文章编号：1003-0077（2017）00-0000-00

**基于大语言模型的BCC语料库自然语言检索技术研究**

作者11 作者21,2 作者31,2

（1.一级单位名称 二级单位名称，省 市 邮编；2.一级单位名称 二级单位名称，省 市 邮编）

**摘要：**语料库是语言学研究和自然语言处理等领域不可或缺的资源。北京语言大学研制的BCC语料库因其丰富的内容和高效的检索系统而备受关注。然而，传统的检索方式对非专业用户来说具有一定的使用难度。本文提出了一种基于大语言的解决方案，旨在降低BCC语料库的使用门槛，通过构建均衡的BCC检索式数据集，将 BCC检索式与自然语言联合训练，实现了自然语言到BCC检索式的自动转换。实验结果表明，本文提出的模型在检索式转换的精确匹配率和执行正确率上具有较高的性能，同时在BCC检索式纠错能力上也表现出色，有效降低了BCC语料库的使用难度，优化了用户体验。

**关键词：**检索表达式；大语言模型；微调

**中图分类号：**TP391  **文献标识码：**A

#### Research on BCC corpus natural language retrieval technology based on large language model

Author1 , author1,2 , and author1,2

(1.Unit ,city, province zip code, China ; 2. Unit, city, province zip code ,China)

**Abstract:** Corpora are an indispensable resource in areas such as linguistic research and natural language processing. The BCC corpus developed by Beijing Language and Culture University has attracted much attention because of its rich content and efficient retrieval system. However, the traditional search method is difficult for non-professional users. This paper proposes a solution based on large language, which aims to lower the threshold for the use of BCC corpus, and realizes the automatic conversion from natural language to BCC search formula by constructing a balanced sample database of BCC search formulas and combining BCC search queries with natural language and natural language training. Experimental results show that the proposed model has high performance in the exact matching rate and execution accuracy of search query conversion, and also performs well in the error correction ability of BCC search query, which effectively reduces the difficulty of using BCC corpus and optimizes the user experience.

**Key words:** search expression; large language models; fine-tuning

**0 引言**

在信息时代的浪潮下，语言数据的海量增长为语言学研究和自然语言处理带来了前所未有的机遇与挑战。其中，语料库作为语言研究中的一种重要工具，其价值和重要性日益凸显。

目前，中文语料库的建设已取得了显著成果，如北京语言大学的BCC语料库[1][2][3]（Beijing Language and Culture University Corpus Center，BCC）、北京大学的CCL现代汉语语料库[4]（Center for Chinese Linguistics Peking University，CCL）以及国家语言资源动态流通语料库[5]（Dynamic Circulation Corpus，DCC）等。这些语料库各具特色，涵盖了丰富的语言资源，为不同领域的研究和应用提供了有力支持。然而，由于语料库在定位和功能上存在着差异，其检索式设计也各不相同，这使得用户在使用不同语料库检索语料时面临一定的困难。

在自然语言处理领域，自然语言查询作为一种重要的查询方式，正逐渐受到广泛关注。自然语言查询则允许用户以日常语言表达查询需求，大大降低了使用门槛。近年来，文本到SQL的转换任务已经取得了显著的进展，使得开发者能够轻松地将自然语言查询转换为SQL语句，进而实现对数据库的高效检索。然而，由于语料库并没有统一的检索语言，在使用自然语言查询语料库领域的研究尚显不足。

鉴于此，本文提出了文本到语料库检索式转换的研究任务。旨在构建一个全面的框架，实现自然语言到语料库检索式的自动转换，从而降低用户在使用语料库时的难度，提高检索效率。

为了实现这一目标，本文以BCC语料库为例，实现自然语言查询功能。我们首先构建了一个大规模的数据集，该数据集包含了从BCC语料库查询日志中筛选的用户查询实例。通过对这些数据进行分析和整理，揭示了用户查询行为的多样性和复杂性，并针对数据稀缺的问题采用了检索式生成技术进行数据增强。此外，我们还设计了基于大型语言模型的检索式到自然语言的转换方法，通过编写针对性的提示语（Prompt），能够将BCC检索式转换为自然语言查询。

在模型训练方面，采取微调已有大模型的方法。利用我们构建的BCC检索式数据集对大模型进行微调，使其能够满足自然语言查询到BCC检索式的转换任务。为了进一步提高模型的性能和泛化能力，我们还引入了错误BCC检索式的修正数据，对模型进行针对性的优化。通过一系列实验验证，本文提出的方法在文本到BCC检索式转换任务上展现出了优异的性能。

本文的研究不仅为语料库的普及和应用提供了新的可能性，还为语言学研究、自然语言处理等领域的研究者和非专业用户提供了一个高效、易用的检索工具。通过降低使用门槛和提高检索效率，我们的方法有望推动相关领域的研究和应用取得更大的进展。

**1 研究背景**

随着信息技术的不断进步，语料库的规模日益扩大，并开始发挥其重要价值。然而，语料库的检索式形式繁多，这在一定程度上阻碍了其进一步普及和应用。因此，人们期待能够利用自然语言查询技术来降低语料库的使用门槛。随着大型语言模型技术的不断发展，这一愿景逐渐成为现实。

**1. 1 语料库**

语料库（Corpus）是按照一定采样标准采集，能够代表一种语言或语言的一种变体或文类的电子文本集[6]。语料库作为提取语言信息的枢纽，在语言数据挖掘过程中发挥着重要作用[7]。借助语料库，研究者能够对语言的规律性、变异性及历史演变进行深入剖析。此外，语料库为语言教学提供辅助手段[8]。在自然语言处理领域，语料库是训练语言模型、开发智能应用的基础[9]。

目前常用的中文语料库，如BCC语料库、CCL语料库等，在提供的内容和功能上各有千秋，但均采用检索式进行信息检索。语料库检索式是用户用来查询语料库返回所需信息的脚本语言。使用语料库检索式需要具备一定的语言学知识，需要记忆检索式的语法格式、词性标签，以及特殊符号的用法等，这对于非专业用户来说具有一定的难度。

鉴于各语料库的定位和功能有所不同，设计上也呈现出显著差异。现有的语料库系统，各自采用了独特的检索方法和检索表达式，呈现出显著的差异。CCL与BCC语料库均采用自定义的检索表达式，在语法规则和使用方式上存在着较大差别。DCC则选用了基于语料库查询语言（Corpus Query Language，CQL）扩展的检索语言，CQL作为从语料库查询处理器（Corpus Query Processor，CQP）发展而来的一系列语言，被一些语料库采用[10]。尽管语料库检索表达式的多样性彰显了语料库检索工具的强大功能，但同时也暴露出语料库检索语言非标准化的问题。对于专业研究者而言，这意味着在使用不同的语料库时，需要投入时间和精力去学习和适应各自独特的检索语法和规则。

**1. 2 自然语言查询**

自然语言查询（Natural Language Query, NLQ）是以自然语言表达为基础的查询，是自然语言处理的内容之一。在人工智能中，通过用户输入自然语言表达，使计算机能接受和处理自然语言，还能对自然语言进行信息加工，以及能理解自然语言的过程，从而对信息进行反馈，这样的过程称为自然语言查询。

典型的NLQ任务是文本到SQL[11]的转换，实现自然语言查询到SQL语句的转换，使得开发者和最终用户都能够轻松地使用自然语言方式完成数据库查询操作。

本文提出了第一个文本到CQL任务，旨在将自然语言自动翻译成CQL。我们为这项任务提出了一个全面的框架，包括一个专门策划的大规模数据集和利用大型语言模型(llm)进行有效的文本到CQL任务的方法。此外，建立了先进的评估指标，以评估生成的查询的语法和语义准确性。我们创建了创新的基于llm的转换方法和详细的实验。结果证明了我们的方法的有效性，并提供了对文本到CQL任务复杂性的见解。

1 介绍

本文提出了第一个文本到CQL任务，旨在将自然语言自动翻译成CQL。我们为这项任务提出了一个全面的框架，包括一个专门策划的大规模数据集和利用大型语言模型(llm)进行有效的文本到CQL任务的方法。此外，建立了先进的评估指标，以评估生成的查询的语法和语义准确性。我们创建了创新的基于llm的转换方法和详细的实验。结果证明了我们的方法的有效性，并提供了对文本到CQL任务复杂性的见解。

1 介绍

在使用自然语言实现语料库查询方面，文献[12]提出将自然语言自动翻译成CQL的方法。设计了一个转换框架，构建了一个CQL数据集和利用大型语言模型将文本转换为CQL任务的方法。还建立了评估指标，评估生成的CQL的语法和语义准确性。

**1. 3 大语言模型及微调**

大语言模型是近年来自然语言处理领域取得重大突破的关键技术之一。大语言模型通过深度学习的方法，在海量文本数据上进行训练，从而能够生成和理解复杂的自然语言文本。这些模型通常包含数十亿甚至数万亿的参数，使得它们能够捕捉语言中的细微模式和长距离依赖关系，从而在多种自然语言处理任务上展现出前所未有的性能。

随着计算能力的提升和大规模数据集的出现，大语言模型不仅在语言理解（如问答系统、情感分析）方面取得了显著成效，还在语言生成（如文本创作、摘要生成）领域展现出了强大的能力。大语言模型的核心在于其规模，巨大的参数空间允许模型捕捉更丰富的语言特征和上下文信息。模型首先在大量未标注文本上进行预训练，学习语言的通用表示；随后，通过针对特定任务的微调，快速适应不同场景的需求。大语言模型展现出了零样本或少样本[10]学习的能力，即无需或仅需少量标注数据即可完成新任务，极大地降低了自然语言处理应用的门槛。

模型微调（Fine-tuning）作为LLM应用中的关键技术，通过利用特定领域的数据对预训练模型进行针对性优化，显著提升模型在特定任务上的表现。然而，对于庞大的LLM而言，直接微调往往伴随着高昂的计算成本和过拟合风险。

为此，研究者们不断探索新的微调策略，产生了很多微调方法，比较流行的有：Prefix-Tuning [13] 、Prompt Tuning[14] 、P-Tuning[15]、P-Tuning v2[16]等方法，其中LoRA（Low-Rank Adaptation）微调[17]尤为引人注目。

LoRA微调通过向模型中注入可训练的低秩矩阵，有效减少了需要微调的参数数量，从而在保持性能提升的同时，大幅降低了计算复杂度和显存需求。这一方法不仅提升了微调过程的效率与稳定性，还使得LLM在更多资源受限的场景下得以应用。目前，许多开源的LLM已提供了LoRA微调的示例代码，为研究者与开发者提供了便捷的实践路径。

**2 构建BCC检索式数据集**

通过对BCC语料库的检索日志和错误日志的分析，我们构建了一个包含正确检索式和错误检索式两大类别的检索式数据集。首先，我们对这两类日志数据进行分类整理，随后依据既定比例从各分类中提取数据。为了确保数据集的均衡性，我们还制定了一套策略，针对数据量不足的分类进行有效补充。最终，成功建立了一个平衡且全面的BCC检索式数据集。BCC检索式数据集的生成流程如图1所示。

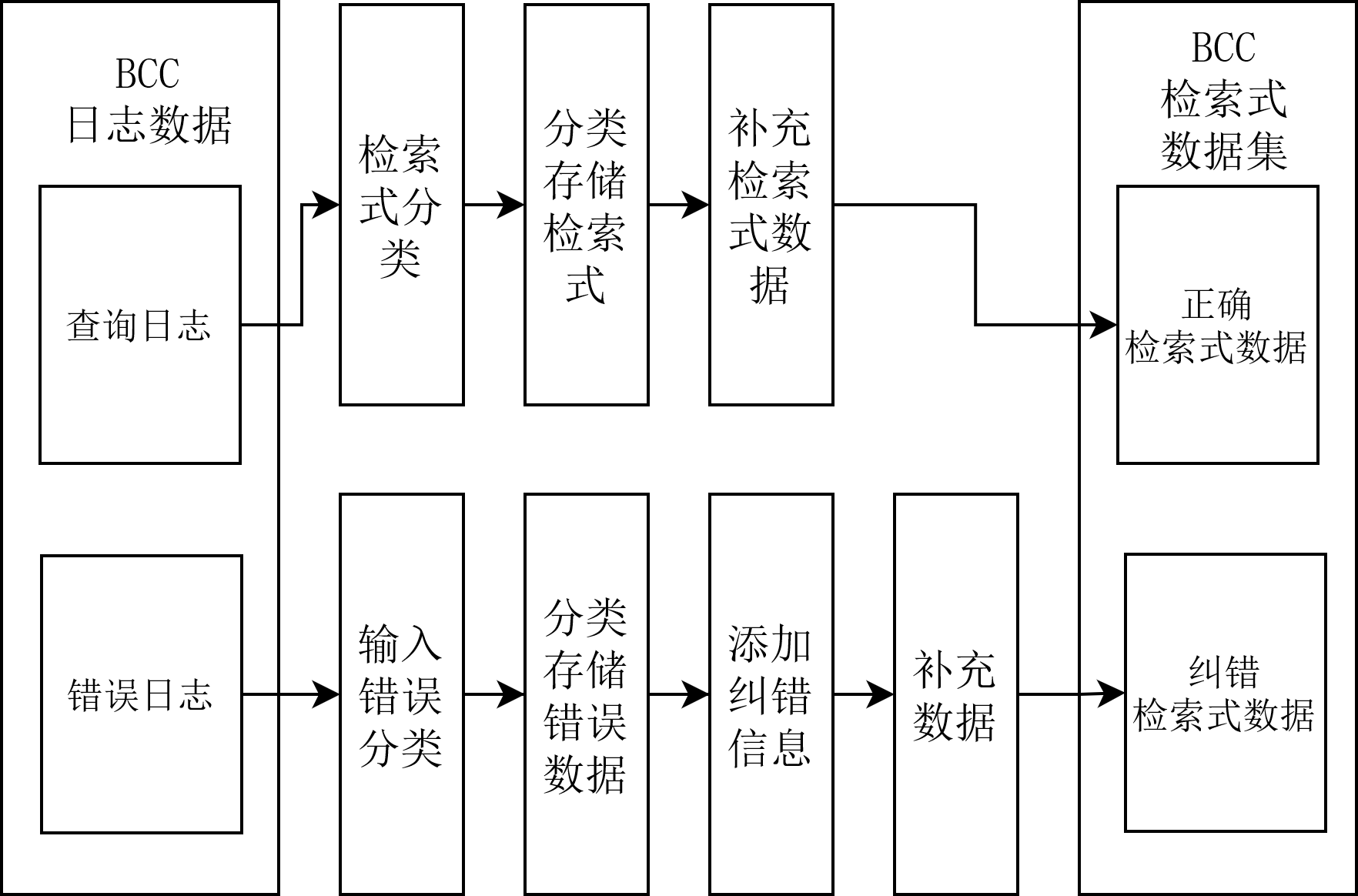


图1 BCC检索式数据集生成流程图

**2. 1 BCC语料库**

北京语言大学研制的现代汉语语料库（BLCU Chinese Corpus，BCC）是以汉语为主、兼有其他语种的在线语料库。语料规模和语料选材是可以全面反映当今社会语言生活的大规模语料库，具有数据量大、领域广和检索便捷等优点。自发布以来，为语言本体研究、语言应用研究、语言教学提供数据和技术支持等功能。其检索系统基于BCC检索式，通过组合字串、词串、词性等属性符号来实现对语料库的查询。

BCC检索表达式由查询对象、限制条件和功能操作三部分组成。语法结构如下：

Query{Condition1;Condition2;…}Operation。

其中Query为查询对象；{}中的内容为限制语句，Condition表示检索内容的条件限定；Operation为功能操作。

查询对象，由汉字串、词性符号以及一些特殊符号组成。词性符号：即词语的词性标签，例如：“n”代表名词，“v”代表动词。一个BCC检索式可以包含多个词性标签。特殊含义符号：通配符、集合符、限定符、离合符、属性约束符号、分隔符等。

限制条件，{}内的内容是限制条件，多个限制条件用“;”间隔，每个Query最多可以包含两处限定条件，被限制部分用()括起来，根据括号出现的顺序，使用$加序号指代。限制条件支持的表达式可以分成内容限制、频次限制、长度限制三个类别。

功能操作，指定检索式检索目的，可以是：实例检索（Context）、频次统计（Freq）或者历时检索（Count）。不指定功能操作时，默认为实例检索。

在进行BCC语料库检索时，通过编写BCC检索式进行语料库查询操作，BCC检索式示例如表1所示。

表1 BCC检索式示例

|  |  |
| --- | --- |
| BCC检索式 | 中文表达 |
| 高大的n | 高大的+名词 |
| v了一v | 动词+了一+动词 |
| 见\*面 | 见后面离合出现面 |
| 洗.澡 | 洗后面隔一个字后接澡 |
| ../v | 二字动词 |
| 爱(v)不(v){$1=$2} | 爱+动词+爱+动词，并且两个动词相同 |
| (nr) 说 m q {len($1)>1; begin($1)=[老]} | 人名加说加数词再加量词，人名长度大于1并且以“老”字开头 |

**2.2 BCC检索日志**

我们从BCC语料库检索日志中精选五万条用户查询实例，形成研究核心数据集，反映用户检索策略与偏好。通过严谨分类，揭示了查询行为的多样性与深度，并统计各分类数量，为数据挖掘与模型训练奠定基础。针对数据稀缺问题，我们采用策略生成技术增强数据，确保全面均衡。同时，设计智能错误检测与纠正机制，提升用户检索体验与系统互动性。最终，成功构建高质量BCC检索式数据集。

2.1.1 BCC检索式分类

我们对BCC检索式进行了分类，依据其构成元素的不同，将检索式划分为六个类别：

1、纯字符串检索式：仅包含待搜索的字符，无特殊符号，体现了最直接的检索需求。

2、词性标签检索式：结合了检索字符与词性标签，体现了用户对词性信息的关注。

3、通配符检索式：包含如“.”、“~”等通配符号，用于模糊匹配，增加检索灵活性。

4、集合符检索式：利用“[]”表示检索单元的取值范围，实现特定集合内的精确检索。

5、属性约束检索式：通过“/”对检索单元施加属性约束，如词语、词性等，增强检索的精确性。

6、限制条件检索式：利用“{}”引入内容、长度、频率等限制条件，对检索结果进行进一步筛选。

值得注意的是，除纯字符串检索式外，其余类别均可能融入词性标签，这反映了用户对于检索深度和精度的多样化需求。

基于上述分类，我们对检索日志数据进行了详尽的统计分析，统计了各分类中不重复检索式的数量及占比，结果如表2所示。数据显示，用户在使用各类BCC检索式时存在显著的不均衡性，特别是属性约束和限制条件检索式的数据量远低于其他类别。

表2 BCC检索式使用分布

|  |  |  |
| --- | --- | --- |
| 类型 | 数量 | 比例 |
| 字符串 | 14767 | 39.17% |
| 词性标签 | 8716 | 23.12% |
| 通配符 | 836 | 2.22% |
| 集合符 | 8065 | 21.39% |
| 属性约束 | 561 | 1.49% |
| 限制条件 | 2918 | 7.74% |

鉴于数据分布的不均衡性，我们采用人工合成数据的方法，有针对性地增强属性约束和限制条件等类别的数据量，以确保模型训练过程中能够获得全面且均衡的数据资源，从而提升模型的性能。

2.1.2 建立纠错数据集

在检索日志中，存在大量用户输入的检索式存在语法错误或符号使用不当的情况。这些错误不仅影响了检索效率，还可能误导用户。为此，针对这些错误检索式进行系统的整理与分类，并尝试进行修复或提供错误提示。

表3 BCC检索式错误示例

|  |  |  |
| --- | --- | --- |
| 类型 | 检索式 | 修复及提示 |
| 函数遗漏 | (v)着{($1)>1} | (v)着{len($1)>1} |
| 符号重复 | 诞辰\*\*\*\*周年 | 诞辰\*周年 |
| 符号缺失 | 没n就v{$1=$2} | 没(n)就(v){$1=$2} |
| 语法错误 | 要是{len(30} | 语法错误，缺少被限定的单元 |
| 语法错误 | 一n一个--人 | 语法错误，存在无效符号“-” |
| 语法错误 | 撞见^^被 | 语法错误，存在无效符号“^” |

对于能够修复的检索式，我们直接给出了修复后的正确形式；对于无法直接修复的，我们则提供了明确的错误提示，引导用户进行修正。最终，我们将这些修复前后的检索式及错误提示整理为BCC检索式纠错数据数据集，为后续的错误检测与自动修正提供数据支持。

**2. 2 BCC检索式生成**

2.2.1 生成例句标注数据集

为了确保训练数据的全面性和代表性，我们从BCC语料库中精心挑选了2万条句子，这些句子的长度介于5至15个汉字之间，并涵盖了文学、报刊和对话等多个领域，从而保证了数据的多样性和广泛性。

BCC语料库采用的是北京大学的词性标签体系[18]。本研究采用了北京语言大学开发的基于网格的自然语言结构分析框架（Grid-based Parsing Framework, GPF）[19]对这些句子进行了分词和词性标注。值得注意的是，GPF同样采用了北京大学的词性标签体系，这确保了所选句子的标注结果与BCC语料库中的词性标注保持一致。我们利用GPF工具[[1]](#footnote-2)对所有的例句进行分词和磁性标注，并将结果保存为标注数据集。

2.2.2 BCC检索式生成策略

为了进一步提升训练数据的丰富性和分类均衡性，我们设计并实施了一套创新的BCC检索式生成规则。具体而言，我们随机选取例句中的两个位置（a和b），并运用多种生成规则来构建多样化的检索式：

1.纯字符串检索式：直接提取位置a和b的词语作为检索词，形成基本的字符串检索式，这类检索式简单直观，适用于直接匹配查询。

2.词性标签检索式：将位置a和b的词语替换为对应的词性标签，提取包含这两个词性的片段作为检索式，此类检索式侧重于词性层面的匹配，有助于捕捉更广泛的语义关系。

3.通配符检索式：在位置a处使用通配符替换具体词语，并结合词性和词语的约束条件，构建出能够匹配多种可能性的检索式，这类检索式在处理模糊查询和不确定信息时尤为有效。

4.集合符检索式：引入集合符号和相似词语作为备选项，构建出具有灵活性和可扩展性的检索式，这类检索式能够匹配多个相似或相关的查询项，提高查询的召回率。

5.属性约束检索式：在保留位置a和b词语的基础上，附加其词性信息，形成带有属性约束的检索式，这类检索式能够更精确地限定查询范围。

6.限制条件检索式：进一步扩展词性标签检索式的功能，通过增加内容、长度、频次等复杂限制条件，构建出高度定制化的检索式，满足特定场景下的复杂查询需求。

2.2.3构建检索式数据集

我们以检索日志数据为基础，应用BCC检索式生成策略，构建数据集。为了确保各类检索式的均衡分布，我们根据实际需求，按照2:2:1:2:1:1的比例对各类检索式进行了合理的分配和补齐，最终形成了一个包含18000条记录的均衡数据集。

数据以JSON格式存储，记录了检索式及其对应的原始例句，既便于数据的快速检索与处理，也保证了数据的可追溯性和可验证性。这一均衡的检索式数据集为后续的模型训练与评估提供了强有力的数据支持。

**3 TextToBCC模型训练**

为了实现自然语言到BCC检索式的转换，需要使用带有自然语言描述的标注数据进行模型训练。为此，我们利用大语言模型来生成BCC检索式的自然语言描述。接下来，将纠错数据与检索式数据组合成微调数据集，对大语言模型进行微调。最终，经过微调模型被部署为服务，从而实现从自然语言查询到BCC检索式的高效转换。整个TextToBCC模型的训练流程如图2所示。

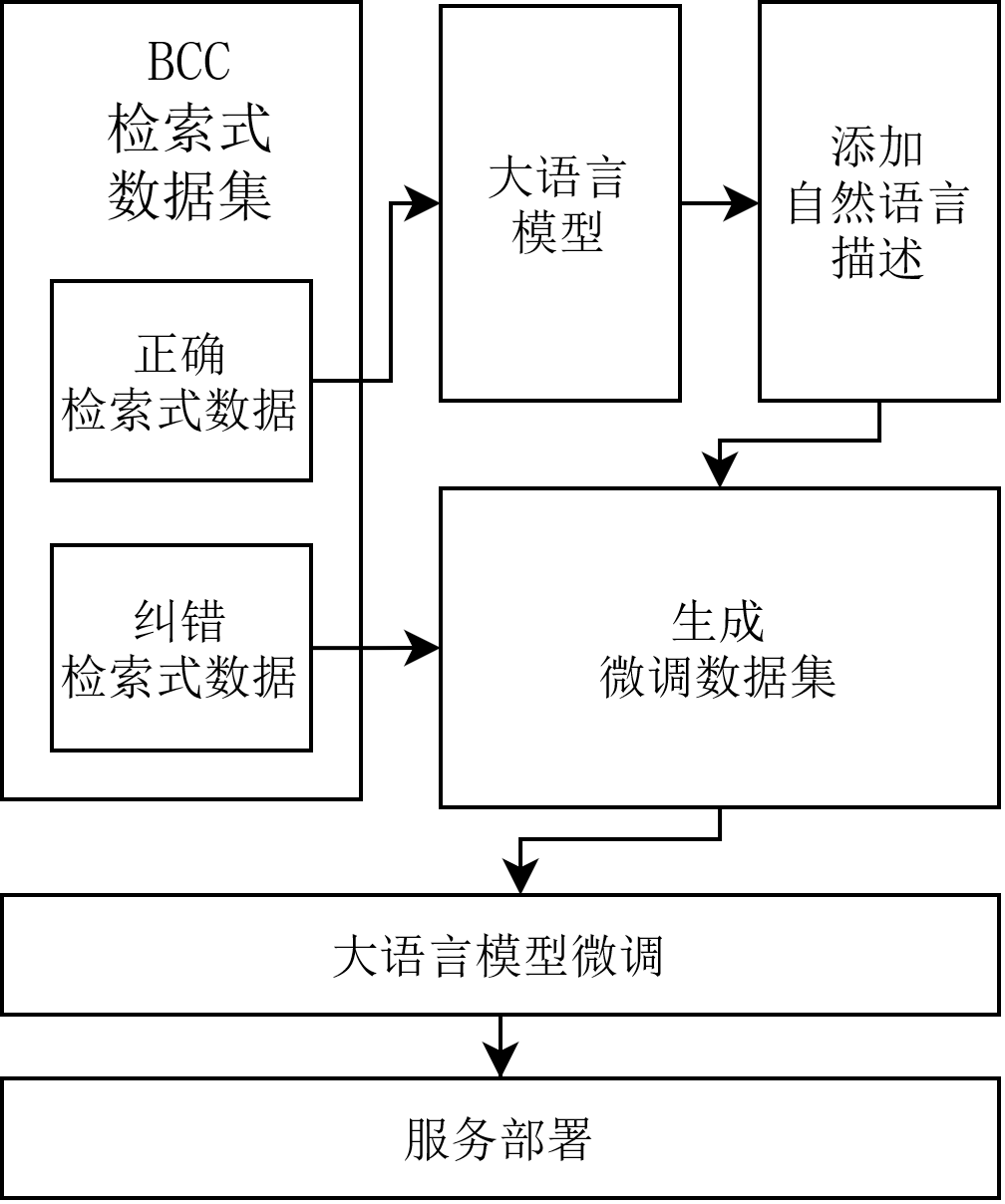


图2 TextToBCC训练流程示意图

**3. 1 生成自然语言描述**

为了优化BCC检索式在自然语言处理中的应用，本文探索了利用在线大型语言模型服务生成其自然语言描述的方法，进而构建了微调数据集。这一流程的核心在于设计提示（prompt），旨在引导大型语言模型精准地生成描述性文本，以匹配BCC检索式。

利用大型语言模型生成BCC检索式的自然语言描述的关键步骤是构建合适的提示。该提示旨在指导模型正确地生成描述文本。具体而言提示内容应包括以下要素：

1、格式与结构解析：首先，清晰定义BCC检索式的格式，包括其结构组成和元素间的逻辑关系，确保提示能够引导模型理解并准确反映其内在结构。

2、特殊符号与规则阐述：详细解释检索式中使用的特殊符号以及使用规则，帮助模型构建必要的上下文理解，从而生成更加准确的自然语言描述。

3、示例引导：提供一系列具体的BCC检索式及其对应的自然语言描述作为示例，为模型生成过程提供直观的学习范本。

4、输出规范：明确指定模型输出内容的JSON格式要求，包括字段名称、数据类型等，确保生成的数据在后续处理中的一致性。

在完成提示编写工作之后，采用通义千问[[2]](#footnote-3)以及ChatGLM[[3]](#footnote-4)在线大语言模型来生成BCC检索式对应的自然语言描述。

为了实现高效的自然语言描述生成，本文采用了一系列优化策略，以确保充分利用大型语言模型的处理能力，同时避免超出其处理范围。具体来说，我们通过API编程接口与大语言模型进行交互，将编写好的提示与待转换的BCC检索式一同发送至模型。这种方法能够充分利用模型的深度学习能力，生成与BCC检索式相匹配的自然语言描述。

考虑到不同模型在处理能力上的差异，本文采取了针对性的策略。对于通义千问模型，我们每次调用时传递40条BCC检索式，而对于ChatGLM模型，则每次调用时传递20条BCC检索式。这种策略能够确保发送和接收的令牌数量在模型的处理范围内，从而避免模型处理能力不足或过载的情况。

在模型处理完毕后，我们提取返回数据中的JSON部分。如果该部分存在且格式合法，我们将返回的数据保存至微调数据集中，以供下一步大模型微调使用。模型生成的自然语言描述示例，如表4所示。

表4 自然语言描述示例

|  |  |  |
| --- | --- | --- |
| 检索式 | 通义千问 | ChatGLM |
| ./d高 ，支持率 ./d | 单音节副词后接高，逗号，支持率，再接单音节副词 | 单音节副词后接高，并且高后面是逗号和空格，接着是支持率，支持率后面再接一个单音节副词 |
| 投票率 ./[d v] 高 | 投票率后接单音节副词或动词，紧接着是高 | 投票率后面隔一个字接一个副词或动词，再接高 |
| ，~ 灿烂 | 句子中以逗号分隔，其后紧跟任何词性的一个词，最后是灿烂 | 逗号后面接一个词，再接灿烂 |
| (v)当代 (vn){len($1)=2; end($2)=[高 妈]} | 长度为2的动词后接当代，再接以高或妈结尾的名动词 | 动词后面接当代，当代后面接名动词，该动词是双音节，并且名动词以高、妈结尾 |
| 正确数量 | 4 | 3 |

观察大模型返回的自然语言描述，通义千问模型在解析BCC检索式方面表现出更自然流畅的语言生成能力，并且准确率相对较高。基于这一优势，本文选择通义千问的输出结果。经过人工审核和必要的修正，最终构建了BCC检索式微调数据集，供大模型微调使用。

**3. 2 生成微调模型**

我们希望微调后的模型需要支持BCC检索式输入以及自然语言输入，并且能够对输入错误的检索式自动纠错，无法自动纠错的需要返回错误提示，因此在构建训练数据集时，需要做以下操作：

1）将BCC检索式数据集中的BCC检索式作为模型输入和输出，即让模型遇到BCC检索式时直接输出BCC检索式。

2）将BCC检索式数据集中的自然语言描述作为模型输入和BCC检索式作为输出，即让模型遇到自然语言查询时转换输出BCC检索式。

3）将BCC检索式纠错数据中的BCC检索式作为输入，纠错后的检索式以及错误提示作为输出，即让模型对错误BCC检索式的自动修复。

在大型语言模型的微调阶段，本文使用ChatGLM3[20]与通义千问[21]两大开源模型。鉴于两模型对训练数据格式的不同要求，我们实施了不同的适配策略。为了明确任务导向，我们在每一条自然语言描述前统一添加了“请将下文解析成BCC检索式：”的指令前缀，旨在引导模型准确理解转换目标。同时，数据集被划分为训练（80%）、验证（10%）、测试（10%）三个子集，以支持模型的评估与优化。

3.2.1 ChatGLM3微调

ChatGLM3模型是由智谱 AI 公司开发的一款大型语言模型，该模型基于GLM130B千亿基础模型，具备强大的自然语言理解和生成能力。

本文采用ChatGLM3-6B模型，为了减少训练参数数量并降低计算成本，本文采用LoRA微调技术进行模型微调。

通过脚本转换，以确保微调数据集完美适配ChatGLM3的数据格式。随后，修改微调配置文件（lora.yaml）设定模型参数、优化器参数及训练参数。在微调过程中，根据实验需求，将最大迭代步数设定为200,000步，并通过命令行执行微调脚本（finetune\_hf.py），启动微调流程。

最终，我们挑选出表现最优的微调权重，将其融入原始模型，以强化模型在自然语言到BCC检索式转换任务中的表现。

3.2.2 通义千问微调

通义千问是阿里云研发的一款超大规模语言模型，它基于Transformer架构，由阿里巴巴自然语言处理实验室开发。是国内最大的中文预训练模型之一。

本文采用Qwen-7B模型进行微调,在数据准备阶段，我们为每条记录分配了唯一的ID，并转换为Qwen要求的格式。利用Qwen官方提供的LoRA微调单GPU脚本，我们仅需调整模型与数据路径参数，并设定最大训练轮数为50轮，即可启动微调过程。

微调结束后，我们将优化后的模型与原始模型进行合并，生成了集强大基础能力与特定任务精细调整于一体的最终模型。这一成果不仅保留了通义千问的广泛语言理解力，还显著提升了其在BCC检索式转换任务上的表现。

**4 实验**

为了验证微调模型的生成效果，我们对Qwen-7B以及ChatGLM3-6B进行了自然语言转换测试、BCC检索式纠错测试以及性能测试。

**4. 1 评价指标**

借鉴单轮文本转SQL的评价方法，本文采用精确匹配率（exact-set-match accuracy, EM）和执行正确率（execution accuracy, EX）作为评价指标[22]。

精确匹配指预测得到的BCC检索式与参考的BCC检索式完全匹配，即整句匹配；执行正确是指执行预测得到的BCC检索式，能够返回正确结果。

**4. 2 检索式转换测试**

分别使用测试集数据，测试微调后的Qwen-7B与ChatGLM3-6B模型转换六类BCC检索式的精确匹配率和执行正确率。

表5 检索式转换结果

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Qwen-7B | | ChatGLM3-6B | |
|  | EM | EX | EM | EX |
| 字符串 | 88.20% | 89.47% | 86.04% | 86.42% |
| 词性标签 | 72.17% | 84.92% | 67.63% | 80.49% |
| 通配符 | 73.94% | 79.34% | 71.83% | 78.40% |
| 限制条件 | 60.96% | 74.07% | 57.25% | 70.68% |
| 集合符 | 65.85% | 72.70% | 63.41% | 69.69% |
| 属性约束 | 56.29% | 66.33% | 55.27% | 66.16% |
| 平均 | 69.61% | 77.74% | 66.85% | 74.96% |

通过对Qwen-7B和ChatGLM3-6B模型进行测试，结果显示Qwen-7B在六种类型的BCC检索式转换上表现更优，平均精确匹配率与平均执行正确率均高出三个百分点左右。这表明Qwen-7B在检索式转换任务上的整体性能优于ChatGLM3-6B。

**4. 3 检索式纠错测试**

在纠错测试方面，分别统计两个错误的修复率。提示正确率是指BCC检索式存在错误，但无法自动修复，模型给出错误提示的情况。纠错正确率是指BCC检索式错误，这些的错误能够自动修正的情况。纠错结果如表6所示。

表6 模型纠错结果

|  |  |  |
| --- | --- | --- |
|  | Qwen-7B | ChatGLM3-6B |
| 提示正确率 | 99.22% | 100% |
| 纠错正确率 | 100% | 99.00% |
| 平均 | 99.56% | 99.56% |

在纠错测试中，Qwen-7B和ChatGLM3-6B都展现了较高的能力来识别和纠正BCC检索式中的错误。两者的平均纠错能力均达到了99.56%的水平。

**4. 4 性能测试**

在性能测试方面，关注自然语言查询转换为BCC检索式的单条请求平均响应时间，具体数值如表7所示。

表7 模型响应时间

|  |  |  |
| --- | --- | --- |
|  | Qwen-7B | ChatGLM3-6B |
| 耗时（秒） | 1800.34 | 3348.32 |
| 样本（条） | 5012 | 5012 |
| 平均（秒） | 0.36 | 0.67 |

从性能测试的结果来看，Qwen-7B单条自然语言查询转换为BCC检索式的平均响应时间为0.36秒，而ChatGLM3-6B的平均响应时间为0.67秒。这意味Qwen-7B在处理速度上存在明显的优势。

**4. 5 结果分析**

在本次实验中，我们针对Qwen-7B和ChatGLM3-6B两个微调模型在自然语言转换至BCC检索式、BCC检索式纠错以及性能方面的表现进行了全面评估。

通过精确匹配率和执行正确率作为衡量标准，我们发现在检索式转换测试中，Qwen-7B的准确率优于ChatGLM3-6B，尤其是在字符串、词性标签和限制条件等类型的检索式转换上表现更佳。

在纠错测试中，两个模型均展现出了极高的错误识别与修正能力,均能有效处理并纠正BCC检索式中的错误。

在性能测试方面，Qwen-7B模型展现出了显著的优势，其平均响应时间远低于ChatGLM3-6B，表明Qwen-7B在处理速度和效率上更为出色。

综上所述，Qwen-7B在检索式转换的准确性和性能上均优于ChatGLM3-6B，尤其是在处理效率和响应时间上表现更为突出。最终，我们选择部署微调后的Qwen-7B实现自然语言描述到BCC检索式的转换服务。

**5 结论**

本文针对BCC检索式非专业用户使用门槛问题，提出了一种基于大语言模型的解决方案。通过构建均衡的BCC检索式数据集，创新性地联合训练BCC检索式与自然语言，实现了将用户自然语言查询自动转换为BCC检索式的功能。本文的主要贡献和结论如下：

1、成功构建了一个全面且均衡的BCC检索式数据集，为模型训练提供了坚实的基础；

2、提出了BCC检索式与自然语言的联合训练方法，模型可以同时支持BCC检索式与自然语言查询，显著提升了模型在BCC检索式转换任务上的性能；

3、通过引入错误BCC检索式的修正数据，实现了模型的纠错功能，大大提高了模型的实用性和准确性。

实验结果表明，本文提出的模型在BCC检索式转换的精确匹配率和执行正确率上具有较高的性能，同时在纠错能力上也表现出色。有效降低了BCC语料库的使用难度，优化了用户体验。

综上所述，本文的研究为BCC语料库的普及和应用提供了新的可能，为语言学研究、自然语言处理等领域的研究者和非专业用户提供了一个高效、易用的检索工具。未来，我们将继续优化模型性能，拓展模型的应用场景，以满足更广泛的研究与实际应用需求。

**参考文献**

1. 荀恩东.自然语言结构计算BCC语料库[M].北京：人民邮电出版社，2023：52-104.
2. 荀恩东, 饶高琦, 肖晓悦, 等. 大数据背景下BCC语料库的研制[J]，语料库语言学，2016(1).
3. 荀恩东, 饶高琦, 谢佳莉,等. 现代汉语词汇历时检索系统的建设与应用[J]. 中文信息学报, 2015, 29(3):169-176.
4. 詹卫东,郭锐,常宝宝,等.北京大学CCL语料库的研制[J].语料库语言学,2019,6(01):71-86+116.
5. 朱君辉, 刘鑫, 杨麟儿, 等. 文心语料库检索平台的研制[C]. 第十二届全国语言文字应用学术研讨会, 2022.
6. 梁茂成, 李文中, 许家金. 语料库应用教程[M]. 北京：外语教学与研究出版社, 2010: 3-25.
7. 程学旗, 靳小龙, 王元卓, 等. 大数据系统和分析技术综述[J]. 软件学报, 2014, 25(9):1889–1908.
8. Alex Boulton. Corpora in language teaching and learning[J]. Language Teaching, 2017, 50(4): 483–506.
9. 奚雪峰, 周国栋. 面向自然语言处理的深度学习研究[J]. 自动化学报, 2016, 42(10):1445-1465.
10. 吴良平. CQP语法赋能语言研究及语言学习[J]. 语料库语言学, 2023, 10:98-114.
11. Yisheng Song, Ting Wang, Subrota K Mondal, et al. A Comprehensive Survey of Few-shot Learning: Evolution, Applications, Challenges, and Opportunities. ArXiv, 2022. arXiv:2205.06743v2
12. Bowen Qin, Binyuan Hui, Lihan Wang, et al. A Survey on Text-to-SQL Parsing: Concepts, Methods, and Future Directions. arViv, 2022, arXiv: 2208.13629v1.
13. Luming Lu, Jiyuan An, Yujie Wang, et al. From Text to CQL: Bridging Natural Language and Corpus Search Engine. ArXiv, 2024. abs/2402.13740
14. Xiang Lisa Li, Percy Liang. Prefix-Tuning: Optimizing Continuous Prompts for Generation[J]. 2021. DOI:10.48550/arXiv.2101.00190.
15. Brian Lester, Rami Al-Rfou, Noah Constant. The Power of Scale for Parameter-Efficient Prompt Tuning[J]. 2021. DOI:10.48550/arXiv.2104.08691.
16. Xiao Liu, Yanan Zheng, Zhengxiao Du, et al. GPT Understands, Too[J]. 2021. DOI:10.48550/ arXiv.2103.10385.
17. Xiao Liu, Kaixuan Ji, Yicheng Fu, et al. P-Tuning v2: Prompt Tuning Can Be Comparable to Fine-tuning Universally Across Scales and Tasks[J]. 2021. DOI:10.48550/arXiv.2110.07602.
18. Edward J. Hu, Yelong Shen, Phillip Wallis,et al. LoRA: Low-Rank Adaptation of Large Language Models[J]. 2021. DOI:10.48550/arXiv.2106.09685.
19. 俞士汶,段慧明,朱学锋,等.北京大学现代汉语语料库基本加工规范[J].中文信息学报, 2002.
20. 荀恩东.自然语言结构计算——GPF结构分析框架[M].北京:人民邮电出版社.2022
21. Team GLM, Aohan Zeng, Bin Xu ,et al. ChatGLM: A Family of Large Language Models from GLM-130B to GLM-4 All Tools. arXiv, 2024. arXiv: 2406.12793.
22. Jinze Bai, Shuai Bai, Yunfei Chu, et al. Qwen Technical Report. arXiv, 2023, arXiv:2309.16609.
23. Dawei Gao, Haibin Wang, Yaliang Li, et al. Text-to-SQL Empowered by Large Language Models: A Benchmark Evaluation. arXiv, 2023. arXiv:2308. 15363v4



第一作者姓名（出生年—），学历，职称，主要研究领域为。

E-mail：\*\*\*@\*\*\*



第二作者姓名（出生年—），学历，职称，主要研究领域为。

E-mail：\*\*\*@\*\*\*



第三作者姓名（出生年—），学历，职称，主要研究领域为。

E-mail：\*\*\*@\*\*\*

1. https://pypi.org/project/gpflib [↑](#footnote-ref-2)
2. <https://tongyi.aliyun.com/qianwen> [↑](#footnote-ref-3)
3. <https://chatglm.cn> [↑](#footnote-ref-4)