

detection_utils.utils.config
_util_test.ConfigUtilTest.testAdam
OptimizerWithNewLearningRate

detection_utils.utils.config
_util_test.ConfigUtilTest.testMomentum
OptimizerWithNewLearningRate

detection_utils.utils.config
_util_test.ConfigUtilTest.testRMSProp
WithNewLearningRate

detection_utils.utils.config
_util_test.ConfigUtilTest.
_assertOptimizerWithNewLearningRate

```
graph LR; A[detection_utils.utils.config  
_util_test.ConfigUtilTest.testAdam  
OptimizerWithNewLearningRate] --> D[detection_utils.utils.config  
_util_test.ConfigUtilTest.  
_assertOptimizerWithNewLearningRate]; B[detection_utils.utils.config  
_util_test.ConfigUtilTest.testMomentum  
OptimizerWithNewLearningRate] --> D; C[detection_utils.utils.config  
_util_test.ConfigUtilTest.testRMSProp  
WithNewLearningRate] --> D;
```

The diagram illustrates a refactoring process. On the left, three separate test methods are listed in white boxes: `testAdamOptimizerWithNewLearningRate`, `testMomentumOptimizerWithNewLearningRate`, and `testRMSPropWithNewLearningRate`. Blue arrows from each of these boxes point to a single, larger gray box on the right. This gray box contains the code for a new assertion method, `_assertOptimizerWithNewLearningRate`, which is intended to replace the three individual test methods.