



(12) 发明专利

(10) 授权公告号 CN 116822632 B

(45) 授权公告日 2024.01.05

(21) 申请号 202311085639.X

(22) 申请日 2023.08.28

(65) 同一申请的已公布的文献号

申请公布号 CN 116822632 A

(43) 申请公布日 2023.09.29

(73) 专利权人 腾讯科技(深圳)有限公司

地址 518044 广东省深圳市南山区高新区

科技中一路腾讯大厦35层

(72) 发明人 孟朋 田恒锋

(74) 专利代理机构 北京同达信恒知识产权代理

有限公司 11291

专利代理师 彭燕

(51) Int. Cl.

G06F 40/205 (2020.01)

G06N 5/04 (2023.01)

(56) 对比文件

CN 114065771 A, 2022.02.18

CN 114329148 A, 2022.04.12

CN 115563976 A, 2023.01.03

US 2023082485 A1, 2023.03.16

US 2023222285 A1, 2023.07.13

WO 2021169400 A1, 2021.09.02

CN 114048289 A, 2022.02.15

CN 110263324 A, 2019.09.20

CN 114676234 A, 2022.06.28

Xuan Ouyang.ERNIE-M: Enhanced

Multilingual Representation by Aligning

Cross-lingual Semantics with Monolingual

Corpora.arXiv.2021,1-12.

审查员 梁滔

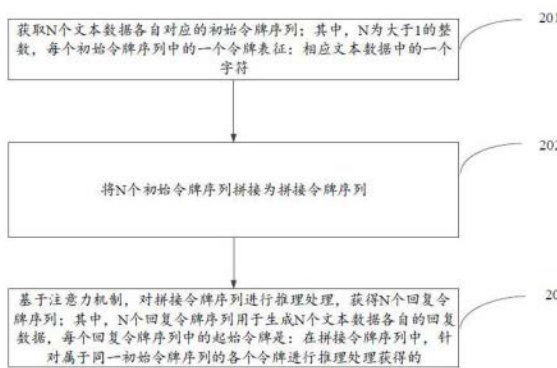
权利要求书2页 说明书18页 附图9页

(54) 发明名称

一种文本数据的推理方法、装置、存储介质和电子设备

(57) 摘要

本申请公开一种文本数据的推理方法、装置、存储介质和电子设备,本申请可应用于云技术、人工智能、智慧交通、辅助驾驶等各种场景,该方法中,获取N个文本数据各自对应的初始令牌序列,N为大于1的整数,每个初始令牌序列中的一个令牌表征:相应文本数据中的一个字符;将N个初始令牌序列拼接为拼接令牌序列;基于注意力机制,对拼接令牌序列进行推理处理,获得N个回复令牌序列,N个回复令牌序列用于生成N个文本数据各自的回复数据,每个回复令牌序列中的起始令牌是:在拼接令牌序列中,针对属于同一初始令牌序列的各个令牌进行推理处理获得的。本方案可降低推理多条文本数据所消耗的GPU资源,提升设备对多条文本数据的推理性能。



1. 一种文本数据的推理方法,其特征在于,所述方法包括:

获取N个文本数据各自对应的初始令牌序列;其中,N为大于1的整数,每个初始令牌序列中的一个令牌表征:相应文本数据中的一个字符;

将N个初始令牌序列拼接为拼接令牌序列;

将所述拼接令牌序列输入目标模型,获得所述目标模型根据注意力机制,针对所述拼接令牌序列中的各个令牌进行推理,输出的候选令牌序列,所述候选令牌序列中的每个令牌为:所述拼接令牌序列中一个令牌的推理结果;

从所述候选令牌序列中,确定所述N个初始令牌序列各自终止令牌所对应的推理结果,获得N个选定令牌;

基于所述目标模型,针对所述N个选定令牌进行迭代推理,获得所述目标模型输出的N个回复令牌序列;其中,所述N个选定令牌分别为所述N个回复令牌序列的起始令牌,所述N个回复令牌序列用于生成所述N个文本数据各自的回复数据。

2. 如权利要求1所述的方法,其特征在于,所述候选令牌序列,是通过如下方式推理获得的:

获取预设注意力矩阵中的Q行元素;其中,Q为所述拼接令牌序列中的令牌总数,所述Q行元素中的每行元素表征:在推理过程中,对所述拼接令牌序列中各个令牌具有的不同的关注程度;

基于所述Q行元素中的各行元素,分别对所述拼接令牌序列中的、属于同一初始令牌序列的各个令牌进行推理处理,获得所述拼接令牌序列中各个令牌各自对应的推理令牌;

获得由各个推理令牌拼接生成的候选令牌序列。

3. 如权利要求1所述的方法,其特征在于,所述N个回复令牌序列,是通过如下方式推理获得的:

将所述N个选定令牌分别作为N个回复令牌序列的起始令牌;

获取预设注意力矩阵中的 $P \times N$ 行元素;其中,P为正整数;

针对所述 $P \times N$ 行元素中的每N行元素,依次执行以下操作:

基于N行元素,分别对当前获得的N个选定令牌进行推理处理,获得N个推理令牌;其中,所述N行元素中的每行元素表征:在推理过程中,分别对当前获得的N个选定令牌具有不同的关注程度;

将所述N个推理令牌分别拼接在所述N个回复令牌序列的尾部,并将所述N个推理令牌作为下一次获得的N个选定令牌;

直到执行P次操作,获得N个回复令牌序列。

4. 如权利要求1~3任一项所述的方法,其特征在于,所述获取N个文本数据各自对应的初始令牌序列,包括:

获取待回复的N个文本数据;

分别针对N个文本数据执行以下操作:将一个文本数据中的各个字符,分别编码为相应的令牌,获得所述一个文本数据对应的初始令牌序列。

5. 如权利要求1~3任一项所述的方法,其特征在于,所述将N个初始令牌序列拼接为拼接令牌序列,包括:

获取基于所述N个初始令牌序列各自的终止令牌,拼接生成的第一令牌序列;

以及,获取基于所述N个初始令牌序列各自除终止令牌以外的令牌,拼接生成的第二令牌序列;

将所述第一令牌序列拼接在所述第二令牌序列的末尾,获得拼接令牌序列。

6.如权利要求5所述的方法,其特征在于,所述第一令牌序列和所述第二令牌序列:分别是按照预设的拼接次序拼接生成的,所述拼接次序表征:所述N个文本数据的处理次序;

则所述获取基于所述N个初始令牌序列各自的终止令牌,拼接生成的第一令牌序列,包括:

基于所述N个文本数据的处理次序,依次对所述N个初始令牌序列各自的终止令牌进行拼接处理,获得第一令牌序列;

则所述获取基于所述N个初始令牌序列各自除终止令牌以外的令牌,拼接生成的第二令牌序列,包括:

基于所述N个文本数据的处理次序,依次将所述N个初始令牌序列各自除终止令牌以外的令牌拼接为第二令牌序列。

7.如权利要求6所述的方法,其特征在于,所述N个文本数据的处理次序,是采用以下任意一种方式确定的:

将所述N个文本数据的获取次序,作为所述N个文本数据的处理次序;

将所述N个文本数据各自对应的时间戳的时间先后次序,作为所述N个文本数据的处理次序;

将所述N个文本数据各自对应的优先级的级别高低次序,作为所述N个文本数据的处理次序。

8.一种文本数据的推理装置,其特征在于,所述装置包括:

获取单元,获取N个文本数据各自对应的初始令牌序列;其中,N为大于1的整数,每个初始令牌序列中的一个令牌表征:相应文本数据中的一个字符;

拼接单元,将N个初始令牌序列拼接为拼接令牌序列;

推理单元,将所述拼接令牌序列输入目标模型,获得所述目标模型根据注意力机制,针对所述拼接令牌序列中的各个令牌进行推理,输出的候选令牌序列,所述候选令牌序列中的每个令牌为:所述拼接令牌序列中一个令牌的推理结果;从所述候选令牌序列中,确定所述N个初始令牌序列各自终止令牌所对应的推理结果,获得N个选定令牌;基于所述目标模型,针对所述N个选定令牌进行迭代推理,获得所述目标模型输出的N个回复令牌序列;其中,所述N个选定令牌分别为所述N个回复令牌序列的起始令牌,所述N个回复令牌序列用于生成所述N个文本数据各自的回复数据。

9.一种计算机设备,包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序,其特征在于,

所述处理器执行所述计算机程序时实现权利要求1至7任一项所述方法的步骤。

10.一种计算机存储介质,其上存储有计算机程序指令,其特征在于,

所述计算机程序指令被处理器执行时实现权利要求1至7任一项所述方法的步骤。

一种文本数据的推理方法、装置、存储介质和电子设备

技术领域

[0001] 本申请涉及计算机技术领域,尤其涉及一种文本数据的推理方法、装置、存储介质和电子设备。

背景技术

[0002] 随着高新技术的快速发展,衍生出具有强大推理能力的大语言模型(Large Language Model,LLM)。例如,计算机设备基于大语言模型的推理能力,通过调用大语言模型内海量的模型参数,可以针对“等闲识得东风面的下一句”进行推理,获得“万紫千红总是春”的推理结果。

[0003] 在上述推理过程中,计算机设备实际所需调用的模型参数量是数千亿级的,例如,在生成式预训练3(Generative Pre-Trained Transformer 3,GPT3)模型的推理过程中,所涉及的模型参数量便达到了1750亿(175B)。这也就导致计算机设备在进行数据推理过程中,需要消耗大量的图形处理器(Graphics Processing Unit,GPU)资源。

[0004] 相关技术中,为避免每次只推理一个文本数据而导致GPU资源利用效率低下,提出将多个文本数据作为同一批次进行数据推理处理的方式;进一步,为推理过程能够适配现有的深度学习模型框架,还提出一种字符补齐方式,也即在相对较短的文本数据中补齐填充(padding)字符,以使参与拼接处理的每个文本数据的长度保持一致。

[0005] 然而,上述通过字符补齐方式所添加的填充字符也将参与实际的推理运算,这样,会造成大量无效的计算,以及大量无效GPU资源的消耗。

发明内容

[0006] 本申请提供一种多文本数据的推理方法、装置、存储介质和电子设备,用以降低推理多条文本数据所消耗的GPU资源,提升计算机设备针对多条文本数据的推理性能。

[0007] 第一方面,本申请提供了一种文本数据的推理方法,所述方法包括:

[0008] 获取N个文本数据各自对应的初始令牌序列;其中,N为大于1的整数,每个初始令牌序列中的一个令牌表征相应文本数据中的一个字符;

[0009] 将所述拼接令牌序列输入目标模型,获得所述目标模型根据注意力机制,针对所述拼接令牌序列中的各个令牌进行推理,输出的候选令牌序列,所述候选令牌序列中的每个令牌为:所述拼接令牌序列中一个令牌的推理结果;

[0010] 从所述候选令牌序列中,确定所述N个初始令牌序列各自终止令牌所对应的推理结果,获得N个选定令牌;

[0011] 基于所述目标模型,针对所述N个选定令牌进行迭代推理,获得所述目标模型输出的N个回复令牌序列;其中,所述N个选定令牌分别为所述N个回复令牌序列的起始令牌,所述N个回复令牌序列用于生成所述N个文本数据各自的回复数据。

[0012] 第二方面,本申请提供了一种文本数据的推理装置,所述装置包括:

[0013] 获取单元,获取N个文本数据各自对应的初始令牌序列;其中,N为大于1的整数,每

个初始令牌序列中的一个令牌表征相应文本数据中的一个字符；

[0014] 拼接单元,将N个初始令牌序列拼接为拼接令牌序列；

[0015] 推理单元,将所述拼接令牌序列输入目标模型,获得所述目标模型根据注意力机制,针对所述拼接令牌序列中的各个令牌进行推理,输出的候选令牌序列,所述候选令牌序列中的每个令牌为:所述拼接令牌序列中一个令牌的推理结果;从所述候选令牌序列中,确定所述N个初始令牌序列各自终止令牌所对应的推理结果,获得N个选定令牌;基于所述目标模型,针对所述N个选定令牌进行迭代推理,获得所述目标模型输出的N个回复令牌序列;其中,所述N个选定令牌分别为所述N个回复令牌序列的起始令牌,所述N个回复令牌序列用于生成所述N个文本数据各自的回复数据。

[0016] 可选的,所述候选令牌序列,是通过如下方式推理获得的,则所述推理单元,还用于:

[0017] 获取预设注意力矩阵中的Q行元素;其中,Q为所述拼接令牌序列中的令牌总数,所述Q行元素中的每行元素表征:在推理过程中,对所述拼接令牌序列中各个令牌具有的不同关注程度;

[0018] 基于所述Q行元素中的各行元素,分别对所述拼接令牌序列中的、属于同一初始令牌序列的各个令牌进行推理处理,获得所述拼接令牌序列中各个令牌各自对应的推理令牌;

[0019] 获得由各个推理令牌拼接生成的候选令牌序列。

[0020] 可选的,所述N个回复令牌序列,是通过如下方式推理获得的,则所述推理单元,还用于:

[0021] 将所述N个选定令牌分别作为N个回复令牌序列的起始令牌;

[0022] 获取预设注意力矩阵中的 $P \times N$ 行元素;其中,P为正整数;

[0023] 针对所述 $P \times N$ 行元素中的每N行元素,依次执行以下操作:

[0024] 基于N行元素,分别对当前获得的N个选定令牌进行推理处理,获得N个推理令牌;其中,所述N行元素中的每行元素表征:在推理过程中,分别对当前获得的N个选定令牌具有不同的关注程度;

[0025] 将所述N个推理令牌分别拼接在所述N个回复令牌序列的尾部,并将所述N个推理令牌作为下一次获得的N个选定令牌;

[0026] 直到执行P次操作,获得N个回复令牌序列。

[0027] 可选的,所述获取单元,具体用于:

[0028] 获取待回复的N个文本数据;

[0029] 分别针对N个文本数据执行以下操作:将一个文本数据中的各个字符,分别编码为相应的令牌,获得所述一个文本数据对应的初始令牌序列。

[0030] 可选的,所述拼接单元,具体用于:

[0031] 获取基于所述N个初始令牌序列各自的终止令牌,拼接生成的第一令牌序列;

[0032] 以及,获取基于所述N个初始令牌序列各自除终止令牌以外的令牌,拼接生成的第二令牌序列;

[0033] 将所述第一令牌序列拼接在所述第二令牌序列的末尾,获得拼接令牌序列。

[0034] 可选的,所述第一令牌序列和所述第二令牌序列:分别是按照预设的拼接次序拼

接生成的,所述拼接次序表征:所述N个文本数据的处理次序;

[0035] 则所述拼接单元,用于获取基于所述N个初始令牌序列各自的终止令牌,拼接生成的第一令牌序列,具体用于:

[0036] 基于所述N个文本数据的处理次序,依次对所述N个初始令牌序列各自的终止令牌进行拼接处理,获得第一令牌序列;

[0037] 则所述拼接单元,用于所述获取基于所述N个初始令牌序列各自除终止令牌以外的令牌,拼接生成的第二令牌序列,具体用于:

[0038] 基于所述N个文本数据的处理次序,依次将所述N个初始令牌序列各自除终止令牌以外的令牌拼接为第二令牌序列。

[0039] 可选的,所述N个文本数据的处理次序,是采用以下任意一种方式确定的,则所述拼接单元,还用于:

[0040] 将所述N个文本数据的获取次序,作为所述N个文本数据的处理次序;

[0041] 将所述N个文本数据各自对应的时间戳的时间先后次序,作为所述N个文本数据的处理次序;

[0042] 将所述N个文本数据各自对应的优先级的级别高低次序,作为所述N个文本数据的处理次序。

[0043] 第三方面,本申请提供了一种计算机设备,包括存储器、处理器及存储在存储器上并可在处理器上运行的计算机程序,所述处理器执行所述计算机程序时实现上述第一方面中任意一种文本数据的推理方法。

[0044] 第四方面,本申请提供了一种计算机存储介质,所述计算机可读存储介质内存储有计算机程序指令,所述计算机程序指令被处理器执行上述第一方面中任意一种文本数据的推理方法。

[0045] 第五方面,本申请实施例提供的一种计算机程序产品,包括计算机程序指令,所述计算机程序指令被处理器执行时实现上述第一方面中任意一种文本数据的推理方法。

[0046] 本申请有益效果如下:

[0047] 本申请实施例中,提出一种文本数据的推理方法,计算设备获取N(N为大于1的整数)个文本数据各自对应的初始令牌序列,其中,获取的每个初始令牌序列中的一个令牌表征:相应文本数据中的一个字符,然后将N个初始令牌序列拼接为拼接令牌序列,再基于注意力机制,对拼接令牌序列进行推理处理,获得N个回复令牌序列,此处的N个回复令牌序列用于生成:N个文本数据各自的回复数据,并且每个回复令牌序列中的起始令牌是:在拼接令牌序列中,针对属于同一初始令牌序列的各个令牌进行推理处理获得的;实现无补齐字符填充的多文本数据推理,降低推理多条文本数据所消耗的GPU资源,提升计算机设备针对多条文本数据的推理性能。

[0048] 具体来说,一方面,提出一种多个文本数据的优化拼接方式,针对获取的N个文本数据各自对应的初始令牌序列进行拼接,获得拼接令牌序列,该拼接令牌序列中包含N个初始令牌序列中的所有令牌。这样,相较现有字符补齐方式,由于无需补齐填充字符,能够有效节约后续推理所需的GPU资源,进而降低针对文本数据的推理成本。

[0049] 另一方面,提出一种多个文本数据的优化推理方式,引入注意力机制,对拼接令牌序列进行推理处理,获得N个回复令牌序列,N个回复令牌序列用于生成N个文本数据各自的

回复数据。此外,每个回复令牌序列中的起始令牌是:在拼接令牌序列中,针对属于同一初始令牌序列的各个令牌进行推理处理获得的,换言之,注意力机制引入推理过程,其目的是要获得N个文本数据各自的回复数据,为了获得准确的N个回复数据,则需要确定每个回复令牌序列的起始令牌,基于此,获得N个回复令牌序列。如此,将注意力机制引入到数据推理的过程中,对拼接令牌序列中的属于不同初始令牌序列的各个令牌进行隔离,保证获得的N个回复令牌序列的推理准确性,进而保证N个文本数据各自的回复数据的准确性。

[0050] 还需说明的是,本申请实施例提供的文本数据的推理方法,在应用大语言模型的推理场景下,可以大幅提升计算设备调用海量模型参数,针对目标序列的推理性能;例如,针对长度差异较大的N个文本数据,则本方案相较现有字符补齐方式,能够提升一倍以上的推理性能,并且基于注意力机制保证推理结果的正确性,不仅能够适配现有深度学习模型框架,也能够实现模型框架的独立化,以适配后面提出的模型框架,提高GPU资源的利用率,降低针对文本数据的推理成本。

[0051] 本申请的其它特征和优点将在随后的说明书中阐述,并且,部分地从说明书中变得显而易见,或者通过实施本申请而了解。本申请的目的和其他优点可通过在所写的说明书、权利要求书、以及附图中所特别指出的结构来实现和获得。

附图说明

[0052] 此处所说明的附图用来提供对本申请的进一步理解,构成本申请的一部分,本申请的示意性实施例及其说明用于解释本申请,并不构成对本申请的不当限定。在附图中:

[0053] 图1为本申请实施例中可选的应用场景的示意图;

[0054] 图2为本申请实施例提供的文本数据的推理方法的流程示意图;

[0055] 图3A~图3B为本申请实施例中可能的对话场景示意图;

[0056] 图4为本申请实施例中基于深度学习模型框架的推理过程示意图;

[0057] 图5为本申请实施例中N个初始令牌序列的补齐示意图;

[0058] 图6A~图6D为本申请实施例中获取拼接令牌序列的过程示意图;

[0059] 图7为本申请实施例基于深度学习模型框架推理拼接令牌序列的过程示意图;

[0060] 图8A~图8B为本申请实施例中可能的注意力矩阵的示意图;

[0061] 图9为本申请实施例提供的文本数据的推理装置的示意图;

[0062] 图10为本申请实施例提供的计算机设备的一种结构示意图。

具体实施方式

[0063] 为了使本申请实施例的目的、技术方案和优点更加清楚,下面将结合本申请实施例中的附图,对本申请实施例中的技术方案进行清楚、完整地描述。

[0064] 本申请实施例中,所涉及的用户个人信息的收集、存储、使用、加工、传输、提供和公开等处理,均符合相关法律法规的规定,且不违背公序良俗。

[0065] 本申请实施例涉及人工智能技术,主要涉及人工智能技术中的自然语言处理技术。

[0066] 人工智能(Artificial Intelligence, AI):是利用数字计算机或者数字计算机控制的机器模拟、延伸和扩展人的智能,感知环境、获取知识并使用知识获得最佳结果的理

论、方法、技术及应用系统。换句话说,人工智能是计算机科学的一个综合技术,它企图了解智能的实质,并生产出一种新的能以人类智能相似的方式做出反应的智能机器。人工智能也就是研究各种智能机器的设计原理与实现方法,使机器具有感知、推理与决策的功能。

[0067] 人工智能技术是一门综合学科,涉及领域广泛,既有硬件层面的技术也有软件层面的技术。其中,预训练模型又称大模型、基础模型,经过微调后可以广泛应用于人工智能各大方向下游任务。人工智能基础技术一般包括如传感器、专用人工智能芯片、云计算、分布式存储、大数据处理技术、操作/交互系统、机电一体化等技术。人工智能软件技术主要包括计算机视觉技术、语音处理技术、自然语言处理技术以及机器学习/深度学习、自动驾驶、智慧交通等几大方向。

[0068] 自然语言处理(Nature Language processing, NLP):是计算机科学领域与人工智能领域中的重要方向。它研究能实现人与计算机之间用自然语言进行有效通信的各种理论和方法。自然语言处理涉及自然语言,即人们日常使用的语言,与语言学研究密切;同时涉及计算机科学和数学。人工智能领域模型训练的重要技术,预训练模型,即是从NLP领域的大语言模型发展而来。经过微调,大语言模型可以广泛应用于下游任务。自然语言处理技术通常包括文本处理、语义理解、机器翻译、机器人问答、知识图谱等技术。预训练模型是深度学习的最新发展成果,融合了以上技术。

[0069] 机器学习:是一门多领域交叉学科,涉及概率论、统计学、逼近论、凸分析、算法复杂度理论等多门学科。专门研究计算机怎样模拟或实现人类的学习行为,以获取新的知识或技能,重新组织已有的知识结构使之不断改善自身的性能。机器学习是人工智能的核心,是使计算机具有智能的根本途径,其应用遍及人工智能的各个领域。机器学习和深度学习通常包括人工神经网络、置信网络、强化学习、迁移学习、归纳学习、式教学习等技术。

[0070] 随着人工智能技术研究和进步,人工智能技术在多个领域展开研究和应用,例如常见的智能家居、智能穿戴设备、虚拟助理、智能音箱、智能营销、无人驾驶、自动驾驶、无人机、数字孪生、虚拟人、机器人、人工智能生成内容(AIGC)、对话式交互、智能医疗、智能客服、游戏AI等,相信随着技术的发展,人工智能技术将在更多的领域得到应用,并发挥越来越重要的价值。

[0071] 在本申请实施例中,将人工智能技术应用于数据推理领域中,用以降低基于目标模型(如:预训练模型),推理多条文本数据所消耗的GPU资源,提升计算机设备针对多条文本数据的推理性能。

[0072] 为便于理解本申请实施例提供的技术方案,下面对本申请实施例使用的一些关键词进行解释。

[0073] 预训练模型(Pretrained Model):是指在大规模的语料库上进行训练得到的模型,通常使用无监督学习的方法,例如自编码器、语言模型等。预训练模型的基本思想是利用大规模的语料库,通过无监督学习的方法,让模型学习到大量的通用知识和规律,从而作为各种自然语言处理任务的基础模型,例如本申请实施例在预训练模型的基础上,增加了适应于推荐理由生成任务的适应性训练,使得得到的模型能够用于推荐理由的生成。

[0074] 作为一种示例,本申请实施例涉及的目标模型为预训练模型,具体可以是任一深度学习模型;例如:生成型预训练变换模型(Chat Generative Pre-trained Transformer, ChatGPT)、预训练语言模型(Bidirectional Encoder Representations from

Transformers, BERT) 等。

[0075] Transformer:是一种常见的深度学习模型架构,在自然语言处理、计算机视觉(Computer Vision, CV)和语音处理等各个领域得到了广泛应用。Transformer最初被提出时是一种用于机器翻译的序列到序列的模型架构,由编码器和解码器组成,编码器和解码器都是一系列相同结构的Transformer块(Block)组成,每个Transformer块至少由多头自注意力层和前馈神经网络层组成。目前Transformer已经成为自然语言处理中的常用架构,且常作为预训练模型使用。除了与语言相关的应用之外,Transformer还被应用于计算机视觉、音频处理等领域。

[0076] 语言模型(Language Model):是一种用于对自然语言进行建模的模型,其目的是预测一个给定的文本序列的下一个单词或字符,语言模型可以用于多种自然语言处理任务,例如文本的语义抽取、文本生成、机器翻译、语音识别等。目前,基于Transformer的预训练语言模型(pre-trained language model, PLM)在自然语言处理的各种任务上较为常见,且通常能够取得较为良好的效果,例如较为常见的预训练语言模型包括基于双向编码表示的Transformer模型(Bidirectional Encoder Representation from Transformers, Bert)以及生成式预训练Transformer模型(Generative Pre-Trained Transformer, GPT)等。

[0077] 大语言模型(Large Language Model, LLM):是指具有大规模参数和训练数据的自然语言处理模型。大语言模型的训练过程通常采用无监督学习的方式,即通过大规模文本语料库来训练模型,从而学习到语言的概率分布和语言规律。在训练过程中,大语言模型通常采用语言模型(Language Model)作为目标函数,即通过最大化下一个单词的预测概率来优化模型参数,例如基于Transformer模型结构的GPT系列模型,它是在大规模语料库上进行训练,可以生成高质量的自然语言文本,如文章、对话等。

[0078] 分词器(tokenizer):是一种将自然语言文本转换为字符、单词或子词序列的工具。在Transformer模型中,tokenizer通常指的是将自然语言文本转换为模型输入所需的token(分词)序列的工具,通常采用的是基于字或子词(sub-token)的分词方法,例如字节对编码(Byte Pair Encoding, BPE)或者句子片段(Sentence Piece)等,这些方法可以将单词或者子词拆分成更小的单元,以便模型更好地处理不常见的单词或者词汇表中没有的单词。

[0079] 注意力机制(Attention机制):是一种通过使用高级信息来衡量网络中间特征,使得网络关注于图像中辅助判断的部分信息,忽略不相关信息的方式,注意力机制的本质来自于人类视觉注意力机制,人们视觉在感知东西的时候一般不会是一个场景从到头看到尾每次全部都看,而往往是根据需求观察注意特定的一部分,而且当人们发现一个场景经常在某部分出现自己想观察的东西时,人们会进行学习在将来再出现类似场景时把注意力放到该部分上。因此,注意力机制实质上是从大量信息中筛选出高价值信息的手段,在大量信息中,不同信息对于结果的重要性是不同的,这种重要性可以通过赋予不同大小的权值来体现,换言之,注意力机制可以理解成对多个来源进行合成时分配权重的一种规则。通常可以用于解决模型输入序列较长的时候很难获得最终合理的向量表示问题,做法是保留模型的中间结果,用新的模型对其进行学习,并将其与输出进行关联,从而达到信息筛选的目的。Attention机制包括Attention机制、自Attention机制、单头Attention机制以及多头

Attention机制等。

[0080] 令牌(token):是一种最小语义单元,又名分词、词元、token。

[0081] 下面对本申请实施例的设计思想进行简要介绍。

[0082] 目前,对话类的大语言模型通常采用Transformer这种深度学习模型架构,并且,现有大语言模型的模型参数量已达到70亿以上,比如GPT3的模型参数便高达175亿,这导致模型推理过程对GPU资源的消耗是巨大的。

[0083] 针对上述问题,为避免每次只推理一个文本数据而导致GPU资源利用效率低下,提出将多个文本数据作为同一批进行数据推理处理的方式,相关技术方案可概括为如下两种:

[0084] 相关方案一:在将多个文本数据作为同一批进行数据推理之前,为保证推理过程能够适配现有的深度学习框架,还需要在相对较短的文本数据中补齐填充(padding)字符,以使参与拼接处理的每个文本数据的长度保持一致。

[0085] 然而,上述通过字符补齐方式所添加的填充字符也将参与实际的推理运算,这样,会造成大量无效的计算,以及大量无效GPU资源的消耗;进一步,即便是从多文本数据中选取长度一致的至少两个文本数据,拼接为处理对象,也无法完全避免填充字符的补齐,并且,只有在面对足够多的文本数据的场景下,才有可能找到两个长度一致的文本数据,换言之,该方式受限于多文本数据的数据量以及多文本数据的长度分布,造成大量无效的计算和大量无效GPU资源的消耗。

[0086] 相关方案二:在相关方案一的基础上,通过改进大语言模型的模型框架,进而基于改进的模型框架,对相关方案一涉及的多余填充字符进行过滤处理,解决由多余填充字符造成的无效计算和无效GPU资源消耗。

[0087] 然而,上述改进模型框架的方式,首先是需要模型框架层面上进行修改处理,仅适用于特定的模型框架,其次是在模型应用层面上存在局限性,就当前应用来说,仅支持Bert类文本理解应用,而不支持对话类或生成类的大语言模型应用。

[0088] 鉴于此,本申请实施例提供了一种文本数据的推理方法,在适用于各种大语言模型应用的推理场景下,可以大幅提升计算设备调用海量模型参数,针对多文本数据的推理性能;例如,针对长度差异较大的N个文本数据,则本方案相较现有字符补齐方式,能够提升一倍以上的推理性能,不仅能够适配现有深度学习模型框架,也能够实现模型框架的独立化,以适配后面提出的模型框架,提高GPU资源的利用率,降低针对文本数据的推理成本。

[0089] 具体来说,本申请实施例中,提供了一种多个文本数据的优化拼接方式,针对获取的N个文本数据各自对应的初始令牌序列进行拼接,获得拼接令牌序列,该拼接令牌序列中包含N个初始令牌序列中的所有令牌。这样,相较现有字符补齐方式,由于无需补齐填充字符,能够有效节约后续推理所需的GPU资源,进而降低针对文本数据的推理成本。

[0090] 其次,本申请实施例中,还提供了一种多个文本数据的优化推理方式,引入注意力机制,对拼接令牌序列进行推理处理,获得N个回复令牌序列,N个回复令牌序列用于生成N个文本数据各自的回复数据。此外,每个回复令牌序列中的起始令牌是:在拼接令牌序列中,针对属于同一初始令牌序列的各个令牌进行推理处理获得的,换言之,注意力机制引入推理过程,其目的是要获得N个文本数据各自的回复数据,为了获得准确的N个回复数据,则需要确定每个回复令牌序列的起始令牌,基于此,获得N个回复令牌序列。如此,将注意力机

制引入到数据推理的过程中,对拼接令牌序列中的属于不同初始令牌序列的各个令牌进行隔离,保证获得的N个回复令牌序列的推理准确性,进而保证N个文本数据各自的回复数据的准确性。

[0091] 下面对本申请实施例的技术方案能够适用的应用场景做一些简单介绍,需要说明的是,以下介绍的应用场景仅用于说明本申请实施例而非限定。在具体实施过程中,可以根据实际需要灵活地应用本申请实施例提供的技术方案。

[0092] 本申请实施例提供的方案可以适用于文本数据的推理场景中,用于降低推理多条文本数据所消耗的GPU资源,提升计算机设备针对多条文本数据的推理性能。如图1所示,为本申请实施例提供的一种应用场景示意图,在该场景中,可以包括终端设备101和服务器102。

[0093] 终端设备101例如可以为手机、平板电脑(PAD)、笔记本电脑、台式电脑、智能电视、智能车载设备、智能可穿戴设备、智能电视以及飞行器等任意涉及到文本数据推理的设备。终端设备101可以安装有目标应用,目标应用可以具备获取使用对象输入的待推理的N个文本数据、展示N个文本数据、获取N个文本数据各自对应的初始令牌序列、将N个初始令牌序列拼接为拼接令牌序列、获取拼接令牌序列、获取及展示N个文本数据各自的回复令牌序列、获取及展示N个文本数据各自的回复数据等功能,例如可以为即时通信应用、音乐应用、游戏应用、视频应用、短视频应用、新闻应用以及购物应用等。本申请实施例涉及的应用可以是软件客户端,也可以是网页、小程序等客户端。服务器102则是与软件或是网页、小程序等相对应的服务器,不限制客户端的具体类型。

[0094] 需要说明的是,上述终端设备101,对于N个文本数据各自对应的初始令牌序列的获取过程、对于N个初始令牌序列拼接为拼接令牌序列、以及对于拼接令牌序列的获取过程并不是必须,还可以是终端设备101将N个文本数据发送至服务器102后,服务器102基于接收的N个文本数据进行处理生成的。

[0095] 服务器102可以为目标应用的后台服务器,用于为其提供相应的后台服务,例如,以及数据推理服务等。其可以是独立的物理服务器,也可以是多个物理服务器构成的服务器集群或者分布式系统,还可以是提供云服务、云数据库、云计算、云函数、云存储、网络服务、云通信、中间件服务、域名服务、安全服务、即内容分发网络(Content Delivery Network, CDN)、以及大数据和人工智能平台等基础云计算服务的云端服务器,但并不局限于此。

[0096] 需要说明的是,本申请实施例中的文本数据的推理方法可以由终端设备101或者服务器102单独执行,也可以由服务器102和终端设备101共同执行。当由终端设备101或者服务器102单独执行时,则应用模型进行数据推理的过程都可以由终端设备101或者服务器102单独实现,例如可以在终端设备101上,通过对待处理的N(N为大于1的整数)个文本数据各自对应的初始令牌序列进行拼接,获得拼接令牌序列,再基于注意力机制,对拼接令牌序列进行推理处理,获得N个回复令牌序列,或者也可以由服务器102执行上述分词处理过程、拼接处理过程和推理处理过程中的一种及组合。当由服务器102和终端设备101共同执行时,则可以由服务器102对大语言模型进行训练之后,将预训练的语言模型部署至终端设备101中,由终端设备101实现数据推理过程,或者,数据推理的部分过程可以由服务器102实现,其他过程可以由终端设备101实现,在实际应用时可以根据情况进行具体的配置,本申

请在此不做具体限定。

[0097] 其中,服务器102和终端设备101均可以包括一个或多个处理器、存储器以及与交互I/O接口等。此外,服务器102还可以配置数据库,可以用于存储训练得到的模型参数、已训练的目标模型等。其中,服务器102和终端设备101的存储器中还可以存储本申请实施例提供的推荐理由生成方法中各自所需执行的程序指令,这些程序指令被处理器执行时能够用以实现本申请实施例提供的推理过程。

[0098] 需要说明的是,当本申请实施例提供的文本数据的推理方法由服务器102或者终端设备101单独执行时,则上述的应用场景中也可以仅包含服务器102或者终端设备101单一设备,或者,也可以认为服务器102和终端设备101其为同一个设备。当然,在实际应用时,当本申请实施例提供的文本数据的推理方法由服务器102和终端设备101共同执行时,服务器102和终端设备101也可以为同一个设备,即服务器102和终端设备101可以为同一设备的不同功能模块,或者由同一物理设备所虚拟出的虚拟设备。

[0099] 本申请实施例中,终端设备101和服务器102之间可以通过一个或者多个网络103进行直接或间接的通信连接。该网络103可以是有线网络,也可以是无线网络,例如无线网络可以是移动蜂窝网络,或者可以是无线保真(Wireless-Fidelity,WIFI)网络,当然还可以是其他可能的网络,本申请实施例对此不做限制。需要说明的是,图1所示只是举例说明,实际上终端设备和服务器的数量不受限制,在本申请实施例中不做具体限定。

[0100] 下面,结合上述描述的应用场景,参考附图来描述本申请示例性实施方式提供的方法,需要注意的是,上述应用场景仅是为了便于理解本申请的精神和原理而示出,本申请的实施方式在此方面不受任何限制。且需要说明的是,下述的方法可以由上述终端设备或者服务器执行,也可以由终端设备和服务器共同执行,这里具体是以终端设备或者服务器执行为例进行示出的。

[0101] 参阅图2所示,为本申请实施例提供的一种文本数据的推理方法的实施流程图,以表征终端设备或服务器的计算设备为执行主体为例,该方法的具体实施流程如下:

[0102] 步骤201:获取N个文本数据各自对应的初始令牌序列;其中,N为大于1的整数,每个初始令牌序列中的一个令牌表征:相应文本数据中的一个字符。

[0103] 本申请实施例中,一个文本数据可以是使用对象触发的一个对话请求,则N个文本数据可以是同一(或不同)使用对象触发的N个对话请求,计算设备响应于接收到N个对话请求,将相应的N个文本数据作为目标模型(如:大语言模型)的模型输入,调用目标模型中海量的模型参数,来对N个文本数据进行推理处理。

[0104] 具体地,以对话应用场景为例,使用对象在计算设备的前端界面输入N个文本数据后,计算设备才会针对这N个文本数据进行处理;或者使用对象在计算机设备的前端界面输入M(M为大于N的整数)个文本数据后,计算机设备提取N个文本数据进行处理。

[0105] 参见图3A所示,为一种对话场景的示意图,其中,使用对象在前端界面中输入“等闲识得东风面的下一句”,相关技术中,计算设备在接收到这个文本数据后,可视为接收到一个对话请求或一个模型输入,则将调用目标模型内海量的模型参数,对“等闲识得东风面的下一句”进行推理,获得“万紫千红总是春”的推理结果。

[0106] 然而,针对图3A所示的方式,计算设备每接收一个文本数据,就需要调用目标模型内海量的模型参数,进而带来大量GPU资源的消耗;为解决该问题,本申请实施例在接收到

两个及两个以上的文本数据后,才会针对这些文本数据进行处理,下面是本申请实施例提供的技术方案所适用于一种对话场景。

[0107] 参见图3B所示,为另一种对话场景的示意图,其中,使用对象在前端界面中输入“等闲识得东风面的下一句”,计算设备在接收到这个文本数据后,可视为接收到一个对话请求或一个模型输入,但不会立即调用目标模型内海量的模型参数,而是在接收到N个文本数据后,才调用目标模型内海量的模型参数,获得相应的推理结果;在此以接收2个文本数据为例,即使用对象再次输入“这首诗的作者是谁”,进而计算设备接收到两个对话请求或两个模型输入后,计算设备才会调用目标模型内海量的模型参数,获得文本数据1对应的推理结果2“万紫千红总是春”,以及文本数据2对应的推理结果2“朱熹”。这样,在实际应用过程中,面对海量的文本数据,本方案对于多个文本数据进行处理的方式,相较单个文本数据逐一处理的方式,可以提高GPU的利用率。

[0108] 进一步,在实际应用中,还需要分别针对获得的N个文本数据执行以下操作:将一个文本数据中的各个字符,分别编码为相应的令牌,获得一个文本数据对应的初始令牌序列。

[0109] 具体的,以一个文本数据为例,可以通过分词器将一个文本数据切分成多个字符,获得相应的字符序列,然后对每个字符进行编码,一个编码后的字符可视为一个token,进而获得相应的初始令牌序列。

[0110] 其中,一个令牌为一个语义最小单元,在本申请实施例主要以字符为例进行说明,例如“等闲识得东风面的下一句”中的字符可以划分为“等-闲-识-得-东-风-面-的-下-一-句”。

[0111] 作为一种更为具体的示例,参见图4所示,为一种基于深度学习模型框架的推理过程示意图,其中, X1、X2、X3、X4、X5表示模型的输入,以对话类的大语言模型(如:ChatGPT)为例,X1、X2、X3、X4、X5可表示:针对使用对象输入的一个文本数据中的各个字符进行编码处理后,获得的一个初始令牌序列中的各个令牌;h1、h2、h3、h4、00表示:一个初始令牌序列中各个令牌经过大语言模型计算后对应的输出(也即:推理令牌),其中h1、h2、h3、h4会省去不用;则00表示模型生成推理结果中的第一个推理令牌;00会输入模型继续计算生成下一个推理令牌。每生成一个推理令牌,该大语言模型中的所有模型参数都将参与计算。在此,以100亿个参数的大语言模型为例,假设模型以半精度(FP16)保存,模型大小为20GB。每生成一个推理令牌,都需要将20GB的大语言模型从显存加载并参与计算一次。因此,容易理解地,基于大语言模型的文本数据推理过程,显存带宽成为主要瓶颈。

[0112] 相关技术中,为了提升GPU利用效率,一般会将多个文本数据拼接到一起,这样,加载一次模型,即可生成多个文本数据各自的推理结果,从而缓解模型推理过程中的显存带宽瓶颈问题。

[0113] 但在实际应用中,获取到的N个文本数据各自对应的初始令牌序列的长度是不一样的,也即N个初始令牌序列中包含的令牌数量不一样,为了匹配使用各种GPU计算加速库(如:CuBLAS)等,相关技术在调用模型之前,需要先采用填充字符,将N个初始令牌序列补齐(padding)到相同长度。

[0114] 参见图5所示,为本申请实施例提供的现有技术的一种N个初始令牌序列的补齐示意图。其中,一行表示:一个初始令牌序列;一个非空白方块表示:一个令牌;一个空白方块

表示:一个用于补齐分词序列的填充字符对应的令牌;0~5的序号分别表示:单个令牌在相应的初始令牌序列中的位置信息。如图5所示,可以看出,其中一半个令牌都是用做补齐的,由于这些补齐的令牌也将参与后面的推理处理,也就导致了大量冗余计算和带宽开销。

[0115] 综上,本申请实施例在获取到N个初始令牌序列后,还需要执行下面步骤202提供的一种无冗余的拼接方式,以提高推理性能。

[0116] 步骤202:将N个初始令牌序列拼接为拼接令牌序列。

[0117] 本申请实施例中,由于无需对N个初始令牌序列进行补齐,因此可以实现对于文本数据的长度分布的去依赖化,相较现有字符补齐方式,有效节约后续推理所需的GPU资源,并且,这样获得的拼接令牌序列能够兼容现有的深度学习框架,进而降低针对多文本数据的推理成本。

[0118] 在一种实施方式中,获取基于N个初始令牌序列各自的终止令牌,拼接生成的第一令牌序列,以及,获取基于N个初始令牌序列各自除终止令牌以外的令牌,拼接生成的第二令牌序列;然后,将第一令牌序列拼接在第二令牌序列的末尾,获得拼接令牌序列。

[0119] 具体来说,针对N个初始令牌序列,将每个初始令牌序列分为两个部分:位于终止位置的终止令牌,以及,位于非终止位置的其他令牌;则第一令牌序列是N个终止令牌拼接而成的,第二令牌序列是若干其他令牌拼接获得的。

[0120] 参见图6A所示,为本申请实施例中针对N个初始令牌序列的划分示意图,其中,涉及四个初始令牌序列:初始令牌序列1包含一个令牌,初始令牌序列2包含四个令牌,初始令牌序列3包含一个令牌,初始令牌序列4包含六个令牌,则确定初始令牌序列1中位置信息表征0的令牌为终止令牌1,确定初始令牌序列2中位置信息表征为3的令牌为终止令牌2,确定初始令牌序列3中位置信息表征为0的令牌为终止令牌3,提取初始令牌序列4中表征为5的令牌为终止令牌4。

[0121] 可选的,第一令牌序列和第二令牌序列:分别是按照预设的拼接次序拼接生成的,拼接次序表征:N个文本数据的处理次序。

[0122] 具体来说,基于预设的拼接次序,将确定的N个终止令牌拼接为第一令牌序列。以及,基于预设的拼接次序,将N个初始令牌序列各自除终止令牌以外的令牌拼接为第二令牌序列。

[0123] 参见图6B所示,为本申请实施例中第一令牌序列的获得示意图,其中,以**初始令牌序列1→初始令牌序列2→初始令牌序列3→初始令牌序列4**的拼接次序为例,依次对初始令牌序列1中位置信息表征为0的终止令牌、初始令牌序列2中位置信息表征为3的终止令牌、初始令牌序列3中位置信息表征为0的终止令牌、初始令牌序列4中位置信息表征为5的终止令牌进行拼接,获得第一令牌序列。

[0124] 相应的,基于N个文本数据的处理次序,依次将N个初始令牌序列各自除终止令牌以外的令牌拼接为第二令牌序列。

[0125] 参见图6C所示,为本申请实施例中第二令牌序列的获得示意图,其中,以**初始令牌序列1→初始令牌序列2→初始令牌序列3→初始令牌序列4**的拼接次序为例,依次对初始令牌序列2中位置信息表征为0、1、2的三个令牌、初始令牌序列4中位置信息表征为0、1、2、3、4的四个令牌进行拼接,获得第二令牌序列。

[0126] 需要说明的是,上述N个文本数据的处理次序,可以采用以下任意一种方式确定,本申请对此不作具体限制。例如,将N个文本数据的获取次序,作为N个文本数据的处理次序;例如,将N个文本数据各自对应的时间戳的时间先后次序,作为N个文本数据的处理次序;再例如,将N个文本数据各自对应的优先级的级别高低次序,作为N个文本数据的处理次序。

[0127] 在获取到第一令牌序列以及第二令牌序列之后,参见图6D所示,为本申请实施例中拼接令牌序列的获得示意图,将第一令牌序列拼接至第二令牌序列的终止位置(也即尾部),获得拼接令牌序列。

[0128] 可选的,作为一种可能的实现方式,在拼接令牌序列中,每个令牌还可关联有如下信息:一个令牌在相应的初始令牌序列中的位置信息,一个令牌与其所属初始令牌序列对应的文本数据之间的关联信息等。

[0129] 示例性的,如图6D所示的目标序列,其中,一个方块对应的令牌是针对相应字符进行编码处理获得的,则方块本身表征相应字符的编码信息(即:令牌),方块花纹用于表征相应令牌与其所属初始令牌序列对应的文本数据之间的关联信息,方块下方标识的数值用于表征相应令牌在其所属初始令牌序列的所处位置信息。

[0130] 综上所述,本申请实施例提供一种针对N个初始令牌序列的无冗余拼接方式,这样获得的拼接令牌序列可避免相关技术对于N个初始令牌序列的长度分布依赖;进一步,为确保拼接令牌序列对应推理结果的正确性,需执行如下步骤203针对拼接令牌列进行推理处理。

[0131] 步骤203:基于注意力机制,对拼接令牌序列进行推理处理,获得N个回复令牌序列;其中,N个回复令牌序列用于生成N个文本数据各自的回复数据,每个回复令牌序列中的起始令牌是:在拼接令牌序列中,针对属于同一初始令牌序列的各个令牌进行推理处理获得的。

[0132] 本申请实施例,在推理过程中引入注意力机制,对拼接令牌序列中的属于不同初始令牌序列的各个令牌进行隔离,保证获得的N个回复令牌序列的推理准确性,进而保证N个文本数据各自的回复数据的准确性。

[0133] 具体来说,将拼接令牌序列输入目标模型,获得目标模型根据注意力机制,针对拼接令牌序列中的各个令牌进行推理,输出的候选令牌序列,候选令牌序列中的每个令牌为:拼接令牌序列中一个令牌的推理结果;然后,从候选令牌序列中,确定N个初始令牌序列各自终止令牌所对应的推理结果,获得N个选定令牌;再基于目标模型,针对N个选定令牌进行迭代推理,获得目标模型输出的N个回复令牌序列,其中N个选定令牌分别为N个回复令牌序列的起始令牌。

[0134] 换言之,是将所述拼接令牌序列输入目标模型,得到目标模型根据注意力机制输出的候选令牌序列,该候选令牌序列中的每个令牌是拼接令牌序列中一个令牌的输出结果;然后,从候选令牌序列的令牌中选取N个令牌作为N个选定令牌,这N个选定令牌在拼接输入令牌序列中对应的令牌、并且是N个初始令牌序列中每个初始令牌序列中的最后一个令牌,每个初始令牌序列中的终止令牌(最后一个令牌)用于表征一个文本数据中的最后一个字符;再将N个选定令牌分别作为N个回复令牌序列中的第一个令牌,输入目标模型,以得到N个回复令牌序列中的其他令牌,在此N个回复令牌序列用于生成N个文本数据各自的回

复数据。

[0135] 为便于理解,下面先结合普适性的深度学习模型框架,来对本申请实施例针对拼接令牌序列的推理过程进行详细说明。

[0136] 参见图7所示,为本申请实施例基于深度学习模型框架推理拼接令牌序列的过程示意图,此处以基于两个初始令牌序列 $I_{00:n-1}$, $I_{10:n-1}$ 拼接获得的拼接令牌序列为例,为便于理解,如图7所示中将该拼接令牌序列示为相应的两个初始令牌序列,本领域技术人员当知,计算设备对拼接令牌序列中的各个令牌进行推理处理,实质是对两个初始令牌序列中的令牌分别进行推理处理。

[0137] 如图7所示,可以看出,将两个初始令牌序列 $I_{00:n-1}$, $I_{10:n-1}$ (也即:拼接令牌序列)输入目标模型, 000 、 010 是模型针对 $I_{00:n-1}$, $I_{10:n-1}$ (也即:拼接令牌序列中的各个令牌)中的各个令牌进行推理后, $I_{00:n-1}$, $I_{10:n-1}$ 各自的终止令牌对应的推理令牌,其中, 000 、 010 各自的第一个0表示模型输出, 000 的第二个0表示对应第一个初始令牌序列, 010 的第二个1表示对应第二个初始令牌序列, 000 、 010 各自的第三个0表示针对相应令牌序列中的终止令牌进行推理后输出的推理令牌,则 000 、 010 也表示两个初始令牌序列各自对应的文本数据的回复令牌序列的首个令牌,然后将 000 、 010 输入目标模型,生成两个输入各自对应的推理令牌,依次循环,直到生成两个文本数据各自对应的回复令牌序列;一个回复令牌序列可以是: 000 、 001 、 \dots 、 $00n$ 的拼接序列;另一个回复令牌序列可以是: 010 、 011 、 \dots 、 $01n$ 的拼接序列。

[0138] 也就是说,本申请实施例在针对拼接令牌序列的推理过程中,主要基于注意力机制,使得目标模型在进行数据推理过程中,保证输出的每个推理令牌仅与其自身对应的输入令牌以及自身对应的已生成的推理令牌进行关联推理,实现在拼接令牌序列的推理过程中,对不同文本数据的推理隔离。

[0139] 针对注意力机制在推理过程的应用形式,可以是注意力矩阵,当然还可以是其他形式,在此不做具体限定,文方案以预设的注意力矩阵为例,下文不再赘述。

[0140] 其中,预设的注意力矩阵表征:在推理过程中,基于对拼接令牌序列中的各个令牌与其所属初始令牌序列之间的关联关系,对各个令牌及其对应的推理令牌之间的关注程度。注意力矩阵中可包含若干行元素,每行元素表征:在推理过程中,对拼接令牌序列中的各个令牌具有的不同的关注程度;或者每行元素表征:在推理过程中,拼接令牌序列中的各个元素、各个元素对应的推理令牌具有的不同的关注程度。

[0141] 作为一种示例,注意力矩阵包含: $Q+P \times N$ 行元素。其中, Q 为拼接令牌序列中的令牌总数, Q 行元素中的每行元素表征:在推理过程中,对拼接令牌序列中各个令牌具有的不同的关注程度, P 为正整数, $P \times N$ 行元素中的每行元素表征,在推理过程中,对拼接令牌序列中各个令牌及其对应的推理令牌具有的不同的关注程度,并且每 N 行元素分别与 N 个文本数据相对应。

[0142] 为便于理解,下面先以单个初始令牌序列对应的注意力矩阵为例,对本申请实施例提出的注意力矩阵的构成结构进行简要阐述。

[0143] 参见图8A所示,为单个初始令牌序列对应的注意力矩阵,其中,注意力矩阵中的列号和行号可以表示令牌的序号,从上往下看,第一行只有第一个是斜杠填充,则表示单个初始令牌序列中的第一个令牌只能与自身进行基于注意力机制的推理处理;第二行只有第一

个和第二个是斜杠填充,则表示单个初始令牌序列中的第二个令牌可以和自身、第一个令牌进行基于注意力机制的推理处理,以此类推;换言之,图8A所示的注意力矩阵表示,单个初始令牌序列中的每个令牌只能与自身(需要说明的是,在此令牌自身也可以是推理令牌)、自身以前的令牌进行基于注意力机制的推理处理。

[0144] 在介绍完单个初始令牌序列对应的注意力矩阵后,下面对本申请实施例提供的拼接令牌序列对应的注意力矩阵做如下具体阐述。

[0145] 针对上述注意力矩阵中的 $Q+P \times N$ 行元素,其中 Q 行元素的排列次序是:基于 N 个初始令牌序列各自包含的、除终止令牌以外的各个令牌的令牌排列次序确定的。则 Q 行元素可以采用以下方式确定:任一选定一个初始令牌序列中的起始令牌(非终止令牌),将之作为推理过程的关注令牌,并确定其在拼接令牌序列中的位置信息,构建出第一行元素,然后再根据同一初始令牌中的各个后续令牌(非终止令牌),执行类似操作,生成相应的各行元素;以此类推,生成 Q 行元素。

[0146] 在一种可能的实现方式中,候选令牌序列,是通过如下方式推理获得的:获取预设注意力矩阵中的 Q 行元素,然后,基于 Q 行元素中的各行元素,分别对拼接令牌序列中的、属于同一初始令牌序列的各个令牌进行推理处理,获得拼接令牌序列中各个令牌各自对应的推理令牌,进而获得由各个推理令牌拼接生成的候选令牌序列。

[0147] 针对上述注意力矩阵中的 $Q+P \times N$ 行元素,其中 $P \times N$ 行元素的排列次序是:基于 N 个文本数据的处理次序确定的,也即与 N 个回复令牌序列的生成次序一致。

[0148] 在一种可能的实现方式中, N 个回复令牌序列,是通过如下方式推理获得的:将 N 个选定令牌分别作为 N 个回复令牌序列的起始令牌;获取预设注意力矩阵中的 $P \times N$ 行元素,然后,针对 $P \times N$ 行元素中的每 N 行元素,依次 P 次操作,获得 N 个回复令牌序列,一次执行操作具体如下:

[0149] 基于 N 行元素,分别对当前获得的 N 个选定令牌进行推理处理,获得 N 个推理令牌,在此, N 行元素中的每行元素表征:在推理过程中,分别对当前获得的 N 个选定令牌具有不同的关注程度,然后,将 N 个推理令牌分别拼接在 N 个回复令牌序列的尾部,并将 N 个推理令牌作为下一次获得的 N 个选定令牌。

[0150] 下面以一个较为完整的实例,对上述目标模型基于注意力矩阵,针对拼接令牌序列进行推理,获得 N 个回复令牌序列的推理过程,做如下示例性阐述。

[0151] 参见图8B所示,为针对拼接令牌序列预设的注意力矩阵,其中,拼接令牌序列是基于包含5个令牌的第一初始令牌序列和包含4个令牌的第二初始令牌序列拼接获得的,拼接获得的方式可以参见步骤202相关阐述,在此不做赘述。

[0152] 如图8B所示,input1包括四行元素,其对应构成拼接令牌序列的第一初始令牌序列中的前4个令牌,input2包含三行元素,其表示构成拼接令牌序列的第二初始令牌序列中的前3个令牌;则input1+input2包括7(即 Q)行元素,基于这七行元素,可推理获得拼接令牌序列对应的候选令牌序列;第8行表示构成拼接令牌序列的第一初始令牌序列中的终止令牌,也即,第一初始令牌序列中的终止令牌只与第一初始令牌序列中的前4个令牌、进行基于注意力机制的推理处理;第9行表示构成拼接令牌序列的第二初始令牌序列中的终止令牌,也即,第二初始令牌序列中的终止令牌只与第二初始令牌序列中的前3个令牌、进行基于注意力机制的推理处理;则第8、9行元素构成 N 行元素,后面根据目标模型的输出长度,注

注意力矩阵可以自行拓展为 $P \times N$ 行元素。

[0153] 需要说明的是,上述注意力矩阵的自行拓展,与 N 个文本数据各自推理结果的生成次序相关,是故,一般应当遵循预设的拼接次序,进行注意力矩阵的自行拓展。

[0154] 下面结合实际应用场景,来对本申请实施例提供的一种多文本数据的推理方法做整体性的阐述,以应用大语言模型的对话场景为例。

[0155] 使用对象在显示界面中输入各种问题或待推理的对话数据作为待处理的文本数据。计算设备在获取到使用对象输入的一个待处理的文本数据后,不直接启用大语言模型,而是在接收 N (N 为大于1的整数)个文本数据后,才准备启用大语言模型,以推理 N 个文本数据。

[0156] 后续,针对待处理的 N 个文本数据,计算设备分别对每个文本数据中的各个字符进行编码处理获得相应的初始令牌序列,再 N 个初始令牌序列分别划分为终止令牌和非终止令牌,基于预设的拼接次序,将 N 个终止令牌拼接而成的第一令牌序列拼接在,若干非终止令牌拼接而成的第二令牌序列的尾部,获得拼接令牌序列,其中拼接令牌序列中包含 N 个初始令牌序列中的所有令牌。

[0157] 然后,将拼接令牌序列输入大语言模型,获得大语言模型根据注意力机制,针对拼接令牌序列中的各个令牌进行推理,输出的候选令牌序列,该候选令牌序列中的每个令牌为:拼接令牌序列中一个令牌的推理结果,再从候选令牌序列中,确定 N 个初始令牌序列各自终止令牌所对应的推理结果,获得 N 个选定令牌;后续,基于目标模型,针对 N 个选定令牌进行迭代推理,获得目标模型输出的 N 个回复令牌序列,其中 N 个选定令牌分别为 N 个回复令牌序列的起始令牌。

[0158] 综上所述,本申请实施例提供一种多文本数据的推理方法,可以大幅提升大语言模型的推理性能,尤其针对Transformer类的大语言模型。在多文本数据的长度差异较大的场景,本方案相比现有补齐文本数据的方法,可以提升1倍以上的推理性能,进而望一定程度缓解当前大语言模型推理成本太高的问题。此外,相比现有深度学习模型框架的优化方案,本方案不仅对多文本数据的分布没有依赖,提升推理性能,并且可以兼容已有的深度学习模型框架(如Huggingface、Pytorch、Tensorflow等),提高针对多文本数据的推理灵活性。

[0159] 参见图9所述,基于同一发明构思,本申请实施例还提供了一种文本数据的推理装置,装置包括:

[0160] 获取单元901,获取 N 个文本数据各自对应的初始令牌序列;其中, N 为大于1的整数,每个初始令牌序列中的一个令牌表征相应文本数据中的一个字符;

[0161] 拼接单元902,将 N 个初始令牌序列拼接为拼接令牌序列;

[0162] 推理单元903,基于注意力机制,对所述拼接令牌序列进行推理处理,获得 N 个回复令牌序列;其中,所述 N 个回复令牌序列用于生成所述 N 个文本数据各自的回复数据,每个回复令牌序列中的起始令牌是:在所述拼接令牌序列中,针对属于同一初始令牌序列的各个令牌进行推理处理获得的。

[0163] 可选的,所述推理单元903,具体用于:

[0164] 将所述拼接令牌序列输入目标模型,获得所述目标模型根据注意力机制,针对所述拼接令牌序列中的各个令牌进行推理,输出的候选令牌序列,所述候选令牌序列中的每

个令牌为:所述拼接令牌序列中一个令牌的推理结果;

[0165] 从所述候选令牌序列中,确定所述N个初始令牌序列各自终止令牌所对应的推理结果,获得N个选定令牌;

[0166] 基于所述目标模型,针对所述N个选定令牌进行迭代推理,获得所述目标模型输出的N个回复令牌序列;其中,所述N个选定令牌分别为所述N个回复令牌序列的起始令牌。

[0167] 可选的,所述候选令牌序列,是通过如下方式推理获得的,则所述推理单元903,还用于:

[0168] 获取预设注意力矩阵中的Q行元素;其中,Q为所述拼接令牌序列中的令牌总数,所述Q行元素中的每行元素表征:在推理过程中,对所述拼接令牌序列中各个令牌具有的不同关注程度;

[0169] 基于所述Q行元素中的各行元素,分别对所述拼接令牌序列中的、属于同一初始令牌序列的各个令牌进行推理处理,获得所述拼接令牌序列中各个令牌各自对应的推理令牌;

[0170] 获得由各个推理令牌拼接生成的候选令牌序列。

[0171] 可选的,所述N个回复令牌序列,是通过如下方式推理获得的,则所述推理单元903,还用于:

[0172] 将所述N个选定令牌分别作为N个回复令牌序列的起始令牌;

[0173] 获取预设注意力矩阵中的 $P \times N$ 行元素;其中,P为正整数;

[0174] 针对所述 $P \times N$ 行元素中的每N行元素,依次执行以下操作:

[0175] 基于N行元素,分别对当前获得的N个选定令牌进行推理处理,获得N个推理令牌;其中,所述N行元素中的每行元素表征:在推理过程中,分别对当前获得的N个选定令牌具有不同的关注程度;

[0176] 将所述N个推理令牌分别拼接在所述N个回复令牌序列的尾部,并将所述N个推理令牌作为下一次获得的N个选定令牌;

[0177] 直到执行P次操作,获得N个回复令牌序列。

[0178] 可选的,所述获取单元901,具体用于:

[0179] 获取待回复的N个文本数据;

[0180] 分别针对N个文本数据执行以下操作:将一个文本数据中的各个字符,分别编码为相应的令牌,获得所述一个文本数据对应的初始令牌序列。

[0181] 可选的,所述拼接单元902,具体用于:

[0182] 获取基于所述N个初始令牌序列各自的终止令牌,拼接生成的第一令牌序列;

[0183] 以及,获取基于所述N个初始令牌序列各自除终止令牌以外的令牌,拼接生成的第二令牌序列;

[0184] 将所述第一令牌序列拼接在所述第二令牌序列的末尾,获得拼接令牌序列。

[0185] 可选的,所述第一令牌序列和所述第二令牌序列:分别是按照预设的拼接次序拼接生成的,所述拼接次序表征:所述N个文本数据的处理次序;

[0186] 则所述拼接单元902,用于获取基于所述N个初始令牌序列各自的终止令牌,拼接生成的第一令牌序列,具体用于:

[0187] 基于所述N个文本数据的处理次序,依次对所述N个初始令牌序列各自的终止令牌

进行拼接处理,获得第一令牌序列;

[0188] 则所述拼接单元902,用于所述获取基于所述N个初始令牌序列各自除终止令牌以外的令牌,拼接生成的第二令牌序列,具体用于:

[0189] 基于所述N个文本数据的处理次序,依次将所述N个初始令牌序列各自除终止令牌以外的令牌拼接为第二令牌序列。

[0190] 可选的,所述N个文本数据的处理次序,是采用以下任意一种方式确定的,则所述拼接单元902,还用于:

[0191] 将所述N个文本数据的获取次序,作为所述N个文本数据的处理次序;

[0192] 将所述N个文本数据各自对应的时间戳的时间先后次序,作为所述N个文本数据的处理次序;

[0193] 将所述N个文本数据各自对应的优先级的级别高低次序,作为所述N个文本数据的处理次序。

[0194] 基于上述装置,通过对待处理的N(N为大于1的整数)个文本数据各自对应的初始令牌序列进行拼接,获得拼接令牌序列,再基于注意力机制,对拼接令牌序列进行推理处理,获得用于生成相应回复数据的N回复令牌结果,用以降低推理多条文本数据所消耗的GPU资源,提升计算机设备针对多条文本数据的推理性能。

[0195] 该装置可以用于执行本申请各实施例中的方法,因此,对于该装置的各功能模块所能够实现的功能等可参考前述实施例的描述,不多赘述。

[0196] 请参见图10所示,基于同一技术构思,本申请实施例还提供了一种计算机设备1000,该计算机设备1000可以为图1所示的终端设备或服务器,该计算机设备1000可以包括存储器1001和处理器1002。

[0197] 存储器1001,用于存储处理器1002执行的计算机程序。存储器1001可主要包括存储程序区和存储数据区,其中,存储程序区可存储操作系统、至少一个功能所需的应用程序等;存储数据区可存储根据计算机设备的使用所创建的数据等。处理器1002,可以是一个中央处理单元(central processing unit, CPU),或者为数字处理单元等等。本申请实施例中不限定上述存储器1001和处理器1002之间的具体连接介质。本申请实施例在图10中以存储器1001和处理器1002之间通过总线1003连接,总线1003在图10中以粗线表示,其它部件之间的连接方式,仅是进行示意性说明,并不引以为限。总线1003可以分为地址总线、数据总线、控制总线等。为便于表示,图10中仅用一条粗线表示,但并不表示仅有一根总线或一种类型的总线。

[0198] 存储器1001可以是易失性存储器(volatile memory),例如随机存取存储器(random-access memory, RAM);存储器1001也可以是非易失性存储器(non-volatile memory),例如只读存储器,快闪存储器(flash memory),硬盘(hard disk drive, HDD)或固态硬盘(solid-state drive, SSD)、或者存储器1001是能够用于携带或存储具有指令或数据结构形式的期望的程序代码并能够由计算机存取的任何其他介质,但不限于此。存储器1001可以是上述存储器的组合。

[0199] 处理器1002,用于调用存储器1001中存储的计算机程序时执行本申请各实施例中设备所执行的方法。

[0200] 在一些可能的实施方式中,本申请提供的方法的各个方面还可以实现为一种程序

产品的形式,其包括程序代码,当程序产品在计算机设备上运行时,程序代码用于使计算机设备执行本说明书上述描述的根据本申请各种示例性实施方式的方法中的步骤,例如,计算机设备可以执行本申请各实施例中设备所执行的方法。

[0201] 程序产品可以采用一个或多个可读介质的任意组合。可读介质可以是可读信号介质或者可读存储介质。可读存储介质例如可以是但不限于电、磁、光、电磁、红外线、或半导体的系统、装置或器件,或者任意以上的组合。可读存储介质的更具体的例子(非穷举的列表)包括:具有一个或多个导线的电连接、便携式盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。

[0202] 尽管已描述了本申请的优选实施例,但本领域内的技术人员一旦得知了基本创造性概念,则可对这些实施例做出另外的变更和修改。所以,所附权利要求意欲解释为包括优选实施例以及落入本申请范围的所有变更和修改。

[0203] 显然,本领域的技术人员可以对本申请进行各种改动和变型而不脱离本申请的精神和范围。这样,倘若本申请的这些修改和变型属于本申请权利要求及其等同技术的范围之内,则本申请也意图包含这些改动和变型在内。

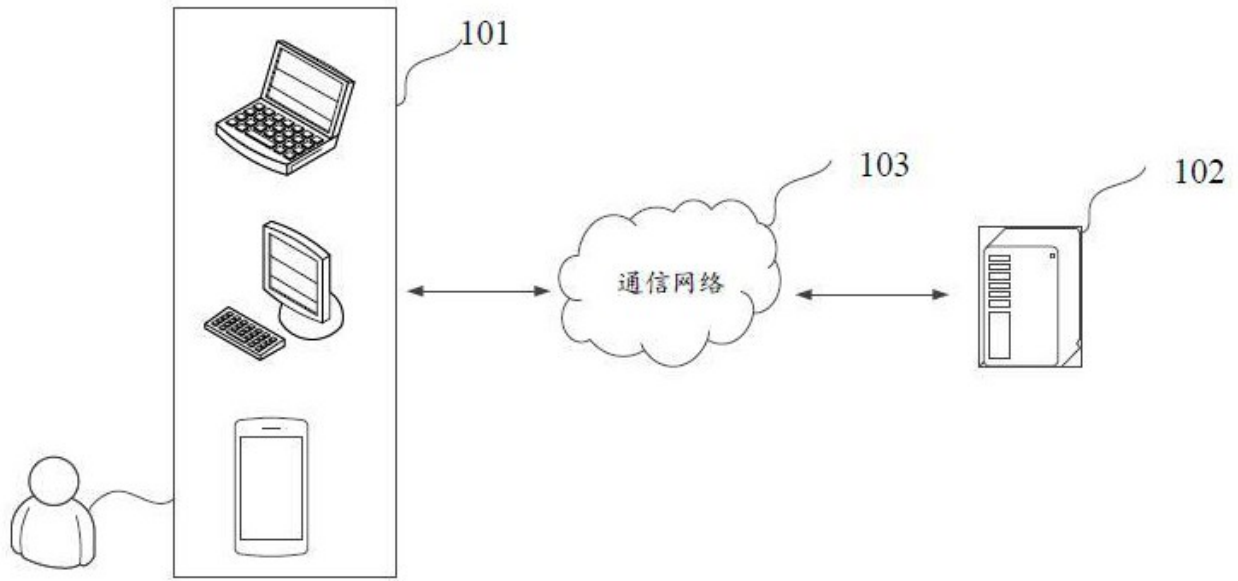


图 1

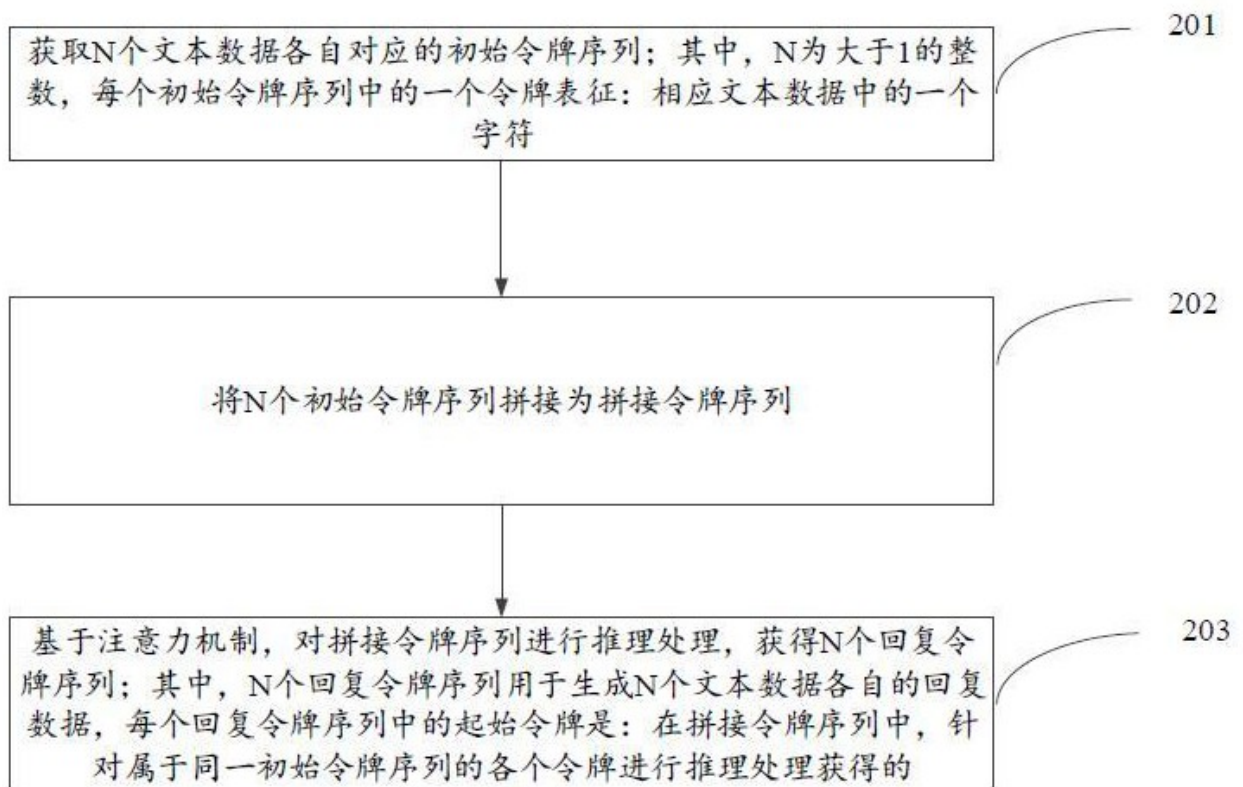


图 2

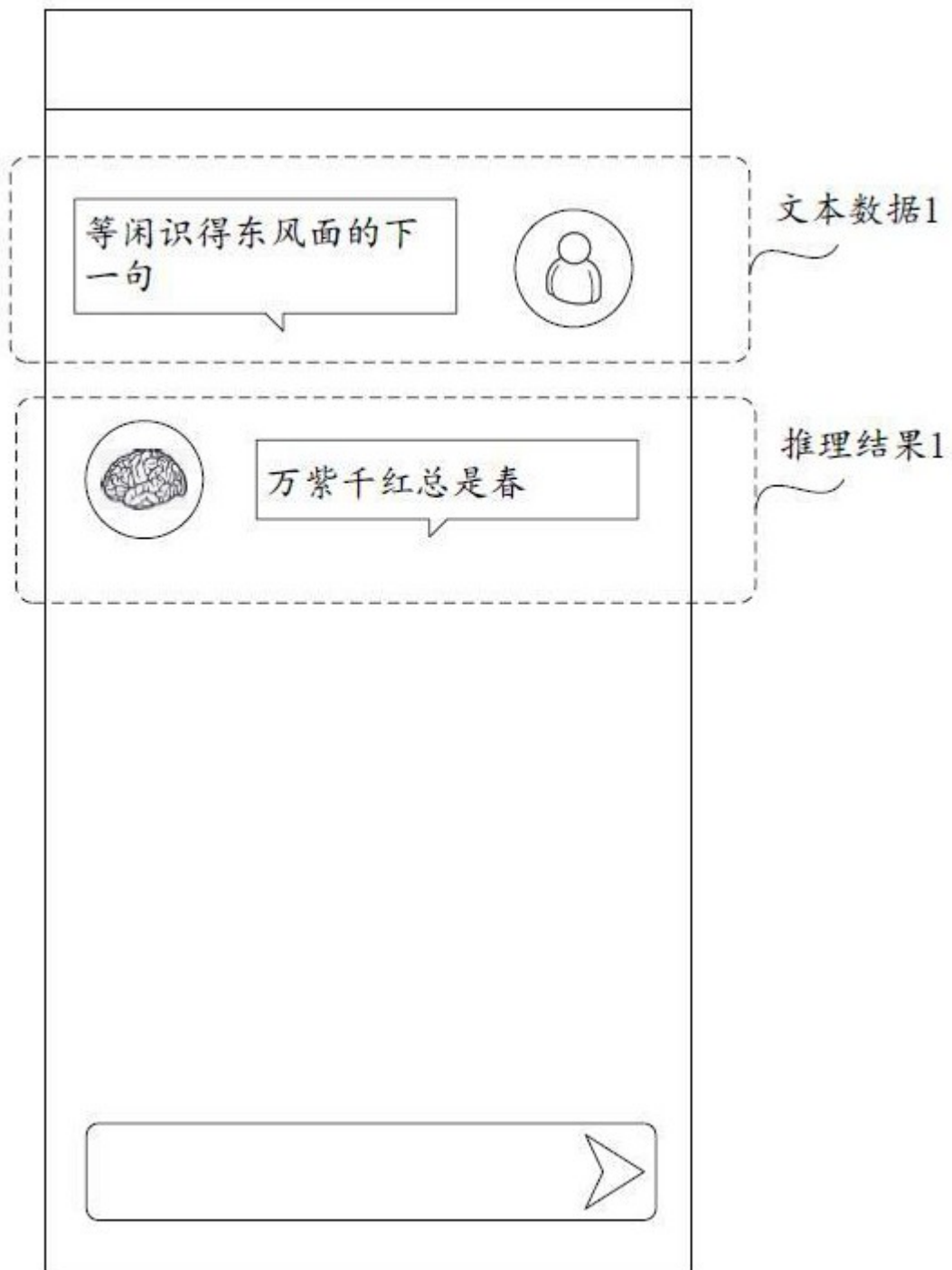


图 3A

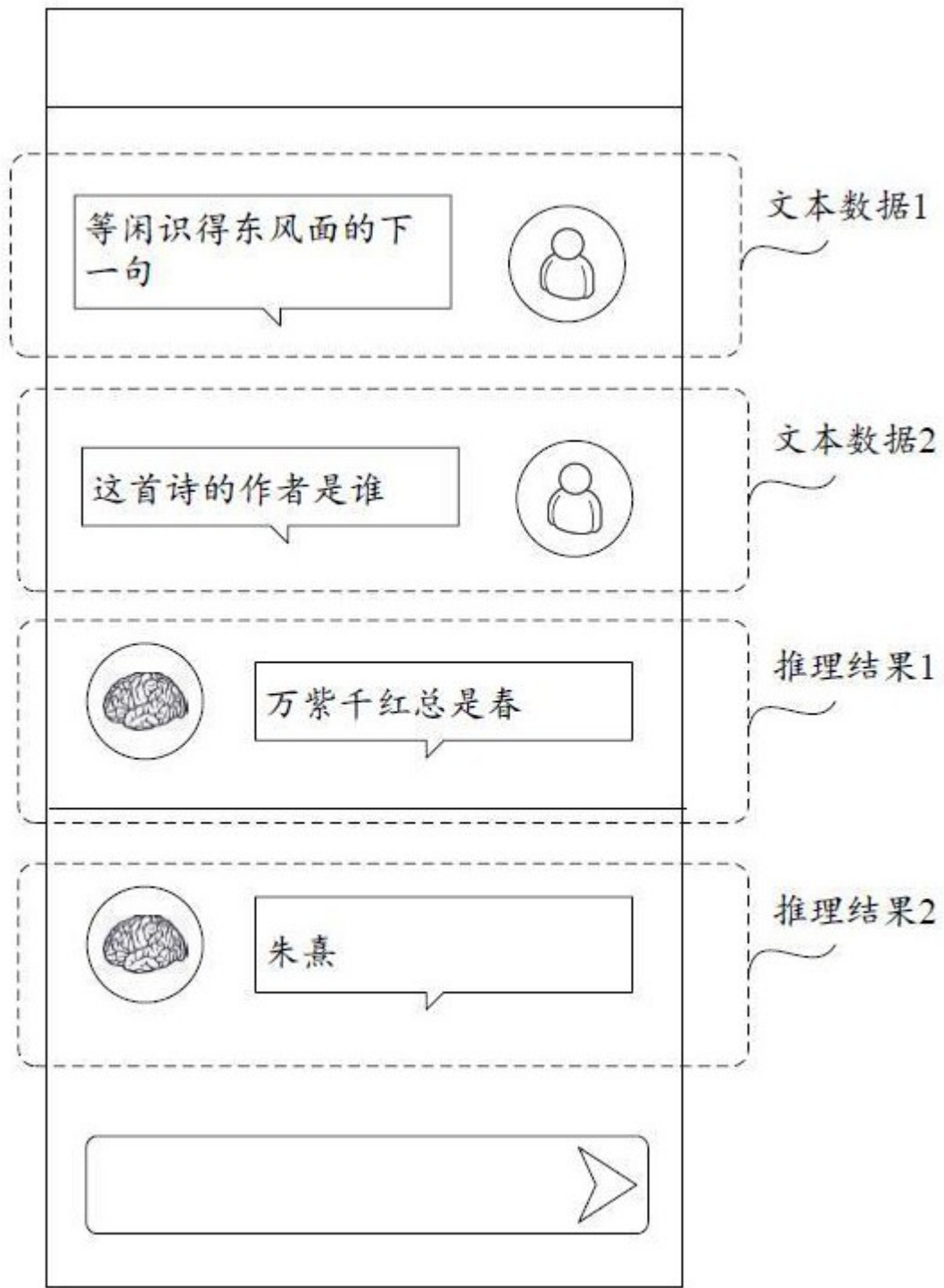


图 3B

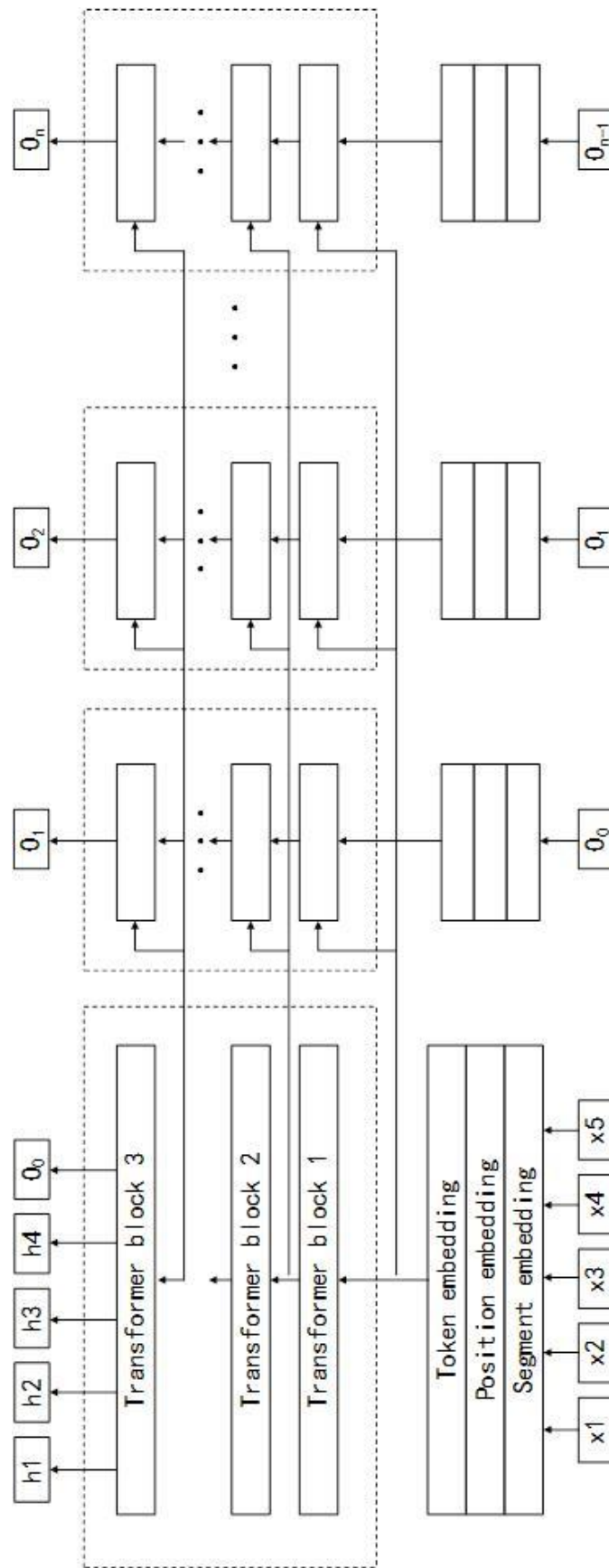


图 4

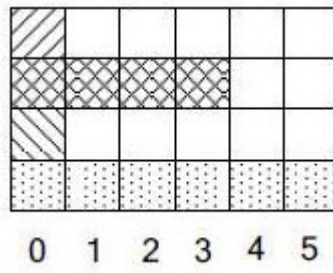


图 5

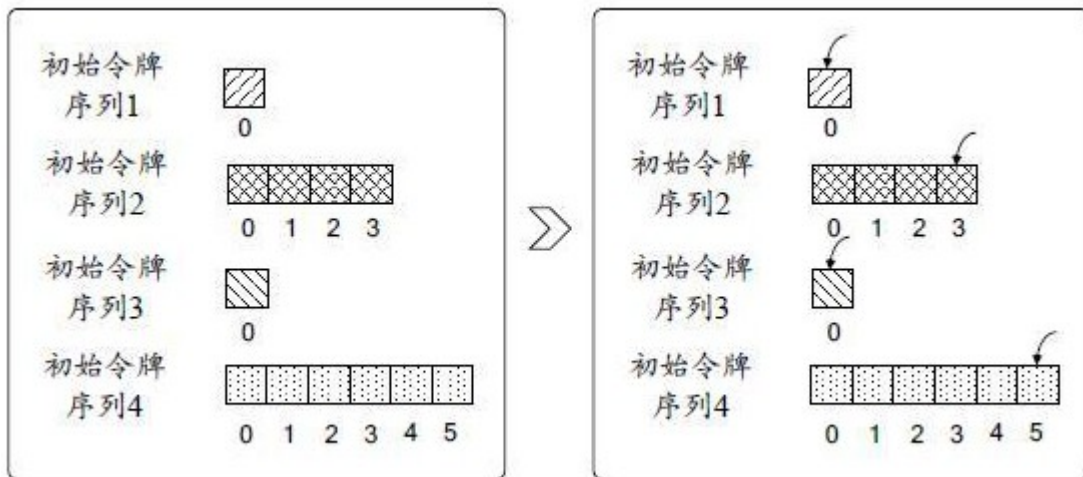


图 6A

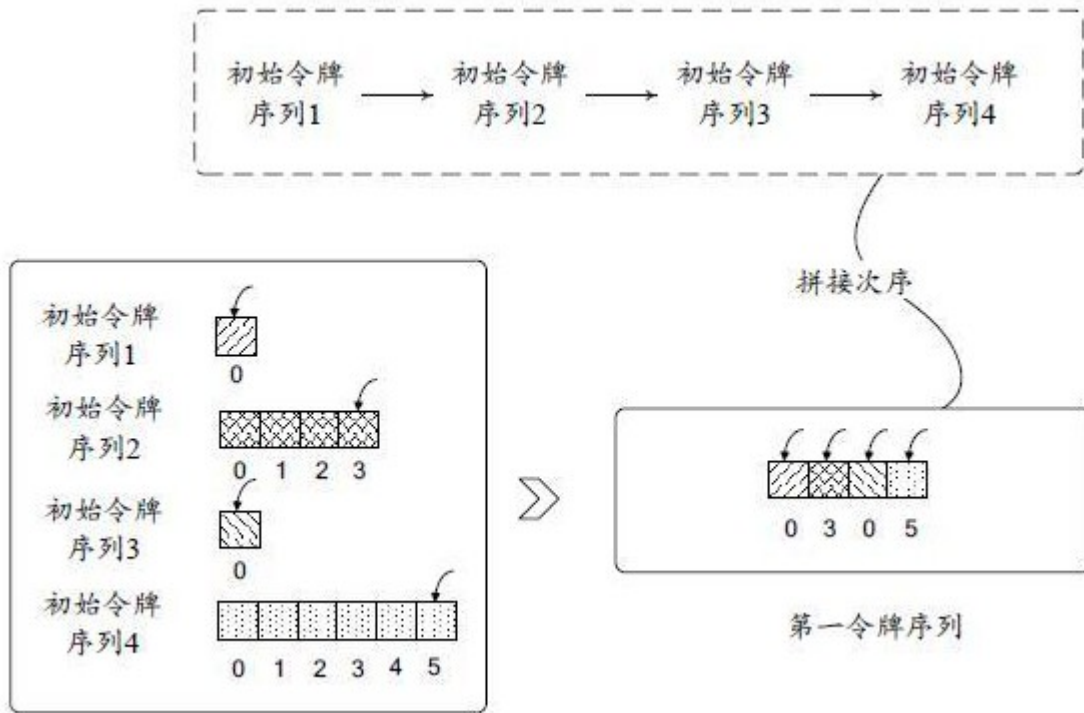


图 6B

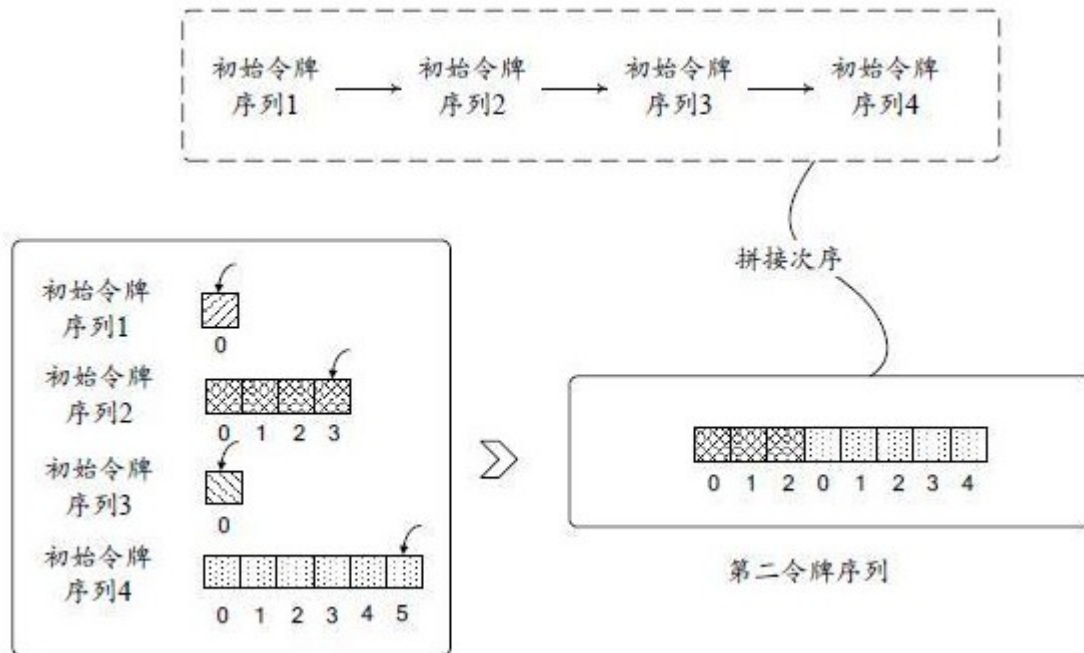


图 6C

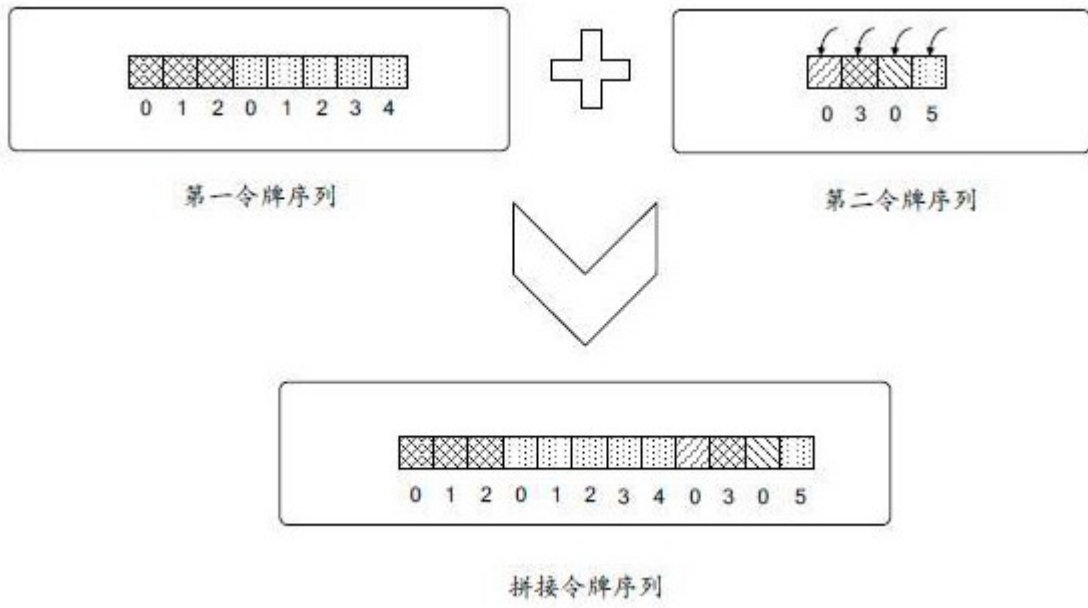


图 6D

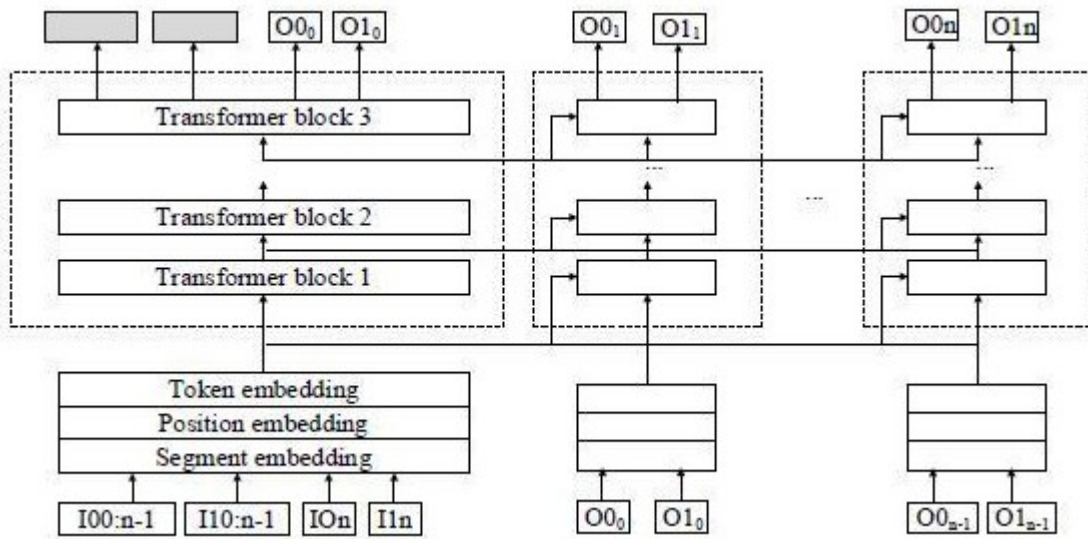


图 7

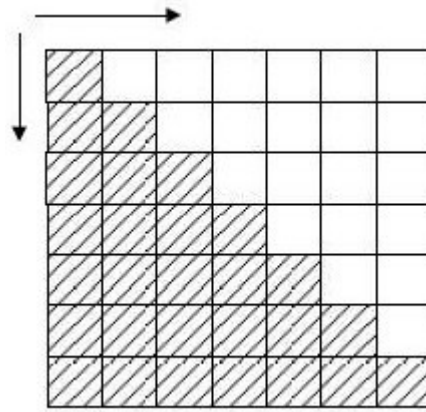


图 8A

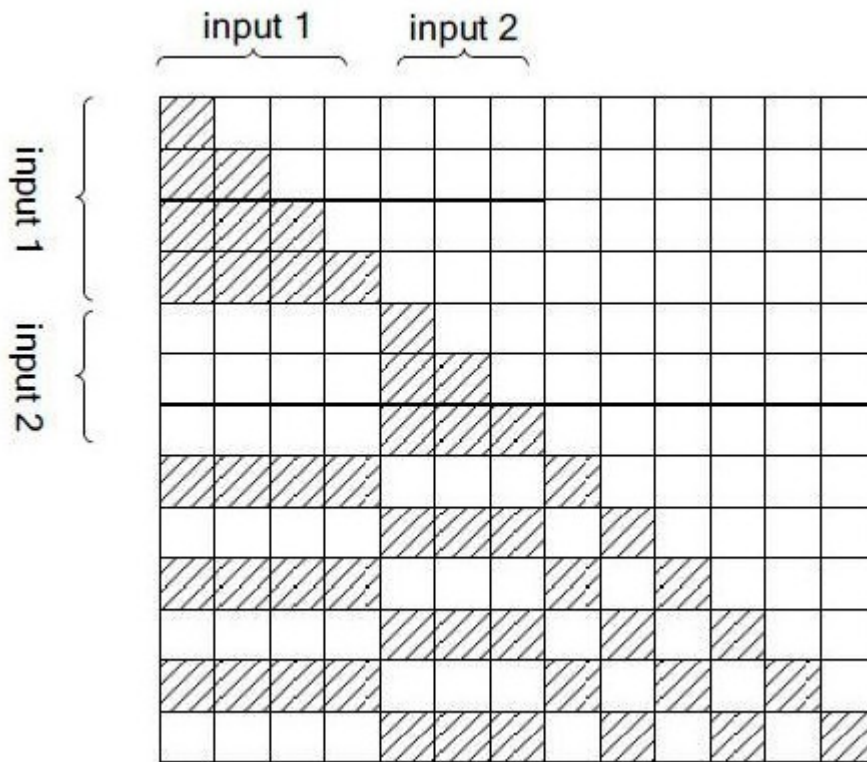


图 8B

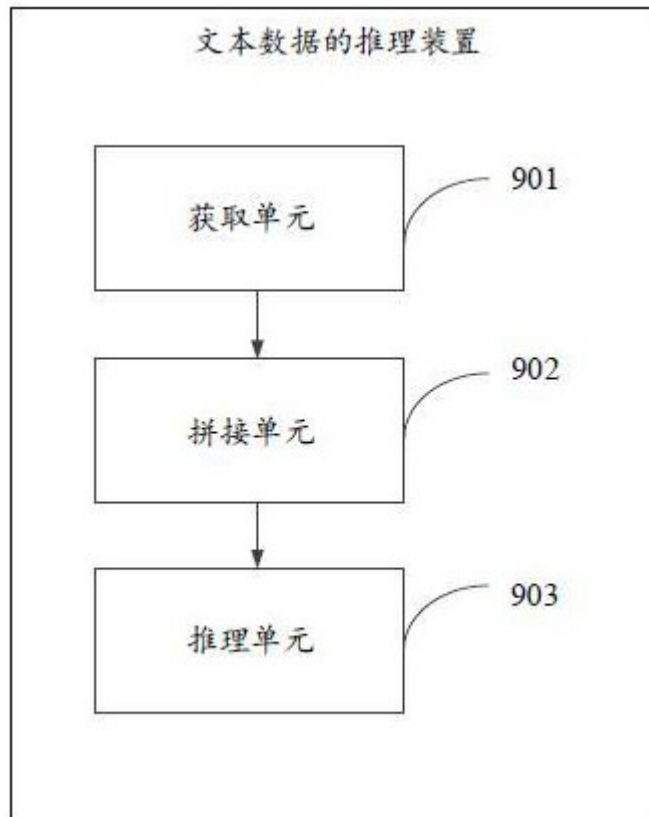


图 9

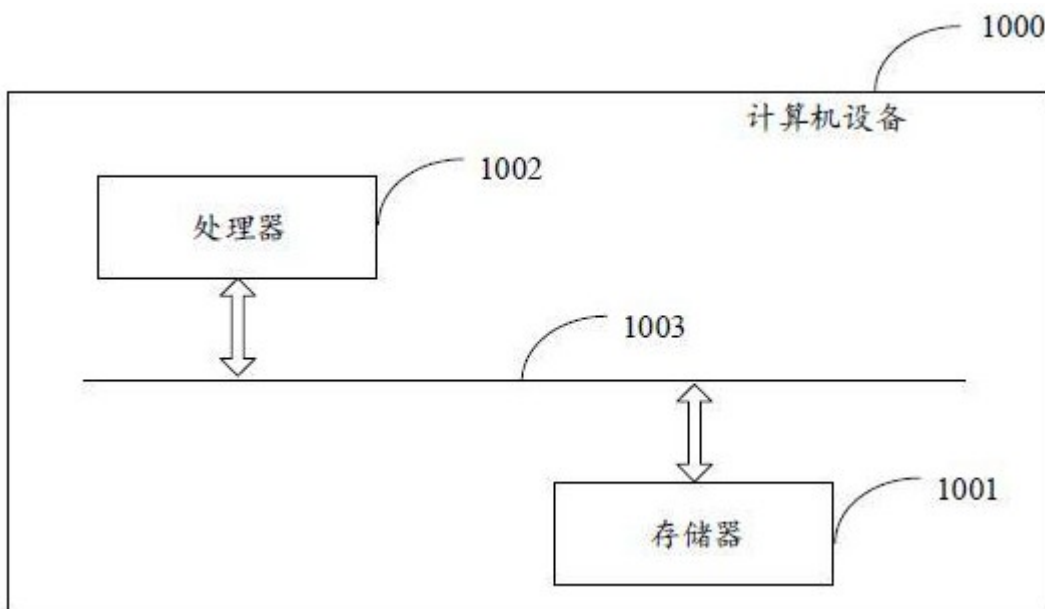


图 10