* Static approaches do not need the code to be executed again. Models built using static features have many advantages and are less costly.
* Pinto et al. built a set of predictors considering that some patterns within the test code may be employed to identify flaky tests automatically.
* The authors came to the conclusion that the vocabulary-based strategy performs poorly when used across projects because it is context-sensitive and prone to overfitting.
* Considering this result, an alternative approach for flaky test prediction based on test smells is used. Test smells are associated with potential design problems in the test code.
* Test smells are a deviation from how tests should be created, arranged, and interacted with one another. That deviation can indicate issues with test design and negatively impact test performance.
* An open-source test smell detection tool, tsDetect is used. For each test case, this tool requires the identification of the corresponding production code to detect the test smells.