数据集包含404290行数据，每行数据包括六个字段，如图4-1所示。其中，id字段唯一标识每条数据，qid1和qid2字段表示不同问题的id，question1和question2表示问题的具体内容，is\_duplicate表示两个问题是否重复，0表示重复，1表示不重复。

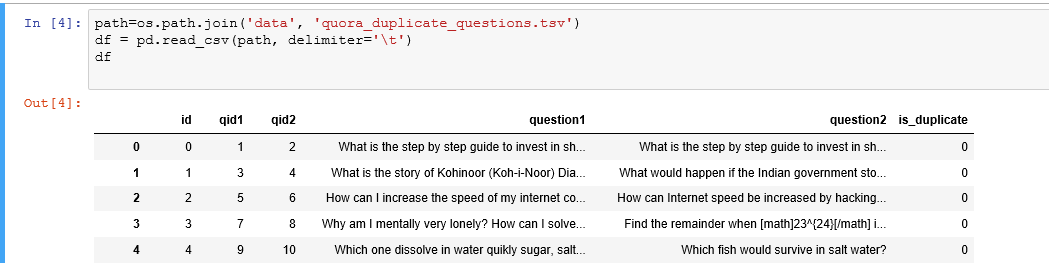


图4-1 数据格式

首先，去除冗余数据与噪声，并获取数据集的统计信息。

* 数据量：404288行
* 重复：不重复 = 149263：255025 = 1：1.708561
* 问题总数：537931

其中，问题总数根据qid进行区分，共得出537931个不同的问题。

对每个问题的长度进行统计，可以得到如下扇形图：

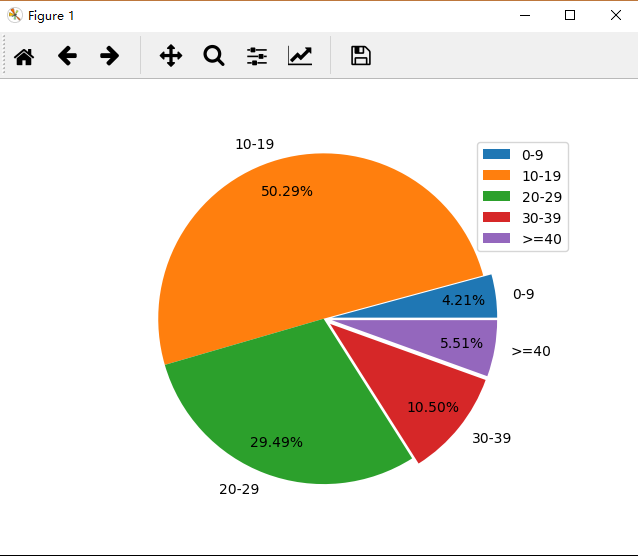


图4-2 问题长度统计

从图中可以看出，超过百分之五十的问题长度在10-19之间，将近百分之八十在10-29之间，由此可见，Quora上问题更趋向于中断句，这也符合人们日常提问的习惯。

4.2 数据预处理

计算TF-IDF

生成word2vec

词向量空间

图4-3 数据预处理流程

如图4-3所示，首先根据数据集，将每一个问题看成一个独立的文档，所有问题组成文档集，计算每个词在该文档集中的TF-IDF值。具体实现调用Python第三方sklearn库，返回结果为dict={'word':tfidfScore}字典，该字典中包含所有词以及每个词的TF-IDF值。

生成word2vec是使用spaCy现成的自然语言文本处理库，spaCy是一个Python自然语言处理工具包，诞生于2014年年中，号称“Industrial-Strength Natural Language Processing in Python”，是具有工业级强度的Python NLP工具包。spaCy里大量使用了Cython来提高相关模块的性能，这个区别于学术性质更浓的Python NLTK，因此具有了业界应用的实际价值。本实验中使用spaCy的en\_core\_web\_md模块，将词转化成300维向量。最后，将300维向量与该词的TF-IDF值相乘，每个问题转化为word\*300的矩阵，其中，word为问题的词个数，对没一维求均值，便得到了最终问题的向量空间。

经过预处理后的数据如图4-4所示，与原始数据相比增加了两个字段，q1Vector、q2Vector均表示问题的向量形式，为300维向量。

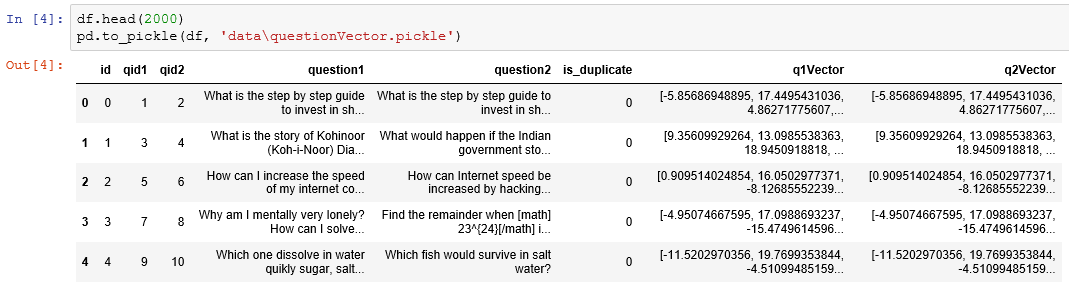


图4-4 预处理后数据

4.3 交叉验证

交叉验证（CV，Cross Validation）是用来验证分类器性能的一种统计分析方法。其基本思想是，在训练开始之前，将原始数据划分为训练集和测试集，利用训练集进行模型训练，之后利用测试集对学习结果进行验证。

本实验采用十折交叉验证，先将数据集切分为十份，再分别取其中一份为测试集，余下九份为训练集，共训练十次，最终F1值取平均数。

使用Sklearn库的KFold函数能较快实现数据集切分。

4.4 搭建网络

使用Keras搭建三层网络结构，每一层使用relu函数作为激活函数，并且每一层使用BatchNormalization进行规范化，BN层在每个batch上将前一层的输出重新规范化，即使得其输出数据的均值接近0，其标准差接近1。实验一开始没有加入BN层，导致输出结果不收敛。

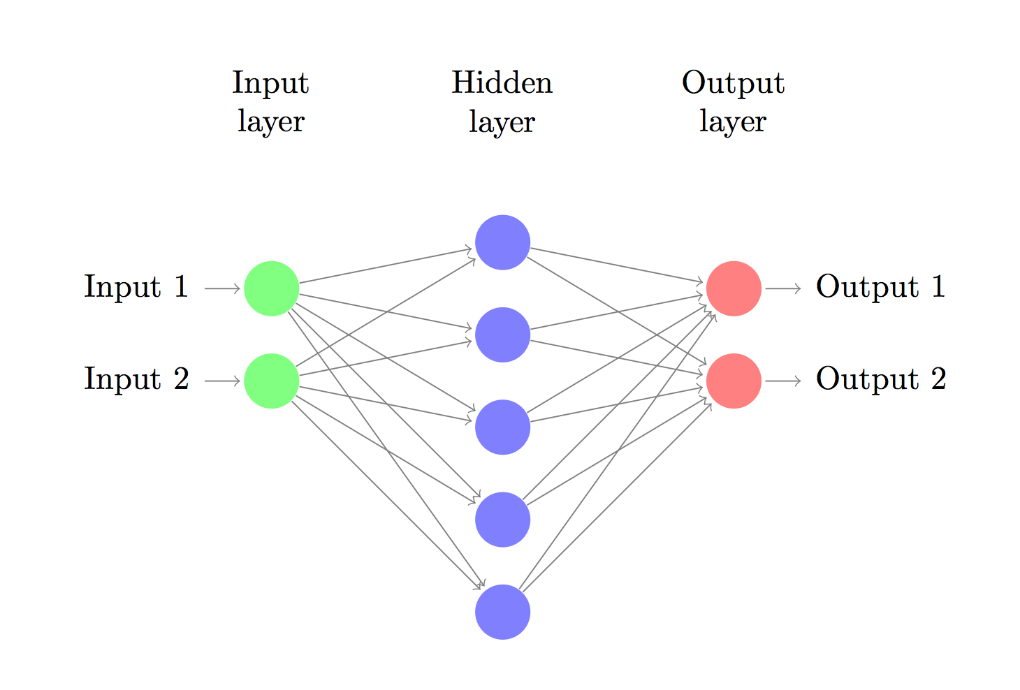


图4-5 三层网络结构

本实验采用Contrastive Loss (对比损失)计算loss，其计算公式为：

最终网络结构如图4-6所示。

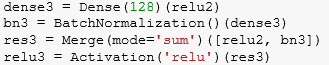
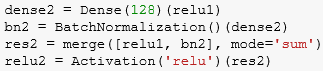
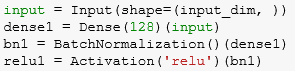


图4-6 网络结构

五、实验结果

如图5-1所示，为第一次交叉验证时，随着迭代次数的增加，正确率，准确率，召回率的变化情况。纵坐标表示百分比，横坐标表示迭代次数，总共迭代50轮。从图中可以看出，随着迭代的进行，性能逐渐提升并趋于平缓，最终收敛。50轮迭代结束，正确率为82.01%，本次交叉验证正确率的最大值为82.25%，全部实验中，正确率的最大值为82.39%。而准确率与召回率存在轻微的波动，并呈现出负相关关系，最终50轮迭代结束，准确率为75.89%，召回率为75.38%。

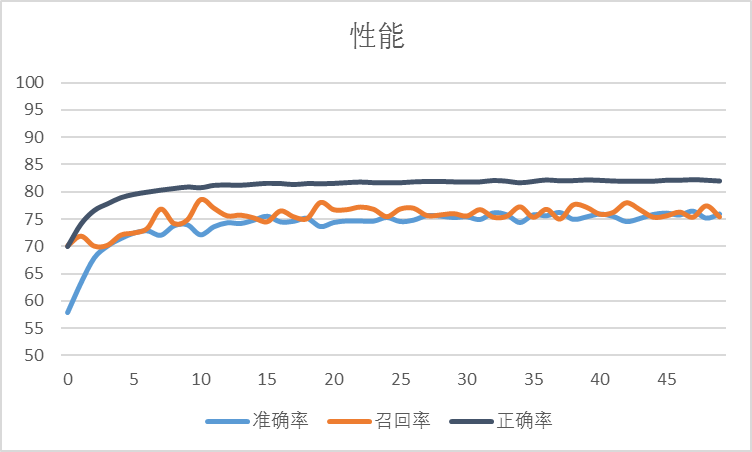


图5-1 第一次交叉验证性能指标

如图5-2所示，为第一次交叉验证时，随着迭代次数的增加，F1的变化情况。

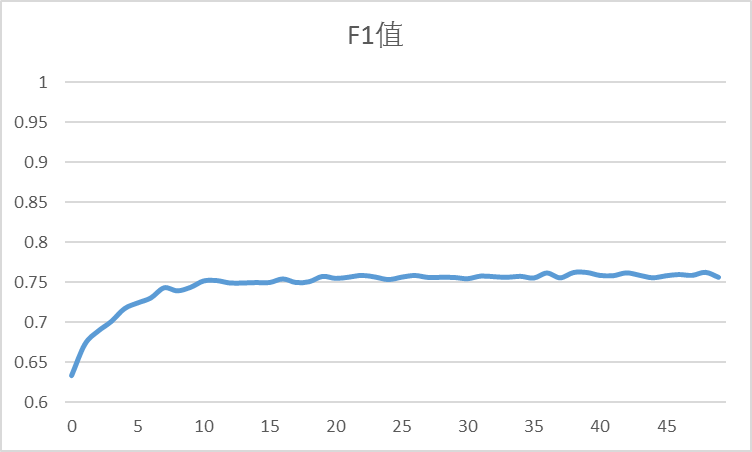


图5-2 第一次交叉验证F1值

从图中可以看出，随着迭代的进行，最终F1值收敛于0.76上下。第50轮迭代结束后，F1值为0.762553。

最终，经过10次交叉验证，每次交叉验证50轮迭代，每次迭代的F1值如表5-1所示，其中最大F1值为0.763701，平均F1值为0.758475。



表5-1 F1值性能

六、实验总结

本实验基于Keras搭建了三层神经网络，旨在解决Quora数据集上的重复问题检测。最终得出F1值为0.758475。

本实验的优点在于：

* 基于Keras进行深度学习，网络的搭建快捷
* 直接使用spaCy自然语言处理库，方便导出词向量空间
* 在词向量空间的基础上增加TF-IDF权值，提高了系统的性能

同时，本实验仍有可以继续改进的地方：

* 在数据统计阶段，得出数据集不平衡的结论，重复：不重复 = 1：1.708561，在后续实验中，由于不确定该比例是否对实验结果会产生影响，暂时没有进行处理。
* 统计中发现，问题的长度集中在10-30词之间，占比将近80%，根据实验的初始需求，未对另外20%的数据进行清理。
* 实验默认数据集给出标签真是可靠，但在Quora官网上指出，部分数据为认为制造，并且可能存在标签不真实。

本次实验搭建的神经网络借鉴于Siamese Networks（孪生网络），通过前期调研以及对网络的搭建，加深了小组成员对神经网络算法的认识，并能将其应用于问答领域的重复检测问题。实验过程中也遇到了各种问题，比如丢失规范层而导致结果不收敛，小组成员通过积极查阅资料，找到原因所在并解决问题。