 DOI: https://doi.org/10.12349/iser.v5i10.3493

Implementation and Evaluation of Dialogue Systems with BART and Qwen Models

Zhenwei Zheng

School of Computing, The Hong Kong Polytechnic University, Hong Kong, 999077, China

Abstract

This study explores two approaches to implementing dialogue systems. The first approach involves creating a custom dataset class MyDataset to handle dialogue data and designing the MyTrainer class for fine-tuning and training the BART-Chinese model. The second approach utilizes a pre-trained Qwen model combined with knowledge embedding technology. Users can choose to invoke the Qwen model or the BART model via a user interface (UI) for dialogue. The fine-tuning of the BART model includes dataset partitioning, data preprocessing, and model training evaluation, ensuring that the data fits the model's requirements. The Qwen model enhances its understanding capability through knowledge graphs, using entity recognition, information retrieval, and similarity calculations to generate responses. The web service is built on Flask, supports cross-origin requests, and uses the Chatbot class to process messages and generate replies. Experimental results show that the BART model performs better in terms of BLEU scores, while the Qwen model has an advantage in generalization ability. Future work will focus on integrating semi-structured knowledge data and dialogue history to improve the context-awareness of the models.

Keywords

BART model fine-tuning; Qwen mode; dialogue system; knowledge embedding; Flask Web service

BART 与 Qwen 模型的对话系统实现及评估

郑振魏

香港理工大学计算机学院，中国 · 香港 999077

摘　要

本研究探讨了两种实现对话系统的途径：一是通过自定义数据集类MyDataset处理对话数据，并设计MyTrainer类进行BART-Chinese模型的微调与训练；二是利用预训练的Qwen模型结合知识嵌入技术。用户可通过UI选择调用Qwen模型或BART模型进行对话。BART模型微调涉及数据集划分、数据预处理da及模型训练评估，确保数据适配模型要求。Qwen模型利用知识图谱增强理解能力，并通过实体识别、信息检索及相似度计算生成回复。Web服务基于Flask构建，支持跨域请求，通过Chatbot类处理消息并生成回复。实验结果显示，BART模型在BLEU分数上表现更佳，但在泛化能力上Qwen模型占优。未来工作将聚焦于整合半结构化知识数据和对话历史，以提升模型的上下文理解能力。

关键词

BART模型微调；Qwen模型；对话系统；知识嵌入；Flask Web服务

1 引言

近年来，随着自然语言处理（NLP）技术的进步，对话系统的开发取得了显著进展。在此背景下，出现了两种主要的方法：一种是通过微调像 BART（双向和自回归变换器）这样的预训练模型，特别是针对中文环境进行优化；另一种则是利用诸如 Qwen 这样的预训练模型，结合知识嵌入技术来增强其理解能力。这两种方法均致力于构建能够生成连贯且符合上下文回应的复杂对话系统 [1-3]。

BART 模型最初由 Facebook AI Research（FAIR）开发，在多种自然语言处理任务中展现出了卓越的性能。当通过微调 [4] 适应中文语言环境时，BART 展现出了增强的语义理解和生成能力，使其特别适用于需要准确流畅沟通的对话系统。微调的过程涉及在特定数据集上调整模型参数，从而使模型能够学习任务特有的模式和细微差别，进而提高其在未见过的数据上的表现。由阿里巴巴云开发的 Qwen 模型，因其强大的知识图谱整合能力而脱颖而出。Qwen 模型在泛化能力方面表现优异，意味着它可以将在训练过程中学到的原则应用到新的情境之中。其从知识库中引入背景信息的能力使其特别擅长提供富有信息量且细节丰富的回应。鉴于这两种模型各自的优势，本研究探讨了在统一框架内实现这两种方法的实际应用。我们研究了用户如何通过友好的用户界面与基于 BART 或 Qwen 的对话系统进行互动，使他们能够选

【作者简介】郑振魏（1999–），男，中国福建人，硕士，从事人工智能、大语言模型及大数据研究。

择最适合其需求的模型 [5]。本研究深入探讨了数据集准备、模型训练、评估指标以及部署策略等方面的细节，提供了关于每种方法面临的挑战和收益的洞见，如图 1 所示。

此外，我们强调了开发支持实时交互的 Web 服务的重要性，如基于 Flask 框架的服务，这些服务不仅促进了用户与对话系统之间的无缝通信，而且还确保了复杂机器学习模型的部署以及易用性和可访问性，这对于实际应用中的广泛应用至关重要。总之，论文旨在为有关构建先进对话系统的持续讨论做出贡献，提供了如何优化性能并提升对话人工智能用户体验方面的宝贵视角。

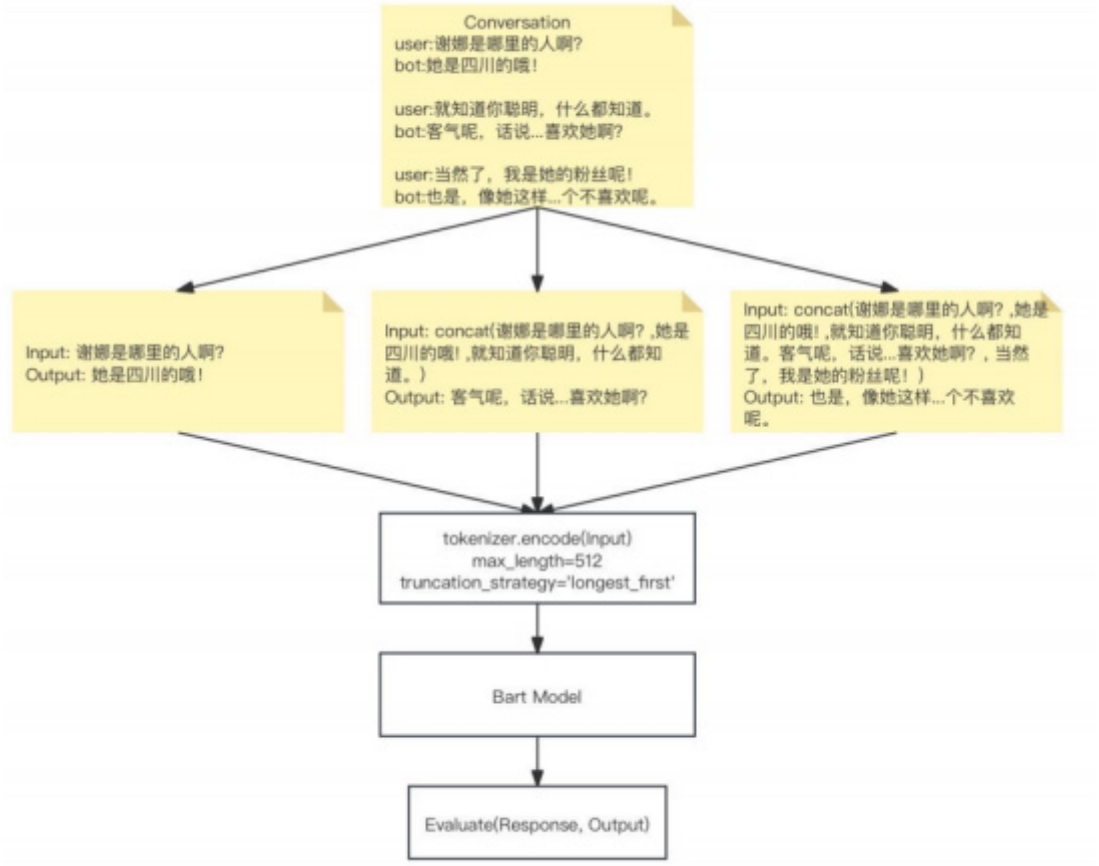


图 1

2 实验原理

2.1 BART 微调：数据集划分与输入方法

数据集划分：数据集被划分为三个独立的部分：训练集、验证集和测试集。这些数据子集的文件路径分别通过配置参数 `train_file`、`dev_file`（有时也称为 `val_file` 或 `validation_file`）和 `test_file` 来指定。在机器学习领域，这种划分是为了确保模型可以在不同的数据集上进行训练、验证和最终的性能评估。训练集用于模型的学习过程，验证集用于调整模型参数（如超参数）和避免过拟合，而测试集则用于评估模型在未见过的数据上的泛化能力。

数据输入方法：在 `MyDataset` 类的初始化方法 `__init__` 中，依据 `data_partition` 参数来确定当前实例所属的数据集角色（训练集、验证集或测试集）。随后，根据指定的文件路径加载相应的数据文件。在加载数据之后，使用 `_cache_instances` 方法对其进行处理。首先，打开指定的数据集文件，并逐行读取对话数据。每一行都会被解析以提取对话历史及其对应的回复。之后，对话历史和回复交替出现，每一轮包含用户输入后跟着模型的回复，如图 2 所示。

接下来，每组对话会进行进一步的预处理。移除不必要的标记，并使用分词器（tokenizer）对对话内容进行分词（tokenization），以创建模型所需的标记序列（token sequence）。在序列的开头和结尾添加特殊的标记（special tokens），如 `<s>` 和 `</s>`，以帮助模型正确地理解输入。在这个转换过程中，对话历史和回复会被格式化为模型可接受的输入格式（input format），如果对话长度超过了指定的最大长度（max_length），则会被截断（truncation）。用户输入和模型回复会被直接拼接（concatenation）在一起。

最后，处理后的对话历史和回复被组合成一个样本，并添加到 `self.data` 列表中以备后续使用。该方法的实现步骤包括数据清洗（data cleaning）、分词（tokenization）、特殊标记插入（special tokens insertion）、截断以及数据格式化。

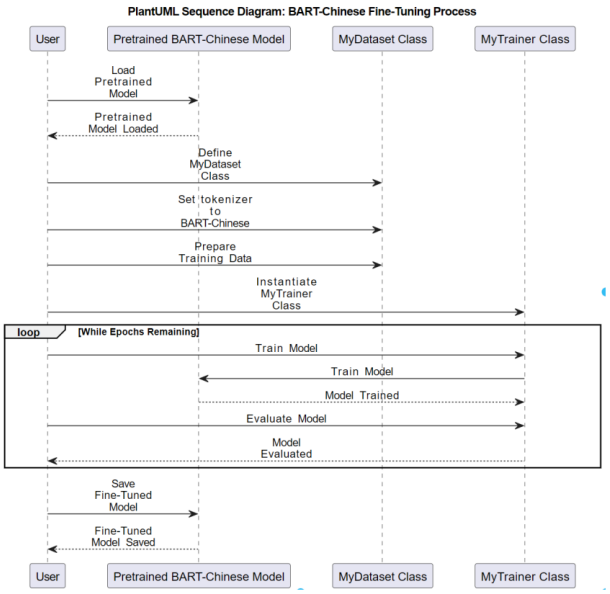


图 2

处理函数：处理函数将对话历史和响应转换为模型所需的输入格式。该函数首先检查对话历史是否超出最大序列长度，并在必要时进行截断（truncation）。然后，在截断后的对话历史和响应前添加序列起始标记（sequence start tokens），并在响应的末尾添加序列结束标记（sequence end tokens）。最后，将处理后的序列存储在一个字典中作为样本实例，确保它们准备好用于模型训练和推理。

2.2 BART 微调：训练、评估与测试

加载预训练的 BART-Chinese 模型。然后，在 `MyDataset` 类的初始化方法中，通过传递 `model_name='bart_chinese'` 参数来指定使用与 BART-Chinese 相关的分词器（tokenizer），以确保数据集的预处理与模型相匹配。接下来，在 `MyTrainer` 类中，传递加载的 BART-Chinese 模型，并对其进行微调。在训练和评估期间，使用与 BART-Chinese 相关的分词器进行数据预处理，以保证与模型的兼容性。

使用 `self.model.named_parameters()` 加载模型的所有命名参数，并将它们的 `requires_grad` 属性设置为 `True`，以确保在训练过程中计算这些参数的梯度。然后，在每个训练周期（epoch）中，遍历训练数据加载器（data loader）以获取输入数据批次，并将模型设置为训练模式（model.train()），以跟踪训练步骤。

我们假设梯度下降的算子为$\nabla_\theta L(\theta)$：

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)\nabla_\theta L(\theta)$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2)\left(\nabla_\theta L(\theta)\right)^2$$

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}$$

$$\hat{v}_t = \frac{v_t}{1-\beta_2^t}$$

$$\theta_t = \theta_{t-1} - \alpha\frac{\hat{m}_t}{\sqrt{\hat{v}_t}+\epsilon} - \lambda\theta_{t-1}$$

对于每个批次的数据，将输入数据馈送到模型中进行前向传播[6]（forward pass），获取模型的输出和损失值。每次参数更新后，清空梯度缓冲区（optimizer.zero_grad()），以准备下一次批次的更新。此外，在达到指定的训练步数时，打印训练日志，包括当前训练步骤和总损失。

$$P_\theta(y_t|y_{<t})$$

$$L(\theta) = -\sum_{t=1}^{T}\log P_\theta(y_t|y_{<t})$$

评估时，将模型设置为评估模式（model.eval()）并初始化分词器进行分词。然后，遍历验证数据集，通过模型进行前向传播以计算预测结果和损失。随后，基于预测结果和标签计算 BLEU 分数，累积每个批次的 BLEU 分数，并最终计算整个验证数据集上的平均 BLEU 分数。公式为 `avg_bleu=sum(bleu_scores)/len（bleu_scores）`。最后，输出评估结果，提供模型在验证集上的性能指标。

2.3 Qwen API 和知识嵌入

“千问”是由阿里云开发的一款极其强大的语言模型。它基于知识图谱和自然语言处理技术，旨在回答各种类型的问题。事实上，它是首批通过大规模模型合规标准评估的四个国产大型模型之一。我们认为知识是以实体、属性和值组成的三元组形式排列的，代表了不需要文档分段或固定块长度的半结构化数据。此外，提到的知识数据共有 30258 条，但只有 2208 个不同的实体。在查询处理方面，方法包括首先通过命名实体识别（NER）识别查询中的实体，然后匹配相关实体数据以缩小搜索范围。在检索到的实体数据中，选取三个最相关的作为背景知识来补充语言模型（LLM）。

在 `call_with_messages` 方法中，第一步是检查输入文本是否为空。如果为空，则返回提示信息和当前对话历史。如果输入文本不为空，则执行以下步骤。首先，调用 `extract_entities` 方法从输入文本中抽取实体，这些实体可以是关键词、命名实体或其他对话中的重要信息。`top_info` 列表存储与输入文本最相关的前五条信息。这五条相关信息随后被添加到知识列表中，以供未来对话使用。这样，知识列表累积并存储与用户输入相关的信息。构造用户输入消息，并将其作为对话历史的一部分传递给生成 API。调用生成 API 生成响应。将知识列表、对话历史和用户输入消息作为输入传递给生成 API 以生成响应。通过将生成的响应添加到对话历史中来更新对话历史。生成的响应内容作为方法的输出返回。

3 结果分析

随着 epoch 数量的增加，BART 模型的 BLEU 分数不断改善。起初，改善显著，但后来改善速率逐渐放缓。另一方面，Qwen 模型的 BLEU 分数也在增加，但最终表现出过拟合的迹象，BLEU 分数不再增加。

$$R_{LCS} = \frac{LCS(C,S)}{\text{len}(S)}$$

$$P_{LCS} = \frac{LCS(C,S)}{\text{len}(C)}$$

$$ROUGE - L = F_{LCS} = \frac{(1+\beta^2)R_{LCS}P_{LCS}}{R_{LCS} + \beta^2 P_{LCS}}$$

Qwen 模型的 ROUGE 分数趋势与 BLEU 分数相似。起初，随着训练的进行，分数有所提高，但最终达到过拟合点，进一步的改进不再发生。

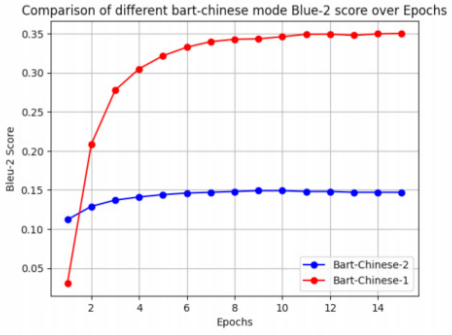
$$BP = \begin{cases} 1 & c > r \\ e^{1-\frac{r}{c}} & c \le r \end{cases}, c \leftarrow \text{len(candidate)}, r \leftarrow \text{len(reference)}$$

$$\text{BLEU} = \text{BP} * \exp\left(\sum_{n=1}^{N} w_n \log p_n\right).$$

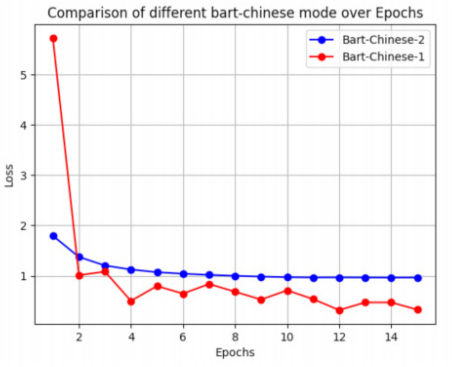
排名行为在对数域中更为直接明显，

$$\log \text{BLEU} = min\left(1 - \frac{r}{c}, 0\right) + \sum_{n=1}^{N} w_n \log p_n$$

$$\text{Count}_{\text{clip}}(n - \text{gram}) = min(\text{Count}, \text{Max\_Reference\_Count})$$



(a) The comparison of bleu score of different bart chinese model every epoch



(b) The comparison of Loss of different bart chinese model every epoch

图 3

BLEU 基于分词计算相似度，这意味着它关注的是匹配个别单词或短语，而不考虑整体意义或上下文。因此，BLEU 分数可能无法准确评估生成回复的质量和连贯性。我们选择了两类问题来评估训练效果。第一类问题同时出现在训练和测试数据集中，这类问题有助于评估模型从训练数据中学习和泛化的能力。第二类问题旨在测试局部知识库的有限范围。我们观察到 BART 仅限于训练数据集中的信息，而 LLM 显示出更强的泛化能力，如图 3 所示。

4 结论与未来工作

基于我们的实验和对比，我们发现微调后的 BART 中文模型的表现优于使用 Qwen API。这一发现表明，对预训练的 BART 模型进行微调增强了其语义理解能力，并使其更好地适应中文环境，从而生成更准确和连贯的回复。在未来的工作中，我们在嵌入半结构化知识数据时遇到了挑战。由于这不是纯粹的自然语言，直接嵌入的方法往往产生平庸的结果。此外，实体之间的关系无法在文本嵌入中有效表示。未来，我们计划以图形化的方式结构化局部知识数据，以增强搜索速度并提高匹配的相关性。另一个未来工作的领域是实现通过文件输入整合对话历史的功能。

参考文献

[1] 郭锦颖.基于BART模型的情感对话生成技术研究[D].长春:长春工业大学,2024.

[2] 李佳沂,黄瑞章,陈艳平,等.结合提示学习和Qwen大语言模型的裁判文书摘要方法[J/OL].清华大学学报(自然科学版),1-12[2024-10-19].https://doi.org/10.16511/j.cnki.qhdxxb.2024.21.028.

[3] 常保发,车超,梁艳.基于大语言模型多轮对话的推荐模型研究[J/OL].计算机科学与探索,1-15[2024-10-19].http://kns.cnki.net/kcms/detail/11.5602.tp.20241016.1506.007.html.

[4] 冯拓宇,李伟平,郭庆浪,等.大语言模型增强的知识图谱问答研究进展综述[J/OL].计算机科学与探索,1-17[2024-10-19].http://kns.cnki.net/kcms/detail/11.5602.TP.20240929.1255.004.html.

[5] 李晓庆,马昊聪,周军华,等.基于大语言模型的知识图谱构建与应用[C]//中国仿真学会.第三十六届中国仿真大会论文集.北京市复杂产品先进制造系统工程技术研究中心北京仿真中心;复杂产品智能制造系统技术国家重点实验室北京电子工程总体研究所;航天系统仿真重点实验室北京仿真中心,2024:11.

[6] 杨钦惠.基于FPGA的循环神经网络前向传播加速技术研究[D].重庆:电子科技大学,2023.