ICS 35.040

CCS L 71

团体标准

深度学习推理引擎应用编程接口（API）

规范

Specification for Application Programming Interface (API) of Deep Learning Inference Engine

点击此处添加与国际标准一致性程度的标识

|  |
| --- |
| DS1.0 |
|  |

XXXX - XX - XX实施

中关村视听产业技术创新联盟 发布

XXXX - XX - XX发布

T/AI XXX.XX—XXXX

|  |  |
| --- | --- |
|  |  |

目  次

前言 III

引言 I

深度学习推理引擎应用编程接口（API）规范 1

1 范围 1

2 规范性引用文件 1

3 术语和定义 1

4 推理引擎应用编程接口框架 2

5 推理引擎接口分类及要求 3

5.1 接口分类 3

5.2 接口要求 3

6 模型管理接口 3

6.1 接口概述 3

6.2 网络接口与参数读取 3

7 设备管理接口 6

7.1 能耗管理接口（可选接口） 6

7.2 工作线程数接口（可选接口） 7

8 输入输出接口 7

8.1 推理对象及相应接口 7

8.2 获取输入数据 8

8.3 获得输出数据 9

9 推理接口 10

9.1 创建推理实例 10

9.2 执行 11

附　录　A （资料性附录） 模型管理接口C++语言描述 12

A.1 网络接口、参数读取 12

A.1.1 set\_model\_dir(model\_dir) 12

A.1.2 model\_dir() 12

A.1.3 set\_model\_file(model\_file) 12

A.1.4 model\_file() 12

A.1.5 set\_param\_file(param\_fie) 13

A.1.6 param\_file() 13

A.1.7 set\_valid\_places(valid places) 13

A.1.8 set\_model\_buffer(model\_buffer, model\_buffer\_size, param\_buffer, param\_buffer\_size) 13

A.1.9 model\_from\_memory() 14

A.1.10 model\_buffer() 14

A.1.11 param\_buffer() 14

附　录　B （资料性附录） 设备管理接口C++语言描述 15

B.1 能耗管理接口 15

B.1.1 set\_power\_mode(mode) 15

B.1.2 power\_mode() 15

B.2 工作线程数接口 15

B.2.1 set\_threads(threads) 15

B.2.2 threads() 15

附　录　C （资料性附录） 输入输出接口C++语言描述 17

C.1 推理对象及相应接口 17

C.2 获取输入数据 17

C.2.1 get\_input(index) 17

C.2.2 get\_input\_names() 17

C.2.3 get\_input\_bynames(name) 17

C.3 获得输出数据 18

C.3.1 get\_output(index) 18

C.3.2 get\_output\_names() 18

C.3.3 get\_output\_bynames() 18

附　录　D （资料性附录） 推理接口C++语言描述 19

D.1 创建推理实例 19

D.1.1 create\_predictor(Config) 19

D.2 执行 19

D.2.1 void run() 19

前  言

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。

本文件由新一代人工智能产业技术创新战略联盟AI标准工作组提出。

本文件由中关村视听产业技术创新联盟归口。

本文件起草单位：北京百度网讯科技有限公司、北京大学、清华大学、北京旷视科技有限公司、上海商汤智能科技有限公司。

本文件主要起草人：胡晓光、杨超、翟季冬、梅敬青、蒋慧、严春伟、范睿博、李笑如、李雨芮、马艳军、于佃海、徐浩、蒋晓琳、张艺伯、吴庚。

引  言

本文件的发布机构提请注意，声明符合本文件时，可能涉及到附录A.2如下1项专利的使用。专利名称如下：

CN202011021003.5 用于确定待部署模型的方法、装置、设备及其存储介质；

本文件的发布机构对于该专利的真实性、有效性和范围无任何立场。

该专利持有人已向本文件的发布机构保证，他愿意同任何申请人在合理且无歧视的条款和条件下，就专利授权许可进行谈判。该专利持有人的声明已在本文件的发布机构备案，相关信息可以通过以下联系方式获得：

联 系 人：黄铁军（新一代人工智能产业技术创新战略联盟秘书长）

通讯地址：北京大学理科2号楼2641室

邮政编码：100871

电子邮件：tjhuang@pku.edu.cn

电话：+8610-62756172

传真：+8610-62751638

网址：http://www.aitisa.org.cn请注意除上述专利外，本文件的某些内容仍可能涉及专利。本文件的发布机构不承担识别这些专利的责任。

深度学习推理引擎应用编程接口（API）规范

1. 范围

本文件规定了深度学习推理引擎应用编程接口规范，定义了模型管理接口、设备管理接口、输入输出接口、推理接口四类接口规范。

本文件适用于部署在设备上的深度学习推理引擎，包括云侧（IDC）推理引擎及端侧（移动端、边缘端）推理引擎，可用于指导推理引擎API的设计与开发，个别条款不适用于特殊行业，专业应用。

1. 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅注日期的版本适用于本文件。凡是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。

ISO/IEC DIS 22989 Information technology — Artificial intelligence — Artificial intelligence concepts and terminology

1. 术语和定义

下列术语和定义适用于本文件。



**推理 inference**

已知信息进行分析得出结论。

注1：在人工智能中，前提是事实、规则、模型、特征或原始数据。

注2：术语“推断”指的是过程及其结果。

注3：具体定义参见ISO/IEC DIS 22989。



**深度学习推理 deep learning inference**

基于深度学习进行推理的过程。



**深度学习推理引擎 deep learning inference engine**

一种专门用于深度学习预测或推理的软件框架，提供一系列高层接口用于加载深度学习模型，搭载并适配硬件执行模型中涉及的所有运算，获得推理计算结果，可应用于图像识别、自然语言处理等场景。



**模型 model**

一种基于输入数据生成推理或预测的计算结构。



**计算图 computational graph**

一种由一系列节点和边构成的有向无环图，节点分为变量和函数，变量节点通过边相连作为函数节点的输入与输出。



**张量 tensor**

多维数组，数组中的每个元素称为张量的元素（tensor element），在深度学习中特指计算图中用于存储计算结果的一个变量节点。

1. 推理引擎应用编程接口框架

深度学习推理引擎是一种提供深度学习预测或推理功能的软件框架，应用编程接口（API）位于推理引擎与应用层之间，如图1所示，上层应用可通过API进行深度学习模型的调用。深度学习推理引擎应用编程接口涉及的内容包括模型管理、输入输出、设备管理和推理接口。

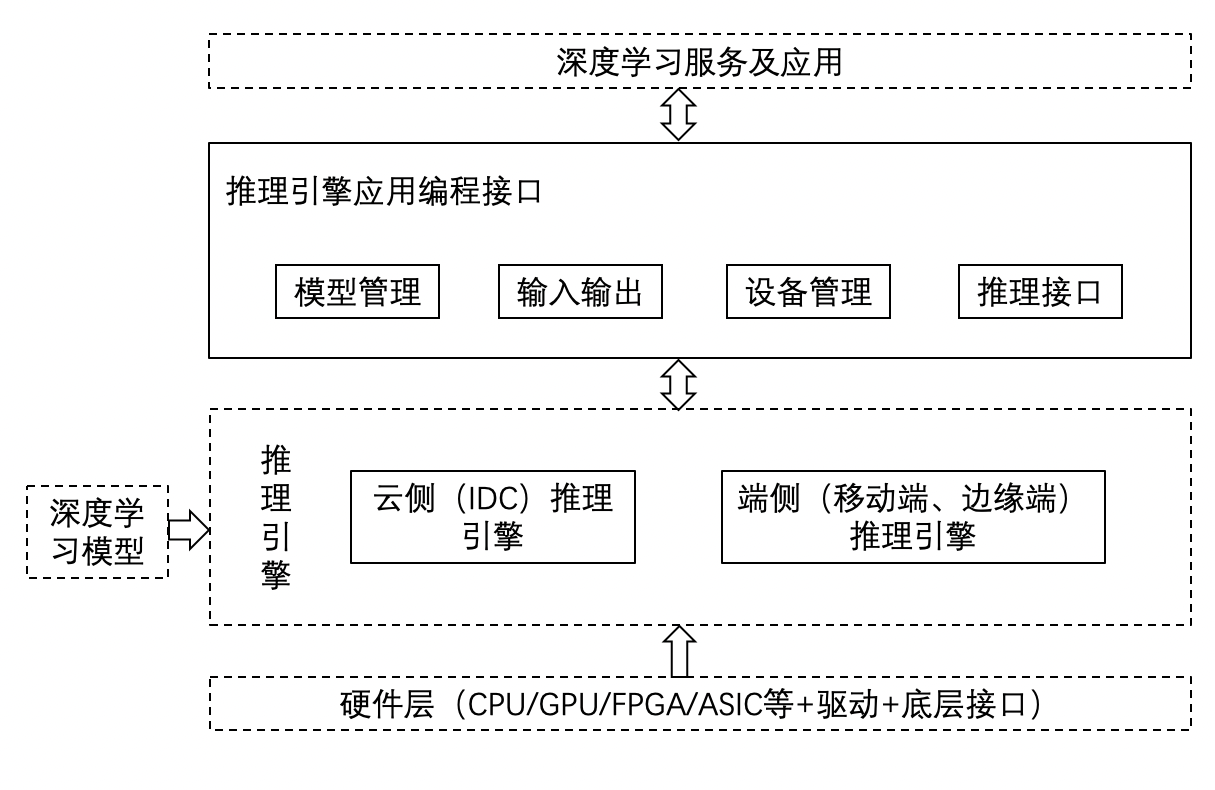


图1 推理引擎应用编程接口框架

框架功能模块如下：

1. 硬件平台：从硬件方面，为深度学习提供计算能力支持，其范围包括芯片、驱动及底层接口。
2. 深度学习模型：用于解决某一特定任务的深度学习算法，通常特指用于表示深度学习算法的计算图结构信息和参数信息。
3. 深度学习推理引擎：一种专门用于深度学习预测或推理的软件框架，提供一系列高层接口用于加载深度学习模型，搭载并适配硬件执行模型中涉及的所有运算，获得推理计算结果，可应用于图像识别、自然语言处理等场景。深度学习推理引擎可以运行在云侧（IDC），也可以运行在端侧（移动端、边缘侧）。
4. 推理引擎应用编程接口：以深度学习推理引擎作为后端运行，通过预先定义的函数，使得深度学习应用及服务开发者可通过调用函数，在无需访问原码，或理解内部工作机制的细节的条件下进行应用开发。

深度学习应用及服务：基于深度学习实现图像分类、语音识别、语义表示等应用或服务功能。深度学习应用及服务可以部署在云侧，也可以部署在端侧。

1. 推理引擎接口分类及要求
   1. 接口分类

按照推理引擎应用编程接口功能进行分类，具体如下：

1. 模型管理接口：用于网络接口与参数读取；
2. 设备管理接口：用于设备能耗及线程管理；
3. 输入输出接口：用于输入数据传入及输出数据获取；
4. 推理接口：用于推理实例创建、推理执行以及推理计算。
   1. 接口要求

推理引擎应用编程接口要求具体如下：

1. 接口函数应开放给用户开发；
2. 接口函数可用多种语言进行实现，C++语言示例见附录；
3. 支持在现有接口的基础上，添加新的接口类型和函数。
4. 模型管理接口
   1. 接口概述

模型管理接口应支持实现网络接口与参数读取等功能，主要包括网络接口与参数读取接口。

* 1. 网络接口与参数读取
     1. 设置模型文件夹路径

该接口用于设置模型文件夹路径，当需要从磁盘加载非组合模型时使用。

表 1 设置模型文件夹路径接口

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 文件夹路径 | 输入参数 | 是 | 设置的模型文件夹路径 |

返回值：调用成功，返回NULL；调用失败，返回非NULL。

* + 1. 获取设置模型文件夹路径

该接口用于返回设置的模型文件夹路径，加载非组合模型时使用。

表 2 获取设置模型文件夹路径接口返回的字段

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 文件夹路径 | 输出参数 | 是 | 返回设置的文件夹路径 |

* + 1. 设置模型文件路径

该接口用于设置模型文件路径，加载组合形式模式时使用。

表 3 设置模型文件路径接口

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 文件路径 | 输入参数 | 是 | 设置的模型文件路径 |

返回值：调用成功，返回NULL；调用失败，返回非NULL。

* + 1. 返回模型文件路径

该接口用于获取设置模型文件路径，加载组合形式模式时使用。

表 4 获取设置模型文件夹路径接口返回的字段

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 文件路径 | 输出参数 | 是 | 获取的模型文件路径 |

* + 1. 设置模型参数文件路径

该接口用于设置模型参数文件路径，加载组合形式模型时使用。

表 5 设置模型参数文件路径接口

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 模型参数文件路径 | 输入参数 | 是 | 加载组合形式模型时使用 |

返回值：调用成功，返回NULL；调用失败，返回非NULL。

* + 1. 获取模型参数文件路径

该接口用于获取模型参数文件路径，加载组合形式模型时使用。

表 6 获取模型参数文件路径接口返回的字段

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 模型参数文件路径 | 输出参数 | 是 | 加载组合形式模型时使用 |

* + 1. 设置可用的计算模式列表

该接口用于设置可用的计算模式列表。

表 7 设置可用的计算模式列表接口

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 计算模式列表 | 输入参数 | 是 | 可用的计算模式列表 |

返回值：调用成功，返回NULL；调用失败，返回非NULL。

* + 1. 设置模型、参数的内存地址

该接口用于设置模型、参数的内存地址，需要从内存加载模型时使用。

表 8 设置模型、参数的内存地址接口

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 模型结构数据 | 输入参数 | 是 | 内存中模型结构数据 |
| 模型结构数据大小 | 输入参数 | 是 | 内存中模型结构数据的大小 |
| 模型参数数据 | 输入参数 | 是 | 内存中模型参数数据 |
| 模型参数数据大小 | 输入参数 | 是 | 内存中模型参数数据大小 |

返回值：调用成功，返回NULL；调用失败，返回非NULL。

* + 1. 从内存加载模型接口

该接口用于判断是否从内存中加载模型。

表 9 从内存加载模型接口的返回的字段

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 是否从内存加载模型 | 输出参数 | 是 | 当调用设置模型、参数的内存地址函数接口时，返回true，否则，返回flase |

* + 1. 获取内存中模型结构数据

该接口用于获取内存中模型结构数据。

表 10 获取内存中模型结构数据接口返回的字段

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 模型结构数据 | 输出参数 | 是 | 内存中模型结构数据 |

* + 1. 获取内存中模型参数数据

该接口用于获取内存中模型参数数据。

表 11 获取内存中模型参数数据接口返回的字段

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 模型参数数据 | 输出参数 | 是 | 内存中模型参数数据 |

1. 设备管理接口
   1. 能耗管理接口（可选接口）
      1. 设置能耗模式

该接口用于设置设备的能耗模式。

表 12 设置能耗模式接口

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 能耗模式 | 输入参数 | 是 | 设置的设备能耗模式 |

返回值：调用成功，返回NULL；调用失败，返回非NULL。

* + 1. 获取能耗模式

该接口用于获取设备的能耗模式。

表 13 获取能耗模式接口返回的字段

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 能耗模式 | 输出参数 | 是 | 获取的设备能耗模式 |

* 1. 工作线程数接口（可选接口）
     1. 设置工作线程数

该接口用于设置设备的工作线程数。

表 14 设置工作线程数接口

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 工作线程数 | 输入参数 | 是 | 设置的设备线程模式。若不设置，则默认使用单线程 |

返回值：调用成功，返回NULL；调用失败，返回非NULL。

* + 1. 获取工作线程数

该接口用于获取设备的线程数。

表 15 获取工作线程数接口返回的字段

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 工作线程数 | 输出参数 | 是 | 获取设置的工作线程数 |

1. 输入输出接口
   1. 推理对象及相应接口

该接口用于创建推理对象，包含了获取输入、获取输出两大类接口及执行函数接口，其中执行接口描述见本标准9.2小节。

表 16 推理对象接口

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 输入张量的指针 | 输入函数 | 是 | 用来设置模型的输入数据 |
| 输出张量的指针 | 输入函数 | 是 | 用来获取模型的输出结果 |
| 获取输入张量的指针 | 输入函数 | 是 | 根据名称获取模型输入张量的指针 |
| 获取输出张量的指针 | 输入函数 | 是 | 根据名称获取模型输出张量的指针 |
| 执行函数 | 输入函数 | 是 | 执行推理 |

* 1. 获取输入数据
     1. 获取输入张量指针

该接口可获取输入张量的指针，用来设置模型的输入数据。

表 17 获取输入张量指针接口

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 张量索引 | 输入参数 | 是 | 输入张量的索引 |

表 18 获取输入张量指针返回的字段

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 张量指针 | 输出参数 | 是 | 根据索引确认输入张量的指针 |

* + 1. 获取所有输入张量名称

该接口可获取所有输入张量的名称。

表 19 获取所有输入数据接口返回的字段

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 张量名称 | 输出参数 | 是 | 所有输入张量的名称 |

* + 1. 根据名称获取输入张量指针

该接口可根据名称获取输入张量的指针。

表 20 根据名称获取输入张量指针的接口

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 张量名称 | 输入参数 | 是 | 输入张量的名称 |

表 21 根据名称获取输入张量指针的接口返回的字段

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 张量指针 | 输出参数 | 是 | 输入张量的指针 |

* 1. 获得输出数据
     1. 获取输出张量指针

该接口可获取输出张量的指针，用来获取模型的输出结果。

表 22 获取输出张量指针接口

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 张量索引 | 输入参数 | 是 | 输出张量的索引 |

表 23 获取模型输出张量接口返回的字段

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 张量指针 | 输出参数 | 是 | 根据索引确认输出张量的指针 |

* + 1. 获取所有输出张量名称

该接口用于获取所有张量的名称。

表 24 获取所有输出张量名称接口返回的字段

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 张量名称 | 输出参数 | 是 | 所有输出张量的名称 |

* + 1. 根据名称获取输出张量指针

该接口用于根据名字获取输出张量的指针，用来获取模型的输出结果。

表 25 根据名称获取输出张量指针接口

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 张量名称 | 输入参数 | 是 | 输出张量的名称 |

表 26 根据名称获取输出张量指针接口返回的字段

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 张量指针 | 输出参数 | 是 | 根据张量名称返回的输出张量指针 |

1. 推理接口
   1. 创建推理实例

该接口可用于创建推理实例，根据配置构建推理器。

表 27 创建推理实例接口

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 配置信息 | 输入参数 | 是 | 构建推理器的配置信息 |

表 28 获取模型输出结果接口返回的字段

|  |  |  |  |
| --- | --- | --- | --- |
| 参数 | 输入/输出类型 | 是否必选 | 参数说明 |
| 推理器 | 输出参数 | 是 | 构建的推理器 |

* 1. 执行

接口用于执行模型预测，需在设置输入数据后调用，无输入输出参数。

1. （资料性附录）  
   模型管理接口C++语言描述
   1. 网络接口、参数读取
      1. set\_model\_dir(model\_dir)

功能描述：

设置模型文件夹路径，当需要从磁盘加载非组合模型时使用。

参数说明：

set\_model\_dir (string)，模型文件夹路径

返回：

None

返回类型：

None

* + 1. model\_dir()

功能描述：

获取设置的模型文件夹路径。

参数说明：

None

返回：

设置的文件夹路径。

返回类型：

string

* + 1. set\_model\_file(model\_file)

功能说明：

设置模型文件路径，加载组合形式模型时使用。

参数说明：

model\_file(string)，模型文件路径

返回：

None

返回类型：

None

* + 1. model\_file()

功能说明：

获取设置模型文件路径，加载组合形式模型时使用。

参数说明：

None

返回：

模型文件路径

返回类型：

string

* + 1. set\_param\_file(param\_fie)

功能说明：

获取设置模型参数文件路径，加载组合形式模型时使用。

参数说明：

param\_fie(str), 模型参数文件路径

返回：

None

返回类型：

None

* + 1. param\_file()

功能说明：

获取设置模型参数文件路径，加载组合形式模型时使用。

参数说明：

None

返回：

模型参数文件路径

返回类型：

string

* + 1. set\_valid\_places(valid places)

功能说明：

设置可用的模型计算模式列表。

参数说明：

valid\_places(list)，可用计算模式列表

返回：

None

返回类型：

None

* + 1. set\_model\_buffer(model\_buffer, model\_buffer\_size, param\_buffer, param\_buffer\_size)

功能说明：

设置模型、参数的内存地址，当需要从内存加载模型时使用。

参数说明：

model\_buffer(const char\*)，内存中模型结构数据；

model\_buffer\_size(size \_t)，内存中模型结构数据的大小；

param\_buffer(const char\*)，内存中模型参数数据；

param\_buffer\_size(size\_t)，内存中模型参数数据的大小。

返回：

None

返回类型：

Void

* + 1. model\_from\_memory()

功能说明：

是否从内存中加载模型，当调用set\_mode\_buffer接口时返回为true。

参数说明：

None

返回：

是否从内存加载模型

返回类型：

bool

* + 1. model\_buffer()

功能说明：

获取内存中模型结构数据。

参数说明：

None

返回：

内存中模型结构数据

返回类型：

string

* + 1. param\_buffer()

功能说明：

获取内存中模型参数数据。

参数说明：

None

返回：

内存中模型参数数据

返回类型：

string

1. （资料性附录）  
   设备管理接口C++语言描述
   1. 能耗管理接口
      1. set\_power\_mode(mode)

功能描述：

设置设备能耗模式。

参数说明：

mode(PowerMode)，设备能耗模式。

返回：

None

返回类型：

void

* + 1. power\_mode()

功能说明：

获取设置的设备能耗模式。

参数说明：

None

返回：

设置的设备能耗模式。

返回类型：

PowerMode

* 1. 工作线程数接口
     1. set\_threads(threads)

功能描述：

设置设备工作线程数。若不设置，则默认使用单线程。

参数说明：

threads(int)，工作线程数。

返回：

None

返回类型：

void

* + 1. threads()

功能说明：

获取设置的工作线程数

参数说明：

None

返回：

工作线程数

返回类型：

int

1. （资料性附录）  
   输入输出接口C++语言描述
   1. 推理对象及相应接口

Predictor为推理库执行期的对象，包含了获取输入、获取输出两大类接口。基本定义如下：

class Predictor {

public:

std::shared\_ptr<Tensor> GetInput(int idx);

std::shared\_ptr<Tensor> GetOutput(int idx);

std::shared\_ptr<Tensor> GetInputByName(const std::string& name);

std::shared\_ptr<Tensor> GetOutputByName(const std::string& name);

virtual void Run() = 0;

};

* 1. 获取输入数据
     1. get\_input(index)

功能说明：

获取输入Tensor指针，用来设置模型的输入数据。

参数说明：

index(int)， 输入Tensor的索引

返回：

第index个输入Tensor的指针

返回类型：

unique\_ptr<Tensor>

* + 1. get\_input\_names()

功能说明：

获取所有输入Tensor的名称。

参数说明：

None

返回：

所有输入Tensor的名称

返回类型：

vector<string>

* + 1. get\_input\_bynames(name)

功能说明：

根据名称获取输出Tensor的指针，用来获取模型的输出结果。

参数说明：

name(string)，输入Tensor的名称

返回：

输入Tensor的指针

返回类型：

unique\_ptr<Tensor>

* 1. 获得输出数据
     1. get\_output(index)

功能说明：

获取输出Tensor的指针，用来获取模型的输出结果。

参数说明：

index(int) ，输出Tensor的索引

返回：

第index个输出Tensor的指针

返回类型：

unique\_ptr<Tensor>

* + 1. get\_output\_names()

功能说明：

获取所有输出Tensor的名称。

参数说明：

None

返回：

所有输出Tensor的名称

返回类型：

vector<string>

* + 1. get\_output\_bynames()

功能说明：

根据名字获取输出张量的指针，用来获取模型的输出结果。

参数说明：

name(string)，输出Tensor的名称

返回：

输出Tensor的指针

返回类型：

unique\_ptr<Tensor>

1. （资料性附录）  
   推理接口C++语言描述
   1. 创建推理实例
      1. create\_predictor(Config)

功能描述：

创建推理实例，根据配置信息构建推理器。

参数说明：

config(Config) 构建Predictor的配置信息

返回：

share\_ptr<Predictor>返回推理器指针

返回类型：

shared\_ptr<Predictor>

* 1. 执行
     1. void run()

功能说明：

执行模型推理，需要在设置输入数据后调用。

参数说明：

None

返回：

None

返回类型：

None

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_