详解CTC

大师兄
澳门大学 计算机科学硕士

关注他

来自专栏 · 深度学习高手笔记 >

本文主要参考自Hannun等人在distill.pub发表的文章（distill.pub/2017/ctc/），感谢Hunnun等人对CTC的梳理。

简介

在语音识别中，我们的数据集是音频文件和其对应的文本，不幸的是，音频文件和文本很难再单词的单位上对齐。除了语言识别，在OCR，机器翻译中，都存在类似的Sequence to Sequence结构，同样也需要在预处理操作时进行对齐，但是这种对齐有时候是非常困难的。如果不使用对齐而直接训练模型时，由于人的语速的不同，或者字符间距离的不同，导致模型很难收敛。

CTC(Connectionist Temporal Classification[+])是一种避开输入与输出手动对齐的一种方式，是非常适合语音识别或者OCR这种应用的。

图1：CTC用于语音识别

给定输入序列 $\boldsymbol{X}=[\boldsymbol{x}_1,\boldsymbol{x}_2,..,\boldsymbol{x}_T]$ 以及对应的标签数据 $\boldsymbol{Y}=[y_1,y_2,..,y_U]$,例如语音识别中的音频文件和文本文件。我们的工作是找到 $\boldsymbol{X}$ 到 $\boldsymbol{Y}$ 的一个映射，这种对时序数据进行分类的算法叫做Temporal Classification。

对比传统的分类方法，时序分类有如下难点：

1. $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 的长度都是变化的；
2. $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 的长度是不相等的；
3. 对于一个端到端的模型，我们并不希望手动设计 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 的之间的对齐。

CTC提供了解决方案，对于一个给定的输入序列 $\boldsymbol{X}$ ，CTC给出所有可能的 $\boldsymbol{Y}$ 的输出分布。根据这个分布，我们可以输出最可能的结果或者给出某个输出的概率。

损失函数：给定输入序列 $\boldsymbol{X}$ ，我们希望最大化 $\boldsymbol{Y}$ 的后验概率 $\boldsymbol{P(Y|X)}$, $\boldsymbol{P(Y|X)}$ 应该是可导的，这样我们能执行梯度下降算法；

测试：给定一个训练好的模型和输入序列 $\boldsymbol{X}$ ，我们希望输出概率最高的 $\boldsymbol{Y}$ ：

$$\boldsymbol{Y^*=argmax_Y p(Y|X)} \tag{1}$$

当然，在测试时，我们希望 $\boldsymbol{Y^*}$ 能够尽快的被搜索到。

算法详解

给定输入 $\boldsymbol{X}$ ，CTC输出每个可能输出及其条件概率。问题的关键是CTC的输出概率是如何考虑 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 之间的对齐的，这种对齐也是构建损失函数的基础。
对齐方式，然后我们在分析CTC的损失函数的构造。

知乎 详解CTC

首发于
深度学习高手笔记

需要注意的是，CTC本身是不需要对齐的，但是我们需要知道 $\boldsymbol{X}$ 的输出路径和最终输出结果的对应关系，因为在CTC中，多个输出路径可能对应一个输出结果，举例来理解。例如在OCR的任务中，输入 $\boldsymbol{X}$ 是含有“CAT”的图片，输出 $\boldsymbol{Y}$ 是文本[C, A, T]。将 $\boldsymbol{X}$ 分割成若干个时间片，每个时间片得到一个输出，一个最简答的解决方案是合并连续重复出现的字母，如图2.

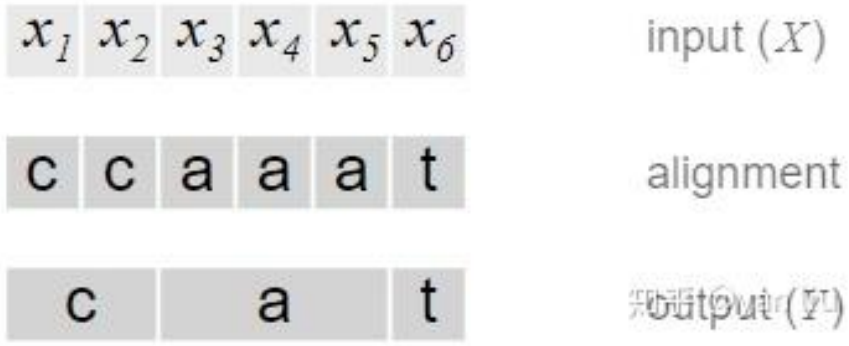


图2：CTC的一种原始对齐策略

这个问题有两个缺点：

1. 几乎不可能将 $\boldsymbol{X}$ 的每个时间片都和输出Y对应上，例如OCR中字符的间隔，语音识别中的停顿;
2. 不能处理有连续重复字符出现的情况，例如单词“HELLO”，按照上面的算法，输出的是“HELO”而非“HELLO”。

为了解决上面的问题，CTC引入了空白字符 ϵ，例如OCR中的字符间距，语音识别中的停顿均表示为 ϵ。所以，CTC的对齐涉及去除重复字母和去除 ϵ 两部分，如图3。

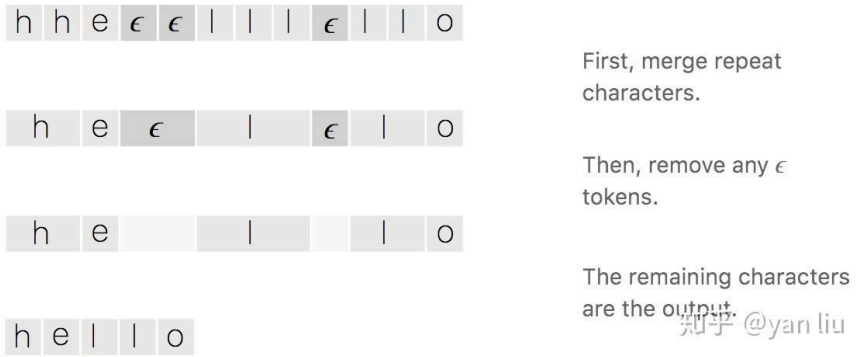


图3：CTC的对齐策略

这种对齐方式有三个特征：

1. $\boldsymbol{X}$ 与 $\boldsymbol{Y}$ 之间的时间片映射是单调的，即如果 $\boldsymbol{X}$ 向前移动一个时间片，$\boldsymbol{Y}$ 保持不动或者也向前移动一个时间片；
2. $\boldsymbol{X}$ 与 $\boldsymbol{Y}$ 之间的映射是多对一的，即多个输出可能对应一个映射，反之则不成立，所以也有了特征3；
3. $\boldsymbol{X}$ 的长度大于等于 $\boldsymbol{Y}$ 的长度。

1.2 损失函数

CTC的时间片的输出和输出序列的映射如图4：

关于作者

大师兄
《深度学习高手笔记》系列...

澳门大学 计算机科学硕士

深度学习（Deep Learning）话题的优秀答主

回答 36
文章 154
关注者 34,208

关注他 发私信

详解CTC

首发于
深度学习高手笔记

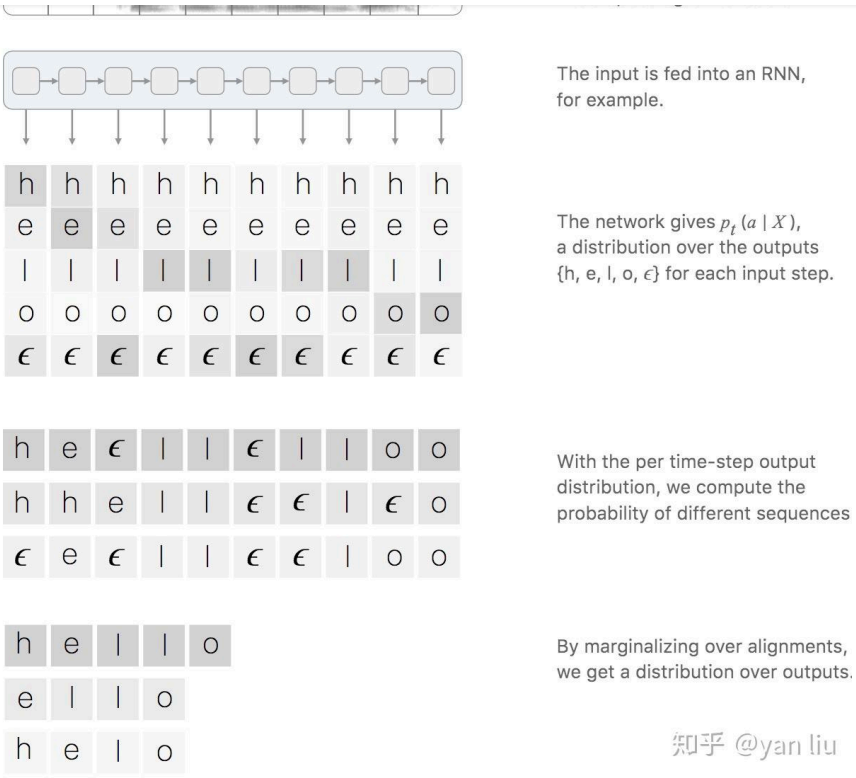


图5：CTC的流程

也就是说，对应标签 Y，其关于输入 X 的后验概率可以表示为所有映射为 Y 的路径之和，我们的目标就是最大化 Y 关于 $x=y$ 的后验概率 $P(Y|X)$。假设每个时间片的输出是相互独立的，则路径的后验概率是每个时间片概率的累积，公式及其详细含义如图5。

图6：CTC的公式及其详细含义

上面的CTC算法存在性能问题，对于一个时间片长度为 T 的 N 分类任务，所有可能的路径数为 N^T，在很多情况下，这几乎是一个宇宙级别的数字，用于计算Loss几乎是不现实的。在CTC中采用了动态规划的思想来对查找路径进行剪枝，算法的核心思想是如果路径 π_1 和路径 π_2 在时间片 t 之前的输出均相等，我们就可以提前合并他们，如图6。

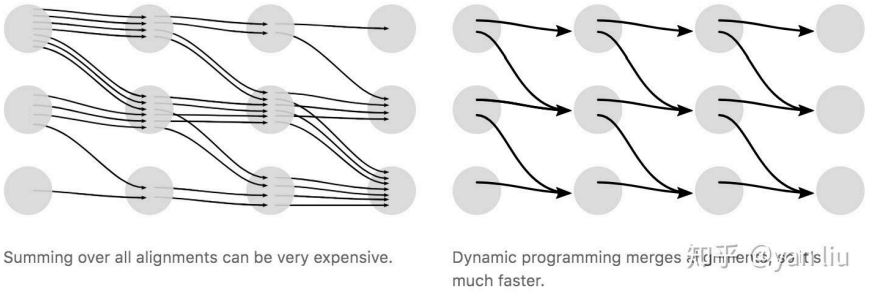


图6：CTC的动态规划计算输出路径

其中，横轴的单位是 X 的时间片，纵轴的单位是 Y 插入 ϵ 的序列 Z。例如对于单词“ZOO”，插入 ϵ 后为：

$$Z = \{\epsilon, Z, \epsilon, O, \epsilon, O, \epsilon\} \tag{2}$$

我们用 $\alpha_{s,t}$ 表示路径中已经合并的在横轴单位为 t，纵轴单位为方式的三个特征，输入有9个时间片，标签内容是“ZOO”，$P($

详解CTC

首发于
深度学习高手笔记

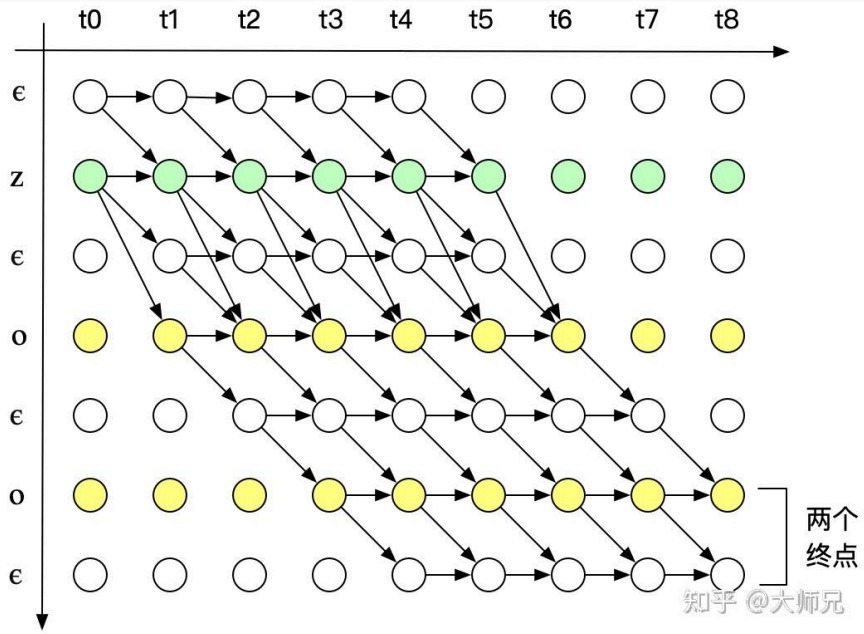


图7：CTC中单词ZOO的所有合法路径

上图分成两种情况

Case 1：

如果 $\alpha_{s,t} = \epsilon$， 则 $\alpha_{s,t}$ 只能由前一个空格 $\alpha_{s-1,t-1}$ 或者其本身 $\alpha_{s,t-1}$ 得到，如果 $\alpha_{s,t}$ 不等于 ϵ，但是 $\alpha_{s,t}$ 为连续字符的第二个，即 $\alpha_s = \alpha_{s-2}$ ，则 $\alpha_{s,t}$ 只能由前一个空格 $\alpha_{s-1,t-1}$ 或者其本身 $\alpha_{s,t-1}$ 得到，而不能由前一个字符得到，因为这样做会将连续两个相同的字符合并成一个。 $p_t(z_s|X)$ 表示在时刻t输出字符 z_s 的概率。

$$\alpha(s,t) = (\alpha(s,t-1) + \alpha(s-1,t-1)) \cdot p_t(z_s|X) \tag{3}$$

Case 2:

如果 $\alpha_{s,t}$ 不等于 ϵ，则 $\alpha_{s,t}$ 可以由 $\alpha_{s,t-1}$ ， $\alpha_{s-1,t-1}$ 以及 $\alpha_{s-2,t-1}$ 得来，可以表示为：

$$\alpha(s,t) = (\alpha(s,t-1) + \alpha(s-1,t-1) + \alpha(s-2,t-1)) \cdot p_t(z_s|X) \tag{4}$$

从图7中我们可以看到，合法路径有两个起始点，合法路径的概率 $p(Y|X)$ 是两个final nodes的概率之和。

现在，我们已经可以高效的计算损失函数，下一步的工作便是计算梯度用于训练模型。由于 $P(Y|X)$ 的计算只涉及加法和乘法，因此其一定是可导函数，进而我们可以使用SGD优化模型。

对于数据集 D，模型的优化目标是最小化负对数似然

$$\sum_{(X,Y)\in D} -logp(Y|X) \tag{5}$$

1.3 预测

当我们训练好一个RNN模型时，给定一个输入序列 X，我们需要找到最可能的输出，也就是求解

$$Y^* = \arg\max_Y p(Y|X) \tag{6}$$

求解最可能的输出有两种方案，一种是Greedy Search，第二种是

每个时间片均取该时间片概率最高的节点作为输出：

$$A^* = \arg\max_{A} \prod t = 1^T p_t(a_t|X) \tag{7}$$

这个方法最大的缺点是忽略了一个输出可能对应多个对齐方式.

1.3.2 Beam Search

Beam Search是寻找全局最优值和Greedy Search在查找时间和模型精度的一个折中。一个简单的beam search在每个时间片计算所有可能假设的概率，并从中选出最高的几个作为一组。然后再从这组假设的基础上产生概率最高的几个作为一组假设，依次进行，直到达到最后一个时间片，下图是beam search的宽度为3的搜索过程，红线为选中的假设。

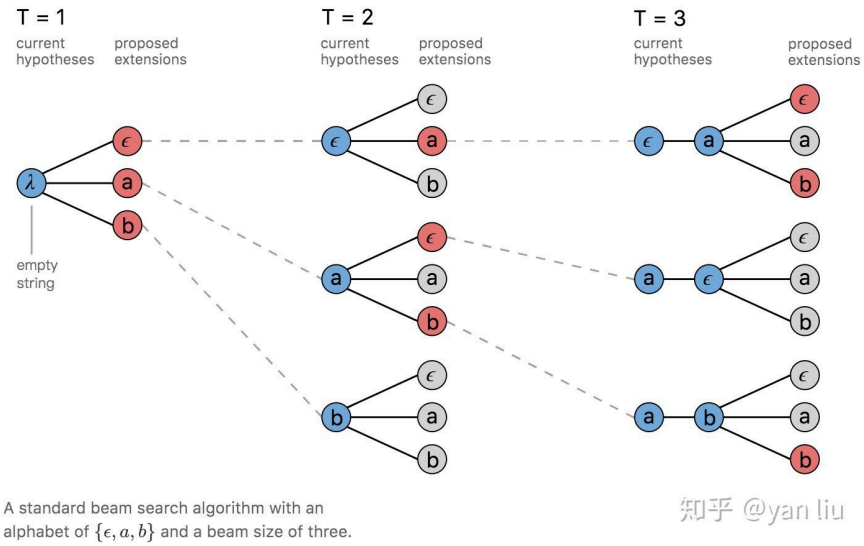


图8：Beam Search

CTC的特征

1. 条件独立：CTC的一个非常不合理的假设是其假设每个时间片都是相互独立的，这是一个非常不好的假设。在OCR或者语音识别中，各个时间片之间是含有一些语义信息的，所以如果能够在CTC中加入语言模型的话效果应该会有提升。
2. 单调对齐：CTC的另外一个约束是输入 X 与输出 Y 之间的单调对齐，在OCR和语音识别中，这种约束是成立的。但是在一些场景中例如机器翻译，这个约束便无效了。
3. 多对一映射：CTC的又一个约束是输入序列 X 的长度大于标签数据 Y 的长度，但是对于 Y 的长度大于 X 的长度的场景，CTC便失效了。

参考文献

[1] Connectionist Temporal Classification : Labelling Unsegmented Sequence Data with Recurrent Neural Networks. Graves, A., Fernandez, S., Gomez, F. and Schmidhuber, J., 2006. Proceedings of the 23rd international conference on Machine Learning, pp. 369--376. DOI: 10.1145/1143844.1143891

[2] Sequence Modeling with CTC. Hunnun, Awni, Distill, 2017

知乎 详解CTC

首发于
深度学习高手笔记

深度学习高手笔记
大师兄 深度学习（Deep Learning）话题下的优秀答主
161 篇内容 · 29959 赞同
订阅

最热内容 · 详解Transformer （Attention Is All You Need）

编辑于 2022-11-02 22:51

深度学习（Deep Learning） 语音识别 OCR（光学字符识别）

理性发言，友善互动

29 条评论 默认 最新

yudonglee
我在我的博客最近也写了一篇CTC的详细介绍，包括算法原理、理论推导、代码Demo实现等各个方面都有，希望对你有帮助。链接：xiaodu.io/ctc-explained...
2019-03-06 回复 12

lulupan
求导部分能详细解释下么，有一些定义没有交代
2020-10-21 回复 1

siggb
看了一样，文章写的很好，待细读，看起来一共要有五个部分，目前是只写完第一部分吗?
2019-03-07 回复 1

查看全部 9 条回复 》

知乎用户mw7gs3
"对于一个时间片长度为 T的 N分类任务，所有可能的路径数为 T^N"
路径总数不应该是N^T吗？每个时刻有N个可能，N*N...*N
2019-09-03 回复 7

袁小方 ‣ 李凤伟
看作英文字母分类，N=26，第N个是字母z的话，为什么不能取呢？比如zealous，第一个应该是z啊。假设从0开始，第N个是添加的-，-zealous也是合法路径
2020-03-09 回复 喜欢

李凤伟 ‣ 袁小方
不是这样的，并不是每个时间片都有N种可能，就比如第一个时间片就不能取第N个值
2020-03-09 回复 喜欢

展开其他 1 条回复 》

Ceres
图7的所有合法路径，我有点小疑问，为什么不能在t1的'z'和t2的'空'之间连线
2020-01-16 回复 4

大师兄 作者
谢谢提醒，这个图确实有问题，我后面仔细修订下
2022-11-02 回复 喜欢

西瓜王
看到了两个地方可能有错误：
（1）在1.2节中，对于一个时间片长度为T的N分类任务，总共路径数应该是N^T，而不是T^N.

知乎 详解CTC

首发于 深度学习高手笔记

也可能是我理解的不对。

2022-11-02 回复 1

大师兄 作者

（1）谢谢提醒，你说的没错

（2）这个图确实有问题，我后面仔细修订下

2022-11-02 回复 喜欢

闲人尔

不应该是多个输入可能对应一个输出吗？为什么是多个输出可能对应一个输入？

2022-06-27 回复 喜欢

Believer

您好，在case1的第一行话中，“只能由前一个空格”这里，应该不是“空格”，是“字符”吧？这是我自己的理解，我也不知道对不对

2022-06-02 回复 喜欢

Dreamer

感觉还是有个小地方没有解释清楚，当知道时间切片的长度和GT序列的时候，对于这个gt 的生成序列的范围就确定了，那预测结果和可以生成gt 的这个序列范围到底是如何产生关联生成loss的呢？希望明白的大佬解答一下

2022-05-08 回复 喜欢

lausvsa

谢谢佬的解释，想请教一个问题，既然ctc的前向计算是得到所有路径的概率，那么自己用pytorch实现ctc的时候，是不是只实现前向计算，得到所有路径概率和的值后调用backward()就可以自动反向传播呢

2022-03-13 回复 喜欢

大师兄 作者

应该是可以的

2022-03-13 回复 1

lausvsa ‣ **大师兄**

太感谢佬的解答了，抱歉再浪费点佬的时间请教一下佬一个小问题，我现在在训练模型的时候，得到ctc前面的linear classifier的输出（log softmax形式的），然后看代码里计算了ctc loss，得到的值是300多，，但是linear classifier输出的序列长度为600，我打印linear classifier的输出，看每个元素的值都是负的9点几，应该计算一种对齐的概率也有5000多了（（因为600x9）我自己在代码里试了一下，确实是5000多），但是它考虑了所有路径算出的ctc loss的值反而是300，比我一条路径手动加和的值反而小，请问这里有什么可能的原因吗

2022-03-13 回复 喜欢

青春作伴好还乡

感谢

2022-03-08 回复 喜欢

虫虫飞

很棒，感谢

2021-08-25 回复 喜欢

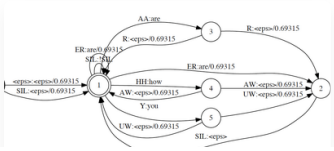
点击查看全部评论 ›

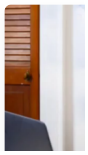
理性发言，友善互动

推荐阅读

CTC理解

看了这篇 Sequence Modeling with CTC搞懂了CTC，简单的记录一下。 CTC（Connectionist T





知乎 详解CTC

首发于
深度学习高手笔记