

看山杯 init 队伍解决方案*

陈云

北京市海淀区
西土城路 10 号
i@knew.be

代成

北京市海淀区
西土城路 10 号
15652965941@163.com

李作潮

北京市海淀区
西土城路 10 号
lzc123ok@163.com

摘要

在看山杯比赛中，我们队伍将深度学习的方法利用到了文本多分类问题之中，训练了多个差异较大的模型，这些模型都取得了相匹敌的优异成绩，并对模型融合提升巨大。我们还使用了多个模型联合训练（MultiModel）的方式，降低复杂模型的训练难度，提升模型融合分数。同时我们利用了数据增强的方法，并显著提升了模型融合的分，最终获得了第一名的好成绩。

关键词

文本分类, 深度学习

1 绪论

本次看山杯竞赛之中，我们利用深度学习在自然语言处理中的应用，训练出了数个效果的模型，并在融合之后取得了不错的成绩。在单模型方面，我们利用 TextCNN，TextRNN 和 TextRCNN 等模型进行文本分类。同时参照 TextCNN 的多尺寸卷积的思路，提出了类似 GooLeNet 的 Inception 结构的深度卷积模型。这些模型在分类上都取得了不错的成绩。在数据处理上，我们对原始数据进行了包括 shuffle 和 drop 等增强处理，对模型的融合分数提升也很明显。除此之外，我们还将 CNN，RNN 等模型进行联合训练，这样的模型，可在一定程度上缓解模型过拟合问题，而且对模型融合提升较大。

2 模型介绍

这次竞赛中，使用的模型大多是参照了已发表的文本分类论文，包括 TextCNN，TextRNN 和 TextRCNN，并参照 GoogLeNet 的结构提出了深度卷积网络 TextInception。

2.1 TextCNN

TextCNN 网络结构如图 1 所示，输入的字或词经过了 Embedding 之后得到一个三维的 tensor，再利用不同尺寸的 1

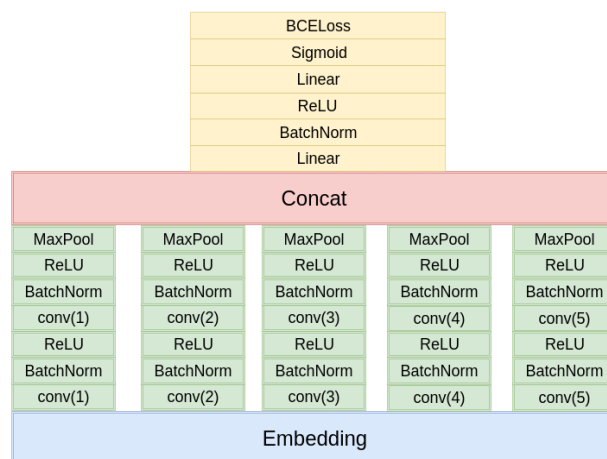


图 1: TextCNN 的网络结构.

维的卷积核对其进行卷积，提取局部特征。相比于原论文，我们做了一些改进，主要包括：

1. 卷积由 1 层变成两层
2. 卷积和激活函数之间使用 BatchNorm 而不是 Dropout
3. 全连接层由一层变成了两层，并使用 BatchNorm

另外这里也有处理不够合理的地方在于我们没有合理的设计卷积核的大小，不同分支的卷积核的感受野差距过大。

2.2 TextRNN

TextRNN 模型利用双向的 LSTM 提取句子的上下文信息和全局信息。与传统的 TextRNN 的主要区别，在于这里我们不是使用最后一个隐藏元作为分类，而是使用了所有的隐藏元进行 K-MaxPooling，然后利用全连接进行分类。这种做法可以看作是提取的每一个词的上下文信息，然后利用 MaxPooling

选择对分类最有效的词。相比于原始论文的做法，这种做法能够在分类的时候利用到更多的全局信息。

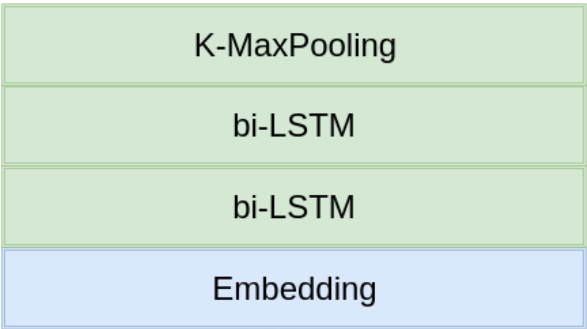


图 2: TextRNN 的网络结构。

2.3 RCNN

RCNN 的模型如图 3 所示，与 RNN 相比，RCNN 不仅使用了双向 LSTM 提取了每个词的上下文信息，还直接使用了 Embedding 获取词的信息。另外 LSTM 和 Embedding 的输出拼接之后直接进入 K-MaxPooling 之后进行卷积操作进一步的提取局部特征。与论文中不同，这里使用的卷积核大小为 2，而不是 1。使用卷积提取特征之后，继续将特征输入到由两层全连接网络组成的分类网络中进行分类。对于 char 的 RCNN 网络中使用了三层的双向 LSTM，用以提取更加深层次的信息。

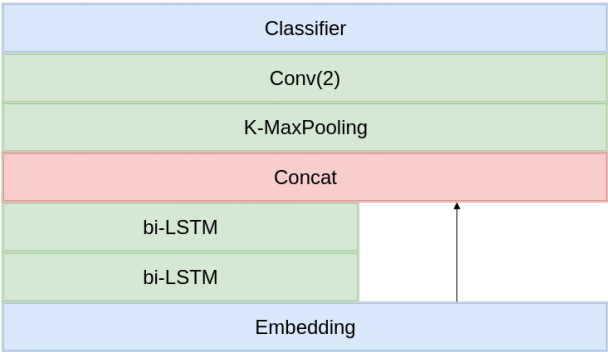


图 3: TextRNN 的网络结构。

2.3 TextInception

Inception 的结构主要是参照谷歌 GoogLeNet 的 Inception 结构，主要思路是不同尺寸的卷积核，提取词的局部信息。相比于 GoogLeNe，这里没有使用下采样技术，卷积尺寸设计也不仅

相同。单层的 inception 结构如图 4 所示，在比赛中，我们使用了两层的 Inception 结构，最多有 4 层卷积。

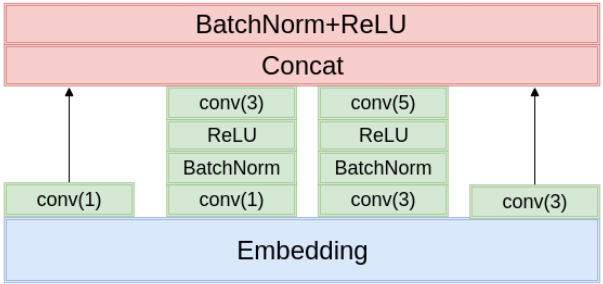


图 4：一个 Inception 单元的网络结构

2.3 MultiModel

MultiModel 是我们提出的一种特殊的模型训练方式，其架构如图 5 所示，分为共享 Embedding 和不共享 Embedding 两种做法。首先它利用预训练好的模型，计算样本属于每个类的概率，然后对这些概率进行累加求均值，继而计算 BCELoss。训练 MultiModel 分为共享 Embedding 和不共享 Embedding 两种方式。MultiModel 利用已经训练好的多个单模型作为它的子模型的初始值，然后利用较大的学习率或者强制使他们共享 Embedding 来使得模型的分数下降，以使模型走出过拟合区域，继而通过训练过程缓缓提升分数。如果不采用这种方式，模型的过拟合会很严重，难以学习到新的特征。

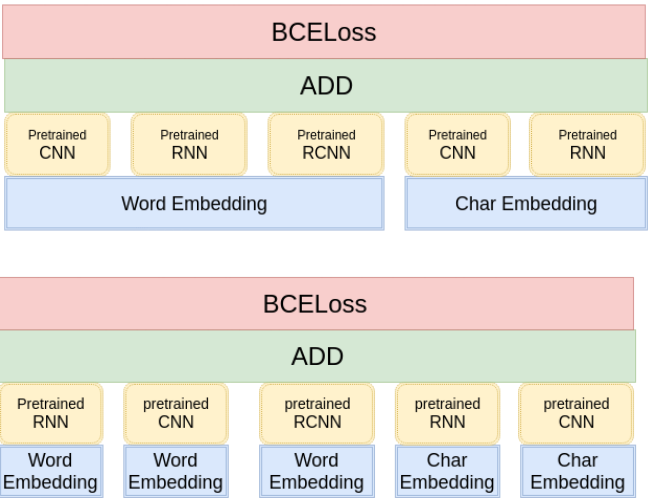


图 5: MultiModel 模型架构

3 实验与结果分析

3.1 数据增强

看山杯 init 队伍解决方案

在实验的时候，我们发现数据量的大小对模型的效果有较大的影响，因此我们对数据进行了增强处理。数据增强主要采取两种方法，一种是 `drop`，对于标题和描述中的字或词，随机的进行删除，用空格代替。另一种是 `shuffle`，即打乱词序。对于“如何评价 2017 知乎看山杯机器学习比赛？”这个问题，使用 `drop` 对词层面进行处理之后，可能变成“如何<s> 2017<s> 看山杯机器学习<s><s>”。如果使用 `shuffle` 进行处理，数据就可能变成“2017 机器学习？如何比赛知乎评价看山杯”。数据增强对于提升训练数据量，抑制模型过拟合等十分有效，而且能够提升模型的差异性从而获得更高的融合分数。

3.2 模型融合

本次实验中采用融合方式十分的简单，每个问题利用不同模型预测出它属于 1999 个类别的概率，乘以权重直接累加即可，绝大多数时候权重使用 1。不同的模型设计带来的结构差异性，数据增强处理方式带来的数据差异性，使得融合之后的分数提升十分明显。这种融合方式甚至比我们试验的 `stack` 方式还要出众，而且实现起十分简单。

3.3 结果

表 1 给出了各种单模型的分数的，注意这个分数是对 `Public` 分数的预估，笔者大多数单模型都未提交到线上评测，因此这个分数是根据模型在验证集上的分数估算出来。由于我们的验证集选取方式比较的特殊，使得验证集和线上 `public leaderboard` 的分数有 5-6 个千分点差距，笔者将验证集的分数增加 0.0053 计算得到下表的结果。可以看出，在不使用数据增强处理的时候，绝大多数的基于词训练的模型分数在 0.416-0.418 之间，基于字训练的模型的分数在 0.407-0.409 之间。采用数据增强处理之后，基于词的模型训练方式分数提升一些，但是基于字训练的模型分数下降严重。

表 1: 单模型分数

模型	类型	数据增强	分数
CNN	word	否	0.4155
RNN	word	否	0.4172
RCNN	word	否	0.4168
Inception	word	否	0.4162
RNN	char	否	0.4084
RCNN	char	否	0.409
Inception	char	否	0.4077
CNN	word	是	0.4158
RNN	word	是	0.4189
RCNN	word	是	0.4187
Inception	word	是	0.4178

CNN	char	是	0.3926
RCNN	char	是	0.4038

除了单模型之外，我们还统计了部分 `MultiModel` 的训练效果，如表 2 所示。`MultiModel` 包含多个子模型，这些子模型直接融合的分数的要大于训练 `MultiModel` 的分数，但是 `MultiModel` 训练出来的模型过拟合问题得到了缓解，并且对最终的模型融合有很大的帮助。我们只采用其中的 6 个 `MultiModel` 进行融合，就能达到 0.435 的分数，超过第二名。

表 2: MultiModel 联合训练的分数

包含的模型	共享 embedding	分数	备注
word: CNN, RNN, Inception char: RNN, Inception	否	0.4309	学习率较小,属于微调模型,训练出的分数实际上并不如直接融合的分数的使用了数据增强,并且采用了较高学习率,模型在训练之初分数严重下降而后缓慢提升
word: CNN, RNN, RCNN char: RCNN, CNN	否	0.4241	
word: CNN, RNN, RCNN char: RCNN	是	0.4288	同上
word: CNN, RNN, RCNN char: Inception, RCNN	是	0.4224	预训练模型采用弱模型(只训了一个 epoch)

3 总结

本次竞赛，我们的解决方案获得了第一名，并且与第二三名相比，有较明显的优势。在比赛中，我们发现数据增强，以及多模型联合训练(`MultiModel`)能够提升模型的差异性，对模型融合提升比较明显，甚至仅仅利用数据增强训练的多模型就能超过第二名的。同时我们也证明了，相较于 `Dropout`, `BatchNorm` 在自然语言处理的分类问题中一样能够取得不错的效果。并且在分类问题中，词语的顺序对于分类不是很关键。