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

刘廷超[1] 鲁鹿鸣[1] 荀恩东[2] 靳泽莹[1] 杨兆勇[1]

（1. 北京语言大学 信息科学学院，北京市 100083；
2. 北京语言大学 语言资源高精尖创新中心，北京市 100083）

摘要：语料库在语言学和自然语言处理领域至关重要。北京语言大学的 BCC 语料库，资源丰富且检索高效，备受推崇，然而，其 BCC 检索式的复杂性限制了普及。为此，本文提出 TextToBCC 模型，目标是实现自然语言对 BCC 语料库的检索。首先构建了一个均衡的 BCC 检索式数据集，利用大语言模型为 BCC 检索式生成了自然语言描述。随后，微调大语言模型进使其能够支持自然语言到 BCC 检索式的转换。实验结果证明了 TextToBCC 模型的优异性能。这一成果不仅降低了 BCC 语料库的使用难度，而且有助于促进其在更广泛领域的传播和应用，为语言学研究和自然语言处理实践带来便利。

关键词：语料库；检索式；大语言模型；微调

中图分类号：TP391 **文献标识码：**A

Natural Language Retrieval of BCC Corpus Based on Large Language Models

Tingchao Liu[1], Luming Lu[1], Endong Xun[2], Zeying Jin[1], Zhaoyong Yang[1]

(1. School of Information Science, Beijing Language and Culture University, Beijing, 100083, China;
2. Advanced Research Center of Language Resources, Beijing Language and Culture University, Beijing, 100083, China)

Abstract: Corpora are crucial in the fields of linguistics and natural language processing. The BCC corpus of Beijing Language and Culture University is highly respected for its rich resources and efficient retrieval. However, the complexity of its BCC search query limits its popularity. To this end, this paper proposes the TextToBCC model, which aims to achieve natural language retrieval of the BCC corpus. First, a balanced BCC search queries dataset was constructed, and a natural language description was generated for the BCC search queries using a large language model. Subsequently, the large language model was fine-tuned to enable it to support the conversion from natural language to BCC search queries. Experimental results demonstrate the excellent performance of the TextToBCC model. This achievement reduces the difficulty of using the BCC corpus, and helps promote its dissemination and application in a wider range of fields, bringing convenience to linguistic research and natural language processing practice.

Key words: corpus; search query; large language models; fine-tuning

0 引言

在信息时代的浪潮下，语言数据的飞速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，例如，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

收稿收稿日期：20XX- XX-XX；**定稿日期**：20XX- XX-XX
基金项目：基金名（基金号）；基金名(基金号)

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。在使用语料库进行语料检索时，用户必须掌握相应的检索式规则和使用方法。例如，在 BCC 语料库中查找以“跑”为首的双音节动词时，需编写检索式“跑./v”。而要查找结构为“名词+山+名词+海”的词组，并且要求两个名词相同，检索式则为“(n)山(n)海{$1=$2}”。掌握这些检索式的语法规则需要投入一定的学习成本，这无疑增加了用户在使用语料库时的负担。

在自然语言处理领域，自然语言查询正逐渐成为一种重要的查询方式，并获得了广泛的关注。用户能够以日常语言表达查询需求，显著降低了使用门槛。近年来，文本到 SQL 的转换任务取得了显著进展，可以将自然语言查询转换为 SQL 语句，实现对数据库的高效检索。然而，由于语料库没有统一的检索语言，自然语言查询到语料库检索式的转换研究进展十分有限。

鉴于此，本文致力于探究自然语言到语料库检索式转换的研究任务。构造一个大语言模型，以实现从自然语言到语料库检索式的转换，降低语料库使用的难度。为实现这一目标，本文选取 BCC 语料库作为研究对象，研发了 TextToBCC 模型[1]，为 BCC 语料库引入自然语言查询功能。

首先，构建了一个庞大的数据集，其中包含了从 BCC 语料库查询日志中筛选出的用户查询实例。通过对这些数据的深入分析，揭示了用户查询行为的多样性和复杂性，并针对数据稀缺的挑战，运用检索式生成技术进行数据增强。

接着，设计了一种基于大语言模型的检索式到自然语言的转换方法，通过编写提示语（Prompt），实现了将 BCC 检索式转化为自然语言的功能。在模型训练阶段，采用了微调现有大语言模型的策略，将 BCC 检索式数据集转换为适合特定模型的微调数据集，以适应自然语言查询到 BCC 检索式的转换任务。为了进一步优化模型的性能，还将错误日志中的错误 BCC 检索式及其对应的纠错信息纳入到大语言模型的微调过程中。这一策略使得大语言模型具备了自动纠正 BCC 检索式错误的能力。

最后，经过一系列实验验证，本文提出的 TextToBCC 模型在自然语言到 BCC 检索式转换任务上展现出了卓越的性能。

本文不仅为语料库的普及和应用开辟了新的途径，同时也为语言学研究、自然语言处理等领域的用户提供了一个高效、便捷的检索工具。通过降低语料库的使用门槛和提高检索效率，促进相关领域的研究和应用实现更大的突破。

1 研究背景

随着信息技术的发展，语料库的规模持续扩大，其重要价值也随之日益凸显。但语料库检索式的多样性在一定程度上限制了它们的普及和应用。因此，人们渴望借助自然语言查询技术来简化语料库的使用流程。随着大语言模型技术的不断进步，这一愿望正在逐步成为现实。

1.1 语料库

语料库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和开发智能应用不可或缺的基础[9]。

当前，主流中文语料库在资源和功能方面各具特色，但无一例外地均采用检索式作为信息检索的核心手段。语料库检索式作为连接用户与所需信息的桥梁，发挥着至关重要的作用。然而，熟练掌握语料库检索式并非易事，它要求用户具备一定的语言学素养，包括但不限于检索式的语法构造、词性标注规则以及特殊符号的恰当运用等方面的知识。这对于非专业背景的用户来说，无疑是一项不小的挑战。

语料库的检索方法存在明显差异。CCL 语料库和 BCC 语料库都使用了自有的检索式体系，它们在语法规则和使用方法上有较大不同。DCC 语料库则采用了基于语料库查询语言（Corpus Query

[1] https://github.com/paineliu/TextToBCC

Language，CQL）扩展的检索语言[10]，许多其他语料库也采用 CQL 或其扩展作为查询语言。尽管语料库检索式的多样性体现了语料库功能的强大，但也暴露了语料库检索语言非标准化的问题。对用户来说，意味着在使用不同的数据库时，需要投入时间和精力去学习每个语料库特有的检索语法和使用规则。

1.2 自然语言查询

自然语言查询（Natural Language Query，NLQ）是基于自然语言表述的一种查询方式，属于自然语言处理领域的关键任务之一。它允许用户以日常使用的自然语言形式进行输入，从而使计算机能够识别、处理并深入加工这些信息。这一过程不仅涉及对自然语言的理解，还能够根据所接收的信息做出相应的反馈。如此一来，用户得以借助直观易懂的自然语言与计算机进行交互，实现高效的信息检索与查询。

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。

在自然语言实现语料库查询的应用场景中，文献[12] 提出了一种将自然语言转换为 CQL 的策略。为此，研究者构建了一个专门的 CQL 数据集，并借助大语言模型的强大能力，将文本信息精准地转化为 CQL 语句。此外，为了衡量转换效果，他们还制定了一套全面的评估指标，用以检验生成的 CQL 语句的准确性。

1.3 大语言模型及微调

大语言模型(Large Language Models, LLMs)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[13]。

随着计算能力的飞速提升以及大规模数据集的涌现，LLMs 在语言理解和语言生成等多个领域均取得了卓越的进展。LLMs 的成功关键在于其庞大的规模，这使得它们能够通过庞大的参数空间捕获更为丰富的语言特征和上下文信息。这些模型首先在大量未标注的文本上进行预训练，以此学习到语言的通用表示形式；接着，通过针对特定任务的微调过程，它们能够迅速适应各种不同的应用场景。值得一提的是，LLMs 还展现出了零样本或少样本学习的能力[14]，这意味着可以在几乎不需要标注数据或仅需少量标注数据的情况下完成新的任务，从而极大地降低了自然语言处理应用的难度和门槛。

模型微调（Fine-tuning）作为 LLMs 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 LLMs 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[15]、Prompt Tuning[16]、P-Tuning[17]、P-Tuning v2[18]等方法，其中 LoRA（Low-Rank Adaptation）微调[19]尤为引人注目。

LoRA 微调技术通过在模型中嵌入可训练的低秩矩阵，巧妙地降低了微调过程中所需的参数数量。这一创新举措不仅确保了模型性能的持续提升，还显著减轻了计算负担和显存占用。LoRA 微调不仅优化了微调的效率和稳定性，还拓宽了 LLMs 在资源受限环境中的应用可能性。目前，众多开源 LLMs 平台已相继推出了 LoRA 微调的示例代码，为学术界和产业界的从业者提供了便捷的实施工具。

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，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，对这两类日志数据进行分类整理，随后依据预设比例从各分类中提取样本。为了确保数据集的均衡性，还设计了检索式生成策略，用以有效补充那些数据量不足的分类。最终，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。

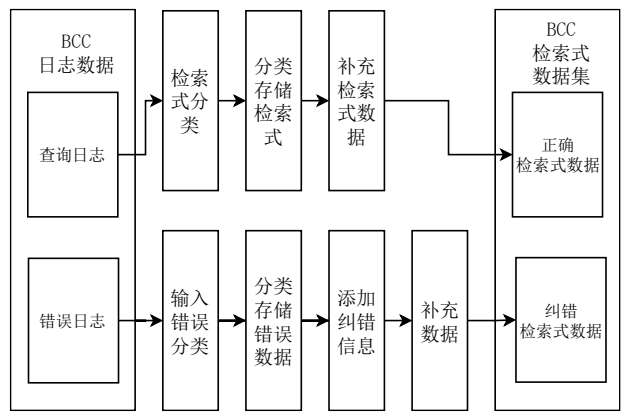

图 1 BCC 检索式数据集生成流程图

2.1 BCC 语料库

BCC 语料库是一个以汉语为主体，同时涵盖多种其他语言的在线资源库。凭借其庞大的体量和多元化的选材范围，该语料库详尽地展现了当代社会的语言生态，凸显了数据量大、领域覆盖全面以及检索高效等突出优势。自推出以来，BCC 语料库已在语言本体研究、应用研究和教学实践等多个领域提供了强有力的数据支撑和技术保障。

BCC 语料库采用检索式来进行高效的语料查询，检索式是一种巧妙组合字符、词串及词性等属性符号以实现精确检索的方法。其结构主要由查询对象、限制条件和功能操作三大部分组成，遵循如下语法规则：Query{Condition1; Condition2; …} Operation。在此结构中，Query 表示待检索的内容；花括号内的部分代表限制条件，用于细化检索范围；而 Operation 则定义了具体的检索功能。

查询对象由汉字串、词性符号及特殊符号共同构成。汉字串用于指定具体的文本内容，词性符号如“n”（名词）和“v”（动词）则为词语打上明确的标签。BCC 检索式支持同时包含多个词性标签，以满足复杂的检索需求。特殊含义符号则用于对检索内容进行抽象或精确限定，涵盖了通配符、集合符、限定符等多种类型。

限制条件通过运用限制函数来实现对检索内容的精细化控制。多个限制条件之间以分号“;”隔开，每个 Query 最多可设置两个限定条件，且需用括号括起来，并通过$加序号的方式进行引用。限制条件支持的函数类型包括内容限制、频次限制及长度限制。

功能操作用于明确检索式的具体目的，如实例检索（Context）、频次统计（Freq）或历时检索（Count）。若未明确指定功能操作，则默认执行实例检索。

在使用 BCC 语料库时，用户需依据上述规则编写 BCC 检索式以执行查询。具体的 BCC 检索式示例可参见表 1。

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

本文从 BCC 语料库的检索日志中筛选出五万条用户查询实例，以此为基础构建了旨在反映用户检索策略与偏好的研究数据集。通过深入细致的数据分析，将检索式细分为六个类别，并对各分类数据的出现频次进行了详尽统计，这一过程为后续的数据挖掘与模型训练提供了强有力的支撑。

为克服数据稀缺性的难题，采用数据生成技术进行了有效的数据增强，从而确保了数据集的完整性及均衡性。此外，还从检索错误日志中提取了一万条数据，并在此基础上创新性地设计了错误检测与纠正机制，以期显著提升用户的检索体验及系统的交互性能。

经过这一系列精心策划与实施的工作，成功打造出了一个高质量的 BCC 检索式数据集。

2.1.1 BCC 检索式分类

本文对 BCC 检索式进行了分类，依据其组成要素的差异，将其细分为六个类别：

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，除纯字符串检索式之外，其他各类别均可能包含词性标签，这一现象揭示了用户在检索过程中对词性标签的高度依赖。

基于上述分类体系，对检索日志数据展开了深入的统计分析，详细考察了各类别中去重后检索式的数量及其所占比例，具体结果详见表 2 所示。

表 2 BCC 检索式使用分布

类型	数量	比例
字符串	14767	41.18%
词性标签	8716	24.30%
通配符	836	2.33%
集合符	8065	22.49%
属性约束	561	1.56%
限制条件	2918	8.14%

数据分析清晰地展示了一个显著的趋势：用户在使用 BCC 检索式的各个类别时存在显著的差异，特别是属性约束和限制条件检索式的数据量相较于其他类别明显不足。为了应对这种数据分布的不平衡现象，采取了人工合成数据的策略，专门针对属性约束和限制条件等数据量较少的类别进行了有针对性的扩充。这一举措确保了模型在训练时能够接触到全面且均衡的数据资源，从而有效提高了模型的综合性能。

2.1.2 建立纠错数据集

在检索错误日志中，发现用户输入的检索式常出现语法错误或符号使用不当的情况。这些错误不仅降低了检索效率，还可能导致用户进行不准确的查询。为应对这一问题，系统地对这些错误的检索式进行了整理与分类，并尝试对其进行修正或提供相应的错误提示。

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“^”

从数据中精心抽取了 2000 条样本进行人工标注，并对其中的错误进行修正。对于那些能够被修复的检索式，直接给出了修正后的正确版本。而对于那些暂时无法修复的检索式，提供了明确的错误提示，旨在引导用户进行适当的修改。最终，将所有修复后的检索式、相应的错误提示以及原始的错误检索式一并整理存储，从而构建了一个宝贵的 BCC 检索式纠错数据集，为未来的错误自动修正工作奠定了坚实基础。

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，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[20]。本文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）[21]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，这确保了句子的标注结果与 BCC 语料库中的词性标注保持一致。本文使用 GPF 工具[2]对所有的例句进行分词和词性标注，并将结果保存为句子标注数据集。

2.2.2 BCC 检索式生成策略

为了增强训练数据的丰富度和确保分类的均衡性，设计了一套 BCC 检索式生成规则。具体操作如下，随机从例句中选取两个位置（标记为 a 和 b），然后应用多种生成规则来构造多样化的检索式：

1）纯字符串检索式：直接采用位置 a 和 b 处的词语作为检索式，生成基础的字符串检索式。

2）词性标签检索式：将位置 a 和 b 的词语替换为词性标签，生成带有词性信息的检索式。

3）通配符检索式：在位置 a 处用通配符替代具体词语，生成包含通配符的检索式。

4）集合符检索式：引入具有相似词性和相似词语的备选项，生成带有集合限制的检索式。

5）属性约束检索式：为 a 和 b 位置的词语添加词性信息，生成带有属性限制的检索式。

6）限制条件检索式：将位置 a 和 b 替换为词性标签，并增加内容、长度、频次等限制，生成带有限制性条件的检索式。

[2] https://pypi.org/project/gpflib

2.2.3 构建检索式数据集

本文以检索日志数据为基础，采用 BCC 检索式生成策略构建数据集。为确保各类检索式在数据集中达到均衡分布，依照 2:2:1:2:1:1 的比例对它们进行了分配和补充，最终构造出一个包含 18000 条记录的检索式数据集。

这些数据以 JSON 格式进行存储，记录了每个检索式及其对应的原始例句。这种存储方式不仅便于数据的快速检索与处理，而且确保了数据的可追溯性和可验证性。这个均衡的检索式数据集将为后续的模型训练与评估提供数据支撑。

3 TextToBCC 模型训练

为实现自然语言向 BCC 检索式的有效转换，本文提出了 TextToBCC 模型，该模型通过对开源大语言模型微调而构建。在 TextToBCC 模型的训练过程中，需利用带有自然语言描述的 BCC 检索式数据。为此，首先借助大语言模型为 BCC 检索式数据生成自然语言描述。紧接着，将经过纠错的检索式数据与添加了自然语言描述的 BCC 检索式数据进行整合，形成用于微调的数据集。随后，利用该数据集对大语言模型进行针对性的微调。最终，将完成微调的 TextToBCC 模型部署为在线服务，从而实现了从自然语言到 BCC 检索式的转换。TextToBCC 模型的完整训练流程如图 2 所示。

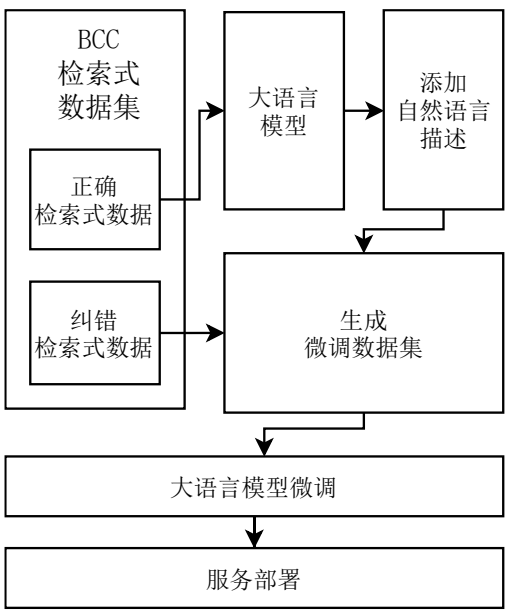

图 2 TextToBCC 训练流程示意图

3.1 生成自然语言描述

为了获取 BCC 检索式的自然语言描述，本文探索了利用在线大型语言模型服务生成这些描述的有效方法。此过程的核心在于设计提示语，以引导大型语言模型准确生成 BCC 检索式的自然语言描述。提示语设计需涵盖以下关键要素：

1）格式与结构解析：明确定义 BCC 检索式的格式，阐明其结构组成和元素间的逻辑关系，确保提示能够帮助模型理解并准确反映检索式的语法结构。

2）特殊符号与规则阐述：详尽阐释检索式中使用的特殊符号及其应用规则，协助模型构建必要的上下文理解，以便生成更为精确的自然语言描述。

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参考示例，为模型生成过程提供直观的学习样本。

4）输出规范：明确规定模型输出内容的 JSON 格式要求，包括字段名称、数据类型等，以确保生成的数据在后续处理中保持一致性。

完成提示设计后，采用通义千问[3]以及 ChatGLM[4]这两款在线大语言模型来生成 BCC 检索式对应的自然语言描述。

本文通过大型语言模型的 API 编程接口与其展开交互，将提示语及待转换的 BCC 检索式一并传输至大型语言模型。在成功接收模型返回的数据后，提取其中的 JSON 部分，并将其保存至微调数据集中。大型语言模型生成的自然语言描述示例，如表 4 所展示。

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 ./[dv] 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	逗号后面接一个词，再接灿

[3] https://tongyi.aliyun.com/qianwen

[4] https://chatglm.cn

	跟任何词性的一个词，最后是灿烂	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

在对比分析通义千问和 ChatGLM 两大语言模型生成的自然语言描述后，观察到通义千问在描述 BCC 检索式时表现出了更为自然流畅的语言表达能力，并且其描述的准确性也相对较高。基于这一发现，本文决定采纳通义千问生成的描述结果。随后，经过人工审核和必要的修正，构建了一个 BCC 检索式自然语言数据集。

3.2 生成 TextToBCC 模型

本文旨在利用开源大型语言模型实现自然语言到 BCC 检索式的有效转换。然而，未经微调的模型无法直接完成这一任务，因此开发了 TextToBCC 模型，通过对大语言模型微调来实现目标。TextToBCC 模型需兼容 BCC 检索式与自然语言输入，并具备自动纠正错误检索式的能力，若无法自动纠正，则返回适当的错误提示。为构建微调数据集，采取以下步骤：

1）将 BCC 检索式自然语言数据集中的 BCC 检索式同时作为模型的输入和期望输出，目的是让模型在接收 BCC 检索式时能够直接输出相同的 BCC 检索式。

2）将 BCC 检索式自然语言数据集中的自然语言描述作为输入，而将对应的 BCC 检索式作为期望输出，这样模型在面对自然语言输入时能够转换为相应的 BCC 检索式。

3）将 BCC 检索式纠错数据集的中的错误检索式作为输入数据，将纠错后的检索式及错误提示作为输出，旨在训练模型识别并修正错误的 BCC 检索式。

在微调阶段，微调了 ChatGLM3[22]与通义千问[23]两个开源模型。为了强化任务指向性，在每条数据前统一添加了指令前缀“请将下文解析成 BCC 检索式：”，以帮助模型准确理解意图。

微调数据集由 BCC 检索式纠错数据集和 BCC 检索式自然语言数据集共同组成，总计包含 2 万条样本。将这两部分数据整合，形成统一的微调数据集。该微调数据集进一步被细分为训练集（占 80%）、验证集（占 10%）和测试集（占 10%）三个子集。样本数据在各子集之间随机分配，确保每个子集均无重复数据。

3.2.1 ChatGLM3 微调

ChatGLM 模型是由智谱 AI 公司开发的一款高性能大语言模型，它基于 GLM130B 千亿基础模型，拥有卓越的自然语言理解和生成能力。

在本文中，采用了 ChatGLM3-6B 模型，为了减小训练参数的数量和降低计算成本，运用了 LoRA 微调技术来进行模型的微调。首先，通过脚本程序对数据格式进行了转换，以确保训练数据集符合 ChatGLM3 的格式要求。接着，调整了微调配置文件（lora.yaml），设定了模型参数、优化器参数以及训练参数，其中最大迭代步数被设置为 20 万步。最后，通过命令行运行微调脚本（finetune_hf.py），启动了微调过程。微调结束后，选取表现最佳的微调权重并整合至原始模型，从而提升模型在自然语言到 BCC 检索式转换任务上的表现。

3.2.2 通义千问微调

通义千问是由阿里云研发的一款超大规模语言模型，基于 Transformer 架构，由阿里巴巴自然语言处理实验室倾力打造，是国内规模领先的中文预训练模型之一。

本文选用 Qwen-7B 模型进行微调。首先，在数据准备阶段，依照 Qwen 微调数据的格式要求，为每条记录指定了唯一的 ID，并将其转换为 Qwen 所需的格式。接着，采用了 Qwen 官方提供的 LoRA 微调单 GPU 脚本，对模型和数据路径参数进行了调整，并将最大训练轮数设置为 50 轮。最后，通过运行脚本启动了微调流程。微调完成后，将优化后的模型与原始模型合并，形成了一个集强大基础能力与针对 BCC 检索式转换任务调整于一体的模型。

4 实验

为了检验微调模型的自然语言转换 BCC 检索式的效果，对 ChatGLM3-6B 微调模型和 Qwen-7B 微调模型进行了一系列测试，包括自然语言转换测试、检索式纠错测试以及性能测试。

4.1 评价指标

受单轮文本转 SQL 评估方法的启发，本文采用了精确匹配率（exact-set-match accuracy，EM）和执行正确率（execution accuracy，EX）作为评价指标[24]。

精确匹配率是指预测生成的 BCC 检索式与标准 BCC 检索式在字符串层面上完全一致的比率；而执行正确率则是指使用预测得到的 BCC 检索式执行查询能够得到正确结果的概率。

4.2 检索式转换测试

本文使用测试集数据对微调后的 ChatGLM3-6B 和 Qwen-7B 模型在六类 BCC 检索式转换任务上的精确匹配率（EM）和执行正确率（EX）进行了全面评估。

精确匹配率的计算方法为：微调模型输出的 BCC 检索式与预期标准 BCC 检索式结果字符串完全一致的情况占全部数据的比例。执行正确率的计算方法则是：在 BCC 语料库上执行微调模型生成的 BCC 检索式所得到的候选结果与预期 BCC 检索式执行结果一致的比率。平均值的计算方式是将六类比率直接取均值，不考虑类别内样本数量的差异。

测试结果表明，Qwen-7B 微调模型在六种类型的 BCC 检索式转换任务上均展现出更优越的性能，其平均精确匹配率和平均执行正确率均比 ChatGLM3-6B 微调模型高出两个百分点。这一结果清晰地表明，在 BCC 检索式转换任务的整体正确率方面，Qwen-7B 微调模型优于 ChatGLM3-6B 微调模型。各分类的转换结果详见表 5。

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	69.57%	77.81%	66.91%	75.31%

4.3 检索式纠错测试

在纠错测试环节，分别统计了两种错误修复的情况。提示正确率是指当 BCC 检索式存在错误且无法自动修复时，模型能够提供正确提示的比例。纠错正确率则是指在 BCC 检索式存在可自动修正的错误时，模型成功修正这些错误的比例。平均正确率的计算方法是忽略两类样本的数量差异，直接取提示正确率和纠错正确率的平均值。检索式纠错结果详见表 6。

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	99.61%	99.50%

在纠错测试中，无论是 Qwen-7B 微调模型还是 ChatGLM3-6B 微调模型，都显示出了较高的错误识别和纠正能力。这两个模型的平均纠错能力均达到了 99% 以上。这一结果表明纠错数据集对用户在使用 BCC 检索式时可能遇到的错误进行了全面覆盖。

4.4 性能测试

在性能测试方面，评估了将自然语言转换为 BCC 检索式的单条查询平均响应时间，具体数据详见表 7。

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

性能测试结果显示，Qwen-7B 微调模型处理单条查询的平均响应时间为 0.36 秒，相较之下，ChatGLM3-6B 微调模型的平均响应时间为 0.67 秒。在处理速度方面，Qwen-7B 微调模型表现出明显的优势。这一差异与模型的架构及测试时使用的硬件环境密切相关。

4.5 结果分析

在本实验中，对 ChatGLM3-6B 和 Qwen-7B 两个微调模型在检索式转换、检索式纠错以及性能表现方面进行了全面的评估。

在检索式转换测试中，Qwen-7B 微调模型在准确性上超越了 ChatGLM3-6B 微调模型，特别是在字符串、词性标签和限制条件等类型的检索式转换任务上表现得尤为出色。

在纠错能力测试中，两个模型均显示出了极高的错误识别和修正能力，能够有效地处理并纠正 BCC 检索式中的错误。

在性能评估方面，Qwen-7B 微调模型具有明显的优势，其平均响应时间显著低于 ChatGLM3-6B 微调模型，这表明 Qwen-7B 微调模型在处理速度方面更为出色。

综合以上分析，Qwen-7B 微调模型在检索式转换的准确率和性能上均优于 ChatGLM3-6B 微调模型。因此，最终部署 Qwen-7B 微调模型作为 TextToBCC 模型，以提供自然语言到 BCC 检索式的转换服务。

5 结论

本文设计了 TextToBCC 模型，为 BCC 语料库增添了自然语言查询的功能。通过构建 BCC 检索式微调数据集，并对大语言模型进行精细微调，顺利实现了自然语言向 BCC 检索式的转换。主要贡献与结论如下：

1）构建了一个全面而均衡的 BCC 检索式数据集，为模型的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，使得 TextToBCC 模型能够同时支持检索式和自然语言查询，显著提升用户体验。

3）通过引入纠错数据，使模型具备了 BCC 检索式的纠错功能，提高了模型的实用性。

实验结果有力证明了 TextToBCC 模型的高性能表现，尤其在纠错能力方面表现突出。有效降低了 BCC 语料库的使用门槛，并显著优化了用户体验。此外，本文提出的方法同样适用于其他自然语言到检索式的转换任务，展现出广泛的适用性。

综上所述，本文为 BCC 语料库的推广与应用开辟了新的途径，为用户提供了一种高效、便捷的语料库检索工具。

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] 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.

[12] Luming Lu, Jiyuan An, Yujie Wang, et al. From Text to CQL: Bridging Natural Language and Corpus Search Engine. ArXiv, 2024. abs/2402.13740

[13] Shervin Minaee, Tomas Mikolov, Narjes Nikzad, et al. Large Language Models: A Survey. arViv, 2024, arXiv: 2402.06196v2.

[14] 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.

[15] Xiang Lisa Li, Percy Liang. Prefix-Tuning: Optimizing Continuous Prompts for Generation[J]. 2021. DOI:10.48550/arXiv.2101.00190.

[16] Brian Lester, Rami Al-Rfou, Noah Constant. The Power of Scale for Parameter-Efficient Prompt Tuning[J]. 2021. DOI:10.48550/arXiv.2104.08691.

[17] Xiao Liu, Yanan Zheng, Zhengxiao Du, et al. GPT Understands, Too[J]. 2021. DOI:10.48550/arXiv.2103.10385.

[18] 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.

[19] 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.

[20] 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.

[21] 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

[22] 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.

[23] Jinze Bai, Shuai Bai, Yunfei Chu, et al. Qwen Technical Report. arXiv, 2023, arXiv:2309.16609.
[24] Dawei Gao, Haibin Wang, Yaliang Li, et al. Text-to-SQL Empowered by Large Language Models: A Benchmark Evaluation. arXiv, 2023. arXiv:2308.15363v4

刘廷超（1978—），博士研究生，主要研究领域为信息检索、自然语言处理。
E-mail：liutingchao@hotmail.com

鲁鹿鸣（1997—），硕士研究生，主要研究领域为自然语言处理，语料库技术。
E-mail：llm410402@gmail.com

荀恩东（1967—），博士，教授，主要研究领域为自然语言处理、基于汉语大数据语言知识抽取、汉语句法语义分析、语言资源建设。
E-mail：edxun@blcu.edu.cn