计算机研究与发展
Journal of Computer Research and Development
https://crad.ict.ac.cn
62(3): 682−693, 2025

玲珑:一个小规模的高质量中文预训练语言模型

李东闻[1] 钟震宇[1] 孙羽菲[1] 申峻宇[1] 马子智[1] 于川越[1] 张玉志[1,2]
[1](南开大学软件学院 天津 300450)
[2](先进计算与关键软件(信创)海河实验室 天津 300450)
(lidongwen@mail.nankai.edu.cn)

LingLong: A High-Quality Small-Scale Chinese Pre-trained Language Model

Li Dongwen[1], Zhong Zhenyu[1], Sun Yufei[1], Shen Junyu[1], Ma Zizhi[1], Yu Chuanyue[1], and Zhang Yuzhi[1,2]
[1](*College of Software*, *Nankai University*, *Tianjin* 300450)
[2](*Haihe Laboratory of Information Technology Application Innovation*, *Tianjin* 300450)

Abstract In recent years, large-scale autoregressive Chinese pre-trained language models (PLMs) have demonstrated outstanding performance on various natural language processing (NLP) tasks. However, these models are computationally expensive, and their word-based vocabulary poses significant challenges for practical applications. In addition, most of them use only unidirectional context information, which may result in performance degradation on many tasks, especially tasks requiring a nuanced understanding of context. To address these challenges, we introduce LingLong, a high-quality small-scale Chinese pre-trained language model. LingLong stands out due to its modest scale, comprising only 317 million parameters, making it highly deployable and resource-efficient. We tokenize the training corpus with a character-based vocabulary to mitigate the negative impacts of unknown tokens and word segmentation errors. Moreover, we go beyond the conventional unidirectional context by introducing a novel backward model. This model is trained by reversing the input order of the training data. Combining LingLong and its backward version allows for the use of bidirectional information on downstream tasks. Extensive experimental results validate the effectiveness of LingLong across a diverse set of NLP tasks. LingLong outperforms similar-sized Chinese PLMs on six downstream tasks and surpasses popular large-scale Chinese PLMs on four downstream tasks. These findings underscore the versatility and efficiency of LingLong, opening up possibilities for practical applications and advancements in the Chinese NLP field.

Key words Chinese pre-trained language model; small-scale; character-based model; backward model; bidirectional information

摘 要 近年来,大规模的、基于自回归的中文预训练语言模型在各种自然语言处理任务上表现出优异性能.然而,高昂的计算成本以及基于中文词切分数据给中文预训练语言模型实际应用带来了巨大挑战.此外,大多基于自回归的模型只能使用单向前文信息,可能会导致模型在上下文敏感任务上的性能有所下降.为了解决以上问题,提出并训练了一个高质量的小型中文预训练语言模型——玲珑.该模型仅有3.17亿个参数,较小的规模使得玲珑十分容易部署和应用.使用基于汉字的策略对训练语料进行切分,可以有效减轻未知标记和分词错误带来的负面影响,增强了玲珑在下游任务上的性能.此外,通过对每条训练数据的输入顺序进行逆序处理,训练了一个反向玲珑模型.将玲珑与其反向版本相结合,可以实现在下游任务中使用双向信息.多种自然语言处理下游任务的实验结果表明,玲珑具有不错的处理下游任务的能力.在6个数据集上玲珑超越了相近规模模型的性能,在5个数据集上超越了大模型的性能.

收稿日期:2023-10-31;**修回日期**:2024-03-22
通信作者:孙羽菲(yufei_sun@sina.com)

关键词 中文预训练语言模型；小规模；基于汉字的模型；反向模型；双向信息

中图法分类号 TP183

DOI: 10.7544/issn1000-1239.202330844 **CSTR:** 32373.14.issn1000-1239.202330844

得益于深度学习模型强大的特征提取和识别能力以及大规模、高质量文本中蕴含的丰富知识，预训练语言模型(pre-trained language models, PLMs)在多种自然语言处理(natural language processing, NLP)下游任务上表现出了卓越性能[1-7].

与有监督学习模型相比，预训练语言模型能够充分利用大规模、无标注数据来学习通用语言特征，具备一定的常识和认知泛化能力. 通过少试(few-shot)、单试(one-shot)甚至零试(zero-shot)学习就能完成各种自然语言处理下游任务. 近年来，越来越多以中文为基础的预训练模型不断涌现[8-14]. 然而，中文预训练语言模型在实际应用中仍面临着许多挑战.

1)在资源有限的环境中部署大型预训练语言模型十分困难. 由于预训练语言模型的性能通常与模型规模成正比[6]，因此许多中文预训练语言模型致力于使用大量的训练数据训练具有更多参数的模型. 例如，首个大型中文预训练语言模型[11]具有26亿个参数，目前最大的中文基座预训练语言模型[12]具有超过2 000亿个参数. 对于大多数研究人员或小型公司/研究机构来说，这类大模型训练和部署所需的时间和资源成本是难以承受的.

2)大多数基于自回归的预训练语言模型只能利用单向信息. 许多自然语言理解(natural language understanding, NLU)任务需要上下文信息进行决策，仅能利用单向信息的基于自回归的预训练语言模型通常在这类任务上表现欠佳[11]. 我们认为，充分利用双向信息可以使具有强大文本生成能力的自回归预训练语言模型，尤其是小型预训练语言模型，在自然语言理解任务中表现出更加优异的性能.

3)现有的大规模自回归预训练语言模型，如CPM-2[14]和PanGu-α[12]，都依赖基于词语的字典来构建输入. 然而，这种策略具有以下缺点：一个问题是借助分词工具或者分词算法进行分词可能会导致分词错误[15]. 例如，对于句子“胜利海上油田产油创新高”来说，正确的分词结果为“胜利|海上|油田|产|油|创|新高”. 然而，CPM-2使用的分词工具将该句子分为“胜利|海上|油田|产|油|创新|高”，这与原始的句意并不一致. 另一个问题是，中文的词语量通常比汉字的量大得多. 对于一份仅需要几千个汉字就能覆盖99.9%数据的语料来说，要达到相同的覆盖率，则需要超过100 000个词语(见2.2节). 字典规模的增大会导致模型参数的增加，从而带来更大的训练开销. 而且，基于词语的策略容易造成数据稀疏和未知词汇(out of vocabulary, OOV)的问题，这种情况下，模型难以充分地学习到不常用字的知识. 因此，我们认为基于汉字的数据处理策略更适合中文预训练语言模型，因为汉字更符合中文的语言特性并能减少模型参数的数量.

为了解决以上挑战，我们提出并训练了一个高质量的、基于自回归的小型中文预训练语言模型——玲珑. 玲珑的名字寓意其规模虽小但是具有强大的能力. 玲珑在50 GB的高质量训练数据上充分训练，并使用基于汉字的字典构造输入. 凭借仅仅3.17亿个参数，玲珑在文本生成、问题解答和数学计算等多种下游任务中取得了出色的结果.

总体来讲，本文有4点贡献：

1)提出并训练了一个仅有3.17亿个参数的高质量中文预训练语言模型——玲珑.

2)基于通用规范汉字表构建了一个基于汉字的字典，有效地避免和减少了未知词汇和分词错误带来的负面影响.

3)训练了玲珑及其反向版本玲珑$_{\mathrm{B}}$，通过下游任务验证了基于自回归的模型可以通过利用双向信息取得更好的性能.

4)制定的模板可以将多种下游任务转换为生成任务. 实验结果表明，在多种任务上，玲珑可以达到与大型预训练语言模型相媲美甚至更好的性能.

1 相关工作

1.1 预训练语言模型结构

语言模型(language model, LM)就是计算序列的概率，因此可以根据已有的一部分语句来预测句子的下一个元素是什么. 其标准定义为：对于语言序列$w_1,w_2,\cdots,w_n$，计算该序列的概率，即$P(w_1,w_2,\cdots,w_n)=P(w_1)P(w_2|w_1)\cdots P(w_n|w_1,w_2,\cdots,w_{n-1})$.

语言模型发展至今经历了3个阶段，分别为80年代的专家语法规则模型、2000年左右的统计语言

模型,以及目前最常使用神经网络预训练语言模型.预训练语言模型通过自监督学习的方式,能够充分利用大规模、无标注的数据学习通用语言特征,具备一定常识和认知泛化能力.

预训练模型结构主要分为3类:基于编码-解码器的模型[5,16-17]、自编码模型[3]和自回归模型[1-2,6].

基于编码-解码器的模型通常在编码器部分采用双向自注意力机制,在解码器部分采用单向自注意力机制,编码器部分学习到的信息会传递给解码器.这类模型适用于解决文本生成、问题回答等条件生成任务.

自编码模型使用双向自注意力机制,通常对原始文本进行遮蔽(mask)处理.这类模型通过理解上下文信息对遮蔽部分进行补充,适用于完形填空等自然语言理解任务.

自回归模型使用单向自注意力机制,按照文本顺序依次进行学习,通常适用于文本生成任务.然而,由于其使用单向自注意力机制,通常无法在自然语言理解相关任务上取得较好的效果.

1.2 中文预训练语言模型

目前已有较多中文预训练模型被提出.一些较为小型的模型有MacBERT[18],NEZHA[19]等.MacBERT[18]在BERT[3]基础上进行了模型结构调整,并利用词汇量达54亿的中文数据进行训练.MacBERT可以在阅读理解、句子分类等自然语言理解任务上取得较好性能.NEZHA使用10台,每台具有8个32 GB内存NVIDIA Tesla V100 GPU的华为云服务器来进行模型训练和微调,在阅读理解、命名实体识别和情绪分类等自然语言理解任务上具有一定竞争力.

大模型有CPM[11]、GLM[13]、PLUG、PanGu-α[12]、CPM-2[14]、Claude-2、通义千问[20]等.CPM[10]从100 GB大规模中文语料库中学习通用语言模式,耗费3周时间使用64块V100 GPU进行模型训练,得到了首个基于解码器结构(自回归)的大规模中文预训练语言模型.实验表明,CPM模型能够用于问题解答、文章摘要、对话以及各类型的生成任务.GLM[13]基于自回归模型,通过结合多种预训练目标,可以更好地兼顾自然语言生成和自然语言理解任务.其公开发布的中文预训练模型参数量高达100亿.2021年4月19日,阿里达摩院发布了具有270亿参数量的中文预训练语言模型PLUG,其训练使用超过1 TB高质量中文文本数据.PLUG以80.614的分数刷新了中文语言理解评测基准CLUE[21]分类榜单纪录.PanGu-α[12]模型参数量高达2 000亿,不仅占据CLUE总榜单榜首,在落地应用上也具有卓越优势.Claude-2模型由Anthropic研发并发布于2023年7月,其参数量高达1 300亿.Claude-2支持高达10万个标记(token)上下文并且训练数据更新到2023年,相对于其他模型,它能够利用更长、更准确的信息.此外,Claude-2在代码和数学方面的能力有显著提升,可以更准确地解析和理解复杂代码和数学表达式.2023年8月,阿里云发布具有70亿参数量的通义千问模型,模型在超过2.2万亿个标记上进行预训练,上下文长度为2 048.在人文、社会科学和其他专业领域的52个主题上进行测试,通义千问在现有相似规模的模型中表现最佳,甚至超过了更大规模的模型.

目前已有的小规模中文预训练模型通常基于自编码结构[22-23],适用于理解任务,在生成任务上表现不佳.而大模型由于其参数量巨大,导致训练成本十分高昂.大模型的使用权和所有权通常掌握在对应研发机构手中,个人研究者或中小型研究机构即使了解模型结构和原理也难以承担训练和使用方面的成本.

2 玲 珑

本文提出了一个基于自回归目标的小型中文预训练语言模型——玲珑.玲珑基于汉字字典使用约50 GB来源广泛的高质量训练数据进行模型预训练.此外,通过将各种类型的自然语言下游任务转变为自然语言生成任务,可以很好地利用玲珑解决问题.结合玲珑与玲珑$_{\mathrm{B}}$可以对双向文本信息加以利用,从而进一步提升模型在下游任务上的性能.

2.1 训练数据

规模适当的高质量中文语料对中文预训练语言模型及其他中文自然语言处理模型的效果起到至关重要的作用.研究表明,使用小型高质量语料库训练的模型比仅使用更大规模的中文维基百科训练的模型表现更好[24].

我们从公开渠道收集了总计近0.72 TB的原始数据,覆盖百科、新闻、教育和网络数据等多种类别.为了构建一个高质量的中文数据集,我们对原始收集得到的数据进行了中文数据提取、基于规则的清理、去重和过滤处理[25],得到约50 GB,包含约1 500万个标记的数据,各类别数据训练时使用的数据量如表1所示.

2.2 基于汉字的预处理

我们主要基于《通用规范汉字表》[26]和清洗后的

Table 1 Statistics of Training Data

表 1 训练数据统计信息 GB

类别	数据量
新闻	36.26
Common Crawl 网络数据	7.93
百科	5.50
网络	2.35
专利	2.68
教育	1.77
小说	0.62

高质量语料构建了基于汉字的字典. 具体来说, 我们首先将《通用规范汉字表》中全部一级和二级汉字加入字典.《通用规范汉字表》是中国政府公布的现行标准汉字表, 一、二级字表合计 6 500 字, 能够满足出版印刷、辞书编纂和信息处理等方面的一般用字需要. 然后, 我们从训练语料中统计经常出现的汉字和其他非中文标记来补充字典. 虽然玲珑是一个中文预训练语言模型, 但我们仍然保留了一些在中文文章或对话中经常出现的非中文标记. 最终构建得到的字典包含 13 312 个标记, 只有其他基于中文词语的字典规模的 1/4~1/2[11-12,27].

对于清洗后得到的高质量数据集中的每篇文章, 我们在其开头和结尾分别添加特殊标记 [START] 和 [END]. 然后, 使用基于汉字的字典将文章中的标记转换为整数 ID. 最后我们将不同文章进行拼接串联, 并使用上下文长度大小的滑动窗口对其进行切分, 这样可以使得到的每条训练数据的长度都保持一致, 模型在训练过程中可以充分利用数据和算力.

2.3 模型结构设计

玲珑是一个基于自回归结构的预训练语言模型, 由 1 个嵌入层、多个解码器模块和 1 个输出层组成. 玲珑对文本生成过程进行建模, 当前位置标记的生成概率取决于序列中前面的标记. 模型整体结构如图 1 所示.

嵌入层用于将高维特征映射到低维, 并且同时考虑标记的含义(文本语义)和位置(文本位置)信息. 因此嵌入层需要同时学习 2 个嵌入矩阵, 分别用于计算标记嵌入和位置嵌入, 即

$$\boldsymbol{h}^{(0)} = t\boldsymbol{W}_t + p\boldsymbol{W}_p, \tag{1}$$

其中 $\boldsymbol{h}^{(0)}$ 表示数据进行嵌入后得到的低维表示, t 表示标记 ID, p 表示位置索引, $\boldsymbol{W}_t$ 和 $\boldsymbol{W}_p$ 分别为标记嵌入矩阵和位置嵌入矩阵.

每个解码器模块包含 1 个单向稀疏多头自注意

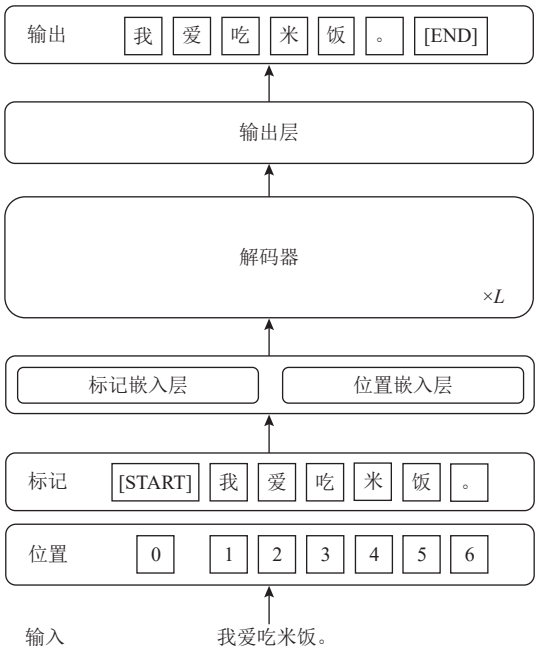


Fig. 1 The overall structure of LingLong

图 1 玲珑模型整体结构

层和 1 个前馈层, 公式表示为

$$\begin{cases} \boldsymbol{a}^{(l-1)} = SparseMultiHeadSelfAttention\left(\boldsymbol{h}^{(l-1)}\right), \\ \boldsymbol{f}^{(l-1)} = FeedForward\left(\boldsymbol{a}^{(l-1)} + \boldsymbol{h}^{(l-1)}\right), \\ \boldsymbol{h}^{(l)} = \boldsymbol{f}^{(l-1)} + \boldsymbol{h}^{(l-1)}, \end{cases} \tag{2}$$

$\boldsymbol{h}^{(l)}$ 是第 l 个解码器模块的输出. 注意层可以学习相同标记在不同语境中可能具有不同的语义, 而稀疏机制则有助于减轻计算开销.

输出层用于预测下一个标记, 计算公式为

$$\boldsymbol{o} = \boldsymbol{h}^{n}\boldsymbol{W}_t^{\mathrm{T}}, \tag{3}$$

$\boldsymbol{o}$ 表示字典中每一个标记出现在下一个位置的概率.

模型训练过程中, 我们使用交叉熵损失函数并持续更新模型参数. 表 2 列出了模型训练使用的重要超参数设置.

Table 2 Hyperparameters of LingLong

表 2 玲珑中使用的超参数

超参数	取值
字典规模	13 312
嵌入维度	1 024
隐藏层维度	1 024
解码模块数量	24
自注意头数量	16
稀疏自注意步长	128
稀疏自注意表现力	8
上下文窗口长度	1 024
可训练参数数量	316 989 440

2.4 双向预训练

我们在模型训练阶段使用相同数据集训练了2个结构完全相同的模型——前向模型(玲珑)和反向模型(玲珑$_{\mathrm{B}}$). 玲珑以自然文本顺序获取训练标记, 玲珑$_{\mathrm{B}}$则以反向文本顺序获取训练标记, 2个模型使用2种不同的输入顺序来利用双向信息, 同时保持自回归的模型训练目标. 例如, 玲珑使用的训练标记是"[START]今天天气不好[END]", 玲珑$_{\mathrm{B}}$相应的训练标记是"[START]好不气天天今[END]". 简单的方法往往是十分有效的, 通过利用双向信息, 许多下游任务的性能都得到了有效提升, 3.2.2节中进行了相关验证.

我们使用Adam优化器进行参数更新过程, 选用的超参数设置为$\beta_1=0.9$, $\beta_2=0.95$, $eps=1\times10^{-8}$. 为了保持训练稳定, 我们在不同阶段使用了不同学习率. 具体来讲, 在预训练早期阶段, 我们使用线性热身(warm-up)策略, 在6×10^8个训练标记中将学习率从0逐步提高到最大值2.5×10^{-4}. 在学习率达到峰值后, 使用余弦衰减策略, 将其缓慢降至一个较小的值, 余弦衰减策略持续1×10^{10}个训练标记. 此外, 我们还会定期以比当前学习率稍高的学习率重新启动训练过程, 以帮助模型摆脱局部极小值. 在实际训练时, 我们使用了数据并行方案, 共计使用20个NVIDIA Tesla V100S GPU完成预训练, 全局批次大小(batch size)为80, 训练过程中在单个GPU上最多需要32 GB GPU内存.

2.5 模型微调

作为自回归模型, 预训练得到的玲珑主要适用于文本生成任务. 因此, 在完成下游任务时我们使用将所有下游任务都转换为文本生成任务的策略, 使微调和预训练目标更加接近, 以更好地利用预训练模型的生成能力. 例如, 对于1对给定的(x,y), 其中x为原始问题, y为标签, 我们可以通过一个转换模板将其转换为生成任务$\hat{y}=G(x)$. 最简单和直观的方式就是采用自然语言编写的能够表达任务语义的模板, 例如, 对于玲珑来说, 一个分词任务可以构造为"原始文本: 回首来时的路, 坚定的信念载着我们走了很远. [SEP] 分词结果: 回首 [SEP2] 来 [SEP2] 时 [SEP2] 的 [SEP2] 路 [SEP2], [SEP2] 坚定 [SEP2] 的 [SEP2] 信念 [SEP2] 载 [SEP2] 着 [SEP2] 我们 [SEP2] 走 [SEP2] 了 [SEP2] 很 [SEP2] 远 [SEP2]. "的格式. 而对于玲珑$_{\mathrm{B}}$来说, 由于模型在训练时采用反向的文本表述顺序, 因此在模型微调时也需对数据进行反向调整. 为了保持认知上的"由因推果", 我们仍然采用问题在前、答案在后的构造顺序, 对于相同的分词任务, 对于玲珑$_{\mathrm{B}}$构造的输入为"原始文本: . 远很了走们我着载念信的定坚, 路的时来首回 [SEP] 分词结果: . [SEP2] 远 [SEP2] 很 [SEP2] 了 [SEP2] 走 [SEP2] 们 我 [SEP2] 着 [SEP2] 载 [SEP2] 念 信 [SEP2] 的 [SEP2] 定 坚 [SEP2], [SEP2] 路 [SEP2] 的 [SEP2] 时 [SEP2] 来 [SEP2] 首回". 表3中展示了我们为多个下游任务设计的正向模板和反向模板.

微调时使用与预训练阶段一致的交叉熵损失函数, 并同样采用热身和衰减的学习率调整计划.

3 实 验

本节首先进行消融实验来验证玲珑采用的策略的有效性, 然后对玲珑在一系列下游任务中的表现进行评估来验证其有效性. 玲珑的代码实现和预训练权重全部开源[28], 均可在GNU GPLv3许可下使用.

Table 3 Templates for Downstream Tasks

表3 下游任务模板

任务	数据集	正向模板	反向模板
文本摘要	CEPSUM	类别: ["*家居用品*", "*箱包*", "*服装*"]; 特征信息: *格式化信息*; 商品描述: *商品描述* [SEP] 商品简介: *商品简介*	类别: ["*装服*", "*包箱*", "*品用居家*"]; 特征信息: *息信化式格*; 商品描述: *述描品商* [SEP] 商品简介: *介简品商*
	LCSTS	文本: *原始文本* [SEP] 摘要: *摘要*	文本: *本文始原* [SEP] 摘要: *要摘*
基于结构化数据的文本生成	AdGen	标题信息: *标题*; 标签信息: *标签*; 特征信息: *商品特征*; [SEP] 商品描述: *商品描述*	标题信息: *题标*; 标签信息: *签标*; 特征信息: *征特品商*; [SEP] 商品描述: *述描品商*
	E-Reviews	特征信息: *特征* [SEP] 广告文案: *文案*	特征信息: *征特* [SEP] 广告文案: *案文*
问答	KBQA	问题: *问题* [SEP] 答案: *实体* [SEP2] *关系*	问题: *题问* [SEP] 答案: *系关* [SEP2] *体实*
中文分词	Weibo & MSR	原始文本: *原始文本* [SEP] 分词结果: *分词结果*	原始文本: *本文始原* [SEP] 分词结果: *果结词分*
句子对分类	LCQMC	"*句子1*"与"*句子2*"的意思是否相似? [SEP]["*是*", "*否*"]	? 似相否是思意的"*1 子句*"与"*2 子句*"[SEP]["*是*", "*否*"]
数学推理	Math23K	问题: *问题* [SEP] 答案: *计算公式*	问题: *题问* [SEP] 答案: *式公算计*
阅读理解	CMRC	文本: *文本*; 问题: *问题* [SEP] 答案: *答案*	文本: *本文*; 问题: *题问* [SEP] 答案: *案答*

注: 斜体文字表示来自数据集中的数据, 正体文字是提示符或特殊分隔符.

3.1 实验设置

为了评估和对比玲珑以及其他以中文为主的预训练语言模型的性能，我们从 CUGE[29] 中精心选择了 7 个不同的自然语言处理下游任务，选定任务可用于全面地评估模型的自然语言生成和自然语言理解能力. 研究中具体使用的评估基准和指标概述如下：

1）文本摘要是一项对给定的一段长文本进行摘要生成的自然语言生成任务. 我们在 CEPSUM 2.0[30] 和 LCSTS[31] 数据集上评估语言模型生成摘要的能力. CEPSUM 2.0 包含家居用品（home applications）、服装（clothing）和箱包（cases and bags）相关的产品描述，分别使用 3 个类别的数据对模型进行微调然后进行评分，使用 3 个类别的平均得分作为 CEPSUM 2.0 的评估结果. 评价指标采用 RouGe-L[32]，它根据 2 个序列的最长公共子序列长度来衡量 2 个序列的相似度.

2）基于结构化数据的文本生成是一项基于结构化数据生成文本的自然语言生成任务. 我们使用 AdGen[33] 数据集来评估模型，使用 BLEU-4[34] 作为评价指标. 该数据集中的每个实例都包含输入的产品信息（表格形式的格式化信息）和预期广告文本（字符串）.

3）问答（question answering, QA）任务需要模型回答用自然语言描述的问题. 我们使用 NLPCC2018-KBQA[35] 数据集来评估玲珑和其他基线模型的自然语言理解能力. NLPCC2018-KBQA 包含一份知识图谱数据和一些针对知识图谱内容的提问及答案. 我们将 NLPCC2018-KBQA 中的问题转化为关系提取问题，具体来讲就是模型接收自然语言描述的问题，期望模型输出提取的实体和关系，然后利用提取的实体和关系在知识图谱中匹配答案. 我们使用准确率，也就是正确回答问题比例作为评价模型性能的指标.

4）中文分词是一项将句子分解为词序列的自然语言理解任务. 中文分词的一个重要挑战在于没有标准词库，也很难定义什么是词. 在不同语境或者不同需求下，分词的标准答案都是不同的. 我们使用微博数据集[36] 和 MSR 数据集[37] 作为评测基准，$F1 = 2\times\frac{precision\times recall}{precision+recall}$ 作为评价指标，其中，精确度（*precision*）是指正确分割单词数与模型预测单词数之比，召回率（*recall*）是指正确分割单词数与标签中单词数之比.

5）句子对分类是一项经典的自然语言理解任务，涉及区分 2 个句子之间的关系，如相似性或包含关系. 我们使用问题匹配数据集 LCQMC[38] 作为基准，使用准确率作为评价指标.

6）数学推理是指利用算法/模型解决用自然语言描述的数学问题的任务. 经过不断尝试，我们发现玲珑的优势在于理解问题而非数值计算. 因此，我们使用模型从原始问题中提取数学公式，然后使用 *eval* 函数（Python 中的函数）计算最终的数值结果. 评测基准使用 Math23K[39] 数据集，评价指标使用准确率.

7）中文拼写检查（Chinese spell-checking, CSC）是一项关于纠正中文句子中拼写错误的任务. 使用 SIGHAN13 数据集[40] 作为评价基准，*F*1作为评估指标，其中精确度是指正确找到的拼写错误数量与模型预测的错误数量之比，召回率表示正确找到的错误数量与实际错误数量之比.

8）阅读理解（reading comprehension）任务是一项自然语言理解任务，旨在让模型理解文本和问题并回答关于给定文本的问题. 在阅读理解任务中，通常会提供一段文本和相关问题，要求模型根据文本内容生成合适的答案. 这个过程涉及到多个步骤，包括语义理解、推理以及答案生成等，需要模型具备较高的语言处理和推理能力. 本文采用 EM（exact match）指标进行性能评估，将预测结果与真实结果进行比较，正确回答问题比例作为评价模型性能的指标. EM 指标值越高，说明模型性能越好.

我们为所有任务构建的模板展示在表 3 中，其中来自数据集中的数据使用斜体展示，其他为提示符或特殊分隔符. 例如，在 LCSTS 的模板中，“*原始文本*”应替换为相应数据的原文，“*摘要*”应替换为相应数据的摘要，“[SEP]”为分隔符，其他文本均为构造的提示文本.

3.2 消融实验

3.2.1 字比词更有优势

为了验证基于汉字进行数据处理策略的优越性，我们使用不同的字典在相同数据集上训练了基于汉字策略和基于词语策略的 2 个规模相近的模型. 字典分别采用本文构建的基于汉字的字典和 CPM-2[14] 中使用的基于词语的字典，CPM-2 字典使用 BPE（byte pair encoding）方式构造得到，2 个字典互不为子集. 此外，出于快速验证的目的，我们将 2 个模型的解码器模块数减少到 12 个，隐藏层维度减少到 768，每个模型大约有 1 亿个参数（基于词语的模型有 1.06 亿个参数，基于汉字的模型有 0.96 亿个参数）.

使用不同标记化策略处理后的预训练数据集信息如表 4 所示. 特殊标记 [UNK] 表示字典中没有的标记. 在使用基于词语的标记策略时，[UNK] 比例达到了 0.417 7%，是基于汉字策略的近 3 倍，这说明了

Table 4 Summary of Our Pre-training Dataset Using Different Tokenization Strategies

表 4 使用不同标记化策略进行预训练数据集处理结果

数据量	基于汉字标记策略	基于词语标记策略
字典规模	13 312	26 240
标记数量	23 710 716 503	19 177 964 849
[UNK] 数量	34 566 039	80 110 810
[UNK] 比例/%	0.145 8	0.417 7

基于词语的策略更容易导致 OOV 问题.

我们还验证了这 2 个模型在 6 个下游任务上的性能, 结果展示在表 5 中. 整体来看, 基于汉字的模型能更好地完成各种下游任务. 这是因为基于词语的模型经常无法分割或者错误地分割原始文本. 此外, 由于大多数中文语素(语言表达中最小的有意义成分)是单个中文汉字, 因此, 使用基于汉字的字典更符合中国人的表达习惯. 基于汉字的字典还可以有效减少 [UNK] 出现, 从而增加数据中有效标记数量.

3.2.2 双向比单向更有优势

我们比较了单向模型(玲珑、玲珑$_{B}$)和双向模型(玲珑$_{F+B}$, 玲珑和玲珑$_{B}$ 的组合)在 7 个下游任务上的性能. 为了保证模型性能, 本文针对每个任务设计相应策略结合玲珑和玲珑$_{B}$ 的输出. 例如, 单向模型完成中文拼写检查任务时, 我们使用模型计算句子中每个位置上字典中的标记出现的概率, 并使用每个位置的 *top-k* 个标记作为该位置的候选字. 如果实际句子中的标记不在候选集合中, 那么我们认为该位置的标记是错误的. 当使用双向模型时, 我们使用玲珑和玲珑$_{B}$ 的 *top-k* 标记共同构建候选集. 表 6 展示了我们为下游任务设计的双向信息使用方式. 虽然为不同任务设计不同双向信息的使用方式会带来额外开销, 并损失一定泛化性. 但对于小规模模型来说, 以较少的资源就可以实现并行, 推理时间不会显著增加, 精巧的模板构建和结果融合设计可以显著地提升模型在下游任务上的性能, 因此这部分的开销是必要和值得的.

表 7 展示了双向模型与单向模型在下游任务上的实验结果. 双向模型相对于玲珑和玲珑$_{B}$ 的平均改进率为 6.97% 和 20.45%. 尤其是在中文拼写检查(SIGHAN)等自然语言推理任务上大大优于单向模型. 这是因为对于自然语言理解任务, 上下文信息对于做判断来说更为重要.

对于中文拼写检查任务, 我们没有设计模板, 而是直接使用模型计算当前位置每个标记出现的概率, 因此模型在预测第i个位置的标记时, 只能利用前$i-1$个位置的信息. 而由于句子可以以任何标记开头, 仅使用玲珑无法确定句子第 1 个标记是否正确, 需要双向信息才能更好地确定句子每一个位置标记的正确性.

总之, 通过结合使用玲珑和玲珑$_{B}$ 模型可以更有效地利用双向信息, 对于许多下游任务来说, 双向模型比单向模型更可靠. 虽然双向模型增加了一定开销, 但它仍然是用户友好的, 玲珑和玲珑$_{B}$ 可以以较少的计算资源独立运行或并行推理, 使用双向信息

Table 5 Performance of Models Trained Using Character-Based Strategy and Word-Based Strategy on Downstream Task Datasets

表 5 基于汉字策略和基于词语策略训练模型在下游任务数据集上的性能 %

策略	文本摘要		基于结构化数据的文本生成	问答	中文分词		句子对分类	数学推理
	CEPSUM 2.0	LCSTS	AdGen	KBQA	Weibo	MSR	LCQMC	Math23K
基于词语策略模型	19.21	23.75	6.56	56.90	52.78	60.19	80.63	6.10
基于汉字策略模型	23.73	30.85	9.28	73.00	93.97	95.37	83.00	54.10

注: 对于所有任务来说, 数值越高越好.

Table 6 Methods for Aggregating the Outputs of LingLong and LingLong$_{B}$

表 6 玲珑和玲珑$_{B}$ 输出结果的结合方式

任务	中文拼写检查	文本摘要	基于结构化数据的文本生成	问答	中文分词	句子对分类	数学推理	阅读理解
方案	结合 2 个模型输出结果, 将所有查找到的错字进行输出.	2 个模型分别输出完整结果, 将结果与原始输入计算 Rouge-L 分数, 选取得分较高的 1 个.	2 个模型分别输出完整结果, 将结果与原始输入计算 Rouge-L 分数, 选取得分较高的 1 个.	分别使用 2 个模型得到的结果进行查询. 若均能查询到结果, 且结果不一致, 则取正向模型输出结果(查询方式见 3.1 节).	结合 2 个模型结果, 给出 1 个细粒度分词方案和 1 个粗粒度分词方案. (粗粒度: 仅当玲珑和玲珑$_{B}$ 均认为应当进行分词时才进行分词; 细粒度: 有任意 1 个模型认为应该分词时即进行分词).	选择 2 个模型输出中概率最高的类别.	从模型输出结果中选取语法正确(函数 *eval* 能够正确解析则为语法正确)的 1 个作为最终结果. 若均语法正确, 且结果不一致, 则取正向模型输出结果.	2 个模型分别输出完整结果, 将结果与原始输入计算 Rouge-L 分数, 选取得分较高的 1 个.

Table 7 Performance of Bidirectional Models Versus Unidirectional Models on Downstream Tasks

表 7 双向模型与单向模型在下游任务上的性能 %

模型	中文拼写检查	文本摘要		基于结构化数据的文本生成	问答	中文分词		句子对分类	数学推理	阅读理解
	SIGHAN	CEPSUM 2.0	LCSTS	AdGen	KBQA	Weibo	MSR	LCQMC	Math23K	CMRC
玲珑	49.80	25.23	41.07	19.24	84.80	95.41	96.03	84.46	68.60	70.09
玲珑$_{B}$	43.88	23.93	33.92	9.46	69.20	95.04	95.84	85.91	59.90	46.63
玲珑$_{F+B}$	63.10	26.04	45.98	20.57	84.90	95.20	96.14	91.60	75.60	78.20

增强模型表现的同时不显著增加推理时间. 实验结果也表明, 结合双向信息通常能取得明显优于单向模型的性能. 总体来说, 使用双向模型是一种灵活可选的方法, 可以根据特定需求进行定制. 在资源很有限的情况下, 我们更推荐单独使用正向玲珑模型.

3.3 文本生成

文本生成任务使用预训练语言模型根据前文信息生成后续文本, 因此可以直接使用玲珑的预训练版本. 值得注意的是, 我们没有对生成结果进行任何后期编辑. 与 CPM 和 PanGu-α一样, 只有当模型没有在合理的点停止生成时, 我们才会截断生成文本. 在进行文本生成任务时, 我们使用 *top-k* = 10, *top-p* = 0.9, 以及 *temperature* = 1.0作为超参数设置, 以合理地控制模型生成文本的正确性和多样性. 表 8 中展示了一些生成示例, 通过这些示例, 可以对玲珑的能力有一个直观的了解和认识. 在检查数千个生成结果后, 我们发现玲珑在人物传记生成、对联生成和故事生成等任务中表现十分出色.

Table 8 Examples of Text Generated by LingLong

表 8 玲珑文本生成示例

类别	提示文本	生成文本
人物介绍	齐小明，知名学者	文化产业研究者. 2013 年，出版《文化产业的新趋势》《中国文化产业与资本市场》.
对联	上联：一帆风顺年年好； 下联：	风平浪静日日新
故事	有一天，一只小蝴蝶正在森林里飞，	突然，它看见一只老鹰正从空中俯瞰着她.

3.4 下游任务

3.4.1 实验设置

本节对比了玲珑与其他 7 个大小相近的中文预训练语言模型以及 5 个大型中文预训练语言模型在 7 个下游任务上的性能. 在微调和评估阶段, 我们使用模板将下游任务转换为生成任务, 模板构造方式见表 3. 为了保证与其他模型进行公平的比较, 我们尽量复用了模型公开测评时报告的分数. 在缺少结果的数据集上对已经公开预训练权重的模型进行测评, 对于规模相近的模型测评时使用与微调玲珑时一样的策略和模板, 对于大模型采用不微调直接测评的方式.

3.4.2 实验结果

如表 9 所示, 总体来说玲珑在各个任务和数据集上均表现出了强大的性能. 和规模相近的预训练模型相比, 比如 ERNIE 3.0 XBase 和 mT5-small, 玲珑的性能更具优势. 虽然这些规模相似的预训练语言模型同样针对下游任务进行了微调, 然而它们只能在少数任务上获得与玲珑相似的性能, 而在其他任务上表现不佳. 例如, CPT 系列模型(CPT$_{g}$/CPT$_{u}$)在文本摘要任务中表现不错, 但在数学推理任务上表现非常差. 此外, 玲珑在经过微调之后可以获得与其他大型预训练语言模型可比的结果. 例如, mT5 XXL 和 CPM-2 分别具有 130 亿个和 110 亿个参数, 但在所有任务上, 微调后的玲珑都具有更好的性能. 这也说明了这些在大规模语料库上训练参数超过 100 亿的模型的零样本能力仍然有所欠缺, 不能直接应用于下游任务, 需要进行微调. 玲珑仅有 3.17 亿个参数, 微调玲珑更容易也更节约资源, 使得玲珑相比大模型更适合现实世界应用.

接下来, 我们将详细讨论玲珑在每个任务上的性能. 在文本摘要任务中, 玲珑具有正确生成简洁摘要的能力. 然而, 由于数据本身的限制, 玲珑的能力不能得到完全地发挥和评测. 例如一些标注摘要是输入文本的标题, 导致标注摘要不能完全覆盖文本表达的信息, 或者标注摘要包含输入文本中不存在的额外背景信息. 在任务更加明确的情况下(即微调数据更贴合任务目标, 不包含干扰信息), 玲珑的能力有希望进一步得到提升. 此外, 由于 ERNIE 3.0 具有大规模模型参数和为特定任务设计的表示模块, 因此在文本摘要任务上比玲珑具有优势. 这提示我们设计任务特定的模块有助于提升模型在特定任务上的表现.

在基于结构化数据的文本生成任务中, 玲珑的性能明显优于其他规模相似的模型. 由于使用了基

Table 9 Performance of Different Language Models on Six Downstream Tasks

表 9 不同语言模型在 6 个下游任务的性能 %

模型	规模	文本摘要		基于结构化数据的文本生成	问答	中文分词		句子对分类	数学推理	阅读理解
		CEPSUM 2.0	LCSTS	AdGen	KBQA	Weibo	MSR	LCQMC	Math23K	CMRC2018
ZEN 2.0[41]	233M*	-	-	-	-	-	98.35	88.81	-	70.77
ERNIE 3.0 XBase[42]	280M	2.35†	18.86†	7.80†	0†	50.55†	70.37†	89.06†	0.70†	75.99
mT5-small[43]	300M	9.02†	33.10	10.20	0†	49.18†	49.32†	82.10	18.40	0.90
CPT_g[44]	393M	**26.26**†	42.80	10.70	84.38†	-	-	90.68†	36.90†	-
CPT_u[44]	393M	-	-	-	-	42.00†	98.51†	91.29†	-	68.80
BART[5]	406M	22.16†	40.90	12.68†	84.68†	46.14†	44.42†	90.93†	49.10†	61.32
GLM	335M	17.27†	34.25†	1.30†	**99.10**†	92.70†	89.18†	84.61†	51.70†	70.74
LEBERT[45]	7.5B*	-	-	-	-	-	**98.69**	-	-	-
ERNIE 3.0[42]	10B	-	**48.46**	**30.16**	-	-	-	90.38	75.00	75.30
CPM-2[14]	11B	0.91‡	35.90	10.60	0‡	34.72‡	33.22‡	89.16	69.37	15.66
mT5-XXL[43]	13B	0.17‡	34.80	9.80	0‡	42.07‡	43.51‡	88.30	61.60	25.20
Yuan 1.0[27]	245B	-	-	-	-	-	-	-	**76.90**	5.58‡
玲珑$_{F+B}$	317M	26.04	45.98	20.57	84.90	**95.20**	96.14	**91.60**	75.60	**78.20**

注：在“规模”列中，“50M”表示该模型有 5 亿（50 million）个参数，“10B”表示该模型有 100 亿（10 billion）个参数. “*”表示该模型规模由估算得到；“†”表示该结果由本文使用与玲珑相同策略进行微调和评估得到；“‡”表示该结果在零样本模式（未微调）下评估得到；“-”表示由于模型无法在对应任务/基准上评估或者由于模型权重不公开，因此无法得到结果. 黑体数值表示最优结果.

于汉字的输入策略，玲珑能够从较短的结构化输入中提取有效信息来生成连贯、有趣的长篇广告文本.

在问答任务中，玲珑准确率达到 84.90%，且所有模型使用我们设计的模板进行微调后均能取得 84% 以上的分数. 这充分说明模型成功地从自然语言问题中提取到了解决问题的关键信息，也说明了我们为 NLPCC2018-KBQA 数据集设计的关系提取策略和提示模板的有效性. GLM 模型在规模相近的情况下可以取得 99.10% 的分数，是所有模型中效果最好的. 经过分析，我们认为这得益于 GLM 的训练目标与自回归语言模型不同，GLM 将输入文本中的 1 个或多个词用 1 个特殊标记进行替换，然后训练模型预测被替换掉的词. 这种训练目标使得 GLM 具有更强的预测实体或者关系（实体与关系均为完整词语）的能力.

在中文分词任务中，玲珑取得了与其他预训练语言模型可以相比的结果，表明了通过构造合适的模板，小型预训练语言模型可以很好地执行分词任务. 然而，玲珑的性能略低于小规模的 ZEN 2.0 和 CPT_u，尤其是在 MSR 数据集上. 经过分析，我们认为 ZEN 2.0 和 CPT_u 执行任务时使用的输入模板是由研究人员通过实验或经验选择的更适合其参数和模型结构的. 因为在使用不同模板时，这些模型的性能波动很大，正如 CPT_u 在微博数据集上的得分远远低于在 MSR 数据集上的得分. 此外，大模型 LEBERT 在 MSR 数据集上取得了较好的性能. 然而相对于 LEBERT，玲珑仍具有推理速度快、开销低的优势，更具成本收益.

在句子对分类任务中，由于 LCQMC 数据集中每个句子对中的句子通常在结构和措辞上非常相似. 例如，在句子对“古诗咏柳中的咏字是什么意思？”和“古诗咏柳是什么意思？”中，只有 3 个字不同，但是这 2 个句子的含义是完全不同的. 语言模型必须深入理解句子的含义才能区分它们. 整体来看，玲珑的性能略高于所有其他模型，无论是小规模还是大规模的预训练语言模型都证明玲珑能够很好地理解文本意图.

数学推理任务中，玲珑准确率达到了 75.60%，接近 ERNIE 3.0 和 Yuan 1.0 的性能，表明了即使是小规模模型也具有一定的解决数学问题的能力. 此外，mT5-small 和 mT5-XXL 的结果表明，随着模型尺寸增加，模型性能仍有一定改进空间. 即便如此，玲珑的准确率显著高于 mT5-small 模型，甚至优于相同结构且具有更大参数量的 mT5-XXL 模型.

在阅读理解任务上，结合了双向信息的玲珑相对其他测评方法来说具有较大优势. 结合表 7 中的结果来看，仅使用玲珑时可以取得与其他测评方法可比较的结果，通过结合玲珑与玲珑$_B$，模型整体性能有了显著提升，2 个模型起到了较好的互补作用.

玲珑仍然有很大改进空间，特别是在需要更具

体知识或更好微调策略的任务中. 尽管如此, 实验结果表明, 玲珑是一个很有前景的模型, 即使与规模大得多的模型相比, 它也可以在广泛的自然语言处理任务中获得有竞争力的性能.

4 总　　结

在本文介绍的工作中, 我们训练了一个基于自回归的中文预训练语言模型——玲珑, 该模型具有约 3.17 亿个参数. 玲珑利用经过完整清洗流程处理的高质量语料库进行训练, 训练数据采用了基于汉字的标记化策略. 在预训练阶段我们还引入一个新颖的反向训练流程, 得到了玲珑$_{B}$. 通过将玲珑与玲珑$_{B}$结合来完成下游任务, 使得自回归语言模型具有了处理双向信息的能力. 大量实验结果表明, 与相近规模的预训练语言模型相比, 玲珑适用于更广泛的下游任务且具有更加优秀的性能; 与更大的模型相比, 玲珑在自然语言处理下游任务中也可以获得相当的性能, 而玲珑以较少的参数量在使用时具有更小的资源需求和更少的推理时间. 总体来讲, 玲珑为后续研究奠定了良好的基础.

5 局限性分析

尽管玲珑在低资源环境中可以很好地处理各种下游任务, 但它仍有进一步优化的空间.

首先, 本文使用了手工制定的方法为每个任务构建模板. 如何自动构建模板或如何使用连续/自动提示(soft prompt)来帮助模型更加自适应地获得更好的结果仍然是值得研究的问题. 此外, 一些大型的英文基座预训练语言模型, 如 GPT-3, 在零样本模式下取得了优异效果, 而现有中文基座预训练语言模型, 无论其规模大小, 仍然需要微调才能较好地适用于下游任务. 因此, 积累足够的高质量训练数据或设计更好的模型结构, 使中文预训练语言模型能够在零样本模式下取得优异的结果也是至关重要的. 最后, 玲珑生成的文本在语法或道德上并不总是合适的, 生成的文本中可能包含冒犯性词语或不恰当的短语. 如何通过控制模型学习不当知识或者通过对模型生成的结果进行一定处理, 使生成的文本更容易被直接使用也是个迫在眉睫的课题.

作者贡献声明: 李东闻负责算法和实验方案设计、部分实验验证以及论文撰写; 钟震宇负责算法和实验方案设计、部分实验验证并修改论文; 孙羽菲提供方案及论文指导; 申峻宇完成部分实验验证以及数据集收集; 马子智完成部分实验; 于川越整理实验数据和文献; 张玉志提供整体方案及论文指导.

参　考　文　献

[1] Radford A, Narasimhan K, Salimans T, et al. Improving language understanding by generative pre-training[EB/OL]. 2018[2023-09-19]. https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf

[2] Radford A, Wu J, Child R, et al. Language models are unsupervised multitask learners[J/OL]. OpenAI Blog, 2019, 1(8): 9. [2023-09-20]. https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

[3] Devlin J, Chang M W, Lee K, et al. BERT: Pre-training of deep bidirectional transformers for language understanding[C]//Proc of the 2019 Conf of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg, PA: ACL, 2019: 4171−4186

[4] Yang Zhlin, Dai Zihang, Yang Yiming. XLNet: Generalized autoregressive pretraining for language understanding[C]//Proc of the Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2019: 5754−5764

[5] Lewis M, Liu Yinhan, Goyal N, et al. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension[J]. arXiv preprint, arXiv: 1910.13461, 2019

[6] Brown T, Mann B, Ryder N, et al. Language models are few-shot learners[C]//Proc of the Advances in Neural Information Processing Systems. Cambridge, MA: MIT, 2020: 1877−1901

[7] Shu Wentao, Li Ruixiao, Sun Tianxiang, et al. Large language models: Principles, implementation, and progress[J]. Journal of Computer Research and Development, 2024, 61(2): 351−361 (in Chinese)
(舒文韬, 李睿潇, 孙天祥, 等. 大型语言模型: 原理、实现与发展[J]. 计算机研究与发展, 2024, 61(2): 351−361)

[8] Diao Shizhe, Bai Jiaxin, Song Yan, et al. ZEN: Pre-training Chinese text encoder enhanced by n-gram representations[J]. arXiv preprint, arXiv: 1911.00720, 2019

[9] Liu Yihan, Ott M, Goyal N, et al. RoBERTa: A robustly optimized BERT pretraining approach[J]. arXiv preprint, arXiv: 1907.11692, 2019

[10] Sun Yu, Wang Shuohuan, Li Yukun, et al. ERNIE: Enhanced representation through knowledge integration[J]. arXiv preprint, arXiv: 1904.092233, 2019

[11] Zhang Zhengyan, Han Xu, Zhou Hao, et al. CPM: A large-scale generative Chinese pre-trained language model[J]. AI Open, 2021, 2: 93−99

[12] Zeng Wei, Ren Xiaozhe, Su Teng, et al. PanGu-α: Large-scale autoregressive pretrained Chinese language models with auto-parallel computation[J]. arXiv preprint, arXiv: 2104.12369, 2021

[13] Du Zhengxiao, Qian Yujie, Liu Xiao, et al. GLM: General language model pretraining with autoregressive blank infilling[C]//Proc of the 60th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2022: 320–335

[14] Zhang Zhengyan, Gu Yuxian, Han Xu, et al. CPM-2: Large-scale cost-effective pre-trained language models[J]. AI Open, 2021, 2: 216−224

[15] Li Xiaoya, Meng Yuxian, Sun Xiaofei, et al. Is word segmentation necessary for deep learning of Chinese representations[C]//Proc of the 57th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2019: 3242–3252

[16] Song Kaitao, Xu Tan, Tao Qin, et al. MASS: Masked sequence to sequence pre-training for language generation[C]//Proc of the 36th Int Conf on Machine Learning. New York: ACM, 2019: 9−15

[17] Bi Bin, Li Chenliang, Wu Chen, et al. PALM: Pre-training an autoencoding & autoregressive language model for context-conditioned generation[C]//Proc of the 2020 Conf on Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL, 2020: 8681−8691

[18] Cui Yiming, Che Wanxiang, Liu Ting, et al. Revisiting pre-trained models for Chinese natural language processing[C]//Findings of the Association for Computational Linguistics: EMNLP. Stroudsburg, PA: ACL, 2020: 657–668

[19] Wei Junqiu, Ren Xiaozhe, Li Xiaoguang, et al. NEZHA: Neural contextualized representation for Chinese language understanding[J]. arXiv preprint, arXiv: 1909.00204, 2019

[20] Bai Jinze, Bai Shuai, Chu Yunfei. Qwen technical report[J]. arXiv preprint, arXiv: 2309.16609, 2023

[21] Xu Liang, Hu Hai, Zhang Xuanwei, et al. CLUE: A Chinese language understanding evaluation benchmark[C]//Proc of the 28th Int Conf on Computational Linguistics. Stroudsburg, PA: ACL, 2020: 4762−4772

[22] Wang Wei, Bi Bin, Yan Ming, et al. StructBERT: Incorporating language structures into pre-training for deep language understanding [C/OL]//Proc of the 8th Int Conf on Learning Representations. Washington DC: ICLR, 2020[2024-09-19]. https://openreview.net/forum?id=BJgQ4lSFPH

[23] Lai Yuxuan, Liu Yijia, Feng Yansong, et al. Lattice-BERT: Leveraging multi-granularity representations in Chinese pre-trained language models[C]//Proc of the 2021 Conf of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. Stroudsburg, PA: ACL, 2021: 1716–1731

[24] Xu Liang, Zhang Xuanwei, Dong Qianqian. CLUECorpus 2020: A large-scale Chinese corpus for pre-training language model[J]. arXiv preprint, arXiv: 2003.01355, 2020

[25] Li Dongwen, Zhong Zhenyu, Shen Junyu, et al. NKCorpus: Extracting high quality large Chinese dataset from web data[J]. Frontiers of Data and Computing, 2022, 4(3): 30−45 (in Chinese)
(李东闻, 钟震宇, 申峻宇, 等. NKCorpus: 利用海量网络数据构建大型高质量中文数据集[J]. 数据与计算发展前沿, 2022, 4(3): 30−45)

[26] State Language Commission, Ministry of Education of the People's Republic of China. GF 0023-2020 List of Commonly Used Standard Chinese Characters[S]. Beijing: People's Publishing House, 2021 (in Chinese)
(中华人民共和国教育部, 国家语言文字工作委员会. GF 0023−2020 通用规范汉字表[S]. 北京: 人民出版社, 2021)

[27] Wu Shaohua, Zhao Xudong, Yu Tong, et al. Yuan 1.0: Large-scale pre-trained language model in zero-shot and few-shot learning[J]. arXiv preprint, arXiv: 2110.04725, 2021

[28] Intelligent Computing Laboratory, College of Software, Nankai University. LingLong-mcpt[CP/OL]. 2023[2023-09-19]. https://github.com/NKCSlCLab/linglong-mcpt (in Chinese)
(南开大学软件学院智能计算实验室. LingLong-mcpt[CP/OL]. 2023[2023-09-19]. https://github.com/NKCSICLab/linglong-mcpt)

[29] Yao Yuan, Dong Qingxiu, GuaN Jian, et al. CUGE: A Chinese language understanding and generation evaluation benchmark[J]. arXiv preprint, arXiv: 2112.13610, 2021

[30] Li Haoran, Yuan Peng, Xu Song, et al. Aspect-aware multimodal summarization for Chinese e-commerce products[C]//Proc of the AAAI Conf on Artificial Intelligence. Palo Alto, CA: AAAI, 2020: 8188−8195

[31] Hu Baotian, Chen Qingcai, Zhu Fangze. LCSTS: A large scale Chinese short text summarization dataset[C]//Proc of the 2015 Conf on Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL, 2015: 1967–1972

[32] Lin C Y. ROUGE: A package for automatic evaluation of summaries[C/OL]//Proc of the Text Summarization Branches Out. 2004: 74−81. [2024-09-19]. https://aclanthology.org/W04-1013.pdf

[33] Shao Zhihong, Huang Minlie, Wen Jiangtao, et al. Long and diverse text generation with planning-based hierarchical variational model[C]//Proc of the 2019 Conf on Empirical Methods in Natural Language Processing and the 9th Int Joint Conf on Natural Language Processing (EMNLP-JCNLP). Stroudsburg, PA: ACL, 2019: 3257–3268

[34] Papineni K, Roukos S, Ward T, et al. BLEU: A method for automatic evaluation of machine translation[C]//Proc of the 40th Annual Meeting of the Association for Computational Linguistics. Stroudsburg, PA: ACL, 2002: 311−318

[35] Li Juntao, Yan Rui. Overview of the NLPCC 2018 shared task: Multi-turn human-computer conversations[C]//Proc of the 7th Natural Language Processing and Chinese Computing. Berlin: Springer, 2018: 446−451

[36] Qiu Xipeng, Qian Peng, Shi Zhan. Overview of the NLPCC-ICCPOL 2016 shared task: Chinese word segmentation for micro-blog texts[C]//Proc of the Natural Language Understanding and Intelligent Applications: 5th CCF Conf on Natural Language Processing and Chinese Computing (NLPCC 2016), and 24th Int Conf on Computer Processing of Oriental Languages. Berlin: Springer, 2016: 901−906

[37] Emerson T. The second international Chinese word segmentation bakeoff[C/OL]//Proc of the 4th SIGHAN Workshop on Chinese Language Processing. 2005[2024-09-19]. https://aclanthology.org/I05-3017/

[38] Liu Xin, Chen Qingcai, Deng Chong, et al. LCQMC: A large-scale Chinese question matching corpus[C]//Proc of the 27th Int Conf on Computational Linguistics. Stroudsburg, PA: ACL, 2018: 1952−1962

[39] Wang Yan, Liu Xiaojiang, Shi Shuming. Deep neural solver for math

word problems[C]//Proc of the 2017 Conf on Empirical Methods in Natural Language Processing. Stroudsburg, PA: ACL, 2017: 845−854

[40] Wu Shih-Hung, Liu Chao-Lin, Lee Lung-Hao. Chinese spelling check evaluation at SIGHAN bake-off 2013[C/OL]//Proc of the 7th SIGHAN Workshop on Chinese Language Processing. 2013: 35−42. [2024-09-19]. https://aclanthology.org/W13-4406.pdf

[41] Song Yan, Zhang Tong, Wang Yonggang, et al. ZEN 2.0: Continue training and adaption for n-gram enhanced text encoders[J]. arXiv preprint, arXiv: 2105.01279, 2021

[42] Sun Yu, Wang Shuohuan, Feng Shikun, et al. ERNIE 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation[J]. arXiv preprint, arXiv: 2107.02137, 2021

[43] Xue L, Constant N, Roberts A, et al. mT5: A massively multilingual pre-trained text-to-text transformer[J]. arXiv preprint, arXiv: 2010.11934, 2020

[44] Shao Yunfan, Geng Zhichao, Liu Yitao, et al. CPT: A pre-trained unbalanced transformer for both Chinese language understanding and generation[J]. arXiv preprint, arXiv: 2109.05729, 2021

[45] Liu Wei, Fu Xiyan, Zhang Yue, et al. Lexicon enhanced Chinese sequence labeling using BERT adapter[C]//Proc of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th Int Joint Conf on Natural Language Processing (Volume 1: Long Papers). Stroudsburg, PA: ACL, 2021: 5847–5858



Li Dongwen, born in 1997. PhD candidate. Student member of CCF. Her main research interests include natural language processing and deep learning.

李东闻, 1997 年生. 博士研究生. CCF 学生会员. 主要研究方向为自然语言处理、深度学习.



Zhong Zhenyu, born in 1997. PhD candidate. His main research interests include natural language processing, high-performance computing, and artificial intelligence operations.

钟震宇, 1997 年生. 博士研究生. 主要研究方向为自然语言处理、高性能计算、智能运维.



Sun Yufei, born in 1976. PhD, professor, master supervisor. Her main research interests include deep learning, heterogeneous computing, and artificial intelligence.

孙羽菲, 1976 年生. 博士, 教授, 硕士生导师. 主要研究方向为深度学习、异构计算、人工智能.



Shen Junyu, born in 2001. Master candidate.His main research interest includes natural language processing.

申峻宇, 2001 年生. 硕士研究生. 主要研究方向为自然语言处理.



Ma Zizhi, born in 2000. Master candidate. His main research interest includes natural language processing.

马子智, 2000 年生. 硕士研究生. 主要研究方向为自然语言处理.



Yu Chuanyue, born in 2001. Master candidate. Her main research interest includes natural language processing.

于川越, 2001 年生. 硕士研究生. 主要研究方向为自然语言处理.



Zhang Yuzhi, born in 1964. PhD, professor, PhD supervisor. Member of CCF. His main research interests include deep learning and other artificial intelligence-related fields.

张玉志, 1964 年生. 博士, 教授, 博士生导师. CCF 会员. 主要研究方向为深度学习、其他人工智能相关领域.