



中华人民共和国国家标准化指导性技术文件

GB/Z 222—2026

人工智能 模型即服务(MaaS)性能规范

Artificial intelligence—Model as a service(MaaS) performance specification

2026-08-27 发布

国家市场监督管理总局 发布
国家标准化管理委员会

目 次

前言 III

1 范围 1

2 规范性引用文件 1

3 术语和定义 1

4 缩略语 1

5 MaaS 性能参数框架 1

 5.1 概述 1

 5.2 性能参数 2

 5.2.1 模型开发性能参数 2

 5.2.2 模型服务性能参数 3

 5.2.3 AI 应用开发性能参数 3

6 参数测试方法 3

 6.1 模型开发 3

 6.1.1 训练 3

 6.1.2 推理 6

 6.2 模型服务 8

 6.2.1 模型服务时延 8

 6.2.2 模型服务吞吐率 8

 6.2.3 模型服务请求成功率 9

 6.2.4 生成式语言模型首 token 时延 9

 6.2.5 生成式语言模型 token 吞吐率 10

 6.3 AI 应用开发 10

 6.3.1 AI 应用服务时延 10

 6.3.2 AI 应用服务吞吐率 11

 6.3.3 AI 应用服务请求成功率 11

 6.3.4 生成式 AI 应用服务首 token 时延 12

 6.3.5 生成式 AI 应用服务 token 吞吐率 12

前 言

本文件为规范类指导性技术文件。

本文件按照 GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由全国信息技术标准化技术委员会(SAC/TC 28)提出并归口。

本文件起草单位：中国电子技术标准化研究院、北京百度网讯科技有限公司、华为云计算技术有限公司、上海商汤智能科技有限公司、浪潮电子信息产业股份有限公司、浙江大华技术股份有限公司、浪潮通信信息系统有限公司、百度云智(北京)科技有限公司、中国移动通信集团有限公司、阿里云计算有限公司、普元信息技术股份有限公司、浪潮软件科技有限公司、中移(苏州)软件技术有限公司、中电信数智科技有限公司、中国移动通信集团设计院有限公司、中国科学院信息工程研究所、中国南方电网有限责任公司超高压输电公司、中移(杭州)信息技术有限公司、杭州海康威视数字技术股份有限公司、山东省人工智能研究院、上海文镭信息科技有限公司、北京神舟航天软件技术股份有限公司、国家能源投资集团有限责任公司信息技术分公司、广电运通集团股份有限公司、深信服科技股份有限公司。

本文件主要起草人：谢永康、鲍薇、杨语澈、李景秋、徐洋、曹晓琦、邸贺亮、吴庚、杨雨泽、黎世勇、施恩、郑子木、李旭东、孔维生、肖红梅、张婷、曹汐、刘增志、彭骏涛、李照川、杨灿坤、仲凯韬、蔡洛维、刘井强、周震震、付涛、谢小燕、王珂琛、杨经纬、张悦、崔洪志、张天霖、密伟、张传刚、陈欢、梁秉豪、周彩红、秦日臻、潘风文、侯金磊、徐天适、舒明雷、陈利明、单珂、高宇、徐金涛。

人工智能 模型即服务(MaaS)性能规范

1 范围

本文件确立了模型即服务(MaaS)性能参数框架,并描述了对应的参数测试方法。
本文件适用于模型即服务(MaaS)的性能参数测试。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中,注日期的引用文件,仅该日期对应的版本适用于本文件;不注日期的引用文件,其最新版本(包括所有的修改单)适用于本文件。

GB/Z 221—2026 人工智能 模型即服务(MaaS)参考架构

3 术语和定义

GB/Z 221—2026 界定的以及下列术语和定义适用于本文件。

3.1

模型即服务 model as a service

提供模型开发、使用和管理能力及所需资源且可规模化复制的服务模式。

4 缩略语

下列缩略语适用于本文件。

AI:人工智能(Artificial Intelligence)

GPU:图形处理单元(Graphics Processing Unit)

I/O:输入/输出(Input/Output)

MaaS:模型即服务(Model as a Service)

TPU:张量处理单元(Tensor Processing Unit)

XLA:加速线性代数(Accelerated Linear Algebra)

5 MaaS 性能参数框架

5.1 概述

MaaS 性能影响任务响应速度、业务生产效率、用户使用体验、模型使用成本等,是评估 MaaS 的重要依据。本文件定义了 MaaS 性能参数框架,如图 1 所示。主要包括:

- 模型开发性能,包括训练性能和推理性能等方面的参数,具体见 5.2;
- 模型服务性能,包括模型服务、生成式语言模型服务等方面参数,具体见 5.2;
- AI 应用开发性能,包括 AI 应用服务、生成式 AI 应用服务等方面的指标,具体见 5.2。

相应参数的一般性测试方法在第 6 章给出。

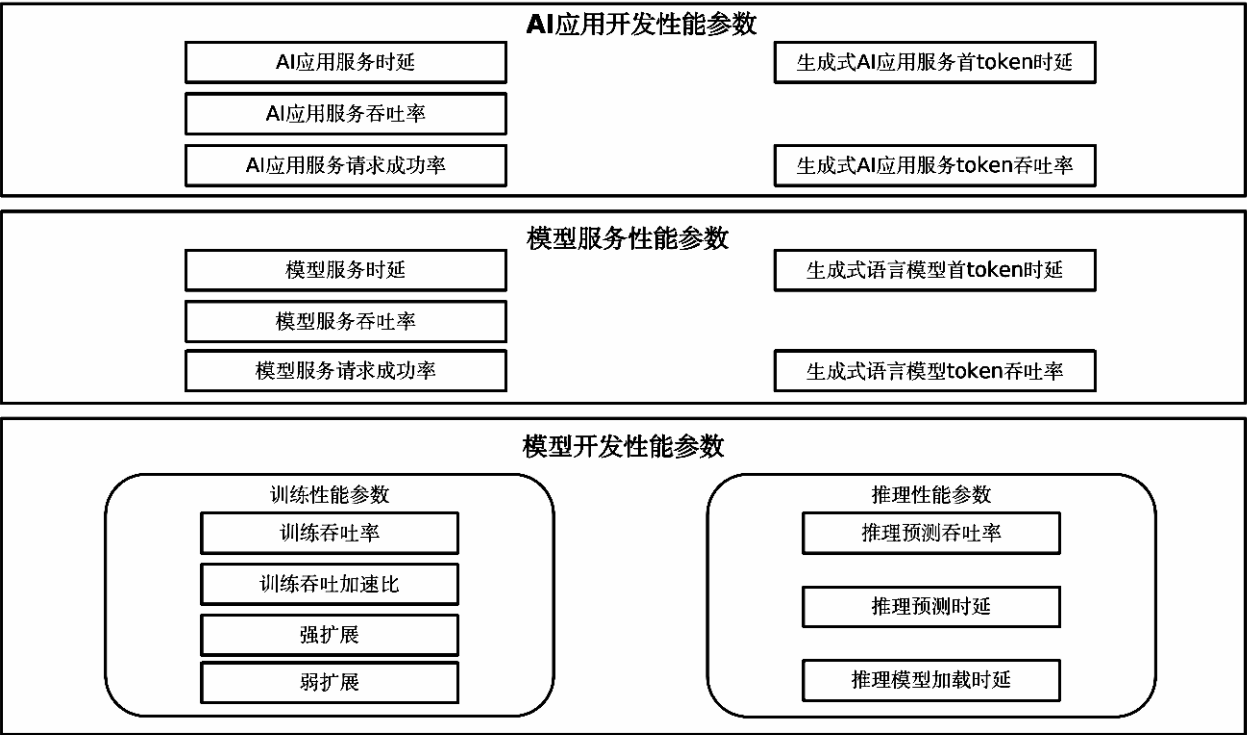


图 1 MaaS 性能参数框架

5.2 性能参数

5.2.1 模型开发性能参数

MaaS 模型开发性能参数包括训练性能、推理性能，具体参数见表 1。

表 1 模型开发性能参数

序号	参数名称	参数描述
训练性能		
1	训练吞吐率	单位时间内模型训练处理的事务数量,单位每秒事务数(例如样本/秒或者 token/s)
2	训练吞吐加速比	在相同资源环境下,不同超参数配置下的训练吞吐率与默认配置下训练吞吐率的比值。超参数配置包括批大小(BatchSize)、数据精度、GPU 是否开启 XLA 等
3	弱扩展	在训练处理数据总量一定时,人工智能加速处理器增加前后的训练吞吐率之比
4	强扩展	单位人工智能加速处理器的训练处理数据量一定时,训练处理数据总量、人工智能加速处理器同时增加前后的训练吞吐率之比
推理性能		
1	推理预测吞吐率	单位时间内模型推理的事务数量,单位每秒事务数
2	推理预测时延	模型开发推理单个实例的延迟时间,单位为毫秒(ms)
3	推理模型加载时延	当模型参数规模一定时,模型开发将模型从存储文件到加载为可用服务所需要的时间,单位为秒(s)



5.2.2 模型服务性能参数

模型服务性能参数见表 2。

表 2 模型服务性能参数

序号	参数名称	参数描述
1	模型服务时延	模型服务处理单个请求所需时间,单位为毫秒(ms),包括平均响应时间、90 分位响应时间、95 分位响应时间和 99 分位响应时间等
2	模型服务吞吐率	单位时间处理的模型服务事务数量,单位每秒事务数
3	模型服务请求成功率	模型服务成功请求数量占总请求数量的百分率
4	生成式语言模型首 token 时延	生成式语言模型服务从接收到单个请求到生成并返回第一个 token 所需要的时间,单位为毫秒(ms),包括平均响应时间、90 分位响应时间、95 分位响应时间和 99 分位响应时间等
5	生成式语言模型 token 吞吐率	生成式语言模型服务单位时间生成的 token 数量,单位每秒 token 数量

5.2.3 AI 应用开发性能参数

AI 应用开发性能参数见表 3。

表 3 AI 应用开发性能参数

序号	参数名称	参数描述
1	AI 应用服务时延	用户创建的 AI 应用服务,单次请求所需时间,单位为秒(s),包括平均响应时间、90 分位响应时间、95 分位响应时间、99 分位响应时间
2	AI 应用服务吞吐率	用户创建的 AI 应用服务在单位时间内能够处理的事务数量,单位每秒事务数
3	AI 应用服务请求成功率	AI 应用服务成功请求数量占总请求数量的百分率
4	生成式 AI 应用服务首 token 时延	生成式 AI 应用服务从系统接收到用户请求到生成并返回第一个 token 所需要的时间,单位为秒(s),包括平均响应时间、90 分位响应时间、95 分位响应时间、99 分位响应时间
5	生成式 AI 应用服务 token 吞吐率	生成式 AI 应用服务单位时间生成的 token 数量,单位每秒 token 数量

6 参数测试方法

6.1 模型开发

6.1.1 训练

6.1.1.1 训练吞吐率

模型开发训练吞吐率测试按表 4 的规定进行。

表 4 模型开发训练吞吐率测试方法

测试项目	模型开发训练吞吐率
测试对象	模型开发平台
测试数据	依据模型开发平台训练测试任务,选择对应的测试数据集,如图片、文本、结构化数据等
测试工具	资源监控工具,如 Prometheus
测试环境配置	安装模型开发平台训练测试任务相关依赖,包括深度学习框架
测试步骤	a) 确定训练测试任务场景,如训练图像识别模型、广告点击率预测模型、文本合成语音模型; b) 设置深度学习框架,如 TensorFlow、PyTorch、PaddlePaddle 等; c) 设置人工智能加速处理器类别和数量,如 1GPU 卡、4GPU 卡、8GPU 卡、1TPU 卡、4TPU 卡等; d) 设置超参数,如 BatchSize、数据精度、GPU 是否开启 XLA 等; e) 使用资源监控工具,监控被测模型服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; f) 导入测试数据集; g) 创建模型开发平台训练测试任务; h) 执行模型开发平台训练测试任务; i) 记录模型开发平台训练吞吐率
示例: 图像识别测试数据如 ImageNet ILSVRC2012 数据集。	

6.1.1.2 训练吞吐加速比

模型开发训练吞吐加速比测试按表 5 的规定进行。

表 5 模型开发训练吞吐加速比测试方法

测试项目	模型开发训练吞吐加速比
测试对象	模型开发平台
测试数据	依据模型开发平台训练测试任务,选择对应的测试数据集,如图片、文本、结构化数据等
测试工具	资源监控工具,如 Prometheus
测试环境配置	安装模型开发平台训练测试任务相关依赖,包括深度学习框架
测试步骤	a) 确定训练测试任务场景,如训练图像识别模型、广告点击率预测模型、文本合成语音模型; b) 设置深度学习框架,如 TensorFlow、PyTorch、PaddlePaddle 等; c) 设置人工智能加速处理器类别和数量,如 1GPU 卡、4GPU 卡、8GPU 卡等; d) 设置参数,如 BatchSize、数据精度、GPU 是否开启 XLA 等; e) 使用资源监控工具,监控被测模型服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; f) 导入测试数据集; g) 创建模型开发平台训练测试任务; h) 执行模型开发平台训练测试任务; i) 记录模型开发平台训练吞吐率; j) 深度学习模型、框架、人工智能加速处理器类别和数量与步骤 a)、b)、c)的设置保持一致; k) 结合测试任务,重新设置 BatchSize、数据精度、GPU 是否开启 XLA 等参数,参数值与步骤 d)不同; l) 导入测试数据集,测试数据集与步骤 e)保持一致; m) 创建模型开发平台训练测试任务; n) 执行模型开发平台训练测试任务; o) 记录模型开发平台训练吞吐率; p) 计算不同参数配置下的模型开发平台训练吞吐加速比; q) 记录不同参数配置下的模型开发平台训练吞吐加速比

6.1.1.3 弱扩展

模型开发弱扩展测试按表 6 的规定进行。

表 6 模型开发弱扩展测试方法

测试项目	模型开发弱扩展
测试对象	模型开发平台
测试数据	依据模型开发平台训练测试任务,选择对应的测试数据集,如图片、文本、结构化数据等
测试工具	资源监控工具,如 Prometheus
测试环境配置	安装模型开发平台训练测试任务相关依赖,包括深度学习框架
测试步骤	a) 确定训练测试任务场景,如训练图像识别模型、广告点击率预测模型、文本合成语音模型; b) 设置深度学习框架,如 TensorFlow、PyTorch、PaddlePaddle 等; c) 设置人工智能加速处理器类别和数量,类别如 GPU 卡等,数量为 1 卡; d) 设置 BatchSize、数据精度、GPU 是否开启 XLA 等参数; e) 使用资源监控工具,监控被测模型服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; f) 导入测试数据集; g) 创建模型开发平台训练测试任务; h) 执行模型开发平台训练测试任务; i) 记录模型开发平台训练吞吐率; j) 结合测试任务,再次设置深度学习框架、模型,与步骤 a)、b)的设置保持一致; k) 设置人工智能加速处理器类别,和步骤 c)的设置保持一致; l) 设置人工智能加速处理器数量为多卡,如 4GPU 卡、8GPU 卡等; m) 设置 BatchSize、数据精度、GPU 是否开启 XLA 等参数,与步骤 d)的设置保持一致; n) 导入测试数据集,测试数据集与步骤 e)保持一致; o) 创建模型开发平台训练测试任务; p) 执行模型开发平台训练测试任务; q) 记录模型开发平台训练吞吐率; r) 计算在训练处理数据量一定时,计算资源增加前后的训练吞吐率之比,得出扩展能力的 Weak scaling; s) 记录模型开发平台 Weak scaling

6.1.1.4 强扩展

模型开发强扩展测试按表 7 的规定进行。

表 7 模型开发强扩展测试方法

测试项目	模型开发强扩展
测试对象	模型开发平台
测试数据	依据模型开发平台训练测试任务,选择对应的测试数据集,如图片、文本、结构化数据等
测试工具	资源监控工具,如 Prometheus
测试环境配置	安装模型开发平台训练测试任务相关依赖,包括深度学习框架

表 7 模型开发强扩展测试方法（续）

测试项目	模型开发强扩展
测试步骤	a) 确定训练测试任务场景,如训练图像识别模型、广告点击率预测模型、文本合成语音模型; b) 设置深度学习框架,如 TensorFlow、PyTorch、PaddlePaddle 等; c) 设置人工智能加速处理器类别和数量,类别如 GPU 卡等,数量为 1 卡; d) 设置 BatchSize、数据精度、GPU 是否开启 XLA 等参数; e) 导入测试数据集; f) 使用资源监控工具,监控被测模型服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; g) 创建模型开发平台训练测试任务; h) 执行模型开发平台训练测试任务; i) 记录模型开发平台训练吞吐率; j) 结合测试任务,再次设置深度学习框架、模型,与步骤 a)、b)的设置保持一致; k) 设置人工智能加速处理器类别,和步骤 c)的设置保持一致; l) 设置人工智能加速处理器数量为步骤 c)的 n 倍; m) 设置 BatchSize、数据精度、GPU 是否开启 XLA 等参数,与步骤 d)的设置保持一致; n) 设置资源参数,与步骤 d)的设置保持一致; o) 导入测试数据集,数据量为步骤 e)的 n 倍,n 值与步骤 k)一致; p) 创建模型开发平台训练测试任务; q) 执行模型开发平台训练测试任务; r) 记录模型开发平台训练吞吐率; s) 计算在单位人工智能加速处理器的训练处理数据量一定时,训练处理数据量和计算资源数量同时增加前后的训练吞吐率之比,得出强扩展指标; t) 记录模型开发平台强扩展指标

6.1.2 推理

6.1.2.1 推理预测吞吐率

模型开发推理预测吞吐率测试按表 8 的规定进行。

表 8 模型开发推理预测吞吐率测试方法

测试项目	模型开发推理预测吞吐率
测试对象	模型开发平台
测试数据	依据模型开发平台推理预测测试任务,选择对应的测试数据集,如图片、文本、结构化数据等
测试工具	资源监控工具,如 Prometheus
测试环境配置	安装模型开发平台推理预测测试任务相关依赖,包括深度学习框架
测试步骤	a) 选择测试的推理预测模型,如图像识别模型、广告点击率预测模型、文本合成语音模型; b) 设置数据精度、GPU 是否开启 XLA 等参数; c) 使用资源监控工具,监控被测模型服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; d) 导入测试数据; e) 创建模型推理预测任务; f) 执行模型推理预测任务; g) 记录模型推理预测吞吐率

6.1.2.2 推理预测时延

模型开发推理预测时延测试按表 9 的规定进行。

表 9 模型开发推理预测时延测试方法

测试项目	模型开发推理预测时延
测试对象	模型开发平台
测试数据	依据模型开发平台推理预测测试任务,选择对应的测试数据集,如图片、文本、结构化数据等
测试工具	资源监控工具,如 Prometheus
测试环境配置	安装模型开发平台推理预测测试任务相关依赖,包括深度学习框架
测试步骤	a) 选择测试的推理预测模型,如图像识别模型、广告点击率预测模型、文本合成语音模型; b) 设置数据精度、GPU 是否开启 XLA 等参数; c) 使用资源监控工具,监控被测模型服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; d) 导入测试数据; e) 创建模型推理预测任务; f) 执行模型推理预测任务; g) 记录模型推理预测平均响应时间、90 分位响应时间、95 分位响应时间和 99 分位响应时间

6.1.2.3 推理模型加载时延

模型开发推理模型加载时延测试按表 10 的规定进行。

表 10 模型开发推理模型加载时延测试方法

测试项目	模型开发推理模型加载时延
测试对象	模型开发平台
测试数据	依据模型开发平台推理预测测试任务,选择对应的测试数据集,如图片、文本、结构化数据等
测试工具	资源监控工具,如 Prometheus
测试环境配置	安装模型开发平台推理预测测试任务相关依赖,包括深度学习框架
测试步骤	a) 选择测试的推理预测模型,如图像识别模型、广告点击率预测模型、文本合成语音模型; b) 设置数据精度、GPU 是否开启 XLA 等参数; c) 使用资源监控工具,监控被测模型服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; d) 导入测试数据; e) 加载推理模型; f) 启动推理模型服务; g) 记录推理模型加载平均响应时间、90 分位响应时间、95 分位响应时间和 99 分位响应时间

6.2 模型服务

6.2.1 模型服务时延

模型服务时延测试按表 11 的规定进行。

表 11 模型服务时延测试方法

测试项目	模型服务时延
测试对象	模型服务
测试数据	根据待测试模型服务,选取图像或文本测试数据格式,准备文件大小分别为 1 KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据集
测试工具	人工智能负载测试工具,如 Locust(Python 运行环境);资源监控工具,如 Prometheus
测试环境配置	安装人工智能负载测试工具运行环境,如 Python;安装人工智能负载测试工具;安装资源监控工具
测试步骤	a) 获取待测试模型服务,发送请求确认服务可用; b) 使用资源监控工具,监控被测模型服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; c) 获取测试数据集,在测试工具导入测试数据; d) 启动人工智能负载测试工具,依次选取 1 KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据,运行模型服务测试请求,每项请求运行测试 30 min; e) 记录请求平均响应时间、90 分位响应时间、95 分位响应时间和 99 分位响应时间

6.2.2 模型服务吞吐量

模型服务吞吐量测试按表 12 的规定进行。

表 12 模型服务吞吐量测试方法

测试项目	模型服务吞吐量
测试对象	模型服务
测试数据	根据待测试模型服务,选取图像或文本测试数据格式,准备文件大小分别为 1 KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据集
测试工具	人工智能负载测试工具,如 Locust(Python 运行环境);资源监控工具,如 Prometheus
测试环境配置	安装人工智能负载测试工具运行环境,如 Python;安装人工智能负载测试工具;安装资源监控工具
测试步骤	a) 获取待测试模型服务,发送请求确认服务可用; b) 使用资源监控工具,监控被测模型服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; c) 获取测试数据集,在测试工具导入测试数据; d) 启动人工智能负载测试工具,依次选取 1 KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据,运行模型服务测试请求,每项请求运行测试 30 min; e) 记录请求平均吞吐率和最大吞吐率

6.2.3 模型服务请求成功率

模型服务请求成功率测试按表 13 的规定进行。

表 13 模型服务请求成功率测试方法

测试项目	模型服务请求成功率
测试对象	模型服务
测试数据	根据待测试模型服务,选取图像或文本测试数据格式,准备文件大小分别为 1 KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据集
测试工具	人工智能负载测试工具,如 Locust(Python 运行环境);资源监控工具,如 Prometheus
测试环境配置	安装人工智能负载测试工具运行环境,如 Python;安装人工智能负载测试工具;安装资源监控工具
测试步骤	a) 获取待测试模型服务,发送请求确认服务可用; b) 使用资源监控工具,监控被测模型服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; c) 获取测试数据集,在测试工具导入测试数据; d) 启动人工智能负载测试工具,依次选取 1KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据,运行模型服务测试请求,每项请求运行测试 30 min; e) 记录请求中成功返回的请求数量以及总请求数量; f) 计算模型服务请求成功率; g) 记录模型服务请求成功率

6.2.4 生成式语言模型首 token 时延

生成式语言模型首 token 时延测试按表 14 的规定进行。

表 14 生成式语言模型首 token 时延测试方法

测试项目	生成式语言模型首 token 时延
测试对象	生成式语言模型服务
测试数据	根据待测试生成式语言模型服务,选取文本测试数据格式,准备 token 数量分别为 100、500、1 000、10 000、100 000 的测试数据集
测试工具	人工智能负载测试工具,如 Locust(Python 运行环境);资源监控工具,如 Prometheus
测试环境配置	安装人工智能负载测试工具运行环境,如 Python;安装人工智能负载测试工具;安装资源监控工具
测试步骤	a) 获取待测试模型服务,发送请求确认服务可用; b) 使用资源监控工具,监控被测模型服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; c) 获取测试数据集,在测试工具导入测试数据; d) 启动人工智能负载测试工具,依次选取 100、500、1 000、10 000、100 000 的测试数据,运行模型服务测试请求,每项请求运行测试 30 min; e) 记录生成首 token 的平均响应时间、90 分位响应时间、95 分位响应时间和 99 分位响应时间

6.2.5 生成式语言模型 token 吞吐率

生成式语言模型 token 吞吐率测试按表 15 的规定进行。

表 15 生成式语言模型 token 吞吐率测试方法

测试项目	生成式语言模型 token 吞吐率
测试对象	生成式语言模型服务
测试数据	根据待测试模型服务,选取文本测试数据格式,准备 token 数量分别为 100、500、1 000、10 000、100 000 的测试数据集
测试工具	人工智能负载测试工具,如 Locust(Python 运行环境);资源监控工具,如 Prometheus
测试环境配置	安装人工智能负载测试工具运行环境,如 Python;安装人工智能负载测试工具;安装资源监控工具
测试步骤	a) 获取待测试模型服务,发送请求确认服务可用; b) 使用资源监控工具,监控被测模型服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; c) 获取测试数据集,在测试工具导入测试数据; d) 启动人工智能负载测试工具,依次选取 token 数量分别为 100、500、1 000、10 000、100 000 的测试数据,运行模型服务测试请求,每项请求运行测试 30 min; e) 记录 token 平均吞吐率和最大吞吐率

6.3 AI 应用开发

6.3.1 AI 应用服务时延

AI 应用服务时延测试按表 16 的规定进行。

表 16 AI 应用服务时延测试方法

测试项目	AI 应用服务时延
测试对象	AI 应用服务
测试数据	根据待测试 AI 应用服务,选取图像或文本测试数据格式,准备文件大小分别为 1 KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据集
测试工具	人工智能负载测试工具,如 Locust(Python 运行环境);资源监控工具,如 Prometheus
测试环境配置	安装人工智能负载测试工具运行环境,如 Python;安装人工智能负载测试工具;安装资源监控工具
测试步骤	a) 获取待测试 AI 应用服务,发送请求确认服务可用; b) 使用资源监控工具,监控被测 AI 应用服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; c) 获取测试数据集,在测试工具导入测试数据; d) 启动人工智能负载测试工具,依次选取 1 KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据,运行 AI 应用服务测试请求,每项请求运行测试 30 min; e) 记录请求平均响应时间、90 分位响应时间、95 分位响应时间和 99 分位响应时间

6.3.2 AI 应用服务吞吐量

AI 应用服务吞吐量测试按表 17 的规定进行。

表 17 AI 应用服务吞吐量测试方法

测试项目	AI 应用服务吞吐量
测试对象	AI 应用服务
测试数据	根据待测试 AI 应用服务,选取图像或文本测试数据格式,准备文件大小分别为 1 KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据集
测试工具	人工智能负载测试工具,如 Locust(Python 运行环境);资源监控工具,如 Prometheus
测试环境配置	安装人工智能负载测试工具运行环境,如 Python;安装人工智能负载测试工具;安装资源监控工具
测试步骤	a) 获取待测试 AI 应用服务,发送请求确认服务可用; b) 使用资源监控工具,监控被测 AI 应用服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; c) 获取测试数据集,在测试工具导入测试数据; d) 启动人工智能负载测试工具,依次选取 1 KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据,运行 AI 应用服务测试请求,每项请求运行测试 30 min; e) 记录请求平均吞吐率和最大吞吐率

6.3.3 AI 应用服务请求成功率

AI 应用服务请求成功率测试按表 18 的规定进行。

表 18 AI 应用服务请求成功率测试方法

测试项目	AI 应用服务请求成功率
测试对象	AI 应用服务
测试数据	根据待测试 AI 应用服务,选取图像或文本测试数据格式,准备文件大小分别为 1 KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据集
测试工具	人工智能负载测试工具,如 Locust(Python 运行环境);资源监控工具,如 Prometheus
测试环境配置	安装人工智能负载测试工具运行环境,如 Python;安装人工智能负载测试工具;安装资源监控工具
测试步骤	a) 获取待测试 AI 应用服务,发送请求确认服务可用; b) 使用资源监控工具,监控被测 AI 应用服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; c) 获取测试数据集,在测试工具导入测试数据; d) 启动人工智能负载测试工具,依次选取 1 KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据,运行 AI 应用服务测试请求,每项请求运行测试 30 min; e) 记录请求中成功返回的请求数量以及总请求数量; f) 计算 AI 应用服务请求成功率; g) 记录 AI 应用服务请求成功率

6.3.4 生成式 AI 应用服务首 token 时延

生成式 AI 应用服务首 token 时延测试按表 19 的规定进行。

表 19 生成式 AI 应用服务首 token 时延测试方法

测试项目	生成式 AI 应用服务首 token 时延
测试对象	生成式 AI 应用服务
测试数据	根据待测试生成式 AI 应用服务,选取图像或文本测试数据格式,准备文件大小分别为 1 KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据集
测试工具	人工智能负载测试工具,如 Locust(Python 运行环境);资源监控工具
测试环境配置	安装人工智能负载测试工具运行环境,如 Python;安装人工智能负载测试工具;安装资源监控工具
测试步骤	a) 获取待测试生成式 AI 应用服务,发送请求确认服务可用; b) 使用资源监控工具,监控被测 AI 应用服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; c) 获取测试数据集,在测试工具导入测试数据; d) 启动人工智能负载测试工具,依次选取 1 KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据,运行 AI 应用服务测试请求,每项请求运行测试 30 min; e) 记录生成首 token 的平均响应时间、90 分位响应时间、95 分位响应时间和 99 分位响应时间

6.3.5 生成式 AI 应用服务 token 吞吐量

生成式 AI 应用服务 token 吞吐量测试按表 20 的规定进行。

表 20 生成式 AI 应用服务 token 吞吐量测试方法

测试项目	生成式 AI 应用服务 token 吞吐量
测试对象	生成式 AI 应用服务
测试数据	根据生成式待测试 AI 应用服务,选取图像或文本测试数据格式,准备文件大小分别为 1 KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据集
测试工具	人工智能负载测试工具,如 Locust(Python 运行环境);资源监控工具
测试环境配置	安装人工智能负载测试工具运行环境,如 Python;安装人工智能负载测试工具;安装资源监控工具
测试步骤	a) 获取待测试生成式 AI 应用服务,发送请求确认服务可用; b) 使用资源监控工具,监控被测 AI 应用服务的资源状态,记录磁盘 I/O、网络带宽、CPU 利用率、内存利用率、人工智能加速处理器利用率、人工智能加速处理器显存使用率等资源数据; c) 获取测试数据集,在测试工具导入测试数据; d) 启动人工智能负载测试工具,依次选取 1 KB、100 KB、500 KB、1 MB、4 MB、20 MB 的测试数据,运行 AI 应用服务测试请求,每项请求运行测试 30 min; e) 记录 token 平均吞吐率和最大吞吐率