计算系统、多轮会话推理方法、装置和计算设备集群

本申请要求于2024年01月31日提交的申请号为202410137574.7、发明名称为“一种数据处理的方法、装置及其他设备”，以及，2024年02月28日提交的申请号为202410223741.X、发明名称为“计算系统、多轮会话推理方法、装置和计算设备集群”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

技术领域

本申请涉及云计算技术领域，特别涉及一种计算系统、多轮会话推理方法、装置和计算设备集群。

背景技术

基于Transformer（翻译器）架构的生成式大模型能够应用于多轮会话推理的场景。在一轮会话中，该生成式大模型能够对输入的词元序列token[1:s]（也即是问题提示）进行处理，推理出第s+1个词元token[s+1]，以得到新的词元序列token[1:s+1]，再将token[1:s+1]输入模型中，推理出token[s+2]，以此类推，直至得到完整的词元序列或出现推理终止符，该生成式大模型输出推理得到的词元序列，也即是输出回答，从而完成一轮会话，其中，s为大于1的整数。在推理一个token的过程中，该生成式大模型在处理输入的任一个token时均会产生对应的中间数据key（键）和value（值），而所产生的中间数据还会用在推理后续token的过程中，此外，在发生多轮会话时，每轮会话都会使用到前几轮会话输入和产生的token，进而也会使用到前几轮会话输入和产生的token对应的中间数据，因此，为了节约计算资源和提高推理效率，将前几轮会话输入和产生的token对应的中间数据存储在加速器内部的高带宽内存（high-bandwidth memory，HBM）中，也即是采用加速器内部的HBM来存储会话的历史键值缓存（KV cache），以在进行后续会话时能够直接复用前几轮会话的历史KV cache进行推理，无需进行重新计算。

然而，上述方法中，依赖HBM来存储KV cache，而HBM的容量有限，在面临HBM可用存储空间不足的问题时，加速器会根据业务需求，对HBM中存储的KV cache进行删除，从而导致后续会话无法复用历史KV cache，生成式大模型面临对历史KV cache的巨大重计算开销，仍然会造成推理效率的降低和计算资源的浪费。

发明内容

本申请实施例提供了一种计算系统、多轮会话推理方法、装置和计算设备集群，能够避免进行KV cache重计算，提高多轮会话推理的效率。该技术方案如下。

第一方面，提供了一种计算系统，该计算系统包括主机、该主机的加速器以及外部存储设备；

其中，外部存储设备用于存储已完成的会话的历史KV cache；主机用于在加速器处理会话的同时，将任务队列中待处理会话的历史KV cache从外部存储设备中预加载至主机的内存中，加速器用于在对会话进行生成式大模型的第i层计算的同时，将第i+1层计算所需的历史KV cache从主机的内存中预加载至加速器内。

上述系统中，由于外部存储设备的容量远大于加速器内的HBM的容量，因此，外部存储设备能够保存大量的历史KV cache，进而提高历史KV cache的命中率，从而避免进行KV cache重计算，能够提高多轮会话推理的效率，节约计算资源；并且，由于计算过程与数据加载过程同步进行，因此计算过程无需等待数据加载完成，从而能够隐藏加速器访问外部存储设备的时间开销，提高多轮会话推理的效率。

在一些实施例中，该第一会话的输入在第i+1层的历史KV cache与该第一会话的输入的位置编码无关。

其中，无关是指第一会话的输入的位置编码对第一会话的历史KV cache不产生影响。

在一些实施例中，该第一会话的输入包括的词元数量大于或等于该生成式大模型的上下文窗口所能够容纳的词元数量；

该加速器用于：在对该第一会话进行第i层计算时，从该内存中加载第一子输入在第i+1层的历史KV cache，该第一子输入为该第一会话的输入的子输入，该第一子输入包括的词元数量小于该上下文窗口所能够容纳的词元数量。

上述系统中，所存储的KV cache与位置编码解耦，在上下文窗口溢出时，生成式大模型的输入的位置编码改变不会导致所存储的KV cache失效，因此，加速器能够复用裁剪后的输入的历史KV cache，从而避免了由于上下文窗口溢出导致的KV cache重计算，能够提高推理效率，节约计算资源。

在一些实施例中，该加速器还用于：

将该第一子输入的位置编码嵌入该第一子输入在第i+1层的历史KV cache；基于该第一会话的输入和嵌入位置编码后的该第一子输入在第i+1层的历史KV cache，对该第一会话进行第i+1层计算。

其中，该位置编码为相对位置编码（relative positional encoding，RPE）。

在一些实施例中，该加速器包括第一缓冲区和第二缓冲区，该第一缓冲区用于存储该第一会话的历史KV cache，该第二缓冲区用于存储该任务队列中的第二会话的输入在第1层的历史KV cache，该第二会话为该任务队列中该第一会话之后的第一个待处理会话；

该加速器用于：在对该第一会话进行第i层计算时，从该内存中加载该第一会话的输入在第i+1层的历史KV cache至该第一缓冲区；在对该第一会话进行第N层计算时，从该内存中加载该第二会话的输入在第1层的历史KV cache至该第二缓冲区。

上述系统中，加速器的HBM包括第一缓冲区和第二缓冲区，该第一缓冲区用于存储加速器当前正在处理的会话的KV cache，第二缓冲区用于存储加速器待处理的会话的KV cache，加速器在基于第一缓冲区中的KV cache处理第一会话的同时，将第二会话的第1层计算所需的KV cache加载至第二缓冲区，当加速器完成对第一会话的处理后，基于第二缓冲区中的KV cache处理第二会话，此时，第二缓冲区成为执行缓冲区，加速器在处理第二会话的同时，通过异步线程对第一缓冲区进行释放，使得加速器的计算线程在完成对第一会话的处理后能够立即开始对第二会话进行处理，而无需等待第一缓冲区释放完成，从而隐藏了两个会话之间由于加载下一个会话所需的数据造成的时间空档，提高了推理的效率。

在一些实施例中，该加速器还用于：在对该第一会话进行第i+1层计算时，将对该第一会话进行第i层计算所产生的KV cache写入该内存中。

上述系统中，在加速器进行第i+1层的计算的同时，将第i层计算生成的KV cache写回至主机的内存中，与一轮会话推理完成后再对本轮会话生成的KV cache写入主机的内存相比，能够隐藏相邻两轮会话之间由于写回上一轮会话生成的数据造成的时间空档，能够提高推理的效率。

在一些实施例中，该加速器包括第三缓冲区，该第三缓冲区用于存储该生成式大模型处理该第一会话所产生的KV cache；

该主机的加速器用于：若完成对该第一会话的第N层计算时，该生成式大模型处理该第一会话所生成的KV cache中存在尚未写入该内存的KV cache，则将该尚未写入该内存的KV cache写入该第三缓冲区；在通过该生成式大模型处理该任务队列中的第二会话时，将该第三缓冲区中的KV cache写入该内存。

上述系统中，加速器的HBM还包括第三缓冲区，加速器在完成对第一会话的处理后，将在处理第一会话的过程中产生的且尚未写回内存的KV cache由第一缓冲区写入第三缓冲区，由于第一缓冲区和第三缓冲区之间的数据拷贝效率较高，因此，加速器在完成对第一会话的处理后，能够快速释放第一缓冲区，从而能够加载后续处理所需的KV cache至第一缓冲区，而无需等待将处理第一会话生成的KV cache全部写入内存再释放第一缓冲区，隐藏了两个会话之间由于等待上一个会话的数据写回造成的时间空档，提高了推理的效率。

在一些实施例中，该主机用于：若该内存中不存在该任务队列中第一待处理会话的历史KV cache，则从该外部存储设备中加载该第一待处理会话的历史KV cache至该内存，该第一待处理会话为该任务队列中队头第一数量的待处理会话。

上述系统中，主机能够在加速器处理会话的同时，将任务队列中待处理会话的历史KV cache从外部存储设备中预加载至主机的内存中，由于计算过程与数据加载过程同步进行，因此计算过程无需等待数据加载完成，从而能够隐藏加速器访问外部存储设备的时间开销，提高多轮会话推理的效率。

在一些实施例中，该第一待处理会话包括的会话轮数基于该内存的容量确定。

在一些实施例中，该主机用于：若该内存的可用存储空间足够，则从该外部存储设备中加载该第一待处理会话的历史KV cache至该内存；若该内存的可用存储空间不足，则将该任务队列中第二待处理会话的历史KV cache从该内存写入该外部存储设备中，从该外部存储设备中加载该第一待处理会话的历史KV cache至该内存，该第二待处理会话为该任务队列中除该第一待处理会话之外的会话。

在一些实施例中，该主机还用于：若该外部存储设备的可用存储空间不足，则将该任务队列中第三待处理会话的历史KV cache从该外部存储设备中删除，该第三待处理会话为该任务队列中队尾第二数量的待处理会话。

第二方面，提供了一种多轮会话推理方法，应用于计算系统，该计算系统包括主机、该主机的加速器以及外部存储设备，该外部存储设备用于存储会话的历史键值缓存KV cache，该主机用于从该外部存储设备中加载任务队列中待处理会话的历史KV cache至该主机的内存，该方法包括：

通过该加速器，基于该任务队列，通过生成式大模型对该任务队列中的第一会话进行推理，该第一会话为多轮会话中的一轮，该生成式大模型包括N层，其中，在通过该加速器对该第一会话进行第i层计算时，从该内存中加载该第一会话的输入在第i+1层的历史KV cache至该加速器，i为大于或等于1且小于N的整数。

在一些实施例中，该第一会话的输入在第i+1层的历史KV cache与该第一会话的输入的位置编码无关。

在一些实施例中，该第一会话的输入包括的词元数量大于或等于该生成式大模型的上下文窗口所能够容纳的词元数量；

在对该第一会话进行第i层计算时，从该内存中加载该第一会话的输入在第i+1层的历史KV cache，包括：在对该第一会话进行第i层计算时，从该内存中加载第一子输入在第i+1层的历史KV cache，该第一子输入为该第一会话的输入的子输入，该第一子输入包括的词元数量小于该上下文窗口所能够容纳的词元数量。

在一些实施例中，该通过生成式大模型处理该任务队列中的第一会话，包括：

将该第一子输入的位置编码嵌入该第一子输入在第i+1层的历史KV cache；基于该第一会话的输入和嵌入位置编码后的该第一子输入在第i+1层的历史KV cache，对该第一会话进行第i+1层计算。

在一些实施例中，该加速器包括第一缓冲区和第二缓冲区，该第一缓冲区用于存储该第一会话的历史KV cache，该第二缓冲区用于存储该任务队列中的第二会话的输入在第1层的历史KV cache，该第二会话为该任务队列中该第一会话之后的第一个待处理会话；

在对该第一会话进行第i层计算时，从该内存中加载该第一会话的输入在第i+1层的历史KV cache，包括：在该加速器对该第一会话进行第i层计算时，通过该加速器从该内存中加载该第一会话的输入在第i+1层的历史KV cache至该第一缓冲区。

在一些实施例中，该方法还包括：

在该加速器对该第一会话进行第N层计算时，通过该加速器从该内存中加载该第二会话的输入在第1层的历史KV cache至该第二缓冲区。

在一些实施例中，该方法还包括：

在该加速器对该第一会话进行第i+1层计算时，通过该加速器将对该第一会话进行第i层计算所产生的KV cache写入该内存中。

在一些实施例中，该加速器包括第三缓冲区，该第三缓冲区用于存储该生成式大模型处理该第一会话所产生的KV cache；

在对该第一会话进行第i+1层计算时，将对该第一会话进行第i层计算所产生的KV cache写入该内存中，包括：若该加速器完成对该第一会话的第N层计算时，该生成式大模型处理该第一会话所生成的KV cache中存在尚未写入该内存的KV cache，则通过该加速器将该尚未写入该内存的KV cache写入该第三缓冲区；在该加速器通过该生成式大模型处理该任务队列中的第二会话时，通过该加速器将该第三缓冲区中的KV cache写入该内存。

在一些实施例中，该方法还包括：若该内存中不存在该任务队列中第一待处理会话的历史KV cache，则通过该主机从该外部存储设备中加载该第一待处理会话的历史KV cache至该内存，该第一待处理会话为该任务队列中队头第一数量的待处理会话。

在一些实施例中，该第一待处理会话包括的会话轮数基于该内存的容量确定。

在一些实施例中，该若该内存中不存在该任务队列中第一待处理会话的历史KV cache，则通过该主机从该外部存储设备中加载该第一待处理会话的历史KV cache至该内存，包括：

若该内存的可用存储空间足够，则通过该主机从该外部存储设备中加载该第一待处理会话的历史KV cache至该内存；若该内存的可用存储空间不足，则通过该主机将该任务队列中第二待处理会话的历史KV

cache从该内存写入该外部存储设备中，从该外部存储设备中加载该第一待处理会话的历史KV cache至该内存，该第二待处理会话为该任务队列中除该第一待处理会话之外的会话。

在一些实施例中，该方法还包括：

若该外部存储设备的可用存储空间不足，则通过该主机将该任务队列中第三待处理会话的历史KV cache从该外部存储设备中删除，该第三待处理会话为该任务队列中队尾第二数量的待处理会话。

第三方面，提供了一种多轮会话推理方法，应用于计算系统中的主机的加速器，该计算系统还包括主机以及外部存储设备，该外部存储设备用于存储会话的历史键值缓存KV cache，该主机用于从该外部存储设备中加载任务队列中待处理会话的历史KV cache至该主机的内存，该方法包括：

基于该任务队列，通过生成式大模型对该任务队列中的第一会话进行推理，该第一会话为多轮会话中的一轮，该生成式大模型包括N层，其中，在对该第一会话进行第i层计算时，从该内存中加载该第一会话的输入在第i+1层的历史KV cache，i为大于或等于1且小于N的整数。

在一些实施例中，该第一会话的输入在第i+1层的历史KV cache与该第一会话的输入的位置编码无关。

在一些实施例中，该第一会话的输入包括的词元数量大于或等于该生成式大模型的上下文窗口所能够容纳的词元数量；

在对该第一会话进行第i层计算时，从该内存中加载该第一会话的输入在第i+1层的历史KV cache，包括：在对该第一会话进行第i层计算时，从该内存中加载第一子输入在第i+1层的历史KV cache，该第一子输入为该第一会话的输入的子输入，该第一子输入包括的词元数量小于该上下文窗口所能够容纳的词元数量。

在一些实施例中，该通过生成式大模型处理该任务队列中的第一会话，包括：

将该第一子输入的位置编码嵌入该第一子输入在第i+1层的历史KV cache；基于该第一会话的输入和嵌入位置编码后的该第一子输入在第i+1层的历史KV cache，对该第一会话进行第i+1层计算。

在一些实施例中，该加速器包括第一缓冲区和第二缓冲区，该第一缓冲区用于存储该第一会话的历史KV cache，该第二缓冲区用于存储该任务队列中的第二会话的输入在第1层的历史KV cache，该第二会话为该任务队列中该第一会话之后的第一个待处理会话；

该在对该第一会话进行第i层计算时，从该内存中加载该第一会话的输入在第i+1层的历史KV cache，包括：在对该第一会话进行第i层计算时，从该内存中加载该第一会话的输入在第i+1层的历史KV cache至该第一缓冲区。

在一些实施例中，该方法还包括：在对该第一会话进行第N层计算时，从该内存中加载该第二会话的输入在第1层的历史KV cache至该第二缓冲区。

在一些实施例中，该方法还包括：在对该第一会话进行第i+1层计算时，将对该第一会话进行第i层计算所产生的KV cache写入该内存中。

在一些实施例中，该加速器包括第三缓冲区，该第三缓冲区用于存储该生成式大模型处理该第一会话所产生的KV cache；

在对该第一会话进行第i+1层计算时，将对该第一会话进行第i层计算所产生的KV cache写入该内存中，包括：若完成对该第一会话的第N层计算时，该生成式大模型处理该第一会话所生成的KV cache中存在尚未写入该内存的KV cache，则将该尚未写入该内存的KV cache写入该第三缓冲区；在通过该生成式大模型处理该任务队列中的第二会话时，将该第三缓冲区中的KV cache写入该内存。

第四方面，提供了一种多轮会话推理方法，应用于计算系统中的主机，该计算系统还包括主机的加速器以及外部存储设备，该外部存储设备用于存储会话的历史键值缓存KV cache，该加速器用于基于该任务队列，通过生成式大模型对该任务队列中的第一会话进行推理，该第一会话为多轮会话中的一轮，该生成式大模型包括N层，其中，在对该第一会话进行第i层计算时，从该内存中加载该第一会话的输入在第i+1层的历史KV cache，i为大于或等于1且小于N的整数；

该方法包括：

从该外部存储设备中加载任务队列中待处理会话的历史KV cache至该主机的内存；存储该待处理会话的历史KV cache。

在一些实施例中，从该外部存储设备中加载任务队列中待处理会话的历史KV cache至该主机的内存，包括：

若该内存中不存在该任务队列中第一待处理会话的历史KV cache，则从该外部存储设备中加载该第一待处理会话的历史KV cache至该内存，该第一待处理会话为该任务队列中队头第一数量的待处理会话。

在一些实施例中，该第一待处理会话包括的会话轮数基于该内存的容量确定。

在一些实施例中，该若该内存中不存在该任务队列中第一待处理会话的历史KV cache，则该主机从该外部存储设备中加载该第一待处理会话的历史KV cache至该内存，包括：

若该内存的可用存储空间足够，则从该外部存储设备中加载该第一待处理会话的历史KV cache至该内存；若该内存的可用存储空间不足，则将该任务队列中第二待处理会话的历史KV cache从该内存写入该外部存储设备中，从该外部存储设备中加载该第一待处理会话的历史KV cache至该内存，该第二待处理会话为该任务队列中除该第一待处理会话之外的会话。

在一些实施例中，该方法还包括：

若该外部存储设备的可用存储空间不足，则将该任务队列中第三待处理会话的历史KV cache从该外部存储设备中删除，该第三待处理会话为该任务队列中队尾第二数量的待处理会话。

第五方面，提供了一种多轮会话推理装置，应用于计算系统中的主机的加速器，该装置包括至少一个功能模块，该至少一个功能模块用于执行如前述第三方面或第三方面中任一种可能的实现方式所提供的多轮会话推理方法。

第六方面，提供了一种多轮会话推理装置，应用于计算系统中的主机，该装置包括至少一个功能模块，该至少一个功能模块用于执行如前述第四方面或第四方面中任一种可能的实现方式所提供的多轮会话推理方法。

第七方面，提供了一种加速器，该加速器包括计算核心和存储器，该加速器用于执行如前述第三方面或第三方面中任一种可能的实现方式所提供的多轮会话推理方法，该存储器用于存储计算数据，该计算核心用于对存储器中存储的计算数据执行计算操作。

第八方面，提供了一种主机，该主机包括处理器和存储器，该主机用于执行如前述第四方面或第四方面中任一种可能的实现方式所提供的多轮会话推理方法。

第九方面，提供了一种计算设备集群，该计算设备集群包括至少一个计算设备，每个该计算设备包括主机、主机的加速器，外部存储设备位于至少一个计算设备，该外部存储设备用于存储会话的历史KV cache，该主机用于从该外部存储设备中加载任务队列中待处理会话的历史KV cache至该主机的内存；

该计算设备集群中的该主机用于执行如前述第四方面或第四方面中任一种可能的实现方式所提供的多轮会话推理方法；

该计算设备集群中的该加速器用于执行如前述第三方面或第三方面中任一种可能的实现方式所提供的多轮会话推理方法。

第十方面，提供了一种包含指令的计算机程序产品，当该指令被计算设备集群运行时，使得该计算设备集群执行如前述第三方面或第三方面中任一种可能的实现方式所提供的多轮会话推理方法，或执行如前述第四方面或第四方面中任一种可能的实现方式所提供的多轮会话推理方法。

第十一方面，提供了一种计算机可读存储介质，包括计算机程序指令，当该计算机程序指令由计算设备集群执行时，该计算设备集群执行如前述第三方面或第三方面中任一种可能的实现方式所提供的多轮会话推理方法，或执行如前述第四方面或第四方面中任一种可能的实现方式所提供的多轮会话推理方法。

本申请在上述各方面提供的实现方式的基础上，还可以进行进一步组合以提供更多实现方式。

附图说明

图1是本申请实施例提供的一种计算系统的结构示意图；

图2是本申请实施例提供的一种计算系统的功能示意图；

图3是本申请实施例提供的一种多轮会话推理方法的流程图；

图4是本申请实施例提供的一种上下文窗口溢出后直接裁剪KV cache的示意图；

图5是本申请实施例提供的一种基于与位置编码解耦的KV cache进行推理的示意图；

图6是本申请实施例提供的一种多轮会话推理方法中计算过程与数据加载过程并行的示意图；

图7是本申请实施例提供的一种多轮会话推理方法中计算过程与数据写回过程并行的示意图；

图8是本申请实施例提供的一种多轮会话推理过程中主机的内存和外部存储设备之间的数据迁移过程示意图；

图9是本申请实施例提供的一种多轮会话推理装置的结构示意图；

图10是本申请实施例提供的一种多轮会话推理装置的结构示意图；

图11是本申请实施例提供的一种计算设备的结构示意图；

图12是本申请实施例提供的一种计算设备集群的示意图；

图13是本申请实施例提供的一种计算设备集群的可能实现方式的示意图。

具体实施方式

为使本申请的目的、技术方案和优点更加清楚，下面将结合附图对本申请实施方式作进一步地详细描述。

首先，本申请涉及基于Transformer架构的生成式大模型在多轮会话推理中的应用，为了便于理解本申请实施例的内容，下面本申请实施例涉及的若干技术术语进行解释说明。

token（词元）：用向量表示的最小语义单位，其中，该最小语义单位例如是字符、字、单词或者短语等等。多个token组成的词元序列作为生成式大模型的输入。在多轮会话推理场景中，输入生成式大模型的词元序列也即是会话的问题提示。生成式大模型对该输入的词元序列进行推理，通过多次迭代，逐个得到问题提示的后续词元，直至得到完整的后续词元序列或出现推理终止符。生成式大模型输出所生成的词元序列，生成的词元序列也即是该问题提示的回答。例如，该生成式大模型的输入的词元序列为X[1:s]，生成式大模型基于该词元序列X[1:s]进行第一次迭代，得到第s+1个词元token[s+1]；生成式大模型基于X[1:s]和token[s+1]，进行第二次迭代，得到第s+2个词元token[s+2]；以此类推，直至得到完整的词元序列或出现推理终止符，该生成式大模型输出推理得到的词元序列，也即是输出回答，从而完成对一轮对话推理，其中，s为大于1的整数。

基于Transformer架构的生成式大模型：包括N个Transformer层，N为大于1的整数。每个Transformer层包括自注意力机制（self-attention）和前馈神经网络（feed forward network，FFN）。

一轮会话推理：在每次迭代中，生成式大模型的N个Transformer层依次对本次迭代输入的词元序列进行处理，得到一个词元。其中，通过生成式大模型的第i层对输入的词元序列进行处理的过程包括：将本次迭代的输入中的每个词元分别与第i层中的自注意力机制对应的模型参数进行投影，产生每个词元对应的中间数据key（键）和value（值）；对key和value进行非线性转换，该非线性转换例如是残差连接和归一化，将非线性转换的结果传递给第i层中的前馈神经网络；第i层中的前馈神经网络对该非线性转换的结果进行处理，得到第i层的处理结果，将第i层的处理结果传递给第i+1层进行处理。其中，i为大于或等于1且小于N的整数。上述通过生成式大模型的第i层对输入的词元序列进行处理，得到第i层的处理结果的过程也即是第i层计算。

多轮会话推理：为了保持会话上下文的语义一致性和保证对当前会话的输入的准确理解，生成式大模型在生成当前会话的回答时，会参考与同一对象的历史会话的输入和输出，也即是多轮会话中的一轮会话的输入为：历史会话中输入的词元序列、历史会话中输出的词元序列以及对象在该轮会话中输入的词元序列。

键值缓存（KV cache）：在生成式大模型的一次迭代中（也即是生成一个词元的过程中），生成式大模型在处理任何一个词元时均会产生对应的中间数据key和value，而所产生的中间数据还会用在后续迭代中，

此外，在发生多轮会话时，每轮会话都会使用到历史会话中输入和输出的词元序列，进而也会使用到历史会话中输入和输出的词元对应的中间数据，因此，为了节约计算资源和提高推理效率，将历史会话中输入和输出的词元对应的中间数据进行存储，形成会话的历史KV cache，以在进行会话推理时能够直接复用会话的历史KV cache进行推理，无需进行重新计算。

预填充阶段：该预填充阶段也即是一轮会话推理过程中的第1次迭代。在预填充阶段，生成式大模型基于该轮会话中输入的词元序列X[1:s]进行第1次迭代，得到token[s+1]，并将该次迭代得到的X[1:s]中每个词元对应的key和value进行存储，形成KV cache[1:s]。

解码阶段：该解码阶段也即是一轮会话推理过程中的第2次迭代至最后一次迭代。在解码阶段的任一次迭代中，生成式大模型计算上一次迭代得到的token的对应的key和value，读取已存储的历史KV cache，基于本次迭代计算得到的key和value和历史KV cache，得到本次迭代得到的token，并将本次得到计算得到的key和value进行存储，也即是对历史KV cache进行更新。

上下文窗口（context window）：生成式大模型能够同时处理的词元的数量。

加速器流（stream）：加速器执行并行计算任务的抽象单元。

执行缓存区（execution buffer）：高带宽内存（high-bandwidth memory，HBM）中用于存储加速器推理计算流当前可处理的KV Cache的区域。

上面介绍了本申请实施例涉及的若干技术术语，下面介绍本申请实施例的实施环境。

图1是本申请实施例提供的一种计算系统的结构示意图，如图1所示，该计算系统包括主机101、主机的加速器102和外部存储设备103，其中，主机101、加速器102和外部存储设备103之间通过有线网络或无线网络进行通信。

其中，该计算系统能够部署在计算设备集群上，该计算设备集群包括至少一个计算设备，该计算设备可以是服务器，例如是中心服务器、边缘服务器，或者是本地数据中心中的本地服务器。在一些实施例中，计算设备也可以是台式机、笔记本电脑或者智能手机等终端设备。

其中，该外部存储设备103例如是固态硬盘（solid state disk，SSD）或机械硬盘(hard disk drive，HDD)、又例如是对象存储服务(object storage service，OBS)、云硬盘(elastic volume service，EVS)或弹性文件服务(scalable file service，SFS)等云存储服务，本申请实施例不对外部存储设备103的类型做出限定。示意性地，在多级存储系统中，外部存储设备103用于存储会话的历史KV cache。

其中，该主机101例如是计算设备的主机，又例如是计算设备集群中用于控制和管理其他计算设备的主计算设备。该主机101的内存例如是动态随机存取内存（dynamic random access memory，DRAM），示意性地，主机101上运行有任务调度器，该任务调度器维护有任务队列，该任务队列用于指示加速器的待处理会话；该主机101的内存用于存储任务队列中待处理会话的历史KV cache；该主机101用于基于该任务队列，将该任务队列中待处理会话的历史KV cache从该外部存储设备101中加载至该主机101的内存中。

其中，该加速器102例如是一个通用中央处理器（central processing unit，CPU）、图形处理器（graphics processing unit，GPU）、交换模块的处理器单元（switching module processor unit，SMPU）、网络处理器单元（network processor unit，NPU）、微处理器，又例如是一个或多个用于实现本申请方案的集成电路，例如，专用集成电路(application-specific integrated circuit，ASIC)，可编程逻辑器件(programmable logic device，PLD）或其组合。上述PLD可以是复杂可编程逻辑器件（complex programmable logic device，CPLD），现场可编程逻辑门阵列（field-programmable gate array，FPGA），通用阵列逻辑（generic array logic，GAL）或其任意组合。本申请实施例不对加速器的类型做出限定。示意性地，该加速器102中的HBM用于存储该第一会话的历史KV cache。该加速器102能够通过生成式大模型对任务队列中的第一会话进行推理，也即是进行生成式大模型第i层的计算，从而生成该第一会话的回答。其中，该第一会话为多轮会话中的一轮，该加速器102在进行第i层计算时，复用该加速器102的HBM中第一会话的输入在第i层的历史KV cache，并且在进行第i层计算时，从内存中将第一会话在第i+1层的历史KV cache加载至HBM中。

在一些实施例中，计算设备集群中的每个计算设备包括主机101、加速器102和外部储存设备103，也即是利用每个计算设备配置的存储设备作为该外部存储设备103；在另一些实施例中，该计算设备集群中每个计算设备包括该主机101和该加速器102，该计算集群中具有存储能力的至少一个计算设备充当该外部存储设备103。本申请实施例对此不做限定。

示例性地，图2是本申请实施例提供的一种计算系统的功能示意图，如图2所示，该计算系统包括控制系统和多级存储系统，该多级存储系统包括该主机101的内存、该加速器102中的HBM以及该外部存储设备103。其中，该控制系统运行在主机101和加速器102上，该控制系统包括任务调度器和键值缓存管理单元。该任务调度器维护有任务队列，该任务队列用于指示加速器的待处理会话；该键值缓存管理单元用于控制主机101的内存和外部存储设备103之间的数据迁移。具体地，该键值缓存管理子单元包括键值缓存拉取子单元和键值缓存放置子单元，该键值缓存拉取子单元用于将任务队列中待处理会话的历史KV cache预拉取至主机101的内存中，以及用于将主机101的内存中的KV cache加载至加速器的HBM中；该键值缓存放置子单元用于在主机101的内存的可用存储空间不足时将KV cache从主机101的内存驱逐至外部存储设备103中，以及用于将加速器处理会话生成的KV cache存储（写回）主机的内存中。需要说明的是，图2所示的对计算系统的功能模块的划分仅是示例性地，本申请实施例不对计算系统的功能模块划分方式做出限定。

在一些实施例中，上述的无线网络或有线网络使用标准通信技术和/或协议。网络通常为数据中心网络中的传输控制协议/网际互连协议（transmission control protocol/internet protocol，TCP/IP）网络以及RDMA网络，如基于聚合以太网的RDMA（RDMA over converged Ethernet，RoCE）网络、无限带宽（InfiniBand，IB）网络等，对此不作限定。在另一些实施例中，还能够使用定制和/或专用数据通信技术取代或者补充上述数据通信技术。

下面介绍本申请实施例提供的一种多轮会话推理方法，该方法应用于上述计算系统，该方法中，外部存储设备存储了已完成的会话的历史KV cache，由于外部存储设备的容量远大于加速器内的HBM的容量，因此，外部存储设备能够保存大量的历史KV cache，进而提高历史KV cache的命中率，从而避免进行KV cache重计算，能够提高多轮会话推理的效率，节约计算资源；同时，主机能够在加速器处理会话的同时，将任务队列中待处理会话的历史KV cache从外部存储设备中预加载至主机的内存中，加速器能够在对会话进行生成式大模型的第i层计算的同时，将第i+1层计算所需的历史KV cache从主机的内存中预加载至加速器内，由于计算过程与数据加载过程同步进行，因此计算过程无需等待数据加载完成，从而能够隐藏加速器访问外部存储设备的时间开销，提高多轮会话推理的效率。

上述方法涉及多轮会话推理过程中，加速器和主机的内存之间的数据迁移过程以及主机的内存和外部存储设备之间的数据迁移过程。下面将分别对这两个数据迁移过程进行介绍。

首先介绍加速器和主机的内存之间的数据迁移过程。图3是本申请实施例提供的一种多轮会话推理方法的流程图，如图3所示，该方法应用于计算系统，该计算系统包括主机、主机的加速器和外部储存设备，该方法包括下述步骤301至步骤308。

步骤301、加速器将第一会话的输入在生成式大模型的第1层的历史KV cache从主机的内存中加载至加速器的第一缓冲区，该生成式大模型包括N层，N为大于1的整数。

其中，第一会话为第一对象发起的多轮会话中的一轮。第一会话的输入包括第一会话的历史会话中输入的词元序列、历史会话中输出的词元序列以及对象在该第一会话中输入的词元序列。其中，同一对象发起的多轮会话具有相同的会话窗口标识。第一会话的历史会话也即是与该第一会话具有相同的会话窗口标识的已完成会话。例如，第一对象发起了三轮会话，第一轮会话和第二轮会话已完成，该第一会话为第三轮会话，其中，第一轮会话中输入的词元序列为Q1，输出的词元序列为A1，第二轮会话中输入的词元序列为Q2，输出的词元序列为A2，第三轮会话输入的词元序列为Q3，则第一会话的输入为[Q1 A1 Q2 A2 Q3]。

其中，第一会话的输入在第1层的历史KV cache也即是第一会话的历史KV cache中与第1层关联的KV cache。该第一缓冲区为加速器的HBM中用于存储加速器当前可处理的KV cache的区域，也即是加速器的执行缓冲区。

其中，在加速器执行对第一会话的第1次迭代的第1层计算之前，该加速器启动数据读取线程，通过该数据读取线程，从主机的内存中加载第一会话的输入在第1层的历史KV cache至该第一缓冲区。上述方法中，在第1层计算发生之前，通过数据读取线程使得该层计算所需的数据在加速器的第一缓冲区中就绪，能够避免由于从主机的内存中加载数据造成的时间空档，从而提高推理效率。

其中，该历史KV cache为长度可变的多维向量。历史KV cache可以按照不同的粒度进行存储。例如，在一些实施例中，以一轮会话为粒度对历史KV cache进行存储，也即是将通过生成式大模型处理多个会话得到的中间数据key和value按照会话存储为多个KV cache，每个KV cache具有会话标识。又例如，在另一些

实施例中，以生成式大模型的一层为粒度对历史KV cache进行存储，也即是将通过生成式大模型多层得到的中间数据key和value按照层存储为多个KV cache，每个KV cache具有会话标识和层标识。再例如，在又一些实施例中，以一组多轮会话为粒度对历史KV cache进行存储，也即是将生成式大模型处理同一对象发起的多轮会话得到的中间数据key和value存储为一个KV cache。需要说明的是，上述对于历史KV cache的存储粒度的说明仅是示例性地，本申请实施例不对KV cache的存储粒度做出限定。

其中，第一会话的输入在第1层的历史KV cache与第一会话的输入的位置编码无关。其中，无关是指第一会话的输入的位置编码对第一会话的历史KV cache不产生影响。

在一些实施例中，第一会话的输入包括的词元数量大于生成式大模型的上下文窗口所能够容纳的词元数量；加速器从主机的内存中加载第一子输入在第1层的历史KV cache，该第一子输入为第一会话的输入的子输入，第一子输入包括的词元数量小于上下文窗口所能够容纳的词元数量。该过程可以理解为，在第一会话的输入造成上下文窗口溢出时，对第一会话的输入进行裁剪，以丢弃第一会话的输入中靠前词元，将第一会话的第一子输入作为生成式大模型的输入，相对应地，加速器从主机的内存中加载第一子输入对应的历史KV cache，也即是上下文窗口溢出后直接裁剪KV cache。下面通过图4对上述加载过程进行举例说明。图4是本申请实施例提供的一种上下文窗口溢出后直接裁剪KV cache的示意图，如图4所示，主机的内存中存储有第一会话的输入在第1层的历史KV cache，第一会话的输入的位置编码为[0:2048]；在第一会话的输入造成上下文窗口溢出时，加速器从主机的内存中加载第一会话的第一子输入在第1层的历史KV cache；第一子输入的位置编码为[0:1536]，加速器将第一子输入的位置编码嵌入第一子输入在第1层的历史KV cache，基于嵌入位置编码后的历史KV cache进行后续推理。其中，加速器将位置编码嵌入历史KV cache是指将位置编码嵌入历史KV cache中的key。

上述实施例中，所存储的KV cache与位置编码解耦，在上下文窗口溢出时，生成式大模型的输入的位置编码改变不会导致所存储的KV cache失效，因此，加速器能够复用裁剪后的输入的历史KV cache，从而避免了由于上下文窗口溢出导致的KV cache重计算，能够提高推理效率，节约计算资源。

需要说明的是，第一会话的历史KV cache存储在外部存储设备中，在加速器通过生成式大模型对第一会话进行推理之前，主机将第一会话的历史KV cache从外部存储设备中预加载至主机的内存中，加速器再从主机的内存中加载第一会话的输入在第1层的历史KV cache。外部存储设备和主机的内存之间的数据迁移过程将在后续实施例中进行介绍，此处不再赘述。

步骤302、加速器基于第一会话的输入，通过生成式大模型，对该第一会话进行第1次迭代的第1层计算，同时，该加速器从主机的内存中加载该第一会话的输入在第2层的历史KV cache至该加速器的第一缓冲区。

其中，加速器启动计算线程，通过该计算线程，对该第一会话进行第1次迭代的第1层计算。

其中，在第一会话的输入包括的词元数量小于生成式大模型的上下文窗口所能够容纳的词元数量的情况下，也即是上下文窗口未溢出的情况下，加速器通过生成式大模型对第一会话进行第1次迭代的第1层计算的过程包括：将第一会话的输入词元序列分别与第1层中的自注意力机制对应的模型参数进行投影，产生第一会话的输入词元序列中每个词元对应的中间数据key和value；基于第一会话的输入词元序列在第1层的key和value，对第一会话的输入在第1层的历史KV cache进行更新；将第一会话的输入的位置编码嵌入该更新后的历史KV cache中，该位置编码为相对位置编码（relative positional encoding，RPE）；对嵌入位置编码后的历史KV cache中的key和value进行非线性转换，该非线性转换包括残差连接和归一化，将非线性转换的结果传递给第1层中的前馈神经网络；通过第1层中的前馈神经网络对该非线性转换的结果进行处理，得到第1层的计算结果。其中，加速器基于第一会话的输入词元序列在第1层的key和value，对第一会话的输入在第1层的历史KV cache进行更新是指：加速器计算线程每计算得到一个中间数据key或value，将该中间数据追加至第一缓冲区中第一会话的输入在第1层的历史KV cache中。

下面通过图5对上述第1层的计算过程进行举例说明，图5是本申请实施例提供的一种基于与位置编码解耦的KV cache进行推理的示意图，如图5所示，加速器将第一会话的输入词元序列（input）分别与第1层的自注意力机制对应的模型参数（W^k、W^v）进行投影，产生第一会话的输入词元序列中每个词元对应的中间数据key和value；加速器对历史KV cache进行更新；加速器将第一会话的输入的位置编码嵌入更新后的历史KV cache中的key中，并基于嵌入位置编码的KV cache进行后续推理。

其中，在第一会话的输入包括的词元数量大于或等于生成式大模型的上下文窗口所能够容纳的词元数

量的情况下，也即是上下文窗口溢出的情况下，加速器基于第一会话的第一子输入，通过生成式大模型，对该第一会话进行第1次迭代的第1层计算，该计算过程与上下文窗口未溢出的情况下的计算过程同理，不同之处在于，在上下文窗口溢出的情况下，生成式大模型的输入和加速器从主机的内存中加载的历史KV cache是裁剪之后的，同理之处不再赘述。

其中，加速器从主机的内存中加载第一会话的输入在第2层的历史KV cache至第一缓冲区的过程与步骤301中加载第1层的历史KV cache的过程同理，不再赘述。

步骤303、加速器对该第一会话进行第1次迭代的第2层计算，同时，该加速器从主机的内存中加载该第一会话的输入在第3层的历史KV cache至第一缓冲区，将第1次迭代的第1层计算得到的KV cache写入主机的内存中。

其中，加速器通过计算线程对该第一会话进行第1次迭代的第2层计算，同时，加速器通过数据读取线程从主机的内存中加载第一会话的输入在第3层的历史KV cache，加速器通过数据写回线程将第1次迭代的第1层计算得到的KV cache从第一缓冲区中写入主机的内存中。

其中，加速器通过计算线程，基于对该第一会话进行第1次迭代的第1层计算的计算结果，对第一会话进行第1次迭代的第2层计算，第2层的计算过程与第1层的计算过程同理，不再赘述。

其中，加速器通过数据读取线程，从主机的内存中加载第一会话的输入在第3层的历史KV cache至第一缓冲区的过程与步骤301中加载第1层的历史KV cache的过程同理，不再赘述。

上述步骤303中，在加速器的计算线程正在进行某一层的计算时，加速器的数据读取线程同时开始将后续层的KV cache加载到第一缓冲区，在当前层的计算完成后，计算线程能够立即基于已加载的数据开始进行后续层的计算，隐藏了相邻两层之间由于加载下一层所需的数据造成的时间空档，从而提高了会话推理的效率；并且，在加速器的计算线程正在进行某一层的计算时，加速器的数据写回线程同时上一层生成的KV cache写回至主机的内存中，与一次迭代完成后再对本次迭代生成的KV cache写入主机的内存相比，能够隐藏相邻两次迭代之间由于写回上一次迭代生成的数据造成的时间空档，能够提高推理的效率。

步骤304、加速器按照与步骤303同理的方式，对该第一会话依次进行第1次迭代的第3层计算至第N层计算，推理出第一会话的输出中的第1个词元。

上述步骤301至步骤304介绍了对第一会话进行第1次迭代的过程，该第1次迭代也即是第一会话推理过程中的预填充阶段，经过该预填充阶段，推理出第一会话的输出中的第1个词元。下面通过步骤305对第一会话推理过程中的解码阶段进行介绍，其中，该解码阶段包括至少一次迭代，每一次迭代推理出第一会话的输出中的一个词元。

步骤305、加速器按照与上述步骤301至步骤304同理的方式，对该第一会话依次进行第2次迭代至第M次迭代，依次推理出第一会话的输出中的第2个词元至第M个词元，M等于对第一会话的总迭代次数，M为大于或等于2整数。

其中，第2次迭代至第M次迭代中任一次迭代与第1次迭代同理，不同之处在于，第1次迭代中每层计算产生第一会话输入的词元序列（第一会话的问题提示）中每个词元对应的中间数据key和value，而第2次迭代至第M次得到中任一次迭代中，每层计算产生上一次迭代得到的词元对应的中间数据key和value，同理之处不再赘述。

步骤306、在加速器对该第一会话进行第M次迭代的第N层计算的同时，加速器从主机的内存中加载任务队列中第二会话的输入在第1层的历史KV cache至加速器的第二缓冲区。

其中，该任务队列用于指示待处理会话。该第二会话为任务队列中的第一个待处理会话。该第二缓冲区为加速器的HBM中用于存储加速器待处理的KV cache的区域，也即是用于存储第二会话的输入在第1层的历史KV cache。

其中，加速器通过数据读取线程从主机的内存中加载第二会话的输入在第1层的历史KV cache至该第二缓冲区。

上述方法中，加速器的HBM包括第一缓冲区和第二缓冲区，该第一缓冲区用于存储加速器当前正在处理的会话的KV cache，第二缓冲区用于存储加速器待处理的会话的KV cache，加速器在基于第一缓冲区中的KV cache处理第一会话的同时，将第二会话的第1层计算所需的KV cache加载至第二缓冲区，当加速器完成对第一会话的处理后，加速器基于第二缓冲区中的KV cache处理第二会话，此时，第二缓冲区成为执行缓冲区，加速器在处理第二会话的同时，通过异步线程对第一缓冲区进行释放，使得加速器的计算线程在

完成对第一会话的处理后能够立即开始对第二会话进行处理，而无需等待第一缓冲区释放完成，从而隐藏了两个会话之间由于加载下一个会话所需的数据造成的时间空档，提高了推理的效率。

需要说明的是，上述步骤306为可选步骤，在一些实施例中，加速器仅包括第一缓冲区；加速器在完成对第一会话的第M次迭代的第N层计算后，对第一缓冲区进行释放，再将第二会话的输入在第1层的历史KV cache加载至该第一缓冲区。本申请实施例对此不做限定。上述实施例中，加速器的HBM全部用作第一缓冲区，能够提高第一缓冲区的容量，在会话的历史KV cache数据量较大时，能够容纳会话的完整的历史KV cache，能够避免由于容量不足导致的后加载的历史KV cache覆盖先加载的历史KV cache的情况，从而能够提高第一缓冲区中历史KV cache的命中率，避免历史KV cache的重复加载。

步骤307、若在加速器完成对第一会话的第M次迭代的第N层计算时，该生成式大模型处理该第一会话所生成的KV cache中存在尚未写入该内存的KV cache，则加速器将尚未写入内存的KV cache从第一缓冲区写入加速器的第三缓冲区。

其中，该生成式大模型处理该第一会话所生成的KV cache存储在第一缓冲区中，在加速器通过计算线程进行第i+1层的计算时，加速器的数据写回线程将第i层计算生成的KV cache从该第一缓冲区写回主机的内存中，i为大于或等于1且小于N的整数。该第三缓冲区为加速器的HBM中用于存储尚未写回的KV cache的区域。

上述方法中，加速器的HBM还包括第三缓冲区，加速器在完成对第一会话的处理后，将在处理第一会话的过程中产生的且尚未写回内存的KV cache由第一缓冲区写入第三缓冲区，由于第一缓冲区和第三缓冲区之间的数据拷贝效率较高，因此，加速器在完成对第一会话的处理后，加速器能够快速地将第一缓冲区中尚未写回内存的KV cache写入第三缓冲区中，进而能够快速释放第一缓冲区，从而能够加载后续处理所需的KV cache至第一缓冲区，而无需等待将处理第一会话生成的KV cache全部写入内存再释放第一缓冲区，隐藏了两个会话之间由于等待上一个会话的数据写回造成的时间空档，提高了推理的效率。

需要说明的是，上述步骤307为可选步骤，在一些实施例中，加速器仅包括第一缓冲区，加速器在完成对第一会话的第M次迭代的第N层计算后，待处理该第一会话所生成的KV cache全部从第一缓冲区写入主机的内存后，再对该第一缓冲区进行释放。上述实施例中，加速器的HBM全部用作第一缓冲区，能够提高第一缓冲区的容量，在会话的历史KV cache数据量较大时，能够容纳会话的完整的历史KV cache，能够避免由于容量不足导致的后加载的历史KV cache覆盖先加载的历史KV cache的情况，从而能够提高第一缓冲区中历史KV cache的命中率，避免历史KV cache的重复加载。

步骤308、加速器按照与上述步骤301至步骤307同理的方式，对第二会话进行处理，同时，将加速器的第三缓冲区中的KV cache写入主机的内存。

其中，该步骤308中，加速器将第三缓冲区中的KV cache写入主机的内存的过程与上述步骤303中加速器将第1次迭代的第1层计算得到的KV cache写入主机的内存中的过程同理，不再赘述。

上述方法中，加速器能够在对会话进行生成式大模型的第i层计算的同时，将第i+1层计算所需的历史KV cache从主机的内存中预加载至加速器内，由于计算过程与数据加载过程同步进行，因此计算过程无需等待数据加载完成，从而能够隐藏加速器访问外部存储设备的时间开销，提高多轮会话推理的效率；进一步地，在加速器进行第i+1层的计算的同时，将第i层计算生成的KV cache写回至主机的内存中，与一轮会话推理完成后再对本轮会话生成的KV cache写入主机的内存相比，能够隐藏相邻两轮会话之间由于写回上一轮会话生成的数据造成的时间空档，能够提高推理的效率。

下面通过图6和图7对上述步骤301至步骤308所示的流程进行示例性说明。图6是本申请实施例提供的一种多轮会话推理方法中计算过程与数据加载过程并行的示意图，如图6所示，生成式大模型包括3层（L1、L2和L3），在上一会话处理完成后，加速器通过该生成式大模型处理当前会话，其中，在L1层的计算发生之前，该层所需的KV cache已在加速器的HBM中的第一缓冲区中就绪。其中，加速器的数据加载线程，也即是KV cache读取流，将L1层计算所需的KV cache加载至第一缓冲区中；加速器的计算线程，也即是执行流，开始进行L1层的计算；在加速器的计算线程正在进行L1层的计算时，数据加载线程同时开始将L2层计算所需的KV cache加载到第一缓冲区中，以此类推，使得加速器从主机的内存中加载KV cache的时间与计算时间相互重叠，从而能够隐藏从主机的内存中加载KV cache的时间开销，显著地提高推理效率。

图7是本申请实施例提供的一种多轮会话推理方法中计算过程与数据写回过程并行的示意图，如图7所示，生成式大模型包括3层（L1、L2和L3），在预填充阶段，加速器的计算线程，也即是执行流，进行L1层

的计算；在加速器的计算线程完成L1层的计算后，加速器的计算线程进行L2层的计算，同时，加速器的数据写回线程，也即是KV cache写回流，将L1层计算生成的KV cache从第一缓冲区中写回主机的内存中；在加速器的计算线程完成L2层的计算后，加速器的计算线程进行L3层的计算，同时，加速器的数据写回线程，将L2层计算生成的KV cache从第一缓冲区中写回主机的内存中，以此类推；解码阶段的数据写回过程与预填充阶段的数据写回过程同理，不再赘述。在图7所示的数据写回过程中，加速器将生成的KV cache写回主机的内存的时间与计算时间相互重叠，从而能够隐藏将生成的KV cache写回主机的内存的时间开销，显著地提高推理效率。

上面介绍了加速器和主机的内存之间的数据迁移过程，下面介绍主机的内存和外部存储设备之间的数据迁移过程。

其中，主机的内存和外部存储设备之间的数据迁移过程包括：若主机的内存中不存在任务队列中第一待处理会话的历史KV cache，则主机从外部存储设备中加载该第一待处理会话的历史KV cache至主机的内存，该第一待处理会话为任务队列中队头第一数量的待处理会话。

其中，第一待处理会话包括的会话轮数基于主机的内存的容量确定，也即是该第一数量基于主机的内存的容量确定。例如，第一数量=主机的内存的可用存储空间÷会话的历史KV cache的平均数据量。需要说明的是，对于加速器正在处理的第一会话的历史KV cache，加速器在进行计算的同时，将新生成的KV cache写回至主机的内存中，以对主机的内存中第一会话的历史KV cache进行更新，主机的内存中第一会话的历史KV cache所占用的空间属于系统占用，该空间在第一会话处理完成之前不能被释放，因此，该主机的内存的可用存储空间=主机的内存容量-系统占用的空间的容量。

在一些实施例中，若主机的内存的可用存储空间足够，则主机从外部存储设备中加载第一待处理会话的历史KV cache至该主机的内存；若主机的内存的可用存储空间不足，则主机将任务队列中第二待处理会话的历史KV cache从主机的内存写入外部存储设备中，从外部存储设备中加载第一待处理会话的历史KV cache至主机的内存，该第二待处理会话为任务队列中除第一待处理会话之外的会话。

其中，本领域技术人员能够根据实际需要确定如何判断主机的内存的可用存储空间是否足够，例如，在一些实施例中，若主机的内存的可用存储空间容量大于或等于目标阈值，则说明主机的可用存储空间足够；若主机的内存的可用存储空间容量小于该目标阈值，则说明主机的可用存储空间不足。又例如，在另一些实施例中，若主机的内存的可用存储空间容量大于或等于第一待处理会话的历史KV cache的数据量，则说明主机的可用存储空间足够；若主机的内存的可用存储空间容量小于第一待处理会话的历史KV cache的数据量，则说明主机的可用存储空间不足。需要说明的是，上述对于主机的内存的可用存储空间是否足够的判断方式的说明仅是示例性地，本申请实施例不对该判断方式做出限定。

在一些实施例中，在主机的内存的可用存储空间之外，主机的内存还包括第四缓冲区，该第四缓冲区用于：在主机的内存的可用存储空间不足时，存储从外部存储设备中加载的历史KV cache。通过该实施例，主机的内存包括第四缓冲区，确保了将KV cache从外部存储设备迁移至主机的内存时有足够的迁移空间，从而在主机的可用存储空间不足时，主机能够将从外部存储设备中预加载的KV cache写入该第四缓冲区，而无需等待将主机的内存中的KV cache驱逐至外部存储设备，避免主机的可用存储空间不足时等待主机的内存驱逐KV cache造成的阻塞。

在一些实施例中，若外部存储设备的可用存储空间不足，则将任务队列中第三待处理会话的历史KV cache从外部存储设备中删除，该第三待处理会话为任务队列中队尾第二数量的待处理会话。其中，该第二数量可以根据实际情况确定，例如，第二数量为1或2等等，本申请实施例对此不做限定。

下面通过图8对主机的内存和外部存储设备之间的数据迁移过程进行举例说明。图8是本申请实施例提供的一种多轮会话推理过程中主机的内存和外部存储设备之间的数据迁移过程示意图，如图8所示，加速器正在处理Job1，该Job1也即是第一会话；任务队列包括8个待处理会话，从队头至队尾依次为Job2至Job9；主机的内存包括第四缓冲区（图8中“buf”所指示的区域），且该主机的内存中存储有Job1、Job2和Job4的历史KV cache，分别为KV1、KV2和KV4；外部存储设备为磁盘（disks），其中存储有Job9、Job8、Job7和Job3的历史KV cache，分别为KV9、KV8、KV7和KV3。主机中的任务调度器维护有前视窗口（prefetching window），该前视窗口用于确定任务队列中队头第一数量的待处理会话，也即是该第一待处理会话，图8中，该前视窗口的大小为2，指示第一待处理会话包括两轮会话，也即是该第一数量为2，如图8所示，该第一待

处理会话为Job2和Job3；主机中的任务调度器还维护有淘汰豁免窗口（eviction window），当外部存储设备的可用存储空间不足时，该淘汰豁免窗口所指示的会话将被豁免从外部存储设备设备中删除，图8中，该淘汰豁免窗口的大小为6。如图8所示，第一待处理会话（Job2和Job3）中，Job2的历史KV cache（KV2）存在于主机的内存中、Job3的历史KV cache（KV3）不存在于主机的内存中且主机的内存的可用存储空间不足，则主机将KV3从磁盘中加载至主机的内存中的第四缓冲区中，并将任务队列中第二待处理会话的历史KV cache从主机的内存写入磁盘中，其中，该第二待处理会话的历史KV cache存在于主机的内存中且该第二待处理会话不是该第一待处理会话，该第二待处理会话不是该第一待处理会话是指该第二待处理会话在该前视窗口之外，图8中，该第二待处理会话为Job4，主机将job4的历史KV cache（KV4）从主机的内存中写入磁盘中。由于磁盘的可用存储空间不足，主机将第三待处理会话的历史KV cache从磁盘中删除，该第三待处理会话为任务队列中在该淘汰豁免窗口之外的队尾第二数量的会话，图8中，该第二数量取值为1，则该第三待处理会话为Job9，主机将Job9的历史KV cache（KV9）从磁盘中删除。需要说明的是，图8仅是对主机的内存和外部存储设备之间的数据迁移过程的举例，不对本申请做出限定。

上述方法中，外部存储设备存储了已完成的会话的历史KV cache，由于外部存储设备的容量远大于加速器内的HBM的容量，因此，外部存储设备能够保存大量的历史KV cache，进而提高历史KV cache的命中率，从而避免进行KV cache重计算，能够提高多轮会话推理的效率，节约计算资源；同时，主机能够在加速器处理会话的同时，将任务队列中待处理会话的历史KV cache从外部存储设备中预加载至主机的内存中，由于计算过程与数据加载过程同步进行，因此计算过程无需等待数据加载完成，从而能够隐藏加速器访问外部存储设备的时间开销，提高多轮会话推理的效率。

需要说明的是，上面为了便于介绍本申请实施例所提供的多轮会话推理方法，采用了两个实施例来分别介绍加速器和主机的内存之间的数据迁移过程以及主机的内存和外部存储设备之间的数据迁移过程，但是并不表示上述两个实施例是相互独立和分别发生的，也即是，本申请实施例所提供的多轮会话推理方法中，在加速器和主机的内存之间进行数据迁移的同时，主机的内存和外部存储设备之间也能够发生数据迁移。

图9是本申请实施例提供的一种多轮会话推理装置的结构示意图，应用于计算系统中的主机的加速器，该计算系统还包括主机以及外部存储设备，该外部存储设备用于存储会话的历史键值缓存KV cache，该主机用于从该外部存储设备中加载任务队列中待处理会话的历史KV cache至该主机的内存，该装置包括处理模块901和加载模块902。

该处理模块901，用于基于该任务队列，通过生成式大模型对该任务队列中的第一会话进行推理，该第一会话为多轮会话中的一轮，该生成式大模型包括N层；

该加载模块902，用于在对该第一会话进行第i层计算时，从该内存中加载该第一会话的输入在第i+1层的历史KV cache，i为大于或等于1且小于N的整数。

在一些实施例中，该第一会话的输入在第i+1层的历史KV cache与该第一会话的输入的位置编码无关。

在一些实施例中，该第一会话的输入包括的词元数量大于或等于该生成式大模型的上下文窗口所能够容纳的词元数量；

该加载模块902用于：

在对该第一会话进行第i层计算时，从该内存中加载第一子输入在第i+1层的历史KV cache，该第一子输入为该第一会话的输入的子输入，该第一子输入包括的词元数量小于该上下文窗口所能够容纳的词元数量。

在一些实施例中，该处理模块901，包括：

嵌入单元，用于将该第一子输入的位置编码嵌入该第一子输入在第i+1层的历史KV cache；

计算单元，用于基于该第一会话的输入和嵌入位置编码后的该第一子输入在第i+1层的历史KV cache，对该第一会话进行第i+1层计算。

在一些实施例中，该加速器包括第一缓冲区和第二缓冲区，该第一缓冲区用于存储该第一会话的历史KV cache，该第二缓冲区用于存储该任务队列中的第二会话的输入在第1层的历史KV cache，该第二会话为该任务队列中该第一会话之后的第一个待处理会话；

该加载模块902用于：

在对该第一会话进行第i层计算时，从该内存中加载该第一会话的输入在第i+1层的历史KV cache至该第一缓冲区。

在一些实施例中，该加载模块902还用于：

在对该第一会话进行第N层计算时，从该内存中加载该第二会话的输入在第1层的历史KV cache至该第二缓冲区。

在一些实施例中，该装置还包括：

写入模块，用于在对该第一会话进行第i+1层计算时，将对该第一会话进行第i层计算所产生的KV cache写入该内存中。

在一些实施例中，该加速器包括第三缓冲区，该第三缓冲区用于存储该生成式大模型处理该第一会话所产生的KV cache；

该写入模块用于：

若完成对该第一会话的第N层计算时，该生成式大模型处理该第一会话所生成的KV cache中存在尚未写入该内存的KV cache，则将该尚未写入该内存的KV cache写入该第三缓冲区；

在通过该生成式大模型处理该任务队列中的第二会话时，将该第三缓冲区中的KV cache写入该内存。

图10是本申请实施例提供的一种多轮会话推理装置的结构示意图，应用于计算系统中的主机，该计算系统还包括主机的加速器以及外部存储设备，该外部存储设备用于存储会话的历史键值缓存KV cache，该加速器用于基于该任务队列，通过生成式大模型对该任务队列中的第一会话进行推理，该第一会话为多轮会话中的一轮，该生成式大模型包括N层，其中，在对该第一会话进行第i层计算时，从该内存中加载该第一会话的输入在第i+1层的历史KV cache，i为大于或等于1且小于N的整数；该装置包括加载模块1001和存储模块1002。

该加载模块1001，用于从该外部存储设备中加载任务队列中待处理会话的历史KV cache至该主机的内存；

该存储模块1002，用于存储该待处理会话的历史KV cache。

在一些实施例中，该加载模块1001包括：

加载单元，用于若该内存中不存在该任务队列中第一待处理会话的历史KV cache，则从该外部存储设备中加载该第一待处理会话的历史KV cache至该内存，该第一待处理会话为该任务队列中队头第一数量的待处理会话。

在一些实施例中，该第一待处理会话包括的会话轮数基于该内存的容量确定。

在一些实施例中，该加载单元用于：

若该内存的可用存储空间足够，则从该外部存储设备中加载该第一待处理会话的历史KV cache至该内存；

若该内存的可用存储空间不足，则将该任务队列中第二待处理会话的历史KV cache从该内存写入该外部存储设备中，从该外部存储设备中加载该第一待处理会话的历史KV cache至该内存，该第二待处理会话为该任务队列中除该第一待处理会话之外的会话。

在一些实施例中，该装置还包括删除模块，用于：

若该外部存储设备的可用存储空间不足，则将该任务队列中第三待处理会话的历史KV cache从该外部存储设备中删除，该第三待处理会话为该任务队列中队尾第二数量的待处理会话。

其中，处理模块901、加载模块902、加载模块1001和存储模块1002均可以通过软件实现，或者可以通过硬件实现。示例性的，接下来以处理模块901为例，介绍处理模块901的实现方式。类似的，加载模块902、加载模块1001和存储模块1002的实现方式可以参考处理模块901的实现方式。

模块作为软件功能单元的一种举例，处理模块901可以包括运行在计算实例上的代码。其中，计算实例可以包括物理主机（计算设备）、虚拟机、容器中的至少一种。进一步地，上述计算实例可以是一台或者多台。例如，处理模块901可以包括运行在多个主机/虚拟机/容器上的代码。需要说明的是，用于运行该代码的多个主机/虚拟机/容器可以分布在相同的区域（region）中，也可以分布在不同的region中。进一步地，用于运行该代码的多个主机/虚拟机/容器可以分布在相同的可用区（availability zone，AZ）中，也可以分布在

不同的AZ中，每个AZ包括一个数据中心或多个地理位置相近的数据中心。其中，通常一个region可以包括多个AZ。

同样，用于运行该代码的多个主机/虚拟机/容器可以分布在同一个虚拟私有云（virtual private cloud，VPC）中，也可以分布在多个VPC中。其中，通常一个VPC设置在一个region内，同一region内两个VPC之间，以及不同region的VPC之间跨区通信需在每个VPC内设置通信网关，经通信网关实现VPC之间的互连。

模块作为硬件功能单元的一种举例，处理模块901可以包括至少一个计算设备，如服务器等。或者，处理模块901也可以是利用专用集成电路（application-specific integrated circuit，ASIC）实现、或可编程逻辑器件（programmable logic device，PLD）实现的设备等。其中，上述PLD可以是复杂程序逻辑器件（complex programmable logical device，CPLD）、现场可编程门阵列（field-programmable gate array，FPGA）、通用阵列逻辑（generic array logic，GAL）或其任意组合实现。

处理模块901包括的多个计算设备可以分布在相同的region中，也可以分布在不同的region中。处理模块901包括的多个计算设备可以分布在相同的AZ中，也可以分布在不同的AZ中。同样，处理模块901包括的多个计算设备可以分布在同一个VPC中，也可以分布在多个VPC中。其中，所述多个计算设备可以是服务器、ASIC、PLD、CPLD、FPGA和GAL等计算设备的任意组合。

需要说明的是，在其他实施例中，处理模块901可以用于执行多轮会话推理方法中的任意步骤，加载模块902可以用于执行多轮会话推理方法中的任意步骤，加载模块1001可以用于执行多轮会话推理方法中的任意步骤，存储模块1002可以用于执行多轮会话推理方法中的任意步骤，处理模块901、加载模块902、加载模块1001和存储模块1002负责实现的步骤可根据需要指定，通过处理模块901和加载模块902分别实现多轮会话推理方法中由加速器执行的不同的步骤来实现如图9所示的多轮会话推理装置的全部功能；或者，通过加载模块1001和存储模块1002分别实现多轮会话推理方法中由主机执行的不同的步骤来实现如图10所示的多轮会话推理装置的全部功能。

本申请还提供一种计算设备1100。图11是本申请实施例提供的一种计算设备的结构示意图，如图11所示，计算设备1100包括：总线1101、处理器1102、存储器1103和通信接口1104。处理器1102、存储器1103和通信接口1104之间通过总线1101通信。计算设备1100可以是计算设备或终端设备。应理解，本申请不限定计算设备1100中的处理器、存储器的个数。

总线1101可以是外设部件互连标准（peripheral component interconnect，PCI）总线或扩展工业标准结构（extended industry standard architecture，EISA）总线等。总线可以分为地址总线、数据总线、控制总线等。为便于表示，图11中仅用一条线表示，但并不表示仅有一根总线或一种类型的总线。总线1101可包括在计算设备1100各个部件（例如，存储器1103、处理器1102、通信接口1104）之间传送信息的通路。

处理器1102可以包括中央处理器(central processing unit，CPU)、图形处理器（graphics processing unit，GPU)、微处理器（micro processor，MP）或者数字信号处理器（digital signal processor，DSP）等处理器中的任意一种或多种。

存储器1103可以包括易失性存储器(volatile memory)，例如随机存取存储器(random access memory，RAM)。存储器1103还可以包括非易失性存储器(non-volatile memory)，例如只读存储器(read-only memory，ROM)，快闪存储器，机械硬盘（hard disk drive，HDD）或固态硬盘（solid state drive，SSD)。

存储器1103中存储有可执行的程序代码，处理器1102执行该可执行的程序代码以分别实现前述处理模块901和加载模块902的功能，从而实现多轮会话推理方法中由加速器执行的步骤，或者，实现前述加载模块1001和存储模块1002的功能，从而实现多轮会话推理方法中由主机执行的步骤。也即，存储器1103上存有用于执行多轮会话推理方法的指令。图11仅示例性地示出存储器1103存储实现前述处理模块901和加载模块902的功能的程序代码。

通信接口1104使用例如但不限于网络接口卡、收发器一类的收发模块，来实现计算设备1100与其他设备或通信网络之间的通信。

本申请实施例还提供了一种计算设备集群。图12是本申请实施例提供的一种计算设备集群的示意图，如图12所示，该计算设备集群包括至少一个计算设备1100。计算设备集群中的一个或多个计算设备1100中的存储器1103中可以存有相同的用于执行多轮会话推理方法的指令。

在一些可能的实现方式中，该计算设备集群中的一个或多个计算设备1100的存储器1103中也可以分别存有用于执行多轮会话推理方法的部分指令。换言之，一个或多个计算设备1100的组合可以共同执行用于执行多轮会话推理方法的指令。

需要说明的是，计算设备集群中的不同的计算设备1100中的存储器1103可以存储不同的指令，分别用于执行多轮会话推理装置的部分功能。也即，不同的计算设备1100中的存储器1103存储的指令可以实现前述处理模块901、加载模块902、加载模块1001和存储模块1002中的一个或多个模块的功能。

应理解，图12中示出的计算设备1100的功能也可以由多个计算设备1100完成。

在一些可能的实现方式中，计算设备集群中的一个或多个计算设备可以通过网络连接。其中，该网络可以是广域网或局域网等等。图13示出了一种可能的实现方式。图13是本申请实施例提供的一种计算设备集群的可能实现方式的示意图，如图13所示，计算设备1100A和计算设备1100B之间通过网络进行连接。具体地，通过各个计算设备中的通信接口与该网络进行连接。在这一类可能的实现方式中，计算设备1100A中的存储器1103中存有执行处理模块901的功能的指令。图13中以计算设备1100A中的存储器1103中存有执行处理模块901的功能的指令为例。同时，计算设备1100B中的存储器1103中存有执行加载模块902的功能的指令。图13中以计算设备1100B中的存储器1103中存有执行加载模块902的功能的指令为例。

本申请实施例还提供了另一种计算设备集群。该计算设备集群中各计算设备之间的连接关系可以类似的参考图12或图13的连接方式。不同的是，该计算设备集群中的一个或多个计算设备1100中的存储器1103中可以存有相同的用于执行多轮会话推理方法的指令。

在一些可能的实现方式中，该计算设备集群中的一个或多个计算设备1100的存储器1103中也可以分别存有用于执行多轮会话推理方法的部分指令。换言之，一个或多个计算设备1100的组合可以共同执行用于执行多轮会话推理方法的指令。

需要说明的是，计算设备集群中的不同的计算设备1100中的存储器1103可以存储不同的指令，分别用于执行多轮会话推理装置的部分功能。也即，不同的计算设备1100中的存储器1103存储的指令可以实现前述处理模块901、加载模块902、加载模块1001和存储模块1002中的一个或多个模块的功能。

本申请实施例提供了一种加速器，该加速器包括计算核心和存储器，该加速器用于执行如前述方法实施例中任一种可能的实现方式所提供的多轮会话推理方法中由加速器执行的步骤，该存储器用于存储计算数据，该计算核心用于对存储器中存储的计算数据执行计算操作。

本申请实施例提供了一种主机，该主机包括处理器和存储器，该主机用于执行如前述方法实施例中中任一种可能的实现方式所提供的多轮会话推理方法中由主机执行的步骤。

本申请实施例还提供了一种包含指令的计算机程序产品。所述计算机程序产品可以是包含指令的，能够运行在计算设备上或被储存在任何可用介质中的软件或程序产品。当所述计算机程序产品在至少一个计算设备上运行时，使得至少一个计算设备执行多轮会话推理方法。

本申请实施例还提供了一种计算机可读存储介质。所述计算机可读存储介质可以是计算设备能够存储的任何可用介质或者是包含一个或多个可用介质的数据中心等数据存储设备。所述可用介质可以是磁性介质，（例如，软盘、硬盘、磁带）、光介质（例如，DVD）、或者半导体介质（例如固态硬盘）等。该计算机可读存储介质包括指令，所述指令指示计算设备执行多轮会话推理方法。

需要说明的是，本申请所涉及的信息（包括但不限于用户设备信息、用户个人信息等）、数据（包括但不限于用于分析的数据、存储的数据、展示的数据等）以及信号，均为经用户授权或者经过各方充分授权的，且相关数据的收集、使用和处理需要遵守相关国家和地区的相关法律法规和标准。例如，本申请中涉及到的会话和存储空间等都是在充分授权的情况下获取的。

本领域普通技术人员可以意识到，结合本文中所公开的实施例中描述的各方法步骤和单元，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实施例的步骤及组成。这些功能究竟以硬件还是软件方式来执行，取决于技术方案的特定应用和设计约束条件。本领域普通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，但是这种实现不应认为超出本申请的范围。

所属领域的技术人员可以清楚地了解到，为了描述的方便和简洁，上述描述的系统、装置和单元的具体工作过程，可以参见前述方法实施例中的对应过程，在此不再赘述。

在本申请所提供的几个实施例中，应该理解到，所揭露的系统、装置和方法，可以通过其它的方式实现。例如，以上所描述的装置实施例仅是示意性的，例如，该单元的划分，仅仅为一种逻辑功能划分，实际实现时可以有另外的划分方式，例如多个单元或组件可以结合或者可以集成到另一个系统，或一些特征可以忽略，或不执行。另外，所显示或讨论的相互之间的耦合或直接耦合或通信连接可以是通过一些接口、装置或单元的间接耦合或通信连接，也可以是电的，机械的或其它的形式连接。

该作为分离部件说明的单元可以是或者也可以不是物理上分开的，作为单元显示的部件可以是或者也可以不是物理单元，即可以位于一个地方，或者也可以分布到多个网络单元上。可以根据实际的需要选择其中的部分或者全部单元来实现本申请实施例方案的目的。

另外，在本申请各个实施例中的各单元可以集成在一个处理单元中，也可以是各个单元单独物理存在，也可以是两个或两个以上单元集成在一个单元中。上述集成的单元既可以采用硬件的形式实现，也可以采用软件单元的形式实现。

该集成的单元如果以软件功能单元的形式实现并作为独立的产品销售或使用时，可以存储在一个计算机可读存储介质中。基于这样的理解，本申请的技术方案本质上或者说对现有技术做出贡献的部分，或者该技术方案的全部或部分可以以软件产品的形式体现出来，该计算机软件产品存储在一个存储介质中，包括若干指令用以使得一台计算设备（可以是个人计算机，服务器，或者计算设备等）执行本申请各个实施例中方法的全部或部分步骤。而前述的存储介质包括：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以存储程序代码的介质。

本申请中术语“第一”“第二”等字样用于对作用和功能基本相同的相同项或相似项进行区分，应理解，“第一”、“第二”、“第n”之间不具有逻辑或时序上的依赖关系，也不对数量和执行顺序进行限定。还应理解，尽管以下描述使用术语第一、第二等来描述各种元素，但这些元素不应受术语的限制。这些术语只是用于将一元素与另一元素区别分开。例如，在不脱离各种示例的范围的情况下，第一会话可以被称为第二会话，并且类似地，第二会话可以被称为第一会话。第一会话和第二会话都可以是节点会话，并且在某些情况下，可以是单独且不同的会话。

本申请中术语“至少一个”的含义是指一个或多个，本申请中术语“多个”的含义是指两个或两个以上，本文中术语“系统”和“网络”经常可互换使用。

还应理解，术语“如果”可被解释为意指“当……时”（“when”或“upon”）或“响应于确定”或“响应于检测到”。类似地，根据上下文，短语“如果确定……”或“如果检测到［所陈述的条件或事件］”可被解释为意指“在确定……时”或“响应于确定……”或“在检测到［所陈述的条件或事件］时”或“响应于检测到［所陈述的条件或事件］”。

以上描述，仅为本申请的具体实施方式，但本申请的保护范围并不局限于此，任何熟悉本技术领域的技术人员在本申请揭露的技术范围内，可轻易想到各种等效的修改或替换，这些修改或替换都应涵盖在本申请的保护范围之内。因此，本申请的保护范围应以权利要求的保护范围为准。

在上述实施例中，可以全部或部分地通过软件、硬件、固件或者其任意组合来实现。当使用软件实现时，可以全部或部分地以计算机程序产品的形式实现。该计算机程序产品包括一个或多个计算机程序指令。在计算机上加载和执行该计算机程序指令时，全部或部分地产生按照本申请实施例中的流程或功能。该计算机可以是通用计算机、专用计算机、计算机网络、或者其他可编程装置。

该计算机指令可以存储在计算机可读存储介质中，或者从一个计算机可读存储介质向另一个计算机可读存储介质传输，例如，该计算机程序指令可以从一个网站站点、计算机、服务器或数据中心通过有线或无线方式向另一个网站站点、计算机、服务器或数据中心进行传输。该计算机可读存储介质可以是计算机能够存取的任何可用介质或者是包含一个或多个可用介质集成的服务器、数据中心等数据存储设备。该可用介质可以是磁性介质（例如软盘、硬盘、磁带）、光介质（例如，数字视频光盘（digital video disc，DVD）、

或者半导体介质（例如固态硬盘））等。

本领域普通技术人员可以理解实现上述实施例的全部或部分步骤可以通过硬件来完成，也可以通过程序来指令相关的硬件完成，该程序可以存储于一种计算机可读存储介质中，上述提到的存储介质可以是只读存储器，磁盘或光盘等。

以上实施例仅用以说明本申请的技术方案，而非对其限制；尽管参照前述实施例对本申请进行了详细的说明，本领域的普通技术人员应当理解：其依然可以对前述各实施例所记载的技术方案进行修改，或者对其中部分技术特征进行等同替换；而这些修改或者替换，并不使相应技术方案的本质脱离本申请各实施例技术方案的范围。

权利要求书

1.一种计算系统，其特征在于，所述计算系统包括主机、所述主机的加速器以及外部存储设备；其中，

所述外部存储设备用于存储会话的历史键值缓存KV cache；

所述主机用于从所述外部存储设备中加载任务队列中待处理会话的历史KV cache至所述主机的内存；

所述加速器用于基于所述任务队列，通过生成式大模型对所述任务队列中的第一会话进行推理，所述第一会话为多轮会话中的一轮，所述生成式大模型包括N层，其中，

在对所述第一会话进行第i层计算时，从所述内存中加载所述第一会话的输入在第i+1层的历史KV cache，i为大于或等于1且小于N的整数。

2.根据权利要求1所述的计算系统，其特征在于，所述第一会话的输入在第i+1层的历史KV cache与所述第一会话的输入的位置编码无关。

3.根据权利要求2所述的计算系统，其特征在于，所述第一会话的输入包括的词元数量大于或等于所述生成式大模型的上下文窗口所能够容纳的词元数量；

所述加速器用于：

在对所述第一会话进行第i层计算时，从所述内存中加载第一子输入在第i+1层的历史KV cache，所述第一子输入为所述第一会话的输入的子输入，所述第一子输入包括的词元数量小于所述上下文窗口所能够容纳的词元数量。

4.根据权利要求3所述的计算系统，其特征在于，所述加速器还用于：

将所述第一子输入的位置编码嵌入所述第一子输入在第i+1层的历史KV cache；

基于所述第一会话的输入和嵌入位置编码后的所述第一子输入在第i+1层的历史KV cache，对所述第一会话进行第i+1层计算。

5.根据权利要求1所述的计算系统，其特征在于，所述加速器包括第一缓冲区和第二缓冲区，所述第一缓冲区用于存储所述第一会话的历史KV cache，所述第二缓冲区用于存储所述任务队列中的第二会话的输入在第1层的历史KV cache，所述第二会话为所述任务队列中所述第一会话之后的第一个待处理会话；

所述加速器用于：

在对所述第一会话进行第i层计算时，从所述内存中加载所述第一会话的输入在第i+1层的历史KV cache至所述第一缓冲区；

在对所述第一会话进行第N层计算时，从所述内存中加载所述第二会话的输入在第1层的历史KV cache至所述第二缓冲区。

6.根据权利要求1所述的计算系统，其特征在于，所述加速器还用于：

在对所述第一会话进行第i+1层计算时，将对所述第一会话进行第i层计算所产生的KV cache写入所述内存中。

7.根据权利要求6所述的计算系统，其特征在于，所述加速器包括第三缓冲区，所述第三缓冲区用于存储所述生成式大模型处理所述第一会话所产生的KV cache；

所述主机的加速器用于：

若完成对所述第一会话的第N层计算时，所述生成式大模型处理所述第一会话所生成的KV cache中存在尚未写入所述内存的KV cache，则将所述尚未写入所述内存的KV cache写入所述第三缓冲区；

在通过所述生成式大模型处理所述任务队列中的第二会话时，将所述第三缓冲区中的KV cache写入所述内存。

8.根据权利要求1所述的计算系统，其特征在于，所述主机用于：

若所述内存中不存在所述任务队列中第一待处理会话的历史KV cache，则从所述外部存储设备中加载所述第一待处理会话的历史KV cache至所述内存，所述第一待处理会话为所述任务队列中队头第一数量的待处理会话。

9.根据权利要求8所述的计算系统，其特征在于，所述第一待处理会话包括的会话轮数基于所述内存的容量确定。

10.根据权利要求8或9所述的计算系统，其特征在于，所述主机用于：

若所述内存的可用存储空间足够，则从所述外部存储设备中加载所述第一待处理会话的历史KV cache至所述内存；

若所述内存的可用存储空间不足，则将所述任务队列中第二待处理会话的历史KV cache从所述内存写入所述外部存储设备中，从所述外部存储设备中加载所述第一待处理会话的历史KV cache至所述内存，所述第二待处理会话为所述任务队列中除所述第一待处理会话之外的会话。

11.根据权利要求1所述的计算系统，其特征在于，所述主机还用于：

若所述外部存储设备的可用存储空间不足，则将所述任务队列中第三待处理会话的历史KV cache从所述外部存储设备中删除，所述第三待处理会话为所述任务队列中队尾第二数量的待处理会话。

12.一种多轮会话推理方法，其特征在于，应用于计算系统，所述计算系统包括主机、所述主机的加速器以及外部存储设备，所述外部存储设备用于存储会话的历史键值缓存KV cache，所述主机用于从所述外部存储设备中加载任务队列中待处理会话的历史KV cache至所述主机的内存，所述方法包括：

通过所述加速器，基于所述任务队列，通过生成式大模型对所述任务队列中的第一会话进行推理，所述第一会话为多轮会话中的一轮，所述生成式大模型包括N层，其中，

在通过所述加速器对所述第一会话进行第i层计算时，从所述内存中加载所述第一会话的输入在第i+1层的历史KV cache至所述加速器，i为大于或等于1且小于N的整数。

13.一种多轮会话推理方法，其特征在于，应用于计算系统中的主机的加速器，所述计算系统还包括主机以及外部存储设备，所述外部存储设备用于存储会话的历史键值缓存KV cache，所述主机用于从所述外部存储设备中加载任务队列中待处理会话的历史KV cache至所述主机的内存，所述方法包括：

基于所述任务队列，通过生成式大模型对所述任务队列中的第一会话进行推理，所述第一会话为多轮会话中的一轮，所述生成式大模型包括N层，其中，

在对所述第一会话进行第i层计算时，从所述内存中加载所述第一会话的输入在第i+1层的历史KV cache，i为大于或等于1且小于N的整数。

14.根据权利要求13所述的方法，其特征在于，所述第一会话的输入在第i+1层的历史KV cache与所述第一会话的输入的位置编码无关。

15.根据权利要求14所述的方法，其特征在于，所述第一会话的输入包括的词元数量大于或等于所述生成式大模型的上下文窗口所能够容纳的词元数量；

所述在对所述第一会话进行第i层计算时，从所述内存中加载所述第一会话的输入在第i+1层的历史KV cache，包括：

在对所述第一会话进行第i层计算时，从所述内存中加载第一子输入在第i+1层的历史KV cache，所述第一子输入为所述第一会话的输入的子输入，所述第一子输入包括的词元数量小于所述上下文窗口所能够容纳的词元数量。

16.根据权利要求15所述的方法，其特征在于，所述通过生成式大模型处理所述任务队列中的第一会话，

包括：

将所述第一子输入的位置编码嵌入所述第一子输入在第i+1层的历史KV cache；

基于所述第一会话的输入和嵌入位置编码后的所述第一子输入在第i+1层的历史KV cache，对所述第一会话进行第i+1层计算。

17.根据权利要求13所述的方法，其特征在于，所述加速器包括第一缓冲区和第二缓冲区，所述第一缓冲区用于存储所述第一会话的历史KV cache，所述第二缓冲区用于存储所述任务队列中的第二会话的输入在第1层的历史KV cache，所述第二会话为所述任务队列中所述第一会话之后的第一个待处理会话；

所述在对所述第一会话进行第i层计算时，从所述内存中加载所述第一会话的输入在第i+1层的历史KV cache，包括：

在对所述第一会话进行第i层计算时，从所述内存中加载所述第一会话的输入在第i+1层的历史KV cache至所述第一缓冲区。

18.根据权利要求17所述的方法，其特征在于，所述方法还包括：

在对所述第一会话进行第N层计算时，从所述内存中加载所述第二会话的输入在第1层的历史KV cache至所述第二缓冲区。

19.根据权利要求13所述的方法，其特征在于，所述方法还包括：

在对所述第一会话进行第i+1层计算时，将对所述第一会话进行第i层计算所产生的KV cache写入所述内存中。

20.根据权利要求19所述的方法，其特征在于，所述加速器包括第三缓冲区，所述第三缓冲区用于存储所述生成式大模型处理所述第一会话所产生的KV cache；

所述在对所述第一会话进行第i+1层计算时，将对所述第一会话进行第i层计算所产生的KV cache写入所述内存中，包括：

若完成对所述第一会话的第N层计算时，所述生成式大模型处理所述第一会话所生成的KV cache中存在尚未写入所述内存的KV cache，则将所述尚未写入所述内存的KV cache写入所述第三缓冲区；

在通过所述生成式大模型处理所述任务队列中的第二会话时，将所述第三缓冲区中的KV cache写入所述内存。

21.一种多轮会话推理方法，其特征在于，应用于计算系统中的主机，所述计算系统还包括主机的加速器以及外部存储设备，所述外部存储设备用于存储会话的历史键值缓存KV cache，所述加速器用于基于任务队列，通过生成式大模型对所述任务队列中的第一会话进行推理，所述第一会话为多轮会话中的一轮，所述生成式大模型包括N层，其中，在对所述第一会话进行第i层计算时，从所述主机的内存中加载所述第一会话的输入在第i+1层的历史KV cache，i为大于或等于1且小于N的整数；

所述方法包括：

从所述外部存储设备中加载任务队列中待处理会话的历史KV cache至所述主机的内存；

存储所述待处理会话的历史KV cache。

22.根据权利要求21所述的方法，其特征在于，所述从所述外部存储设备中加载任务队列中待处理会话的历史KV cache至所述主机的内存，包括：

若所述内存中不存在所述任务队列中第一待处理会话的历史KV cache，则从所述外部存储设备中加载所述第一待处理会话的历史KV cache至所述内存，所述第一待处理会话为所述任务队列中队头第一数量的待处理会话。

23.根据权利要求22所述的方法，其特征在于，所述第一待处理会话包括的会话轮数基于所述内存的容量确定。

24.根据权利要求22或23所述的方法，其特征在于，所述若所述内存中不存在所述任务队列中第一待处理会话的历史KV cache，则所述主机从所述外部存储设备中加载所述第一待处理会话的历史KV cache至所述内存，包括：

若所述内存的可用存储空间足够，则从所述外部存储设备中加载所述第一待处理会话的历史KV cache至所述内存；

若所述内存的可用存储空间不足，则将所述任务队列中第二待处理会话的历史KV cache从所述内存写入所述外部存储设备中，从所述外部存储设备中加载所述第一待处理会话的历史KV cache至所述内存，所述第二待处理会话为所述任务队列中除所述第一待处理会话之外的会话。

25.根据权利要求21至24中任一项所述的方法，其特征在于，所述方法还包括：

若所述外部存储设备的可用存储空间不足，则将所述任务队列中第三待处理会话的历史KV cache从所述外部存储设备中删除，所述第三待处理会话为所述任务队列中队尾第二数量的待处理会话。

26.一种多轮会话推理装置，其特征在于，应用于计算系统中的主机的加速器，所述计算系统还包括主机以及外部存储设备，所述外部存储设备用于存储会话的历史键值缓存KV cache，所述主机用于从所述外部存储设备中加载任务队列中待处理会话的历史KV cache至所述主机的内存，所述装置包括：

处理模块，用于基于所述任务队列，通过生成式大模型对所述任务队列中的第一会话进行推理，所述第一会话为多轮会话中的一轮，所述生成式大模型包括N层；

加载模块，用于在对所述第一会话进行第i层计算时，从所述内存中加载所述第一会话的输入在第i+1层的历史KV cache，i为大于或等于1且小于N的整数。

27.一种多轮会话推理装置，其特征在于，应用于计算系统中的主机，所述计算系统还包括主机的加速器以及外部存储设备，所述外部存储设备用于存储会话的历史键值缓存KV cache，所述加速器用于基于任务队列，通过生成式大模型对所述任务队列中的第一会话进行推理，所述第一会话为多轮会话中的一轮，所述生成式大模型包括N层，其中，在对所述第一会话进行第i层计算时，从所述主机的内存中加载所述第一会话的输入在第i+1层的历史KV cache，i为大于或等于1且小于N的整数；

所述装置包括：

加载模块，从所述外部存储设备中加载任务队列中待处理会话的历史KV cache至所述主机的内存。

28.一种加速器，其特征在于，所述加速器包括计算核心和存储器，所述加速器用于执行权利要求13至权利要求20中任一项多轮会话推理方法，所述存储器用于存储计算数据，所述计算核心用于对存储器中存储的计算数据执行计算操作。

29.一种主机，其特征在于，所述主机包括处理器和存储器，所述主机用于执行权利要求21至权利要求25中任一项多轮会话推理方法。

30.一种计算设备集群，其特征在于，所述计算设备集群包括至少一个计算设备，每个所述计算设备包括主机、主机的加速器，外部存储设备位于至少一个计算设备，所述外部存储设备用于存储会话的历史KV cache，所述主机用于从所述外部存储设备中加载任务队列中待处理会话的历史KV cache至所述主机的内存；

所述计算设备集群中的所述主机用于执行程序代码，以实现如上述权利要求13至20中任一项所述的多轮会话推理方法中由主机执行的步骤；

所述计算设备集群中的所述加速器用于执行程序代码，以实现如上述权利要求21至25中任一项所述的多轮会话推理方法中由加速器执行的步骤。

31.一种包含指令的计算机程序产品，其特征在于，当所述指令被计算设备集群运行时，使得所述计算设备集群执行如权利要求13至20、或权利要求21至25任一所述的多轮会话推理方法。

32.一种计算机可读存储介质，其特征在于，包括计算机程序指令，当所述计算机程序指令由计算设备集群执行时，所述计算设备集群执行如权利要求13至20、或权利要求21至25任一所述的多轮会话推理方法。