2021“曙光杯”参赛过程简述及部署说明*

柳明§ 王子文†, § 刘欢 张楚之
School of CEI/AI, Nanjing Normal University

1. 课题背景

公文写作是政府机关日常办公过程中的重要活动，承担着传达政策指令、讲话精神、时事情报等重要任务。书写规范正确是一篇好的公文的基本要求，公文发布前，经常需要写作者花费大量精力仔细校对和检查其中的书写错误情况，错误主要有同音字错误、形近字错误、多字少字这三种。通过自然语言处理技术，将文本纠错过程自动化，可以节省用户大量精力，提高写作质量。

2. 对若干环节的简述

本小节结合相关资料就我们的参赛准备工作的若干环节做一些简单的陈述。

开发环境. 我们将开发环境的相关信息列在表1中。

描述项	信息
操作系统	Ubuntu 18.04.5 LTS
程序语言	Python (≥3.6)
开发工具	PyCharm 2021.2.3 (Educational liscense)
	Visual Studio Code 1.62.3
服务器配置	TITAN X (Pascal) (12G)
	GeForce RTX 3090 (31G)

表 1. 开发环境信息。

技术选型. 经过分头查阅资料之后汇总讨论，我们选取了百度飞桨开发者技术专家徐明的开源工程 Pycorrector [6]作为项目的原型。并且采取 MacBERT 模型 [2, 3]，它使用整词掩码技术（Whole Word Masking）和 N-gram 掩码技术选择待掩码的标记（token）以适配中文表达，和通过用其相似的单词来掩盖单词的方法，缩小了训练前和微调阶段之间的差距，受 ALBERT [5]启发，MacBERT 同样也将 BERT [4]中的下一个句子预测任务（Next Sentence Prediction, NSP）替换为了句子顺序预测任务（Sentence Order Prediction, SOP）。基于此类预训练模型

§Equal contribution to this work.

†Captain of the team.

*本文档的编写基于 CVPR'11 论文 [1]的 LaTeX 模板。

的预训练任务中都有 MASK 掩码的特征，可以简单改造预训练模型用于纠错。我们准备对赛事方提供的数据集进行清洗和处理之后再结合纠错模型做 fine-tuning 和其他的细节工作。

数据集. 赛事方提供了 1000 条训练数据作为参考，我们对其经过数据清洗和处理后转化为 1000 个错误-正确句子对集合并标注出错误位置，便于直接读取后进行训练。考虑到因样本容量过小而可能带来的局限性，并且出现的错误也有其他一些非专业词汇、较贴近日常生活用语的错误，我们将 Sugon_data 数据集（即赛事方提供的数据集）和 SIGHAN 2013-15 的数据集 [7, 9, 10]做了融合。之后我们将整个数据集按比例切分之后进行训练，保存 fine-tuning 训练后的模型文件。后续加载模型文件即可用于纠错任务的预测。

3. 小结

通过加载 fine-tuning 之后的模型文件对一些用例进行测试，我们观察到纠错的效果基本上差强人意，多数常见错误都能够被发现，但是也存在不少符合语法、只是用词不当的错误并未被判断出来。我们认为这很可能是由于训练集中公文写作这一垂直领域的语料不够丰富，很多较专业的用词的语义也就相应地很难被模型充分学习到，造成了面对此类错误，我们的模型纠错效果还做得并不够好。

经过一周多的结果导向型学习，我们也有了一定的收获。在实践上，我们对近年来占据主流的的“预训练 + 精调范式”(pre-training+fine-tuning) 有了初步的认识和了解，通过在大量 task-free 的语料上训练出一个通用的预训练语言模型，再针对具体的下游任务引入各种辅助 loss 和 task-specific 的数据，然后继续训练，可以让我们的预训练语言模型更加地适配下游任务。

我们也更加深刻地体会到了具体问题具体分析的重要性。面对具体任务首先要了解、分析数据与相应的任务要求及其内容，对数据集的分布、实际情境中的情况都要先认识清楚之后，才能更好地利用相应的模型和方法去处理问题。

4. 展望

在比赛的准备过程中和最后阶段工程重构、资料整理的时候，我们也做了一些自己的思考，认为这项工作在以下一些方面应当还有很大改进的空间。

数据集. 理论上来说，公文这一领域有着相当庞大的语料资源，但是这些资源往往并未被标注过，可靠数据集的有限性也限制了许多 SOTA 的监督学习模型的使用，针对纠错任务的效果很难达到预期。

腾讯汪鼎民等 [8]关于 CSC（Chinese Spelling Checking）任务的这一痛点提出了一种自动生成语料的混合方法：爬取人民日报的文本（文本是经过很多编辑、校对筛选后登报的，可以近似认定为是正确的），在每个句子中随机选择 1 到 2 个字被 OCR 检测，对转化的图片进行部分模糊处理再让 OCR 多次检测，选出 OCR 识别出错的结果，就相当于生成了一条形近字错误样本；选取一个公开的普通话演讲语料库，其中包含大约 14 万条句子，用自动语音识别技术将演讲音频识别转化为文本，与演讲对应的语料文本对比，选择不一致的作为音近字错误样本。汪鼎民等在 GitHub 上公开了含 271329 条句子的数据集（Wang271K 数据集），因为其中含有大量源于人民日报的新闻数据，对模型学习公文领域一些专有名词的语义应该会有帮助，所以我们在下载并处理之后尝试将 Sugon_data 数据集与其融合再对模型进行训练，但受限于硬件条件和时间因素，我们没有完成这个想法。

多模型的实验对比. 字节 AI-Lab 与复旦大学在 2020 年提出的 Soft-Masked BERT [11]对 baseline 方法做了很多改进，主要的创新点有两个：将整个网络模型划分为检测网络（Detection）和纠正网络（Correction）两部分，检测网络的输出作为纠正网络的输入；以检测网络的输出作为权重，将 masking-embedding 以 “soft 方式” 添加到各个字符特征上。Soft-Masked BERT 在所选取的两份数据集上几乎都取得了最好结果，实验结果同时也证明，fine-tune 对于原始 BERT 的表现具有巨大的促进作用。

按照预先的设想，我们应当基于融合后的数据集构建一个测试数据集，评估不同模型如 BERT、MacBERT 和 Soft-Masked BERT 精调之后在其上面的表现，也很可惜并没有完成这个计划。

专有名词. 我们发现提供的训练样本中还有一些是针对公务人员姓名和行政机构及非政府组织相关的名词中的错字纠正，如将 “蔡隆翔” 改为 “蔡龙翔”，将 “皇位彬” 改为 “黄炜彬”。经过思考我们也并未想出比较有效的解决思路，爬取公务人员姓名构建混淆集的方法似乎也并不现实，关于这点仍有继续思考的空间。

参考文献

[1] M.-M. Cheng, N. J. Mitra, X. Huang, P. H. S. Torr, and S.-M. Hu. Global contrast based salient region detection. In *IEEE CVPR*, pages 409–416, 2011. 1

[2] Y. Cui, W. Che, T. Liu, B. Qin, S. Wang, and G. Hu. Revisiting pre-trained models for Chinese natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 657–668, Online, Nov. 2020. Association for Computational Linguistics. 1

[3] Y. Cui, W. Che, T. Liu, B. Qin, and Z. Yang. Pre-training with whole word masking for chinese bert, 2021. 1

[4] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 1

[5] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*, 2019. 1

[6] X. Ming. pycorrector: Text error correction tool, 2021. 1

[7] Y.-H. Tseng, L.-H. Lee, L.-P. Chang, and H.-H. Chen. Introduction to sighan 2015 bake-off for chinese spelling check. In *Proceedings of the Eighth SIGHAN Workshop on Chinese Language Processing*, pages 32–37, 2015. 2

[8] D. Wang, Y. Song, J. Li, J. Han, and H. Zhang. A hybrid approach to automatic corpus generation for chinese spelling check. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2517–2527, 2018. 2

[9] S.-H. Wu, C.-L. Liu, and L.-H. Lee. Chinese spelling check evaluation at sighan bake-off 2013. In *Proceedings of the Seventh SIGHAN Workshop on Chinese Language Processing*, pages 35–42, 2013. 2

[10] L.-C. Yu, L.-H. Lee, Y.-H. Tseng, and H.-H. Chen. Overview of sighan 2014 bake-off for chinese spelling check. In *Proceedings of The Third CIPS-SIGHAN Joint Conference on Chinese Language Processing*, pages 126–132, 2014. 2

[11] S. Zhang, H. Huang, J. Liu, and H. Li. Spelling error correction with soft-masked bert. *arXiv preprint arXiv:2005.07421*, 2020. 3

部署说明

在 Linux 环境中，进入工程主目录下，执行命令安装依赖。

```
cd macbert/
pip install -r requirements.txt
```

将输入数据 input.txt 放至 macbert/目录下，之后执行

```
python3 macbert_corrector.py
```

则在该目录下生成名为 output.txt 的输出数据文本文件。