**开课吧人工智能课学院**

**自然语言片与人工智能第5期 项目二报告书**

项目名：用户评论的细粒度情感分析系统

队名：冯・诺依曼队

在本项目中，我们实现了一个面向餐饮领域的用户评论的细粒度情感分析系统模型的构建及优化，并以WEB应用的形式编写了可以与用户进行交互的展示程序。

在实现项目目标的过程中，我们小组运用了多种从NLP课程中学习到的方法和技术，并根据本项目自身的特点，灵活地做了一些调整和优化，成功地达到了预期的目标。

但是，由于项目时间和规模有限，以及我们对自然语言处理技术还处在逐步的学习探索过程中，因此，实现的项目成果还有一些有待改进的地方。

以下，我们简单总结了一下我们小组的项目二成果的优点和不足之处。

**【优点】**

**1、选取的模型简单可靠**

在创建模型阶段，我们在充分理解并反复测试过baseline表现的基础之上，尝试过多种模型。这些模型包括：改进后的SVM、随机森林、DT树、朴素贝叶斯、Adaboost、Xgboost、多层textcnn、cnn+rnn、双向rnn+attention (加入attention)。在反复实验并观察评分结果和训练时间后，大致确定了以CNN为核心的模型结构。选定这种模型结构的主要考虑是，它的结构相对简单可靠，易于实现和修改，并且，训练时间和训练后实测的精确度都在可接收的范围内。对比其它模型结构，我们认为最终选定的模型结构是能在复杂度、精确度和训练时间三者之间取得平衡并达到系统整体最优的解决方案。因此，我们在这种模型结构的基础上，修改了代码进行实现，并以此作为baseline，进行后续更进一步的调参和测试。

**2、训练快，迭代快，调整快**

对于本项目的细粒度分类任务场景来说，模型训练所需的时间相对较长是一个很大的挑战。对于需要反复多次进行的【分析现象 -> 提出假设 -> 改变模型做实验 -> 观察是否符合期望】这样的调参过程来说，长时间的模型训练使得调参的结果很难在短时间内得到反馈，从而使得整个模型的迭代效率大大降低。

为了从根本上提高模型的优化迭代速度，我们从关键性的“模型训练时间”着手，采取了多种方法来提高模型训练的效率，有效果地减少了在此环节上的时间浪费，并提高了训练结果的精确度。我们总结分析了多次的训练过程，发现碰到的问题主要有以下3个：

1、loss函数为NAN （训练不收敛）

2、过拟合(overfitting)

3、loss函数下降慢（在有限时间内很难有效完成训练）

对于第一个问题，我们采用了适当减小batch size并减小学习率的手段来改善此问题。

对于第二个问题，我们增加了L2正则，使用了batch normalization和early stoppint的方法，并将适当增加了dropout层。

对于第三个问题，我们使用了学习率自适应下降（ lr-schedular）的方法来尝试解决。

经过以上努力，我们将训练的时间控制在了合理的范围内，并且使得训练结果的有效性在很大程度上得到了保证。这就为我们后续反复调整参数，优化模型精确度的工作打下了基础。

**3、灵活地改进了词向量化方法**

对于word embedding 阶段，我们经反复实验发现，将传统方法中的以“词”为单位的分词改为以“单字”为单位切分并进行向量化，取得的最终效果(F1 score分数)更好。

并且，我们还尝试过引入了微博词向量(<https://github.com/Embedding/Chinese-Word-Vectors>)。这对提升模型最终的表现也有一定帮助。

**4、有效运用开发框架和可视化库**

在将模型构建成WEB应用程序的过程中，我们使用了「Django」框架，使得前端和后端能在很短时间内搭建起来。这大大节省了搭建WEB应用所需时间精力，使得我们可以集中精力进行模型的持续改进工作中。并且，框架本身的功能结合互联网上的一些资源，使得前端的展示页面能够拥有漂亮简洁的UI界面。

在数据的可视化方面，我们使用了开源的「ECharts」库。ECharts库提供了多种数据可视化图表类型，包括非常适用于本次项目中表现【大分类-子分类形式】的饼图。使用ECharts库可以快速实现输出结果的可视化，使得用户可以直观清晰地看到情感分析的结果。

并且，ECharts库也提供了图表的交互功能，用户点击饼图的大分类时可以展开小分类的数据饼图。这使得用户可以方便快捷地确认大分类和小分类的数据。

**5、良好的交互效果**

为了提高用户的交互体验，我们设计了清晰友好的输入界面和简单明了操作方式。评论文本的情感分析后的输出结果以图表的形式非常直观地显示在WEB页面上。用户可以很方便很直观地确认到结果。并且，图表还附件了交互功能，用户点击饼图的大分类时可以展开小分类的数据饼图。这使得用户可以方便快捷地在不同的大分类和小分类的数据图表间快速切换。

并且，为了进一步提高交互的便捷程度，我们设计了两种不同的用户交互模式。

第一种是用户手动输入对餐馆的评价文本。在这种交互方式下，用户可以自行决定输入内容，复制粘贴到输入框，然后按下按钮即可获得摘要结果。这种交互方式允许用户选择指定的输入内容，从而能达到更好的有方向性的测试效果。

另一种方式是随机生成评价文本用于输入。用户在这种模式下不需要手动输入评价文本，只需要点击一下按钮即可得到输出的情感分析结果。自动输入的评价文本是来自事先保存的测试用数据集合。这种交互模式对用户非常友好，省时省力。用户可以在段时间内快速查看多次情感分析的结果。

**【有待改进的地方】**

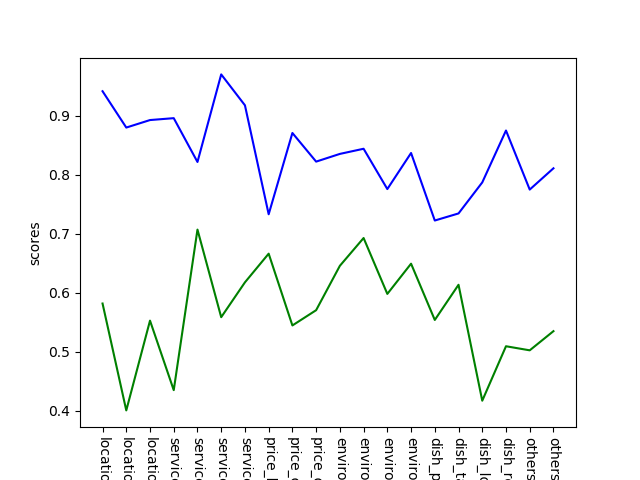
1. **F1 score分数不是非常理想。**

根据最终的测试结果，最终的F1 score分数为0.62，距离理想的准确度还有一定的距离。导致结果不是非常理想的原因主要有以下几个方面。

首先，用来训练语言模型的数据集的数据量比较有限。如果能够使用更大的数据量来训练模型的话，可能会对提高摘要准确度有帮助。

其次，受限于开发者的机器性能和部署服务器的性能，我们没有采用一些复杂度高耗时高但精度更好的模型结构。今后可以考虑使用算力更强大的服务器来实现更为复杂但效果更好的模型结构 。

还有，虽然在模型训练时间的优化上做了不少工作，但机器性能所限，反复进行的模型训练还是需要花费不少时间。由于项目二时间有限，所以无法再进行更进一步的参数调优了。如果时间允许，也许还有进一步优化参数的空间。



蓝色为acc，绿色为F1。

1. **性能问题。**

在使用训练好的模型来计算一段文本的情感分类时，由于需要跑20个模型（对于每个小分类标签都要计算），因此总共花的时间将近1分钟。用户的等待时间还是有点长的，这导致用户体验不是很好。我们认为这部分还存在可以改进的余地。比如，使用性能更好的部署服务器，或者优化计算逻辑等。

1. **训练数据集不均衡**

本次项目用于训练的数据集是AI Challenger 2018大赛官方提供的。但是在使用过程中，我们发现数据集中的数据的分布不是很均衡。有很多小分类在绝大多数和训练用评价文本中都未提及，也就是-2分居多，很小出现0分或1分。只有像口味、价格之类的少数小分类的提及度较高，也就是说从-2, -1, 0, 1, 2分的分布较为均衡。这导致了模型训练的结果受到了一定的影响。如果有机会使用更为均衡的训练数据的话，有可能能够使得最终的情感分析结果更为准确。

以上关于本次项目二成果的分析报告。

谢谢各位老师的指导和辛勤付出。

队名：冯・诺依曼队

成员：李鹏飞、吕汶颖、刘迪

时间：2020年1月13日