CCL24-Eval任务2系统报告：基于多个大语言模型微调的中文意合图语义解析

李让
北京理工大学计算机科学与技术学院
lirang@bit.edu.cn

摘要

中文意合图对句中成分间的关系进行层次化标注，能有效表示汉语的深层语义结构。传统方法难以对中文意合图中的特殊成分进行特征表示，而近期大语言模型性能的快速提高为复杂自然语言处理任务提供了一种全新思路。在本次任务中，我们尝试使用Prompt-Response方式对大模型进行LoRA微调，让大模型根据输入直接生成格式化的中文意合图三元组序列。我们广泛测试来自不同研发团队、拥有不同参数规模的七个主流大模型，评估基座模型、参数规模、量化训练等因素对微调后模型性能的影响。实验表明，我们的方法展现出远超依存模型的性能，在测试集和盲测集上的F1分别为0.6956和0.7206，获得了本次评测榜一的成绩。

关键词： 中文意合图；语义解析；大模型微调；模型对比

System Report for CCL24-Eval Task 2: Chinese Parataxis Graph Parsing Based on Fine-Tuning Multiple Large Language Models

Rang Li
School of Computer Science and Technology, Beijing Institute of Technology
lirang@bit.edu.cn

Abstract

Chinese Parataxis Graph (CPG) annotates the relationships between sentence components hierarchically, which can effectively represent the deep semantic structure of Chinese language. Traditional methods struggle to extract features from the special components in CPG, while the rapid improvement in the performance of large language models has provided a novel approach for complex natural language processing tasks like this. In this task, we attempt to fine-tune large language models using LoRA method with Prompt-Response, allowing the models to directly generate formatted triple sequences of CPG based on our input. We extensively tested seven popular large language models from different teams with varying parameter sizes, evaluating the influence of base, size, and quantization on the performance of fine-tuned models. Experiments indicate that our method outperformed existing models by a significant margin, with F1 scores of 0.6956 and 0.7206 on the test set and blind test set respectively, achieving the top position in this evaluation.

Keywords: Chinese Parataxis Graph , Semantic Parsing , Large Language Model Fine-Tuning , Model Comparison

第二十三届中国计算语言学大会论文集，第43页-第50页，太原，中国，2024年7月25日至28日。
卷3：评测报告
(c) 2024 中国中文信息学会计算语言学专业委员会

1 引言

语义解析是自然语言处理（NLP）中的一个核心任务，旨在将自然语言转换成一种高度格式化的逻辑结构，可供进一步执行多种下游任务使用。这一过程涉及到识别词汇的语义角色、解析句中的实体关系以及理解语境中的隐含意义等。

由于语法结构的灵活性，中文的语义解析相比英文等语法规则相对严格的语言更加困难。以省略现象为例，中文表达经常涉及到句中某些成分的省略，其中大部分情况下是论元的省略，如谓词所对应的主体或客体。对于针对中文的语义解析方法而言，识别和处理此类特殊现象对于正确理解句子意义十分重要。

本次评测任务的中文意合图（Chinese Parataxis Graph）以事件为中心构建单根有向图，图中的节点对应承载事件、实体、属性的单元，有向边表示单元间的语义关系。中文意合图脱离了传统方法中句法形式的限制，将事件结构相对独立于句法结构进行表示，能较好地表示中文的特殊语义结构 (Guo et al., 2024)。由于中文意合图是一个较新的概念，领域内相关研究较少，本次评测中给出的依存模型也仅取得31.03%的F1分数。因此，我们决定从问题转化出发，分析使用传统方法和大模型微调的可行性。

2 问题转化

在本次任务中，我们需要以三元组的形式表示中文意合图中两个节点和它们之间的关系，这极大地依赖于句中每个词的上下文语义。因此，使用传统方法的一个直接的思路是先使用BERT (Devlin et al., 2018)获得句中每个词结合上下文语义的编码表示，接着对句中任意两个词的组合进行多分类，判断它们之间的关系。然而，考虑到中文意合图结构的特殊性，以下问题不能得到很好的处理。根据体系标签的定义，中文意合图的节点并不只包含句中的词，还可能包含表示句子结构的“ROOT”、表示逻辑关系的“因果关系”以及表示省略的“ls”等特殊成分 (Guo et al., 2024)，它们难以使用BERT进行编码表示，但在多分类时又和句中词语处于同等地位。由于每个句子中存在这些关系的种类和数量的不确定性，简单地添加特殊token对它们进行表示也不能取得很好的效果。

考虑到以上分析，我们决定将目光转向当今发展迅速的大语言模型（LLM）。近年来，大模型性能的快速提高为众多NLP任务的实现提供了一种全新的方式，即构建合适的Prompt进行输入，引导大模型给出问题的答案。对于更为复杂的任务，我们可以采用对大模型进行微调的方式，使用训练集构造合适的Prompt-Response集对大模型进行微调训练，显著提高大模型在解决特定任务方面的能力。在以上分析中，我们已经看到，由于中文意合图体系结构定义的复杂性，传统模型的实现会变得格外繁琐，且种种限制下也很难取得令人满意的分类性能。同时，若只是采用Zero-shot或Few-shot的方式让大模型完成该任务，Prompt中的描述甚至不足以将中文意合图符号的完整定义表达清楚。综合以上原因，我们决定将本次任务转化为大模型微调。

具体而言，我们首先对数据集进行格式化处理构造Prompt和Response，以便大模型更好地理解任务内容。接着，我们广泛采用目前中文性能较好的各种开源与闭源大模型进行微调并进行性能对比，其中开源大模型包括通义千问发布的Qwen-1.5系列 (Bai et al., 2023)中7B、14B、32B、72B四个参数规模的模型以及百川智能发布的Baichuan2-7B (Yang et al., 2023)和零一万物发布的Yi-1.5-9B (Young et al., 2024)，闭源大模型包括百度文心系列的ERNIE4.0-Speed-8K。最后，我们将测试集的Prompt输入大模型，得到相应的推理结果并进行后处理，丢弃无用的脏数据并转化为符合要求的格式。

3 数据预处理

本次任务是对于每一个给定的已分词句子，输出其相应的中文意合图三元组表示。原始数据集中每一个句子及其中包含的所有三元组被以字典的形式给出，若直接输入大模型显然太过繁琐。因此，我们要采用合适的策略在原始数据集之上构造Prompt和Response，以尽可能高的效率和大模型传递信息。同时，考虑到原始训练集为2000条，验证集为1000条，我们对其进行重新配比，取3000条数据中的5%作为验证集，其余均作为训练集。

©2024 中国计算语言学大会
根据《Creative Commons Attribution 4.0 International License》许可出版

3.1 Prompt构造

在原始数据集中，句子作为字典中的一个键值对，键为固定字符串“sent”，值为一个列表，列表中的元素为按顺序排列的分词结果。由于我们是使用Prompt-Response对大模型进行微调，输入的Prompt只需要为分词后的句子即可。因此，我们丢弃原始数据集中的冗余信息，只保留分词信息，以字符‘/’按照原始的分词信息分隔句子，并且句子的末尾也以字符‘/’标记。表1为一个Prompt构造的示例。

原始句子	“sent”: [“你”, “还是”, “自己”,“观察”, “吧”, “。”]
Prompt构造	你/还是/自己/观察/吧/。/

Table 1: Prompt构造示例

3.2 Response构造

在原始数据集中，句子中的每一个三元组关系都被编码成一个字典加入关系列表中，包含词语内容和索引编号的信息。由于采用字典结构，原始的关系编码中存在大量的冗余字符，如重复出现的“word1”“relData”等键名。因此，我们将原始数据集中的关系字典按照三元组的格式构造成Response，省略了大量重复的内容，只结构清晰地保留需要传递的词语内容、索引编号、关系名。构造的三元组中，前两个元素是(词，位置) 的二元组，第三个元素是关系名的字符串。同一句子的多个关系三元组之间不使用任何分隔符。表2为一个Response构造的示例。

原始关系字典	“relData”: [{ “word1”: {“word”: “你”, “idx”: 0},“word2”: { “word”: “Ref”,“idx”: -4},“relVal”: “Entity”},{ “word1”: { “word”: “还是”, “idx”: 1 }, “word2”: {“word”: “观察”,“idx”: 3},“relVal”: “Mod”}]
Response构造	(('你', 0), ('Ref', -4), 'Entity')(('还是', 1), ('观察', 3), 'Mod')

Table 2: Response构造示例

4 模型微调

4.1 微调方式选择

在大模型微调领域，目前常用的方案主要有两种，即全参数微调和LoRA (Hu et al., 2021)微调。全参数微调对整个模型的参数进行更新，需要耗费大量的显存，而LoRA方式利用低秩矩阵的性质，能够在保证性能的前提下大大减少所需更新的参数量，显著降低了对显存的需求。本次任务我们决定采取LoRA方式进行微调。

4.2 损失函数

由于我们将中文意合图三元组的构建任务转化为了大语言模型的序列生成任务，微调过程中的优化目标是最小化负对数似然损失函数。具体到本任务而言，优化目标的形式化描述如下。

设训练集$\mathcal{D} = \{(x^{(i)}, y^{(i)})\}_{i=1}^{N}$，其中$x^{(i)}$ 是第i个包含分词信息的句子，$y^{(i)}$ 是该句所包含的中文意合图三元组按前文格式拼接成的序列，即

$$y^{(i)} = y_1^{(i)} \oplus y_2^{(i)} \oplus \cdots \oplus y_{t_i}^{(i)} \tag{1}$$

其中，$y_j^{(i)}$ 为第i 个句子中包含的第j 个格式化的中文意合图三元组，t_i 表示第i 个句子中包含的三元组数量。

我们需要通过优化模型参数来最小化生成序列的负对数似然损失，即

$$\min_{\theta} -\sum_{i=1}^{N} \log P(y^{(i)} \mid x^{(i)}; \theta) \tag{2}$$

(c) 2024 中国中文信息学会计算语言学专业委员会

其中，$P(y^{(i)} \mid x^{(i)}; \theta)$表示模型在给定已分词句子$x^{(i)}$的情况下预测出该句的中文意合图三元组序列$y^{(i)}$的概率，该概率由模型参数$\theta$确定。

4.3 环境和参数

对于开源模型微调，我们的硬件平台采用8块4090显卡阵列，根据模型大小分配合适的显卡数量；软件平台采用CentOS7.9操作系统，安装LLaMA-Factory (Zheng et al., 2024)，使用DeepSpeed (Rasley et al., 2020) ZeRO-3平均分配显存。针对参数规模较小的模型，如Qwen-1.5-7B、Baichuan2-7B和Yi-1.5-9B，我们使用两块4090显卡进行LoRA微调；中等参数规模的Qwen-1.5-14B模型使用四块4090显卡进行LoRA微调；大参数规模的Qwen-1.5-32B模型使用八块4090显卡进行LoRA微调。对于超大参数规模的Qwen-1.5-72B模型，八块4090显卡阵列已经不能满足LoRA微调的显存需要，因此我们采取FSDP (Zhao et al., 2023)结合QLoRA (Dettmers et al., 2024)的方式，基于4bit量化进行微调。对于闭源的文心系列ERNIE4.0-Speed-8K模型，我们使用百度智能云的千帆大模型平台创建微调任务。

在超参数的选择上，开源模型根据不同基座模型的官方文档并综合考虑参数规模进行配置。对于ERNIE4.0-Speed-8K闭源模型，我们也采用LoRA方法进行训练，超参数如下表所示。

超参数	数值	说明
迭代轮次	5	过小可能欠拟合，过大可能过拟合，根据训练集大小调整
梯度累计步数	0	累加多次梯度一次性进行更新，取0代表让千帆平台自动计算
学习率	0.0003	过高或过低都会影响微调收敛，使用平台默认值
LoRA所有线性层	True	启用后可以提高模型表达能力，但计算量也会增加
LoRA 策略中的秩	8	过高增加计算复杂度，过低可能限制模型性能
学习率调整计划	linear	训练时学习率的变化方式，千帆平台默认为线性
序列长度	4096	每个输入序列的最大长度，根据数据集输入长度调整
随机种子	42	用于结果复现的随机种子
预热比例	0.1	训练开始时学习率预热阶段所占比例
正则化系数	0.01	用于防止过拟合，但过大可能导致性能下降

Table 3: ERNIE4.0-Speed-8K超参数

5 数据后处理

虽然我们使用标准格式的Prompt-Response对大模型进行训练，但考虑到大模型生成内容具有一定的随机性，大模型生成的内容中可能存在部分不符合格式要求的脏数据。因此，我们在将大模型生成的内容转化为最终预测结果的过程中，需要对相关数据进行后处理操作，具体包括以下措施。需要说明的是，以下特殊情况出现的概率较小，大模型的推理结果总体上是符合格式要求的。

- **忽略无效行**：某些输入句子的内容可能会触发预训练大模型中的判定机制，得到类似“作为一个人工智能语言模型，我还没学习如何回答这个问题”的回复。对于这种情况，我们的后处理程序需要忽略该行内容并返回空的关系三元组列表。
- **确保索引值为数字**：大模型可能在本该是数字的索引值位置返回其他字符，这会导致提供的F1计算代码抛出错误。对于此类三元组，我们认为大模型没有预测出准确的索引值，将问题处索引值置为0。
- **删去格式错误的三元组**：大模型返回的三元组还可能出现一系列的格式错误，例如字符串缺少引号导致出现SyntaxError，缺少元素导致出现IndexError等等不可预知的情况。因此，我们在逐个处理三元组时使用try/except语句，对于一切属于Exception类的错误均跳过。

在完成上述特殊情况的处理后，我们得到了严格符合前文所述Response编码的数据，再转化为题目所需标准的字典格式，并按照所给分词结果加入原始语句，最后写入json文件即可。

6 实验结果

6.1 实验数据

我们首先测试采用普通LoRA方式进行微调的五个开源模型和一个闭源模型，经过量化的Qwen-1.5-72B模型在6.4中进行评估。下表为各开源模型使用LoRA方式的资源消耗和训练推理速度的有关数据。表中'-'表示微调工具不支持反馈该模型的该项数据。

模型	4090显卡数	每秒训练样本数	每秒推理样本数
Baichuan2-7B	2	1.647	-
Qwen-1.5-7B	2	1.452	2.739
Qwen-1.5-14B	4	0.881	1.648
Qwen-1.5-32B	8	0.985	1.937
Yi-1.5-9B	2	1.163	2.334

Table 4: 训练和推理速度报告

我们用测试集1000条语句对以上模型进行评测，结果如下表所示。其中，我们选择表现最好的ERNIE4.0-Speed-8K以及在开源模型中表现最好的Yi-1.5-9B绘制训练损失变化曲线。

模型	Precision	Recall	F1
Baichuan2-7B	0.4739	0.4861	0.4800
Qwen-1.5-7B	0.5294	0.5425	0.5359
Qwen-1.5-14B	0.5083	0.5174	0.5128
Qwen-1.5-32B	0.5226	0.5328	0.5276
Yi-1.5-9B	0.6003	0.6039	0.6021
ERNIE-Speed	0.6778	0.7142	0.6956

Table 5: 各模型的性能指标

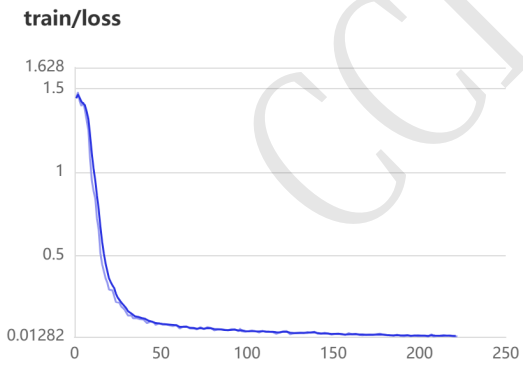

Figure 1: ERNIE-SPEED损失变化

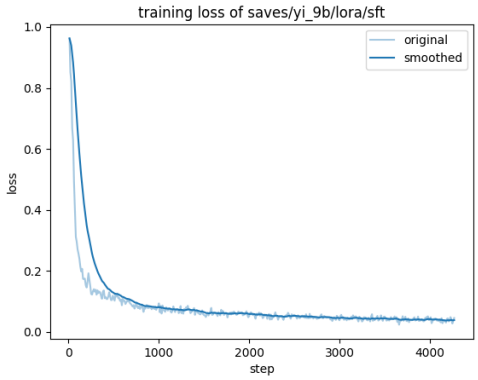

Figure 2: Yi-1.5-9B损失变化

6.2 性能分析

通过分析以上模型的F1分数和损失变化，我们初步得到如下结论。

- **大模型表现普遍优异：**排除受量化影响的72B模型，即使表现最差的Baichuan2-7B模型也达到了0.48的F1分数，表现最好的文心模型更是达到了接近0.7的F1分数，远高于依存模型的表现。在过去的一年中各类大模型的性能提升迅速，在诸多任务上经过微调性能可以达

(c) 2024 中国中文信息学会计算语言学专业委员会

到或超过传统模型，而在一年前大模型的表现通常不及传统模型。使用大模型完成NLP任务逐渐成为一种可靠的选择。

- **文心闭源模型性能突出**：经过我们的测试，一众开源模型微调后的F1分数都大致处于0.4-0.6的区间；而文心闭源模型在测试集的F1分数接近0.7，提交评测后在盲测集的F1分数超过了0.72，与开源模型之间拉开了明显的差距。对比训练集最终损失，开源模型中最优的Yi-1.5-7B模型只能降低到0.077，而文心模型降低到了0.013。

由于ERNIE4.0-Speed-8K闭源模型的详细参数和技术细节均未公布，为了进一步探究各种因素对于模型性能影响的程度，我们采取控制变量的思想对开源模型的实验数据进行对照评估，分别研究参数规模和基座模型系列两个因素的影响。

6.3 对照评估

6.3.1 参数规模

首先，我们探究参数规模对模型性能的影响。为此，我们选择同属Qwen-1.5系列的7B、14B、32B三种参数规模模型进行对比。这三个模型属于同一研发团队的同一系列，能有效排除无关因素的影响。超参数选择上，根据模型大小略有不同，但均保证足够的迭代轮数，并观察到训练结束时损失曲线趋于平稳。下表为各模型在测试集1000条语句上的评估结果。

模型	Qwen-1.5-7B	Qwen-1.5-14B	Qwen-1.5-32B
Precision	0.5294	0.5083	0.5226
Recall	0.5425	0.5174	0.5328
F1	0.5359	0.5128	0.5276

Table 6: 同系列基座不同参数规模对比

我们发现，参数规模对模型性能的影响较小。虽然14B模型和32B模型的参数量分别为7B模型的两倍和四至五倍，但在本任务上的表现却完全处于同一水平，并不符合参数越多性能越好的直观认知。

6.3.2 基座系列

接着，我们探究基座模型系列对模型性能的影响。为此，我们选择Baichuan2-7B、Qwen-1.5-7B和Yi-1.5-9B三个开源模型，它们拥有大致相等的参数规模。同时，我们采用完全相同的超参数和工具进行微调，最大程度上排除了其他因素的影响。其中，迭代轮数均设置为6，训练损失曲线后期均趋于平稳。下表为各模型在测试集1000条语句上的评估结果。

模型	Baichuan2-7B	Qwen-1.5-7B	Yi-1.5-9B
Precision	0.4739	0.5294	0.6003
Recall	0.4861	0.5425	0.6039
F1	0.4800	0.5359	0.6021

Table 7: 不同系列基座相近参数规模对比

我们看到，三个参数规模相近的开源模型采用完全相同的超参数，但F1分数却相差很大，其中最低的Baichuan2-7B模型F1分数仅有0.48，最高的Yi-1.5-9B模型F1分数超过了0.60，这充分说明了基座模型对最终性能的影响。由于来自不同研发团队的基座模型的技术细节和预训练语料均有所不同，模型性能和所擅长领域均存在差异。

综合以上分析，我们认为文心系列ERNIE4.0-Speed-8K模型在本任务上的出色表现主要并不来源于参数规模，而是可能与百度的预训练语料、模型的技术细节以及千帆平台的超参数组合等因素更为相关。

(c) 2024 中国中文信息学会计算语言学专业委员会

6.4 量化损失与错误分析

如前文所述，由于Qwen-1.5-72B模型参数规模很大，八块4090显卡已经不能满足LoRA微调的显存需要，因此我们采取FSDP (Zhao et al., 2023)结合QLoRA (Dettmers et al., 2024)的方式，基于4bit量化进行微调。这种方式使用4bit取代原有浮点数精度对模型的权重进行量化，可以有效降低显存需求，但也会带来模型性能的损失。经过初步测试，4bit量化后的Qwen-1.5-72B模型在测试集前250条语句上的F1值仅为0.2173，远低于上文中所有未经过量化的模型。

为了对模型表现和量化损失有更具体的了解，我们对模型的预测结果进行错误分析。考虑到中文意合图的体系标签定义中除了显式出现的词语，还存在“ROOT”“因果关系”和“ls”等隐式事件词 (Guo et al., 2024)，我们将关系集合分为“只包含显式事件词”和“包含隐式事件词”两类，分别评估模型在这两类关系上的预测表现。下表为Qwen-1.5系列和ERNIE-Speed在测试集前250条语句上错误分析的结果。表中*标识说明微调和推理过程对模型进行了量化处理。

模型	显式F1	隐式F1	总F1
Qwen-1.5-7B	0.5853	0.4575	0.5482
Qwen-1.5-14B	0.5870	0.4232	0.5390
Qwen-1.5-32B	0.5797	0.4466	0.5400
Qwen-1.5-72B*	0.2142	0.2266	0.2173
ERNIE-Speed	0.7663	0.5929	0.7137

Table 8: 部分模型的错误分析

- **隐式错误率更高：** 对于未量化的模型而言，在“只包含显式事件词”的关系上F1均明显高于“包含隐式事件词”的关系，说明在隐式事件词关系上错误率更高。考虑到隐式事件词关系的自由度更高、预测难度更大，该结果符合一般预期。
- **量化损失严重：** 对于经过量化的Qwen-1.5-72B，其显式和隐式两类关系的F1均显著低于同系列未经过量化的模型，说明在本次实验中量化损失是全面的。与Qwen-1.5-32B相比，两类F1下降幅度分别为0.3655和0.2200，在“只包含显式事件词”的关系上性能损失更加严重。

7 结语

在本次任务中，我们分析了使用传统方法完成中文意合图语义解析所遇到的困难，并且阐述了选择大模型微调的原因。通过对数据集和生成内容进行有效预处理和后处理操作，我们测试了七个主流大模型微调后的性能表现，并最终选择文心系列ERNIE4.0-SPEED-8K模型的预测结果进行评测。该模型在测试集和盲测集上分别取得0.6956和0.7206的F1分数，获得本次评测榜一的成绩。我们工作的主要贡献有：

- 将大语言模型微调的方法引入中文意合图语义解析，取得远超依存模型的性能，获得本次评测榜一的成绩，证明了该方法的有效性。
- 广泛测试多个主流大模型，控制变量地评估参数规模和基座系列对模型性能的影响程度；讨论量化损失对模型性能的影响，并进行简要的错误分析。这些工作对后续优化该任务的大模型方案具有一定的方向性指导意义。

下一步，我们将继续探究全参数微调、更多的超参数设置、不同的Prompt-Response构造方式等因素对模型性能的影响。

致谢

本工作受北京理工大学计算机学院辛欣老师所开《知识工程》课程的启发，感谢辛欣老师对相关工作的指导与支持。感谢CCL评测组委会和北京语言大学任务组织者荀恩东老师、饶高琦老师、唐共波老师以及任务联系人郭梦溪、李梦的支持。

(c) 2024 中国中文信息学会计算语言学专业委员会

参考文献

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2024. Qlora: Efficient finetuning of quantized llms. *Advances in Neural Information Processing Systems*, 36.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3505–3506.

Aiyuan Yang, Bin Xiao, Bingning Wang, Borong Zhang, Ce Bian, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, et al. 2023. Baichuan 2: Open large-scale language models. *arXiv preprint arXiv:2309.10305*.

Alex Young, Bei Chen, Chao Li, Chengen Huang, Ge Zhang, Guanwei Zhang, Heng Li, Jiangcheng Zhu, Jianqun Chen, Jing Chang, et al. 2024. Yi: Open foundation models by 01. ai. *arXiv preprint arXiv:2403.04652*.

Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, et al. 2023. Pytorch fsdp: experiences on scaling fully sharded data parallel. *arXiv preprint arXiv:2304.11277*.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. *arXiv preprint arXiv:2403.13372*.

郭梦溪, 荀恩东, 李梦, 饶高琦. 2024. 意合图：中文多层次语义表示方法. 第二十三届中国计算语言学大会.

郭梦溪, 李梦, 荀恩东, 饶高琦, 于钟洋. 2024. 基于意合图语义理论的结构标注体系与资源建设. 第二十三届中国计算语言学大会.

CCL 2024