Down The Dependency Rabbit Hole

Machine Learning as a first line of defense in Intel's Dependency Review Process

$ env
NAME=John Andersen
HOME=Portland OR
USER=pdxjohnny
WORK=Intel
ROLE=Open Source Security Software Engineer
EMAIL=john.s.andersen@intel.com
INTERESTS=Embedded, Linux, Containers, Concurrency, Web Apps, Python
THESIS=Machine Learning

Dependency Evaluation Review Form

SECTION	GRADE	GRADING GUIDELINES
First look		A - Mentions security audit or other proactive security activity. B - No major warning signs, and code is used professionally. C - No major warning signs, but not widely used or not well-supported. D - Code has minor warning signs that need to be investigated in more detail. F - Code has known issues, major warning signs, or is abandoned
Contributors and activity		A - At least five significant, active contributors. B - More than two significant, active contributors. C - Only one major contributor who is active. D - Project has been inactive for nine months or less. F - Project has been inactive for more than one year.
Security issues		A - Project has had previous security issues and handled them quickly and well. Bonus if they also mention doing proactive security such as fuzz testing, static analysis, or security audits. B - Project has a plan for handling security issues but hasn't had to use it much yet. C - Project does not have a plan for security issues but at least has an active bug tracker and issues get resolved. D - Project does not seem to resolve many open bugs. F - Project has open security issues that are not in the process of being resolved.
Test suite		A - Project has test suite with good coverage of positive and negative test cases set up as part of continuous integration, and test results are published for each build. B - Project has test suite with good coverage but no continuous integration. C - Test suite mostly covers positive test cases; very few or no error cases. D - Test suite has very low coverage or is only a few examples. F - No test suite.

Automation Attempt One

- Initial dataset is made up of
  - URL of source repo
  - Security team's classification (Good / Bad)
  - Review form data
- Plan
  - Train model on dataset
    - Assess accuracy
  - Given URL, collect data to answer form questions
  - Predict classification by feeding collected data to model

Reviewers Rarely Fill Out Evaluation Form

SECTION	GRADE	GRADING GUIDE
First look		A - Mentions … audit or other proactive security activity. B - No … warning signs, and code is used professionally. C - … major warning signs, but not widely used or not well-supported. … ode has minor warning signs that need to be investigated in more detail. … Code has known issues, major warning signs, or is abandoned
Contributors and activity		A - At least five significant, active contribut… B - More than two significant, active contri… C - Only one major contributor who is acti… D - Project has been inactive for ni… m…hs or less. F - Project has been ina…e … mor… one year.
Security issues		A - Project has had pr… …ecu… issues and handled … …kly and well. Bonus if they al… …ention doing proactive security such as fuzz testi… …c analysis, or secu… …dits. B - Project has a plan for handling security i…ues but … …ad to use it much yet. C - Project does not have a plan for security i…s but at least has an active bug tracker and i… …s get resolved. D - Project does not seem to resolve m… …n bugs. F - Project has open security … …hat a… …n the process of being resolved.
Test suite		A - Project has test suite w… good co…rage of positive and negative test cases set up a… …rt of continuous integration, and … results are published for e… …d. B - …ject has test suite with good coverage but no continuous integration. C - Te… …e mostly covers positive test cases; very few or no error cases. D - Test sui… …very low coverage or is only a few examples. F - No test suite.

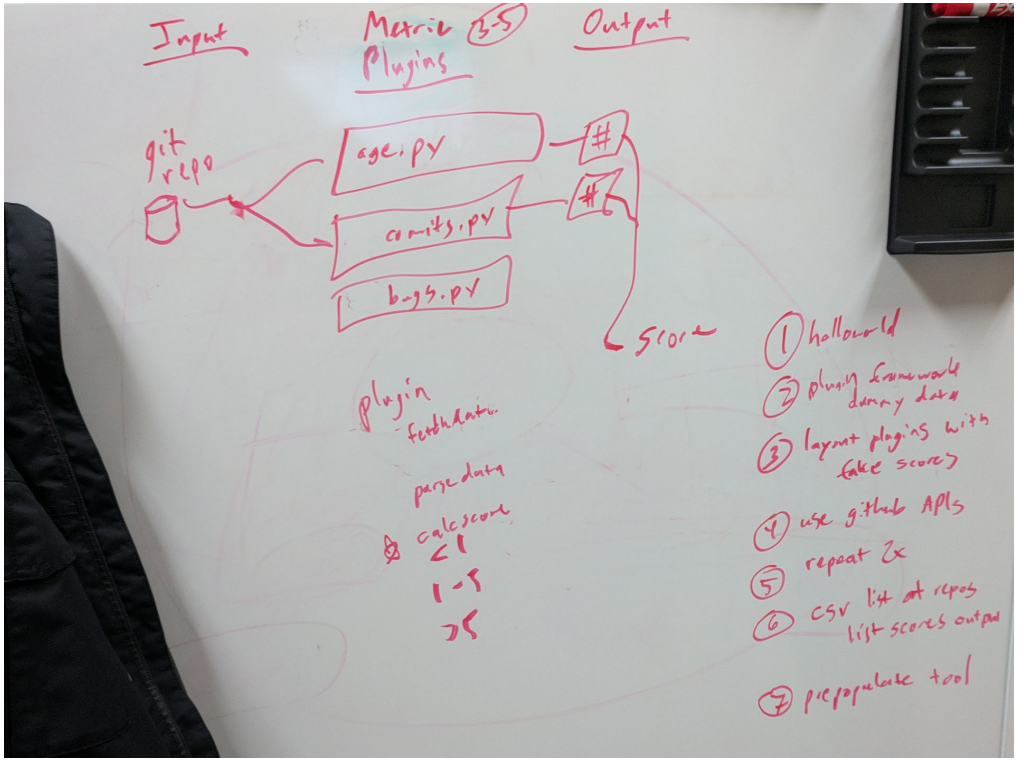
Automation Attempt One

- Initial dataset is made up of
  - URL of source repo
  - Security team's classification (Good / Bad)
  - Forms mostly not filled out 👎
- Plan
  - Train model on dataset
    - Assess accuracy
  - Given URL, collect data to answer form questions
  - Predict classification by feeding collected data to model
- ~60% Accuracy 🚫

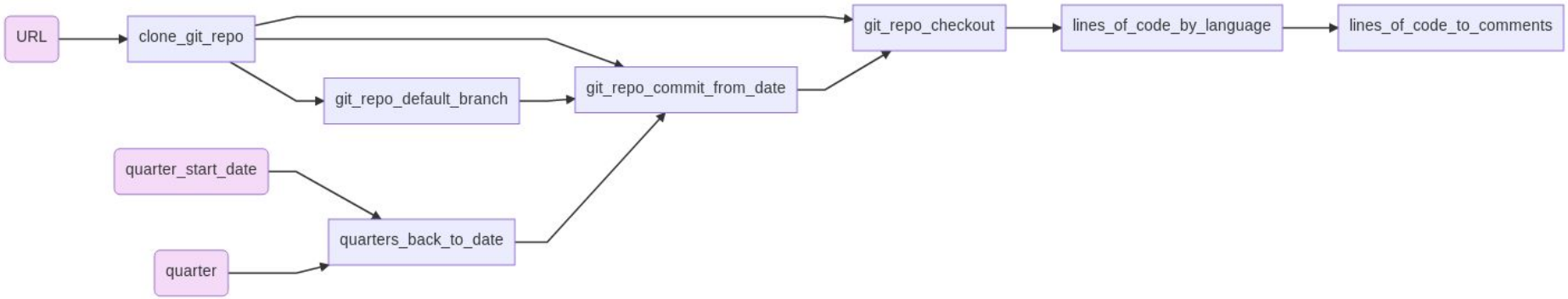
Automation Attempt Two

- Initial dataset is made up of
  - URL of source repo
  - Security team's classification (Good / Bad)
- Plan
  - Given URL, collect data
  - Train model on dataset
    - Assess accuracy
  - Predict classification by feeding collected data to model

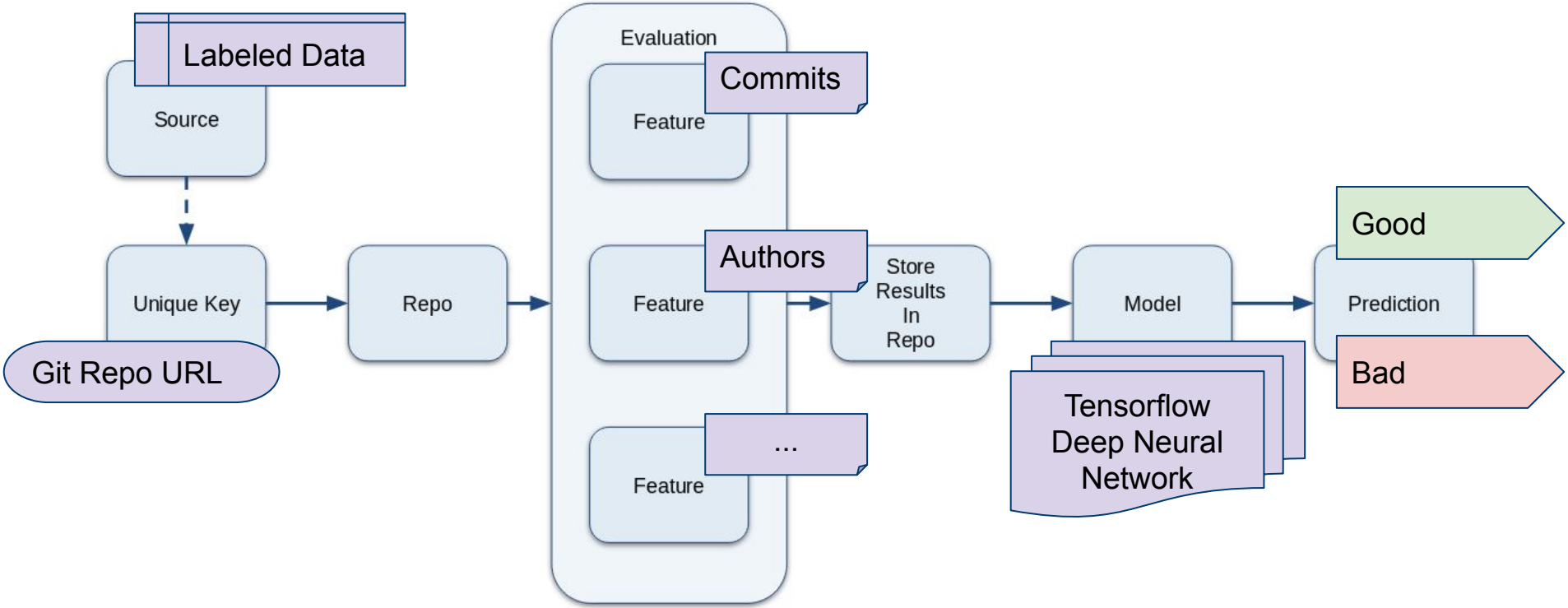
Brainstorming



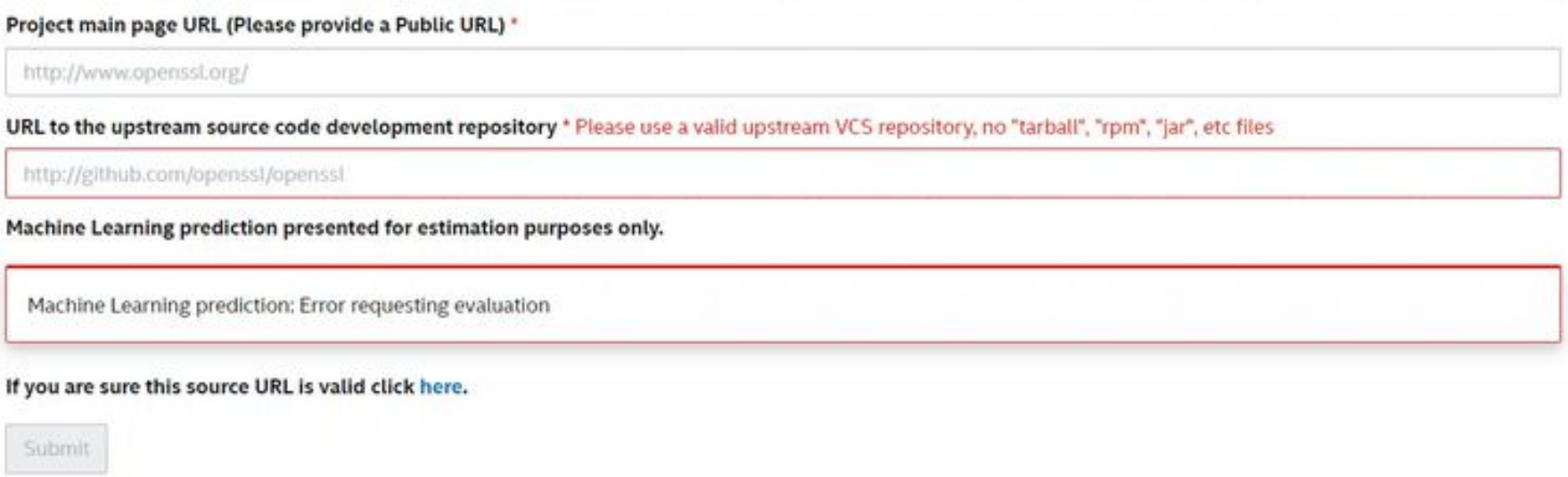
Quarterly Ratio of Lines of Comments to Code



Prediction Data Flow



Request Classification Estimation



Data Flow Facilitator for Machine Learning

Machine Learning made easy

Abstractions DFFML Provides

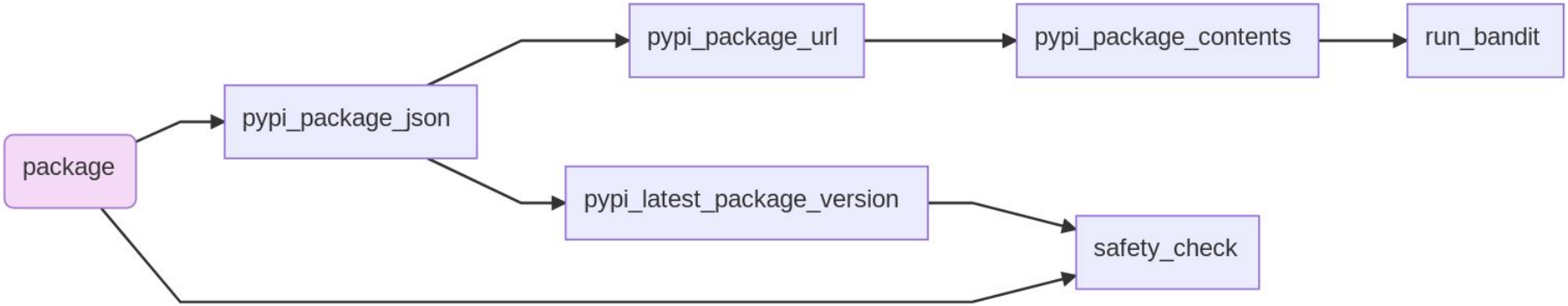
- Data Flow
  - Dataset generation
  - Concurrency without dealing with locking
- Sources
  - CSV
  - JSON
  - MySQL
- Models
  - Tensorflow
  - SciKit

Consistent API

- Python Library
- Command Line Interface
- HTTP API

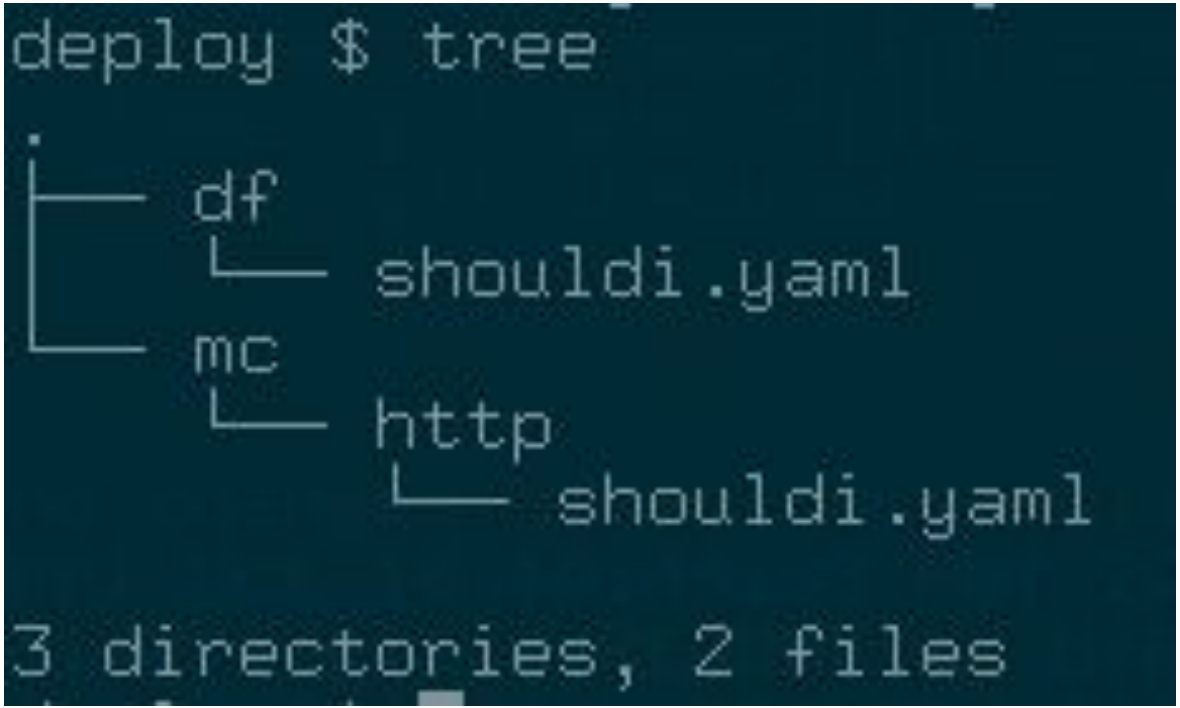
Should I Be Installing This?

What is a Data Flow?



Deploy Anywhere - Command Line

Deploy Anywhere - HTTP



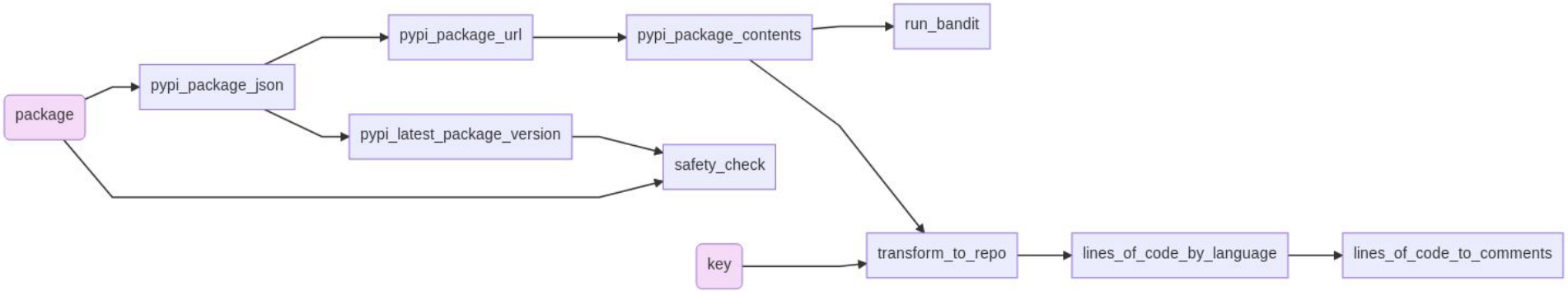
Deploy Anywhere - HTTP

```
deploy $ cat mc/http/shouldi.yaml
path: /shouldi
presentation: json
asynchronous: false
deploy $
```

```
flow:
  bandit:
    pkg:
    - pkg.contents.directory
  pkg.cleanup:
    directory:
    - pkg.contents.directory
  pkg.contents:
    url:
    - pkg.url.url
  pkg.json:
    package:
    - seed
  pkg.url:
    response_json:
    - pkg.json.response_json
  pkg.version:
```

Deploy Anywhere - HTTP

Extend Without Writing Code - Modify DataFlow



Where To Go From Here

- How to Integrate Machine Learning Tutorial
  - https://intel.github.io/dffml/usage/integration.html
- shouldi
  - pip install shouldi && shouldi install some-package-name
  - https://intel.github.io/dffml/tutorials/operations.html
- Use and Contribute!
  - Weekly Meetings: Tuesdays at 9 AM
  - Gitter, Mailing List, and Meeting Links: https://intel.github.io/dffml/community.html
  - Documentation: https://intel.github.io/dffml
- Q&A