数据处理方法、文本处理方法、图像处理方法、计算设备、计算机存储介质以及计算机程序产品

本公开要求申请号为202410217614.9的中国专利申请的优先权，该中国专利申请于2024年02月27日提交中国专利局，申请名称为“数据处理方法、文本处理方法、图像处理方法、计算设备、计算机存储介质以及计算机程序产品”，其全部内容通过引用结合在本公开中。

技术领域

本说明书实施例涉及计算机技术领域，特别涉及一种数据处理方法、文本处理方法、图像处理方法、计算设备、计算机存储介质以及计算机程序产品。

背景技术

对于传统的深度学习推理加速方案，可利用静态图优化技术大幅提高模型推理效率，即通过对计算图中的节点进行融合，并选择高度优化的内核函数执行运算，从而减少计算机硬件资源的计算开销和提升模型推理过程的执行速度。

然而，面对Stable-Diffusion（图像生成模型，其工作原理涉及到多次迭代过程以逐渐从噪声向清晰图像转化，每次迭代都需要根据前一次迭代的结果动态调整参数或计算流程，这种动态和递归性质使得该模型难以直接用静态图表示和优化）这类具有迭代扩散过程和动态行为的模型时，这种静态图解决方案由于无法灵活应对模型在推理过程中根据前一步骤动态决定后续步骤这一特性，因此难以直接应用并发挥同样的加速效果；这意味着对于此类模型，需要发展新的适应其内在动态属性的优化技术。

发明内容

有鉴于此，本说明书实施例提供了一种数据处理方法、文本处理方法、图像处理方法、计算设备、计算机存储介质以及计算机程序产品，以解决现有技术中存在的技术缺陷。

根据本说明书实施例的第一方面，提供了一种数据处理方法，包括：

确定目标模型的目标计算图以及输入所述目标模型的目标数据，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计算图；

从所述目标计算图中确定处理所述目标数据的当前节点，其中，所述当前节点为所述多个节点中的任意一个节点；

确定所述当前节点对应的数据处理计算图，根据所述当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果。

根据本说明书实施例的第二方面，提供了一种数据处理装置，包括：

数据确定模块，被配置为确定目标模型的目标计算图以及输入所述目标模型的目标数据，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计

算图；

节点确定模块，被配置为从所述目标计算图中确定处理所述目标数据的当前节点，其中，所述当前节点为所述多个节点中的任意一个节点；

结果获得模块，被配置为确定所述当前节点对应的数据处理计算图，根据所述当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果。

根据本说明书实施例的第三方面，提供了一种文本处理方法，包括：

确定目标模型的目标计算图以及输入所述目标模型的目标文本，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计算图；

从所述目标计算图中确定处理所述目标文本的当前节点，其中，所述当前节点为所述多个节点中的任意一个节点；

确定所述当前节点对应的文本处理计算图，根据所述当前节点和/或所述文本处理计算图，处理所述目标文本，获得文本处理结果。

根据本说明书实施例的第四方面，提供了一种文本处理装置，包括：

文本确定模块，被配置为确定目标模型的目标计算图以及输入所述目标模型的目标文本，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计算图；

节点确定模块，被配置为从所述目标计算图中确定处理所述目标文本的当前节点，其中，所述当前节点为所述多个节点中的任意一个节点；

结果获得模块，被配置为确定所述当前节点对应的文本处理计算图，根据所述当前节点和/或所述文本处理计算图，处理所述目标文本，获得文本处理结果。

根据本说明书实施例的第五方面，提供了一种图像处理方法，包括：

确定目标模型的目标计算图以及输入所述目标模型的目标图像，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计算图；

从所述目标计算图中确定处理所述目标图像的当前节点，其中，所述当前节点为所述多个节点中的任意一个节点；

确定所述当前节点对应的图像处理计算图，根据所述当前节点和/或所述图像处理计算图，处理所述目标图像，获得图像处理结果。

根据本说明书实施例的第六方面，提供了一种图像处理装置，包括：

图像确定模块，被配置为确定目标模型的目标计算图以及输入所述目标模型的目标图像，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计算图；

节点确定模块，被配置为从所述目标计算图中确定处理所述目标图像的当前节点，其中，所述当前节点为所述多个节点中的任意一个节点；

结果获得模块，被配置为确定所述当前节点对应的图像处理计算图，根据所述当前节点和/或所述图像处理计算图，处理所述目标图像，获得图像处理结果。

根据本说明书实施例的第七方面，提供了一种计算设备，包括：

存储器和处理器；

所述存储器用于存储计算机程序/指令，所述处理器用于执行所述计算机程序/指令，该计算机程序/指令被处理器执行时实现上述数据处理方法、或文本处理方法、或图像处理方法的步骤。

根据本说明书实施例的第八方面，提供了一种计算机可读存储介质，其存储有计算机程序/指令，该计算机程序/指令被处理器执行时实现上述数据处理方法、或文本处理方法、或图像处理方法的步骤。

根据本说明书实施例的第九方面，提供了一种计算机程序产品，包括计算机程序/指令，该计算机程序/指令被处理器执行时实现上述数据处理方法、或文本处理方法、或图像处理方法的步骤。

本说明书一个实施例提供一种数据处理方法，包括：确定目标模型的目标计算图以及输入所述目标模型的目标数据，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计算图；从所述目标计算图中确定处理所述目标数据的当前节点，其中，所述当前节点为所述多个节点中的任意一个节点；确定所述当前节点对应的数据处理计算图，根据所述当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果。

本说明书实施例提供的数据处理方法，通过确定目标模型的目标计算图，在给目标模型输入目标数据的情况下，由于目标计算图为在目标模型运行时不能进行更新的计算图，即目标计算图在目标模型运行前就被完全定义并编译，这有利于在进行模型推理前进行对目标计算图的编译进行优化，降低处理器的运算负担，提高处理器的执行效率；同时，通过灵活地调用数据处理计算图，能确保目标模型对目标数据进行高效、准确的前置处理或后置处理操作，实现在目标计算图中动态插入计算流程，且无需二次编译，节省计算图编译的开销，减少对计算机硬件资源的需求和消耗，提高模型推理效率。

附图说明

图 1 是本说明书一个实施例提供的一种数据处理方法的场景示意图；
图 2 是本说明书一个实施例提供的一种数据处理方法的流程图；
图 3 是本说明书一个实施例提供的一种前置处理表的示意图；
图 4 是本说明书一个实施例提供的一种 UNET 模型结构示意图；
图 5 是本说明书一个实施例提供的一种目标映射表的示意图；
图 6 是本说明书一个实施例提供的一种数据处理方法的单节点处理过程流程图；
图 7 是本说明书一个实施例提供的一种文本处理方法的流程图；
图 8 是本说明书一个实施例提供的一种图像处理方法的流程图；
图 9 是本说明书一个实施例提供的一种数据处理装置的结构示意图；
图 10 是本说明书一个实施例提供的一种文本处理装置的结构示意图；
图 11 是本说明书一个实施例提供的一种图像处理装置的结构示意图；
图 12 是本说明书一个实施例提供的一种计算设备的结构框图。

具体实施方式

在下面的描述中阐述了很多具体细节以便于充分理解本说明书。但是本说明书能够以很多不同于在此描述的其它方式来实施，本领域技术人员可以在不违背本说明书内涵的情况下做类似推广，因此本说明书不受下面公开的具体实施的限制。

在本说明书一个或多个实施例中使用的术语是仅仅出于描述特定实施例的目的，而非旨在限制本说明书一个或多个实施例。在本说明书一个或多个实施例和所附权利

要求书中所使用的单数形式的“一种”、“所述”和“该”也旨在包括多数形式，除非上下文清楚地表示其他含义。还应当理解，本说明书一个或多个实施例中使用的术语“和/或”是指并包含一个或多个相关联的列出项目的任何或所有可能组合。

应当理解，尽管在本说明书一个或多个实施例中可能采用术语第一、第二等来描述各种信息，但这些信息不应限于这些术语。这些术语仅用来将同一类型的信息彼此区分开。例如，在不脱离本说明书一个或多个实施例范围的情况下，第一也可以被称为第二，类似地，第二也可以被称为第一。取决于语境，如在此所使用的词语“如果”可以被解释成为“在……时”或“当……时”或“响应于确定”。

此外，需要说明的是，本说明书一个或多个实施例所涉及的用户信息（包括但不限于用户设备信息、用户个人信息等）和数据（包括但不限于用于分析的数据、存储的数据、展示的数据等），均为经用户授权或者经过各方充分授权的信息和数据，并且相关数据的收集、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，并提供有相应的操作入口，供用户选择授权或者拒绝。

本说明书一个或多个实施例中，大模型是指具有大规模模型参数的深度学习模型，通常包含上亿、上百亿、上千亿、上万亿甚至十万亿以上的模型参数。大模型又可以称为基石模型/基础模型（Foundation Model），通过大规模无标注的语料进行大模型的预训练，产出亿级以上参数的预训练模型，这种模型能适应广泛的下游任务，模型具有较好的泛化能力，例如大规模语言模型（Large Language Model，LLM）、多模态预训练模型（multi-modal pre-training model）等。

大模型在实际应用时，仅需少量样本对预训练模型进行微调即可应用于不同的任务中，大模型可以广泛应用于自然语言处理（Natural Language Processing，简称 NLP）、计算机视觉等领域，具体可以应用于如视觉问答（Visual Question Answering，简称 VQA）、图像描述（Image Caption，简称 IC）、图像生成等计算机视觉领域任务，以及基于文本的情感分类、文本摘要生成、机器翻译等自然语言处理领域任务，大模型主要的应用场景包括数字助理、智能机器人、搜索、在线教育、办公软件、电子商务、智能设计等。

首先，对本说明书一个或多个实施例涉及的名词术语进行解释。

Stable-Diffusion：一种文本到图像的生成模型，它主要用于根据文本的描述生成详细图像，它也可以应用于其他任务，如内补绘制、外补绘制，以及在提示词指导下产生图生图的转变。

Controlnet：Stable Diffusion 模型中的一个重要配套插件，它的作用是通过对特征表示的控制来引导图像的生成过程，以保证生成的图像与输入的文本描述更加一致。

TensorRT：一款高性能深度学习推理 SDK（Software Development Kit，软件开发工具包），包含深度学习推理优化器和运行环境，可为深度学习推理应用提供低延迟和高吞吐量。

对于现代的大模型，如 LLM（Large Language Model，大规模语言模型）和 Stable-Diffusion 视觉生成模型，在推理过程中采用了一系列轻量化控制技术来精确调整输出结果。例如，Controlnet（Stable Diffusion 模型中的一个重要配套插件）设计了一种并行结构，通过一个专门处理参考图像信息的独立网络模块，将其产生的计算结果张量整合到 Stable Diffusion 模型的计算中，以控制、影响 Stable Diffusion 模型生成的图片。另一方法则通过在模型内部插入特定的上采样-下采样模块，实现在不改变整体模型架构的情况下，对生成的图片进行风格迁移和动态样式控制。

现有技术一方法基于动态图可动态调整的特性，通过替换计算图中的特定算子，可以针对不同输入或阶段自适应地优化推理过程，实现模型推理加速，但这种方法限

制了可以优化的方式，不能实现更大规模上的算子融合；现有技术的另一加速方法则完全使用静态图计算，无法动态修改计算图，无法动态支持如 Controlnet 这样的方案。

因此，本说明书实施例提供了一种数据处理方法，以静态图优化为基础，同时保留计算图动态修改，以在节省计算机硬件资源的基础上，实现对计算图的动态修改。

在本说明书中，提供了一种数据处理方法。本说明书一个或者多个实施例同时涉及一种数据处理装置，一种文本处理方法，一种文本处理装置，一种图像处理方法，一种图像处理装置，一种计算设备，一种计算机可读存储介质以及一种计算机程序产品，在下面的实施例中逐一进行详细说明。

参见图 1，图 1 示出了根据本说明书一个实施例提供的一种数据处理方法的场景示意图。

具体的，该数据处理方法应用客户端 102 以及服务端 104 实现，客户端 102 用于向服务端 104 发送目标数据，在服务端 104 中部署有目标模型，将目标模型转换为静态计算图，其中静态计算图中的节点代表数学运算（即目标模型进行数据处理时的处理操作），边则代表数据流（即处理操作之间的关联关系）；并对目标模型的静态计算图可以进行编译优化，比如通过合并重复操作、减少冗余计算来提高处理效率，并能够利用 GPU（Graphics Processing Unit，图形处理器）等硬件加速计算，编译后的静态计算图能被高效地执行。

在将目标数据输入目标模型的情况下，执行目标模型的推理流程，根据静态计算图逐个节点的进行计算，即从目标模型的静态计算图中确定处理目标数据的当前节点，并确定当前节点对应的数据处理计算图，其中，数据处理计算图用于对目标处理数据进行前置和/或后置处理，根据当前节点和/或数据处理计算图，处理目标数据，获得数据处理结果，并将该数据处理结果返回至客户端 102。

具体的，通过判断当前节点是否存在其对应的数据处理计算图，并在存在的情况下调用当前节点对应的数据处理计算图，实际应用中，数据处理计算图包括前置数据处理计算图、后置处理数据计算图，从而实现当前节点前后的数据处理计算图的增加、删除、修改的功能。

客户端 102 可以包括浏览器、APP（Application，应用程序）、或网页应用如 H5（Hyper Text Markup Language5，超文本标记语言第 5 版）应用、或轻应用（也被称为小程序，一种轻量级应用程序）或云应用等，客户端可以基于服务端提供的相应服务的软件开发工具包（SDK，Software Development Kit），如基于实时通信（RTC，Real Time Communication）SDK 开发获得等。客户端可以部署在电子设备中，需要依赖设备运行或者设备中的某些 APP 而运行等。电子设备可以具有显示屏并支持信息浏览等，如可以是个人移动终端如手机、平板电脑、个人计算机等。在电子设备中通常还可以配置各种其它类应用，例如人机对话类应用、模型训练类应用、数据处理类应用、网页浏览器应用、购物类应用、搜索类应用、即时通信工具、邮箱客户端、社交平台软件等。

服务端 104 可以理解为提供各种服务的服务端，包括物理服务端、云服务端，例如为多个客户端提供通信服务的服务端，又如为客户端上使用的模型提供支持的用于后台训练的服务端，又如对客户端发送的数据进行处理的服务端等。需要说明的是，服务端 104 可以实现成多个服务端组成的分布式服务端集群，也可以实现成单个服务端。服务端 104 也可以为分布式系统的服务端，或者是结合了区块链的服务端。服务端 104 也可以是云服务、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、内容分发网络（CDN，Content Delivery Network）、以及大数据和人工智能平台等基础云计算服务的云服务端，或者是带人工智能技术的

智能云计算服务端或智能云主机。

值得说明的是，本说明书实施例中提供的数据处理方法可以由服务端 104 执行，在本说明书的其它实施例中，可以将目标模型部署在客户端 102 中，使得客户端 102 也可以与服务端 104 具有相似的功能，从而执行本说明书实施例所提供的数据处理方法；在其它实施例中，本说明书实施例所提供的数据处理方法还可以是由客户端 102 与服务端 104 共同执行。

本说明书实施例提供的数据处理方法，通过确定目标模型的目标计算图，在给目标模型输入目标数据的情况下，由于目标计算图为在目标模型运行时不能进行更新的计算图，即目标计算图在目标模型运行前就被完全定义并编译，这有利于在进行模型推理前进行对目标计算图的编译进行优化，降低处理器的运算负担，提高处理器的执行效率；同时，通过灵活地调用数据处理计算图，能确保目标模型对目标数据进行高效、准确的前置处理或后置处理操作，实现在目标计算图中动态插入计算流程，且无需二次编译，节省计算图编译的开销，减少对计算机硬件资源的需求和消耗，提高模型推理效率。

参见图 2，图 2 示出了本说明书一个实施例提供的一种数据处理方法的流程图，具体包括以下步骤。

具体的，所述数据处理方法应用于支持机器学习模型计算的计算机硬件。

步骤 202：确定目标模型的目标计算图以及输入所述目标模型的目标数据，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计算图。

其中，目标模型可以理解为任意一种的数据处理模型，如文本处理模型、图像处理模型，在此不做限定；目标计算图可以理解为静态计算图，根据目标模型的模型结构以及运算流程构建得到，且为一个在目标模型执行推理流程前定义并编译好的、不可变的计算图。

具体的，确定目标模型的目标计算图，以及确定输入目标模型的目标数据，输入的目标数据会被转换为目标计算图的输入节点，通过模型推理在目标模型中进行数据传输，并得到对目标数据处理后的数据处理结果。

实际应用中，在确定目标模型的静态计算图的情况下，由于静态计算图较动态计算图来说，允许更高级别的优化，如内存预分配、计算图优化和硬件特定优化等，因此在部署和大规模生产环境时，利用静态计算图编译的目标模型相比动态计算图能够提升更高的性能，执行效率更快。

在本说明书一个或多个实施例中，由于动态计算图的运算和构建是一起进行的，也就是先计算动态计算图中前面的节点的值，再根据这些得到的值构建后面的动态计算图，而在需要对构建好的动态计算图进行修改的情况下，整个动态计算图需重新构建，也就需要重新编译动态计算图，编译过程会存在大量计算机硬件资源的开销，因此为节省编译带来的计算机硬件资源的开销，确定目标模型的静态计算图。具体实现方式如下所述：

所述确定目标模型的目标计算图以及输入所述目标模型的目标数据，包括：

根据所述支持机器学习模型计算的计算机硬件的硬件资源，确定目标模型的目标计算图以及输入所述目标模型的目标数据。

其中，硬件资源包括但不限于 CPU（Central Processing Unit，中央处理器）、内存、存储器、网络等。

具体的，在利用目标模型进行实际推理流程前，需要对目标模型对应的计算图进行编译，即将源代码转换为计算机执行的目标代码；而实际应用中，在进行计算图编

译的情况下，不论是静态计算图还是动态计算图，均需要大量的计算机硬件资源的开销。

例如，编译过程中需要CPU来完成大量的计算，如词法分析、语法分析等；需要内存来存储数据，包括源代码、中间代码、目标代码等；需要利用存储器资源来读取和写入大量的文件，如源代码文件、头文件等；且在需要从网络下载依赖项或库的情况下，要用到网络资源。

由于动态计算图一边运算一边构建的特性，在需要对构建好的动态计算图进行修改的情况下，会重新构建动态计算图，此时则需要进行二次编译，即重新对新构建的动态计算图机进行编译，存在消耗大量计算机硬件资源的问题。

在本说明书实施例提供的数据处理方法，应用于支持机器学习模型计算的计算机硬件的基础上，根据所述支持机器学习模型计算的计算机硬件的硬件资源，来确定目标模型的目标计算图，即在计算机硬件的硬件资源受限、配置不足的情况下，可将目标模型转换为静态计算图，在需要对静态计算图进行修改的情况下，利用动态的调用数据处理计算图的方式实现，因此不会存在二次编译的情况，即无需重新编译静态计算图，能够节省大量计算图编译的开销，减少对计算机硬件资源的需求和消耗。

在本说明书一个或多个实施例中，为准确的构建目标模型的目标计算图，在获得目标模型的情况下，根据目标模型确定目标计算图中的节点以及边。具体实现方式如下所述：

所述确定所述目标模型的目标计算图，包括：

获取所述目标模型，将所述目标模型进行数据处理的处理操作作为所述节点，将所述处理操作之间的关联关系作为边；

根据所述节点和所述边，构建所述目标计算图。

其中，节点可以理解为，一个具体的数学运算或者数据处理操作；边用于表示数据流的方向和数据处理操作之间的依赖关系。

具体的，首先根据机器学习任务的需求设计并实现目标模型的模型结构，从而获得目标模型，或者可以获取一个已设计完成的目标模型，将目标模型在进行数据处理过程中的处理操作作为构建目标计算图的节点，例如，在神经网络中，节点可以代表矩阵乘法、加法、激活函数应用、损失函数计算、优化器更新步骤等；并根据各个处理操作之间的关联关系作为构建目标计算图的边，比如，在目标模型推理过程中，输入数据经过第一个卷积层的处理后得到的输出，作为下一个卷积层或池化层的输入，这种前后层之间的联系就可以通过边来表达；从而根据节点和边，构建目标计算图。

本说明实施例提供的数据处理方法，通过构建目标计算图的方式，能够明确表示出目标计算图中可以并行执行的计算步骤，这有利于大规模数据集上的高效的进行模型推理流程。在分布式环境下，根据目标计算图的结构还可将数据处理任务分解到不同的计算资源上，提高计算机系统的性能和稳定性。

在本说明书一个或多个实施例中，首先根据节点和边，构建初始计算图，在初始计算图也为静态计算图的情况下，通过编译初始计算图中的各个节点，实现图优化，得到目标计算图。具体实现方式如下所述：

所述根据所述节点和所述边，构建所述目标计算图，包括：

根据所述节点和所述边，构建初始计算图；

对所述初始计算图中的目标节点以及目标边进行冗余消除操作，获得所述目标模型的目标计算图，以使目标模型在目标处理器低使用率的状态下运行。

其中，目标节点可以理解为，需要优化或者调整的节点，例如重复的计算节点、可以融合的节点等；目标边则可以理解为，与目标节点相关联的边，如在对目标节点

删除的情况下，需要对与其相关联的目标边也进行删除；目标处理器可以理解为，运行目标模型的处理器，用于为目标模型进行模型推理流程时提供计算资源，如中央处理器、图形处理器等，在此不做限定。

具体的，根据节点以及边，构建初始计算图的情况下，由于初始计算图为静态计算图，可以对初始计算图中的节点进行编译，实现对初始计算图的图优化，包括但不限于常量折叠、算子融合、死代码删除等冗余消除操作，从而得到更简洁、高效的目标计算图。

在通过消除不必要的重复计算或合并连续的操作，获得目标计算图的情况下，可以减少目标模型推理过程中的计算步骤和内存访问次数，且去除目标节点后，不需要为这些无效或重复的计算分配额外的内存，有助于降低内存占用，实现目标处理器能够以低使用率的状态运行目标模型。

本说明实施例提供的数据处理方法，通过对初始计算图中的节点进行冗余消除操作，能够提高目标模型的执行性能，更简洁高效的目标计算图也有助于目标模型在不同硬件平台上的部署与移植。

步骤 204：从所述目标计算图中确定处理所述目标数据的当前节点，其中，所述当前节点为所述多个节点中的任意一个节点。

其中，当前节点可以理解为，目标模型按照目标计算图的结构顺序执行运算时所处的具体节点。

具体的，在目标模型执行推理过程时，引擎根据目标计算图逐个节点的进行计算，其中，在深度学习框架中，引擎可以理解为处理目标计算图的内核部分，能够解析目标计算图的结构，调度节点执行顺序，并利用底层计算机硬件进行并行计算和内存管理。

实际应用中，当目标数据输入目标模型后，目标数据会首先被传递到目标计算图中的第一个节点进行处理，该节点可以为对应于目标模型输入层的节点，负责对目标数据进行初步转换；随着计算流程的推进，每个节点根据其定义的处理操作和关联边接收上一个节点的输出作为输入，并产生新的输出。

步骤 206：确定所述当前节点对应的数据处理计算图，根据所述当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果。

其中，数据处理计算图可以理解为，在目标计算图之外的、进行数据处理的静态计算图。

具体的，可以确定当前节点对应的数据处理计算图，根据是否需要对目标处理数据进行前置和/或后置处理的需求，利用当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果。

实际应用中，在数据处理计算图也为静态计算图的情况下，通过目标计算图与数据处理计算图结合实现的计算图框架，能够实现多变的动态计算逻辑，并且数据处理计算图可以根据实际需求随时进行适应性的调用，灵活度大。

在本说明书一个或多个实施例中，通过判断当前节点对应的目标模型的目标结构单元是否在前置处理表中，确定目标结构单元对应的前置数据处理计算图。具体实现方式如下所述：

所述确定所述当前节点对应的数据处理计算图，包括：

根据目标映射表，确定所述当前节点对应的所述目标模型的目标结构单元；

在所述目标结构单元在前置处理表中的情况下，从所述前置处理表中确定所述目标结构单元对应的前置数据处理计算图，其中，所述前置处理表中包括所述目标结构单元与所述前置数据处理计算图的映射关系，所述前置数据处理计算图用于对目标处

理数据进行前置处理。

其中，目标映射表可以理解为，描述目标计算图中的节点与目标模型的结构单元之间的对应关系的数据结构；其中，目标模型的结构单元可以理解为，构建目标模型的基本组成元素，例如在神经网络中，结构单元可以理解为网络层，包括但不限于输入层、隐藏层（全连接层、卷积层、循环层等）、输出层等；目标结构单元可以理解为，根据目标映射表确定的、与当前节点对应的结构单元。

前置数据处理计算图可以理解为，执行当前节点之前，对目标数据进行处理的计算图。

具体的，首先根据目标映射表确定当前节点对应的目标模型的目标结构单元，查找目标结构单元是否在前置处理表中，若在，从所述前置处理表中确定所述目标结构单元对应的前置数据处理计算图，若不在，则根据当前节点对目标数据进行处理。

如图 3 所示，图 3 示出了本说明书实施例提供的一种前置处理表的示意图。

例如在 Controlnet 中，通常将输入数据与一个目标计算图外的常量值进行相加，实现对输入数据的处理；此时根据目标映射表，确定当前节点对应的目标模型的目标结构单元为结构单元 1，通过如图 3 所示的前置处理表，结构单元 1 指向一个外部加法计算图（即前置数据处理计算图），其中，输入[0]=获取输入()（input[0]=getInputs()）表示获取输入数据；输入[1]=常量()（input[1]=Constant()）表示设置一个常量值；输出[0]=输入[0]+输入[1]（output[0]=input[0]+input[1]）表示将输入数据与常量值相加，得到输出结果；返回输出[0]（return output[0]）表示返回计算后的输出结果；由此，经过外部加法计算图，实现在当前节点之前将输入数据与外部的数据进行加法，完成外部加法计算图的插入。

本说明实施例提供的数据处理方法，在根据目标映射表确定的目标结构单元在前置处理表中的情况下，从前置处理表中确定目标结构单元对应的前置数据处理计算图，以通过前置数据处理计算图对目标数据进行处理，实现在当前节点前动态插入计算图。

在本说明书一个或多个实施例中，根据目标计算图中的节点以及目标模型的结构单元之间的映射关系，构建目标映射表。具体实现方式如下所述：

所述根据目标映射表，确定所述当前节点对应的目标模型的目标结构单元之前，还包括：

确定构建所述目标模型的结构单元，其中，所述结构单元为目标模型的网络层；

确定所述目标计算图中各节点对应的结构单元；

根据所述目标计算图中的节点以及所述目标模型的结构单元之间的映射关系，构建所述目标映射表。

具体的，以一个包含多个结构单元的 UNET 模型为例，其中，UNET（U-Net）是一种卷积神经网络（Convolutional Neural Network, CNN）架构，U-Net 的名字来源于其独特的 U 形结构。如图 4 所示，图 4 示出了本说明书实施例提供的一种 UNET 模型结构示意图；UNET 模型的结构主要包含两部分：编码器块（Encoder block）和解码器块（Decoder block）。

而编码器块通常包含多层结构，每一层又包括自注意力机制（Self-Attention）和前馈神经网络（Feedforward Neural Network，简称 FFN）等，因此，每个编码器块包括有子模块，类似的，解码器块包括有子模块。

在将目标模型转换为目标计算图的过程中时，目标模型会被转换为包含若干节点的静态计算图，记录目标计算图中的各节点在目标模型中的结构单元位置和索引，记录为目标映射表，即节点与每个 block 及其子模块之间的对应关系表；如图 5 所示，图 5 示出了本说明书实施例提供的一种目标映射表的示意图，节点 1 与结构单元 0.0

对应，节点 2 与结构单元 0.1 对应，节点 3 与结构单元 0.2 对应，节点 4 与结构单元 1.0 对应，直到节点 n 与结构单元 n.n 对应。

本说明实施例提供的数据处理方法，通过构建目标映射表，实现根据目标计算图中当前节点，找到其所对应的目标模型内部的具体结构单元，并准确地进行相关的数据处理和运算；从而更清晰地了解并控制整个目标模型计算流程，确保数据在经过目标计算图中的每个节点时，都能够按照目标模型设计的要求进行处理。

在本说明书一个或多个实施例中，通过前置数据处理计算图、当前节点、后置数据处理计算图，根据实际情况准确的进行数据处理，且通过前置数据处理计算图以及后置数据处理计算图实现对目标计算图中节点的插入、删除等功能。具体实现方式如下所述：

所述根据所述当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果，包括：

根据所述前置数据处理计算图，处理所述目标数据，获得前置处理数据；

在根据所述前置处理数据确定执行所述当前节点的情况下，根据所述当前节点处理所述前置处理数据，获得当前处理数据；

在目标结构单元在后置处理表中的情况下，从所述后置处理表中确定所述目标结构单元对应的后置数据处理计算图，其中，所述后置处理表中包括所述目标结构单元与所述后置数据处理计算图的映射关系，所述后置数据处理计算图用于对目标处理数据进行后置处理；

根据所述后置数据处理计算图，处理所述当前处理数据，获得所述数据处理结果。

其中，前置处理数据可以理解为，利用前置数据处理计算图对目标数据进行处理获得的处理数据；当前处理数据可以理解为，利用当前节点对前置数据处理计算图输出的前置处理数据，进行处理获得的处理结果。

后置数据处理计算图可以理解为，执行当前节点之后，对目标处理数据进行处理的计算图，其中，在获得前置处理数据且不执行当前节点的情况下，目标处理数据可以理解为前置处理数据，即利用后置数据处理计算图对前置处理数据进行处理，获得数据处理结果；在获得前置处理数据且执行当前节点的情况下，目标处理数据可以理解为当前处理数据，即利用后置数据处理计算图对当前处理数据，获得数据处理结果；或者，在当前节点对应的结构单元不在前置处理表中的情况下，利用当前节点对目标数据进行处理，获得目标处理结果，此时，目标处理数据可以理解为目标处理结果，即利用后置数据处理计算图对目标处理结果进行处理，获得数据处理结果。

具体的，在目标结构单元在前置处理表中的情况下，从前置处理表中确定目标结构单元对应的前置数据处理计算图，利用前置数据处理计算图，处理目标数据，获得前置处理数据；且根据前置处理数据判断是否要继续执行当前节点（例如，如果前置数据处理计算图用于检测环境准备就绪，为在执行当前节点之前的必要步骤，当前置处理数据的状态为“就绪”时才继续执行后续节点），若是，则根据所述当前节点处理前置处理数据，获得当前处理数据，此时即在当前节点前实现了计算图的添加。

在执行当前节点之后，判断目标结构单元是否在后置处理表中，若是，从后置处理表中目标结构单元对应的后置数据处理计算图，并根据后置数据处理计算图，处理当前处理数据，获得数据处理结果；若否，则将当前处理数据，确定为数据处理结果。

所述根据所述当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果，包括：

根据所述前置数据处理计算图，处理所述目标数据，获得前置处理数据；

在根据所述前置处理数据确定不执行所述当前节点、且目标结构单元在后置处理

表中的情况下，从所述后置处理表中确定所述目标结构单元对应的后置数据处理计算图，根据所述后置数据处理计算图，处理所述前置处理数据，获得所述数据处理结果，或者，

在根据所述前置处理数据确定不执行所述当前节点、且目标结构单元不在后置处理表的情况下，将所述前置处理数据确定为数据处理结果。

具体的，在目标结构单元在前置处理表中的情况下，从前置处理表中确定目标结构单元对应的前置数据处理计算图，利用前置数据处理计算图，处理目标数据，获得前置处理数据，而在根据前置处理数据确定不继续执行当前节点的情况下，判断目标结构单元是否在后置处理表中，若是，利用后置数据处理计算图，处理前置处理数据，获得数据处理结果；若否，即目标结构单元不在后置处理表中，将前置处理数据，确定为数据处理结果；在不继续执行当前节点的情况下，实现了对当前节点的删除。

本说明实施例提供的数据处理方法，通过在目标模型推理过程中，在执行当前节点之前/之后插入判断是否存在前置数据处理计算图、后置数据处理计算图的方式，实现在目标计算图中的动态的插入其他计算流程。

在本说明书一个或多个实施例中，在目标结构单元不在前置处理表中的情况下，利用当前节点对目标数据进行处理。具体实现方式如下所述：

所述确定所述当前节点对应的数据处理计算图，包括：

根据目标映射表，确定所述当前节点对应的所述目标模型的目标结构单元；

在所述目标结构单元不在前置处理表中的情况下，根据所述当前节点处理所述目标数据，获得当前处理数据；

在目标结构单元在后置处理表中的情况下，从所述后置处理表中确定所述目标结构单元对应的后置数据处理计算图。

具体的，在目标结构单元不在前置处理表中的情况下，则按照目标计算图中的顺序执行当前节点，利用当前节点对目标数据进行处理，获得当前处理数据，并在执行当前节点之后，判断目标结构单元是否在后置处理表，若在，则从后置处理表中所述目标结构单元对应的后置数据处理计算图。

本说明实施例提供的数据处理方法，通过前置处理表和后置处理表，能够对复杂模型中的各个结构单元的数据处理过程进行有序管理，保证数据在不同阶段按照预设的规则流动和处理。

在本说明书一个或多个实施例中，在执行当前节点之后，可在目标结构单元在后置处理表中的情况下，确定后置数据处理计算图，并利用后置数据处理计算图对当前处理数据进行处理。具体实现方式如下所述：

所述根据所述当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果，包括：

根据所述后置数据处理计算图，处理所述当前处理数据，获得所述数据处理结果。

具体的，在目标结构单元在后置处理表中的情况下，从后置处理表中确定目标结构单元对应的后置数据处理计算图，从而利用后置数据处理计算图，处理当前处理数据，获得所述数据处理结果。

本说明实施例提供的数据处理方法，在执行当前节点之后，判断目标结构单元是否在后置处理表中，在目标结构单元在后置处理表时，则可以从后置处理表中找到对应的后置数据处理计算图，确保数据在进入下一节点之前经过必要的预处理，提高数据传递的准确性。

在本说明书一个或多个实施例中，目标模型的目标计算图存在多个节点，在执行完整个当前节点的计算流程后，判断是否还存在当前节点的下一节点，若存在，则将

数据处理结果传递给下一节点，并将下一节点确定为当前节点。具体实现方式如下所述：

所述获得所述数据处理结果之后，还包括：

判断是否存在所述当前节点的下一节点，

若存在，将所述当前节点的下一节点确定为当前节点，并将所述数据处理结果作为所述目标数据，执行所述确定所述当前节点对应的数据处理计算图，根据所述当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果的步骤。

若不存在，输出所述数据处理结果。

在目标模型的推理过程中，检查当前节点是否还有后续的节点；如果当前节点有下一节点，则将下一节点设置为当前节点，并将当前节点得到的数据处理结果作为目标数据，然后，根据新确定的当前节点执行所述确定所述当前节点对应的数据处理计算图，根据所述当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果的步骤，继续对目标数据进行相应的处理，获得数据处理结果；重复这个过程，直到目标计算图中的节点都执行完毕。

在当前节点没有下一节点的情况下，意味着模型推理流程已经到达了目标计算图的末端，此时，不再继续迭代到下一节点，而是直接输出当前的数据处理结果。

本说明实施例提供的数据处理方法，通过该种方式，确保了输入的目标数据能够按照目标计算图的设计，逐层经过各个节点的运算和处理，最终得出模型预测或推理的结果。

本说明书实施例提供的数据处理方法，通过确定目标模型的目标计算图，在给目标模型输入目标数据的情况下，由于目标计算图为在目标模型运行时不能进行更新的计算图，即目标计算图在目标模型运行前就被完全定义并编译，这有利于在进行模型推理前进行对目标计算图的编译进行优化，降低处理器的运算负担，提高处理器的执行效率；同时，通过灵活地调用数据处理计算图，能确保目标模型对目标数据进行高效、准确的前置处理或后置处理操作，实现在目标计算图中动态插入计算流程，且无需二次编译，节省计算图编译的开销，减少对计算机硬件资源的需求和消耗，提高模型推理效率。

参见图 6，图 6 示出了本说明书一个实施例提供的一种数据处理方法的单节点处理过程流程图，具体包括以下步骤。

步骤 602：根据目标映射表，得到当前节点对应的目标结构单元。

实际应用中，获取以动态图架构实现的目标模型，并将目标模型的动态计算图转换为静态计算图，并根据静态计算图中各个节点与模型结构中的各个结构单元的对应关系，构建 node->block 映射表（即目标映射表），node 即静态计算图中各个节点，block 即模型结构中的结构单元。

在执行模型推理过程的情况下，根据静态计算图逐节点进行计算；而针对当前节点首先根据上述 node->block 映射表得到当前节点对应的 block。

步骤 604：判断目标结构单元是否在前置处理表中，若是，执行步骤 606；若否，则执行步骤 610。

判断当前节点对应的 block 是否在 pre_process 表（即上述实施例中的前置处理表）中，若在，执行步骤 606，即执行 pre_process 表中该 block 对应的计算图，若否，执行 610，即直接执行当前节点计算。

步骤 606：执行前置处理表中目标结构单元对应的前置数据处理计算图。

在当前节点对应的 block 在 pre_process 表中的情况下，执行 pre_process 表中的该 block 对应的计算图，并得到返回数据（即上述实施例中的前置处理数据）以及是否继

续计算的结果。

步骤 608：判断是否继续计算，若是，执行步骤 610；若否，则执行步骤 612。

在确定继续计算的情况下，执行当前节点计算，此时相当于在当前节点前插入了计算图，实现了计算图的添加功能；在不继续计算的情况下，则跳过当前节点的计算，并将返回数据传递给当前节点的下一个节点，即实现了当前节点的删除功能。

步骤 610：执行当前节点计算。

步骤 612：判断目标结构单元是否在后置处理表中，若是，执行步骤 614；若否，则执行步骤 616。

类似的，在执行完当前节点后，也根据 node->block 映射表得到当前节点的 block，而后查找 block 是否在 after_process 表（后置处理表）中，如果在 after_process 表中，则执行步骤 614，即执行 after_process 表中的 block 对应的计算图；若不在 after_process 表中，则直接结束当前节点的整个计算流程。

步骤 614：执行后置处理表中目标结构单元对应的后置数据处理计算图。

在当前节点的 block 在 after_process 表中的情况下，执行 after_process 表中的该 block 对应的计算图，并得到返回数据（即上述实施例中的数据处理结果）。

步骤 616：结束。

完成当前节点的整个计算流程，并将得到的数据处理结果传递给当前节点的下一节点。

本说明书实施例提供的数据处理方法，通过灵活地调用数据处理计算图，能确保目标模型对目标数据进行高效、准确的前置处理或后置处理操作，实现在目标计算图中动态插入计算流程，且无需二次编译，节省计算图编译的开销，减少对计算机硬件资源的需求和消耗，提高模型推理效率。

参见图 7，图 7 示出了本说明书一个实施例提供的一种文本处理方法的流程图，具体包括以下步骤。

步骤 702：确定目标模型的目标计算图以及输入所述目标模型的目标文本，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计算图；

步骤 704：从所述目标计算图中确定处理所述目标文本的当前节点，其中，所述当前节点为所述多个节点中的任意一个节点；

步骤 706：确定所述当前节点对应的文本处理计算图，根据所述当前节点和/或所述文本处理计算图，处理所述目标文本，获得文本处理结果。

具体的，以目标模型为一个深度学习的文本分类模型为例，对该文本处理方法进行详细说明。

文本分类模型以动态图形式构建，首先，将其转换为静态计算图以便于管理和优化推理过程。根据模型结构（如词嵌入层、卷积层、循环神经网络层、全连接层等），构建一个 node->block 映射表，其中每个节点代表静态计算图中的运算步骤，而 block 则对应文本分类模型实际的结构单元。

在模型推理阶段，在输入目标文本的情况下，根据静态计算图按顺序执行节点计算，假设当前节点对应的是词嵌入层（block），在前置处理表中查找到该 block 存在前置文本处理计算图，比如对输入的目标文本进行分词或者填充等预处理，得到预处理后的数据，并判断是否满足继续计算的条件（如数据有效、长度合适等）；若满足继续计算条件，则执行当前节点（词嵌入层）的实际计算，将预处理后的文本转化为词向量表示。

当前节点计算完成后，进一步检查该当前节点在后置处理表中是否存在前置文本

处理计算图，比如进行数据归一化或者其他后处理步骤；如果存在，则执行相应的后置数据处理计算图并获取结果；若不存在，则直接结束当前节点计算流程，将计算结果传递给下一个节点。

本说明书实施例提供的文本处理方法，通过调用数据处理计算图，可以灵活地插入或删除静态计算图的部分，以适应不同的数据预处理需求或后处理需求，从而优化整个目标模型的推理效率与效果。

参见图 8，图 8 示出了本说明书一个实施例提供的一种图像处理方法的流程图，具体包括以下步骤。

步骤 802：确定目标模型的目标计算图以及输入所述目标模型的目标图像，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计算图；

步骤 804：从所述目标计算图中确定处理所述目标图像的当前节点，其中，所述当前节点为所述多个节点中的任意一个节点；

步骤 806：确定所述当前节点对应的图像处理计算图，根据所述当前节点和/或所述图像处理计算图，处理所述目标图像，获得图像处理结果。

具体的，以目标模型为图像分类模型为例，对该图像处理方法进行详细说明。

该图像分类模型包含多个结构单元，如输入层、卷积层、池化层、全连接层和输出层等。在将动态图转换为静态计算图后，首先，根据图像分类模型的静态计算图，构建节点与模型结构单元之间的映射关系。比如，某个节点对应着第一个卷积层的计算。

在模型推理阶段，在输入目标图像的情况下，根据静态计算图按顺序逐个节点执行节点计算，根据 node->block 映射表，得知当前节点对应的是第一个卷积层。

判断该卷积层是否在前置处理表中，若在 pre_process 表中，执行前置处理表中该卷积层对应的计算图，根据该计算图对输入的目标图像进行预处理操作，得到标准化后的图像数据；并确定需要继续执行当前节点计算，执行当前节点（即第一个卷积层）的计算，对预处理过的图像数据进行特征提取，获得特征提取结果；判断该当前节点是否在后置处理表中，若不存在则将特征提取结果传递给下一个节点。

本说明书实施例提供的图像处理方法，通过调用数据处理计算图，可以灵活地插入或删除静态计算图的部分，以适应不同的数据预处理需求或后处理需求，从而优化整个目标模型的推理效率与效果。

与上述方法实施例相对应，本说明书还提供了数据处理装置实施例，图 9 示出了本说明书一个实施例提供的一种数据处理装置的结构示意图。如图 9 所示，该装置包括：

数据确定模块 902，被配置为确定目标模型的目标计算图以及输入所述目标模型的目标数据，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计算图；

节点确定模块 904，被配置为从所述目标计算图中确定处理所述目标数据的当前节点，其中，所述当前节点为所述多个节点中的任意一个节点；

结果获得模块 906，被配置为确定所述当前节点对应的数据处理计算图，根据所述当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果。

可选地，所述结果获得模块 906，进一步被配置为：

根据目标映射表，确定所述当前节点对应的所述目标模型的目标结构单元；

在所述目标结构单元在前置处理表中的情况下，从所述前置处理表中确定所述目

标结构单元对应的前置数据处理计算图，其中，所述前置处理表中包括所述目标结构单元与所述前置数据处理计算图的映射关系，所述前置数据处理计算图用于对目标处理数据进行前置处理。

可选地，所述结果获得模块 906，进一步被配置为：

根据所述前置数据处理计算图，处理所述目标数据，获得前置处理数据；

在根据所述前置处理数据确定执行所述当前节点的情况下，根据所述当前节点处理所述前置处理数据，获得当前处理数据；

在目标结构单元在后置处理表中的情况下，从所述后置处理表中确定所述目标结构单元对应的后置数据处理计算图，其中，所述后置处理表中包括所述目标结构单元与所述后置数据处理计算图的映射关系，所述后置数据处理计算图用于对目标处理数据进行后置处理；

根据所述后置数据处理计算图，处理所述当前处理数据，获得所述数据处理结果。

可选地，所述结果获得模块 906，进一步被配置为：

根据所述前置数据处理计算图，处理所述目标数据，获得前置处理数据；

在根据所述前置处理数据确定不执行所述当前节点、且目标结构单元在后置处理表中的情况下，从所述后置处理表中确定所述目标结构单元对应的后置数据处理计算图，根据所述后置数据处理计算图，处理所述前置处理数据，获得所述数据处理结果，或者，

在根据所述前置处理数据确定不执行所述当前节点、且目标结构单元不在后置处理表的情况下，将所述前置处理数据确定为数据处理结果。

可选地，所述结果获得模块 906，进一步被配置为：

根据目标映射表，确定所述当前节点对应的所述目标模型的目标结构单元；

在所述目标结构单元不在前置处理表中的情况下，根据所述当前节点处理所述目标数据，获得当前处理数据；

在目标结构单元在后置处理表中的情况下，从所述后置处理表中确定所述目标结构单元对应的后置数据处理计算图。

可选地，所述结果获得模块 906，进一步被配置为：

根据所述后置数据处理计算图，处理所述当前处理数据，获得所述数据处理结果。

可选地，所述数据确定模块 902，进一步被配置为：

获取所述目标模型，将所述目标模型进行数据处理的处理操作作为所述节点，将所述处理操作之间的关联关系作为边；

根据所述节点和所述边，构建所述目标计算图。

可选地，所述数据确定模块 902，进一步被配置为：

根据所述节点和所述边，构建初始计算图；

对所述初始计算图中的目标节点以及目标边进行冗余消除操作，获得所述目标模型的目标计算图，以使目标模型在目标处理器低使用率的状态下运行。

所述装置，还包括：

构建模块，被配置为确定构建所述目标模型的结构单元，其中，所述结构单元为目标模型的网络层；

确定所述目标计算图中各节点对应的结构单元；

根据所述目标计算图中的节点以及所述目标模型的结构单元之间的映射关系，构建所述目标映射表。

所述装置，还包括：

后处理模块，被配置为判断是否存在所述当前节点的下一节点，

若存在，将所述当前节点的下一节点确定为当前节点，并将所述数据处理结果作为所述目标数据，执行所述确定所述当前节点对应的数据处理计算图，根据所述当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果的步骤，

若不存在，输出所述数据处理结果。

可选地，所述数据确定模块902，进一步被配置为：

根据所述支持机器学习模型计算的计算机硬件的硬件资源，确定目标模型的目标计算图以及输入所述目标模型的目标数据。

上述为本实施例的一种数据处理装置的示意性方案。需要说明的是，该数据处理装置的技术方案与上述的数据处理方法的技术方案属于同一构思，数据处理装置的技术方案未详细描述的细节内容，均可以参见上述数据处理方法的技术方案的描述。

与上述方法实施例相对应，本说明书还提供了文本处理装置实施例，图10示出了本说明书一个实施例提供的一种文本处理装置的结构示意图。如图10所示，该装置包括：

文本确定模块1002，被配置为确定目标模型的目标计算图以及输入所述目标模型的目标文本，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计算图；

节点确定模块1004，被配置为从所述目标计算图中确定处理所述目标文本的当前节点，其中，所述当前节点为所述多个节点中的任意一个节点；

结果获得模块1006，被配置为确定所述当前节点对应的文本处理计算图，根据所述当前节点和/或所述文本处理计算图，处理所述目标文本，获得文本处理结果。

上述为本实施例的一种文本处理装置的示意性方案。需要说明的是，该文本处理装置的技术方案与上述的数据处理方法的技术方案属于同一构思，文本处理装置的技术方案未详细描述的细节内容，均可以参见上述数据处理方法的技术方案的描述。

与上述方法实施例相对应，本说明书还提供了图像处理装置实施例，图11示出了本说明书一个实施例提供的一种图像处理装置的结构示意图。如图11所示，该装置包括：

图像确定模块1102，被配置为确定目标模型的目标计算图以及输入所述目标模型的目标图像，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计算图；

节点确定模块1104，被配置为从所述目标计算图中确定处理所述目标图像的当前节点，其中，所述当前节点为所述多个节点中的任意一个节点；

结果获得模块1106，被配置为确定所述当前节点对应的图像处理计算图，根据所述当前节点和/或所述图像处理计算图，处理所述目标图像，获得图像处理结果。

上述为本实施例的一种图像处理装置的示意性方案。需要说明的是，该图像处理装置的技术方案与上述的数据处理方法的技术方案属于同一构思，图像处理装置的技术方案未详细描述的细节内容，均可以参见上述数据处理方法的技术方案的描述。

图12示出了根据本说明书一个实施例提供的一种计算设备1200的结构框图。该计算设备1200的部件包括但不限于存储器1210和处理器1220。处理器1220与存储器1210通过总线1230相连接，数据库1250用于保存数据。

计算设备1200还包括接入设备1240，接入设备1240使得计算设备1200能够经由一个或多个网络1260通信。这些网络的示例包括公用交换电话网（PSTN，Public Switched Telephone Network）、局域网（LAN，Local Area Network）、广域网（WAN，

Wide Area Network）、个域网（PAN，Personal Area Network）或诸如因特网的通信网络的组合。接入设备 1240 可以包括有线或无线的任何类型的网络接口（例如，网络接口卡（NIC，network interface controller））中的一个或多个，诸如 IEEE802.11 无线局域网（WLAN，Wireless Local Area Network）无线接口、全球微波互联接入（Wi-MAX，Worldwide Interoperability for Microwave Access）接口、以太网接口、通用串行总线（USB，Universal Serial Bus）接口、蜂窝网络接口、蓝牙接口、近场通信（NFC，Near Field Communication）。

在本说明书的一个实施例中，计算设备 1200 的上述部件以及图 12 中未示出的其他部件也可以彼此相连接，例如通过总线。应当理解，图 12 所示的计算设备结构框图仅仅是出于示例的目的，而不是对本说明书范围的限制。本领域技术人员可以根据需要，增添或替换其他部件。

计算设备 1200 可以是任何类型的静止或移动计算设备，包括移动计算机或移动计算设备（例如，平板计算机、个人数字助理、膝上型计算机、笔记本计算机、上网本等）、移动电话（例如，智能手机）、可佩戴的计算设备（例如，智能手表、智能眼镜等）或其他类型的移动设备，或者诸如台式计算机或个人计算机（PC，Personal Computer）的静止计算设备。计算设备 1200 还可以是移动式或静止式的服务器。

其中，处理器 1220 用于执行如下计算机程序/指令，该计算机程序/指令被处理器执行时实现上述数据处理方法、或文本处理方法、或图像处理方法的步骤。

本说明书中的各个实施例均采用递进的方式描述，各个实施例之间相同相似的部分互相参见即可，每个实施例重点说明的都是与其他实施例的不同之处。尤其，对于计算设备实施例而言，由于其基本相似于数据处理方法实施例，所以描述的比较简单，相关之处参见数据处理方法实施例的部分说明即可。

本说明书一实施例还提供一种计算机可读存储介质，其存储有计算机程序/指令，该计算机程序/指令被处理器执行时实现上述数据处理方法、或文本处理方法、或图像处理方法的步骤。

本说明书中的各个实施例均采用递进的方式描述，各个实施例之间相同相似的部分互相参见即可，每个实施例重点说明的都是与其他实施例的不同之处。尤其，对于计算机可读存储介质实施例而言，由于其基本相似于数据处理方法实施例，所以描述的比较简单，相关之处参见数据处理方法实施例的部分说明即可。

本说明书一实施例还提供一种计算机程序产品，包括计算机程序/指令，该计算机程序/指令被处理器执行时实现上述数据处理方法、或文本处理方法、或图像处理方法的步骤。

上述为本实施例的一种计算机程序产品的示意性方案。需要说明的是，该计算机程序产品的技术方案与上述的数据处理方法的技术方案属于同一构思，计算机程序产品的技术方案未详细描述的细节内容，均可以参见上述数据处理方法的技术方案的描述。

上述对本说明书特定实施例进行了描述。其它实施例在所附权利要求书的范围内。在一些情况下，在权利要求书中记载的动作或步骤可以按照不同于实施例中的顺序来执行并且仍然可以实现期望的结果。另外，在附图中描绘的过程不一定要求示出的特定顺序或者连续顺序才能实现期望的结果。在某些实施方式中，多任务处理和并行处理也是可以的或者可能是有利的。

所述计算机指令包括计算机程序代码，所述计算机程序代码可以为源代码形式、对象代码形式、可执行文件或某些中间形式等。所述计算机可读介质可以包括：能够携带所述计算机程序代码的任何实体或装置、记录介质、U 盘、移动硬盘、磁碟、光

盘、计算机存储器、只读存储器（ROM, Read-Only Memory）、随机存取存储器（RAM, Random Access Memory）、电载波信号、电信信号以及软件分发介质等。需要说明的是，所述计算机可读介质包含的内容可以根据专利实践的要求进行适当的增减，例如在某些地区，根据专利实践，计算机可读介质不包括电载波信号和电信信号。

需要说明的是，对于前述的各方法实施例，为了简便描述，故将其都表述为一系列的动作组合，但是本领域技术人员应该知悉，本说明书实施例并不受所描述的动作顺序的限制，因为依据本说明书实施例，某些步骤可以采用其它顺序或者同时进行。其次，本领域技术人员也应该知悉，说明书中所描述的实施例均属于优选实施例，所涉及的动作和模块并不一定都是本说明书实施例所必须的。

在上述实施例中，对各个实施例的描述都各有侧重，某个实施例中没有详述的部分，可以参见其它实施例的相关描述。

以上公开的本说明书优选实施例只是用于帮助阐述本说明书。可选实施例并没有详尽叙述所有的细节，也不限制该发明仅为所述的具体实施方式。显然，根据本说明书实施例的内容，可作很多的修改和变化。本说明书选取并具体描述这些实施例，是为了更好地解释本说明书实施例的原理和实际应用，从而使所属技术领域技术人员能很好地理解和利用本说明书。本说明书仅受权利要求书及其全部范围和等效物的限制。

权 利 要 求 书

1.一种数据处理方法，包括：

确定目标模型的目标计算图以及输入所述目标模型的目标数据，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计算图；

从所述目标计算图中确定处理所述目标数据的当前节点，其中，所述当前节点为所述多个节点中的任意一个节点；

确定所述当前节点对应的数据处理计算图，根据所述当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果。

2.根据权利要求 1 所述的数据处理方法，所述确定所述当前节点对应的数据处理计算图，包括：

根据目标映射表，确定所述当前节点对应的所述目标模型的目标结构单元；

在所述目标结构单元在前置处理表中的情况下，从所述前置处理表中确定所述目标结构单元对应的前置数据处理计算图，其中，所述前置处理表中包括所述目标结构单元与所述前置数据处理计算图的映射关系，所述前置数据处理计算图用于对目标处理数据进行前置处理。

3.根据权利要求 2 所述的数据处理方法，所述根据所述当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果，包括：

根据所述前置数据处理计算图，处理所述目标数据，获得前置处理数据；

在根据所述前置处理数据确定执行所述当前节点的情况下，根据所述当前节点处理所述前置处理数据，获得当前处理数据；

在目标结构单元在后置处理表中的情况下，从所述后置处理表中确定所述目标结构单元对应的后置数据处理计算图，其中，所述后置处理表中包括所述目标结构单元与所述后置数据处理计算图的映射关系，所述后置数据处理计算图用于对目标处理数据进行后置处理；

根据所述后置数据处理计算图，处理所述当前处理数据，获得所述数据处理结果。

4.根据权利要求 2-3 任一项所述的数据处理方法，所述根据所述当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果，包括：

根据所述前置数据处理计算图，处理所述目标数据，获得前置处理数据；

在根据所述前置处理数据确定不执行所述当前节点、且目标结构单元在后置处理表中的情况下，从所述后置处理表中确定所述目标结构单元对应的后置数据处理计算图，根据所述后置数据处理计算图，处理所述前置处理数据，获得所述数据处理结果，或者，

在根据所述前置处理数据确定不执行所述当前节点、且目标结构单元不在后置处理表的情况下，将所述前置处理数据确定为数据处理结果。

5.根据权利要求 1-4 任一项所述的数据处理方法，所述确定所述当前节点对应的数据处理计算图，包括：

根据目标映射表，确定所述当前节点对应的所述目标模型的目标结构单元；

在所述目标结构单元不在前置处理表中的情况下，根据所述当前节点处理所述目标数据，获得当前处理数据；

在目标结构单元在后置处理表中的情况下，从所述后置处理表中确定所述目标结构单元对应的后置数据处理计算图。

6.根据权利要求 5 所述的数据处理方法，所述根据所述当前节点和/或所述数据处

理计算图，处理所述目标数据，获得数据处理结果，包括：

根据所述后置数据处理计算图，处理所述当前处理数据，获得所述数据处理结果。

7.根据权利要求 1-6 任一项所述的数据处理方法，所述确定所述目标模型的目标计算图，包括：

获取所述目标模型，将所述目标模型进行数据处理的处理操作作为所述节点，将所述处理操作之间的关联关系作为边；

根据所述节点和所述边，构建所述目标计算图。

8.根据权利要求 7 所述的数据处理方法，所述根据所述节点和所述边，构建所述目标计算图，包括：

根据所述节点和所述边，构建初始计算图；

对所述初始计算图中的目标节点以及目标边进行冗余消除操作，获得所述目标模型的目标计算图，以使目标模型在目标处理器低使用率的状态下运行。

9.根据权利要求 2-8 任一项所述的数据处理方法，所述根据目标映射表，确定所述当前节点对应的目标模型的目标结构单元之前，还包括：

确定构建所述目标模型的结构单元，其中，所述结构单元为目标模型的网络层；

确定所述目标计算图中各节点对应的结构单元；

根据所述目标计算图中的节点以及所述目标模型的结构单元之间的映射关系，构建所述目标映射表。

10.根据权利要求 3 或 4 或 6 所述的数据处理方法，所述获得所述数据处理结果之后，还包括：

判断是否存在所述当前节点的下一节点，

若存在，将所述当前节点的下一节点确定为当前节点，并将所述数据处理结果作为所述目标数据，执行所述确定所述当前节点对应的数据处理计算图，根据所述当前节点和/或所述数据处理计算图，处理所述目标数据，获得数据处理结果的步骤，

若不存在，输出所述数据处理结果。

11.根据权利要求 1-10 任一项所述的数据处理方法，所述数据处理方法应用于支持机器学习模型计算的计算机硬件。

12.根据权利要求 11 所述的数据处理方法，所述确定目标模型的目标计算图以及输入所述目标模型的目标数据，包括：

根据所述支持机器学习模型计算的计算机硬件的硬件资源，确定目标模型的目标计算图以及输入所述目标模型的目标数据。

13.一种文本处理方法，包括：

确定目标模型的目标计算图以及输入所述目标模型的目标文本，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计算图；

从所述目标计算图中确定处理所述目标文本的当前节点，其中，所述当前节点为所述多个节点中的任意一个节点；

确定所述当前节点对应的文本处理计算图，根据所述当前节点和/或所述文本处理计算图，处理所述目标文本，获得文本处理结果。

14.一种图像处理方法，包括：

确定目标模型的目标计算图以及输入所述目标模型的目标图像，其中，所述目标模型为机器学习模型，所述目标计算图由多个节点以及连接各节点之间的边构建，且所述目标计算图为在所述目标模型运行时不能进行更新的计算图；

从所述目标计算图中确定处理所述目标图像的当前节点，其中，所述当前节点为

所述多个节点中的任意一个节点；

确定所述当前节点对应的图像处理计算图，根据所述当前节点和/或所述图像处理计算图，处理所述目标图像，获得图像处理结果。

15.一种计算设备，包括：

存储器和处理器；

所述存储器用于存储计算机程序/指令，所述处理器用于执行所述计算机程序/指令，该计算机程序/指令被处理器执行时实现权利要求 1 至 14 任意一项所述方法的步骤。

16.一种计算机可读存储介质，其存储有计算机程序/指令，该计算机程序/指令被处理器执行时实现权利要求 1 至 14 任意一项所述方法的步骤。

17.一种计算机程序产品，包括计算机程序/指令，该计算机程序/指令被处理器执行时实现权利要求 1 至 14 任意一项所述方法的步骤。