依瞳人工智能开放平台

产品白皮书

依瞳科技（深圳）有限公司

2020.6.30

目 录

1 概述

1.1 产品概述

依瞳人工智能开放平台，具有先进的集群管理、分布式计算、大规模数据存储与管理、超高速网络技术，充分释放 CPU/GPU/NPU 服务器集群的算力，兼容私有云、公有云，赋能 AI 开发人员进行超大规模人工智能模型训练与评估，同时也满足用户高速大规模数据处理的需求。

平台融合了 Tensorflow、PyTorch、MindSpore、Caffe、MxNet、CNTK 等开源深度学习框架，提供从数据预处理、数据标注、模型训练、模型超参调优、模型部署等一站式开发环境，方便 AI 开发者/研究人员快速搭建人工智能开发环境，开展 AI 开发应用。平台具备良好的计算资源管理能力，在监控模块基础上搭建预警模块，将平台异常实时通知管理员，提升平台的预警效率及安全性能。

平台底层采用更轻量级的虚拟化技术，如 Docker 容器，将任何一个或多个应用程序封装起来，并为容器提供标准的管理接口，使得每个容器之间互相隔离、互不影响。对部署容器化的应用，采用 Kubernetes 集群技术，进行自动化部署、规划、更新和维护，避免运维人员进行复杂的手工配置和处理，从而提高效率，降低成本。

1.2 术语解释

序号	术语名称	术语定义

1.	模型	机器学习训练完的结果（包括网络结构、参数等）。
2.	数据预处理	数据预处理是指在主要的处理以前对数据进行的一些处理。处理有多种方法：数据清理，数据集成，数据变换，数据归约等。
3.	数据标注	将原始数据通过某种工具或者方法，进行分类、框图等多种形态的标注。
4.	特征工程	特征工程指的是把原始数据转变为模型的训练数据的过程，它的目的就是获取更好的训练数据特征，最大限度地从原始数据中提取特征以供算法和模型使用。
5.	超参数	在机器学习的中，超参数是在开始学习过程之前设置值的参数，而不是通过训练得到的参数数据。通常情况下，需要对超参数进行优化，选择一组最优超参数，以提高学习的性能和效果。
6.	TensorFlow	TensorFlow 由 Google 大脑主导开发，是一个分布式系统上的大规模深度学习框架。移植性好，可以运行在移动设备上，并支持分布式多机多卡训练，支持多种深度学习模型。
7.	MindSpore	MindSpore 是端边云全场景按需协同的华为自研 AI 计算框架，提供全场景统一 API，为全场景 AI 的模型开发、模型运行、模型部署提供端到端能力。
8.	PyTorch	不同于 TensorFlow，PyTorch 采用动态计算图的方式，并提供良好的 Python 接口，代码简单灵活。
9.	Caffe	Caffe 是一个兼具表达性、速度和思维模块化的深度学习框架。由伯克利人工智能研究小组和伯克利视觉和学习中心开发。
10.	MXNet	MXNet 是一个深度学习框架，旨在提高效率和灵活性。MXNet 的核心是一个动态依赖调度程序，可以动态地自动并行化符号和命令操作。

11.	CNTK	CNTK 是微软出品的一个开源的深度学习工具包，可以运行在 CPU 上，也可以运行在 GPU 上。
12.	Jupyter Notebook	Jupyter Notebook 是一个基于 Web 的交互式计算环境，支持运行多种编程语言。平台支持使用 Jupyter NoteBook 的方式进行算法代码编写，模型训练任务的提交，以及结果查看等操作。
13.	Spark	Spark 是加州大学伯克利分校 AMP 实验室开发的通用内存并行计算框架，专为大规模数据处理而设计的快速通用的计算引擎。
14.	Kubernets	简称 K8S，是一个开源的，用于管理云平台中多个主机上的容器化的应用，目标是让部署容器化的应用简单并且高效，Kubernetes 提供了应用部署，规划，更新，维护的一种机制。
15.	Pod	Pod 是 Kubernetes 中能够创建和部署的最小单元，是 Kubernetes 集群中的一个应用实例。
16.	Docker	Docker 是一个开源的应用容器引擎，让开发者可以打包他们的应用以及依赖包到一个可移植的镜像中，然后发布到任何流行的 Linux 或 Windows 机器上，也可以实现虚拟化。
17.	Prometheus	Prometheus 是一套开源的系统监控和报警工具，可实现包括监控主机和容器、服务发现、警报管理，以及 Kubernetes 和运行其上的应用程序的监控。
18.	collectd	一个守护(daemon)进程，用来定期收集系统和应用程序的性能指标，同时提供了机制，以不同的方式来存储这些指标值。
19.	VC	Virtual Cluster 虚拟集群，对物理集群内所有 AI 计算芯片进行分组管理，每一个组就是一个虚拟集群。

2 平台优势

2.1 一站式开发平台

依瞳人工智能开放平台从 AI 开发应用角度出发，基于国内外主流深度学习框架，提供从数据处理->模型开发->模型调优->模型部署->模型发布的一站式开发流程，方便企事业 AI 开发人员/科院机构快速开展人工智能应用开发。

2.2 异构集群

依瞳人工智能平台除了支持 X86-GPU 服务器外，同时兼容国内主流服务器厂商如华为基于 ARM 架构的鲲鹏服务器，搭载昇腾 910 训练芯片，构建异构人工智能开发应用平台。

2.3 并行分布式模型训练

平台支持集成 TensorFlow、PyTorch、MindSpore、Caffe、MXNet、CNTK、Keras 等多种主流深度学习框架，支持 GPU/NPU 训练加速，方便 AI 开发者开展多机多卡并行分布式模型训练，充分利用集群算力资源，提升模型开发效率。

2.4 Spark 分布式计算

针对机器学习中需要对数据进行频繁迭代计算或多次操作特定数据集的场景，平台支持集成 Spark 计算引擎，方便开发者进行大数据分布式计算。

2.5 灵活部署

平台可根据用户具体使用场景，灵活支持私有化/定制化的部署方案，也可支持对接市面上主流公有云环境进行部署。

2.6 贴合 AI 开发者习惯

平台支持主流深度学习框架的集成，如 TensorFlow、PyTorch、MindSpore、CNTK、Caffe 等，同时以插件化的方式集成多种开发工具，如 Jupyter Notebook、VS Code、Anaconda、Colab 等，方便 AI 开发者快速开展人工智能开发应用。

3 平台架构设计

依瞳人工智能开放平台意在为 AI 开发者提供一套便捷的、一站式的 AI 开发应用环境，用户无需关注底层具体的硬件资源、集群组件、各开发组件之间兼容等问题，使用者可以快速构建超大规模应用集群，充分释放 GPU/NPU 运算能力，开展超大规模人工智能模型的开发应用，同时平台的高扩展性、高灵活性，支持对接主流公有云或私有化环境部署，大大降低二次开发及大规模部署成本，提升开发效率。

平台总体架构设计如下图 1 所示，

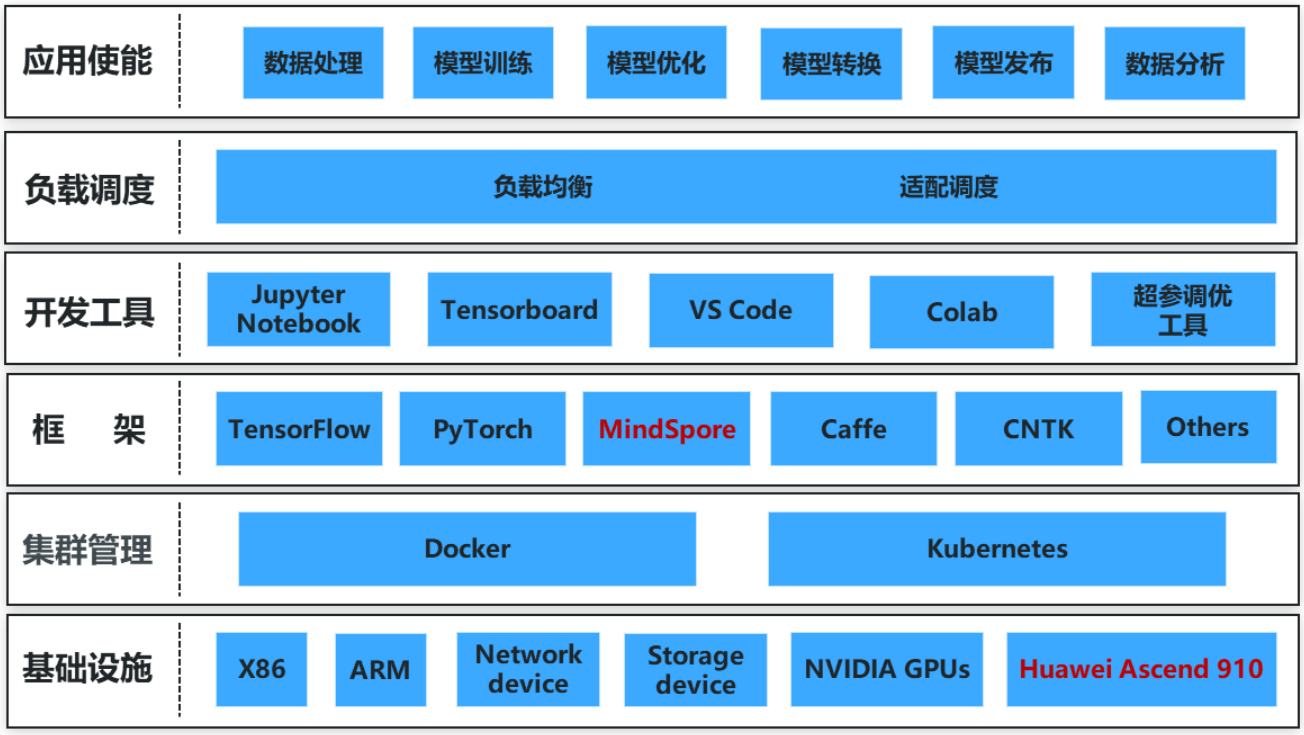

图 1 依瞳人工智能开放平台设计架构图

平台基础设施层采用 Docker 容器技术对算力资源进行池化，通过 Kubernetes 进行整体的资源管理、资源分配、任务运行、状态监控等，同时平台集成了主流深度学习框架、AI 开发工具等多种开源组件，通过数据集预处理、模型训练、模型优化、模型发布等服务，灵活开展 AI 开发应用。

- 支持底层硬件资源异构化

平台可以较好兼容底层 X86/ARM 不同架构的服务器，除了支持 NVIDIA 系列 GPU 外，可以同时兼容 HUAWEI 昇腾系列训练芯片 Ascend910 接入管理，方便用户构建异构集群，也丰富 AI 开发人员对于人工智能底层训练芯片的选择多样性。

- 集群资源管理

平台基于 Kubernets 实现大规模服务器的资源管理，包含资源分配、弹性伸缩、任务调度、资源监控、运维告警等功能，提升资源利用率，为用户提供多集群、多租户的资源管理能力。平台支持 Docker 容器技术，实现应用运行环境的细粒度划分、多租户隔离，让开发者可以快速构建可随时运行的容器化应用程序，简化应用的管理和部署。

- 兼容主流深度学习框架

平台兼容主流国深度学习框架，如 TensorFlow、MindSpore、PyTorch、Caffe、Keras、MxNet、CNTK 等，用户无需考虑平台与框架之间的兼容性问题，可根据自身的开发习惯自主选择开发框架。

- 拥抱开源社区，合理选择开发组件

平台设计之初充分考虑开发者的使用习惯，合理选择 AI 开发工具，如 Jupyter Notebook、Tensorboad、Colab、VS Code、超参调优工具等，用户可以快速上手使用，开展人工智能开发应用。

- 负载调度

平台原生支持深度学习开发应用，具备大规模任务的负载均衡管理、灵活调度，大幅提升集群资源利用率。

- 丰富的 AI 业务应用功能

平台提供了从数据预处理、数据标注、模型训练、超参调优、模型转换、模型发布等一站式开发应用功能，解决用户从原始数据到开发模型、落地使用的全流程需求。平台同时集成 Spark 分布式数据计算引擎，用户可根据具体使用场景灵活开展数据分析类应用。

4 一站式 AI 开发功能介绍

依瞳人工智能开放平台为 AI 开发人员提供从数据集管理（数据预处理）、模型开发、模型超参调优、模型部署、推理预测等全流程、一站式服务，快速打造人工智能开发业务。

4.1 数据集管理

为了灵活处理和使用数据集的差异性，支持数据集的版本管理。根据数据集不同容量大小，可以通过网页或后台进行上传，同时支持对数据集的删除、修改、下载等功能。

4.2 模型开发

4.2.1 多种开发模式

平台考虑不同开发者的使用习惯，支持多种开发模式，如 SSH 命令行模式、Jupyter Notebook 交互式模式，方便 AI 开发者自由选择。开发人员可以使用 Notebook 进行算法代码的编写，模型训练任务的提交，以及结果查看等操作，帮助提升开发效率。

平台支持诸如 Jupyter Notebook 等 IDE 开发工具以插件的形式集成进来使用，对于 AI 开发者具有较大的灵活性。

4.2.2 并行分布式模型训练

用户应用场景需要针对超大规模数据进行并行训练时，平台多机多卡并行分布式训练的支持，无论是对于 GPU 还是 NPU 的算力需求，根据异构集群基础算力资源，都能很好的灵活调度适配，释放算力，显著缩短模型开发所需时间。

4.2.3 模型训练可视化

为了更好的管理、调试和优化神经网络的训练过程，平台集成了 Tensorboard 可视化工具，支持训练过程中实时分析模型文件、日记文件的输出，直观的展示模型计算图结构、训练精度、损失值等参数，可以更好的可视化神经网络模型训练过程中各种指标的变化趋势，直观地了解神经网络的训练情况。

4.3 模型超参调优

机器学习模型开发过程中，非常重要的一环是模型的探索以及模型参数的优化，需要开发人员投入较大的精力、时间及算力支持。依瞳人工智能平台支持高效的模型参数自动调优工具，支持多种 AutoML 算法，且原生支持多机分布式调度，加速超参调优，省去编程和参数调优的冗繁重复工作。

模型参数自动调优工具适配主流 AI 框架，同时支持部分基于基于 GBDT 的算法，如 Scikit-learn、XGBoost、lightGBM 等，帮助 AI 开发者自动的进行特征工程探索、神经网络架构搜索、超参调优以及模型压缩等工作，且工具支持可视化 UI 界面，能够让使用者在整个实验过程中获得对训练结果的直观理解，便于分析。

4.4 模型试验管理

用户新建一个实验项目后，可以进行多次实验。每次实验时，可以上传所需要的不同版本的数据集和代码，实验完成后，生成对应的模型文件和日志文件，并且可以进行预览和下载。

4.5 模型部署

训练后的模型，平台支持多种灵活的部署方式，满足云-端-边多场景不同的部署需求，解决了用户模型部署使用的最后一公里问题。

4.6 模型推理

平台支持对所有推理任务列表进行管理。当创建推理任务后，可以用指定的模型进行推理。

5 平台功能介绍

5.1 用户管理

用户支持三种登录方式，账号密码、微信账号、微软账号。用户是基于角色的访问控制进行管理的，并且对用户可以进行分组管理。对用户和用户组赋予不同的角色，不同的角色可以拥有不同的权限。

5.2 资源管理

依瞳人工智能开放平台监控管理当前集群内的硬件资源信息，包含单节点、多节点及整个异构集群的计算资源（CPU/内存）、算力资源（GPU/NPU）、存储资源、网络资源等，当某节点资源发生变化、或集群管理员执行在线伸缩操作时，平台实时感知资源变化，并更新至资源管理中。平台采用分布式架构设计，为每个节点提供相应的管理服务，任何单一节点故障都不会引起整个平台的管理中断。

在多用户管理的前提下，平台资源可提供按用户分级分权的管理，针对不同的平台用户，可以管理平台分配的对应资源，并且针对每种资源对象，可以设置更加精细化的权限管理和配额控制，为平台中的多租户使用底层资源提供更高的灵活性。

5.3 在线扩/缩容

平台基于 Kubernets 作集群资源调度管理，支持 Node 节点在线扩容/缩容。当实际生产作业环境中遇到业务需求增大，而现有硬件资源无法满足业务使用时，需要新增服务器节点以提升 Pod 数量来应对更大的请求，根据 K8S 提供的原生服务，在不影响现有应用的前

提下，支持对集群进行在线水平扩展以达到扩容的效果，应用 Pod 可平滑迁移至新增 Node 节点。

当集群中某节点出现故障无法继续满足业务使用时，应用 Pod 会迁移至集群中的其余 Node，集群告警服务会通知管理员对该节点进行缩容操作，以保障集群健康运行。

5.4 智能调度

平台初始化时，通过 device-plugin 加载识别当前集群中的各类硬件资源，区分 GPU、NPU 不同底层芯片设备，并通知系统资源管理模块统一监控维护。当使用人员需要开展深度学习训练任务，配置不同的训练芯片时，平台会根据集群中各个节点资源的空闲情况，智能调度训练任务的请求至不同节点的训练芯片资源中，以保障训练任务可以灵活开展。

5.5 平台兼容性

平台具有良好的兼容性。底层硬件层面，支持底层硬件资源异构化，适配 X86/ARM 架构服务器，同时兼容 NVIDIA GPU 及 HUAWEI Ascend910（NPU）等不同的训练芯片资源。软件框架层面，兼容主流的深度学习框架，如 TensorFlow、MindSpore、PyTorch、Caffe、MxNet 等；贴合 AI 开发者习惯，支持不同 AI 开发工具以插件化形式与平台对接，如 Jupyter Notebook、Visual Studio Code、Anaconda 等，开发人员可以根据使用习惯来选择 IDE 工具。

5.6 安全特性

平台具备以下几种安全特性，保障集群用户的使用安全、数据安全：

1) 容器应用与其所属 Pod 所在的 Node 节点隔离，某个容器应用出问题不会对整个平台使用造成影响；

2) 基于角色的访问控制，平台可以控制哪些用户具体可执行哪些操作以及他们的权限范围（权限管理），区分普通用户及管理员；

3) 使用命名空间（虚拟集群 VC）进行逻辑隔离，可以按 VC 划分整个集群资源，结合用户管理，当前 VC 内的用户不具备访问其它 VC 内的私有资源，可用于跨多个团队或项目的多用户环境资源、权限管理。

除以上主要安全策略外，平台还支持可配置集群网络策略、设置 Pod 安全策略、控制对敏感端口的网络访问等，确保不同用户在平台上进行深度学习开发应用的安全使用。

5.7 镜像管理

平台支持从 Docker Hub 中直接拉取镜像文件，或者通过配置从国内各大公有云服务商的镜像仓库中拉取镜像文件来部署应用。而对于需要在私有化环境中部署及使用的用户时，平台支持搭建企业内部私有镜像仓库，可以将自制镜像或外部镜像导入到本地的私有镜像仓库中，支持镜像的权限及版本配置，实现企业镜像模板的集中管理、便捷配置使用。

5.8 日记管理

平台支持 Pod/容器应用日记、Node 节点日记、集群状态日记等三个层级完整的日记管理功能，便于集群在出现故障或 bug 时，提供较完备的日记支持分析定位。

5.9 监控告警

平台通过 Prometheus 与 collectd 基于时间序列来完成整个集群的运行状态信息采集及监控，实时监控各个 Node 节点的系统状况、CPU、内存、存储 IO、网络 IO、GPU、NPU 等各参数指标的变化，同时 Prometheus 云原生技术的支持，可对 Pod 中各容器的应用运行状态、资源使用情况监控，查找并分析集群性能瓶颈，预测未来整体集群的负载情况（容量规划）等。

平台支持集群中告警规则的设置，当某个监控指标达到预警级别时（如 Node 内存使用率达 70%），可通过 email 将告警信息发送给平台管理人员，及时采取相关措施处理告警，避免影响整体集群的正常使用。

6 典型组网推荐

6.1 GPU 集群

GPU 集群的典型组网推荐，一台服务器（CPU）作为集群的 master 节点，两台服务器（GPU）作为集群的两个 worker 节点，再加上一台存储服务器。

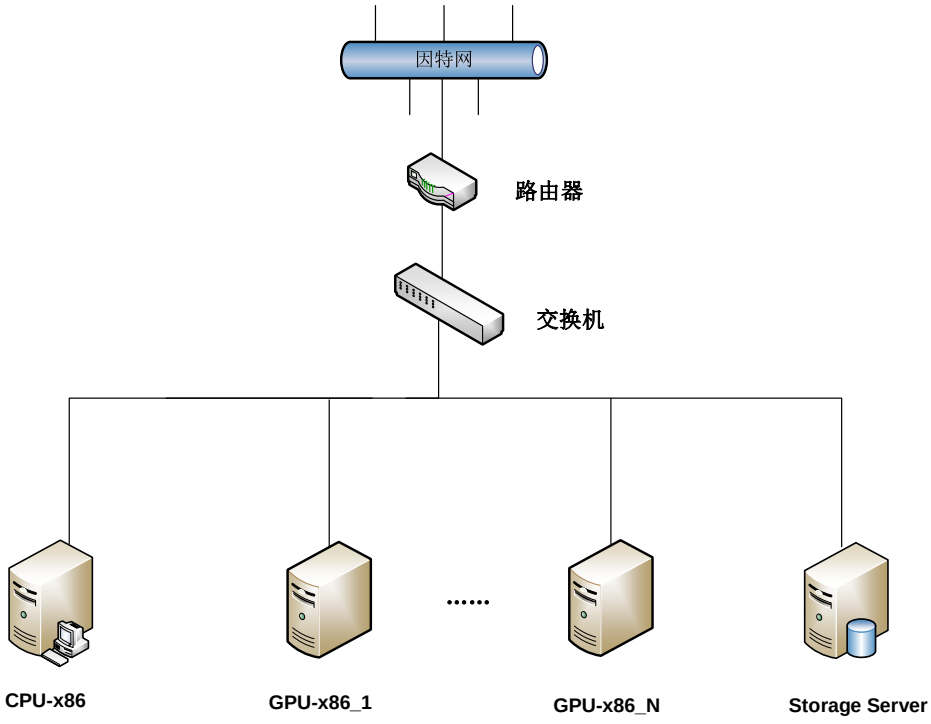

图 2 GPU 集群组网拓扑图

6.2 异构集群

异构集群的典型组网推荐，一台服务器（CPU）作为集群的 master 节点，服务器（GPU/NPU）作为集群的 worker 节点，再加上一台存储服务器。

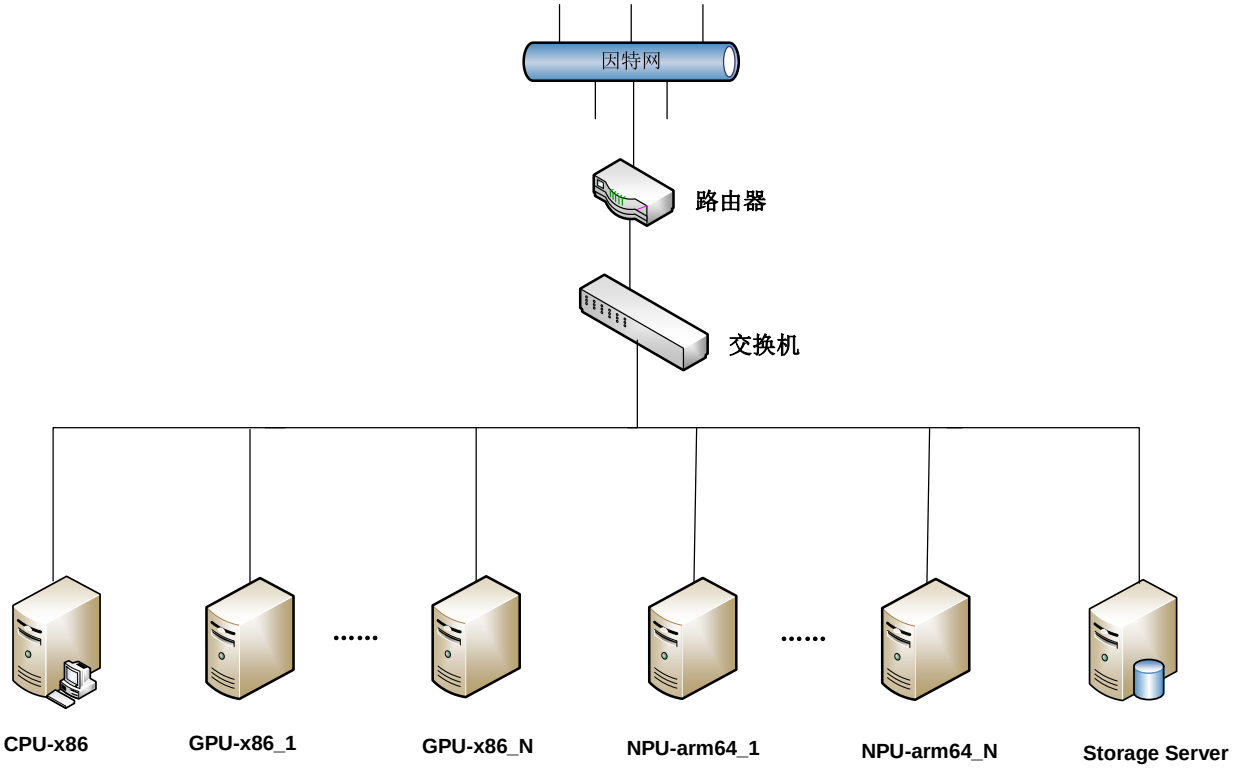

图 3 异构集群组网拓扑图