天数智芯

PyTorch 框架功能说明

版本：V4.0.0-MR

日期：2024.4.25

适用产品：智铠 50| 智铠 100

1 声明

1.1 版权声明

1.2 免责声明

天数智芯可以随时对本文档或本文档中描述的产品进行改进和/或更改。本文档包括与天数智芯产品有关的信息，作为说明典型应用的一种方式, 因此，不一定提供足以进行生产设计的完整信息。对于本文档中内容的准确性或完整性，天数智芯不做任何陈述或保证。

1.3 联系方式

地址：上海闵行区陈行公路 2168 号 3 幢

电话：021-68886607

网址：www.iluvatar.com

Contents

2 PyTorch 框架功能说明

2.1 PyTorch 框架概览

天数智算软件栈提供天数智芯适配版的 PyTorch v2.1.1 深度学习编程框架，开发者可以基于天数智芯加速卡开发更加简洁且通用的应用。得益于天数智算软件栈对 PyTorch 常用算子及网络模型的支持，开发者针对天数智芯加速卡开发应用时，可以便捷地调用深度学习以及各类数据科学应用开发所需的算子，灵活地构造各类深度神经网络模型以及其他机器学习领域的算法。此外，当您将应用从其它硬件平台迁移至天数智芯硬件平台时，您无需修改代码就可进行模型训练和推理，无需特殊设置您就可以尝试直接运行应用。

天数智芯适配版的 PyTorch v2.1.1 深度学习编程框架功能包括：

- 提供与 PyTorch 官方开源框架一致的算子
  - torch.nn
  - torch.optim
  - torch.Tensor
  - torch.autograd
  - torch.cuda
  - torch.fft
  - torch.jit
  - torch.utils.cpp_extension
  - torch.distributions
  - torch.distributed
  - torch.distributed.optim

  更多算子支持，请参考 PyTorch 官方 PyTorch API 列表。
- 支持常见的网络模型，覆盖图像分类、目标检测、图形分割和自然语言处理等
- 支持多卡分布式训练，通信后端支持 Gloo 和 NCCL（注：支持 NCCL 部分功能）
- 提供 Horovod 深度分布式训练框架
- 提供 DALI 加速库，加速计算机视觉和音频深度学习应用程序的数据加载和预处理
- 提供 Apex 加速库，用于自动混合精度和分布式训练
- 支持 PyTorch 3D，加速面向 3D 计算机视觉的深度学习
- 提供 FlashAttention-2 高性能算子库，提高运行效率并减少显存使用

您可参考《软件栈安装指南》安装天数智芯适配版深度学习框架和加速库。

2.2 已验证通过的网络模型

天数智芯适配版深度学习框架已验证对如下常见 PyTorch 网络模型的支持：

- AlexNet
- BERT

- Inception_V3
- ResNet50
- VGG16
- YOLOv5

2.3 天数适配版 Horovod

Horovod 是基于 Ring-AllReduce 方法的深度学习插件，通过数据并行实现分布式训练，利用环形拓扑结构来实现高效的 GPU 间通信。

深度学习应用的开发者只需要为 Horovod 进行配置，即可实现更快、更轻松的分布式训练，具体内容请参考 Horovod 官方网站（https://horovod.ai/）。

天数适配版 Horovod 支持 PyTorch 和 TensorFlow 深度学习框架。

本次适配的版本为官方 Horovod v0.27.0，天数适配版 Horovod 支持部分 GPU 集合操作，支持的集合操作 API 包括但不限于下表所列：

集合操作	API 举例
allreduce	hvd.allreduce hvd.allreduce_ hvd.allreduce_async
allgather	hvd.allgather hvd.allgather_async
broadcast	hvd.broadcast hvd.broadcast_ hvd.broadcast_async
alltoall	hvd.alltoall
reducescatter	hvd.reducescatter hvd.reducescatter_async
join	hvd.join hvd.synchronize
group	hvd.grouped_allreduce hvd.grouped_allreduce_ hvd.grouped_allgather hvd.grouped_reducescatte r
process_set	hvd.add_process_set hvd.remove_process_set

2.3.1 运行 Horovod

Horovod 提供以下两种 Controller 供您选择：

2.3.1.1 方法一：使用 MPI

```
$ install-mpi
$ horovodrun --mpi
```

Tip

由于天数适配版 Horovod 适配 OpenMPI 4.0.7，所以您需要执行 install-mpi 命令安装 openmpi-4.0.7。如果您希望使用环境中已安装的 Open MPI 版本，则无需执行 install-mpi 指令，但程序运行的正确性需要您自行验证。

2.3.1.2 方法二：使用 Gloo

```
$ horovodrun --gloo
```

2.3.1.3 horovodrun 命令说明

您需要使用 horovodrun 命令的 **-np** 选项指定进程数量，如：

- 单机多卡，如在单机 4 卡上运行 Horovod 程序:

  ```
  $ horovodrun -np 4 -H localhost:4 python train.py
  ```

- 多机多卡，如使用 4 台机器、每机 4 卡的集群上运行 Horovod 程序，在单机上执行以下命令：

  ```
  $ horovodrun -np 16 -H server1:4,server2:4,server3:4,server4:4 python train.py
  ```

Tip

无论是单机多卡，还是多机多卡，都只需在一台机器上执行一次命令即可，Horovod 会用 MPI 启动进程和传递数据。

2.4 支持的加速库

2.4.1 Apex

Apex(A PyTorch Extension) 是 NVIDIA 的一个 PyTorh 扩展库，具体介绍可参考 官方文档 。

本次适配的版本为 Apex master 分支 f03c6fb67e5 commit。

本次发布支持以下编译选项：

```
--cpp_ext
--cuda_ext
--distributed_adam
--distributed_lamb
--deprecated_fused_adam
--deprecated_fused_lamb
--fast_layer_norm
--transducer
--fmha
--bnp
--xentropy
--fast_multihead_attn
--permutation_search
--focal_loss
--index_mul_2d
--nccl_p2p
```

2.4.2 DALI

NVIDIA Data Loading Library (DALI) 可用于加速计算机视觉和音频深度学习应用程序的数据加载和预处理，DALI 详细的介绍可以参考 NVIDIA DALI 。

下表列出了支持的算子：

算子	CPU	GPU	功能介绍
audio_decoder	支持		Legacy alias for decoders.audio().
batch_permutation	支持		Produces a batch of random integers which can be used as indices for indexing samples in the batch.
bb_flip	支持	支持	Flips bounding boxes horizontaly or verticaly (mirror).
bbox_paste	支持		Transforms bounding boxes so that the boxes remain in the same place in the image after the image is pasted on a larger canvas.
box_encoder	支持	支持	Encodes the input bounding boxes and labels using a set of default boxes (anchors) passed as an argument.

算子	CPU	GPU	功能介绍
brightness	支持	支持	Adjusts the brightness of the images.
brightness_contrast	支持	支持	Adjusts the brightness and contrast of the images.
caffe2_reader	支持		Legacy alias for readers.caffe2().
caffe_reader	支持		Legacy alias for readers.caffe().
cast	支持	支持	Cast tensor to a different type.
cat	支持	支持	Joins the input tensors along an existing axis.
coco_reader	支持		Legacy alias for readers.coco().
coin_flip	支持	支持	Generates random boolean values following a bernoulli distribution.
c olor_space_conversion	支持	支持	Converts between various image color models.
color_twist	支持	支持	Adjusts hue, saturation and brightness of the image.
compose	支持	支持	Returns a meta-operator that chains the operations in op_list.
constant	支持	支持	Produces a batch of constant tensors.
contrast	支持	支持	Adjusts the contrast of the images.
coord_flip	支持	支持	Transforms vectors or points by flipping (reflecting) their coordinates with respect to a given center.
coord_transform	支持	支持	Applies a linear transformation to points or vectors.
copy	支持	支持	Creates a copy of the input tensor.
crop	支持	支持	Crops the images with the specified window dimensions and window position (upper left corner).

算子	CPU	GPU	功能介绍
crop_mirror_normalize	支持	支持	Performs fused cropping, normalization, format conversion (NHWC to NCHW) if desired, and type casting.
dl_t ensor_python_function	支持	支持	Executes a Python function that operates on DLPack tensors.
dump_image	支持	支持	Save images in batch to disk in PPM format.
element_extract	支持	支持	Extracts one or more elements from input sequence.
erase	支持	支持	Erases one or more regions from the input tensors.
expand_dims	支持	支持	Insert new dimension(s) with extent 1 to the data shape.
external_source	支持	支持	Allows externally provided data to be passed as an input to the pipeline.
fa st_resize_crop_mirror	支持		Performs a fused resize, crop, mirror operation.
file_reader	支持		Legacy alias for readers.file().
flip	支持	支持	Flips the images in selected dimensions (horizontal, vertical, and depthwise).
gaussian_blur	支持	支持	Applies a Gaussian Blur to the input.
grid_mask	支持	支持	Performs the gridmask augumentation.
hsv	支持	支持	Adjusts hue, saturation and value (brightness) of the images.
hue	支持	支持	Changes the hue level of the image.
image_decoder	支持		Legacy alias for decoders.image().
image_decoder_crop	支持		Legacy alias for d ecoders.image_crop().
imag e_decoder_random_crop	支持		Legacy alias for decoders .image_random_crop().

算子	CPU	GPU	功能介绍
image_decoder_slice	支持		Legacy alias for de coders.image_slice().
jitter		支持	Performs a random Jitter augmentation.
jpeg_c ompression_distortion	支持	支持	Introduces JPEG compression artifacts to RGB images.
lookup_table	支持	支持	Maps the input to output by using a lookup table that is specified by keys and values, and a default_value for unspecified keys.
mel_filter_bank	支持	支持	Converts a spectrogram to a mel spectrogram by applying a bank of triangular filters.
mfcc	支持	支持	Computes Mel Frequency Cepstral Coefficiencs (MFCC) from a mel spectrogram.
multi_paste	支持	支持	Performs multiple pastes from image batch to each of outputs
mxnet_reader	支持		Legacy alias for readers.mxnet().
nemo_asr_reader	支持		Legacy alias fo r readers.nemo_asr().
nonsilent_region	支持		Performs leading and trailing silence detection in an audio buffer.
normal_distribution	支持	支持	Generates random numbers following a normal distribution.
normalize	支持	支持	Normalizes the input by removing the mean and dividing by the standard deviation.
numba_function	支持		Invokes a njit compiled Numba function.
numpy_reader	支持	支持	Legacy alias for readers.numpy().
old_color_twist	支持		A combination of hue, saturation, contrast, and brightness.

算子	CPU	GPU	功能介绍
one_hot	支持	支持	Produces a one-hot encoding of the input.
optical_flow			Calculates the optical flow between images in the input.
pad	支持	支持	Pads all samples with the fill_value in the specified axes to match the biggest extent in the batch for those axes or to match the minimum shape specified.
paste		支持	Pastes the input images on a larger canvas, where the canvas size is equal t o input size * ratio.
peek_image_shape	支持		Obtains the shape of the encoded image.
permute_batch	支持	支持	Returns a batch of tensors constructed by selecting tensors from the input based on indices given in indices argument:
power_spectrum	支持		Calculates power spectrum of the signal.
preemphasis_filter	支持	支持	Applies preemphasis filter to the input data.
python_function	支持	支持	Executes a Python function.
random_bbox_crop	支持		Applies a prospective random crop to an image coordinate space while keeping the bounding boxes, and optionally labels, consistent.
random_resized_crop	支持	支持	Performs a crop with a randomly selected area and aspect ratio and resizes it to the specified size.
reinterpret	支持	支持	Treats content of the input as if it had a different type, shape, and/or layout.

算子	CPU	GPU	功能介绍
reshape	支持	支持	Treats content of the input as if it had a different shape and/or layout.
resize	支持	支持	Resize images.
resize_crop_mirror	支持		Performs a fused resize, crop, mirror operation. Both fixed and random resizing and cropping are supported.
roi_random_crop	支持		Produces a fixed shape cropping window, randomly placed so that as much of the provided region of interest (ROI) is contained in it.
rotate	支持	支持	Rotates the images by the specified angle.
saturation	支持	支持	Changes the saturation level of the image.
sequence_reader	支持		Legacy alias fo r readers.sequence().
sequence_rearrange	支持	支持	Rearranges frames in a sequence.
shapes	支持	支持	Returns the shapes of inputs.
slice	支持	支持	Extracts a subtensor, or slice.
spectrogram	支持	支持	Produces a spectrogram from a 1D signal (for example, audio).
sphere	支持	支持	Performs a sphere augmentation.
squeeze	支持	支持	Removes the dimensions given a s axes or axis_names.
ssd_random_crop	支持		Performs a random crop with bounding boxes where Intersection Over Union (IoU) meets a randomly selected threshold between 0-1.
stack	支持	支持	Joins the input tensors along a new axis.
tfrecord_reader	支持		Legacy alias fo r readers.tfrecord().

算子	CPU	GPU	功能介绍
to_decibels	支持	支持	Converts a magnitude (real, positive) to the decibel scale by using the following formula:
torch_python_function	支持	支持	Executes a function that is operating on Torch tensors.
transpose	支持	支持	Transposes the tensors by reordering the dimensions based on the perm parameter.
uniform	支持	支持	Generates random numbers following a uniform distribution.
video_reader			Legacy alias for readers.video().
video_reader_resize			Legacy alias for re aders.video_resize().
warp_affine	支持	支持	Applies an affine transformation to the images.
water	支持	支持	Performs a water augmentation, which makes the image appear to be underwater.
decoders.audio	支持		Decodes waveforms from encoded audio data.
decoders.image	支持		Decodes images.
decoders.image_crop	支持		Decodes images and extracts regions-of-interest (ROI) that are specified by fixed window dimensions and variable anchors.
decod ers.image_random_crop	支持		Decodes images and randomly crops them.
decoders.image_slice	支持		Decodes images and extracts regions of interest.
noise.gaussian	支持	支持	Applies gaussian noise to the input.
noise.salt_and_pepper	支持	支持	Applies salt-and-pepper noise to the input.
noise.shot	支持	支持	Applies shot noise to the input.

算子	CPU	GPU	功能介绍
random.coin_flip	支持	支持	Generates random boolean values following a bernoulli distribution.
random.normal	支持	支持	Generates random numbers following a normal distribution.
random.uniform	支持	支持	Generates random numbers following a uniform distribution.
readers.caffe	支持		Reads (Image, label) pairs from a Caffe LMDB.
readers.caffe2	支持		Reads sample data from a Caffe2 Lightning Memory-Mapped Database (LMDB).
readers.coco	支持		Reads data from a COCO dataset that is composed of a directory with images and annotation JSON files.
readers.file	支持		Reads file contents and returns file-label pairs.
readers.mxnet	支持		Reads the data from an MXNet RecordIO.
readers.nemo_asr	支持		Reads automatic speech recognition (ASR) data (audio, text) from an NVIDIA NeMo compatible manifest.
readers.numpy	支持	支持	Reads Numpy arrays from a directory.
readers.sequence	支持		Reads [Frame] sequences from a directory representing a collection of streams.
readers.tfrecord	支持		Reads samples from a TensorFlow TFRecord file.
readers.video			Loads and decodes video files using FFmpeg and NVDECODE, which is the hardware-accelerated video decoding feature in the NVIDIA(R) GPU.

算子	CPU	GPU	功能介绍
readers.video_resize			Loads, decodes and resizes video files with FFmpeg and NVDECODE, which is NVIDIA GPU's hardware-accelerated video decoding.
reductions.max	支持	支持	Gets maximal input element along provided axes.
reductions.mean	支持	支持	Gets mean of elements along provided axes.
r eductions.mean_square	支持	支持	Gets mean square of elements along provided axes.
reductions.min	支持	支持	Gets minimal input element along provided axes.
reductions.rms	支持	支持	Gets root mean square of elements along provided axes.
reductions.std_dev	支持	支持	Gets standard deviation of elements along provided axes.
reductions.sum	支持	支持	Gets sum of elements along provided axes.
reductions.variance	支持	支持	Gets variance of elements along provided axes.
segmentat ion.random_mask_pixel	支持		Selects random pixel coordinates in a mask, sampled from a uniform distribution.
segmentati on.random_object_bbox	支持		Randomly selects an object from a mask and returns its bounding box.
segm entation.select_masks	支持		Selects a subset of polygons by their mask ids.
transforms.combine	支持		Combines two or more affine transforms.
transforms.crop	支持		Produces an affine transform matrix that maps a reference coordinate space to another one.
transforms.rotation	支持		Produces a rotation affine transform matrix.
transforms.scale	支持		Produces a scale affine transform matrix.

算子	CPU	GPU	功能介绍
transforms.shear	支持		Produces a shear affine transform matrix.
t ransforms.translation	支持		Produces a translation affine transform matrix.

2.4.3 PyTorch 3D

PyTorch 3D 是一个基于 PyTorch 的高效、可复用的 3D 计算机视觉库。PyTorch 3D 实现了以下新特性：

1. 新的 3D 数据结构 Meshes，可以更好地存储和修改三角网格的数据
2. 高效处理三角网格的算子，如投影变换、图卷积、采样、损失函数等
3. 可微分的网格生成器

更多内容可参考 PyTorch 3D 官网 。

本次发布已验证了 PyTorch 开源项目提供的示例，包括：

- Deform a sphere mesh to dolphin
- Bundle adjustment
- Render textured meshes
- Camera position optimization
- Render textured pointclouds
- Fit a mesh with texture
- Render DensePose data
- Load & Render ShapeNet data
- Fit Textured Volume
- Fit A Simple Neural Radiance Field

2.4.4 FlashAttention-2

FlashAttention-2 是由 Tri Dao 博士开源的一个高性能 Attention 算子库，其主要借鉴了 Apex 扩展包中的 FMHA (Fused Multi-Head Attention) 的实现，整体融合 Attention 算子，并对 softmax 运算进行分块和增量计算。通过重计算而非重加载访存，以避免频繁显存访问，实现了运算效率的提高并降低峰值显存占用。

FlashAttention-2，即 FlashAttention v2 版本，是基于 FlashAttention v1 版本的进一步优化支持，提出了更少的非矩阵乘法 Flops，更好的序列长度并行化，更好的 Warp 分区等思想。

2.4.4.1 支持的模块

本次适配的版本为 FlashAttention v2.0.1 (commit ID b252072409e)，支持的模块列表如下：

编译模块	功能支持情况
flash-attention/csrc/flash-attn	支持 float16 和 bfloat16 两种数据类型 支持 casual mask 和 alibi mask 支持 arbitrary seq len (i.e. unpad input) 支持 arbitrary head dim (最大到 256) 支持 multi-query attention 和 group-query attention 支持 dropout
flash-attention/csrc/rotary	支持
flash-attention/csrc/fused_softmax	支持
flash-attention/csrc/layer_norm	支持
flash-attention/csrc/fused_dense_l ib	支持
flash-attention/csrc/xentropy	支持

其中，flash-attn 作为核心功能模块，支持接口和用法如下：

flash_attn_varlen_func

```
def flash_attn_varlen_func(q, k, v, cu_seqlens_q, cu_seqlens_k, max_seqlen_q, max_seqlen_k,
                           dropout_p=0.0, softmax_scale=None, causal=False,
                           return_attn_probs=False, use_alibi, alibi_mode)
```

flash_attn_varlen_kvpacked_func

```
def flash_attn_varlen_kvpacked_func(q, kv, cu_seqlens_q, cu_seqlens_k, max_seqlen_q,
↪  max_seqlen_k,
                                    dropout_p=0.0, softmax_scale=None, causal=False,
                                    return_attn_probs=False, use_alibi, alibi_mode):
```

flash_attn_varlen_qkvpacked_func

```
def flash_attn_varlen_qkvpacked_func(qkv, cu_seqlens, max_seqlen, dropout_p=0.0,
↪  softmax_scale=None,
                                     causal=False, return_attn_probs=False, use_alibi,
↪  alibi_mode):
```

flash_attn_func

```
def flash_attn_func(q, k, v, dropout_p=0.0, softmax_scale=None, causal=False,
                    return_attn_probs=False, use_alibi, alibi_mode):
```

flash_attn_kvpacked_func

```
def flash_attn_kvpacked_func(q, kv, dropout_p=0.0, softmax_scale=None, causal=False,
                             return_attn_probs=False, use_alibi, alibi_mode):
```

flash_attn_qkvpacked_func

```
def flash_attn_qkvpacked_func(qkv, dropout_p=0.0, softmax_scale=None, causal=False,
                              return_attn_probs=False, use_alibi, alibi_mode):
```

关于 FlashAttention-2 的更多内容，请参考 https://github.com/Dao-AILab/flash-attention/tree/v2.0.1。

Important

目前 flash-attn 底层模块可切换调用 ixDNN 库,此时需要设置环境变量 `export ENABLE_FLASH_ATTENTION_WITH_IXDNN=1`。

2.5 支持的算子

2.5.1 torch.nn

```
AdaptiveAvgPool1d
AdaptiveAvgPool2d
AdaptiveAvgPool3d
AdaptiveLogSoftmaxWithLoss
AdaptiveMaxPool1d
AdaptiveMaxPool2d
AdaptiveMaxPool3d
AlphaDropout
AvgPool1d
AvgPool2d
AvgPool3d
BatchNorm1d
BatchNorm2d
BatchNorm3d
BCELoss
BCEWithLogitsLoss
Bilinear
CELU
ConstantPad1d
ConstantPad2d
ConstantPad3d
Conv1d
Conv2d
Conv3d
```

```
ConvTranspose1d
ConvTranspose2d
ConvTranspose3d
CosineEmbeddingLoss
CosineSimilarity
CrossEntropyLoss
CrossMapLRN2d
CTCLoss
DataParallel
Dropout
Dropout2d
Dropout3d
ELU
Embedding
EmbeddingBag
FeatureAlphaDropout
Flatten
Fold
FractionalMaxPool2d
FractionalMaxPool3d
GELU
GLU
GroupNorm
GRU
GRUCell
Hardshrink
Hardsigmoid
Hardswish
Hardtanh
HingeEmbeddingLoss
Identity
InstanceNorm1d
InstanceNorm2d
InstanceNorm3d
KLDivLoss
L1Loss
LayerNorm
LazyConv1d
LazyConv2d
LazyConv3d
LazyConvTranspose1d
LazyConvTranspose2d
LazyConvTranspose3d
LazyLinear
LeakyReLU
Linear
```

```
LocalResponseNorm
LogSigmoid
LogSoftmax
LPPool1d
LPPool2d
LSTM
LSTMCell
MarginRankingLoss
MaxPool1d
MaxPool2d
MaxPool3d
MaxUnpool1d
MaxUnpool2d
MaxUnpool3d
Module
ModuleDict
ModuleList
MSELoss
MultiheadAttention
MultiLabelMarginLoss
MultiLabelSoftMarginLoss
MultiMarginLoss
NLLLoss
NLLLoss2d
PairwiseDistance
ParameterDict
ParameterList
parallel.DistributedDataParallel
PixelShuffle
PixelUnshuffle
PoissonNLLLoss
PReLU
ReflectionPad1d
ReflectionPad2d
ReLU
ReLU6
ReplicationPad1d
ReplicationPad2d
ReplicationPad3d
RNN
RNNBase
RNNCell
RNNCellBase
RReLU
SELU
Sequential
```

```
Sigmoid
SiLU
SmoothL1Loss
SoftMarginLoss
Softmax
Softmax2d
Softmin
Softplus
Softshrink
Softsign
SyncBatchNorm
Tanh
Tanhshrink
Threshold
Transformer
TransformerDecoder
TransformerDecoderLayer
TransformerEncoder
TransformerEncoderLayer
TripletMarginLoss
Unflatten
Unfold
Upsample
UpsamplingBilinear2d
UpsamplingNearest2d
utils.rnn.pack_padded_sequence
utils.rnn.pad_packed_sequence
ZeroPad2d
utils.clip_grad_norm_
utils.clip_grad_value_
utils.parameters_to_vector
utils.vector_to_parameters
utils.prune.PruningContainer
utils.prune.Identity
utils.prune.RandomUnstructured
utils.prune.L1Unstructured
utils.prune.RandomStructured
utils.prune.LnStructured
utils.prune.CustomFromMask
utils.prune.identity
utils.prune.random_unstructured
utils.prune.l1_unstructured
utils.prune.random_structured
utils.prune.ln_structured
utils.prune.global_unstructured
utils.prune.custom_from_mask
```

```
utils.prune.remove
utils.prune.is_pruned
utils.weight_norm
```

2.5.2 torch.optim

```
Adagrad
Adam
Adamax
AdamW
ASGD
LBFGS
Adadelta
RMSprop
Rprop
SGD
SparseAdam
```

2.5.3 torch.Tensor

```
abs
abs_
absolute
absolute_
acos
acos_
add
add_
addbmm
addcdiv
addcdiv_
addcmul
addcmul_
addmm
addmm_
addmv
addr
addr_
allclose
amax
amin
```

```
angle
apply_
argmax
argmin
argsort
asin
asin_
as_strided
atan
atan_
atan2
atan2_
baddbmm
bernoulli
bernoulli_
bincount
bitwise_not
bitwise_and
bitwise_or
bitwise_xor
bmm
bool
byte
broadcast_to
cauchy_
cat
ceil
ceil_
char
cholesky_inverse
chunk
clamp
clamp_
clamp_max
clamp_min
clone
contiguous
copy_
conj
cos
cos_
cosh
cosh_
acosh
acosh_
cpu
```

```
cross
cuda
logcumsumexp
cumprod
cumsum
cummax
cummin
deg2rad
diag
diff
fill_diagonal_
digamma
digamma_
dim
dist
div
div_
divide
divide_
eq
equal
erf
erf_
erfc
erfc_
erfinv
erfinv_
exp
exp_
exp2
expm1
expm1_
expand
exponential_
flatten
flip
fliplr
flipud
float
floor
floor_
floor_divide
floor_divide_
fmod
fmod_
frac
```

```
gather
geometric_
gt
half
hypot
hypot_
i0
i0_
index_add_
index_add
index_copy_
index_copy
index_fill_
index_fill
index_put_
index_select
isfinite
isinf
nan_to_num
is_contiguous
kthvalue
le
lerp
lerp_
lgamma
lgamma_
linalg_inv
linalg_norm
linalg_pinv
linalg_slogdet
linalg_solve
linalg_svd
log
log_
log_normal_
log10
log10_
log1p
log1p_
log2
log2_
logical_and
logical_not
logical_or
logical_xor
logit
```

```
logit_
masked_scatter
masked_select
max
min
mean
moveaxis
matmul
median
mm
movedim
mode
mul
mul_
multinomial
nanmedian
narrow
neg
neg_
nonzero
norm
normal_
numpy
pinverse
polygamma
pow
pow_
prod
random_
reciprocal
reciprocal_
remainder_
renorm
renorm_
repeat
round
round_
rsqrt
rsqrt_
scatter
scatter_
scatter_add_
scatter_add
select
sigmoid
sigmoid_
```

```
sign
sign_
sgn
sgn_
sin
sin_
sinc
sinc_
sinh
sinh_
asinh
asinh_
size
sort
split
sqrt
sqrt_
squeeze
std
svd
storage
stride
sub
sum
to
tan
tan_
tanh
tanh_
atanh
atanh_
tensor_split
topk
trace
transpose
transpose_
triangular_solve
tile
tril
tril_indices
triu
triu_indices
trunc
trunc_
unique
unbind
```

```
unfold
uniform_
unsqueeze
var
vstack
view
view_as
xlogy
zero_
all
any
div_floor
stack
dstack
hstack
remainder
randperm
zeros
__lshift__
randperm
__rshift__
pdist
```

2.5.4 torch.autograd

```
backward
grad
functional.jacobian
no_grad
profiler.profile
```

2.5.5 torch.cuda

```
current_device
current_stream
device
device_count
set_device
synchronize
manual_seed
manual_seed_all
```

```
seed
seed_all
cudart
```

2.5.6 torch.fft

Note

- 当前仅支持部分 torch.fft 算子。
- FP16 暂不支持该算子。

```
fft
ifft
fft2
ifft2
fftn
ifftn
rfft
irfft
rfft2
irfft2
rfftn
irfftn
hfft
ihfft
fftfreq
rfftfreq
fftshift
ifftshift
```

2.5.7 torch.jit

```
script
trace
script_if_tracing
trace_module
fork
wait
ScriptModule
ScriptFunction
freeze
save
```

```
load
ignore
unused
isinstance
Attribute
annotate
```

2.5.8 torch.utils.cpp_extension

```
CppExtension
CUDAExtension
BuildExtension
load
load_inline
include_paths
check_compiler_abi_compatibility
is_ninja_available
```

2.5.9 torch.distributions

```
distribution.Distribution
exp_family.ExponentialFamily
bernoulli.Bernoulli
beta.Beta
binomial.Binomial
categorical.Categorical
cauchy.Cauchy
chi2.Chi2
continuous_bernoulli.ContinuousBernoulli
dirichlet.Dirichlet
exponential.Exponential
fishersnedecor.FisherSnedecor
gamma.Gamma
geometric.Geometric
gumbel.Gumbel
half_cauchy.HalfCauchy
half_normal.HalfNormal
independent.Independent
kumaraswamy.Kumaraswamy
lkj_cholesky.LKJCholesky
laplace.Laplace
```

```
log_normal.LogNormal
lowrank_multivariate_normal.LowRankMultivariateNormal
mixture_same_family.MixtureSameFamily
multinomial.Multinomial
multivariate_normal.MultivariateNormal
negative_binomial.NegativeBinomial
normal.Normal
one_hot_categorical.OneHotCategorical
pareto.Pareto
poisson.Poisson
relaxed_bernoulli.RelaxedBernoulli
relaxed_bernoulli.LogitRelaxedBernoulli
relaxed_categorical.RelaxedOneHotCategorical
studentT.StudentT
transformed_distribution.TransformedDistribution
uniform.Uniform
von_mises.VonMises
weibull.Weibull
kl.kl_divergence
kl.register_kl
transforms.Transform
transforms.ComposeTransform
transforms.IndependentTransform
transforms.ReshapeTransform
distributions.transforms.ExpTransform
torch.distributions.transforms.PowerTransform
transforms.SigmoidTransform
transforms.TanhTransform
transforms.AbsTransform
transforms.AffineTransform
transforms.CorrCholeskyTransform
transforms.SoftmaxTransform
transforms.StickBreakingTransform
transforms.LowerCholeskyTransform
transforms.StackTransform
constraints.Constraint
constraint_registry.ConstraintRegistry
```

2.5.10 torch.distributed

```
is_available
init_process_group
is_nccl_available
get_backend
```

```
get_rank
get_world_size
new_group
broadcast
all_reduce
all_gather
reduce_scatter
reduce
```

2.5.11 torch.distributed.optim

```
ZeroRedundancyOptimizer
```

3 商标声明

- 天数智芯、天数智芯 logo、Iluvatar CoreX 等商标、标识、组合商标为上海天数智芯半导体有限公司之注册商标或商标，受法律保护。
- 除了天数智芯的注册商标外，本内容中使用的其他产品名称及标志仅用于识别目的，该等名称及标志可能是归属于其各自公司的商标。我们否认对该等名称及标志的所有权利。
- CentOS 标识为 Red Hat 公司的商标。
- Docker 为 Docker 公司在美国和其他国家的商标或注册商标。
- Linux 为 Linus Torvalds 在美国和其它国家的注册商标。
- NVIDIA 和 CUDA 为 NVIDIA 公司在美国和/或其它国家的商标和/或注册商标。
- PyTorch 为 Facebook 公司的商标。
- TensorFlow 为 Google 公司的商标。
- Ubuntu 为 Canonical 公司的注册商标。