语言分析与机器翻译课程学习报告

计算机科学与工程学院 计硕2006班 2001716 白刚

语言分析与机器翻译这门课给我的最大收获是让我认知了语言模型，并了解了机器翻译的基本原理和技术发展历史。通过与老师和同学的交流，尤其是跟肖桐老师，还有自然语言实验室那些优秀的博士，硕士，已经经过很多知识学习和实际项目历练的优秀同学的交流，让我认知到光学理论是远远不行的，理论很重要，但没有实际项目的实践历练，你很难写出哪怕几行代码，更别说是做一个实际任务，甚至

修改别人写的程序了。

课程的几节上机课，我认为都很重要和实用。首先让我们有一个基本的上机，访问服务器和程序调试技能，然后通过老师和博士生的讲解、指导和上机实验，让我们真正应用理论，体验项目。由简及深，

让我们通过成熟的项目、实际语言模型的训练和训练结果的提交，真正认识到了语言模型的意义，机器翻译技术由统计机器翻译到神经机器翻译的发展历史，这其中的缘由，基本原理和模型的改进、性能的

提升。

课程最后布置的模型改写和训练作业，逼迫我不得不通过实际动手做项目了解模型怎么来做，程序怎么写。实际这个过程真的很难，但真的非常非常实用。因为这就是我们以后在实验室乃至在毕业后会做的东西，必须掌握的能力和技术。学校的理论学习给了我们知识和理论水平，而把这些知识和理论应用于实际场景是我们以后真正的价值所在。

我所做的这个小项目是一个基于RNN模型的歌词训练和编写器。因为RNN擅长于处理序列数据，比如声音或文章，所以比较适合于语言模型。这里就是利用RNN来训练一个歌手的歌词风格模型，然后通过多轮训练，比较了两种不同的训练方法，利用一个平均交叉熵损失来衡量、比较这两种方法的性能优劣。

语言模型是自然语言处理的重要技术。自然语言处理中最常见的数据是文本数据。我们可以把一段自然语言文本看作一段离散的时间序列。假设一段长度为T的文本中的词依次为w1,w2,…,wT，那么在离散的时间序列中，wt（1≤t≤T）可看作在时间步t的输出或标签。给定一个长度为T的词的序列w1,w2,…,wT，语言模型将计算该序列的概率：

P(w1,w2,…,wT).

语言模型可用于提升语音识别和机器翻译的性能。

为了计算语言模型，我们需要计算词的概率，以及一个词在给定前几个词的情况下的条件概率，即语言模型参数。设训练数据集为一个大型文本语料库，如维基百科的所有条目。词的概率可以通过该词在训练数据集中的相对词频来计算。

n元语法通过马尔可夫假设简化了语言模型的计算。这里的马尔可夫假设是指一个词的出现只与前面n个词相关，即n阶马尔可夫链。如果n=1，那么有P(w3∣w1,w2)=P(w3∣w2)。如果基于n−1阶马尔可夫链，我们可以将语言模型改写为

P(w1,w2,…,wT)≈∏P(wt∣wt−(n−1),…,wt−1).

以上也叫n元语法。 它是基于n−1阶马尔可夫链的概率语言模型。

故项目构思和实现过程如下：

1.考虑到歌词之间存在语义相关性、连贯性，在模型训练时建立时间步模型，即每次输入一段歌词，把歌词的每个字符看作一个时间步，让上一个时间步t-1的隐藏层变量和当前时间步的输入共同决定当前时间步t的隐藏变量。隐藏层使用了激活函数（tanh，双曲正切函数）对二者连接后作非线性变换，再进行全连接。当前时间步的隐藏状态Ht也将参与下一个时间步t+1的隐藏状态计算，并同时输入到当前时间步的全连接输出层。

2.对每个时间步t的输出层使用softmax运算，转换成概率模型，使用交叉熵损失函数计算与标签的误差，每个标签是一个one-hot向量，也就是把所有训练歌词编成字典后，生成字典长度的向量，预测的歌词下一个字符的标签是该向量对应该字符索引处的分量为1，其余都是0.

具体项目流程：（具体代码见程序）

1.读取歌词数据集并生成字典、字符索引，根据字典和字符索引就可以在后面的处理中将该歌手歌词数据集里的任意字符转为one-hot向量用于训练模型的输入。

2.编写采样函数，目的是生成多个小批量的输入，后面用小批量随机梯度下降进行模型训练。将训练数据的歌词序列、对应的歌词下一个字歌词序列（也就是对应标签）转为多批对应索引序列，每批里有一个或多个连贯的歌词段用于RNN时间步模型的训练。

3.编写RNN模型。初始化隐藏层、输出层参数，还要有一个初始化的隐藏状态用于时间步之间隐藏层的传递。RNN模型主要就是处理多个时间步的各层输入、计算、输出，其中时间步之间t-1步隐藏层和t步的输入层各自计算，然后两层进行加和运算后共同决定t步的隐藏层，这样由Ht和Ht-1的关系可知，隐藏层能够捕捉到至该时间步之前的序列历史信息，就像有先前歌词的记忆一样。隐藏层使用激活函数。最后将所有时间步的输出作为一个数组输出，再将最后一个时间步的隐藏状态输出，用于后续模型处理。

4.编写预测函数。这是根据训练完的RNN模型，根据给出的前缀歌词段进行预测，在输出中将所有前缀歌词保留，后面的部分是使用前一个时间步的预测输出来作为当前时间步的输入进行预测的。这些预测也是根据前缀歌词连续的隐藏层传递作为记忆，与本时间步输入一起叠加来完成的。

5.编写模型训练函数。这里在前面RNN模型定义的基础上，采用了两种使用不同方法的采样函数各自训练最后进行损失对比来展示优劣。采用上文的RNN模型函数使用每批量所有歌词和对应标签进行训练，采用softmax处理每批量所有的模型输出，并与标签进行交叉熵损失计算，使用每个字符的交叉熵损失函数求和取平均作为每个小批量的优化目标函数，使用小批量随机梯度下降算法在每个小批量中对模型参数进行优化。所有批量都完成训练作为一轮，多轮训练。每隔一定轮数输出最后一轮的全量训练集合的平均损失，这个损失是将平均交叉熵损失做了一个指数运算处理便于进行观察。然后调用预测函数对给出的歌词前缀进行一定长度的歌词预测输出，也就是使用该模型进行歌词预测和编写。

两种不同的采样训练方法的说明和结果对比：

1.随机采样是不同训练小批次之间的歌词段没有语义或时序的联系，每个歌词段本身永远是有序的歌词，但歌词段之间是随机打乱顺序的，不是连续的几段歌词。这样无法用上一个批量最终时间步的隐藏状态来初始化本批量的隐藏状态，记忆并不强烈。所以每次随机采样前都要重新初始化隐藏状态。

2.相邻采样是相邻的两个随机小批量在相同行上的歌词语义上是相邻的（批量内的不同歌词段不一定相邻），这样就可以用上一个小批量最终时间步的隐藏状态来初始化本批量隐藏状态，从而使该批量输出也取决于上一批量。这样只需（在训练函数中）在每个完整数据迭代周期开始时初始化隐藏状态。但模型参数的梯度计算将依赖于所有串联的小批量.梯度计算开销将随迭代次数增加而越来越大。所以需要在每次读取小批量前将隐藏状态从计算图中分离出来（只要上一批量的隐藏状态参与计算，而不计算参数关于该隐藏状态的梯度，也就是只参与算总值，而不作为一层关于参数的复合函数）

3.结果对比：根据每多轮训练后最后一轮的全量数据平均损失对比，相邻采样结果更好。但实际感觉歌词语义性和连贯性上感觉最后结果是随机采样的更好。考虑相邻采样在多轮过后可能存在过拟合。这里没有计算在测试数据上的平均损失，而只衡量对比了训练数据上的。

课程及项目总结：

这门课程让我真正体会到了实践出真知。通过上机课程操作和实际项目构思、知识应用、调试改进、运行对比，让我真正理解了课程内容，不同学习算法的模型、效率优劣对比，体验到了从构思到开发到真正实现的应用经历，享受到了从发现到收获的艰难而美好的过程。

这个小项目还有一些不足和需要进一步改进的地方，这是在做到后期项目本身慢慢启发和提示给我的。比如上文提到的过拟合等问题。模型还有进一步改进的空间，比如可以尝试多层神经网络，或当下流行的其他神经网络模型，或进一步改进采样方法，以及尝试融合不同歌手、风格的歌词，进行风格化歌词创作等。在日后的学习和研究生涯中，我将以此为起点和契机，对这些问题进行深入解决和实践。也希望大家积极踊跃选修这门课程，学习实践，各显其能，让这门课程真正帮助到大家，也让这门课程越办越好！