

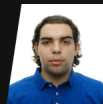
## CUSTOMER STORY

# DVC: Ensuring robustness of a popular machine-learning toolkit with DeepSource



"DeepSource is static code analysis for humans. Stop wasting your time setting up and maintaining CLI tools on CI, just use DeepSource"

— Ruslan Kupriev, Senior Software Engineer, DVC



806

issues fixed

593

security issues and anti-patterns patched

1.13M+

lines of code analyzed

## Background

DVC, created by [Iterative](#) is an open-source dataset and machine learning version control system designed to track the complete evolution of ML models, making it easy to switch back and forth between experiments. It makes the training process reproducible, sharable and easy to collaborate.

## Challenge

Data scientists have to switch between numerous time intensive experiments until they get the algorithm right. They use DVC to streamline this iterative process so they can make the switch instantaneously. A single critical bug in the tool can wreck the progress made on building the model which is why shipping quality, reliable code is taken very seriously at DVC.

While DVC has been using existing static analysis tools, they were on the lookout for a better tool that can:

- **Flag elusive issues** which go undetected easily otherwise or during manual code reviews
- **Report the most accurate issues** based on the context of their code to avoid dealing with irrelevant issues
- Blend with the existing development workflow so the team **doesn't lose track of issues** and can easily act on them

## Solution

### Native integration with GitHub giving a quick start

"The GitHub integration is flawless," said Ruslan, Sr. Software Engineer, DVC. From signing in with GitHub to installing DeepSource, the configuration is straight forward. It took him a few clicks and less than ten minutes to start reviewing the code.

### Discovering 'hard-to-spot' flaws in the source code

DeepSource analyzers detect 520+ type of potential security flaws, bug risks, anti-patterns along with other trivial issues (style & syntax) for Python. After integrating DeepSource, in the first analysis itself, Ruslan discovered over 200 complex issues in the codebase.

A few instances of the errors detected:

Assigning result of a function call, where the function returns None
PYL-E1128
MAJOR
BUG RISK

dvc/remotes/base.py

An assignment is done on a function call but the inferred function returns None. While this is allowed in Python, it is an anti-pattern and can lead to unexpected bugs.

298 if not self.exists(path\_info):
299 return None
300
301 checksum = self.state.get(path\_info)
302
303 # If we have dir checksum in state db, but dir cache file is lost,
304 # then we need to recollect the dir via .get\_dir\_checksum() call below,

Do not use len(SEQUENCE) to determine if a sequence is empty
PYL-C1801
MAJOR
PERFORMANCE

dvc/scm/git/\_init\_.py

Using the len function to check if a sequence is empty is not idiomatic and can be less performant than checking for truthness of the sequence.

len doesn't know the context in which it is called, so if computing the length means traversing the entire sequence, it must; it doesn't know that the result is just being compared to 0. Computing the boolean value can stop after it sees the first element, regardless of how long the sequence actually is.

95
96 entry = os.path.relpath(path, ignore\_file\_dir).replace(os.sep, "/")
97 # NOTE: using '/' prefix to make path unambiguous
98 if len(entry) > 0 and entry[0] != "/":
99 entry = "/" + entry
100
101 gitignore = os.path.join(ignore\_file\_dir, self.GITIGNORE)

In addition, there are some issues that reviewers know of and don't need to be fixed. To avoid flagging such issues, DVC used DeepSource's 'This violation is intentional' feature to tweak the analysis. It helped the team focus only on important warnings.

### Quality checks within the workflow to keep track of issues easily

Issues found during reviews can slip through the cracks and end up in production if not logged properly, be it due to human error or lack of proper tools. Since DeepSource flags issues directly in the pull requests & commits alongside the CI/CD checks, keeping track of all the issues detected and acting on them, without missing any, has been a smooth sailing for the team.

## Results

From the past six months, DVC has been reviewing all their PRs using DeepSource which helped them ship quality code to production with minimal errors.