

WEAVER – Supported Keras Layers (v 1.0.0)

WEAVER is an inference engine that works with neural networks created in TensorFlow/Keras and saved as .H5 file. The below table summarizes which Keras layers are supported and which standard networks make use of them. “Optional” means that the given layer appears in some variants of the architecture.

NOTE: Any layer you find important for your project, but not yet implemented, please contact our technical support for help.

Keras layer name	Support in WEAVER	Use in popular architectures (based on Keras.applications unless specified otherwise)						
		MobileNet	MobileNet V2	VGG19	ResNet50	Mask RCNN	YOLOv3	U-Net
Activation	activation=relu/tanh/sigmoid/softmax	Yes (Softmax)	-	-	-	-	-	-
Add	two inputs only	-	Yes	-	-	Yes	Yes	-
Average	Not implemented	-	-	-	-	-	-	-
BatchNormalization	axis=-1	Yes	Yes	-	Yes	-	Yes	-
Conv2D	activation=None/tanh/relu/sigmoid/softmax/elu	Yes	Yes	-	Yes	Yes	Yes	Yes
Conv2DTranspose	activation=None/tanh/relu/sigmoid/softmax/elu output_padding=None	-	-	-	-	-	-	-
Concatenate	axis=-1	-	-	-	-	Yes	Yes	Yes
Cropping2D	Fully supported	-	-	-	-	-	-	-
Dense	activation=None/tanh/relu/sigmoid/softmax/elu	-	Optional	Optional	Optional	-	-	-
DepthwiseConv2D	activation=None/tanh/relu/sigmoid/softmax/elu; depth_multiplier=1	Yes	Yes	-	-	-	-	-
DetectionLayer (custom)	Implemented, but differently to the original	-	-	-	-	Yes	-	-
Dot	Not implemented	-	-	-	-	-	-	-
Dropout	Fully supported	Yes	Yes	-	-	-	-	-
ELU	Fully supported	-	-	-	-	-	-	-
Flatten	Not implemented	-	-	Optional	-	-	-	-
GlobalAveragePooling2D	Fully supported	Optional	-	Optional	Optional	-	-	-
GlobalMaxPooling2D	Not implemented	Optional	Optional	-	Optional	-	-	-
InputLayer	Fully supported	-	-	-	-	-	-	-
Lambda	Not supported (contact us for custom development if needed)	-	-	-	-	Yes	-	-
LayerNormalization	Not implemented	-	-	-	-	-	-	-
LeakyReLU	Fully supported	-	-	-	-	-	Yes	-
LocallyConnected	Not implemented	-	-	-	-	-	-	-
MaxPooling2D	padding=valid	-	-	Yes	Yes	Yes	-	Yes
Maximum	Not implemented	-	-	-	-	-	-	-
Minimum	Not implemented	-	-	-	-	-	-	-
Model (nested)	Fully supported	-	-	-	-	Yes	-	-
Multiply	two inputs only	-	-	-	-	-	-	-
ProposalLayer (custom)	Implemented, but differently to the original	-	-	-	-	Yes	-	-
PyramidROIAAlign (custom)	Implemented, but differently to the original	-	-	-	-	Yes	-	-
ReLU	negative_slope=0; threshold=0	Yes	Yes	-	Yes	Yes	-	Yes
Reshape	target_shape have to count 1, 2 or 3 values	Yes	-	-	-	Yes	-	-
PReLU	Shared_axes=None/[1, 2]	-	-	-	-	-	-	-
SeparableConv	Not implemented	-	-	-	-	-	-	-
SoftMax	Fully supported	-	-	-	-	Yes	-	-
Subtract	Not implemented	-	-	-	-	-	-	-
TimeDistributed	Fully supported for: BatchNormalization, Conv2D, Conv2DTranspose, Dense, Softmax	-	-	-	-	Yes	-	-
UpSampling2D	interpolation=nearest Executing on CUDA: both values in size must be the same	-	-	-	-	Yes	Yes	Yes
ZeroPadding2D	Fully supported	Yes	Yes	-	Yes	Yes	Yes	-

Additional remarks:

- For layers that have "data_format" parameter, only "channels_last" is supported.
- Nested models are also supported as long as they contain only supported layers.
- WEAVER works with float32 data type.
- WEAVER works with Keras Functional models.
- Mask RCNN based on implementation on https://github.com/matterport/Mask_RCNN with parameters: mode=inference, config.BACKBONE=resnet101/resnet50.
- YOLOv3 based on implementation on <https://github.com/experiencor/keras-yolo3> (inference model).
- U-Net based on original paper, "U-Net: Convolutional Networks for Biomedical Image Segmentation", Olaf Ronneberger and Philipp Fischer and Thomas Brox.