

tf.losses.sigmoid_cross_entropy

```
sigmoid_cross_entropy(  
    multi_class_labels,  
    logits,  
    weights=1.0,  
    label_smoothing=0,  
    scope=None,  
    loss_collection=tf.GraphKeys.LOSSES,  
    reduction=Reduction.SUM_BY_NONZERO_WEIGHTS  
)
```

Defined in [tensorflow/python/ops/losses/losses_impl.py](#).

Creates a cross-entropy loss using `tf.nn.sigmoid_cross_entropy_with_logits`.

weights acts as a coefficient for the loss. If a scalar is provided, then the loss is simply scaled by the given value. If **weights** is a tensor of shape `[batch_size]`, then the loss weights apply to each corresponding sample.

If **label_smoothing** is nonzero, smooth the labels towards 1/2:

```
new_multiclass_labels = multiclass_labels * (1 - label_smoothing)  
                      + 0.5 * label_smoothing
```

Args:

- multi_class_labels**: `[batch_size, num_classes]` target integer labels in `(0, 1)`.
- logits**: Float `[batch_size, num_classes]` logits outputs of the network.
- weights**: Optional **Tensor** whose rank is either 0, or the same rank as **labels**, and must be broadcastable to **labels** (i.e., all dimensions must be either `1`, or the same as the corresponding **losses** dimension).
- label_smoothing**: If greater than `0` then smooth the labels.
- scope**: The scope for the operations performed in computing the loss.
- loss_collection**: collection to which the loss will be added.
- reduction**: Type of reduction to apply to loss.

Returns:

Weighted loss **Tensor** of the same type as **logits**. If **reduction** is **NONE**, this has the same shape as **logits**; otherwise, it is scalar.

Raises:

- ValueError**: If the shape of **logits** doesn't match that of **multi_class_labels** or if the shape of **weights** is invalid, or if **weights** is None. Also if **multi_class_labels** or **logits** is None.

Stay Connected

- Blog
- GitHub
- Twitter

Support

- Issue Tracker
- Release Notes
- Stack Overflow

English

Terms | Privacy