WORK ORDER TASK PROCESSING METHOD, COMPUTER SYSTEM AND STORAGE MEDIUM

工单任务的处理方法、计算机系统及存储介质

技术领域

本说明书一个或多个实施例涉及大模型领域，特别涉及一种工单任务的处理方法、计算机系统及存储介质。

背景技术

随着互联网技术的不断发展，逐渐涌现出了很多大型的互联网服务平台（例如支付平台，购物平台，社交平台以及各种综合性平台等），为用户提供了各种类型的平台服务。为了保证服务平台能够正常运营，为用户提供更好的服务，通常采用工单系统进行平台维护。例如，服务平台可以向用户提供输入接口，用户可以通过该输入接口输入使用服务平台时遇到的问题，从而生成用于解决平台问题的工单任务。目前来说，需要一种处理工单任务的方案。

发明内容

本说明书一个或多个实施例提供一种工单任务的处理方法、计算机系统及存储介质。

根据第一方面，提供一种工单任务的处理方法，包括至少一次执行以下目标操作；所述目标操作包括：获取工单任务对应的当前任务信息；基于所述当前任务信息，利用目标大模型以及目标图谱，确定至少一个任务处理接口；其中，所述目标图谱中包括多个表示备选任务处理接口的第一节点；调用所述任务处理接口，处理所述工单任务，得到本次的结果；若基于所述本次的结果确定未处理完成所述工单任务，继续执行下一次目标操作；若基于所述本次的结果确定处理完成所述工单任务，基于所述本次的结果输出目标结果。

根据第二方面，提供一种计算机系统，包括存储器、处理器及存储在所述存储器上并可在所述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程序时至少一次执行以下目标操作，通过所述目标操作实现如下步骤：获取工单任务对应的当前任务信息；基于所述当前任务信息，利用目标大模型以及目标图谱，确定至少一个任务处理接口；其中，所述目标图谱中包括多个表示备选任务处理接口的第一节点；调用所述任务处理接口，处理所述工单任务，得到本次的结果；若基于所述本次的结果确定未处理完成所述工单任务，继续执行下一次目标操作；若基于所述本次的结果确定处理完成所述工单任务，基于所述本次的

结果输出目标结果。

根据第三方面，提供一种计算机可读存储介质，所述存储介质存储有计算机程序，所述计算机程序被处理器执行时实现上述第一方面中任一项所述的方法。

本说明书的实施例提供的技术方案可以包括以下有益效果：

本说明书的实施例提供的工单任务的处理方法，至少一次执行以下目标操作：获取工单任务对应的当前任务信息，基于当前任务信息，利用目标大模型以及目标图谱，确定至少一个任务处理接口，其中，目标图谱包括多个表示备选任务处理接口的第一节点，调用任务处理接口，处理工单任务，得到本次的结果。若基于本次的结果确定未处理完成工单任务，继续执行下一次目标操作，若基于本次的结果确定处理完成工单任务，基于本次的结果输出目标结果。从而能够高效且准确地处理完成工单任务，降低了出错率。

应当理解的是，以上的一般描述和后文的细节描述仅是示例性和解释性的，并不能限制本申请。

附图说明

为了更清楚地说明本说明书实施例的技术方案，下面将对实施例描述中所需要使用的附图作简单地介绍，显而易见地，下面描述中的附图仅仅是本说明书中记载的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动性的前提下，还可以根据这些附图获得其他的附图。

图1是本说明书根据一示例性实施例示出的一种工单任务的处理场景示意图架构图；

图2是本说明书根据一示例性实施例示出的一种工单任务的处理方法流程图；

图3是本说明书根据一示例性实施例示出的一种图谱示意图；

图4是本说明书根据一示例性实施例示出的一种计算机系统示意图。

具体实施方式

为了使本技术领域的人员更好地理解本说明书中的技术方案，下面将结合本说明书实施例中的附图，对本说明书实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例仅仅是本说明书一部分实施例，而不是全部的实施例。基于本说明书中的实施例，本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例，都应当属于本说明书保护的范围。

随着互联网技术的不断发展，逐渐涌现出了很多大型的互联网服务平台（例如支付平台，购物平台，社交平台以及各种综合性平台等），为用户提供了各种类型的平台服务。为了保

证服务平台能够正常运营，为用户提供更好的服务，通常采用工单系统进行平台维护。例如，服务平台可以向用户提供输入接口，用户可以通过该输入接口输入使用服务平台时遇到的问题，从而生成用于解决平台问题的工单任务。

在相关技术中，一般由运维人员人工处理工单任务，解决服务平台的问题。也可以由平台调用预先编写的规则，解决服务平台的问题。然而，通过运维人员人工处理工单任务，需要消耗人力的资源，并且，由人工处理工单任务的效率一般也比较低，且容易出错。而通过调用预先编写的规则处理工单任务的问题在于，难以编写全面准确的规则集，规则集的维护成本较高，且规则集可能无法覆盖所有的情况，导致调用结果的准确性和完整性受到限制。

本说明书实施例提供了一种工单任务的处理方案，至少一次执行以下目标操作：获取工单任务对应的当前任务信息，基于当前任务信息，利用目标大模型以及目标图谱，确定至少一个任务处理接口，其中，目标图谱包括多个表示备选任务处理接口的第一节点，调用任务处理接口，处理工单任务，得到本次的结果。若基于本次的结果确定未处理完成工单任务，继续执行下一次目标操作，若基于本次的结果确定处理完成工单任务，基于本次的结果输出目标结果。从而能够高效且准确地处理完成工单任务，降低了出错率。

图1是根据一示例性实施例示出的一种工单任务的处理场景示意图架构图。

如图1所示，具体的应用场景可以为，用户通过购物平台购买商品后，进行退款，发现退款后银行到账金额和实付款金额不符。用户可以通过购物平台的个人账户页面，找到问题输入接口，通过问题输入接口输入使用自然语言描述的问题内容Q。或者，也可以由店铺客服，通过购物平台的店铺管理页面中的问题输入接口，输入使用自然语言描述的问题内容Q。例如，问题内容Q可以是“交易单号为123456789，退款后银行到账金额和实付款金额不符，要求查明原因”。在输入问题内容并提交之后，购物平台生成一个新的工单任务W。

首先，可以获取工单任务W的当前任务信息A1，该当前任务信息A1可以包括问题内容Q，大模型引导词以及对任务处理接口的描述。其中，大模型引导词可以是用于指示大模型扮演的角色，或者指示大模型需要做的事情的文本内容。大模型引导词可以是用户输入问题内容Q时一并输入的内容，也可以是问题内容输入模版中预先设置好的。对任务处理接口的描述，可以是对每个任务处理接口的用途的描述等。

接着，可以进行第一轮操作，具体地，可以将当前任务信息A1输入至大模型M1，得到大模型M1输出的结果，该结果可以包括描述任务意图的任务目的信息B1。其中，在本应用场景中，描述任务意图的任务目的信息B1可以是“查询该交易的银行支付历史”。可以将当前任务信息A1和任务目的信息B1输入至大模型M2，得到大模型M2输出的结果。可以从第二大模型输出的结果中获取任务处理接口的输入参数类型和输入参数值。若该结果中未包括该输入

参数值，可以基于当前任务信息A1和任务目的信息B1，利用预设的正则表达式，获取该输入参数值。其中，任务处理接口的输入参数类型可以是使用任务处理接口时，需要输入任务处理接口的参数的类型，可以将该输入参数的类型作为键k，则该输入参数的值为该键k的值v。在本应用场景中，输入参数类型可以是“交易单号”，该输入参数值可以是“123456789”。

然后，可以获取预先构建的图谱T，图谱T中可以包括多个节点，一部分节点表示任务处理接口，一部分节点表示参数类型。节点之间具有连接边，连接边为有向边，连接边表示任务处理接口与参数类型之间的输入输出关系。例如，节点1表示任务处理接口a，节点2表示参数类型b，节点3表示参数类型c。节点1和节点2之间具有连接边21，连接边21通过箭头指向节点1，连接边21表示任务处理接口a的输入参数类型可以是参数类型b。节点2和节点3之间具有连接边23，连接边23通过箭头指向节点3，连接边23表示任务处理接口a的输出参数类型可以是参数类型c。可以预先收集多个任务处理接口及其各自的输入、输出参数类型，然后，基于多个任务处理接口及其各自的输入、输出参数类型，构建相应的图谱T，并将图谱T进行存储。在确定任务处理接口时，可以根据任务处理接口的输入参数类型、输入参数值和任务目的信息B1，从图谱T中寻找从输入参数达到输出参数的目标路径。

可以确定图谱T中的目标路径包括的节点所表示的至少一个任务处理接口，并基于输入参数值，依次调用该至少一个任务处理接口，从而得到本次的结果。本应用场景中，任务目的信息B1是“查询该交易的银行支付历史”，任务处理接口可以包括接口K1和接口K2，接口K1的输入参数类型可以是“交易单号”，输入参数值可以是“123456789”，接口K1的输出参数类型可以是“银行支付号”，接口K1的输出参数值可以是“kajdajkd”。接口K2的输入参数类型可以是“银行支付号”，输入参数值可以是“kajdajkd”，接口K2的输出参数类型可以是“支付金额”和“退款金额”，接口K2的输出参数值可以是“n”和“n”，输出参数值“n”和“n”即为本次的结果。

接着，可以将本次的结果和当前任务信息A1输入至大模型M3，大模型M3可以根据本次的结果和当前任务信息A1，输出用于指示当前是否处理完成工单任务的结果。在本应用场景中，该结果可以是“未处理完成工单任务”，因此，还需要执行第二轮操作。具体来说，可以将第一轮操作的输出参数值“n”和“n”，以及问题内容Q作为第二轮操作的当前任务信息A2。再将当前任务信息A2输入至大模型M1中，得到大模型M1输出结果，该结果可以包括描述任务意图的任务目的信息B2。在本应用场景中，描述任务意图的任务目的信息B2可以是“查询该交易实付款金额以及明细”。

可以将当前任务信息A2和任务目的信息B2输入至大模型M2，得到大模型M2输出的结果。根据第二大模型输出的结果，确定任务处理接口的输入参数类型和输入参数值。并根据任务

处理接口的输入参数类型、输入参数值和任务目的信息B2，从图谱T中寻找目标路径。确定图谱T中的目标路径包括的节点所表示的至少一个任务处理接口，并基于输入参数值，依次调用该至少一个任务处理接口，从而得到第二轮操作的结果。在本应用场景中，输入参数类型可以是“交易单号”，该输入参数值可以是“123456789”。任务目的信息B2是“查询该交易实付款金额以及明细”，任务处理接口可以包括接口K3，接口K3的输入参数类型可以是“交易单号”，输入参数值可以是“123456789”，接口K3的输出参数类型可以是“交易实付款金额及明细”，接口K3的输出参数值可以是“现金n，红包y”，即为第二轮的结果。

最后，将第二轮的结果和当前任务信息A2输入至大模型M3，大模型M3可以根据第二轮的结果和当前任务信息A2，输出用于指示当前是否处理完成工单任务的结果。在本应用场景中，该结果可以是“处理完成工单任务”，因此，无需执行第三轮操作。可以通过预设的规则集或者基于另外的大模型，对问题内容Q、第一轮的结果“n”和“n”以及第二轮的结果“现金n，红包y”进行分析，得出结论，并将结论返回给用户。结论可以是采用自然语言的文本内容，例如，可以是“实付款金额中包括现金n和红包y，现金部分n由银行账户支付，现已全额退回银行账户，红包部分y返回交易账户”。

下面将结合具体的实施例对本说明书提供的方案进行详细描述。

如图2所示，图2是根据一示例性实施例示出的一种工单任务的处理方法的流程图，可以由服务平台执行，该服务平台可以实现为任何具有计算、处理能力的设备、服务器或设备集群。该方法包括以下步骤：

在步骤201中，获取工单任务对应的当前任务信息。

在本实施例中，所涉及的服务平台可以包括但不限于购物平台、支付平台、社交平台以及其它各种综合性平台等。用户通过服务平台进行交易时，或者进行交易之后，如果遇到问题，可以通过该服务平台提供的问题输入接口，输入使用自然语言描述的问题内容，以查询或解决问题。在输入问题内容并提交之后，服务平台会生成一个新的工单任务。

在处理一个工单任务时，需要执行至少一次目标操作，若首次执行目标操作，可以获取用户针对该工单任务输入的任务内容，作为当前任务信息。若非首次执行目标操作，可以获取用户针对该工单任务输入的任务内容和上次执行目标操作得到的结果，作为当前任务信息。

在步骤202中，基于当前任务信息，利用目标大模型以及目标图谱，确定至少一个任务处理接口。

具体来说，首先，可以利用目标大模型，基于当前任务信息，确定参数信息，参数信息可以包括任务处理接口的任务目的信息、输入参数类型和输入参数值。其中，目标大模型可以是通用的大语言模型，本实施例对目标大模型的具体类别方面不限定。任务处理接口可以

是用于处理任务的接口工具，任务处理接口对应有输入参数和输出参数，可以调用任务处理接口对输入参数的值进行处理，得到输出参数的值。

在一种实现方式中，可以利用一个目标大模型，基于当前任务信息，确定参数信息。在另一种实现方式中，还可以利用多个目标大模型，基于当前任务信息，确定参数信息。例如，目标大模型可以包括第一大模型和第二大模型，可以将当前任务信息输入至第一大模型，得到描述任务意图的任务目的信息，再将当前任务信息和任务目的信息输入至第二大模型，基于第二大模型输出的结果，确定输入参数类型和输入参数值。例如，可以从第二大模型输出的结果中获取输入参数类型和所述输入参数值。若确定第二大模型输出的结果中未包括输入参数值，则可以进一步基于当前任务信息、输入参数类型和任务目的信息，利用预设的正则表达式，获取输入参数值。

接着，可以基于参数信息以及目标图谱，确定至少一个任务处理接口。其中，目标图谱可以是基于收集到的多个任务处理接口，以及各个任务处理接口的输入和输出参数的类型预先构建的图谱。目标图谱中包括多个节点，一部分节点表示任务处理接口，一部分节点表示参数类型。可以将表示任务处理接口的节点作为第一节点，将表示参数类型的节点作为第二节点。节点之间具有连接边，连接边为有向边，连接边表示任务处理接口与参数类型之间的输入输出关系。

如图3所示，节点101~105均为表示任务处理接口的第一节点，节点106~111均为表示参数类型的第二节点。其中，节点106为节点101和节点102的输入参数类型，节点104和节点108为节点101的输出参数类型，节点107为节点102的输出参数类型，节点107为节点103的输入参数类型，节点104为节点103的输出参数类型……。

具体地，可以基于上述参数信息，从目标图谱中寻找从输入参数达到输出参数的目标路径，基于目标路径，确定至少一个任务处理接口。例如，可以采用Dijkstra算法，根据上述参数信息，从目标图谱中寻找从输入参数达到输出参数的最短路径，作为目标路径。然后，将目标路径中包括的任务处理接口作为用于处理工单任务的任务处理接口。

需要说明的是，若需要添加新的备选任务处理接口，可以对目标图谱进行更新，目标图谱可以通过以下方式更新：先获取待添加的备选任务处理接口，确定该待添加的备选任务处理接口对应的输入参数类型和输出参数类型。可以基于待添加的备选任务处理接口对应的输入参数类型和输出参数类型，遍历目标图谱中的节点，以确定与待添加的备选任务处理接口关联的第二节点。其中，与待添加的备选任务处理接口关联的第二节点所表示的参数类型，与待添加的备选任务处理接口对应的输入参数类型或输出参数类型相同。针对待添加的备选任务处理接口，在目标图谱中生成新的第一节点，并在新的第一节点与关联的第二节点之间

生成连接边。

在步骤203中，调用任务处理接口，处理工单任务，得到本次的结果。

在本实施例中，若确定有一个任务处理接口，可以调用该任务处理接口，处理上述参数信息中包括的输入参数值，即将上述参数信息中包括的输入参数值，直接输入至该任务处理接口中，得到该任务处理接口输出的输出参数值，作为本次的结果。若确定有多个任务处理接口，可以按照目标路径中包括的任务处理接口的顺序，依次调用该多个任务处理接口，即将输入参数值输入至首个任务处理接口中，并将第一个任务处理接口的输出参数值输入至下一个任务处理接口中，以此类推，最终获取最后一个任务处理接口输出的输出参数值，作为本次的结果。

在步骤204中，若基于本次的结果确定未处理完成工单任务，继续执行下一次目标操作。若基于本次的结果确定处理完成工单任务，基于本次的结果输出目标结果。

在本实施例中，在得到本次的结果之后，可以基于本次的结果确定是否处理完成工单任务。例如，可以基于预设规则，确定是否处理完成工单任务。也可以将本次的结果和当前任务信息输入至第三大模型中，基于第三大模型输出的结果，确定当前是否处理完成工单任务，第三大模型可以直接输出用于指示是否处理完成工单任务的信息。

若基于本次的结果确定未处理完成工单任务，则继续执行下一次目标操作。即将本次的结果和任务内容，作为下一次目标操作的当前任务信息，基于下一次目标操作的当前任务信息，利用目标大模型以及目标图谱，确定任务处理接口。若基于本次的结果确定已处理完成工单任务，则基于本次的结果输出目标结果。可以基于预设的规则，根据本次的结果确定目标结果，也可以采用大模型，根据本次的结果确定目标结果。可以理解，可以采用任意合理的方式确定目标结果，本实施例对确定目标结果的具体方式方面方面不限定。

本说明书实施例提供的工单任务的处理方法，至少一次执行以下目标操作：获取工单任务对应的当前任务信息，基于当前任务信息，利用目标大模型以及目标图谱，确定至少一个任务处理接口，其中，目标图谱包括多个表示备选任务处理接口的第一节点，调用任务处理接口，处理工单任务，得到本次的结果。若基于本次的结果确定未处理完成工单任务，继续执行下一次目标操作，若基于本次的结果确定处理完成工单任务，基于本次的结果输出目标结果。从而能够高效且准确地处理完成工单任务，降低了出错率。

应当注意，尽管在上述实施例中，以特定顺序描述了本说明书实施例的方法的操作，但是，这并非要求或者暗示必须按照该特定顺序来执行这些操作，或是必须执行全部所示的操作才能实现期望的结果。相反，流程图中描绘的步骤可以改变执行顺序。附加地或备选地，可以省略某些步骤，将多个步骤合并为一个步骤执行，和/或将一个步骤分解为多个步骤执行。

The specification also provides a computer-readable non-transitory storage medium, wherein the storage medium stores a computer program, which, when executed by a processor, can be used to perform one or more steps of one or more methods described or illustrated herein or provides functionality described or illustrated herein. Herein, the computer-readable non-transitory storage medium or media may include one or more semiconductor-based or other integrated circuits (ICs) (such, as for example, field-programmable gate arrays (FPGAs) or application-specific ICs (ASICs)), hard disk drives (HDDs), hybrid hard drives (HHDs), optical discs, optical disc drives (ODDs), magneto-optical discs, magneto-optical drives, floppy diskettes, floppy disk drives (FDDs), magnetic tapes, solid-state drives (SSDs), RAM-drives, SECURE DIGITAL cards or drives, any other suitable computer-readable non-transitory storage media, or any suitable combination of two or more of these, where appropriate. A computer-readable non-transitory storage medium may be volatile, non-volatile, or a combination of volatile and non-volatile, where appropriate.

The specification also provides a computer system. FIG. 4 illustrates an example computer system 1100. In particular embodiments, one or more computer systems 1100 perform one or more steps of one or more methods described or illustrated herein. In particular embodiments, one or more computer systems 1100 provide functionality described or illustrated herein. In particular embodiments, software running on one or more computer systems 1100 performs one or more steps of one or more methods described or illustrated herein or provides functionality described or illustrated herein. Particular embodiments include one or more portions of one or more computer systems 1100. Herein, reference to a computer system may encompass a computing device, and vice versa, where appropriate. Moreover, reference to a computer system may encompass one or more computer systems, where appropriate.

This disclosure contemplates any suitable number of computer systems 1100. This disclosure contemplates computer system 1100 taking any suitable physical form. As example and not by way of limitation, computer system 1100 may be an embedded computer system, a system-on-chip (SOC), a single-board computer system (SBC) (such as, for example, a computer-on-module (COM) or system-on-module (SOM)), a desktop computer system, a laptop or notebook computer system, an interactive kiosk, a mainframe, a mesh of computer systems, a mobile telephone, a personal digital assistant (PDA), a server, a tablet computer system, or a combination of two or more of these. Where appropriate, computer system 1100 may include one or more computer systems 1100; be unitary or distributed; span multiple locations; span multiple machines; span multiple data centers; or reside in a cloud, which may include one or more cloud components in one or more networks. Where appropriate, one or more computer systems 1100 may perform without substantial spatial or temporal limitation one or more steps of one or more methods described or illustrated herein. As an example and not by way of limitation, one or more computer systems 1100 may perform in real time or in batch mode one or more steps of one or more methods described or illustrated herein.

One or more computer systems 1100 may perform at different times or at different locations one or more steps of one or more methods described or illustrated herein, where appropriate.

In particular embodiments, computer system 1100 includes a processor 1102, memory 1104, storage 1106, an input/output (I/O) interface 1108, a communication interface 1110, and a bus 1112. Although this disclosure describes and illustrates a particular computer system having a particular number of particular components in a particular arrangement, this disclosure contemplates any suitable computer system having any suitable number of any suitable components in any suitable arrangement.

In particular embodiments, processor 1102 includes hardware for executing instructions, such as those making up a computer program. As an example and not by way of limitation, to execute instructions, processor 1102 may retrieve (or fetch) the instructions from an internal register, an internal cache, memory 1104, or storage 1106; decode and execute them; and then write one or more results to an internal register, an internal cache, memory 1104, or storage 1106. In particular embodiments, processor 1102 may include one or more internal caches for data, instructions, or addresses. This disclosure contemplates processor 1102 including any suitable number of any suitable internal caches, where appropriate. As an example and not by way of limitation, processor 1102 may include one or more instruction caches, one or more data caches, and one or more translation lookaside buffers (TLBs). Instructions in the instruction caches may be copies of instructions in memory 1104 or storage 1106, and the instruction caches may speed up retrieval of those instructions by processor 1102. Data in the data caches may be copies of data of 80 in memory 1104 or storage 1106 for instructions executing at processor 1102 to operate on; the results of previous instructions executed at processor 1102 for access by subsequent instructions executing at processor 1102 or for writing to memory 1104 or storage 1106; or other suitable data. The data caches may speed up read or write operations by processor 1102. The TLBs may speed up virtual-address translation for processor 1102. In particular embodiments, processor 1102 may include one or more internal registers for data, instructions, or addresses. This disclosure contemplates processor 1102 including any suitable number of any suitable internal registers, where appropriate. Where appropriate, processor 1102 may include one or more arithmetic logic units (ALUs); be a multi-core processor; or include one or more processors 1102. Although this disclosure describes and illustrates a particular processor, this disclosure contemplates any suitable processor.

In particular embodiments, memory 1104 includes main memory for storing instructions for processor 1102 to execute or data for processor 1102 to operate on. As an example and not by way of limitation, computer system 1100 may load instructions from storage 1106 or another source (such as, for example, another computer system 1100) to memory 1104. Processor 1102 may then load the instructions from memory 1104 to an internal register or internal cache. To execute the

instructions, processor 1102 may retrieve the instructions from the internal register or internal cache and decode them. During or after execution of the instructions, processor 1102 may write one or more results (which may be intermediate or final results) to the internal register or internal cache. Processor 1102 may then write one or more of those results to memory 1104. In particular embodiments, processor 1102 executes only instructions in one or more internal registers or internal caches or in memory 1104 (as opposed to storage 1106 or elsewhere) and operates only on data in one or more internal registers or internal caches or in memory 1104 (as opposed to storage 1106 or elsewhere). One or more memory buses (which may each include an address bus and a data bus) may couple processor 1102 to memory 1104. Bus 1112 may include one or more memory buses, as described below. In particular embodiments, one or more memory management units (MMUs) reside between processor 1102 and memory 1104 and facilitate accesses to memory 1104 requested by processor 1102. In particular embodiments, memory 1104 includes random access memory (RAM). This RAM may be volatile memory, where appropriate. Where appropriate, this RAM may be dynamic RAM (DRAM) or static RAM (SRAM). Moreover, where appropriate, this RAM may be single-ported or multi-ported RAM. This disclosure contemplates any suitable RAM. Memory 1104 may include one or more memories 1104, where appropriate. Although this disclosure describes and illustrates particular memory, this disclosure contemplates any suitable memory.

In particular embodiments, storage 1106 includes mass storage for data or instructions. As an example and not by way of limitation, storage 1106 may include a hard disk drive (HDD), a floppy disk drive, flash memory, an optical disc, a magneto-optical disc, magnetic tape, or a Universal Serial Bus (USB) drive or a combination of two or more of these. Storage 1106 may include removable or non-removable (or fixed) media, where appropriate. Storage 1106 may be internal or external to computer system 1100, where appropriate. In particular embodiments, storage 1106 is non-volatile, solid-state memory. In particular embodiments, storage 1106 includes read-only memory (ROM). Where appropriate, this ROM may be mask-programmed ROM, programmable ROM (PROM), erasable PROM (EPROM), electrically erasable PROM (EEPROM), electrically alterable ROM (EAROM), or flash memory or a combination of two or more of these. This disclosure contemplates mass storage 1106 taking any suitable physical form. Storage 1106 may include one or more storage control units facilitating communication between processor 1102 and storage 1106, where appropriate. Where appropriate, storage 1106 may include one or more storages 1106. Although this disclosure describes and illustrates particular storage, this disclosure contemplates any suitable storage.

In particular embodiments, I/O interface 1108 includes hardware, software, or both, providing one or more interfaces for communication between computer system 1100 and one or more I/O devices. Computer system 1100 may include one or more of these I/O devices, where appropriate. One or more of these I/O devices may enable communication between a person and computer

system 1100. As an example and not by way of limitation, an I/O device may include a keyboard, keypad, microphone, monitor, mouse, printer, scanner, speaker, still camera, stylus, tablet, touch screen, trackball, video camera, another suitable I/O device or a combination of two or more of these. An I/O device may include one or more sensors. This disclosure contemplates any suitable I/O devices and any suitable I/O interfaces 1108 for them. Where appropriate, I/O interface 1108 may include one or more device or software drivers enabling processor 1102 to drive one or more of these I/O devices. I/O interface 1108 may include one or more I/O interfaces 1108, where appropriate. Although this disclosure describes and illustrates a particular I/O interface, this disclosure contemplates any suitable I/O interface.

In particular embodiments, communication interface 1110 includes hardware, software, or both providing one or more interfaces for communication (such as, for example, packet-based communication) between computer system 1100 and one or more other computer systems 1100 or one or more networks. As an example and not by way of limitation, communication interface 1110 may include a network interface controller (NIC) or network adapter for communicating with an Ethernet or other wire-based network or a wireless NIC (WNIC) or wireless adapter for communicating with a wireless network, such as a WI-FI network. This disclosure contemplates any suitable network and any suitable communication interface 1110 for it. As an example and not by way of limitation, computer system 1100 may communicate with an ad hoc network, a personal area network (PAN), a local area network (LAN), a wide area network (WAN), a metropolitan area network (MAN), or one or more portions of the Internet or a combination of two or more of these. One or more portions of one or more of these networks may be wired or wireless. As an example, computer system 1100 may communicate with a wireless PAN (WPAN) (such as, for example, a BLUETOOTH WPAN), a WI-FI network, a WI-MAX network, a cellular telephone network (such as, for example, a Global System for Mobile Communications (GSM) network), or other suitable wireless network or a combination of two or more of these. Computer system 1100 may include any suitable communication interface 1110 for any of these networks, where appropriate. Communication interface 1110 may include one or more communication interfaces 1110, where appropriate. Although this disclosure describes and illustrates a particular communication interface, this disclosure contemplates any suitable communication interface.

In particular embodiments, bus 1112 includes hardware, software, or both coupling components of computer system 1100 to each other. As an example and not by way of limitation, bus 1112 may include an Accelerated Graphics Port (AGP) or other graphics bus, an Enhanced Industry Standard Architecture (EISA) bus, a front-side bus (FSB), a HYPERTRANSPORT (HT) interconnect, an Industry Standard Architecture (ISA) bus, an INFINIBAND interconnect, a low-pin-count (LPC) bus, a memory bus, a Micro Channel Architecture (MCA) bus, a Peripheral Component Interconnect (PCI) bus, a PCI-Express (PCIe) bus, a serial advanced technology

attachment (SATA) bus, a Video Electronics Standards Association local (VLB) bus, or another suitable bus or a combination of two or more of these. Bus 1112 may include one or more buses 1112, where appropriate. Although this disclosure describes and illustrates a particular bus, this disclosure contemplates any suitable bus or interconnect.

还需要说明的是，术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含，从而使得包括一系列要素的过程、方法、商品或者设备不仅包括那些要素，而且还包括没有明确列出的其他要素，或者是还包括为这种过程、方法、商品或者设备所固有的要素。在没有更多限制的情况下，由语句“包括一个……”限定的要素，并不排除在包括所述要素的过程、方法、商品或者设备中还存在另外的相同要素。

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，对于系统实施例而言，由于其基本相似于方法实施例，所以描述的比较简单，相关之处参见方法实施例的部分说明即可。

以上所述仅为本说明书的实施例而已，并不用于限制本说明书。对于本领域技术人员来说，本说明书可以有各种更改和变化。凡在本说明书的精神和原理之内所作的任何修改、等同替换、改进等，均应包含在本申请的权利要求范围之内。

权 利 要 求 书

1．一种工单任务的处理方法，包括至少一次执行以下目标操作；所述目标操作包括：

获取工单任务对应的当前任务信息；

基于所述当前任务信息，利用目标大模型以及目标图谱，确定至少一个任务处理接口；其中，所述目标图谱中包括多个表示备选任务处理接口的第一节点；

调用所述任务处理接口，处理所述工单任务，得到本次的结果；

若基于所述本次的结果确定未处理完成所述工单任务，继续执行下一次目标操作；若基于所述本次的结果确定处理完成所述工单任务，基于所述本次的结果输出目标结果。

2．根据权利要求1所述的方法，其中，所述获取工单任务对应的当前任务信息，包括：

若首次执行目标操作，获取针对所述工单任务输入的任务内容，作为所述当前任务信息；

若非首次执行所述目标操作，获取所述任务内容和上次执行所述目标操作得到的结果，作为所述当前任务信息。

3．根据权利要求1所述的方法，其中，所述基于所述当前任务信息，利用目标大模型以及目标图谱，确定至少一个任务处理接口，包括：

利用所述目标大模型，基于所述当前任务信息，确定参数信息；所述参数信息包括任务目的信息、任务处理接口的输入参数类型和输入参数值；

基于所述参数信息以及所述目标图谱，确定所述至少一个任务处理接口。

4．根据权利要求3所述的方法，其中，所述目标大模型包括第一大模型和第二大模型；其中，所述利用所述目标大模型，基于所述当前任务信息，确定参数信息，包括：

将所述当前任务信息输入至所述第一大模型，得到描述任务意图的任务目的信息；

将所述当前任务信息和所述任务目的信息输入至所述第二大模型，基于所述第二大模型输出的结果，确定所述输入参数类型和所述输入参数值。

5．根据权利要求4所述的方法，其中，所述基于所述第二大模型输出的结果，确定所述输入参数类型和所述输入参数值，包括：

从所述第二大模型输出的结果中获取所述输入参数类型和所述输入参数值；

若所述第二大模型输出的结果中未包括所述输入参数值，基于所述当前任务信息、所述输入参数类型和所述任务目的信息，利用预设的正则表达式，获取所述输入参数值。

6．根据权利要求3所述的方法，其中，所述基于所述参数信息以及所述目标图谱，确定所述至少一个任务处理接口，包括：

基于所述参数信息，从所述目标图谱中寻找从输入参数达到输出参数的目标路径；

基于所述目标路径，确定所述至少一个任务处理接口。

7．根据权利要求1所述的方法，其中，在得到本次的结果之后，所述方法还包括：

将所述本次的结果和所述当前任务信息输入至第三大模型；

基于所述第三大模型输出的结果，确定当前是否处理完成所述工单任务。

8. 根据权利要求 1 所述的方法，其中，所述目标图谱中还包括多个表示参数类型的第二节点；所述目标图谱中的节点之间具有连接边，所述连接边为有向边，所述连接边表示备选任务处理接口与参数类型之间的输入输出关系。

9. 根据权利要求 8 所述的方法，其中，所述目标图谱通过以下方式更新：

获取待添加的备选任务处理接口；

确定所述待添加的备选任务处理接口对应的输入参数类型和输出参数类型；

基于所述待添加的备选任务处理接口对应的输入参数类型和输出参数类型，遍历所述目标图谱中的节点，确定与所述待添加的备选任务处理接口关联的第二节点；

针对所述待添加的备选任务处理接口，在所述目标图谱中生成新的第一节点，并在所述新的第一节点与所述关联的第二节点之间生成连接边。

10、一种计算机系统，包括存储器、处理器及存储在所述存储器上并可在所述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程序时至少一次执行以下目标操作，通过所述目标操作实现如下步骤：

获取工单任务对应的当前任务信息；

基于所述当前任务信息，利用目标大模型以及目标图谱，确定至少一个任务处理接口；其中，所述目标图谱中包括多个表示备选任务处理接口的第一节点；

调用所述任务处理接口，处理所述工单任务，得到本次的结果；

若基于所述本次的结果确定未处理完成所述工单任务，继续执行下一次目标操作；若基于所述本次的结果确定处理完成所述工单任务，基于所述本次的结果输出目标结果。

11. 根据权利要求 10 所述的计算机系统，其中，所述获取工单任务对应的当前任务信息，包括：

若首次执行目标操作，获取针对所述工单任务输入的任务内容，作为所述当前任务信息；

若非首次执行所述目标操作，获取所述任务内容和上次执行所述目标操作得到的结果，作为所述当前任务信息。

12. 根据权利要求 10 所述的计算机系统，其中，所述基于所述当前任务信息，利用目标大模型以及目标图谱，确定至少一个任务处理接口，包括：

利用所述目标大模型，基于所述当前任务信息，确定参数信息；所述参数信息包括任务目的信息、任务处理接口的输入参数类型和输入参数值；

基于所述参数信息以及所述目标图谱，确定所述至少一个任务处理接口。

13．根据权利要求 12 所述的计算机系统，其中，所述目标大模型包括第一大模型和第二大模型；其中，所述利用所述目标大模型，基于所述当前任务信息，确定参数信息，包括：

将所述当前任务信息输入至所述第一大模型，得到描述任务意图的任务目的信息；

将所述当前任务信息和所述任务目的信息输入至所述第二大模型，基于所述第二大模型输出的结果，确定所述输入参数类型和所述输入参数值。

14．根据权利要求 13 所述的计算机系统，其中，所述基于所述第二大模型输出的结果，确定所述输入参数类型和所述输入参数值，包括：

从所述第二大模型输出的结果中获取所述输入参数类型和所述输入参数值；

若所述第二大模型输出的结果中未包括所述输入参数值，基于所述当前任务信息、所述输入参数类型和所述任务目的信息，利用预设的正则表达式，获取所述输入参数值。

15．根据权利要求 12 所述的计算机系统，其中，所述基于所述参数信息以及所述目标图谱，确定所述至少一个任务处理接口，包括：

基于所述参数信息，从所述目标图谱中寻找从输入参数达到输出参数的目标路径；

基于所述目标路径，确定所述至少一个任务处理接口。

16．根据权利要求 10 所述的计算机系统，其中，在得到本次的结果之后，所述处理器执行所述程序时还实现如下步骤：

将所述本次的结果和所述当前任务信息输入至第三大模型；

基于所述第三大模型输出的结果，确定当前是否处理完成所述工单任务。

17．根据权利要求 10 所述的计算机系统，其中，所述目标图谱中还包括多个表示参数类型的第二节点；所述目标图谱中的节点之间具有连接边，所述连接边为有向边，所述连接边表示备选任务处理接口与参数类型之间的输入输出关系。

18．根据权利要求 17 所述的计算机系统，其中，所述目标图谱通过以下方式更新：

获取待添加的备选任务处理接口；

确定所述待添加的备选任务处理接口对应的输入参数类型和输出参数类型；

基于所述待添加的备选任务处理接口对应的输入参数类型和输出参数类型，遍历所述目标图谱中的节点，确定与所述待添加的备选任务处理接口关联的第二节点；

针对所述待添加的备选任务处理接口，在所述目标图谱中生成新的第一节点，并在所述新的第一节点与所述关联的第二节点之间生成连接边。

19．一种计算机可读存储介质，其上存储有计算机程序，当所述计算机程序在计算机中执行时，令所述计算机执行权利要求 1-9 中任一项所述的方法。