口语语义理解大作业

520030910341，*520030910342*，*520030910344* 周骏东，柳纪宇，曾兆钰

1. 摘要

口语语言理解是任务型对话系统的核心组件，其目的在于从用户的话语中提取出用户的意图，以为后续模块提供信息，最终顺利完成用户的任务。在本次任务中我们使用由动作（act），语义槽（slot），槽值（value）三个部分构成的语义单元来描述用户的语义。在老师和助教提出的 baseline 方法的基础上，我们又做了如下改进：

2. baseline 方法介绍

baseline 模型主要基于 BIO 序列标注方法实现。输入的词序列在经过嵌入层转化成特征向量后输入 RNN 层中得到隐层信息，接着将隐层信息输入 FNN 中进行解码，得到预测结果和准确率。由预测结果以及序号与词之间的对应关系得到 slot-value 的标注结果。

3. 算法实现与改进

3.1. 预训练模型加载

预训练模型是在大量预先准备的数据上训练好的模型，因为它们已经学会了一些通用的特征表示。故可以 q 起到降低训练时间、提高模型效果等效果。在本次大作业中，我们分别将 bert 预训练模型'bert-base-chinese' 应用到了词嵌入和编码两个方面中。

在./model/bert_embedding.py 中，我们使用 bert 预训练模型代替了原先的嵌入层，将 input_ids 输入预训练模型后取最后 4 层隐层向量，将它们拼接在一起作为编码 RNN 的输入。最终得到的结果见1中的 Bert (Embed) 一栏。

在./model/bert_tagging.py 中，我们使用 bert 预训练模型代替了原先的嵌入层和编码层。为了让数据格式符合 bert 模型的输入要求，我们对 input_ids、tag_ids、tag_mask 都做了预处理，使用与模型匹配的 tokenizer 生成 input_ids，将 tag_ids 和 tag_mask 补充到固定长度。由于算力限制，我们没有得到该模型的最终结果。

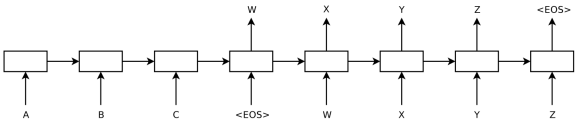

图 1: *Seq2Seq* 模型架构

3.2. Seq2Seq 模型

Seq2Seq 是一种用于序列到序列转换的深度学习模型。Seq2Seq 方法 [1]（见示意图1）与原序列标注方法最大的区别在于，我们只取编码器输出的最后一层信息作为解码器的原始输入，并将解码器的当前输出和隐层信息作为下一次解码时的输入。最后将每次解码的结果拼成一个输出张量经过 FNN 得到预测结果。

在./model/Seq2Seq.py 中我们实现了 Seq2Seq 模型的第一个版本，实验结果见1中的 Seq2Seq 及 Attention 栏。此外，为了进一步复杂化模型结构和简化接口，我们还在./model/Seq2Seq2.py 中编写了一个将编码器、解码器、结果输出层单独抽象出来的模型。

3.3. 注意力机制

当前的 Seq2Seq 模型存在以下问题：当输入序列较长时，将其转化为一个定长的向量会有较多的信息损失；从解码的角度看，某个位置解码的结果应当与输入序列对应位置附近的输入有更高的相关性，通过一个向量笼统地解码也会造成解码准确率的降低。

因此我们决定采用自注意力机制 [2] 对模型进行优化，在./model/attention.py 中我们编写了各种版本的注意力类，并在原先的 Seq2Seq 模型中

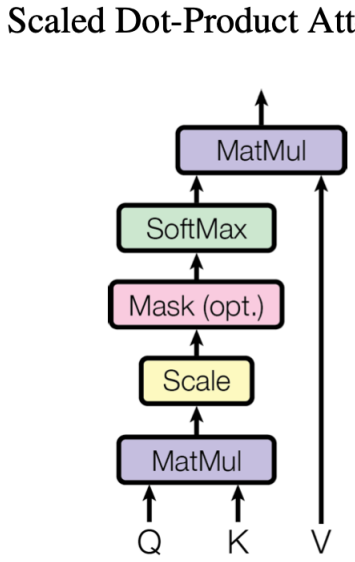

图 2: 添加在解码器上的自注意力层

加入了注意力层，使得解码器能够更好地分配输入序列的权重。

3.4. 指针网络

传统 Seq2Seq 模型在执行序列标注任务时依靠 ontology 构建的词表生成输出向量，这就使得模型在面对词表之外（OOV）的目标值时无法很好地完成任务。由于序列标注任务中输出的槽值往往是输入的子集，因此通过引入指向输入序列的指针，我们便能够从输入序列中选取词作为输出，从而更好地解决 OOV 槽值的问题。

指针网络 [3] 通过注意力层计算当前解码器输入下编码器上下文的权重，取权重最高的一项作为当前指针的位置。在解码器最终输出时，我们通过一定的比例权衡原词表输出与指针输出，从而达到既不浪费词表信息又充分利用输入信息的效果。在./model/pointer.py 中，我们实现了指针网络的解码器并将其整合进整个网络架构中。解码器的输入为嵌入词向量、编码器输出和最后一层的隐层状态，输出解码结果、指针位置和隐层信息。此处的实现部分参考了 [4] 中的代码。

尽管我们已经实现了模型的大体框架，但在结合原 RNN 输出与指针网络输出时我们遇到了一些困难。我们尝试过将指针网络输出补全到固定长度与 RNN 输出进行拼接，也尝试过通过 RNN 将指针网络的输出转化成与 RNN 输出维度相同的张量与其相加，但效果都不尽人意。经过一系列的调研，我们发现指针网络在执行序列标注任务时需要使用输入序列构建 OOV 词表拓展原先的词表，这就要求我们对数据的读取方式进行比较大的改动，由于时间和能力的限制，我们没有进一步完成指针网络的实现。

4. 结果展示

下表展示了所有上述方法的测试结果

算法	正确率	fscore
baseline	71.84	77.90
Bert (Embed)	65.36	68.84
Seq2Seq	70.06	77.21
Attention	70.84	75.51

表 1: 各算法测试结果

5. 总结

在本次大作业中，我们在 baseline 方法的基础上完成了 Seq2Seq 模型和注意力机制的实现以及 bert 预训练模型的使用。此外，我们还实现了指针网络的基本结构，但由于时间不足没有进一步实现与其适配的数据读取方式。

6. References

[1] Sutskever(2014): Sequence to Sequence Learning with Neural Networks

[2] Vaswani, Ashish and Shazeer: Attention is All you Need

[3] Abgail Lee, Peter J.liu, Christopher D.Manning: Get To The Point: Summarization with Pointer-Generator Networks

[4] https://github.com/shirgur/PointerNet