涂文奇, 李柏岩, 刘晓强, 等. NL2SQL 融合知识图谱在设备运维数据检索中的应用[J]. 智能计算机与应用,2024,14(9):118-124. DOI:10.20169/j.issn.2095-2163.240918

NL2SQL 融合知识图谱在设备运维数据检索中的应用

涂文奇, 李柏岩, 刘晓强, 郑佳明
(东华大学 计算机科学与技术学院, 上海 201620)

摘 要: 设备运维现场需要进行设备信息和运行状况的信息检索,但功能固定的查询程序无法满足多变的信息获取需求,而使用自然语言的交互检索可以提供更友好的支持。本文以行车设备运维的数据检索为目标,基于 M-SQL 模型设计并实现了自然语言到结构化查询语句(NL2SQL)的生成和自动检索,并融合知识图谱提高 SQL 语句中字段表达的准确性。首先基于开源通用领域数据集训练得到基于子任务的 NL2SQL 多任务学习模型;然后在标注的行车领域查询数据集上进行微调,使模型理解行车领域词汇;最后利用行车数据库知识图谱实体链接修正生成的 SQL 查询语句中数据不规范的问题。实验结果表明,该系统方案可以显著提高模型在行车数据的查询效率和准确性,满足使用自然语言的人机交互查询需求。

关键词: 行车运维; NL2SQL 模型; 知识图谱; 信息检索
中图分类号: TP391.1 **文献标志码**: A **文章编号**: 2095-2163(2024)09-0118-07

Application of NL2SQL with knowledge graph fusion in equipment maintenance data retrieval

TU Wenqi, LI Baiyan, LIU Xiaoqiang, ZHENG Jiaming
(College of Computer Science and Technology, Donghua University, Shanghai 201620, China)

Abstract: Information retrieval of equipment information and operation status is needed in the equipment operation and maintenance site, but the query program with fixed function can not meet the changeable information acquisition demand, and interactive retrieval using natural language can provide more friendly support. Aiming at the data retrieval of crane equipment operation and maintenance, based on M-SQL model, the generation and automatic retrieval from natural language to structured query statement (NL2SQL) are designed and realized, and knowledge map is integrated to improve the accuracy of field expression in SQL statement. Firstly, NL2SQL multi-task learning model based on subtasks was obtained by training based on open source universal domain data set, and then fine-tuned on the marked query data set of crane domain to make the model understand the vocabulary of crane domain. Finally, the problem of nonstandard data in the generated SQL query statement was corrected by using the entity link of knowledge map of crane database. The experimental application shows that the system scheme can significantly improve the efficiency and accuracy of the model in the query of crane data, and meet the requirements of human-computer interactive query using natural language.

Key words: equipment maintenance; NL2SQL model; knowledge graph; information retrieval

0 引 言

制造型企业管理着庞大的生产设备群,并且会不断增加新设备,设备随着运维也会新增信息。由于运维工作通常在设备现场进行,随时会提出不同的信息获取需求,这些需求往往具有突发性和不确定性。功能固定的查询程序无法满足多变的信息获取需求,使用自然语言作为查询输入成为一种更灵活、友好的交互检索方式。

行车又称起重机,是制造型企业最常用的一种起重设备,主要作用是完成重物的移动。在工业生产中往往会使用到多种类型行车,其组成包括了挠性构件与卷绕装置、取物装置、制动装置等,每个装备零件都是保证行车正常运转的重要部分[1]。工厂的行车数

作者简介: 涂文奇(1997-),男,硕士研究生,主要研究方向:自然语言处理,NL2SQL; 刘晓强(1968-),女,博士,教授,硕士生导师,主要研究方向:智能信息处理,知识管理; 郑佳明(1998-),男,硕士研究生,主要研究方向:自然语言处理,知识图谱。
通讯作者: 李柏岩(1968-),男,博士,副教授,硕士生导师,主要研究方向:机器学习,信号处理。Email:libaiyan@dhu.edu.cn

收稿日期: 2023-05-07

据，包含了行车的各种属性、状态以及历史运维记录等，大部分以结构化形式存储在表文件中，及时获取行车数据有助于帮助操作人员或者检修人员更好地监测行车的运行状态，发现潜在的故障危险[2]，这对于企业的运营管理和设备维护至关重要。

传统的查询方式通常使用功能固定的查询工具，虽然简单但缺乏灵活性，查询内容扩展能力有限，不适应现场应用需求。为了支持用户使用自然语言进行灵活的数据查询，采用 NL2SQL 技术，实现查询请求向 SQL 查询语句的转化，并融合知识图谱来解决问句中语义的歧义问题[3]，提高 SQL 语句的准确性。

1　相关技术

NL2SQL 是自然语言到结构化查询语言的转换技术，主要应用在以自然语言进行关系数据库查询场景中。Zhong[4] 等在发布英文数据集 WikiSQL 时，提出基于序列到序列的方法 Seq2SQL，使用深度神经网络将自然语言问句转成相应的 SQL 查询。为了识别聚合函数和投影字段，使用了指针网络，并采用基于策略的强化学习来生成查询条件，最终在英文领域取得了 59.4% 的准确率。一些研究者引入注意力机制，帮助模型关注输入问句的关键部分，不仅大大提高了模型的准确率，还减少了对计算资源的需求。2018 年谷歌提出基于 Transforme[5] 的预训练模型 BERT，该模型在自然语言处理领域的多个任务中取得了重大突破。因此，在 2019 年 Hwang[6] 提出 SQLova 模型，该模型使用 BERT 作为编码器。同时 He[7] 等提出以 BERT 变体 MT-DNN[8] 作编码器的 X-SQL 网络模型，将 NL2SQL 在英文数据集上的准确率提高到了 88.7%。在中文领域，2019 年追一科技推出 TableQA 数据集时，Zhang[9] 等提出 M-SQL 模型，该模型以 BETR 作为编码器，同时根据 SQL 查询语句的结构，将 NL2SQL 分为 8 个子任务，获得了在 TableQA 数据集上最高的准确率。

基于深度学习的 NL2SQL 方法虽然在通用数据集上取得了显著成果，而当面对不规范数据集时，该方法遇到的语义歧义问题仍然限制了其应用的效果。由于不规范的数据集语义不明确，导致模型预测出来的 SQL 语句无法正确执行或者得不到正确的查询结果。在行车数据中，大量的英文简写和特殊符号使得数据之间相关性较差，相对于 TableQA 数据集而言更为复杂。因此，利用融合知识图谱的 NL2SQL 方法，将自然语言问句转换成 SQL 查询，使用知识图谱解决问句中语义歧义的问题[10]，以帮助用户更准确地查询数据。

2　行车数据库知识图谱构建

相较于通用领域数据集 TableQA，行车领域数据集的主要挑战在于模型输入是实际应用过程中用户所使用的不规范问句。这些不规范问句表现在以下方面：

（1）问句中包含英文和拼音首字母缩写；

（2）设备名称通常具有口语化或者别称；

（3）有些问句未指定 SQL 语句的所有必要组成成分（如：缺少表名或者投影字段）。针对上述实际应用场景中出现的问题，设计并构建了一个行车数据库知识图谱。

知识图谱是一种基于图形结构的知识表示和存储方法，构建图谱的数据源包括结构化、半结构化和非结构化数据[11]。因此，采用某工厂提供的行车结构化和半结构化数据来构建知识图谱，其模式层如图 1 所示。

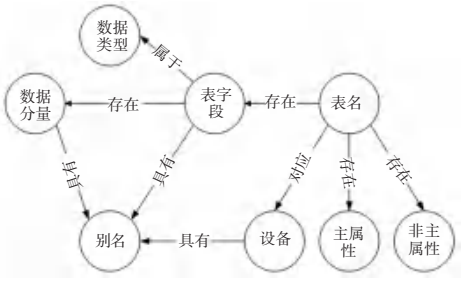

图 1　行车数据库知识图谱模式层

Fig. 1　Mode layer of knowledge map of crane database

其中，别名以外的实体和关系均由结构化数据得到，别名实体及其相关关系从行车人员提供的操作日志文件获取，通过文本清洗和解析，将文本数据转换为结构化数据。然后，采用基于规则的方式从半结构化的日志文件中抽取别名实体，得到类似“（‘大口径直缝焊管’ 别名 ‘UOE’）”格式的三元组。通过对日志文件的解析以及百度百科别名信息栏的爬虫筛选[12]，获取了 1.2 万条同义词三元组，其中包括“设备-别名”、“数据分量-别名”、“表字段-别名”3 种关系。

3　基于 M-SQL 多任务学习模型的 SQL 生成

3.1　SQL 查询语句预测任务分析

NL2SQL 任务的目标是在给定数据表信息的情况下，将用户提出的问句解析成 SQL 查询语句。例如，对行车基本信息表提出的问题是：“大口径直缝钢管作业区中车间是三车间的行车，其设备六位码

和设备编号是什么?",模型可以解析生成图 2 中的 SQL 语句。

图 2 行车领域数据集样例

Fig. 2 Sample data set of crane field

根据 SQL 规范,SQL 查询语句一般由投影字段、表名、条件子句等多个部分构成[13]。基于 SQL 查询语句的结构设计深度学习模型,可以将预测任务分解成多个子任务,每个子任务分别对应 SQL 的某一个组成部分。这种分解方式不仅可以简化网络结构,而且具有更强的解释性。采用通用的 SQL 查询语句结构定义预测任务,其结构如下:

SELECT #S-AGG(#S-COL),……

FROM #T-NAME

WHERE #W-COL #W-OP #W-VAL #W-CON ……

其中,前缀带有#的部分表示需要预测的目标文本,省略号表示投影字段和条件子句存在多个。预测子任务中,#S-AGG 表示聚合函数,#S-COL 表示投影字段;#T-NAME 表示表名;#W-COL 表示条件子句的条件列,#W-OP 表示条件子句的操作符,取值可以是"=""!="、">"和"<";#W-VAL 表示条件子句的条件值,#W-CON 表示多个条件子句之间的连接符,取值可以是"null"、"and"、"or"。

3.2 基于 M-SQL 的多任务学习模型

M-SQL 是一种将自然语言转为 SQL 语句的多任务表示学习方法,主要由编码问句、基于注意力机制编码列表示,以及 8 个子任务模型组成。在基准模型中,输入是由问句和列名拼接而成,并通过预训练模型编码后得到编码向量。然而,行车领域数据集中存在较多的长字段名,且表中列名数量众多,将所有列名和问句一起作为输入序列会使编码模型的输入超过预训练模型规定的长度限制,损失超出部分的语义信息。此外,问句和过多的无关列名一起编码也会影响后续预测投影字段任务的准确率。因此,提出采用列名分别和问句拼接的方式作为编码器的输入,通过预训练模型 BERT-wwm-ext 对输入进行编码之后,再进行各个子任务预测。主干模型的整体结构如图 3 所示。

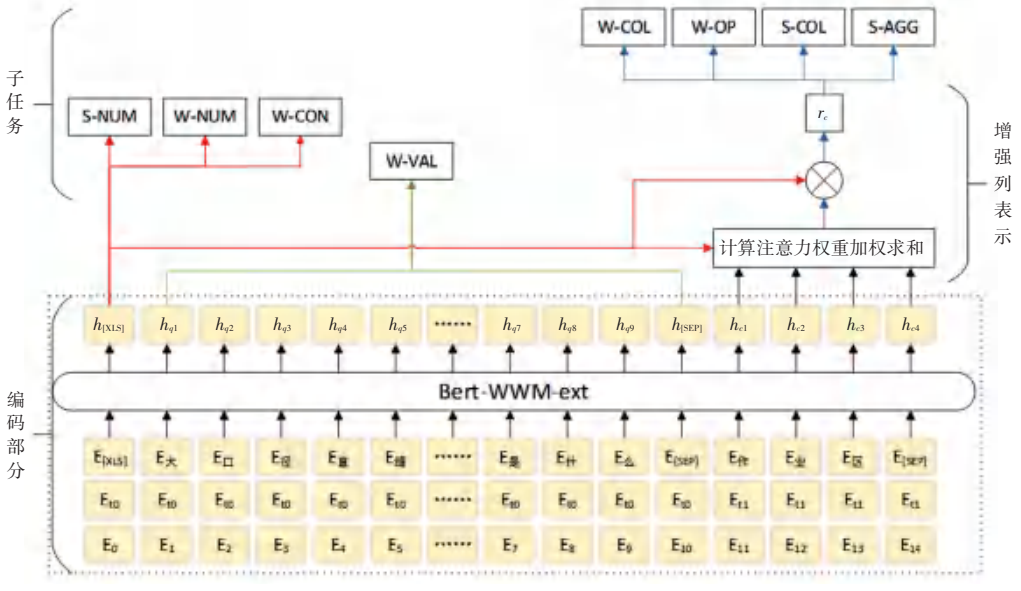

图 3 主干模型整体结构

Fig. 3 Overall structure diagram of trunk model

在编码部分,首先需要将输入的问句与其对应的列名进行拼接,如图 3 中模型的输入是图 2 中问句和数据表的字段名拼接得到的,并且在问句和表名之间通过[SEP]标记进行分割。使用[XLS]表示序列信息,[pad]表示填充符。特征表达编码的输入信息是 3 种类型信息的叠加,其中包括问句词嵌入、类型嵌入和位置嵌入。类型嵌入是对输入序列的类型信息进行编码,其中 E_{t0} 表示输入的问句,E_{t1} 表示数据表的列名。E_i 表示位置嵌入,在编码过程中使得模型具备学习词序的能力。

与 M-SQL 类似,在预训练模型对输入序列进行编码后,使用问句全局语义信息的 $h_{[xls]}$,通过注

意力机制来增强每列的语义表示，使得网络能够专注特定有用的输入[14]。计算全局信息 $h_{[\mathrm{xls}]}$ 和第 t 列的注意力权重如下所示：

$$S_t = dot(Uh_{[\mathrm{xls}]}, V\boldsymbol{h}_{ct}) \tag{1}$$

$$a_t = \frac{S_t}{\sum_{m=1}^{n} S_m} \tag{2}$$

式(1)中，$\boldsymbol{h}_{ct}$ 表示当前和问句拼接的列名的第 t 个字符对应的编码向量；U 和 V 表示模型可学习的参数，经过点积得到 $h_{[\mathrm{xls}]}$ 和列名中第 t 个 token 之间的匹配程度。在式(2)中，根据列名每个字符对应的相似度[15]，计算得到列名中第 t 个 token 的注意力权重。最后列的表示如下：

$$\bar{r}_c = \sum_{t=1}^{n} a_t \boldsymbol{h}_{ct} \tag{3}$$

$$r_c = \bar{r}_c + h_{[\mathrm{xls}]} \tag{4}$$

其中，n 是列名的字符个数，最终的列表示由带有注意力权重的列编码求和之后再和 $h_{[\mathrm{xls}]}$ 相加得到。在式(4)中加入 $h_{[\mathrm{xls}]}$ 是为了建立预测投影字段个数和条件子句个数两个子任务和其他子任务之间的联系。

通过编码向量可以预测 SQL 的其他组成部分，子任务均采用了全连接层，其主要作用就是将前面提取的语义特征在该层经过非线性变化，提取这些特征之间的关联，最后映射到输出空间，得到子任务的预测结果。

1)S-NUM、W-NUM、W-CON 预测子任务

子任务 S-NUM 预测的是投影字段的个数；W-NUM 预测条件子句的个数，根据实际应用场景规定可预测的个数最多是 3 个；子任务 W-CON 负责预测条件子句之间的连接符。这 3 个子任务都是多分类任务，均基于 $h_{[\mathrm{xls}]}$ 来完成，其概率分布如下：

$$p_i = \mathrm{Softmax}(W_i h_{[\mathrm{xls}]}) \tag{5}$$

2)S-COL、S-AGG、W-COL、W-OP 预测子任务

这些子任务分别预测投影字段、聚合函数、条件子句条件列和条件子句操作符。在 $h_{[\mathrm{xls}]}$ 通过注意力机制来增强每列的语义表示之后，使用代表列名语义信息的编码向量来做分类任务，其概率分布如下：

$$p_i = \mathrm{Softmax}(W_i r_{ci}) \tag{6}$$

3)W-VAL 预测子任务

与前述子任务不同，条件子句列值 W-VAL 预测结果的取值范围是不确定的。因此，采用分类标签的方式将条件值在问句的位置标记出来，将自然语言问句中的每个字符标记上 0 或者 1 的标签，其用 1 表示属于条件值。如图 2 所示的问句，模型会在“大口径直缝”和“三车间”处标记 1，其他位置为 0。可通过下式计算问句中每个字符是 0 或者 1 的概率。

$$p_{w\text{-}val} = Sigmoid(W_{w\text{-}val}\boldsymbol{h}_{qi}) \tag{7}$$

其中，$\boldsymbol{h}_{qi}$ 表示问句第 i 个字符的编码向量。使用该向量进行二分类任务，判断每个字符是否为条件值。

4　融合知识图谱的 SQL 语句修正

4.1　基于 Jaccard 相似指数的表名提取

基于 M-SQL 深度学习模型的前提都是已知数据表名称，但在实际的工业场景中，用户往往并不会提供要查询的表名，因此需要根据问句语义信息来预测表名。利用自定义词典和 jieba 分词技术对问句进行分词，获得分词后的词汇集合，并采用 Jaccard 相似度指数计算该集合与数据库表名集合之间的相似度，以此来预测表名。Jaccard 相似指数是用来衡量两个集合之间相似性的指标，其被定义为两个集合交集的元素个数除去并集的元素个数，具体计算方法如下：

$$J_i(A, B_i) = \frac{|A \cap B_i|}{|A \cup B_i|} \tag{8}$$

其中，集合 A 代表从问句中获取的所有名词集合，集合 B 代表和数据表相关的词汇集合，其元素由知识图谱的别名、表名、表字段实体获得，i 表示数据库中存在的表名数量。由于单表的 NL2SQL 问句只涉及一个表名，因此在设计 A、B 两个集合的内容时，考虑表名、表名别名、表名缩写、数据表中所有的列名信息。通过对比问句分词集合和每个表名相关词汇集合的 Jaccard 距离，来选择最符合问句意图的表名。

例如图 2 所示的问句，得到的集合 A 和集合 B_i 部分元素如图 4 所示。由于集合 B_1 和 A 交集最多，在保证 B_i 元素数量一致的情况下，计算得到的相似指数得分最高，而 B_1 元素来源是“行车基本信息表”，所以可以认为问句和该表最相关。

图 4　预测表名示例

Fig. 4　Example of forecast table name

4.2 结合编辑距离的投影字段修正

行车领域数据集问句大多语义不明确,模型预测的列名和问句中的内容关系不大,图 2 的例子中投影字段是“设备六位码”和“设备编号”如果问句使用字段别名(如问句是“它的唯一序列号和机器序列号是什么?”),则模型无法准确匹配到这两个投影字段。因为“唯一序列号”并不是数据表的字段名,只是字段“六位码”的一种别称。此时采用基于编辑距离和引入知识图谱别名的方式来选择和问句最相关的列名,例如和问句最相关的字段名是“六位码”,同时将问句涉及到的设备重要的属性作为投影字段,例如“车间”和“作业区”等主属性。知识图谱中投影字段别名关系和主属性如图 5 所示。

图 5 部分知识图谱数据层

Fig. 5 Partial knowledge map data layer

在基于编辑距离和知识图谱的文本匹配中,问句通常由多个标准词组成,并可能存在前后无意义词汇以及列名的中英文缩写或特殊符号。针对由多个标准词组成的较短文本,可通过计算 Levenshtein Ratio 得到最优匹配[16]。其中规定针对文本的删除和插入操作都会赋予编辑距离值加 1,替换操作加 2。然而,编辑距离无法解决同义词匹配的问题,进一步结合知识图谱中别名关系来做文本匹配任务。具体处理流程如下:

(1)构建停用词表,去除问句中的语气词、代词和逗号等不相关的内容,使问句的关键词密度更高。同时遍历知识图谱别名构建语义词典,其内容格式为键值对,主要存储的是知识图谱的别名关系;

(2)处理后的问句分别与列名计算编辑距离,得到最优距离和对应的列名;

(3)遍历词典键值对,如果 key 是问句的一部分,则对应的 value 替换问句中和 key 内容相同的字符;重复步骤(2)直到键值对遍历结束;

(4)输出最优距离对应的列名作为问句最相关列名。

匹配方法的关键是将短文本问句与列名计算编辑距离,然后替换问句中关键字再计算编辑距离,解决编辑距离对同义词效果差的问题。

4.3 融合知识图谱的条件子句生成

在模型通过分类标签得到条件值后,需要和条件列、操作符匹配形成条件子句。但是由于语义不明确,模型标记的条件值可能并非真实条件值。例如图 2 所示的问句,标记的“大口径直缝钢管”在数据表中并不存在,其对应内容应该是“UOE”,通过引入知识图谱在标记条件值之后获得候选条件值集合。候选条件值的作用就是扩大模型预测的取值范围,最后再遍历候选条件值集合,将模型预测出来的 W-COL 和 W-OP 与集合中每个条件值拼接得到若干条件子句表达式,将问句分别和该表达式拼接经过编码后,利用全局语义信息的[XLS]标记做线性变换,确定条件值,其模型结构如图 6 所示。

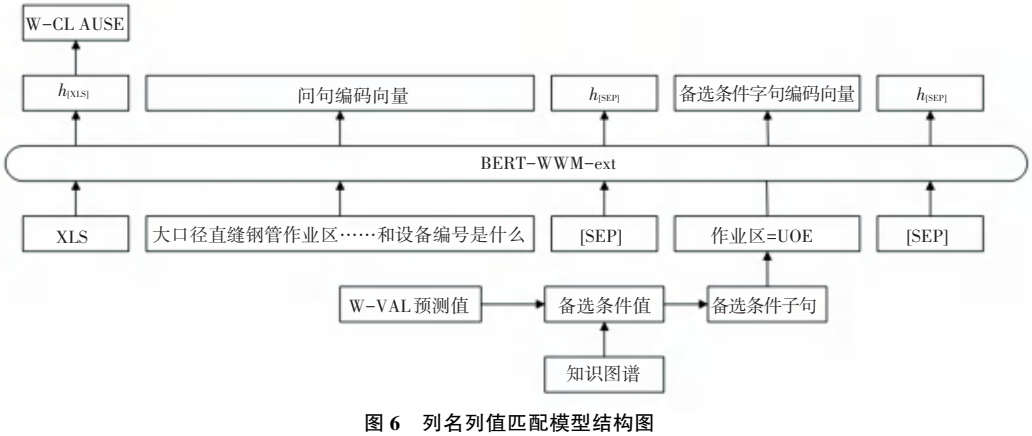

图 6 列名列值匹配模型结构图

Fig. 6 Structure diagram of column name value matching model

在生成候选集时,针对不同类型的条件值,制定不同的规则来提取候选条件值集合。对于数字类型的条件值,先通过一定的规则使得罗马数字形式和汉字简写以及大写加入候选集。例如:“价格超过

2 000 元的行车,其设备六位码是多少?”问句中,当 0/1 标记出“2 000”时,将“两千”、“贰仟”也包括在候选集中。对于字符串类型条件值,主要是错字、别名和英文简写等情况,无法通过简单的处理规则来进行处理,因此采用知识图谱别名关系的方式来解决,图谱中针对每个数据分量存储了大量的别名、简写等内容。

在得到条件值候选集之后,组装成候选条件子句集合,基于深度学习网络模型来预测候选条件子句是否和问句相关。例如图 2 所示的问句,W-VAL 任务标记了“大口径直缝钢管”,则使用 Cypher 语句从 Neo4j 图数据库中找到别名,制备候选集合{‘大口径直缝钢管’,‘UOE’,‘直缝钢管’,‘大口径钢管’,……},然后根据模型预测的条件子句列名“作业区”、“车间”和操作符“=”得到候选条件集合{‘作业区=大口径直缝钢管’,‘车间=大口径直缝钢管’,‘作业区=UOE’,‘车间=UOE’,……},最后筛选与问句最相关的条件子句。

5　实验结果分析

5.1　实验环境及数据集

实验均在操作系统 Ubuntu 18.04.4 LTS 的平台上完成。训练和测试的计算机硬件配置为:CPU 为 Intel(R) Xeon(R) CPU E5-2678 v3,内存 64 G,GPU 为 NVIDIA GeForce RTX2080Ti,开发环境为 Python 3.6、Pytorch 1.7.0。

在目前的中文单表 NL2SQL 研究中,主要使用的是开源中文数据集 CSpider[17] 和追一科技公开的 TableQA 数据集[18]。由于 TableQA 数据集由通用领域数据集构成,不存在工业生产上的专业词汇或者特殊符号简写等内容,问句相对比较规范,语义相对明确。因此在通用领域数据集 TableQA 上训练模型之后,又选择行车相关静态数据标注小型领域数据集进行迁移学习。领域数据集内容上包括行车上装备的各个子装置,其中各种装置涉及到的数据集数量见表 1。

表 1　行车领域数据集各子装置数据集数量

Table 1　Number of data sets of each sub-device in crane field data set

装置名称	数量
行车基本信息	220
车轮	100
变频器	100
电机	100

另外,为了解决问句语义不明确的问题,通过数据库表构建了知识图谱。其中表名和数据分量实体是由数据库表中的表名、表头、表内容获得,用于解决表名预测,主属性实体用于输出设备重要属性,别名实体用于解决语义不明确的问题。别名实体由员工操作记录、设备使用手册以及由百度百科获得。通过挖掘百度百科 infobox 上的别称、外文名信息,同时在设备文件中抽取实体来构建“(实体、具有、别名)”三元组。

5.2　实验结果分析

知识图谱着重解决 NL2SQL 模型中自然语言问句语义不明确的问题。在预测投影字段子任务时,问句语义不明确导致模型无法准确预测投影字段,则使用基于编辑距离结合知识图谱的文本匹配方法来选择和问句最相关的列名。为了选择合适于行车领域数据文本分类任务,对于几种常见的字符串匹配方法进行了实验测试。测试使用的数据集是行车领域问句和标记的问句最相关列名,使用各种文本匹配方法来选择和问句中最相关的列名,与真实答案比较,采用准确率来对比各种方法之间的效果,实验结果见表 2。

表 2　几种字符串匹配算法的对比实验

Table 2　Comparative experiment of several string matching algorithms

方法	准确率/%
Levenshtein 距离	86.52
Levenshtein Ratio	90.46
Jaro 距离	82.65
Synonyms	66.72
本文方法	94.88

根据实验结果可以看到,使用基于 Word2Vec 的中文近义词工具包(Synonyms)来预测行车领域数据,效果不是很好,而几种编辑距离则能够达到比较高的准确率。基于上述实验结果,选择 Levenshtein Ratio 作为文本匹配的方式,同时结合知识图谱构建语义词典解决同义词匹配问题。在实验中应用在行车数据,其准确率能够有效提高。

目前,NL2SQL 任务中预测条件值部分是该任务的难点,实验过程中基于行车领域数据集,复现了目前比较流行的几种 NL2SQL 模型,并且在 SQL 查询语句的条件值上进行了对比实验。将条件子句的逻辑准确率、整体 SQL 查询语句的逻辑形式准确率(AccLa)和执行准确率(AccEa)作为评估模型效果的指标,模型 SQLNet[19]、X-SQL、M-SQL 的实验结果见表 3。

分析表中结果发现,在进行了迁移学习以及引入知识图谱后,自然语言转 SQL 查询语句的准确率

在行车领域数据集上有很大提升。其中重要的原因是解决了模型预测条件值和真实条件值不匹配的问题,同时也解决了语义不明确导致的投影字段不明的问题。例如模型 M-SQL 在 TableQA 数据集上,逻辑形式准确率为 89.13%,执行准确率达到 91.86%。经过复现的 M-SQL 模型在 TableQA 数据集逻辑形式准确率和执行准确率分别是 87.52%、90.14%。但是在行车领域,由于数据集语义不够明确,导致 M-SQL 模型性能下降了 15%左右,在修正后模型在实际使用准确率达到 91.78%。经过知识图谱修正后,各子任务的准确率见表 4。

表 3 几种 NL2SQL 模型在行车领域数据集上的实验结果

Table 3 Experimental results of several NL2SQL models on data sets in the field of crane %

模型方案	W-Clause AccLa	AccLa	AccEa
SQLNet	57.69	46.35	51.84
X-SQL	78.86	68.08	72.72
M-SQL	82.72	74.86	76.02
知识图谱修正后的本文方法	96.85	90.03	91.78

表 4 本文的 Nl2SQL 方法在各个子任务中的准确率

Table 4 Accuracy of Nl2SQL method in this paper in each subtask

SQL 语句部分内容	准确率/%
F-NAME	96.83
S-NUM	99.45
S-COL	97.86
S-AGG	98.24
W-NUM	97.38
W-CON	97.45
W-COL	98.34
W-OP	99.07
W-Clause	96.85

预测其他子任务时需要首先预测表名,上表除 F-NAME 外,其他部分均是在提供正确表名的前提下计算得到的准确率。

6 结束语

本文研究了 NL2SQL 模型融合知识图谱在设备运维数据检索中的应用。通过行车工厂内部信息数据表以及员工操作记录等文件,构建了行车数据库知识图谱。以 M-SQL 为基准模型,设计了自然语言转 SQL 查询语句的网络模型,通过将 SQL 语句结构划分成多个子任务分别进行预测,根据行车领域数据集的特点,优化了模型的输入、编码部分以及部分子任务的细节。使用知识图谱解决 NL2SQL 任务中语义不明确造成的投影字段和条件值预测不准确的问题。实验结果表明,该方法在行车数据中取得了较好的效果,使得在获取行车数据时更加高效和准确。根据该方法设计并开发的行车设备检索交互程序在实际使用中也取得了较好的效果,可以为设备运维数据的查询和分析提供新的思路和方法。

参考文献

[1] GB/T 37366-2019. 塔式起重机安全监控系统及数据传输规范[S].

[2] 郭小雷,王宗彦,吴淑芳,等. 基于 SQL 的桥式起重机智能设计网络系统开发[J]. 机械设计与研究,2017,33(1):182-185.

[3] 李鑫,何芳州. 基于语义信息的大规模知识图谱补全算法[J]. 计算机仿真,2023,40(12):428-433.

[4] ZHONG V, XIONG C, SOCHER R. Seq2sql: Generating structured queries from natural language using reinforcement learning[J]. arXiv preprint arXiv:1709.00103, 2017.

[5] DEVLIN J, CHANG M W, LEE K, et al. Bert: Pre-training of deep bidirectional transformers for language understanding[J]. arXiv preprint arXiv:1810.04805, 2018.

[6] HWANG W, YIM J, PARK S, et al. A comprehensive exploration on wikisql with table-aware word contextualization[J]. arXiv preprint arXiv:1902.01069, 2019.

[7] HE P, MAO Y, CHAKRABARTI K, et al. X-SQL: Reinforce schema representation with context[J]. arXiv preprint arXiv: 1908.08113, 2019.

[8] LIU X, HE P, CHEN W, et al. Multi-task deep neural networks for natural language understanding[J]. arXiv preprint arXiv: 1901.11504, 2019.

[9] ZHANG X, YIN F, MA G, et al. M-SQL: Multi-task representation learning for single-table text2SQL generation[J]. IEEE Access, 2020, 8: 43156-43167.

[10] 贝毅君, 周勇, 高克威. 面向数控机床设备维护的知识问答技术[J]. 计算机集成制造系统, 2022, 28(9): 2881-2893.

[11] WANG Q, MAO Z D, WANG B, et al. Knowledge graph embeddings:A survey of approaches and applications[J]. IEEE Transactions on Knowledge and Data Engineering,2017,29(12): 2724-2743.

[12] 孙振维. 基于知识图谱的自然语言生成 SQL 语句模型研究[D]. 北京: 北京邮电大学,2021.

[13] 梁清源,朱琪豪,孙泽宇,等. 基于深度学习的 SQL 生成研究综述[J]. 中国科学:信息科学,2022,52(8):1363-1392.

[14] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is all you need[C]//Proceedings of the 31st International Conference on Neural Information Processing Systems. IEEE, 2017:6000-6010.

[15] 刘峻松,唐明靖,薛岗,等. 基于 Word2Vec 的编程领域词语拼写错误检测算法[J]. 计算机应用与软件,2022,39(3):277-284.

[16] PRAKOSO D W, ABDI A, AMRIT C. Short text similarity measurement methods: A review[J]. Soft Computing, 2021, 25: 4699-4723.

[17] MIN Q, SHI Y, ZHANG Y. A pilot study for Chinese SQL semantic parsing[J]. arXiv preprint arXiv:1909.13293, 2019.

[18] SUN N, YANG X, LIU Y. Tableqa: A large-scale chinese text-to-sql dataset for table-aware sql generation[J]. arXiv preprint arXiv:2006.06434, 2020.

[19] XU X, LIU C, SONG D. Sqlnet: Generating structured queries from natural language without reinforcement learning[J]. arXiv preprint arXiv:1711.04436, 2017.