

(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国际局

(43) 国际公布日
2021年7月15日 (15.07.2021)



(10) 国际公布号
WO 2021/139358 A1

(51) 国际专利分类号:
G06F 9/445 (2018.01) *G06F 9/48* (2006.01)

(21) 国际申请号: PCT/CN2020/125075

(22) 国际申请日: 2020年10月30日 (30.10.2020)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
202010755915.9 2020年7月30日 (30.07.2020) CN

(71) 申请人: 平安科技(深圳)有限公司(PING AN TECHNOLOGY (SHENZHEN) CO., LTD.) [CN/CN]; 中国广东省深圳市福田区福田街道福安社区益田路5033号平安金融中心23楼, Guangdong 518033 (CN)。

(72) 发明人: 沈冬明(SHEN, Dongming); 中国广东省深圳市福田区福田街道福安社区益田路5033号平安金融中心23楼, Guangdong 518033 (CN)。夏学海(XIA, Xuehai); 中国广东省深圳市福田区福田街道福安社区益田路5033号平安金融中心23楼, Guangdong 518033 (CN)。张国辉(ZHANG, Guohui); 中国广东省深圳市福田区福田街道福安社区益田路5033号平安金融中心23楼, Guangdong 518033 (CN)。宋晨(SONG, Chen); 中国广东省深圳市福田区福田街道福安社区益田路5033号平安金融中心23楼, Guangdong 518033 (CN)。

(74) 代理人: 北京鸿元知识产权代理有限公司(GRANDER IP LAW FIRM); 中国北京市朝阳区光华路7号汉威大厦东区18A6室, Beijing 100004 (CN)。

(54) Title: SERVICE METHOD AND APPARATUS BASED ON MICRO-KERNEL MODEL PLUG-IN MODE, DEVICE, AND STORAGE MEDIUM

(54) 发明名称: 基于微核模型插件模式的服务方法、装置、设备及存储介质

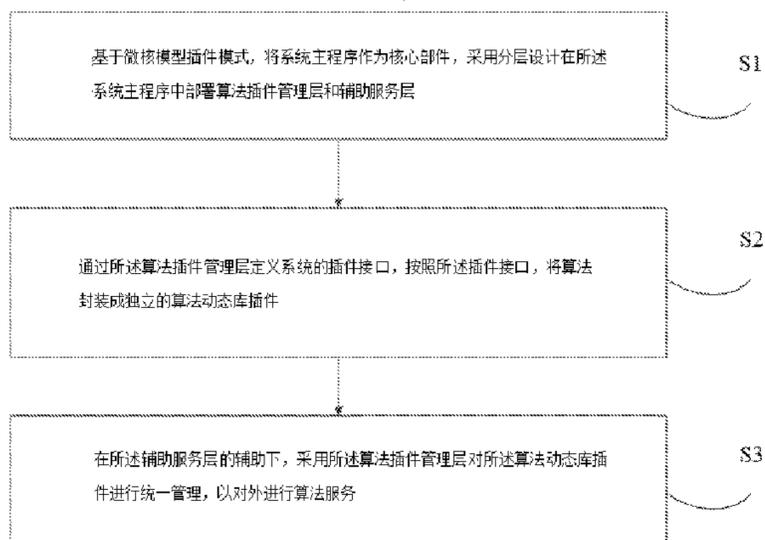


图 1

(57) Abstract: Disclosed are a service method and apparatus based on a micro-kernel model plug-in mode, and an electronic device and a computer-readable storage medium. The method comprises: on the basis of a micro-kernel model plug-in mode, taking a system main program as a core component, and deploying an algorithm plug-in management layer and an auxiliary service layer in the system main program by using a hierarchical design (S1); defining a plug-in interface of a system by means of the algorithm plug-in management layer, and encapsulating an algorithm into an independent algorithm dynamic library plug-in according to the plug-in interface (S2); and with the assistance of the auxiliary service layer, using the algorithm plug-in management layer to carry out unified management on the algorithm dynamic library plug-in so as to carry out algorithm service externally (S3). Therefore, the expansibility, universality, concurrency and compatibility of the algorithm functions can be improved.

(57) 摘要: 一种基于微核模型插件模式的服务方法、装置、电子设备及计算机可读存储介质, 包括: 基于微核模型插件模式, 将系统主程序作为核心部件, 采用分层设计在系统主程序中部署算法插件管理层和辅助服务层(S1); 通过算法插件管理层定义系统的插件接口, 按照插件接口, 将算法封装成独立的算法动态库插件(S2); 在辅助服务层的辅助下, 采用算法插件管理层对算法动态库插件进行统一管理, 以对外进行算法服务(S3), 可以提高算法功能的扩展性、通用性、并发性和兼容性。

S1 ON THE BASIS OF A MICRO-KERNEL MODEL PLUG-IN MODE, TAKE A SYSTEM MAIN PROGRAM AS A CORE COMPONENT, AND DEPLOY AN ALGORITHM PLUG-IN MANAGEMENT LAYER AND AN AUXILIARY SERVICE LAYER IN THE SYSTEM MAIN PROGRAM BY USING A HIERARCHICAL DESIGN
S2 DEFINE A PLUG-IN INTERFACE OF A SYSTEM BY MEANS OF THE ALGORITHM PLUG-IN MANAGEMENT LAYER, AND ENCAPSULATE AN ALGORITHM INTO AN INDEPENDENT ALGORITHM DYNAMIC LIBRARY PLUG-IN ACCORDING TO THE PLUG-IN INTERFACE
S3 WITH THE ASSISTANCE OF THE AUXILIARY SERVICE LAYER, USE THE ALGORITHM PLUG-IN MANAGEMENT LAYER TO CARRY OUT UNIFIED MANAGEMENT ON THE ALGORITHM DYNAMIC LIBRARY PLUG-IN SO AS TO CARRY OUT ALGORITHM SERVICE EXTERNALLY

WO 2021/139358 A1

(81) 指定国(除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, IT, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW。

(84) 指定国(除另有指明, 要求每一种可提供的地区保护): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

- 包括国际检索报告(条约第21条(3))。
- 在修改权利要求的期限届满之前进行, 在收到该修改后将重新公布(细则48.2(h))。
- 根据申请人的请求, 在条约第21条(2)(a)所规定的期限届满之前进行。

基于微核模型插件模式的服务方法、装置、设备及存储介质

5 本申请要求于 2020 年 07 月 30 日提交中国专利局、申请号为 202010755915.9，发明名称为“基于微核模型插件模式的服务方法、装置、设备及存储介质”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

本申请涉及人工智能，尤其是涉及一种基于微核模型插件模式的服务方法、装置、设备及存储介质。

10 背景技术

随着目前 AI 人工智能技术的不断发展、进步，这些技术将能够改变现代生活，甚至能够使得某些行业发生根本性的变革，人工智能的应用逐渐涉及到日常生活的方方面面。AI 人工智能技术主要依靠的是数据、算力、算法，数据是模型训练的基础，算力是训练模型的工具，算法就是模型的训练方法，训练方法不一样，就导致了各种算法的诞生或者衍化。

15 现有技术一般情况下是将模型根据项目的实际需要（实际应用场景）进行深度结合，编码实现（训练出模型），以满足项目或客户的需求，为客户提供某种具体的功能。发明人意识到这种方式，虽然实现了算法的落地，但是功能上无法扩展，客户群体之间无法通用，升级换代需要重新部署，可借鉴参考价值几乎没有，导致研发和运维人员长时间的重复劳动，产出率低下。

发明内容

本申请提供一种基于微核模型插件模式的服务方法、装置、设备及存储介质，其主要目的在于提高算法功能的扩展性、通用性、并发性和兼容性，以满足不同客户、不同场景下的应用。

25 为实现上述目的，本申请提供一种基于微核模型插件模式的服务方法，该方法包括：基于微核模型插件模式，将系统主程序作为核心部件，采用分层设计在所述系统主程序中部署算法插件管理层和辅助服务层；

通过所述算法插件管理层定义系统的插件接口，按照所述插件接口，将算法封装成独立的算法动态库插件；

30 在所述辅助服务层的辅助下，采用所述算法插件管理层对所述算法动态库插件进行统一管理，以对外进行算法服务。

为实现上述目的，本申请还提供一种基于微核模型插件模式的服务装置，所述装置包括：

35 分层部署模块，用于基于微核模型插件模式，以系统主程序为核心部件，采用分层设计在所述系统主程序中部署算法插件管理层和辅助服务层；

封装模块，用于根据所述算法插件管理层定义的系统的插件接口，将算法封装成独立的算法动态库插件；

40 管理服务模块，用于在所述辅助服务层的配合下，通过所述算法插件管理层对所述算法动态库插件进行统一管理，以对外进行算法服务。

为实现上述目的，本申请还提供一种电子设备，所述电子设备包括：

至少一个处理器；以及，

与所述至少一个处理器通信连接的存储器；其中，

所述存储器存储有可被所述至少一个处理器执行的指令，所述指令被所述至少一个处理器执行，以使所述至少一个处理器能够执行如下步骤：

5 基于微核模型插件模式，将系统主程序作为核心部件，采用分层设计在所述系统主程序中部署算法插件管理层和辅助服务层；

通过所述算法插件管理层定义系统的插件接口，按照所述插件接口，将算法封装成独立的算法动态库插件；

10 在所述辅助服务层的辅助下，采用所述算法插件管理层对所述算法动态库插件进行统一管理，以对外进行算法服务。

为实现上述目的，本申请还提供一种计算机可读存储介质，存储有计算机程序，所述计算机程序被处理器执行时实现如下步骤：

基于微核模型插件模式，将系统主程序作为核心部件，采用分层设计在所述系统主程序中部署算法插件管理层和辅助服务层；

15 通过所述算法插件管理层定义系统的插件接口，按照所述插件接口，将算法封装成独立的算法动态库插件；

在所述辅助服务层的辅助下，采用所述算法插件管理层对所述算法动态库插件进行统一管理，以对外进行算法服务。

20 本申请中的系统主程序与各算法之间形成了微核模型插件架构，系统主程序采用分层设计将算法插件管理层独立出来，利用该算法插件管理层对系统的插件接口进行定义，并在辅助服务层的辅助下对算法动态库插件进行统一性管理，实现了各算法间的独立，使得各算法间无耦合，无干扰，同时提高了算法功能的扩展性，通用性，并发性和兼容性。本申请涉及了分布式人工智能。本申请通过对逻辑程序设计，从而可以满足不同客户、不同场景下的应用；有效的降低客户的系统开发成本、后期运维成本。

25 附图说明

图 1 是本申请一实施例提供的基于微核模型插件模式的服务方法的流程图；

图 2 为本申请一实施例提供的微核模型插件架构设计示意图；

图 3 为本申请一实施例提供的基于微核模型插件模式的服务的整体架构设计图；

图 4 为本申请一实施例提供的算法间依赖解耦示意图；

30 图 5 为本申请一实施例提供的外部应用调用 remake 示意图；

图 6 是本申请一实施例提供的基于微核模型插件模式的服务装置的功能模块示意图；

图 7 为本申请一实施例提供的实现基于微核模型插件模式的服务方法的电子设备的内部结构示意图。

本申请目的的实现、功能特点及优点将结合实施例，参照附图做进一步说明。

35 具体实施方式

应当理解，此处所描述的具体实施例仅仅用以解释本申请，并不用于限定本申请。

本申请提供一种基于微核模型插件模式的服务方法。参照图 1 所示，为本申请一实施例提供的基于微核模型插件模式的服务方法的流程示意图。该方法可以由一个装置执行，该装置可以由软件和/或硬件实现。

40 在本实施例中，基于微核模型插件模式的服务方法包括：

S1、基于微核模型插件模式，将系统主程序作为核心部件，采用分层设计在所述系统主程序中部署算法插件管理层和辅助服务层。

其中，基于微核模型插件模式，以系统主程序为核心，将系统主程序与各算法之间设

计成微核模型插件架构。参照图 2 所示,为本申请一实施例提供的微核模型插件架构设计示意图,可以将算法 A 到算法 G 等算法分布式部署加载到系统主程序上以形成一种微核模型插件架构。

5 本申请通过对系统主程序进行分层设计,将算法插件管理层独立出来,辅以辅助服务层,各功能之间相互组合,后续将算法动态库插件接入后,可形成高效的算法服务。其中,所述系统主程序在不接入算法动态库插件的情况下,仍然可以保持系统的正常运行,只是不提供具体的算法功能服务。

S2、通过所述算法插件管理层定义系统的插件接口,按照所述插件接口,将算法封装成独立的算法动态库插件。

10 其中,所述定义系统的插件接口的步骤中,包括:定义插件接口形式的步骤。具体地,可以将插件接口形式定义为 Init、UnInit、Config、Version 和 Process 五大类接口。所述算法插件管理层根据插件的设计思想独立设计标准的系统的插件接口,后续算法动态库插件,只要按照标准提供符合这些定义的插件接口的对外接口,就可以无缝接入到系统主程序中,进行正常工作,提供相应的算法功能。本申请中算法动态库插件可以按照微服务的模式进行分布式或集群式部署,部署方式上比较灵活,可根据实际需要选择合适的算法功能。

15 S3、在所述辅助服务层的辅助下,采用所述算法插件管理层对所述算法动态库插件进行统一管理,以对外进行算法服务。本申请所述算法插件管理层在辅助服务层的配合下,对算法动态库插件进行统一管理,实现了各算法间的独立,使得各算法间无耦合,无干扰;同时提高了算法功能的扩展性,通用性,并发性和兼容性。

进一步地,所述算法插件管理层对算法动态库插件进行统一管理的步骤中,包括:

S31,根据算法服务请求,将算法动态库插件加载到系统主程序中;

S32,根据需要,卸载系统主程序中的算法动态库插件;

20 S33,根据算法间的依赖关系,对算法动态库插件进行依赖调用,以将多个算法动态库插件组合起来加载到系统主程序中。

图 4 示出了本申请一实施例的算法间依赖解耦示意图。如图 4 所示,针对算法动态库插件之间的依赖关系,特别设计了依赖调用接口,将算法间的依赖关系调整为由上层即算法插件管理层选择被依赖算法的调用,也有效的解决了算法间的耦合关系,使得每个算法能够独立运行,在算法内部形成闭环。如图 4 所示,现有技术中算法 B 依赖算法 A,正常情况下,部署算法 B 时必须再带一套算法 A,相当于系统服务内部存在两套算法 A,而本申请在所述算法插件管理层中部署插件依赖调用接口,该依赖调用接口的设置可以将算法 A 和算法 B 组合进行算法服务,避免了系统中存在多个相同算法 A 的问题。需要说明的是,该依赖调用接口为开放接口,其中可以包含有多种算法间的多种依赖关系。采用上述设计模式,能够将各算法有机、任意的组合,提高了算法功能的扩展性,可以提供多种形式的服务功能,满足不同客户、不同场景下的各种应用。

30 在一实施例中,所述方法还包括:根据某些特殊场景或者客户提出的一些定制化需求等,按照实际业务需求,编写指定功能的算法动态库插件,按照上述方法接入系统主程序,就能直接满足要求。这种方式大大减少了研发人员的重复性开发工作,同时也减轻了运维人员的维护工作,使得团队的重心更偏向于算法本身的研究,而无需关心工程化部署的工作。

40 本申请是将算法变成服务的通用型工程化方法,其参照微核模型插件模式和微服务模型的独立功能模式设计思想,将系统主程序部署为核心部件,具体的算法模块封装成各自独立的算法动态库插件,充分考虑了各算法的功能独特性、算法间的依赖性以及多种算法相互融合配合使用的特点,高效的将各算法糅合在一起,并且相互间无耦合,无干扰,使得算法服务功能具备了通用性、扩展性、并发性、兼容性等。针对具体某算法功能而言,

可根据实际部署运行环境，可部署能提供的全部的算法功能服务，也可按照微服务的模式进行分布式、集群式部署。微服务功能是指按照集群式或者分布式，单独部署某套算法或者某一些算法组成的服务。本系统在部署方式上比较灵活，可根据实际需要选择合适的算法功能；能够有效降低开发人员、维护人员的重复性劳动；并且本案对于网络高并发系统或者多系统融合等，都具有极高的参考价值，能够有效的降低客户的系统开发成本、后期运维成本，能够让客户将人力、资金等投入在擅长的领域，创造更大的价值。

进一步地，所述算法插件管理层对所述算法动态库插件进行统一管理的步骤中，还可以包括：S34，构建系统服务配置文件，以将算法动态库插件加载到系统主程序中。

需要说明的是，上述步骤 S31~S34 的描述并不是限定其顺序的先后，仅为了体现统一管理所包含的多个功能作用。

本申请中，所述算法插件管理层对算法动态库插件进行统一管理，是在辅助服务层的辅助下进行的。优选地，所述辅助服务层的辅助，包括：接收外部发出的算法服务请求；维护系统内算法服务请求队列，进行任务调度管理，生成调用请求；将调用请求分发至算法动态库插件中进行处理；记录或查询系统的算法服务的运行状态；对运行时的各项数据进行维护、实时更新和及时生效。

图 3 中示出了本申请一实施例中基于微核模型插件模式的服务的整体架构设计图。本申请的核心思想是将算法插件管理层独立出来，辅以上述的辅助服务层即配置管理、日志管理、并发管理、路由管理、队列管理、数据管理等等，辅助服务层内各功能相互结合，形成高效的算法服务，对外提供算法服务功能。

参考图 3 所示，所述算法插件管理层中可以包括：插件加载，插件卸载，插件接口定义，插件配置，插件调用，依赖调用等等，以实现上述对所述算法动态库插件的统一管理。所述算法插件管理层按照插件的设计思想，独立设计符合该系统的插件接口，以将对应的算法动态库插件接入系统主程序中以对外提供算法服务。

所述辅助服务层可以包括配置管理、日志管理、并发管理、路由管理、队列管理、数据管理等等，以实现上述的辅助服务层对算法插件管理层的辅助作用。具体地，所述辅助服务层通过配置管理提供系统服务的配置参数的读写，算法库配置项的读取；通过日志管理提供系统服务运行状态记录，日志查询等；通过并发管理和路由管理在高并发的调用请求时，有效的进行任务调度管理，将调用请求分发至对应的算法进行处理，提高系统服务的并发处理能力；通过队列管理对系统内消息队列进行维护，支撑并发管理任务调度等；通过数据管理提供系统运行时的各项数据的维护，实时更新，及时生效等。

可选地，如图 3 所示，还可以在系统主程序中部署网络服务层，所述网络服务层用于提供系统对外调用的对外服务接口，并通过 ZMQ 进行接口通信。具体地，在所述网络服务层中设置对外服务接口，根据外部网络环境，通过所述网络服务层调用所述对外服务接口，并基于 ZMQ 的通信模式进行通信连接。从而能适配各种编程语言及机器网络环境，也避免了由于硬件条件、开发人员技能水平、技术架构等因素导致的不兼容问题，有效降低了使用方的开发维护成本。

本申请中，步骤 S34，所述构建系统服务配置文件以将算法动态库插件加载到系统主程序中，包括：S341，识别算法动态库插件的名称，获取算法动态库插件的配置信息的所在路径；S342，根据所在路径，将所述算法动态库插件的配置信息加入到系统主程序的配置文件中，构建得到系统服务配置文件；S343，读取所述系统服务配置文件，以将算法动态库插件加载到系统主程序中。

进一步地，读取所述系统服务配置文件以将算法动态库插件加载到系统主程序中的步骤之后，还包括：S345，读取所述算法动态库插件的配置项；根据所述配置项的 `enalbed` 值，判断所述算法动态库插件是否生效；若 `enalbed` 值配置成 1，则加载成功且所述算法动态库插件生效，否则，所述算法动态库插件未生效。

下面结合一个实施例对步骤 S34 进行详细描述：以接入 remake 动态库为例，工作过程如下所述：

```

{
  "enabled":1,
  "command":1001,
  "name":"Remake",
  "lib_path":"libs/plugins/libmodRemake.so",
  "config_file":"plugins/remake/config.ini",
  "init":"mod_remake_Init",
  "uninit":"mod_remake_UnInit",
  "config":"mod_remake_Config",
  "version":"mod_remake_Version"
  "proc":"mod_remake_Detect",
}

```

1. 在系统主程序的配置文件中，加入 remake 动态库的配置信息，以获得系统服务配置
5 置文件。其中，enabled 决定该动态库是否生效，只有配置成 1，才说明加载了该动态库并且会正常工作，否则即使加载了，该动态库也不能提供对应的功能；command 是该动态库在系统服务中对应的唯一命令号；name 是该动态库在系统服务中的标识名称，即为 Remake；lib_path、config_file 是该动态库文件所在路径，及该动态库依赖的配置信息文件的所在路径；init、uninit、config、version、proc 是动态库按照系统服务中的算法插件管理
10 层的要求，提供的功能接口。

2. 设置好系统服务配置文件后，系统主程序会读取关于 remake 动态库的配置文件中
的配置信息，并且根据要求将该动态库加载到系统服务的内存中，加载成功后，remake 动
态库就能提供对应的功能。

3. 外部应用向系统的网络服务层发起命令号为 1001 的请求，经过辅助服务层将该请
15 求转发至 remake 动态库中进行处理。图 5 示出了本申请该实施例提供的外部应用调用
Remake plugin 算法插件的示意图，具体如图 5 中虚线所示。

本申请封装后的算法动态库插件构成了算法动态库。本申请集成了算法动态库的热部
署功能，采用算法功能的动态加载技术，在服务初始化启动时加载算法，或者需要使用指
定算法时，单独加载该算法；在不需要时，或者更新某算法时，根据算法卸载请求，可以
20 动态卸载指定算法。算法动态库插件在系统服务内部的状态有：未加载、已加载、已生效、
待卸载，正常运行时默认状态为已生效，如检测到待卸载状态时，则系统服务选择该算
法动态库插件空闲时进行动态库卸载操作；在算法动态库插件更新后，系统服务检测到算
法动态库插件未加载，则进行算法动态库插件的加载操作，成功则切换至已加载状态；根据
25 算法动态库插件的配置项的 enabled 值决定该算法动态库插件是否生效。采用该技术，可
以达到不停服务程序进行算法更新，也能有效的降低服务停机的损失及客户的运营成本。

如图 6 所示，是本申请一实施例提供的基于微核模型插件模式的服务装置 100 的功能
模块图。参考图 6 所示，本申请所述基于微核模型插件模式的服务装置 100 可以安装于电
子设备中。根据实现的功能，所述基于微核模型插件模式的服务装置 100 可以包括分层部
署模块 101、封装模块 102 以及管理服务模块 103。本发所述模块也可以称之为单元，是
30 指一种能够被电子设备处理器所执行，并且能够完成固定功能的一系列计算机程序段，其
存储在电子设备的存储器中。

本申请基于微核模型插件模式，以系统主程序为核心，将系统主程序与各算法之间设
计成微核模型插件架构；通过对系统主程序进行分层设计，将算法插件管理层独立出来，
辅以辅助服务层，各功能之间相互组合，后续将算法动态库插件接入后，可形成高效的算
35 法服务。其中，所述系统主程序在不接入算法动态库插件的情况下，仍然可以保持系统的

正常运行，只是不提供具体的算法功能服务。

在本实施例中，关于各模块/单元的功能如下：

所述分层部署模块 101 用于基于微核模型插件模式，以系统主程序为核心部件，采用分层设计在所述系统主程序中部署算法插件管理层和辅助服务层。

5 优选地，所述算法插件管理层可以包括：插件加载，插件卸载，插件接口定义，插件配置，插件调用，依赖调用等等，以实现与所述算法动态库插件的统一管理。

10 优选地，所述辅助服务层可以包括：配置管理、日志管理、并发管理、路由管理、队列管理、数据管理等等，以实现辅助服务层对算法插件管理层的辅助作用。具体地，所述辅助作用可以包括：接收外部发出的算法服务请求；维护系统内算法服务请求队列，进行任务调度管理，生成调用请求；将调用请求分发至算法动态库插件中进行处理；记录或查询系统的算法服务的运行状态；对运行时的各项数据进行维护、实时更新和及时生效。

进一步地，所述分层部署模块 101 还用于在所述系统主程序中部署网络服务层，并在所述网络服务层中设置对外服务接口；根据外部网络环境，通过所述网络服务层调用所述对外服务接口，并基于 ZMQ 的通信模式进行通信连接。

15 所述封装模块 102 用于根据所述算法插件管理层定义的系统的插件接口，将算法封装成独立的算法动态库插件。所述算法插件管理层根据插件的设计思想独立设计标准的系统的插件接口，后续算法动态库插件，只要按照标准提供符合这些定义的插件接口的对外接口，就可以无缝接入到系统主程序中，进行正常工作，提供相应的算法功能。本申请中算法动态库插件可以按照微服务的模式进行分布式或集群式部署，部署方式上比较灵活，可根据实际需要选择合适的算法功能。

所述管理服务模块 103 用于在所述辅助服务层的配合下，通过所述算法插件管理层对所述算法动态库插件进行统一管理，以对外进行算法服务。所述算法插件管理层在辅助服务层的配合下，对算法动态库插件进行统一管理，实现了各算法间的独立，使得各算法间无耦合，无干扰；同时提高了算法功能的扩展性，通用性，并发性和兼容性。

25 其中，通过所述算法插件管理层对所述算法动态库插件进行统一管理，包括：根据算法服务请求，将算法动态库插件加载到系统主程序中；根据需要，卸载系统主程序中的算法动态库插件；根据算法间的依赖关系，对算法动态库插件进行依赖调用，以将多个算法动态库插件组合起来加载到系统主程序中。

30 进一步地，通过所述算法插件管理层对所述算法动态库插件进行统一管理，还可以包括：构建系统服务配置文件，以将算法动态库插件加载到系统主程序中。具体地，所述构建系统服务配置文件，以将算法动态库插件加载到系统主程序中，包括：识别所述算法动态库插件的名称，获取所述算法动态库插件的配置信息的所在路径；根据所在路径，将所述算法动态库插件的配置信息加入到系统主程序的配置文件中，构建得到系统服务配置文件；读取所述系统服务配置文件，以将算法动态库插件加载到系统主程序中。

35 进一步地，所述构建系统服务配置文件，以将算法动态库插件加载到系统主程序中，还可以包括：读取所述算法动态库插件的配置项；根据所述配置项的 `enabled` 值，判断所述算法动态库插件是否生效；若 `enabled` 值配置成 1，则加载成功且所述算法动态库插件生效，否则，所述算法动态库插件未生效。

40 优选地，所述基于微核模型插件模式的服务装置 100 还包括：添加插件模块，用于根据定制化需求，编写指定功能的算法动态库插件，以加载到系统主程序中进行相应的算法服务。

如图 7 所示，是本申请实现服务程序的电子设备的结构示意图。

所述电子设备 1 可以包括处理器 10、存储器 11 和总线，还可以包括存储在所述存储器 11 中并可在所述处理器 10 上运行的计算机程序，如服务程序 12。

45 其中，所述存储器 11 至少包括一种类型的可读存储介质，所述可读存储介质包括闪存、

移动硬盘、多媒体卡、卡型存储器（例如：SD 或 DX 存储器等）、磁性存储器、磁盘、光盘等。所述存储器 11 在一些实施例中可以是电子设备 1 的内部存储单元，例如该电子设备 1 的移动硬盘。所述存储器 11 在另一些实施例中也可以是电子设备 1 的外部存储设备，例如电子设备 1 上配备的插接式移动硬盘、智能存储卡（Smart Media Card, SMC）、安全数字（Secure Digital, SD）卡、闪存卡（Flash Card）等。进一步地，所述存储器 11 还可以既包括电子设备 1 的内部存储单元也包括外部存储设备。所述存储器 11 不仅可以用于存储安装于电子设备 1 的应用软件及各类数据，例如服务程序的代码等，还可以用于暂时地存储已经输出或者将要输出的数据。

所述处理器 10 在一些实施例中可以由集成电路组成，例如可以由单个封装的集成电路所组成，也可以是由多个相同功能或不同功能封装的集成电路所组成，包括一个或者多个中央处理器（Central Processing unit, CPU）、微处理器、数字处理芯片、图形处理器及各种控制芯片的组合等。所述处理器 10 是所述电子设备的控制核心（Control Unit），利用各种接口和线路连接整个电子设备的各个部件，通过运行或执行存储在所述存储器 11 内的程序或者模块（例如服务程序等），以及调用存储在所述存储器 11 内的数据，以执行电子设备 1 的各种功能和处理数据。

所述总线可以是外设部件互连标准（peripheral component interconnect, 简称 PCI）总线或扩展工业标准结构（extended industry standard architecture, 简称 EISA）总线等。该总线可以分为地址总线、数据总线、控制总线等。所述总线被设置为实现所述存储器 11 以及至少一个处理器 10 等之间的连接通信。

图 7 仅示出了具有部件的电子设备的结构，本领域技术人员可以理解的是，图 6 示出的结构并不构成对所述电子设备 1 的限定，可以包括比图示更少或者更多的部件，或者组合某些部件，或者不同的部件布置。

例如，尽管未示出，所述电子设备 1 还可以包括给各个部件供电的电源（比如电池），优选地，电源可以通过电源管理装置与所述至少一个处理器 10 逻辑相连，从而通过电源管理装置实现充电管理、放电管理、以及功耗管理等功能。电源还可以包括一个或一个以上的直流或交流电源、再充电装置、电源故障检测电路、电源转换器或者逆变器、电源状态指示器等任意组件。所述电子设备 1 还可以包括多种传感器、蓝牙模块、Wi-Fi 模块等，在此不再赘述。

进一步地，所述电子设备 1 还可以包括网络接口，可选地，所述网络接口可以包括有线接口和/或无线接口（如 WI-FI 接口、蓝牙接口等），通常用于在该电子设备 1 与其他电子设备之间建立通信连接。

可选地，该电子设备 1 还可以包括用户接口，用户接口可以是显示器（Display）、输入单元（比如键盘（Keyboard）），可选地，用户接口还可以是标准的有线接口、无线接口。可选地，在一些实施例中，显示器可以是 LED 显示器、液晶显示器、触控式液晶显示器以及 OLED（Organic Light-Emitting Diode, 有机发光二极管）触摸器等。其中，显示器也可以适当的称为显示屏或显示单元，用于显示在电子设备 1 中处理的信息以及用于显示可视化的用户界面。

应该了解，所述实施例仅为说明之用，在专利申请范围上并不受此结构的限制。

所述电子设备 1 中的所述存储器 11 存储的服务程序 12 是多个指令的组合，在所述处理器 10 中运行时，可以实现：基于微核模型插件模式，将系统主程序作为核心部件，采用分层设计在所述系统主程序中部署算法插件管理层和辅助服务层；通过所述算法插件管理层定义系统的插件接口，按照所述插件接口，将算法封装成独立的算法动态库插件；在所述辅助服务层的辅助下，采用所述算法插件管理层对所述算法动态库插件进行统一管理，以对外进行算法服务。

具体地，所述处理器 10 对上述指令的具体实现方法可参考图 1 对应实施例中相关步

骤的描述，在此不赘述。

进一步地，所述电子设备 1 集成的模块/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，可以存储在一个计算机可读存储介质中。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）。

此外，本申请实施例还提出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非易失性，也可以是易失性，计算机可读存储介质中包括计算机程序，该计算机程序被处理器执行时实现如下操作：

基于微核模型插件模式，将系统主程序作为核心部件，采用分层设计在所述系统主程序中部署算法插件管理层和辅助服务层；

通过所述算法插件管理层定义系统的插件接口，按照所述插件接口，将算法封装成独立的算法动态库插件；

在所述辅助服务层的辅助下，采用所述算法插件管理层对所述算法动态库插件进行统一管理，以对外进行算法服务。

本申请计算机可读存储介质的具体实施方式与上述基于微核模型插件模式的服务方法、装置、电子设备的具体实施方式大致相同，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，所揭露的设备，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仅是示意性的，例如，所述模块的划分，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。

所述作为分离部件说明的模块可以是或者也可以不是物理上分开的，作为模块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模块来实现本实施例方案的目的。

另外，在本申请各个实施例中的各功能模块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，也可以采用硬件加软件功能模块的形式实现。

对于本领域技术人员而言，显然本申请不限于上述示范性实施例的细节，而且在不背离本申请的精神或基本特征的情况下，能够以其他的具体形式实现本申请。

因此，无论从哪一点来看，均应将实施例看作是示范性的，而且是非限制性的，本申请的范围由所附权利要求而不是上述说明限定，因此旨在将落在权利要求的等同要件的含义和范围内的所有变化涵括在本申请内。不应将权利要求中的任何附关联图标记视为限制所涉及的权利要求。

此外，显然“包括”一词不排除其他单元或步骤，单数不排除复数。系统权利要求中陈述的多个单元或装置也可以由一个单元或装置通过软件或者硬件来实现。第二等词语用来表示名称，而并不表示任何特定的顺序。

最后应说明的是，以上实施例仅用以说明本申请的技术方案而非限制，尽管参照较佳实施例对本申请进行了详细说明，本领域的普通技术人员应当理解，可以对本申请的技术方案进行修改或等同替换，而不脱离本申请技术方案的精神和范围。

权 利 要 求 书

1. 一种基于微核模型插件模式的服务方法，其中，所述方法包括：
基于微核模型插件模式，将系统主程序作为核心部件，采用分层设计在所述系统主程序中部署算法插件管理层和辅助服务层；
通过所述算法插件管理层定义系统的插件接口，按照所述插件接口，将算法封装成独立的算法动态库插件；
在所述辅助服务层的辅助下，采用所述算法插件管理层对所述算法动态库插件进行统一管理，以对外进行算法服务。
2. 根据权利要求 1 所述的基于微核模型插件模式的服务方法，其中，采用所述算法插件管理层对所述算法动态库插件进行统一管理的步骤中，包括：
根据算法服务请求，将算法动态库插件加载到系统主程序中；
根据算法卸载请求，卸载系统主程序中的算法动态库插件；
根据算法间的依赖关系，对算法动态库插件进行依赖调用，以将多个算法动态库插件组合起来加载到系统主程序中。
3. 根据权利要求 2 所述的基于微核模型插件模式的服务方法，其中，采用所述算法插件管理层对所述算法动态库插件进行统一管理的步骤中，还包括：构建系统服务配置文件，以将算法动态库插件加载到系统主程序中；所述构建系统服务配置文件，以将算法动态库插件加载到系统主程序中的步骤，包括：
识别所述算法动态库插件的名称，获取所述算法动态库插件的配置信息的所在路径；
根据所在路径，将所述算法动态库插件的配置信息加入到系统主程序的配置文件中，构建得到系统服务配置文件；
读取所述系统服务配置文件，以将算法动态库插件加载到系统主程序中。
4. 根据权利要求 3 所述的基于微核模型插件模式的服务方法，其中，读取所述系统服务配置文件以将算法动态库插件加载到系统主程序中的步骤之后，还包括：
读取所述算法动态库插件的配置项；
根据所述配置项的 `enabled` 值，判断所述算法动态库插件是否生效；若 `enabled` 值配置成 1，则加载成功且所述算法动态库插件生效，否则，所述算法动态库插件未生效。
5. 根据权利要求 1 所述的基于微核模型插件模式的服务方法，其中，所述辅助服务层的辅助，包括：
接收外部发出的算法服务请求；
维护系统内算法服务请求队列，进行任务调度管理，生成调用请求；
将调用请求分发至算法动态库插件中进行处理；
记录或查询系统的算法服务的运行状态；
对运行时的各项数据进行维护、实时更新和及时生效。
6. 根据权利要求 1 所述的基于微核模型插件模式的服务方法，其中，所述方法还包括：在所述系统主程序中部署网络服务层，在所述网络服务层中设置对外服务接口；根据外部网络环境，通过所述网络服务层调用所述对外服务接口，并基于 ZMQ 的通信模式进行通信连接。
7. 根据权利要求 1 所述的基于微核模型插件模式的服务方法，其中，所述方法还包括：根据定制化需求，编写指定功能的算法动态库插件，以加载到系统主程序中进行相应的算法服务。
8. 一种基于微核模型插件模式的服务装置，其中，所述装置包括：
分层部署模块，用于基于微核模型插件模式，以系统主程序为核心部件，采用分层设

计在所述系统主程序中部署算法插件管理层和辅助服务层；

封装模块，用于根据所述算法插件管理层定义的系统的插件接口，将算法封装成独立的算法动态库插件；

管理服务模块，用于在所述辅助服务层的配合下，通过所述算法插件管理层对所述算法动态库插件进行统一管理，以对外进行算法服务。

9. 一种电子设备，其中，所述电子设备包括：

至少一个处理器；以及，

与所述至少一个处理器通信连接的存储器；其中，

所述存储器存储有可被所述至少一个处理器执行的指令，所述指令被所述至少一个处理器执行，以使所述至少一个处理器能够执行如下步骤：

基于微核模型插件模式，将系统主程序作为核心部件，采用分层设计在所述系统主程序中部署算法插件管理层和辅助服务层；

通过所述算法插件管理层定义系统的插件接口，按照所述插件接口，将算法封装成独立的算法动态库插件；

在所述辅助服务层的辅助下，采用所述算法插件管理层对所述算法动态库插件进行统一管理，以对外进行算法服务。

10. 根据权利要求 9 所述的电子设备，其中，所述至少一个处理器执行的所述采用算法插件管理层对算法动态库插件进行统一管理的步骤中，包括：

根据算法服务请求，将算法动态库插件加载到系统主程序中；

根据算法卸载请求，卸载系统主程序中的算法动态库插件；

根据算法间的依赖关系，对算法动态库插件进行依赖调用，以将多个算法动态库插件组合起来加载到系统主程序中。

11. 根据权利要求 10 所述的电子设备，其中，所述至少一个处理器执行的所述采用算法插件管理层对算法动态库插件进行统一管理的步骤中，还包括：构建系统服务配置文件，以将算法动态库插件加载到系统主程序中；

所述构建系统服务配置文件，以将算法动态库插件加载到系统主程序中的步骤，包括：

识别所述算法动态库插件的名称，获取所述算法动态库插件的配置信息的所在路径；

根据所在路径，将所述算法动态库插件的配置信息加入到系统主程序的配置文件中，构建得到系统服务配置文件；

读取所述系统服务配置文件，以将算法动态库插件加载到系统主程序中。

12. 根据权利要求 11 所述的电子设备，其中，所述至少一个处理器执行的所述读取系统服务配置文件以将算法动态库插件加载到系统主程序中的步骤之后，还包括：

读取所述算法动态库插件的配置项；

根据所述配置项的 `enabled` 值，判断所述算法动态库插件是否生效；若 `enabled` 值配置成 1，则加载成功且所述算法动态库插件生效，否则，所述算法动态库插件未生效。

13. 根据权利要求 11 所述的电子设备，其中，所述至少一个处理器执行的在辅助服务层的辅助下采用算法插件管理层对算法动态库插件进行统一管理以对外进行算法服务的步骤中，所述辅助服务层的辅助，包括：

接收外部发出的算法服务请求；

维护系统内算法服务请求队列，进行任务调度管理，生成调用请求；

将调用请求分发至算法动态库插件中进行处理；

记录或查询系统的算法服务的运行状态；

对运行时的各项数据进行维护、实时更新和及时生效。

14. 根据权利要求 11 所述的电子设备，其中，所述至少一个处理器执行的步骤，还包括：在所述系统主程序中部署网络服务层，在所述网络服务层中设置对外服务接口；根据

外部网络环境，通过所述网络服务层调用所述对外服务接口，并基于 ZMQ 的通信模式进行通信连接。

15. 根据权利要求 11 所述的电子设备，其中，所述至少一个处理器执行的步骤，还包括：根据定制化需求，编写指定功能的算法动态库插件，以加载到系统主程序中进行相应的算法服务。

16. 一种计算机可读存储介质，存储有计算机程序，其中，所述计算机程序被处理器执行时实现如下步骤：

基于微核模型插件模式，将系统主程序作为核心部件，采用分层设计在所述系统主程序中部署算法插件管理层和辅助服务层；

通过所述算法插件管理层定义系统的插件接口，按照所述插件接口，将算法封装成独立的算法动态库插件；

在所述辅助服务层的辅助下，采用所述算法插件管理层对所述算法动态库插件进行统一管理，以对外进行算法服务。

17. 根据权利要求 16 所述的计算机可读存储介质，其中，所述计算机程序被处理器执行时实现的所述采用算法插件管理层对算法动态库插件进行统一管理的步骤中，包括：

根据算法服务请求，将算法动态库插件加载到系统主程序中；

根据算法卸载请求，卸载系统主程序中的算法动态库插件；

根据算法间的依赖关系，对算法动态库插件进行依赖调用，以将多个算法动态库插件组合起来加载到系统主程序中。

18. 根据权利要求 17 所述的计算机可读存储介质，其中，所述计算机程序被处理器执行时实现的所述算法插件管理层对算法动态库插件进行统一管理的步骤中，还包括：

构建系统服务配置文件，以将算法动态库插件加载到系统主程序中；

所述构建系统服务配置文件，以将算法动态库插件加载到系统主程序中的步骤，包括：

识别所述算法动态库插件的名称，获取所述算法动态库插件的配置信息的所在路径；

根据所在路径，将所述算法动态库插件的配置信息加入到系统主程序的配置文件中，

构建得到系统服务配置文件；

读取所述系统服务配置文件，以将算法动态库插件加载到系统主程序中。

19. 根据权利要求 18 所述的计算机可读存储介质，其中，所述计算机程序被处理器执行时实现的所述读取系统服务配置文件以将算法动态库插件加载到系统主程序中的步骤之后，还包括：

读取所述算法动态库插件的配置项；

根据所述配置项的 `enabled` 值，判断所述算法动态库插件是否生效；若 `enabled` 值配置成 1，则加载成功且所述算法动态库插件生效，否则，所述算法动态库插件未生效。

20. 根据权利要求 16 所述的计算机可读存储介质，其中，所述计算机程序被处理器执行时实在所述辅助服务层的辅助下采用所述算法插件管理层对算法动态库插件进行统一管理以对外进行算法服务的步骤中，所述辅助服务层的辅助，包括：

接收外部发出的算法服务请求；

维护系统内算法服务请求队列，进行任务调度管理，生成调用请求；

将调用请求分发至算法动态库插件中进行处理；

记录或查询系统的算法服务的运行状态；

对运行时的各项数据进行维护、实时更新和及时生效。

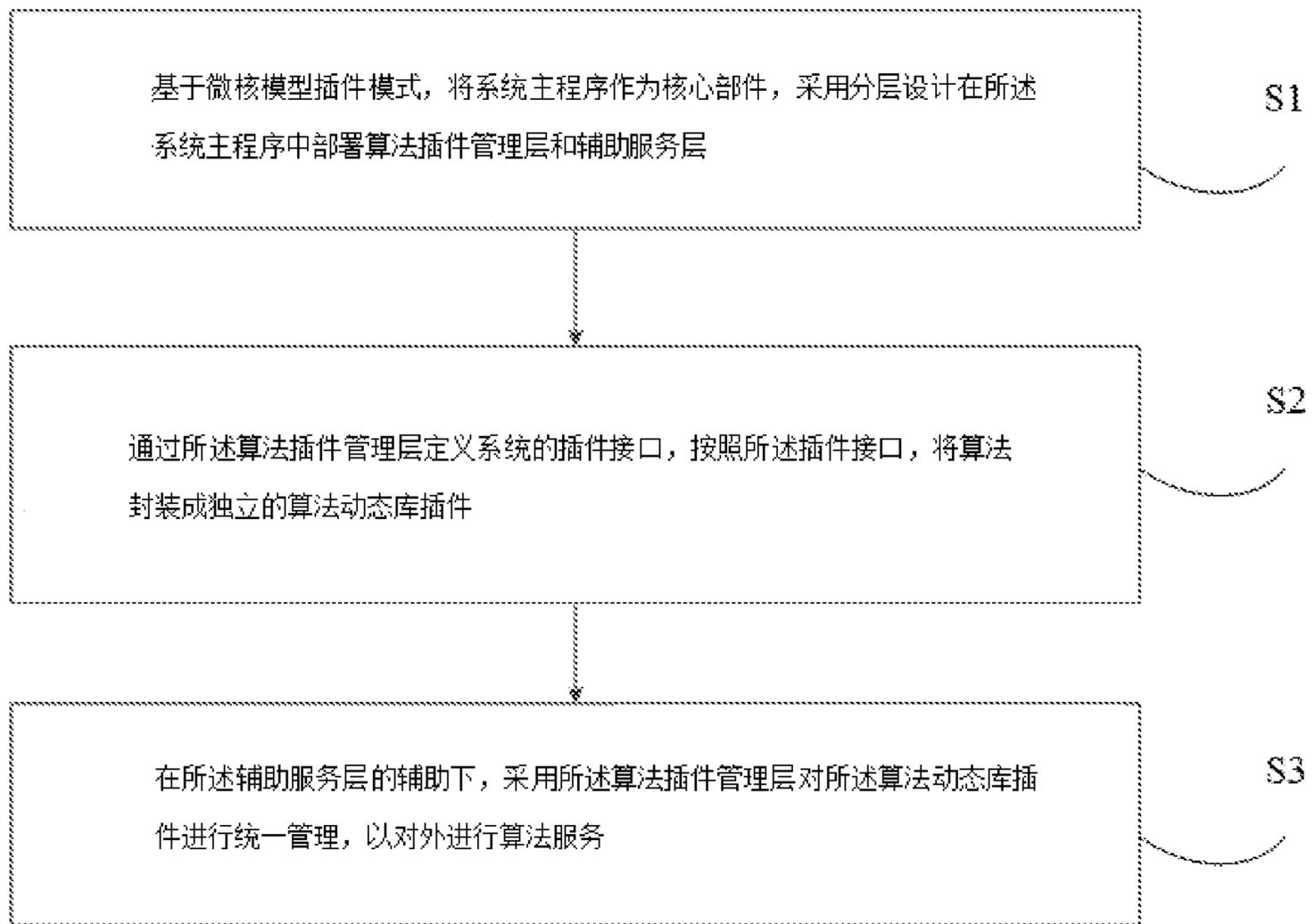


图 1

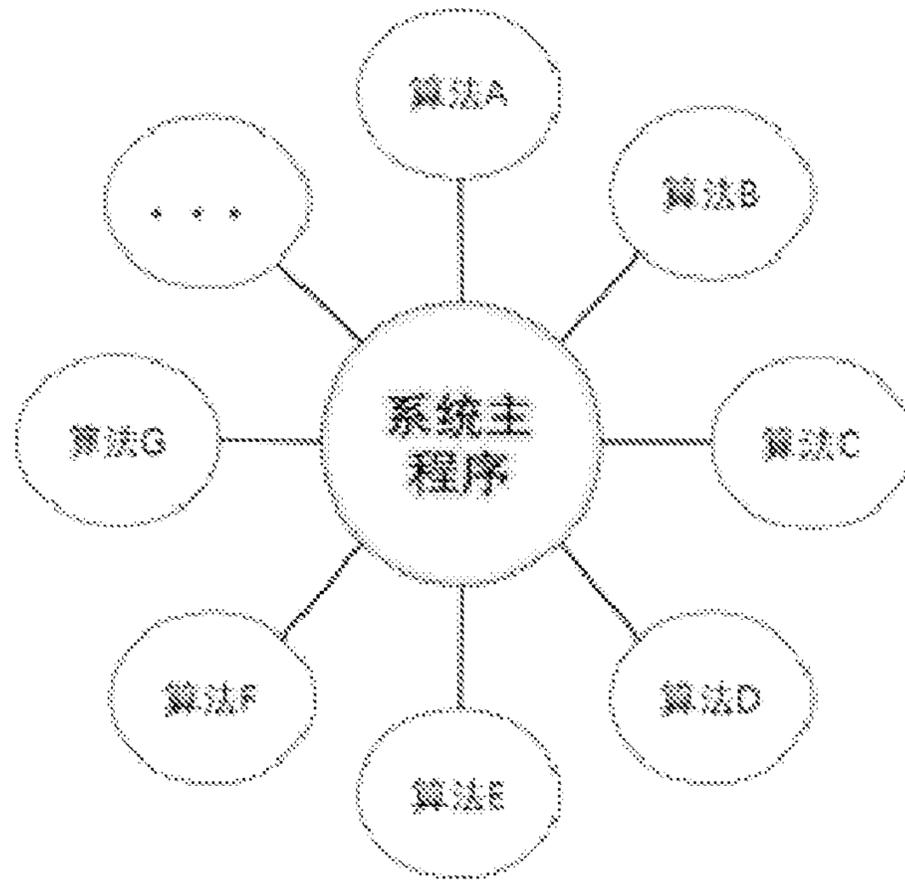


图 2

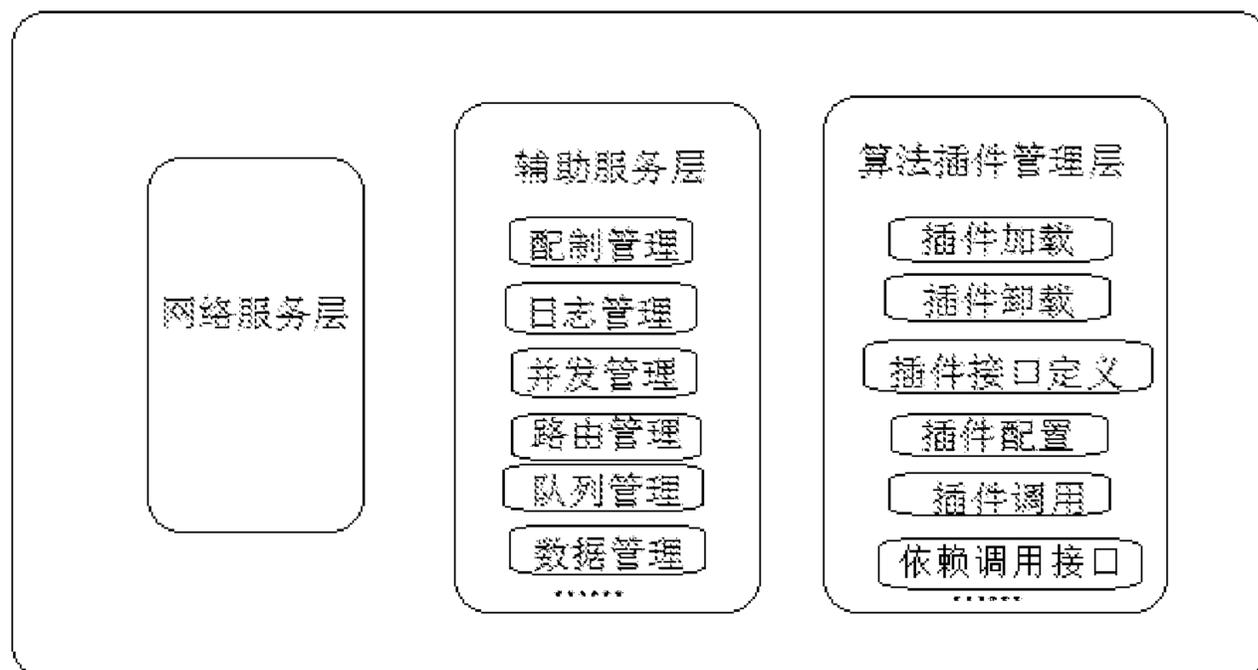


图 3

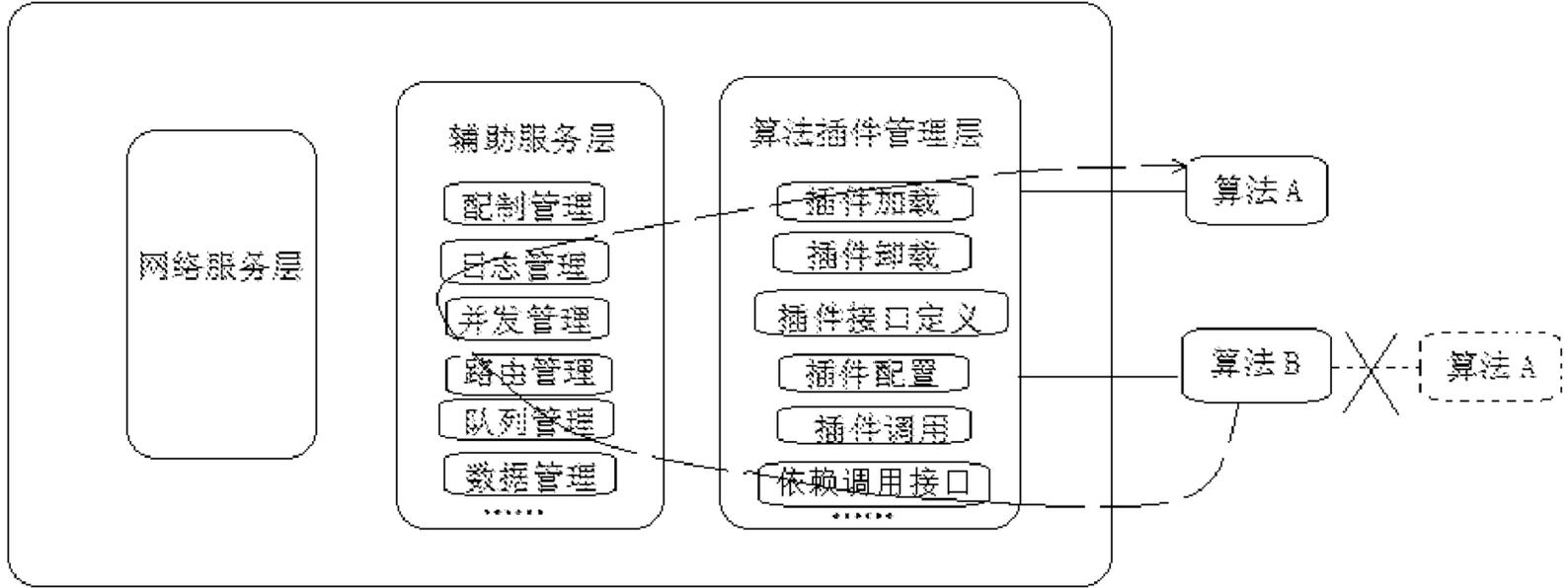


图 4

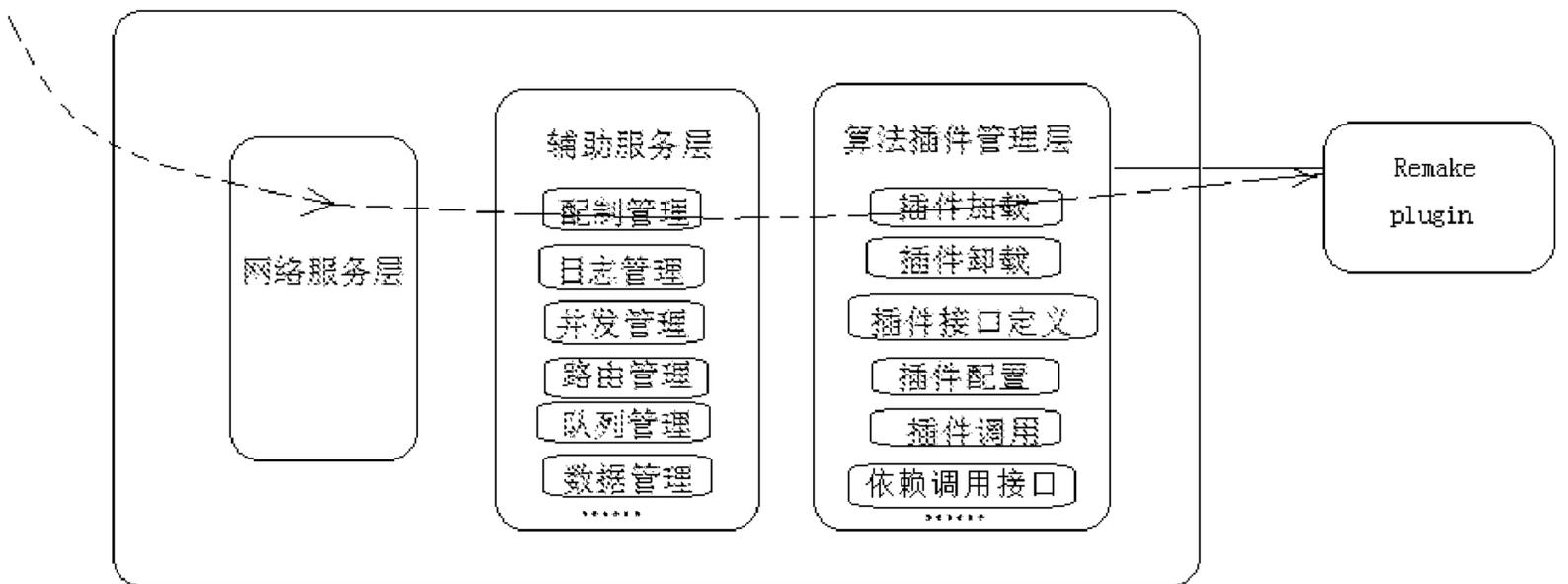


图 5



图 6

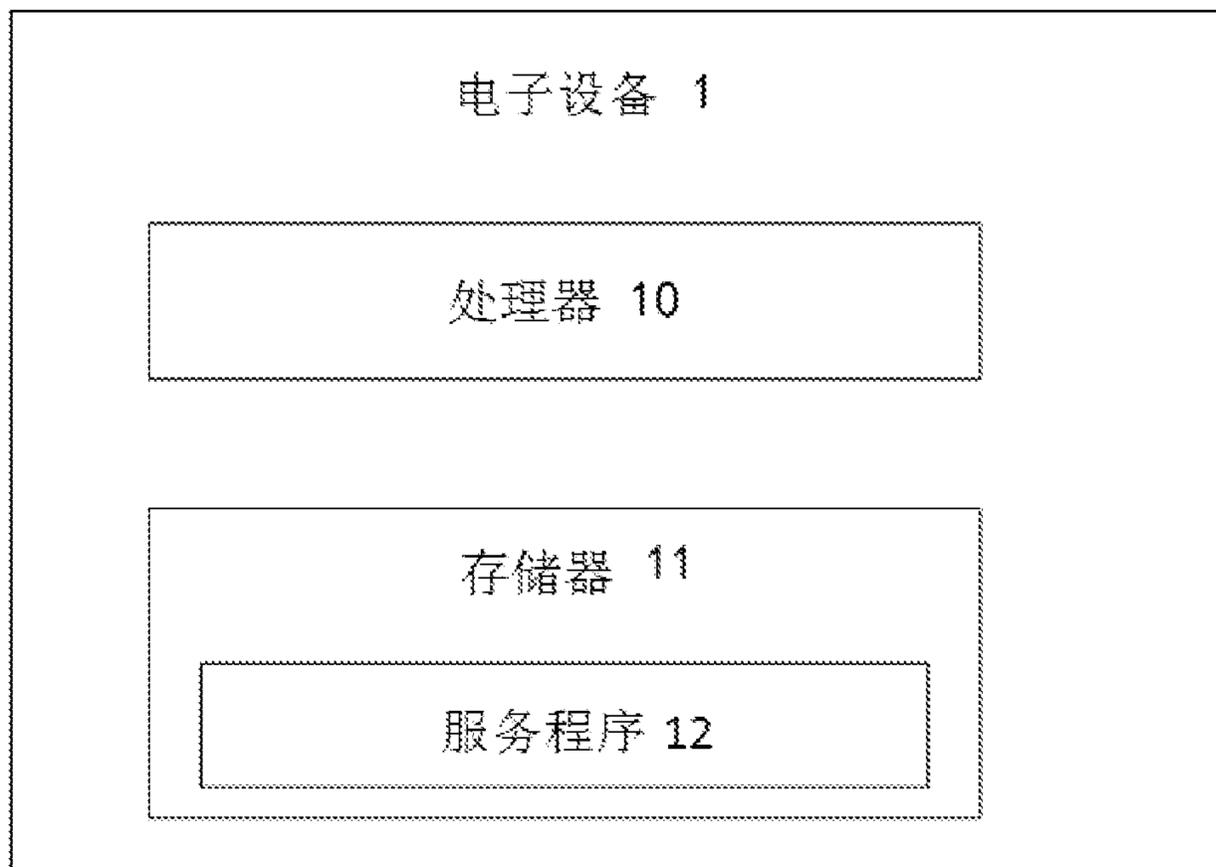


图 7

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2020/125075

A. CLASSIFICATION OF SUBJECT MATTER G06F 9/445(2018.01)i; G06F 9/48(2006.01)i According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED Minimum documentation searched (classification system followed by classification symbols) G06F Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched Electronic data base consulted during the international search (name of data base and, where practicable, search terms used) CNABS; CNTXT; VEN; EPTXT; WOTXT; USTXT; CNKI: 人工智能, 算法, 服务, 插件, 组件, 模块, 微服务, 轻服务, 轻应用, 接口, 库, 封装, AI, algorithm, service, plug-in, component, module, micro service, mini application, interface, API, lib, encapsulate		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	CN 102929606 A (DAWNING INFORMATION INDUSTRY (BEIJING) CO., LTD.) 13 February 2013 (2013-02-13) description, paragraphs 3, 15, 19-24, figure 2	1-2, 5-10, 13-17, 20
Y	CN 102929606 A (DAWNING INFORMATION INDUSTRY (BEIJING) CO., LTD.) 13 February 2013 (2013-02-13) description, paragraphs 3, 15, 19-24, figure 2	3-4, 11-12, 18-19
X	CN 110245031 A (BEIJING BAIDU NETCOM SCIENCE AND TECHNOLOGY CO., LTD.) 17 September 2019 (2019-09-17) description, paragraphs 80-85, 107, 151, 157	1-2, 5-10, 13-17, 20
Y	CN 110245031 A (BEIJING BAIDU NETCOM SCIENCE AND TECHNOLOGY CO., LTD.) 17 September 2019 (2019-09-17) description, paragraphs 80-85, 107, 151, 157	3-4, 11-12, 18-19
X	CN 110489093 A (PENG CHENG LABORATORY) 22 November 2019 (2019-11-22) description paragraphs 27, 34-36	1-2, 5-10, 13-17, 20
Y	CN 110489093 A (PENG CHENG LABORATORY) 22 November 2019 (2019-11-22) description, paragraphs 27, 34-36	3-4, 11-12, 18-19
<input checked="" type="checkbox"/> Further documents are listed in the continuation of Box C. <input checked="" type="checkbox"/> See patent family annex.		
* Special categories of cited documents: “A” document defining the general state of the art which is not considered to be of particular relevance “E” earlier application or patent but published on or after the international filing date “L” document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) “O” document referring to an oral disclosure, use, exhibition or other means “P” document published prior to the international filing date but later than the priority date claimed		“T” later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention “X” document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone “Y” document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art “&” document member of the same patent family
Date of the actual completion of the international search 16 April 2021		Date of mailing of the international search report 28 April 2021
Name and mailing address of the ISA/CN China National Intellectual Property Administration (ISA/ CN) No. 6, Xitucheng Road, Jimenqiao, Haidian District, Beijing 100088 China Facsimile No. (86-10)62019451		Authorized officer Telephone No.

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CN2020/125075**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	CN 109165055 A (BAIDU ONLINE NETWORK TECHNOLOGY (BEIJING) CO., LTD.) 08 January 2019 (2019-01-08) description, paragraph 67	3-4, 11-12, 18-19
A	US 2018357552 A1 (BONSAI AI INC.) 13 December 2018 (2018-12-13) entire document	1-20

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.

PCT/CN2020/125075

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)	Publication date (day/month/year)
CN	102929606	A	13 February 2013	None	
CN	110245031	A	17 September 2019	None	
CN	110489093	A	22 November 2019	None	
CN	109165055	A	08 January 2019	None	
US	2018357552	A1	13 December 2018	None	

国际检索报告

国际申请号

PCT/CN2020/125075

<p>A. 主题的分类</p> <p>G06F 9/445(2018.01)i; G06F 9/48(2006.01)i</p> <p>按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类</p>																										
<p>B. 检索领域</p> <p>检索的最低限度文献(标明分类系统和分类号)</p> <p>G06F</p> <p>包含在检索领域中的除最低限度文献以外的检索文献</p> <p>在国际检索时查阅的电子数据库(数据库的名称, 和使用的检索词(如使用))</p> <p>CNABS;CNTXT;VEN;EPTXT;WOTXT;USTXT;CNKI: 人工智能, 算法, 服务, 插件, 组件, 模块, 微服务, 轻服务, 轻应用, 接口, 库, 封装, AI, algorithm, service, plug-in, component, module, micro service, mini application, interface, API, lib, encapsulate</p>																										
<p>C. 相关文件</p> <table border="1"> <thead> <tr> <th>类型*</th> <th>引用文件, 必要时, 指明相关段落</th> <th>相关的权利要求</th> </tr> </thead> <tbody> <tr> <td>X</td> <td>CN 102929606 A (曙光信息产业北京有限公司) 2013年 2月 13日 (2013 - 02 - 13) 说明书第3、15、19-24段, 图2</td> <td>1-2, 5-10, 13-17, 20</td> </tr> <tr> <td>Y</td> <td>CN 102929606 A (曙光信息产业北京有限公司) 2013年 2月 13日 (2013 - 02 - 13) 说明书第3、15、19-24段, 图2</td> <td>3-4, 11-12, 18-19</td> </tr> <tr> <td>X</td> <td>CN 110245031 A (北京百度网讯科技有限公司) 2019年 9月 17日 (2019 - 09 - 17) 说明书第80-85、107、151、157段</td> <td>1-2, 5-10, 13-17, 20</td> </tr> <tr> <td>Y</td> <td>CN 110245031 A (北京百度网讯科技有限公司) 2019年 9月 17日 (2019 - 09 - 17) 说明书第80-85、107、151、157段</td> <td>3-4, 11-12, 18-19</td> </tr> <tr> <td>X</td> <td>CN 110489093 A (鹏城实验室) 2019年 11月 22日 (2019 - 11 - 22) 说明书第27、34-36段</td> <td>1-2, 5-10, 13-17, 20</td> </tr> <tr> <td>Y</td> <td>CN 110489093 A (鹏城实验室) 2019年 11月 22日 (2019 - 11 - 22) 说明书第27、34-36段</td> <td>3-4, 11-12, 18-19</td> </tr> <tr> <td>Y</td> <td>CN 109165055 A (百度在线网络技术北京有限公司) 2019年 1月 8日 (2019 - 01 - 08) 说明书第67段</td> <td>3-4, 11-12, 18-19</td> </tr> </tbody> </table>			类型*	引用文件, 必要时, 指明相关段落	相关的权利要求	X	CN 102929606 A (曙光信息产业北京有限公司) 2013年 2月 13日 (2013 - 02 - 13) 说明书第3、15、19-24段, 图2	1-2, 5-10, 13-17, 20	Y	CN 102929606 A (曙光信息产业北京有限公司) 2013年 2月 13日 (2013 - 02 - 13) 说明书第3、15、19-24段, 图2	3-4, 11-12, 18-19	X	CN 110245031 A (北京百度网讯科技有限公司) 2019年 9月 17日 (2019 - 09 - 17) 说明书第80-85、107、151、157段	1-2, 5-10, 13-17, 20	Y	CN 110245031 A (北京百度网讯科技有限公司) 2019年 9月 17日 (2019 - 09 - 17) 说明书第80-85、107、151、157段	3-4, 11-12, 18-19	X	CN 110489093 A (鹏城实验室) 2019年 11月 22日 (2019 - 11 - 22) 说明书第27、34-36段	1-2, 5-10, 13-17, 20	Y	CN 110489093 A (鹏城实验室) 2019年 11月 22日 (2019 - 11 - 22) 说明书第27、34-36段	3-4, 11-12, 18-19	Y	CN 109165055 A (百度在线网络技术北京有限公司) 2019年 1月 8日 (2019 - 01 - 08) 说明书第67段	3-4, 11-12, 18-19
类型*	引用文件, 必要时, 指明相关段落	相关的权利要求																								
X	CN 102929606 A (曙光信息产业北京有限公司) 2013年 2月 13日 (2013 - 02 - 13) 说明书第3、15、19-24段, 图2	1-2, 5-10, 13-17, 20																								
Y	CN 102929606 A (曙光信息产业北京有限公司) 2013年 2月 13日 (2013 - 02 - 13) 说明书第3、15、19-24段, 图2	3-4, 11-12, 18-19																								
X	CN 110245031 A (北京百度网讯科技有限公司) 2019年 9月 17日 (2019 - 09 - 17) 说明书第80-85、107、151、157段	1-2, 5-10, 13-17, 20																								
Y	CN 110245031 A (北京百度网讯科技有限公司) 2019年 9月 17日 (2019 - 09 - 17) 说明书第80-85、107、151、157段	3-4, 11-12, 18-19																								
X	CN 110489093 A (鹏城实验室) 2019年 11月 22日 (2019 - 11 - 22) 说明书第27、34-36段	1-2, 5-10, 13-17, 20																								
Y	CN 110489093 A (鹏城实验室) 2019年 11月 22日 (2019 - 11 - 22) 说明书第27、34-36段	3-4, 11-12, 18-19																								
Y	CN 109165055 A (百度在线网络技术北京有限公司) 2019年 1月 8日 (2019 - 01 - 08) 说明书第67段	3-4, 11-12, 18-19																								
<p><input checked="" type="checkbox"/> 其余文件在C栏的续页中列出。 <input checked="" type="checkbox"/> 见同族专利附件。</p>																										
<p>* 引用文件的具体类型:</p> <p>“A” 认为不特别相关的表示了现有技术一般状态的文件</p> <p>“E” 在国际申请日的当天或之后公布的在先申请或专利</p> <p>“L” 可能对优先权要求构成怀疑的文件, 或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件(如具体说明的)</p> <p>“O” 涉及口头公开、使用、展览或其他方式公开的文件</p> <p>“P” 公布日先于国际申请日但迟于所要求的优先权日的文件</p> <p>“T” 在申请日或优先权日之后公布, 与申请不相抵触, 但为了解发明之理论或原理的在后文件</p> <p>“X” 特别相关的文件, 单独考虑该文件, 认定要求保护的发明不是新颖的或不具有创造性</p> <p>“Y” 特别相关的文件, 当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时, 要求保护的发明不具有创造性</p> <p>“&” 同族专利的文件</p>																										
<p>国际检索实际完成的日期</p> <p>2021年 4月 16日</p>		<p>国际检索报告邮寄日期</p> <p>2021年 4月 28日</p>																								
<p>ISA/CN的名称和邮寄地址</p> <p>中国国家知识产权局(ISA/CN) 中国北京市海淀区蓟门桥西土城路6号 100088</p> <p>传真号 (86-10)62019451</p>		<p>授权官员</p> <p>梁静静</p> <p>电话号码 86-(20)-28958273</p>																								

C. 相关文件

类型*	引用文件, 必要时, 指明相关段落	相关的权利要求
A	US 2018357552 A1 (BONSAI AI INC) 2018年 12月 13日 (2018 - 12 - 13) 全文	1-20

国际检索报告
关于同族专利的信息

国际申请号

PCT/CN2020/125075

检索报告引用的专利文件			公布日 (年/月/日)	同族专利	公布日 (年/月/日)
CN	102929606	A	2013年 2月 13日	无	
CN	110245031	A	2019年 9月 17日	无	
CN	110489093	A	2019年 11月 22日	无	
CN	109165055	A	2019年 1月 8日	无	
US	2018357552	A1	2018年 12月 13日	无	