万能的BERT连文本纠错也不放过

原创 孙树兵 夕小瑶的卖萌屋 2020-06-10 10:10:00 手机阅读

来自专辑

卖萌屋@自然语言处理

作者：孙树兵
学校：河北科技大学
方向：QA/NLU/信息抽取
编辑：小轶

背景

文本纠错（Spelling Error Correction）技术常用于文本的预处理阶段。在搜索引擎、输入法和 OCR 中有着广泛的应用。2020年的文本纠错自然也离不开 BERT 的表演。但原生的 BERT 在一些NLP任务如error detection、NER中表现欠佳，说明预训练阶段的学习目标中对相关模式的捕获非常有限，需要根据任务进行一定改造。在文本纠错任务中亦是如此。

此前文本纠错的SOTA方法采用了基于 Bert 的 seq2seq 结构，直接生成纠错后的字符序列。但是经观察发现，这样的方法总是倾向于不进行任何纠错，错误检测能力很低，一种可能的解释是 Bert 在预训练时只掩码了15%的字符，所以并不能够充分学习所有字符的上下文。

为了提高错误检测能力，本文在SOTA方法的基础上又添加了一个错误检测网络。分错误检测和纠正两步走。先检测每一个字的错误概率，然后根据检测结果将可能的错别字 soft-mask，其实就是错误概率：（**滑动查看完整公式**）

$$p \times |mask| embedding + (1-p) \times \text{原字符的 } embedding$$

再输给基于Bert的修正网络。这样就强制修正网络学习了错别字的上下文。下面将详细为大家介绍模型的实现细节。

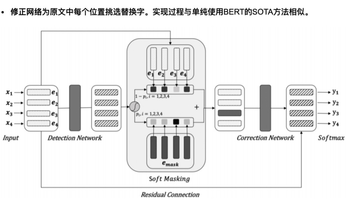
论文链接：https://arxiv.org/pdf/2005.07421.pdf

Arxiv访问慢的小伙伴也可以在订阅号后台回复关键词【0610】下载论文PDF。

模型结构

本文提出的 Soft-Masked Bert 模型可分为三个部分：

- **检测网络采用 Bi-GRU 预测字符在每个位置出现错误的概率。**
- **用错误概率对 input embedding 做 soft-mask。**soft-mask **是传统** hard-mask **的延伸。当错误概率等于1时，前者退化为后者。**
- **修正网络为原文中每个位置挑选替换字。实现过程与单纯使用BERT的SOTA方法相似。**



检测网络

检测网络是一个二分类的序列标注模型。模型的输入是 character embedding 序列 $E = (e_1, e_2, \ldots, e_n)$。其中 e_i 表示字符 x_i 的 character embedding（即word embedding, position embedding 和 segment embedding 的总和）。输出是标签序列 $G = (g_1, g_2, \ldots, g_n)$。$g_i$ 为第 i 个字符的标签，等于 1 表示字符错误，0 表示正确。我们记 p_i 为 g_i 等于 1 的概率。

本文采用双向 GRU(Bi-GRU) 实现检测网络。字符错误概率 p_i 可以定义为

$$p_i = P_d(g_i = 1|X) = \sigma(W_d h_i^d + b_d)$$

其中，$P_d(g_i = 1|X)$ 表示检测网络给出的条件概率，σ 是 sigmoid 函数，h_i^d 为 Bi-GRU 的隐状态，W_d 和 b_d 是参数。隐状态可以定义为：

$$\overrightarrow{h_i^d} = \text{GRU}(\overrightarrow{h}^d_{i-1}, e_i)$$
$$\overleftarrow{h_i^d} = \text{GRU}(\overleftarrow{h}^d_{i+1}, e_i)$$
$$h_i^d = [\overrightarrow{h_i^d}; \overleftarrow{h_i^d}]$$

Soft-Mask

soft-masked embedding 为 input embedding 和 mask embedding 的加权和。权重由该字符的错误概率得到。第i个字符的 soft-masked embedding 可形式化地定义为：

$$e_i = p_i \cdot e_{mask} + (1 - p_i) \cdot e_i$$

e_i 是 input embedding，e_{mask} 是 mask embedding。如果错误概率很高，则 e'_i 接近 e_{mask}。

修正网络

修正网络是一个基于 Bert 的多类别序列标注模型。输入为 soft-masked embedding 序列 $E = (e'_1, e'_2, \ldots, e'_n)$，输出为替换字符序列 $Y = (y_1, y_2, \ldots, y_n)$。

BERT 由12个相同的 block 组成。每个 block 包含一次 multi-head self-attention 操作和一个前馈神经网络。我们将BERT最后一层的隐状态序列记为 $H' = (h'_1, h'_2, \ldots, h'_n)$。则给定待纠错的字符序列 X，字符 x_i 被替换为候选字符表中第 j 个字符的条件概率为

$$P_c(y_i = j|X) = softmax(Wh'_i + b)[j]$$

其中，W 和 b 为参数；h'_i 是 e_i和 Bert 最后一层隐状态 h'_i 通过残差连接后得到的，即 $h'_i = h'_i + e_i$。校正网络的最后一层采用 softmax 函数，从候选字符列表中选择概率最大的字符作为字符作为输出。

训练过程

Soft-masked BERT 的训练是 Seq2seq 进行的。训练目标包括错误检测和错误纠正两部分，其目标函数分别为：

$$\mathcal{L}_d = -\sum_{i=1}^{n} \log P_d(g_i|X)$$
$$\mathcal{L}_c = -\sum_{i=1}^{n} \log P_c(y_i|X)$$

总目标函数为两者的线性组合：$\mathcal{L} = \lambda \cdot \mathcal{L}_c + (1-\lambda) \cdot \mathcal{L}_d$。其中$\lambda \in [0, 1]$。

实验结果

Test Set	Method	Detection				Correction			
		Acc.	Prec.	Rec.	F1.	Acc.	Prec.	Rec.	F1.
SIGHAN	NTOU (2015)	42.2	42.2	41.8	42.0	39.0	38.1	35.2	36.6
	NCTU-NTUT (2015)	60.1	71.7	33.6	45.7	56.4	66.3	26.1	37.5
	HanSpeller++ (2015)	70.1	**80.3**	53.3	64.0	69.2	**79.7**	51.5	62.5
	Hybird (2018b)	-	56.6	69.4	62.3	-	-	-	57.1
	FASPell (2019)	74.2	67.6	60.0	63.5	73.7	66.6	59.1	62.6
	Confusionset (2019)	-	66.8	73.1	69.8	-	71.5	59.5	64.9
	BERT-Pretrain	6.8	3.6	7.0	4.7	5.2	2.0	3.8	2.6
	BERT-Finetune	80.0	73.0	70.8	71.9	76.6	65.9	64.0	64.9
	Soft-Masked BERT	**80.9**	73.7	**73.2**	**73.5**	**77.4**	66.7	**66.2**	**66.4**
News Title	BERT-Pretrain	7.1	1.3	3.6	1.9	0.6	0.6	1.6	0.8
	BERT-Finetune	80.0	65.0	61.5	63.2	76.8	55.3	52.3	53.8
	Soft-Masked BERT	**80.8**	**65.5**	**64.0**	**64.8**	**77.6**	**55.8**	**54.5**	**55.2**

在 SIGHAN 和 News Title 两个数据集上进行了实验。本文的 Soft-Masked BERT方法在两个数据集上基本都取得了最好结果。

总结

本文提出了一种新的神经网络结构Soft-masked Bert，实现中文文本纠错。该结构包含错误检测和修正两个部分。通过Soft-mask技术将检测结果编码到修正网络。实验结果表明该方法的性能优于单纯使用Bert的基线模型。并且这一方法具有较强的普适性，也可用于其他语言的纠错任务。

本文收录于原创专辑：

《卖萌屋@自然语言处理》

重磅惊喜：卖萌屋小可爱们苦心经营的 自然语言处理讨论群 成立三群啦！扫描下方二维码，后台回复「入群」即可加入。众多顶会审稿人、大厂研究员、知乎大V以及美丽小姐姐（划掉）等你来撩哦~（手慢无

可能喜欢

- 告别自注意力，谷歌为Transformer打造新内核Synthesizer
- NLP中的少样本困境问题探究
- ACL20 | 让笨重的BERT问答匹配模型变快！
- 7款优秀Vim插件帮你打造完美IDE
- 卖萌屋原创专辑首发，算法镇魂三部曲！
- GPT-3诞生，Finetune也不再必要了！NLP领域又一核弹！

夕小瑶的卖萌屋

关注&星标小夕，带你解锁AI秘籍

订阅号主页下方「撩一下」有惊喜哦

点击查看精选留言