自然语言推理

Week16 Natural Language Inference

实验任务

- 实验内容
  - 使用给定数据集（QNLI限制长度所得的子集）完成识别文本蕴含任务
- 实验要求
  - 训练过程中某次测试准确率达到55%以上（不要求loss收敛）
  - 分析运行时间和其他指标
- 提交
  - 时限：7.2 晚 23:59，命名：E9_学号
  - 请控制文件大小，勿使用超大附件提交!
    - 请勿提交网上能下载到的预训练的词嵌入文件，在报告中说明使用的词嵌入即可

自然语言推理
Natural Language Inference

- NLI任务
  - 文本蕴含（textual entailment）
  - 指代消解（reference resolution）：识别文本中指代同一对象的词语
  - 问答（question answering）
  - ……
- 应用：问答、阅读理解、数学证明、……
- 识别文本蕴含（Recognizing Textual Entailment）
  - 本质上是分类问题
  - 给定前提premise和假设hypothesis，要求判断它们之间的关系
    - 三分类：不相干neutral、冲突contradiction、蕴含entailment
    - 二分类：不蕴含、蕴含
  - 常用数据集：MNLI，QNLI，RTE，WNLI等
    - MNLI为三分类任务，其余为二分类

QNLI

index	question	sentence	label
56	What causes rock extension? 岩石延伸的原因是什么？	This is primarily accomplished through normal faulting and through the ductile stretching and thinning. 这主要通过正断层和延性拉伸和减薄来实现。	entailment
7	What is the name of the professional skateboarder that lives in southern California? 居住在南加州的职业滑板运动员的名字是什么？	Southern California is also important to the world of yachting. 南加州对游艇界来说也很重要。	not_entailment

实验步骤

- 数据读取
- 数据预处理
- 词嵌入（Embedding）
- 数据对齐（Padding）
- 使用分类模型进行训练
- 输出分析

读取和预处理

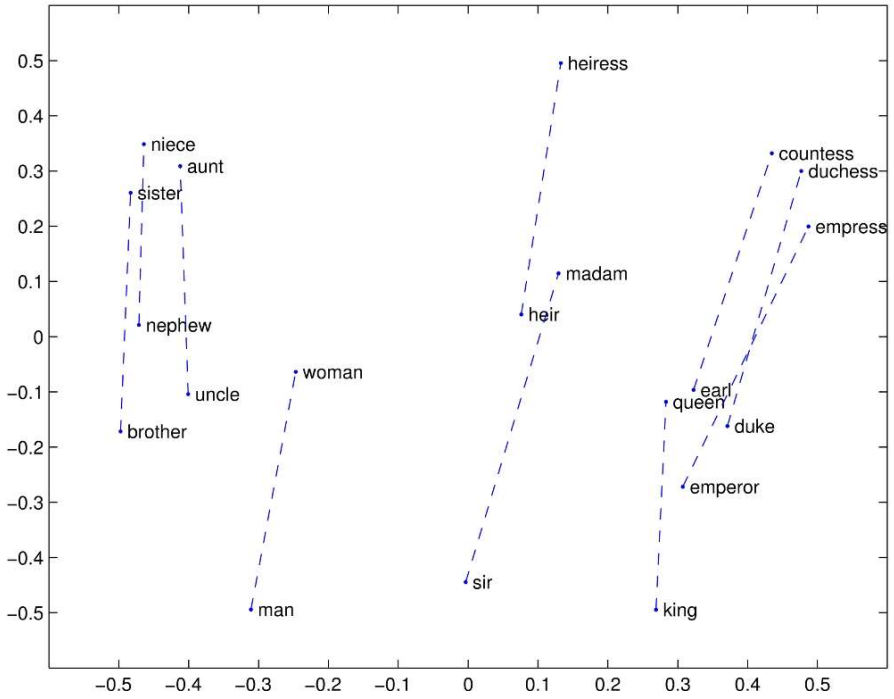
- 读取
  - 数据集每行四列，分别为index、question（前提句子）、sentence（假设句子）和label（entailment或not_entailment）
  - 格式为tsv（tab separated values），即四列用“\t”分开
  - `pandas`
- 预处理
  - 数据集较为杂乱（大小写、标点、其他语言文本），需要预处理
  - 分词（tokenization）
  - `nltk`、`keras`

词嵌入
Word Embedding

- 为了处理单词，需要将其用向量表示

- 能否像之前的实验一样使用one-hot或tf-idf?
  - 太过稀疏、占用空间太大
  - 需要手工构造词表计算
  - 难以捕捉语义上的相似信息

- 词嵌入
  - 将单词表示为固定长度的稠密向量
  - 不需要手工特征工程，结果具有泛化性
  - 语义相似的单词最终会具有相似的向量值



词嵌入
Word Embedding

- 工具
  - Word2Vec训练
  - GloVe（Global Vector）
    - GloVe: Global Vectors for Word Representation (stanford.edu)
    - 无监督的训练词向量的方法
    - 可直接下载训练好的词向量
  - BERT的embedding模块
    - Bidirectional Encoder Representation from Transformers
    - 一种预训练模型
    - base版本768隐藏层，对机器要求高

- 如何处理没出现的词？

词嵌入
Word Embedding

- 可选词向量的维数
- glove.6B.XXXd.txt

```
$ ls
glove.6B.100d.txt  glove.6B.200d.txt  glove.6B.300d.txt  glove.6B.50d.txt
$ head glove.6B.50d.txt
the 0.418 0.24968 -0.41242 0.1217 0.34527 -0.044457 -0.49688 -0.17862 -0.00066023 -0.6566 0.27843 -0.14767 -0.55677 0.1465
8 -0.0095095 0.011658 0.10204 -0.12792 -0.8443 -0.12181 -0.016801 -0.33279 -0.1552 -0.23131 -0.19181 -1.8823 -0.76746 0.09
9051 -0.42125 -0.19526 4.0071 -0.18594 -0.52287 -0.31681 0.00059213 0.0074449 0.17778 -0.15897 0.012041 -0.054223 -0.29871
 -0.15749 -0.34758 -0.045637 -0.44251 0.18785 0.0027849 -0.18411 -0.11514 -0.78581
, 0.013441 0.23682 -0.16899 0.40951 0.63812 0.47709 -0.42852 -0.55641 -0.364 -0.23938 0.13001 -0.063734 -0.39575 -0.48162
0.23291 0.090201 -0.13324 0.078639 -0.41634 -0.15428 0.10068 0.48891 0.31226 -0.1252 -0.037512 -1.5179 0.12612 -0.02442 -0
.042961 -0.28351 3.5416 -0.11956 -0.014533 -0.1499 0.21864 -0.33412 -0.13872 0.31806 0.70358 0.44858 -0.080262 0.63003 0.3
2111 -0.46765 0.22786 0.36034 -0.37818 -0.56657 0.044691 0.30392
. 0.15164 0.30177 -0.16763 0.17684 0.31719 0.33973 -0.43478 -0.31086 -0.44999 -0.29486 0.16608 0.11963 -0.41328 -0.42353 0
.59868 0.28825 -0.11547 -0.041848 -0.67989 -0.25063 0.18472 0.086876 0.46582 0.015035 0.043474 -1.4671 -0.30384 -0.023441
0.30589 -0.21785 3.746 0.0042284 -0.18436 -0.46209 0.098329 -0.11907 0.23919 0.1161 0.41705 0.056763 -6.3681e-05 0.068987
0.087939 -0.10285 -0.13931 0.22314 -0.080803 -0.35652 0.016413 0.10216
of 0.70853 0.57088 -0.4716 0.18048 0.54449 0.72603 0.18157 -0.52393 0.10381 -0.17566 0.078852 -0.36216 -0.11829 -0.83336 0
.11917 -0.16605 0.061555 -0.012719 -0.56623 0.013616 0.22851 -0.14396 -0.067549 -0.38157 -0.23698 -1.7037 -0.86692 -0.2670
4 -0.2589 0.1767 3.8676 -0.1613 -0.13273 -0.68881 0.18444 0.0052464 -0.33874 -0.078956 0.24185 0.36576 -0.34727 0.28483 0.
075693 -0.062178 -0.38988 0.22902 -0.21617 -0.22562 -0.093918 -0.80375
to 0.68047 -0.039263 0.30186 -0.17792 0.42962 0.032246 -0.41376 0.13228 -0.29847 -0.085253 0.17118 0.22419 -0.10046 -0.436
53 0.33418 0.67846 0.057204 -0.34448 -0.42785 -0.43275 0.55963 0.10032 0.18677 -0.26854 0.037334 -2.0932 0.22171 -0.39868
0.20912 -0.55725 3.8826 0.47466 -0.95658 -0.37788 0.20869 -0.32752 0.12751 0.088359 0.16351 -0.21634 -0.094375 0.018324 0.
21048 -0.03088 -0.19722 0.082279 -0.09434 -0.073297 -0.064699 -0.26044
```

对齐
Padding

- 我们可以用tensor的第n列表示句子中第n个词对应的向量
- 但是每个样本的序列长度不一样，导致维数不同
- 需要将样本对齐为一样的大小
  - 手动补全
  - `torch.nn.utils.rnn.pad_sequence`
  - `torch.nn`中的Padding Layers
  - `torch.nn.functional.pad`

- 可直接对所有数据进行对齐，也可对每个batch分别对齐

分类模型

- LSTM（Long Short Term Memory）
  - 带有门控单元的RNN，可捕捉时序信息
  - `torch.nn.LSTM`、`keras.layers.LSTM`
  - 在后面接单个线性层，输出大小为类别数
  - 本次实验中可作为黑盒使用

- 其他模型
  - RNN
  - CNN
  - ……

参考资料

- Python-Word2Vec模块使用详解：https://blog.csdn.net/qq_28840013/article/details/89681499
- GloVe使用以及词表文件详解 https://blog.csdn.net/ycq1041265011/article/details/110139729
- BERT教程 https://zhuanlan.zhihu.com/p/524487313
- PyTorch Custom Dataset教程 https://pytorch.org/tutorials/beginner/data_loading_tutorial.html
- GloVe+RNN（pytorch）https://zhuanlan.zhihu.com/p/562565880
- Glove+Bi-LSTM（pytorch）https://blog.csdn.net/qq_52785473/article/details/122800625
- GloVe+LSTM（keras）https://www.heywhale.com/mw/project/5b6956ea9889570010c33d54

可能的优化、创新点（仅供参考）

- 数据预处理
- 自行训练词向量（建议先做出达标版本）
- 自行添加规则（如：是否满足某种模板则可判断为蕴含）
- 不同预处理方法、词表、参数、模型大小、模型结构等的对比
- 其他可用在分类任务上的优化方法
- 在RTE、WNLI上做到55%+，或在完整的QNLI上做到62%+
- ……