

detection\_utils.core.losses  
\_test.SigmoidFocalClassification  
LossTest.testEasyExamplesProduceSmallLoss  
ComparedToSigmoidXEntropy

detection\_utils.core.losses  
\_test.SigmoidFocalClassification  
LossTest.testHardExamplesProduceLossComparable  
ToSigmoidXEntropy

detection\_utils.core.losses  
\_test.SigmoidFocalClassification  
LossTest.testIgnoreNegativeExampleLoss  
ViaAlphaMultiplier

detection\_utils.core.losses  
\_test.SigmoidFocalClassification  
LossTest.testIgnorePositiveExampleLoss  
ViaAlphaMultiplier

detection\_utils.core.losses  
\_test.SigmoidFocalClassification  
LossTest.testNonAnchorWiseOutputComparable  
ToSigmoidXEntropy

detection\_utils.core.losses  
\_test.\_logit

```
graph LR; A["detection_utils.core.losses  
_test.SigmoidFocalClassification  
LossTest.testEasyExamplesProduceSmallLoss  
ComparedToSigmoidXEntropy"] --> E["detection_utils.core.losses  
_test._logit"]; B["detection_utils.core.losses  
_test.SigmoidFocalClassification  
LossTest.testHardExamplesProduceLossComparable  
ToSigmoidXEntropy"] --> E; C["detection_utils.core.losses  
_test.SigmoidFocalClassification  
LossTest.testIgnoreNegativeExampleLoss  
ViaAlphaMultiplier"] --> E; D["detection_utils.core.losses  
_test.SigmoidFocalClassification  
LossTest.testIgnorePositiveExampleLoss  
ViaAlphaMultiplier"] --> E; F["detection_utils.core.losses  
_test.SigmoidFocalClassification  
LossTest.testNonAnchorWiseOutputComparable  
ToSigmoidXEntropy"] --> E;
```