(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 119512674 A
(43)申请公布日 2025.02.25

(21)申请号 202411724366.3
(22)申请日 2024.11.28
(71)申请人 浙江大学
地址 310058 浙江省杭州市西湖区余杭塘路866号
申请人 杭州星锐网讯科技有限公司
(72)发明人 韩东明 丁子祥 高军 汪建波
(74)专利代理机构 北京集佳知识产权代理有限公司 11227
专利代理师 王燕
(51)Int.Cl.
G06F 9/451(2018.01)
G06F 40/289(2020.01)
G06F 40/284(2020.01)
G06F 17/18(2006.01)
G06F 13/38(2006.01)

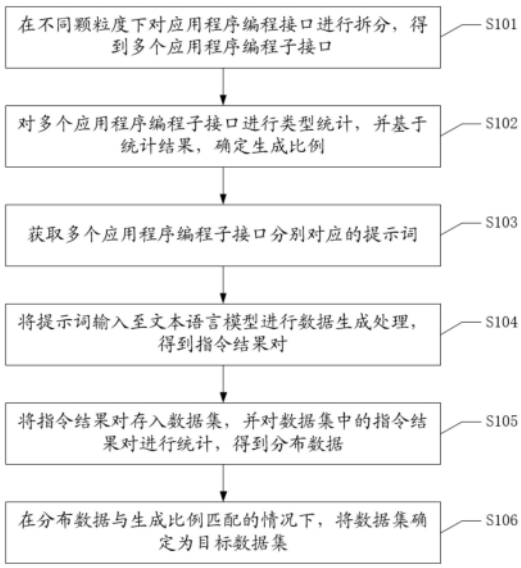
权利要求书2页 说明书16页 附图4页

(54)发明名称

一种数据生成方法、装置、设备及可读存储介质

(57)摘要

本申请公开了一种数据生成方法、装置、设备及可读存储介质,该方法,包括:在不同颗粒度下对应用程序编程接口进行拆分,得到多个应用程序编程子接口;对多个应用程序编程子接口进行类型统计,并基于统计结果,确定生成比例;获取多个应用程序编程子接口分别对应的提示词;将提示词输入至文本语言模型进行数据生成处理,得到指令结果对;其中,文本语言模型已学习了应用程序编程接口的技术文档和使用指南;将指令结果对存入数据集,并对数据集中的指令结果对进行统计,得到分布数据;在分布数据与生成比例匹配的情况下,将数据集确定为目标数据集。本申请技术效果:可以生成指令覆盖率高,且可定制数据分布的目标数据集。



CN 119512674 A

1.一种数据生成方法,其特征在于,包括:

在不同颗粒度下对应用程序编程接口进行拆分,得到多个应用程序编程子接口;

对多个所述应用程序编程子接口进行类型统计,并基于统计结果,确定生成比例;

获取多个所述应用程序编程子接口分别对应的提示词;

将所述提示词输入至文本语言模型进行数据生成处理,得到指令结果对;其中,所述文本语言模型已学习了所述应用程序编程接口的技术文档和使用指南;

将所述指令结果对存入数据集,并对所述数据集中的指令结果对进行统计,得到分布数据;

在所述分布数据与所述生成比例匹配的情况下,将所述数据集确定为目标数据集。

2.根据权利要求1所述的方法,其特征在于,在不同颗粒度下对应用程序编程接口进行拆分,得到多个应用程序编程子接口,包括:

获取所述应用程序编程接口的声明信息;

基于所述声明信息,确定不同的颗粒度;

在不同颗粒度下,对所述程序编程接口进行拆分,得到多个所述应用程序编程子接口。

3.根据权利要求1所述的方法,其特征在于,在所述分布数据与所述生成比例不匹配的情况下,还包括:

通过重新获取新的提示词对所述数据集进行补充,以使补充后的数据集对应的分布数据与所述生成比例匹配。

4.根据权利要求3所述的方法,其特征在于,通过重新获取新的提示词对所述数据集进行补充,包括:

对比所述分布数据与所述生成比例,确定缺乏的指令结果对的目标类型;

将与所述目标类型匹配的应用程序编程子接口确定为目标应用程序编程子接口;

重新获取所述目标应用程序编程子接口对应的提示词;

将新获取的提示词输入至文本语言模型进行数据生成处理,得到补充的指令结果对;

将补充的指令结果对写入所述数据集中。

5.根据权利要求1所述的方法,其特征在于,基于统计结果,确定生成比例,包括:

在可视化界面输出所述统计结果;

获取输入的配置信息;

利用所述配置信息,确定所述生成比例。

6.根据权利要求1至5任一项所述的方法,其特征在于,利用所述配置信息,确定所述生成比例,包括:

从所述配置信息中读取中各类别的类别配比,基于所述类别配比确定所述生成比例。

7.根据权利要求6所述的方法,其特征在于,判断所述分布数据与所述生成比例是否匹配,包括:

利用所述类别配比得到各类别下的子类的子类配比;

从所述配置信息中读取各子类的最少拓展记录数;

利用所述子类配比和所述最少拓展记录数,确定所述子类别对应的最少记录条数;

基于所述最少记录条数计算满足所述类别配比的最小总记录数;

判断所述最少记录条数、所述最小总记录数与所述分布数据是否匹配;

如果是,则确定所述分布数据与所述生成比例匹配;

如果否,则确定所述分布数据与所述生成比例不匹配。

8.一种数据生成装置,其特征在于,包括:

拆分模块,用于在不同颗粒度下对应用程序编程接口进行拆分,得到多个应用程序编程子接口;

生成比例统计模块,用于对多个所述应用程序编程子接口进行类型统计,并基于统计结果,确定生成比例;

提示处理模块,用于获取多个所述应用程序编程子接口分别对应的提示词;

指令结果对生成模块,用于将所述提示词输入至文本语言模型进行数据生成处理,得到指令结果对;其中,所述文本语言模型已学习了所述应用程序编程接口的技术文档和使用指南;

分析模块,用于将所述指令结果对存入数据集,并对所述数据集中的指令结果对进行统计,得到分布数据;

数据生成模块,用于在所述分布数据与所述生成比例匹配的情况下,将所述数据集确定为目标数据集。

9.一种电子设备,其特征在于,包括:

存储器,用于存储计算机程序;

处理器,用于执行所述计算机程序时实现如权利要求1至7任一项所述数据生成方法的步骤。

10.一种可读存储介质,其特征在于,所述可读存储介质上存储有计算机程序,所述计算机程序被处理器执行时实现如权利要求1至7任一项所述数据生成方法的步骤。

一种数据生成方法、装置、设备及可读存储介质

技术领域

[0001] 本申请涉及信息处理技术领域,特别是涉及一种数据生成方法、装置、设备及可读存储介质。

背景技术

[0002] API文档和使用指南通常针对专家领域,其目的是为了准确描述API的功能并避免产生歧义。为此,通常使用标准化的语言和统一的格式来对API进行描述和定义。然而,当普通用户或非专业用户使用API时,他们往往面临理解上的巨大障碍,必须通过查阅文档来理解API的功能和用法。这种理解鸿沟显著增加了他们正确调用API的难度。

[0003] 目前,通过自然语言控制API的调用和参数设置。由于自然语言处理(NLP,Natural Language Processing)技术的发展,使得模型的语义理解与代码的对齐更加精准。然而,虽然模型在广泛领域内具备了出色的泛化能力,但对于定制化API功能,基础模型的数据往往难以覆盖。这时,需要通过大量任务数据来进一步预训练或监督微调(SFT,Supervised Fine-Tuning),以增强模型对特定领域的理解。

[0004] 为了使生成式模型(如GPT)掌握特定领域的API,必须通过大量的含有指令结果对(包括输入指令与API结果的对应关系,输入指令即用户输入的描述指令,API结果即描述指令程序化的指令结果)的数据集进行训练。但是,目前生成这种数据集的方案,存在指令覆盖率低、数据分布不均衡等问题,进一步导致生成式模型无法有效学习到更为全面的API。

[0005] 综上所述,如何有效地解决用于训练生成式模型的数据集的生成等问题,是目前本领域技术人员急需解决的技术问题。

发明内容

[0006] 本申请的目的是提供一种数据生成方法、装置、设备及可读存储介质,以获取指令覆盖率高,且可定制数据分布的目标数据集。

[0007] 为解决上述技术问题,本申请提供如下技术方案:

[0008] 一种数据集生成方法,包括:

[0009] 在不同颗粒度下对应用程序编程接口进行拆分,得到多个应用程序编程子接口;

[0010] 对多个所述应用程序编程子接口进行类型统计,并基于统计结果,确定生成比例;

[0011] 获取多个所述应用程序编程子接口分别对应的提示词;

[0012] 将所述提示词输入至文本语言模型进行数据生成处理,得到指令结果对;其中,所述文本语言模型已学习了所述应用程序编程接口的技术文档和使用指南;

[0013] 将所述指令结果对存入数据集,并对所述数据集中的指令结果对进行统计,得到分布数据;

[0014] 在所述分布数据与所述生成比例匹配的情况下,将所述数据集确定为目标数据集。

[0015] 优选地,在不同颗粒度下对应用程序编程接口进行拆分,得到多个应用程序编程

子接口,包括:

[0016] 获取所述应用程序编程接口的声明信息;

[0017] 基于所述声明信息,确定不同的颗粒度;

[0018] 在不同颗粒度下,对所述程序编程接口进行拆分,得到多个所述应用程序编程子接口。

[0019] 优选地,在所述分布数据与所述生成比例不匹配的情况下,还包括:

[0020] 通过重新获取新的提示词对所述数据集进行补充,以使补充后的数据集对应的分布数据与所述生成比例匹配。

[0021] 优选地,通过重新获取新的提示词对所述数据集进行补充,包括:

[0022] 对比所述分布数据与所述生成比例,确定缺乏的指令结果对的目标类型;

[0023] 将与所述目标类型匹配的应用程序编程子接口确定为目标应用程序编程子接口;

[0024] 重新获取所述目标应用程序编程子接口对应的提示词;

[0025] 将新获取的提示词输入至文本语言模型进行数据生成处理,得到补充的指令结果对;

[0026] 将补充的指令结果对写入所述数据集中。

[0027] 优选地,基于统计结果,确定生成比例,包括:

[0028] 在可视化界面输出所述统计结果;

[0029] 获取输入的配置信息;

[0030] 利用所述配置信息,确定所述生成比例。

[0031] 优选地,利用所述配置信息,确定所述生成比例,包括:

[0032] 从所述配置信息中读取中各类别的类别配比,基于所述类别配比确定所述生成比例。

[0033] 优选地,判断所述分布数据与所述生成比例是否匹配,包括:

[0034] 利用所述类别配比得到各类别下的子类的子类配比;

[0035] 从所述配置信息中读取各子类的最少拓展记录数;

[0036] 利用所述子类配比和所述最少拓展记录数,确定所述子类别对应的最少记录条数;

[0037] 基于所述最少记录条数计算满足所述类别配比的最小总记录数;

[0038] 判断所述最少记录条数、所述最小总记录数与所述分布数据是否匹配;

[0039] 如果是,则确定所述分布数据与所述生成比例匹配;

[0040] 如果否,则确定所述分布数据与所述生成比例不匹配。

[0041] 一种数据生成装置,包括:

[0042] 拆分模块,用于在不同颗粒度下对应用程序编程接口进行拆分,得到多个应用程序编程子接口;

[0043] 生成比例统计模块,用于对多个所述应用程序编程子接口进行类型统计,并基于统计结果,确定生成比例;

[0044] 提示处理模块,用于获取多个所述应用程序编程子接口分别对应的提示词;

[0045] 指令结果对生成模块,用于将所述提示词输入至文本语言模型进行数据生成处理,得到指令结果对;其中,所述文本语言模型已学习了所述应用程序编程接口的技术文档

和使用指南;

[0046] 分析模块,用于将所述指令结果对存入数据集,并对所述数据集中的指令结果对进行统计,得到分布数据;

[0047] 数据生成模块,用于在所述分布数据与所述生成比例匹配的情况下,将所述数据集确定为目标数据集。

[0048] 一种电子设备,包括:

[0049] 存储器,用于存储计算机程序;

[0050] 处理器,用于执行所述计算机程序时实现上述数据生成方法的步骤。

[0051] 一种可读存储介质,所述可读存储介质上存储有计算机程序,所述计算机程序被处理器执行时实现上述数据生成方法的步骤。

[0052] 应用本申请实施例所提供的方法,在不同颗粒度下对应用程序编程接口进行拆分,得到多个应用程序编程子接口;对多个应用程序编程子接口进行类型统计,并基于统计结果,确定生成比例;获取多个应用程序编程子接口分别对应的提示词;将提示词输入至文本语言模型进行数据生成处理,得到指令结果对;其中,文本语言模型已学习了应用程序编程接口的技术文档和使用指南;将指令结果对存入数据集,并对数据集中的指令结果对进行统计,得到分布数据;在分布数据与生成比例匹配的情况下,将数据集确定为目标数据集。

[0053] 在不同颗粒度下对应用程序编程接口进行拆分,从而得到不同颗粒度的应用程序编程子接口。对这些应用程序编程子接口进行类型统计,并基于统计结果可以确定出生成比例。为了使得生成包括有与应用程序编程接口相关的指令结果对的数据集覆盖范围更全面,在生成提示词时,会获取这些应用程序编程子接口分别对应的提示词,然后,利用已经学习了应用程序编程接口的技术文档和使用指南的文本语言模型进行数据生成处理,即可得到指令结果对。将生成的指令结果对写入数据集中,并对数据集中的指令结果对进行统计分析,在分布数据与生产比例匹配的情况下,即可确定完成数据集生成,得到目标数据集。

[0054] 本申请技术效果:在生成数据集中的指令结果对时,通过对应用程序编程接口进行拆分,可使得指令结果对可以覆盖应用程序编程接口的不同颗粒度指令,通过对应用程序编程子接口进行统计,可以确定出生成比例,在获得数据集中的指令结果对的分布数据后,当明确生成比例与分布数据匹配的情况下,确定得到目标数据集,可以使得最终生成的目标数据集的数据分布可控,为进一步训练生成式模型训练,提供全面且数据分布可定制化的训练样本。

[0055] 相应地,本申请实施例还提供了与上述数据生成方法相对应的数据生成装置、设备和可读存储介质,具有上述技术效果,在此不再赘述。

附图说明

[0056] 为了更清楚地说明本申请实施例或相关技术中的技术方案,下面将对实施例或相关技术描述中所需要使用的附图作简单地介绍,显而易见地,下面描述中的附图仅仅是本申请的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0057] 图1为本申请实施例中一种数据生成方法的实施流程图;

[0058] 图2为本申请实施例中一种数据生成方法的实施示意图;

[0059] 图3为本申请实施例中一种数据生成装置的结构示意图;

[0060] 图4为本申请实施例中一种电子设备的结构示意图;

[0061] 图5为本申请实施例中一种电子设备的具体结构示意图。

具体实施方式

[0062] 为了使本技术领域的人员更好地理解本申请方案,下面结合附图和具体实施方式对本申请作进一步的详细说明。显然,所描述的实施例仅仅是本申请一部分实施例,而不是全部的实施例。基于本申请中的实施例,本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例,都属于本申请保护的范围。

[0063] 请参考图1,图1为本申请实施例中一种数据生成方法的流程图,该方法包括以下步骤:

[0064] S101、在不同颗粒度下对应用程序编程接口进行拆分,得到多个应用程序编程子接口。

[0065] 为便于描述,下面用应用程序编程接口的简称用API进行描述,用子AIP来指代应用程序编程子接口。子API即对应API的子功能模块。

[0066] 在本申请中,可以预先设置不同的颗粒度,然后基于不同的颗粒度对API进行拆分,从而得到多个子API。

[0067] 例如,若将API的内部功能按照父子关系和逻辑架构形象化表示为一个树状结构,则从根节点到每一个叶子节点的路径都可对应一个子API。

[0068] 在本申请中的一种具体实施方式中,在不同颗粒度下对应用程序编程接口进行拆分,得到多个应用程序编程子接口,包括:

[0069] 步骤一、获取应用程序编程接口的声明信息;

[0070] 步骤二、基于声明信息,确定不同的颗粒度;

[0071] 步骤三、在不同颗粒度下,对程序编程接口进行拆分,得到多个应用程序编程子接口。

[0072] 为便于描述,下面将上述步骤结合起来进行说明。

[0073] 在本实施例中,可以通过读取存储设备,或接收输入等方式,获得API的声明信息。

[0074] 图2所示,该声明信息可以是技术人员(开发者)对API声明的解释性信息。声明信息(即属于根须要求)中可以具体包括如下特征:

[0075] 描述:对API的整体功能进行详细描述,包括使用场景和适用范围。

[0076] 类型:例如颜色、枚举值、字符串、函数等,明确返回值的类型。

[0077] 层级:确定当前API所属的层级,即它是哪个模块的子节点。

[0078] 依赖关系:说明该API是否需要与其他API协同工作。例如,调整页脚之前,可能需要先启用页脚功能。

[0079] 可枚举项:列出该API支持的枚举值,如字体大小(小四、大四等)、位置(左上、右下等)。

[0080] 拓展词:罗列与当前API相关的关键指令或关键词,如与high这一词相关的最高、

最大、最上等词。

[0081] 举例说明,下方是一种具体的API的声明信息:

[0082] {"1级api": "footnote","1级描述": "脚注,用来数据生成","1级数据生成": "数据生成","2级api_name": "footnote_position","2级描述": "调整文档中脚注的位置","2级类型": "枚举值","2级依赖关系": ["启用脚注"],"2级可枚举项": [{"value": "左下","拓展词": ["leftbottm","左侧"]},{"value": "右下","拓展词": ["rightbottom","右侧"]},{"value": "上","拓展词": ["top","上","最高"]},{ "value": "下", "拓展词": ["bottom","下","最低"]}],"2级拓展词": ["脚注","页脚注","下面的文字"],...其他内容}。

[0083] 基于上一步的API的声明信息中的特征,可将API拆分为不同颗粒度的子API。拆分过程,具体包括:

[0084] 类型组合:根据API的类型、层级、依赖关系、可枚举项和拓展词,将其组合成多个子API,以便更细致地处理和生成数据。例如,一个复杂的API可以拆分成多个子功能模块,各自独立生成数据。

[0085] 层级分析:考虑API的层级关系,将顶层API与其子API拆分,以确保每个层级的API都能得到充分的覆盖。

[0086] 通过这一步的拆分,可以对API进行更加精细的操作和控制,方便后续的数据处理。

[0087] 举例说明:针对上述列举的API的声明信息,可以拆分出以下3个子API:

[0088] 子API1:{"1级api":"footnote","1级描述":"脚注,用来数据生成","1级数据生成":"数据生成","2级api_name":"footnote_position","2级描述":"调整文档中脚注的位置","2级类型":"枚举值","2级依赖关系":["启用脚注"],"2级可枚举项":{"value":"左下","拓展词":{"leftbottm"},"2级拓展词":["脚注"],...其他内容}。

[0089] 子API2:{"1级api":"footnote","1级描述":"脚注,用来数据生成","1级数据生成":"数据生成","2级api_name":"footnote_position","2级描述":"调整文档中脚注的位置","2级类型":"枚举值","2级依赖关系":["启用脚注"],"2级可枚举项":{"value":"左下","拓展词":{"左侧"},"2级拓展词":["页脚注"],...其他内容}。

[0090] 子API3:{"1级api":"footnote","1级描述":"脚注,用来数据生成","1级数据生成":"数据生成","2级api_name":"footnote_position","2级描述":"调整文档中脚注的位置","2级类型":"枚举值","2级依赖关系":["启用脚注"],"2级可枚举项":{"value":"左下","拓展词":{"左侧"},"2级拓展词":["下面的字"],...其他内容}。

[0091] 在实际应用中,在针对某个API生成其对应的数据集时,可仅针对该API进行拆分及后续处理,当针对多个API生成对应的数据集时,可以对这些API都进行拆分,也可以仅针对其中的部分API进行拆分处理。实际拆分情况,可以根据实际训练生成模型的需求而定,在此不做限定。

[0092] S102、对多个应用程序编程子接口进行类型统计,并基于统计结果,确定生成比例。

[0093] 拆分出子API之后,便可对这些子API进行类型统计,从而得到统计结果。基于这些统计结果,可以确定出生成比例。

[0094] 在拆分出子API之后,可以对子API的不同颗粒度和类型的数量分布信息统计,该统计结果中,可以包括不同类型的子API的分布情况,例如,子API的总数量,某类型下的子API的数量等统计信息。

[0095] 在API拆分完成后,接下来需要对不同颗粒度和不同类型的API进行统计,得到它们的数量分布。如此,便可明确数据的分布情况,并为后续的数据比例确定提供依据。

[0096] 得到统计信息之后,可以基于该统计信息直接指定生成比例。

[0097] 在本申请中的一种具体实施方式中,基于统计结果,确定生成比例,包括:

[0098] 步骤一、在可视化界面输出统计结果;

[0099] 步骤二、获取输入的配置信息;

[0100] 步骤三、利用配置信息,确定生成比例。

[0101] 为便于描述,下面将上述三个步骤结合起来进行说明。

[0102] 在可视化界面展示该统计结果,可以通过列表形式,也可以通过树形拓扑等方式进行展示输出。

[0103] 然后,用户根据该统计结果,并根据实际需求,可以输入配置信息。该配置信息中可以直接携带生成比例,也可以直接携带各个类别的生成记录数,从而基于生成记录计算出生成比例。

[0104] 在本申请中的一种具体实施方式中,利用配置信息,确定生成比例,包括:

[0105] 从配置信息中读取中各类别的类别配比,基于类别配比确定生成比例。

[0106] 也就是说,直接基于各类别的类别配比,确定出生成比例。例如,若类别仅有A、B和C三种,且其比例为1:2:1,则A、B和C三种生成比例可以直接为1:2:1。

[0107] S103、获取多个应用程序编程子接口分别对应的提示词。

[0108] 为了使得文本语言模型能够生成各子API的指令结果对,可以获取到个子API分别对应的提示词。

[0109] 具体的,可以根据不同颗粒度的API特性,开发者可以定制化设计提示词。这些提示词将结合API的类型、层级、依赖关系等,以确保生成的数据更加精准。提示词的设计应细化到具体API类别,以避免对其他API类别的干扰。

[0110] 举例说明:一种具体的提示词的如下:

[0111] 任务:用户使用自然语言来控制组件的字段、属性、事件、参数。

[0112] 限制:1.query为用户实际需求的的命令式陈述,要求丰富多样,贴近用户的使用习惯;2.为当前的可选项及拓展词生成对应的API;

[0113] 注意:输出JSON格式,不需要道歉和解释,不需要生成注释。重复一遍,不需要道歉和解释,不需要生成注释。

[0114] 结果:使用json包含List的格式,每个元素包含query和api两个字段,事件的参数要根据对应字段的内容进行生成,query为用户的自然语言描述,api为对应的json格式。

[0115] 案例:输入为{example_input},输出为{example_output};

[0116] 当前:输入为{d},请输出{k}条结果。

[0117] S104、将提示词输入至文本语言模型进行数据生成处理,得到指令结果对。

[0118] 其中,文本语言模型已学习了应用程序编程接口的技术文档和使用指南。

[0119] 需要注意的是,在本申请实施例中,为了使得文本语言模型可以基于提示词生成

指令结果对,可以预先让文本语言模型学习API的技术文档和使用指南。对于具体如何使得文本语言模型学习API的技术文档和使用指南,可以参照文本语言模型的具体训练方案,在此不再一一赘述。

[0120] 该文本语言模型可以具体为诸如GPT(Generative Pretrained Transformer,即生成式预训练Transformer,GPT是一种基于 Transformer 架构的语言模型)生成模型的模型。

[0121] 下面以文本语言模型为GPT生成模型为例,对指令结果对的生成进行详细说明:

[0122] 将定制化的提示词输入GPT生成模型中,利用其强大的生成能力,生成大量结构化的API指令和结果对(即指令结果对)。这一步将生成所需的多样化数据,为后续的数据处理和分析提供基础。

[0123] 其中,指令结果对包括输入的描述指令和描述指令的代码化结果,如,用户输入指令“将标题颜色设为红色”,代码化结果为{title:{color:red}}。

[0124] 例如,通过GPT生成模型生成的指令结果对的部分示例如下:

[0125] "[{\"query\":\"将标签栏组件设置为线型显示\",\"api\":{\"atom_tabs\":{\"type\":\"line\"}}},{\"query\":\"我希望标签栏的样式是线型的\",\"api\":{\"atom_tabs\":{\"type\":\"line\"}}},{\"query\":\"修改标签栏组件为线型布局\",\"api\":{\"atom_tabs\":{\"type\":\"line\"}}},{\"query\":\"设置标签栏组件为线型样式\",\"api\":{\"atom_tabs\":{\"type\":\"line\"}}},{\"query\":\"如何将标签栏组件变成线型显示\",\"api\":{\"atom_tabs\":{\"type\":\"line\"}}},{\"query\":\"让标签栏组件采用线型设计\",\"api\":{\"atom_tabs\":{\"type\":\"line\"}}},{\"query\":\"请将标签栏改为线型样式\",\"api\":{\"atom_tabs\":{\"type\":\"line\"}}},{\"query\":\"我需要标签栏组件显示为线型\",\"api\":{\"atom_tabs\":{\"type\":\"line\"}}},{\"query\":\"更改标签栏组件的样式为线型\",\"api\":{\"atom_tabs\":{\"type\":\"line\"}}},{\"query\":\"切换标签栏组件的类型为线型\",\"api\":{\"atom_tabs\":{\"type\":\"line\"}}},{\"query\":\"设置标签栏为块状类型\",\"api\":{\"atom_tabs\":{\"type\":\"block\"}}},{\"query\":\"我希望标签栏的样式是块状类型的\",\"api\":{\"atom_tabs\":{\"type\":\"block\"}}},{\"query\":\"我想把标签栏组件改成块状类型\",\"api\":{\"atom_tabs\":{\"type\":\"block\"}}},{\"query\":\"如何让标签栏组件呈现为块状类型\",\"api\":{\"atom_tabs\":{\"type\":\"block\"}}},{\"query\":\"将标签栏组件的样式设置为块状类型\",\"api\":{\"atom_tabs\":{\"type\":\"block\"}}},{\"query\":\"更改标签栏组件的类型为块状类型\",\"api\":{\"atom_tabs\":{\"type\":\"block\"}}},{\"query\":\"让标签栏组件以块状类型显示\",\"api\":{\"atom_tabs\":{\"type\":\"block\"}}},{\"query\":\"我希望标签栏组件是卡片类型的\",\"api\":{\"atom_tabs\":{\"type\":\"card\"}}},{\"query\":\"将标签栏组件设置为卡片类型\",\"api\":{\"atom_tabs\":{\"type\":\"card\"}}},{\"query\":\"如何把标签栏组件变成卡片类型\",\"api\":{\"atom_tabs\":{\"type\":\"card\"}}},{\"query\":\"更改标签栏组件的样式为卡片类型\",\"api\":{\"atom_tabs\":{\"type\":\"card\"}}},{\"query\":\"我需要标签栏组件显示为卡片类型\",\"api\":{\"atom_tabs\":{\"

type\":\"card\"}}},{\"query\":\"请让标签栏组件采用卡片类型布局\",\"api\":{\"atom_tabs\":{\"type\":\"card\"}}},{\"query\":\"切换标签栏组件的类型为卡片类型\",\"api\":{\"atom_tabs\":{\"type\":\"card\"}}},{\"query\":\"设置标签栏组件为文字类型\",\"api\":{\"atom_tabs\":{\"type\":\"text\"}}},{\"query\":\"我想要标签栏组件是文字类型的\",\"api\":{\"atom_tabs\":{\"type\":\"text\"}}},{\"query\":\"如何让标签栏组件呈现为文字类型\",\"api\":{\"atom_tabs\":{\"type\":\"text\"}}},{\"query\":\"更改标签栏组件的样式为文字类型\",\"api\":{\"atom_tabs\":{\"type\":\"text\"}}},{\"query\":\"我需要标签栏组件显示为文字类型\",\"api\":{\"atom_tabs\":{\"type\":\"text\"}}},{\"query\":\"让标签栏组件采用文字类型设计\",\"api\":{\"atom_tabs\":{\"type\":\"text\"}}},{\"query\":\"请将标签栏组件设置为文字类型\",\"api\":{\"atom_tabs\":{\"type\":\"text\"}}},{\"query\":\"我希望标签栏组件的类型是文字类型\",\"api\":{\"atom_tabs\":{\"type\":\"text\"}}},{\"query\":\"切换标签栏组件的类型为文字类型\",\"api\":{\"atom_tabs\":{\"type\":\"text\"}}}]"。

[0126] S105、将指令结果对存入数据集，并对数据集中的指令结果对进行统计，得到分布数据。

[0127] 在本实施例中，可以提前创建一个数据集，在该数据集中存放所生成的指令结果对。

[0128] 在生成指令结果对的过程中，或者在完成全部提示词对应的指令结果对生成之后，可以对数据集中的指令结果对进行统计，从而得到指令结果对的分布数据。

[0129] S106、在分布数据与生成比例匹配的情况下，将数据集确定为目标数据集。

[0130] 在明确分布数据与生成比例匹配的情况下，可以确定当前的数据集已满足需求，可将当前的数据集确定为目标数据集。

[0131] 具体的，该分布数据可以具体对对数据集中的不同类别的指令结果对进行数量统计之后，基于数量计算出的分布比例。在该分布比例与生成比例一致的情况下，确定分布数据与生产比例匹配。当然，也可以将生成比例与最小扩展记录数进行计算，确定出不同类型对应的最小记录总数。分布数据可以具体为不同类别分别对应的记录总数，在记录总数大于等于最小记录总数的情况下，确定分布数据与生成比例匹配。

[0132] 应用本申请实施例所提供的方法，在不同颗粒度下对应用程序编程接口进行拆分，得到多个应用程序编程子接口；对多个应用程序编程子接口进行类型统计，并基于统计结果，确定生成比例；获取多个应用程序编程子接口分别对应的提示词；将提示词输入至文本语言模型进行数据生成处理，得到指令结果对；其中，文本语言模型已学习了应用程序编程接口的技术文档和使用指南；将指令结果对存入数据集，并对数据集中的指令结果对进行统计，得到分布数据；在分布数据与生成比例匹配的情况下，将数据集确定为目标数据集。

[0133] 在不同颗粒度下对应用程序编程接口进行拆分，从而得到不同颗粒度的应用程序编程子接口。对这些应用程序编程子接口进行类型统计，并基于统计结果可以确定出生成比例。为了使得生成包括有与应用程序编程接口相关的指令结果对的数据集覆盖范围更全面，在生成提示词时，会获取这些应用程序编程子接口分别对应的提示词，然后，利用已经

学习了应用程序编程接口的技术文档和使用指南的文本语言模型进行数据生成处理,即可得到指令结果对。将生成的指令结果对写入数据集中,并对数据集中的指令结果对进行统计分析,在分布数据与生产比例匹配的情况下,即可确定完成数据集生成,得到目标数据集。

[0134] 本申请技术效果:在生成数据集中的指令结果对时,通过对应用程序编程接口进行拆分,可使得指令结果对可以覆盖应用程序编程接口的不同颗粒度指令,通过对应用程序编程子接口进行统计,可以确定出生成比例,在获得数据集中的指令结果对的分布数据后,当明确生成比例与分布数据匹配的情况下,确定得到目标数据集,可以使得最终生成的目标数据集的数据分布可控,为进一步训练生成式模型训练,提供全面且数据分布可定制化的训练样本。

[0135] 需要说明的是,基于上述实施例,本申请实施例还提供了相应的改进方案。在优选/改进实施例中涉及与上述实施例中相同步骤或相应步骤之间可相互参考,相应的有益效果也可相互参照,在本文的优选/改进实施例中不再一一赘述。

[0136] 在本申请中的一种具体实施方式中,在分布数据与生成比例不匹配的情况下,还包括:

[0137] 通过重新获取新的提示词对数据集进行补充,以使补充后的数据集对应的分布数据与生成比例匹配。

[0138] 也就是说,当发现分布数据与生成比例不匹配的情况下,表面当前的数据集中的数据分布情况还不满足所需的生成比例,此时,可以继续进行数据结果对的生成处理,并对数据集进行补充,从而使得补充后的数据集其数据分布于生产比例相匹配。

[0139] 具体的,对数据集进行补充的方式可以通过重新获取新的提示词,然后基于新的提示词对数据集进行补充。

[0140] 在本申请中的一种具体实施方式中,通过重新获取新的提示词对数据集进行补充,包括:

[0141] 步骤一、对比分布数据与生成比例,确定缺乏的指令结果对的目标类型;

[0142] 步骤二、将与目标类型匹配的应用程序编程子接口确定为目标应用程序编程子接口;

[0143] 步骤三、重新获取目标应用程序编程子接口对应的提示词;

[0144] 步骤四、将新获取的提示词输入至文本语言模型进行数据生成处理,得到补充的指令结果对;

[0145] 步骤五、将补充的指令结果对写入数据集中。

[0146] 为便于描述,下面将上述五个步骤结合起来进行说明。

[0147] 在本实施例中,可以通过比对分布数据与生成比例,确定出缺乏的指令结果对的目标类型。例如,假设类型包括A、B和C三种,当前的数据分布对应的比值是1:2:3,而生成比例是2:2:3,由此可见,A类型的指令结果对还缺数量,因此,可以将类型A确定为目标应用程序子接口,然后,重新获取类型A的提示词,然后借助于文本语言模型生成与该提示词对应的指令结果对。为便于区别,本实施例中,将基于重新获取的提示词所生成的指令结果对称之为补充的指令结果对。然后,将补充的指令结果对写入数据集中,从而对数据集进行补充。

[0148] 在补充了数据集之后,可以再次判断当前的数据集的分布数据与生成比例是否一致,如果一致,则可得到目标数据集;如果不一致,则可再次对数据集中的数据进行补充。

[0149] 需要注意的是,当数据集中的数据已满足最小扩展倍数对应的记录数之后,仅分布数据的比例与生成比例不匹配,还可以通过删除多余的数据结果对的方式,使得数据集的分布数据的比例与生成比例匹配。

[0150] 在本申请中的一种具体实施方式中,判断分布数据与生成比例是否匹配,包括:

[0151] 步骤一、利用类别配比得到各类别下的子类的子类配比;

[0152] 步骤二、从配置信息中读取各子类的最少拓展记录数;

[0153] 步骤三、利用子类配比和最少拓展记录数,确定子类别对应的最少记录条数;

[0154] 步骤四、基于最少记录条数计算满足类别配比的最小总记录数;

[0155] 步骤五、判断最少记录条数、最小总记录数与分布数据是否匹配;

[0156] 步骤六、如果是,则确定分布数据与生成比例匹配;

[0157] 步骤七、如果否,则确定分布数据与生成比例不匹配。

[0158] 为便于描述,下面将上述步骤结合起来进行说明。

[0159] 对于类别集A,A中包括多个子类别a,b,c,d,若类别配比为P,每个子类的配比为Pa,Pb,Pc,Pd。

[0160] 每个子类别的每个子API最少拓展记录数要求为lima,limb,limc,limd。即,即当前子类别的每一个子API最少需要模型生成多少个指令结果对。

[0161] 子API总数为N,各子类别的子API数量为Na,Nb,Nc,Nd。

[0162] 可通过执行以下步骤来进行匹配性判断:

[0163] 步骤1:计算满足最少条数要求的记录数。

[0164] 对于类别集A中每个子类别i,即,计算满足每个子类所需要的最少的记录数Mi,使用当前子类别i的每个子api最少拓展条数limi,乘以当前子类别i的子API的数量Ni,即$M_i = lim_i * N_i$。

[0165] 步骤2:计算满足配比要求P的最小总记录数,计算每个子类别i在满足自己配比要求前提下,最小的总记录数为S,$S = max(M_i/P_i|i \in A)$。

[0166] 步骤3:计算每个子类别需要的最终记录数

[0167] 对于每个类别$i \in A$,计算子类别i需要的最终记录数Si,即$S_i = max(ceil(S * P_i), M_i)$。

[0168] 得到每个子类别所需的最终记录数之后,在得到数据集的分布数据的情况下,可以判断当前各个子类别的统计数量与最终记录数是否一致,如果一致,则说明分布数据与生成比例是匹配的。如果超出可以进行删除,如果差数量,后续可以通过进行补充该子类别的方式进行数据补充。

[0169] 举例说明:若有属性集(即类别集,在此例中属性即对应类别) $A = a,b,c,d$,且原始数据集中各属性的记录数:$N_a = 100, N_b = 200, N_c = 300, N_d = 400$;

[0170] 配比要求$P = P_a = 0.3, P_b = 0.3, P_c = 0.2, P_d = 0.2$;

[0171] 每个子api的最小拓展记录数为$LIM = lim_a = 2, lim_b = 3, lim_c = 4, lim_d = 3$;

[0172] 步骤1:计算满足每一个子类别a,b,c,d最小拓展要求的记录数即:

[0173] $M_a = 2 * 100 = 200, M_b = 3 * 200 = 600, M_c = 4 * 300 = 1200,$

$M_d = 3 * 400 = 1200$ 。

[0174] 步骤2:计算满足配比要求的最小总记录数S为:

$S = max(200/0.3, 600/0.3, 1200/0.2, 1200/0.2) = 6000$ 。

[0175] 步骤3:计算每个子类别需要的最终记录数:

[0176] $S_a = max(ceil(6000 * 0.3), 200) = 1800$,$S_b = max(ceil(6000 * 0.3), 600) = 1800$,

$S_c = max(ceil(6000 * 0.2), 1200) = 1200$,$S_d = max(ceil(6000 * 0.2), 1200) = 1200$ 。

[0177] 最终结果:

[0178] 属性 a 需要 1800 条记录,属性 b 需要 1800条记录,属性 c 需要 1200条记录,属性 d 需要 1200条记录 。

[0179] 验证:每个属性的最终记录数至少是原始记录数的指定倍数:a: 1800 > 2 * 100,b: 1800 > 3 * 200,c: 1200 = 4 * 300,d: 1200 > 3 * 400 。

[0180] 最终配比:a: 1800 / 6000 = 0.3,b: 1800 / 6000 = 0.3,c: 1200 / 6000 = 0.2,d: 1200 / 6000 = 0.2 。

[0181] 经验证可见,每个属性的最终记录数的确定方式,确保了每个属性至少达到了其指定的最小扩展倍数,同时也满足了给定的配比要求。

[0182] 在实际应用中,还可对生成的指令结果对进行严格的筛选,去除生成错误或不符合规范的数据。从而确保最终输出的数据具有高质量和一致性。

[0183] 在数据生成完成后,可以重新计算各API的数据比例,并与之前设定的配比要求进行比较。筛选出数量不足的API,以便进行进一步的补充处理。

[0184] 对于筛选出的数量不足的API,可以通过定制化提示词,将已有的API生成结果作为输入,指导模型生成与已有数据不同的新数据。如此,有助于进一步丰富数据的多样性,避免生成相似或重复的数据。

[0185] 此外,该可以不断生成和优化数据,直到所有API的数量和多样性要求都得到满足。这个迭代过程确保最终的数据集具备高质量、多样性和覆盖率。

[0186] 为便于本领域技术人员更好地理解和实施本申请实施例所提供的数据生成方法,下面结合相关技术方案及具体应用场景为例对数据生成方法进行详细说明。

[0187] 面向API的数据合成方法主要分为以下三类:

[0188] 1.手动数据构造:通过人工方式创建数据,或通过抓取历史数据获取用户手动生成的数据。

[0189] 2.程序自动化生成:通过程序自动化手段,为每个API设定策略,自动生成可能的API结果,然后反向标注对应的输入指令。

[0190] 3.基于API文档的生成:通过构建API文档,并输入每一条API说明,利用已有的开源大模型生成可能的输入输出案例,以此构造训练数据。

[0191] 其中,人工手动构造数据或通过抓取历史数据来生成,存在以下缺点:

[0192] (1)、校验成本高:每一条数据都需要经过手动校验,确保其正确性,这导致耗时耗力。

[0193] (2)、数据生成效率低:依赖大量人工操作,生成数据的效率较低,难以满足大规模数据需求。

[0194] (3)、难以控制数据分布和数量:在数据规模和多样性方面,难以通过人工方式进

行精细化控制,容易出现数据偏差或不足。

[0195] 对于程序自动化生成数据,通过编写程序,针对每个API设定策略,自动枚举生成API的输出结果,再根据输出结果反向标注输入指令。存在缺点:

[0196] (1)、开发成本高:依赖于每个API的详细生成策略,如颜色、字体、预设值等,需要耗费大量的开发时间和资源。

[0197] (2)、拓展性差:新增API、数据类型或返回值时,需要重新设计和实现生成策略,导致维护成本较高。

[0198] (3)、灵活性有限:自动生成的数据往往受到规则约束,难以覆盖到API可能的全部使用场景。

[0199] 对于基于API文档与大模型的自动生成,通过提供API文档,使用开源大模型生成API的输入输出示例数据。存在缺点:

[0200] (1)、指令覆盖率低:大模型生成的输入输出案例往往无法全面覆盖API的全部使用场景,容易遗漏一些重要的用例。

[0201] (2)、数据分布不均衡:生成的数据在分布上可能存在偏差,无法做到数据的全面覆盖。

[0202] (3)、极端值处理不足:对于边界值和异常情况的考虑较少,难以确保数据的全面性。

[0203] (4)、泛化性不强:生成的数据指令缺乏灵活性,难以适应多种API场景下的广泛应用。

[0204] 从上述实施例可知,本申请所提供的技术方案旨在通过引入定义API的特征,并基于这些特征自动化地拆分多个小型子API,以生成大模型训练的专用业务数据。这种方法不仅有助于控制数据的分布比例,还能显著增加生成指令的多样性和覆盖范围。使得可以用一套通用的方案,为多业务方自动化批量可控的生成训练数据。

[0205] 具体的,本申请定义了一套标准的API规范,包括类型、层级、依赖关系、可枚举项和词汇表。不仅提供了清晰的框架,还为后续的自动化拆分和数据生成提供了基础。

[0206] 基于定义的特征,将API拆分为不同颗粒度,并进一步细化和重构子功能模块。这个过程允许更加灵活的API处理方式。

[0207] 统计不同类型和颗粒度API的数量分布,并据此计算数据生成比例,从而控制数据集的平衡性和质量。

[0208] 面向API的分布可控的通用解决框架:通过定义API特征并自动化拆分,实现对大模型训练数据的定制化和多样化生成。这种方法允许使用统一的框架为不同业务场景自动生成具有特定参数和配置的业务数据,同时确保数据的平衡性和覆盖率,提高了数据生成的效率和质量。

[0209] 基于本申请生成的目标数据集,对生成式模型进行训练之后,可以达到以下效果:

[0210] 用户需要通过自然语言来指定使用对应的组件(即api名字),并设置组件的各种参数(api参数),达到快速便捷使用,增加创建效率。

[0211] 前端业务搭建场景中,面向产品经理,可以快速搭建页面,尝试不同方案的对比:例如我需要标签栏组件显示为文字类型,我想让标签栏的第三个标签变为不可用状态。

[0212] 组件调整的场景中,面向用户,可以快速指定效果,无需在大量API中查找所需调

整项:用户需要通过自然语言来设置组件的各种参数:例如字体大小调整为四号,开启脚注,使用季节性图表。

[0213] 金融领域行情图配置中,面向产品经理和设计师,可以快速对比不同的方案的差异,快速构建不同市场的样式和功能:例如蜡烛图外框的颜色使用深红色,十字光标变为蓝色。

[0214] 请参考图2,本申请具体实施操作步骤包括:

[0215] 1、构建标准的API规范:首先定义了一套标准的API规范,包括类型、层级、依赖关系、可枚举项以及可拓展的词汇表。这一步为后续的拆分和数据生成提供了明确的框架和依据。

[0216] 2、API拆分与重构:根据定义的特征,将API拆分为不同的颗粒度。这意味着不仅考虑整体API的功能,还针对其中的子功能模块进行进一步的细化和重构。这有助于更灵活地处理API,并为后续的数据生成奠定基础。

[0217] 3、统计数量分布:对不同颗粒度、不同类型的API进行数量统计,形成一个全面的数据分布图。这一步的目的是确保对每一类API的数量和分布有清晰的掌握。

[0218] 4、计算数据比例:根据上述分布数据,计算出各类API在数据生成过程中所需的比例。这一步骤帮助更精确地控制生成数据的数量,以保证数据集的平衡性。

[0219] 5、定制化提示词构建:为不同颗粒度的API设计定制化的提示词,这些提示词会结合具体的数量需求,以确保生成的数据能够达到预期的效果。

[0220] 6、生成数据:利用GPT生成模型,根据设计的提示词生成数据。这一步利用了生成式AI的强大能力,以大规模生成多样化的数据。

[0221] 7、数据处理与错误筛除:对生成的数据进行后处理,筛选并去除错误数据。这个步骤确保了最终数据的质量和准确性。

[0222] 8、数据比例计算与API筛选:再次计算数据比例,筛选出数量不足的API。这一步的目的是确保每个API都能达到所需的数量和覆盖率。

[0223] 9、数据继续生成:对于在步骤8中筛选出的API,结合步骤5中构建的定制化提示词,并将已有的API生成结果纳入上下文,引导模型生成与已有数据不同的新数据。这一步有助于补齐数据配比,并保证丰富数据的多样性,避免重复和单一化。

[0224] 10、迭代优化:重复执行第6步到第9步,持续优化和生成数据,直到满足所有API的数量和多样性要求为止。

[0225] 相应于上面的方法实施例,本申请实施例还提供了一种数据生成装置,下文描述的数据生成装置与上文描述的数据生成方法可相互对应参照。

[0226] 参见图3所示,该装置包括以下模块:

[0227] 拆分模块101,用于在不同颗粒度下对应用程序编程接口进行拆分,得到多个应用程序编程子接口;

[0228] 生成比例统计模块102,用于对多个应用程序编程子接口进行类型统计,并基于统计结果,确定生成比例;

[0229] 提示处理模块103,用于获取多个应用程序编程子接口分别对应的提示词;

[0230] 指令结果对生成模块104,用于将提示词输入至文本语言模型进行数据生成处理,得到指令结果对;其中,文本语言模型已学习了应用程序编程接口的技术文档和使用指南;

[0231] 分析模块105,用于将指令结果对存入数据集,并对数据集中的指令结果对进行统计,得到分布数据;

[0232] 数据生成模块106,用于在分布数据与生成比例匹配的情况下,将数据集确定为目标数据集。

[0233] 应用本申请实施例所提供的装置,在不同颗粒度下对应用程序编程接口进行拆分,得到多个应用程序编程子接口;对多个应用程序编程子接口进行类型统计,并基于统计结果,确定生成比例;获取多个应用程序编程子接口分别对应的提示词;将提示词输入至文本语言模型进行数据生成处理,得到指令结果对;其中,文本语言模型已学习了应用程序编程接口的技术文档和使用指南;将指令结果对存入数据集,并对数据集中的指令结果对进行统计,得到分布数据;在分布数据与生成比例匹配的情况下,将数据集确定为目标数据集。

[0234] 在不同颗粒度下对应用程序编程接口进行拆分,从而得到不同颗粒度的应用程序编程子接口。对这些应用程序编程子接口进行类型统计,并基于统计结果可以确定出生成比例。为了使得生成包括有与应用程序编程接口相关的指令结果对的数据集覆盖范围更全面,在生成提示词时,会获取这些应用程序编程子接口分别对应的提示词,然后,利用已经学习了应用程序编程接口的技术文档和使用指南的文本语言模型进行数据生成处理,即可得到指令结果对。将生成的指令结果对写入数据集中,并对数据集中的指令结果对进行统计分析,在分布数据与生产比例匹配的情况下,即可确定完成数据集生成,得到目标数据集。

[0235] 本申请技术效果:在生成数据集中的指令结果对时,通过对应用程序编程接口进行拆分,可使得指令结果对可以覆盖应用程序编程接口的不同颗粒度指令,通过对应用程序编程子接口进行统计,可以确定出生成比例,在获得数据集中的指令结果对的分布数据后,当明确生成比例与分布数据匹配的情况下,确定得到目标数据集,可以使得最终生成的目标数据集的数据分布可控,为进一步训练生成式模型训练,提供全面且数据分布可定制化的训练样本。

[0236] 在本申请的一种具体实施方式中,拆分模块,具体用于获取应用程序编程接口的声明信息;

[0237] 基于声明信息,确定不同的颗粒度;

[0238] 在不同颗粒度下,对程序编程接口进行拆分,得到多个应用程序编程子接口。

[0239] 在本申请的一种具体实施方式中,还包括:

[0240] 补充模块,用于在分布数据与生成比例不匹配的情况下,通过重新获取新的提示词对数据集进行补充,以使补充后的数据集对应的分布数据与生成比例匹配。

[0241] 在本申请的一种具体实施方式中,补充模块,具体用于对比分布数据与生成比例,确定缺乏的指令结果对的目标类型;

[0242] 将与目标类型匹配的应用程序编程子接口确定为目标应用程序编程子接口;

[0243] 重新获取目标应用程序编程子接口对应的提示词;

[0244] 将新获取的提示词输入至文本语言模型进行数据生成处理,得到补充的指令结果对;

[0245] 将补充的指令结果对写入数据集中。

[0246] 在本申请的一种具体实施方式中,生成比例统计模块,具体用于在可视化界面输出统计结果;

[0247] 获取输入的配置信息;

[0248] 利用配置信息,确定生成比例。

[0249] 在本申请的一种具体实施方式中,生成比例统计模块,具体用于从配置信息中读取中各类别的类别配比,基于类别配比确定生成比例。

[0250] 在本申请的一种具体实施方式中,匹配判断模块,用于利用类别配比得到各类别下的子类的子类配比;

[0251] 从配置信息中读取各子类的最少拓展记录数;

[0252] 利用子类配比和最少拓展记录数,确定子类别对应的最少记录条数;

[0253] 基于最少记录条数计算满足类别配比的最小总记录数;

[0254] 判断最少记录条数、最小总记录数与分布数据是否匹配;

[0255] 如果是,则确定分布数据与生成比例匹配;

[0256] 如果否,则确定分布数据与生成比例不匹配。

[0257] 相应于上面的方法实施例,本申请实施例还提供了一种电子设备,下文描述的一种电子设备与上文描述的一种数据生成方法可相互对应参照。

[0258] 参见图4所示,该电子设备包括:

[0259] 存储器332,用于存储计算机程序;

[0260] 处理器322,用于执行计算机程序时实现上述方法实施例的数据生成方法的步骤。

[0261] 具体的,请参考图5,图5为本实施例提供的一种电子设备的具体结构示意图,该电子设备可因配置或性能不同而产生比较大的差异,可以包括一个或一个以上处理器(central processing units,CPU)(例如,一个或一个以上处理器)和存储器332,存储器332存储有一个或一个以上的计算机程序342或数据344。其中,存储器332可以是短暂存储或持久存储。存储在存储器332的程序可以包括一个或一个以上模块(图示没标出),每个模块可以包括对数据处理设备中的一系列指令操作。更进一步地,处理器322可以设置为与存储器332通信,在电子设备301上执行存储器332中的一系列指令操作。

[0262] 电子设备301还可以包括一个或一个以上电源326,一个或一个以上有线或无线网络接口350,一个或一个以上输入输出接口358,和/或,一个或一个以上操作系统341。

[0263] 上文所描述的数据生成方法中的步骤可以由电子设备的结构实现。

[0264] 相应于上面的方法实施例,本申请实施例还提供了一种可读存储介质,下文描述的一种可读存储介质与上文描述的一种数据生成方法可相互对应参照。

[0265] 一种可读存储介质,可读存储介质上存储有计算机程序,计算机程序被处理器执行时实现上述方法实施例的数据生成方法的步骤。

[0266] 该可读存储介质具体可以为U盘、移动硬盘、只读存储器(Read-Only Memory,ROM)、随机存取存储器(Random Access Memory,RAM)、磁碟或者光盘等各种可存储程序代码的可读存储介质。

[0267] 本说明书中各个实施例采用递进的方式描述,每个实施例重点说明的都是与其它实施例的不同之处,各个实施例之间相同或相似部分互相参见即可。对于实施例公开的装置而言,由于其与实施例公开的方法相对应,所以描述的比较简单,相关之处参见方法部分

说明即可。

[0268] 本领域技术人员还可以进一步意识到,结合本文中所公开的实施例描述的各示例的单元及算法步骤,能够以电子硬件、计算机软件或者二者的结合来实现,为了清楚地说明硬件和软件的可互换性,在上述说明中已经按照功能一般性地描述了各示例的组成及步骤。这些功能究竟以硬件还是软件的方式来执行,取决于技术方案的特定应用和设计约束条件。本领域技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能,但是这种实现不应该认为超出本申请的范围。

[0269] 结合本文中所公开的实施例描述的方法或算法的步骤可以直接用硬件、处理器执行的软件模块,或者二者的结合来实施。软件模块可以置于随机存储器(RAM)、内存、只读存储器(ROM)、电可编程ROM、电可擦除可编程ROM、寄存器、硬盘、可移动磁盘、CD-ROM、或技术领域内所公知的任意其它形式的存储介质中。

[0270] 最后,还需要说明的是,在本文中,诸如第一和第二等之类的关系属于仅仅用来将一个实体或者操作与另一个实体或者操作区分开来,而不一定要求或者暗示这些实体或操作之间存在任何这种实际的关系或者顺序。而且,术语包括、包含或者其他任何变体意在涵盖非排他性的包含,从而使得包括一系列要素的过程、方法、物品或者设备不仅包括那些要素,而且还包括没有明确列出的其他要素,或者是还包括为这种过程、方法、物品或者设备所固有的要素。

[0271] 本文中应用了具体个例对本申请的原理及实施方式进行了阐述,以上实施例的说明只是用于帮助理解本申请的方法及其核心思想;同时,对于本领域的一般技术人员,依据本申请的思想,在具体实施方式及应用范围上均会有改变之处,综上,本说明书内容不应理解为对本申请的限制。

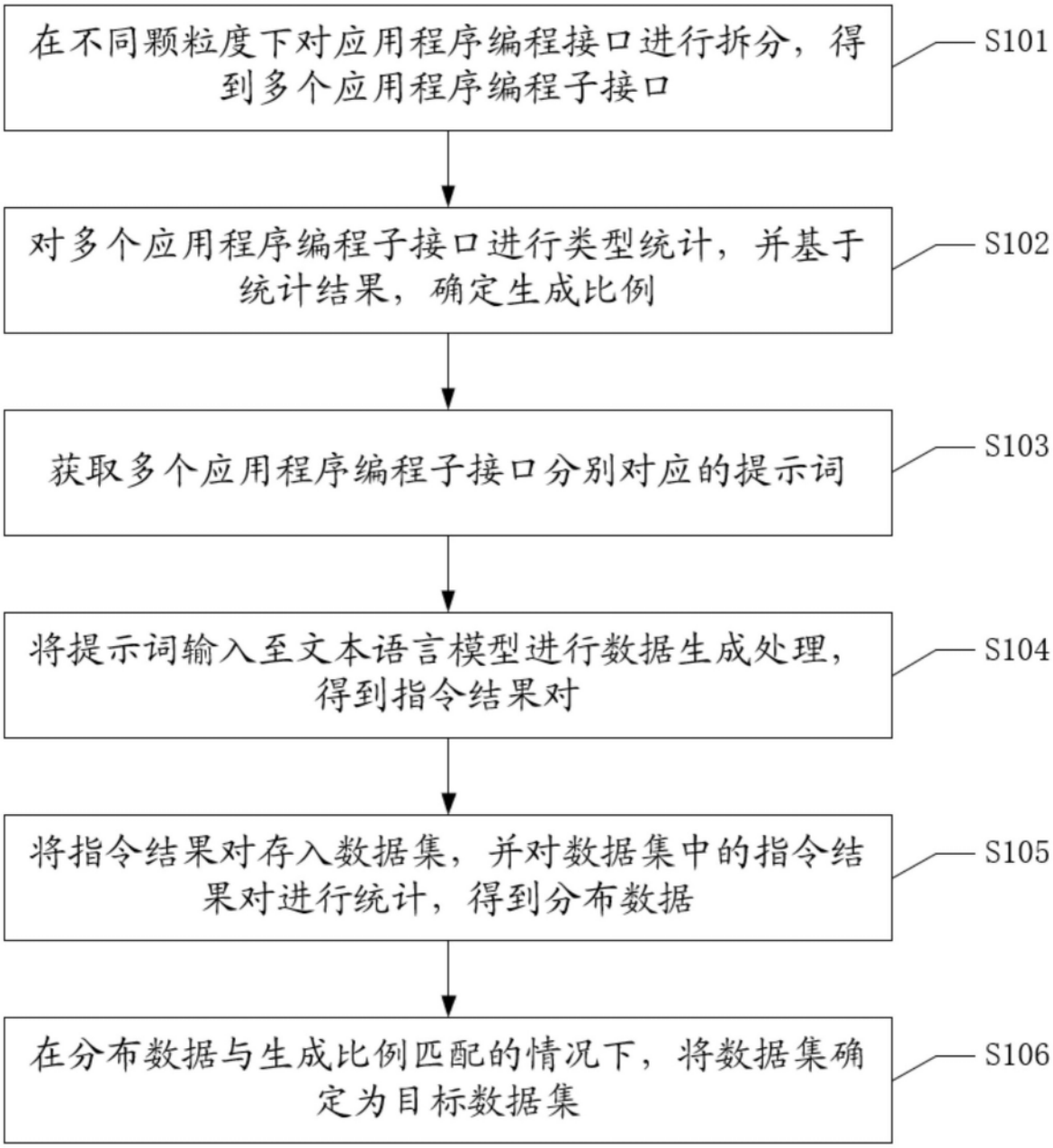

在不同颗粒度下对应用程序编程接口进行拆分，得到多个应用程序编程子接口
S101
对多个应用程序编程子接口进行类型统计，并基于统计结果，确定生成比例
S102
获取多个应用程序编程子接口分别对应的提示词
S103
将提示词输入至文本语言模型进行数据生成处理，得到指令结果对
S104
将指令结果对存入数据集，并对数据集中的指令结果对进行统计，得到分布数据
S105
在分布数据与生成比例匹配的情况下，将数据集确定为目标数据集
S106

图 1

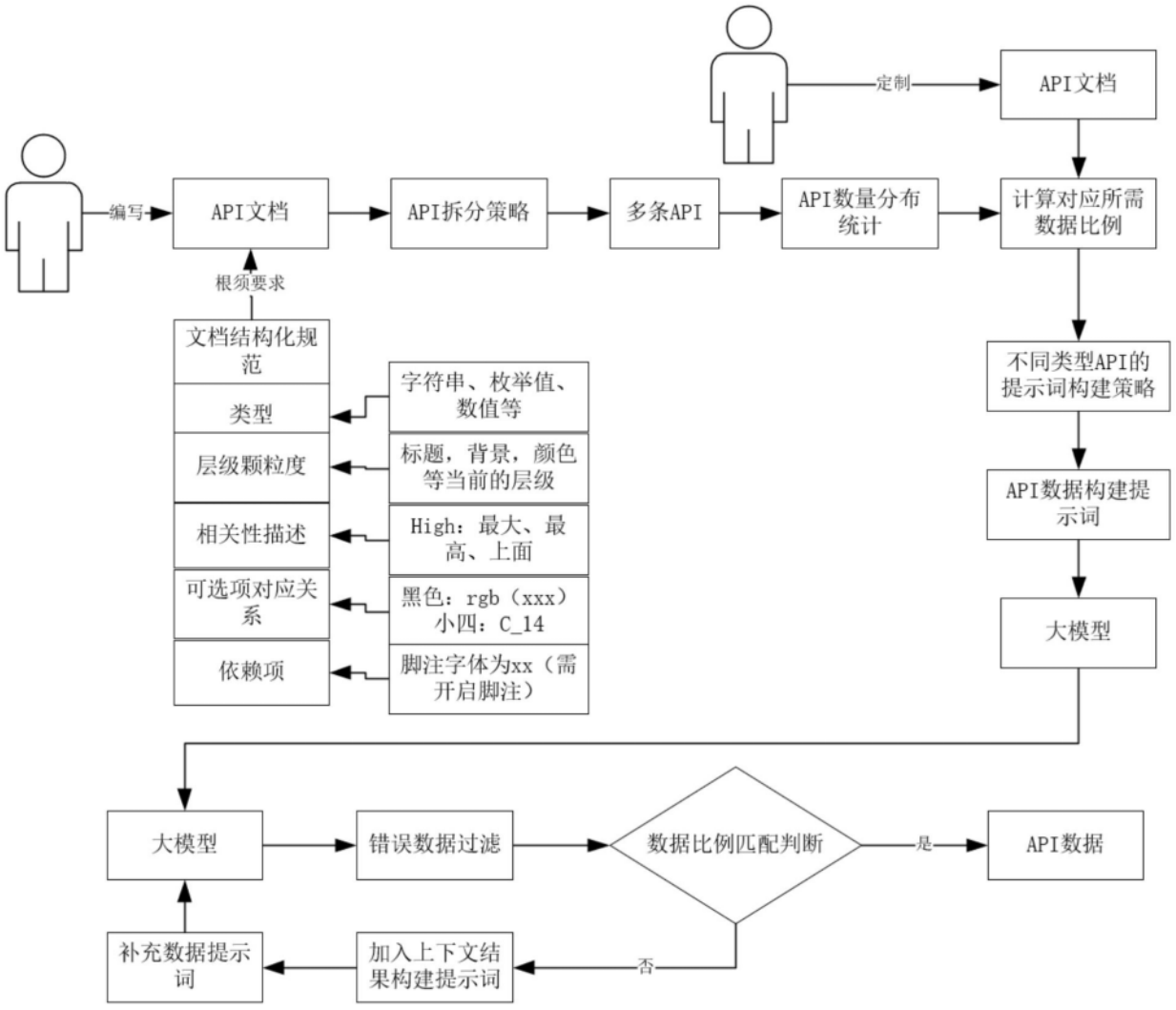

定制
API文档
编写
API文档
API拆分策略
多条API
API数量分布统计
计算对应所需数据比例
根须要求
文档结构化规范
类型
层级颗粒度
相关性描述
可选项对应关系
依赖项
字符串、枚举值、数值等
标题，背景，颜色等当前的层级
High：最大、最高、上面
黑色：rgb（xxx）小四：C_14
脚注字体为xx（需开启脚注）
不同类型API的提示词构建策略
API数据构建提示词
大模型
大模型
错误数据过滤
数据比例匹配判断
是
API数据
否
加入上下文结果构建提示词
补充数据提示词

图 2

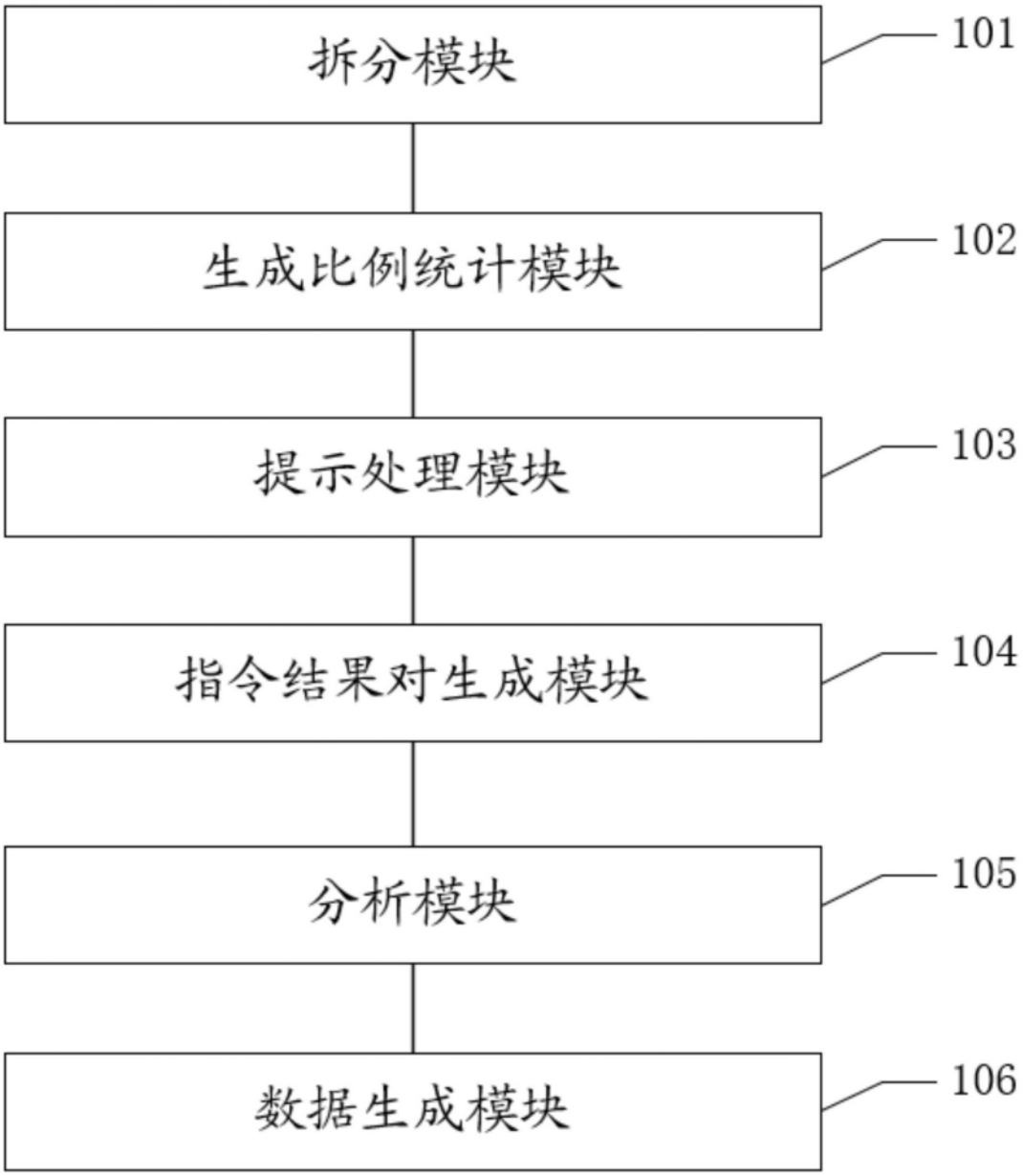

拆分模块
101
生成比例统计模块
102
提示处理模块
103
指令结果对生成模块
104
分析模块
105
数据生成模块
106

图 3

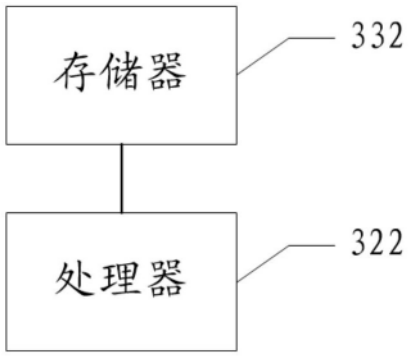

存储器
332
处理器
322

图 4

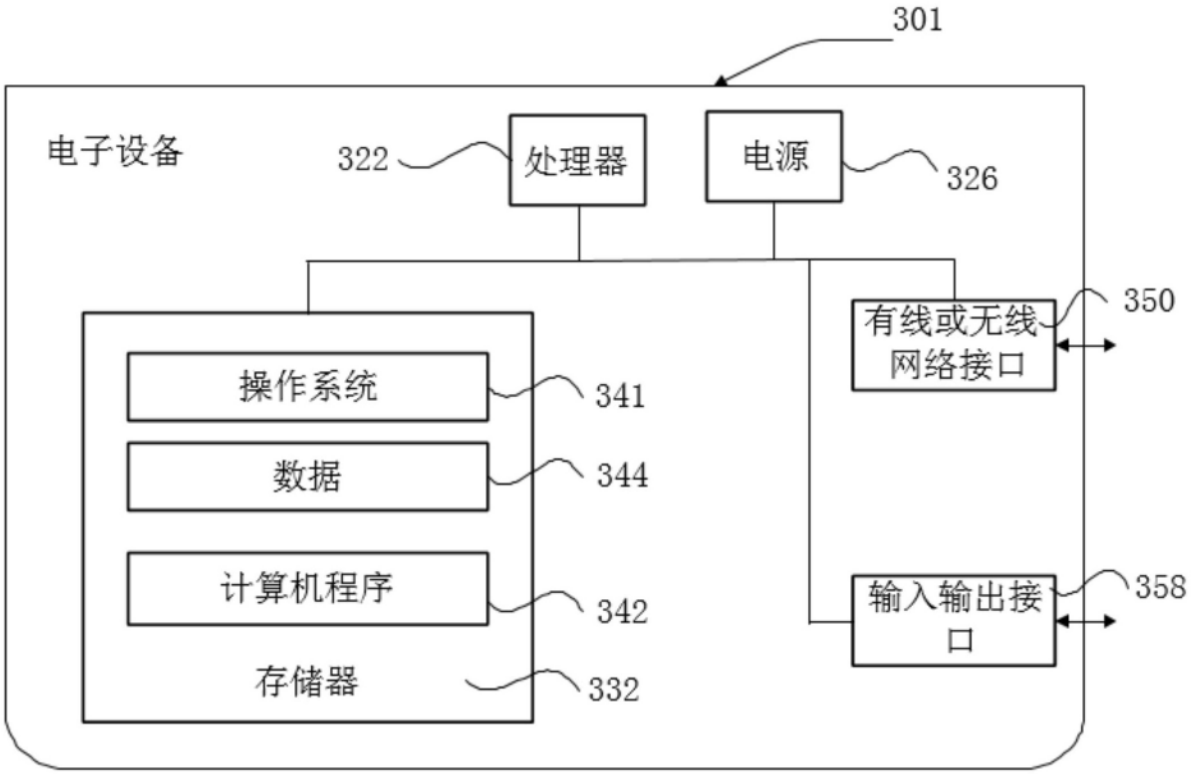

301
电子设备
322
处理器
电源
326
有线或无线网络接口
350
操作系统
341
数据
344
计算机程序
342
存储器
332
输入输出接口
358

图 5