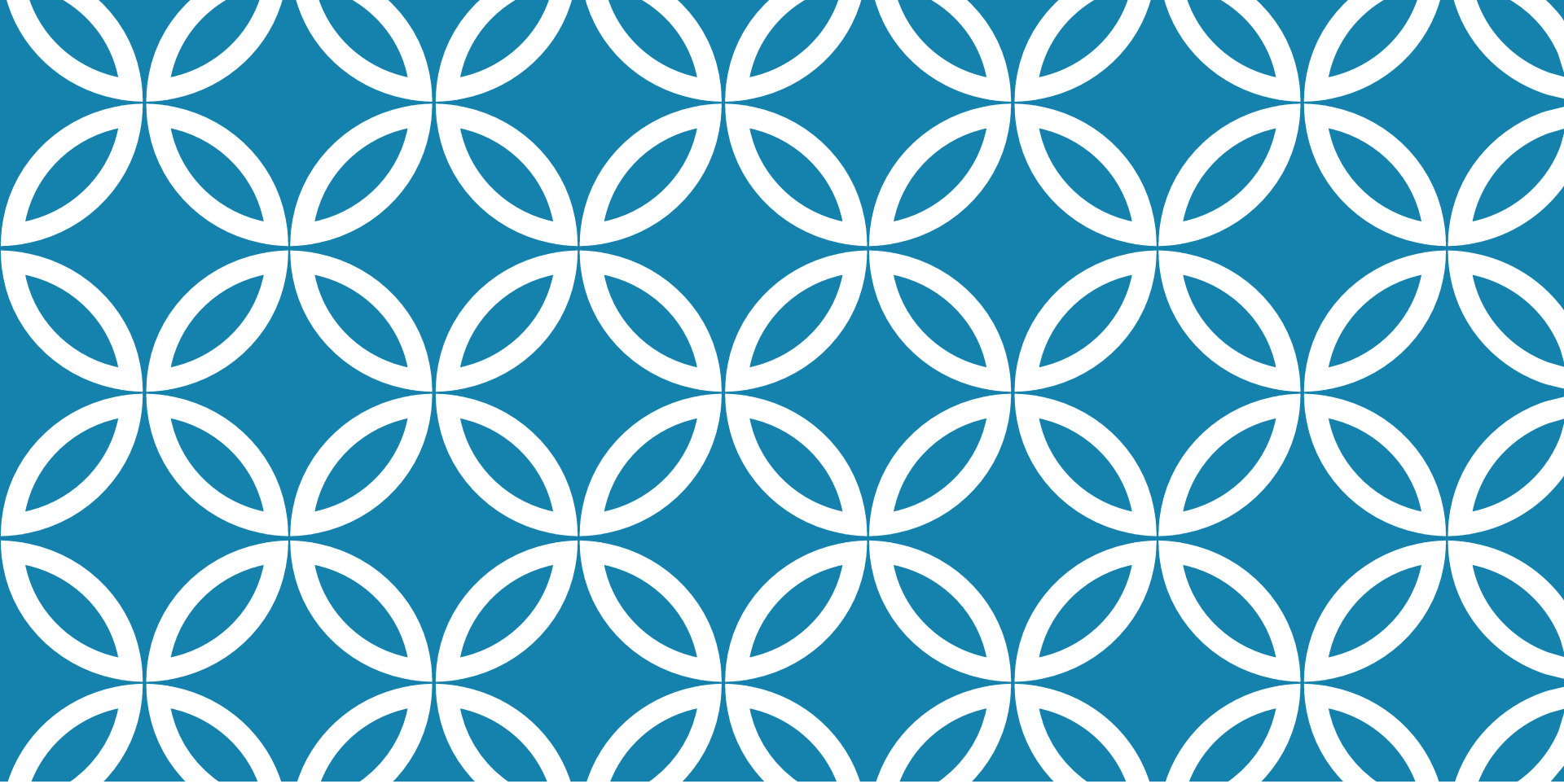


THE CURIOUS CASE OF CODE DUPLICATION IN GITHUB

And other duplicate detection adventures

Crista Lopes, UC Irvine








Crista **Lopes** Pedro **Martins** Vaibhav **Saini** Di **Yang** Hitesh **Sajnani** Petr **Maj** Jakub **Zitny** Jan **Vitek**

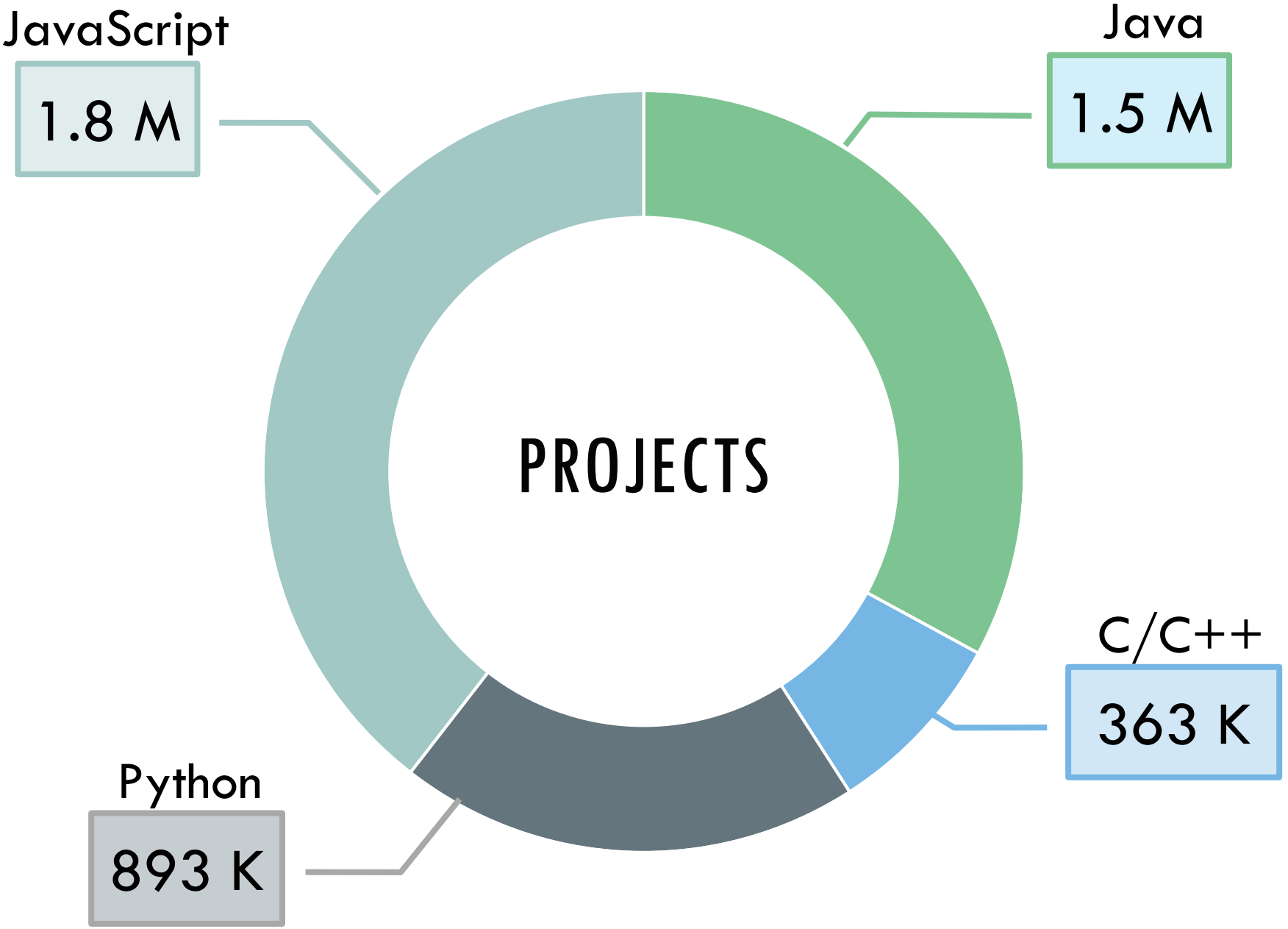
UC Irvine **CTU Prague** **Northeastern**

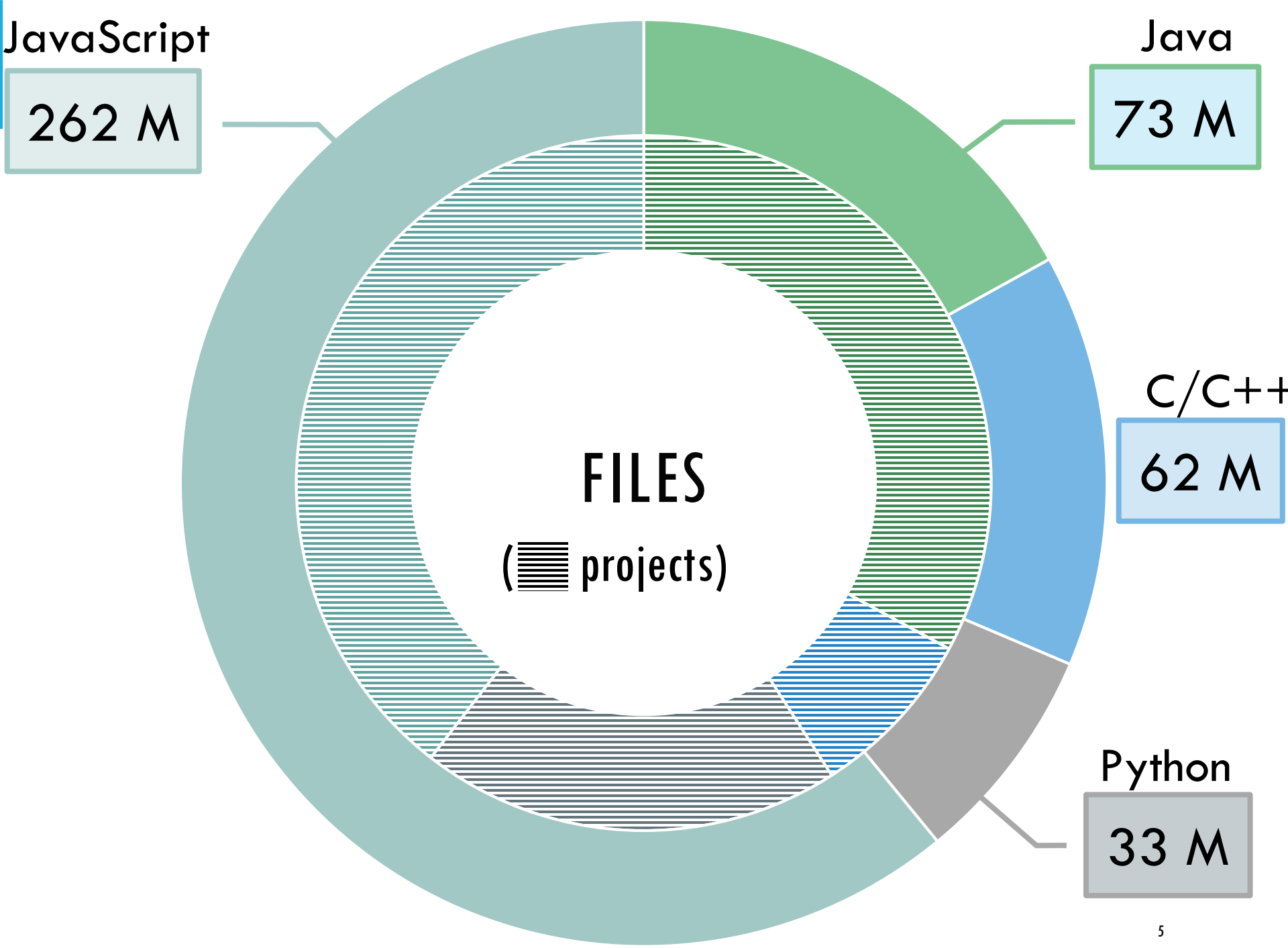
GITHUB

The clone map as of 2017

HOW MANY DUPLICATES ARE THERE ON GITHUB?

(MOTIVATION: AVOID SELECTION BIAS IN STUDIES AND TOOLS)


JavaScript
1.8 M
Java
1.5 M
PROJECTS
C/C++
363 K
Python
893 K


JavaScript
262 M
Java
73 M
C/C++
62 M
Python
33 M
FILES
(projects)

262,000,000

68,644,000,000,000,000

10 MS

21,766,869 YEARS

FILE 1

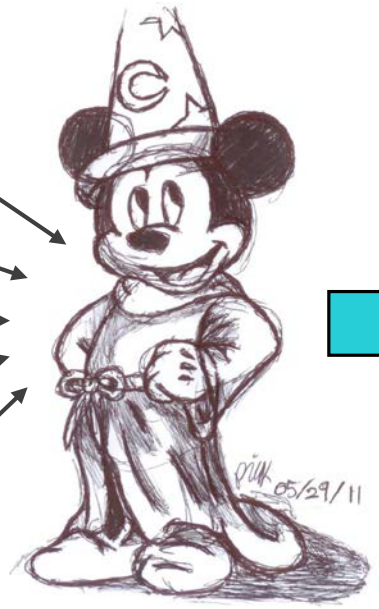
```
void : 1
foobar : 1
int : 3
foo : 3
bar : 3
baz : 3
if : 1
42 : 1
return : 2
else : 1
```

FILE 4

```
double : 3
16 : 1
hello : 3
for : 1
x : 7
y : 3
```

FILE 2

```
for : 1
while : 1
xy : 3
78 : 3
89 : 3
1 : 1
3 : 1
```



SourcererCC

1, 3
2, 5

FILE 3

```
void : 1
foobar : 1
int : 2
double : 1
foo : 3
bar : 3
baz : 3
if : 1
43 : 1
return : 2
else : 1
```

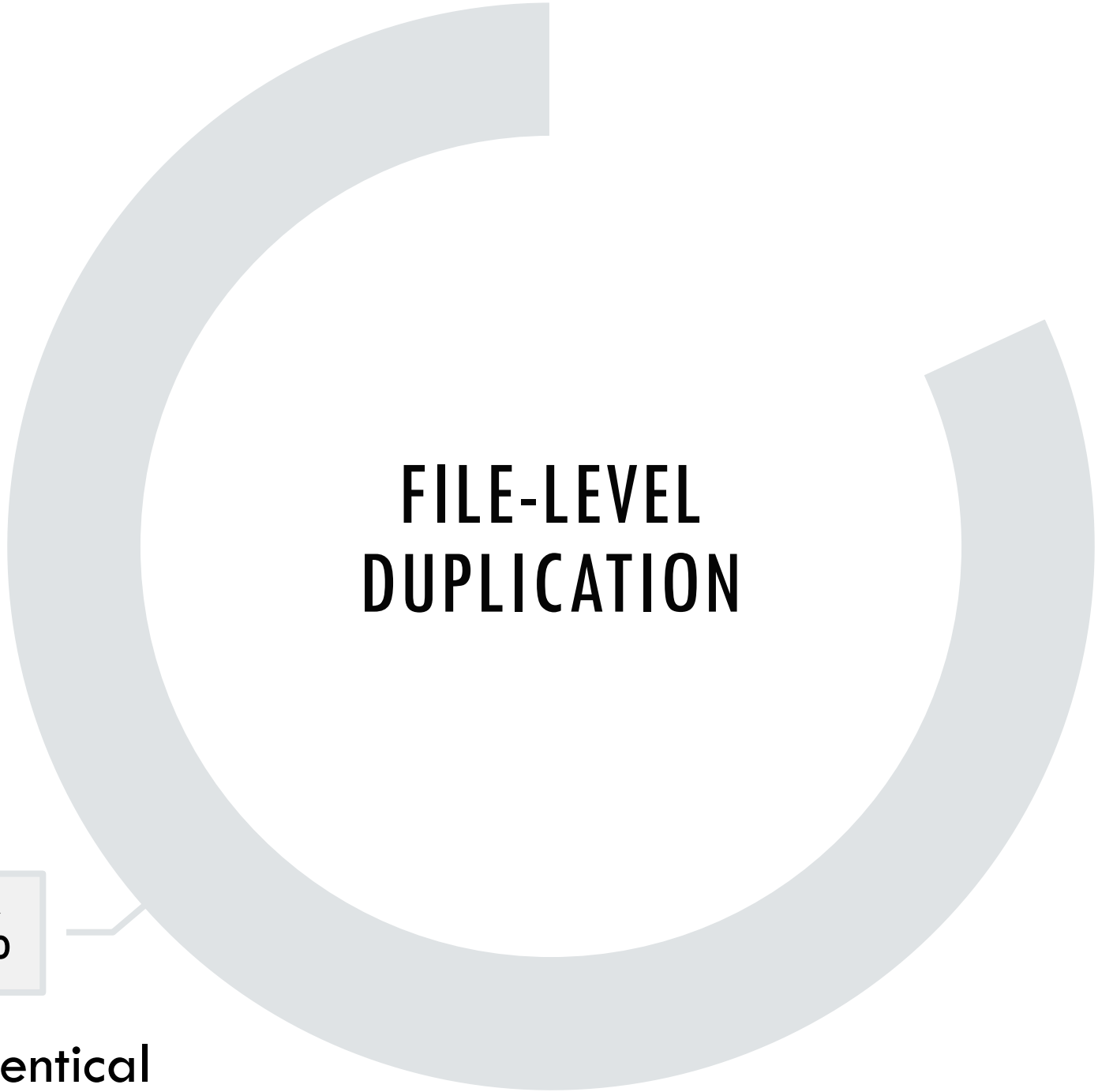
FILE 5

```
double : 3
for : 2
while : 1
xy : 3
78 : 3
89 : 3
1 : 1
3 : 1
```

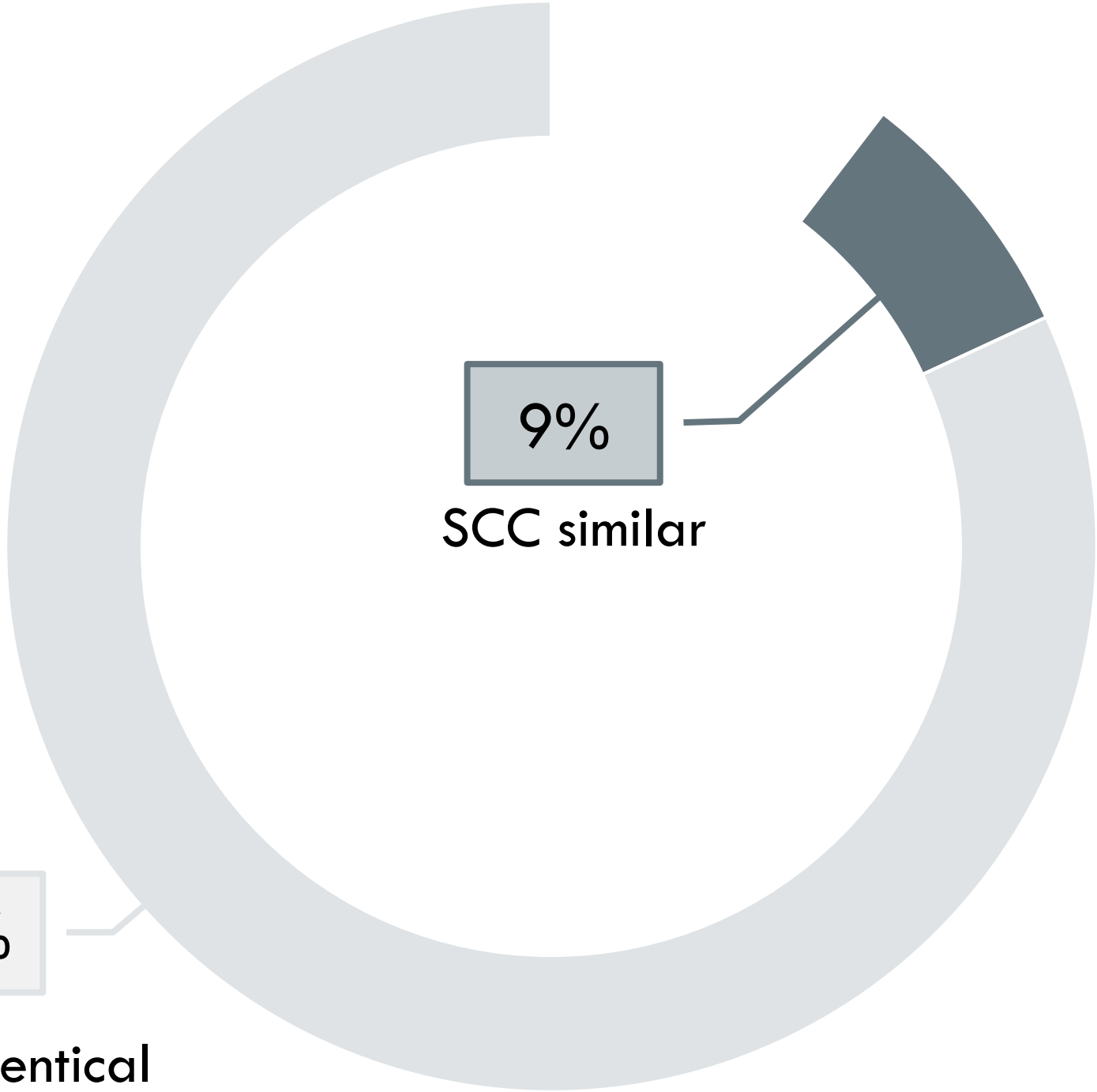
mondego.ics.uci.edu/projects/SourcererCC

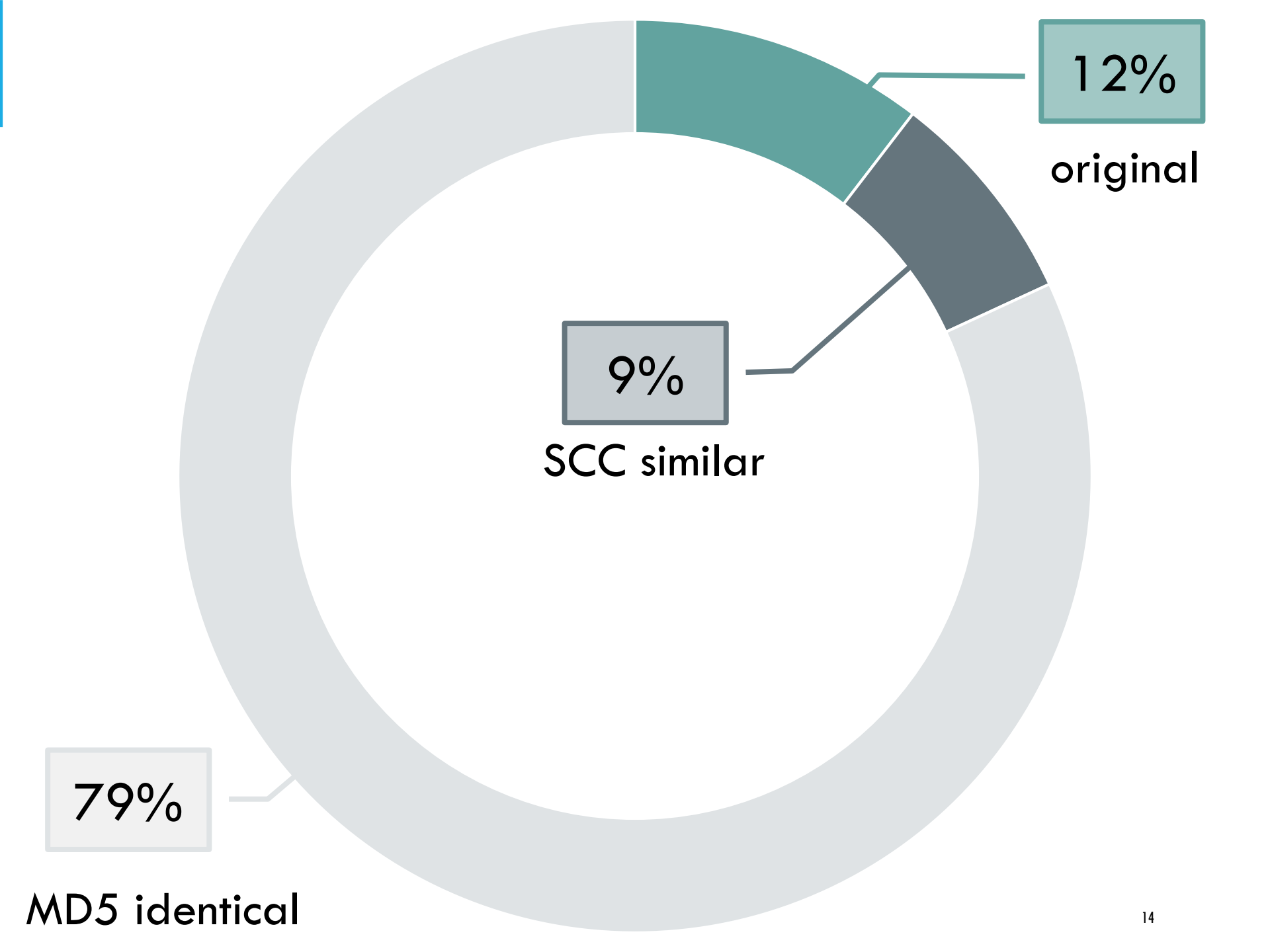
9 MONTHS LATER

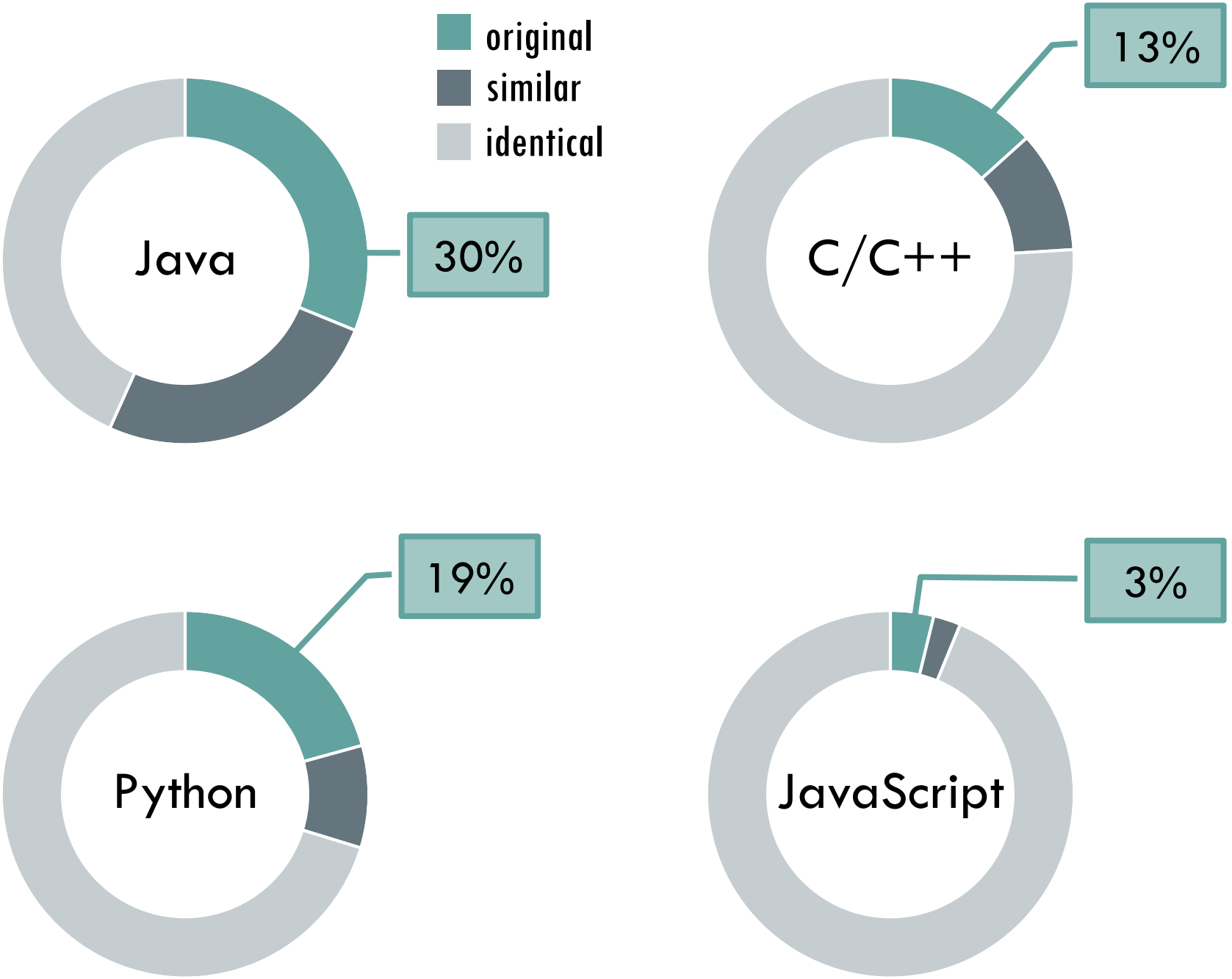
FILE-LEVEL DUPLICATION

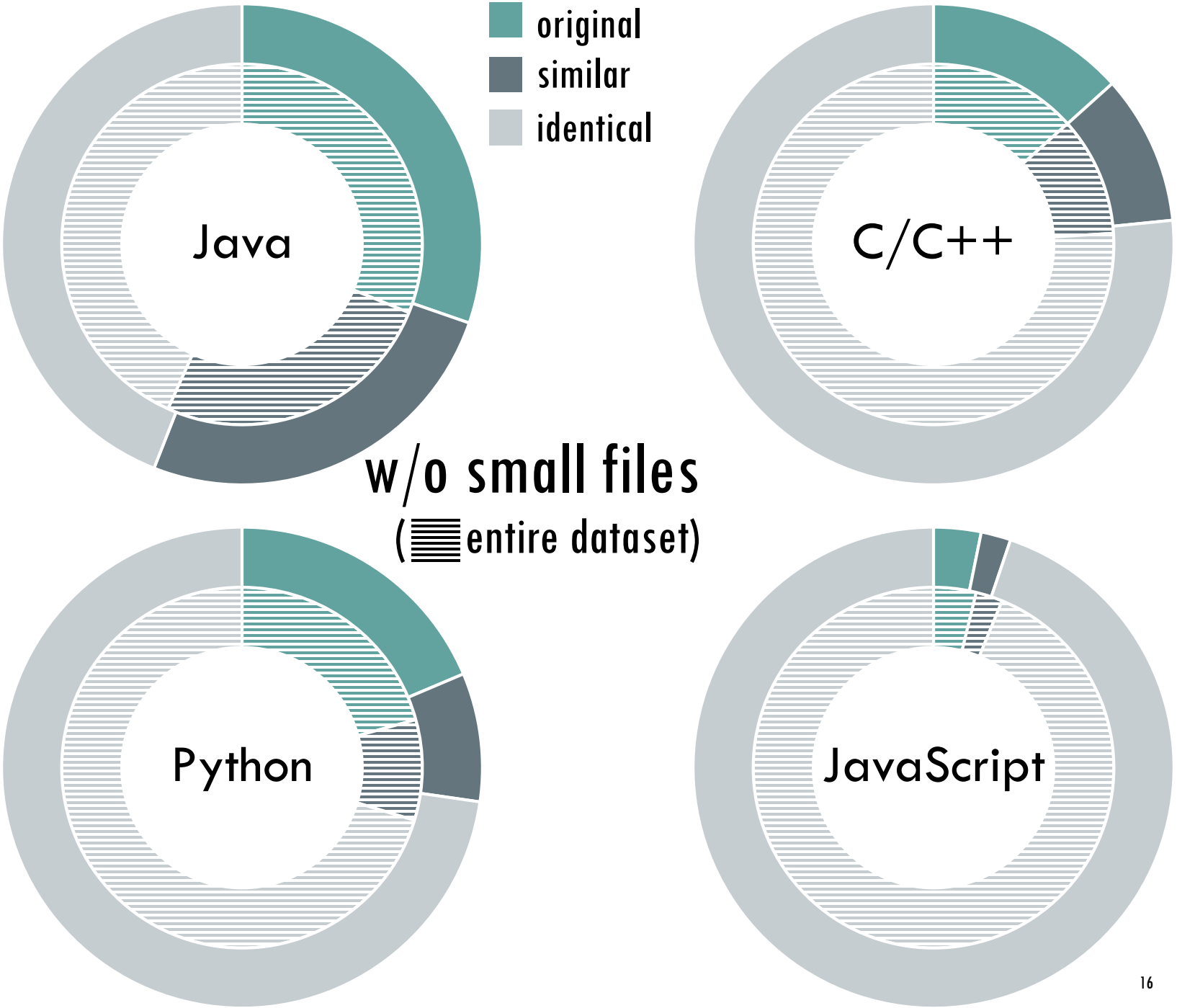


MD5 identical

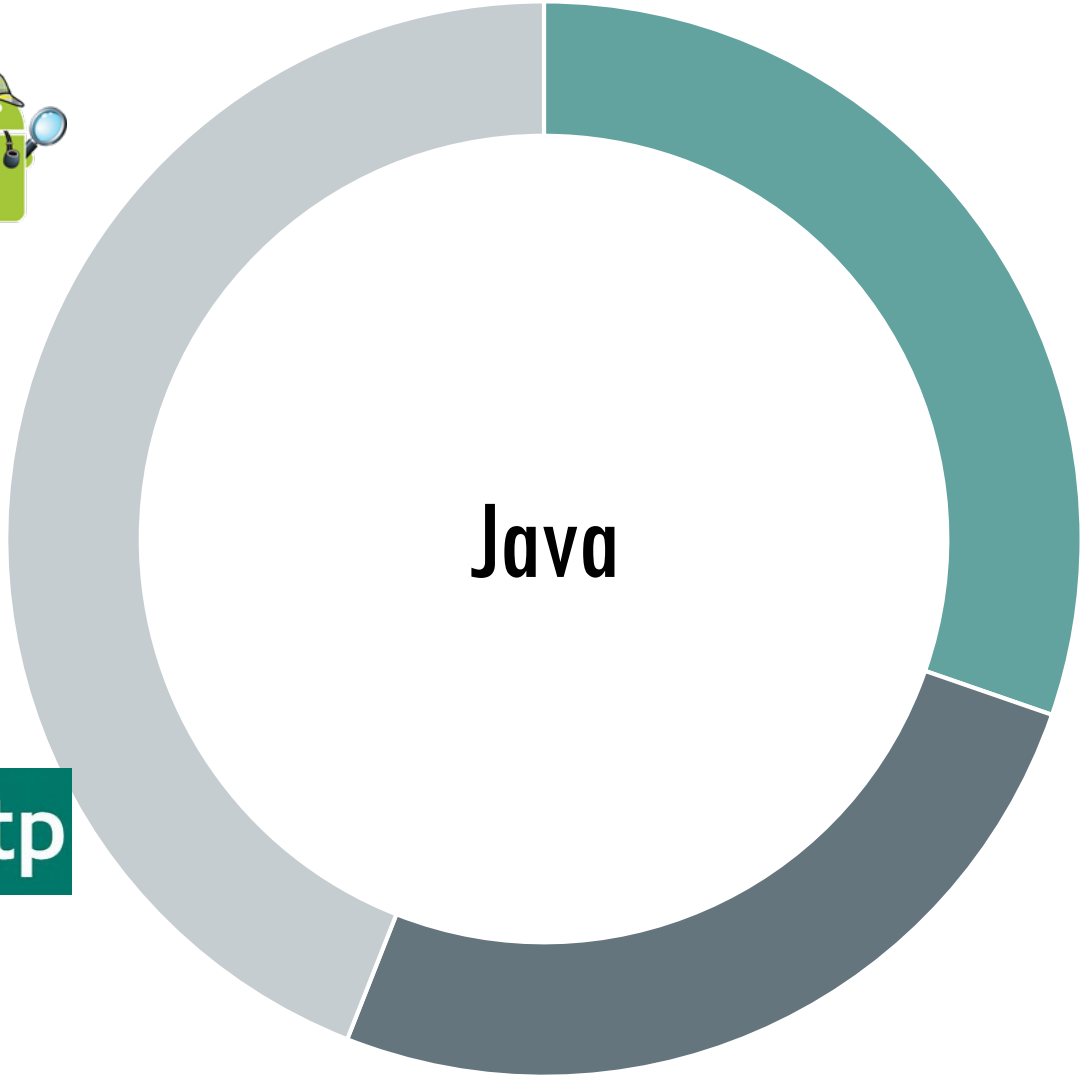

9%
SCC similar
79%
MD5 identical


12%
original
9%
SCC similar
79%
MD5 identical

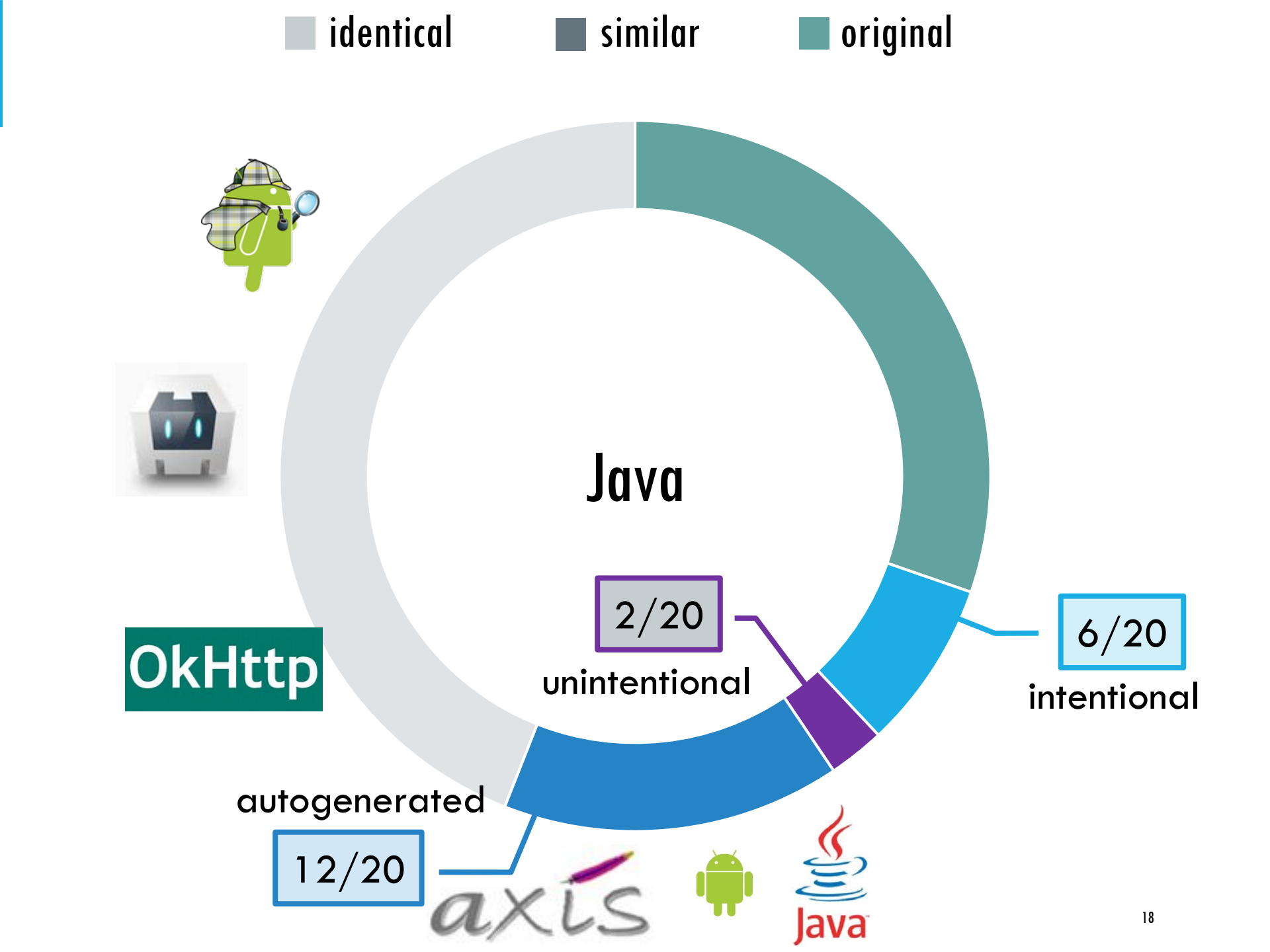

original
similar
identical
Java
30%
C/C++
13%
Python
19%
JavaScript
3%

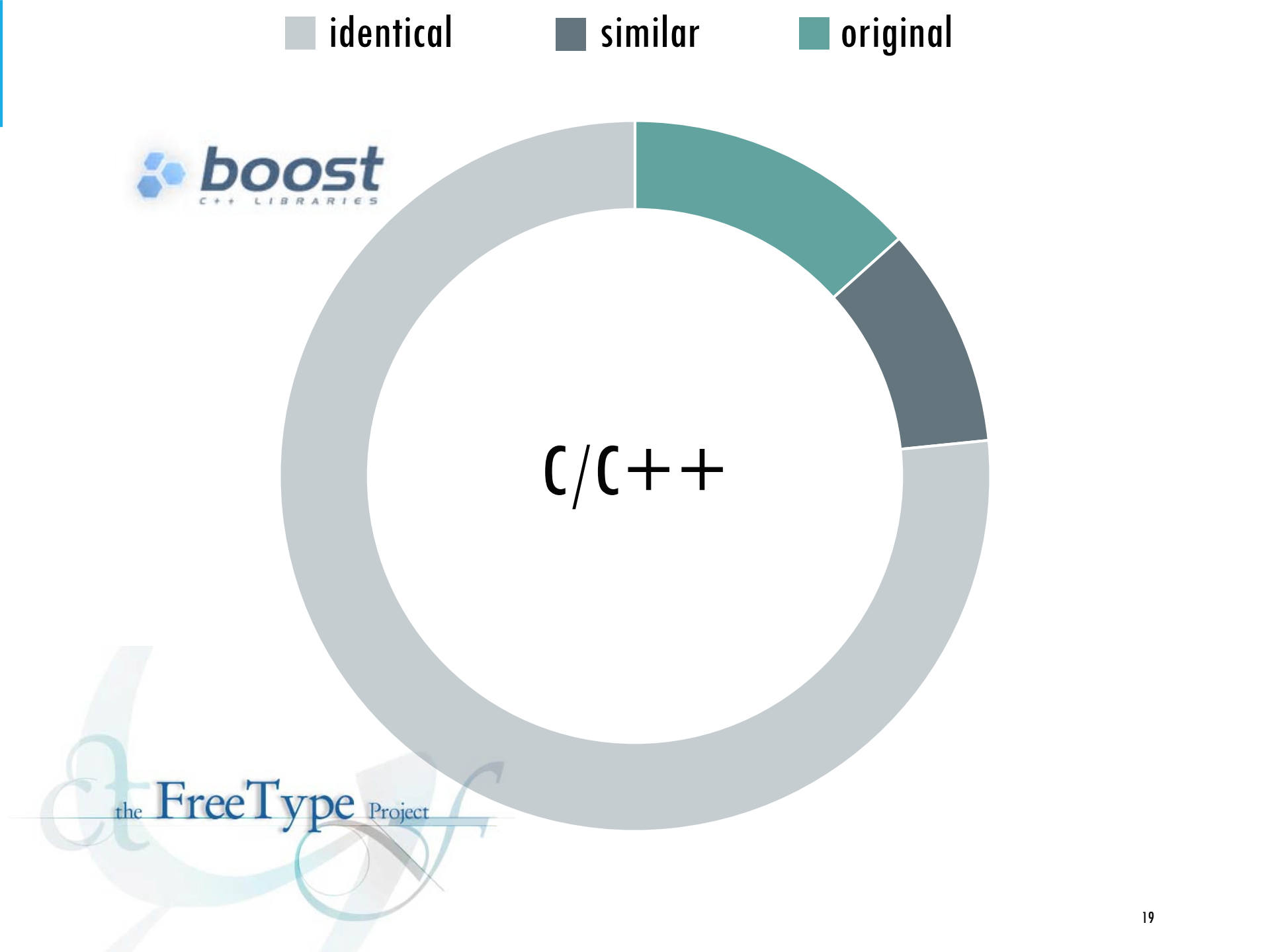

original
similar
identical
Java
C/C++
w/o small files
(entire dataset)
Python
JavaScript

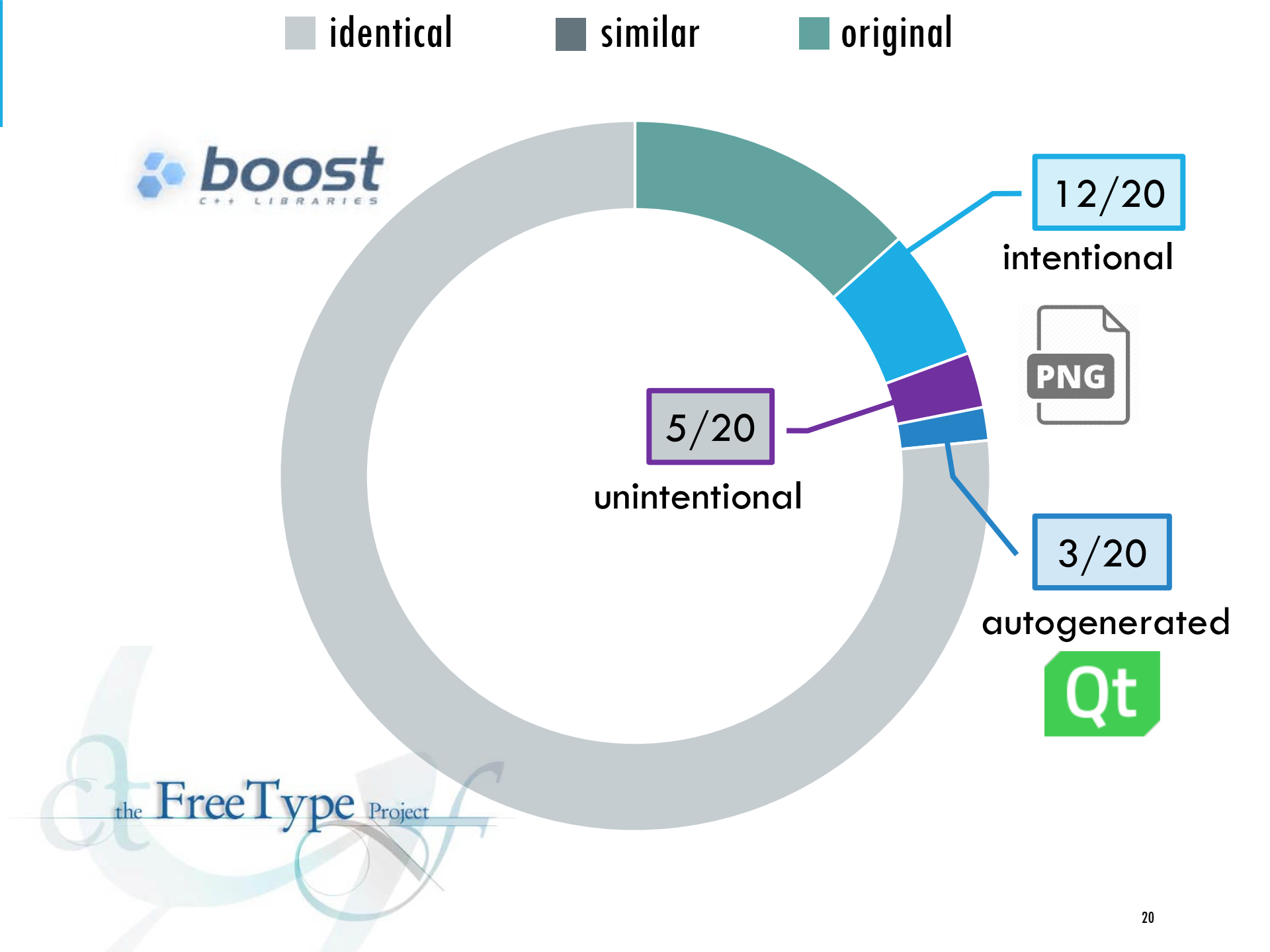
identical similar original

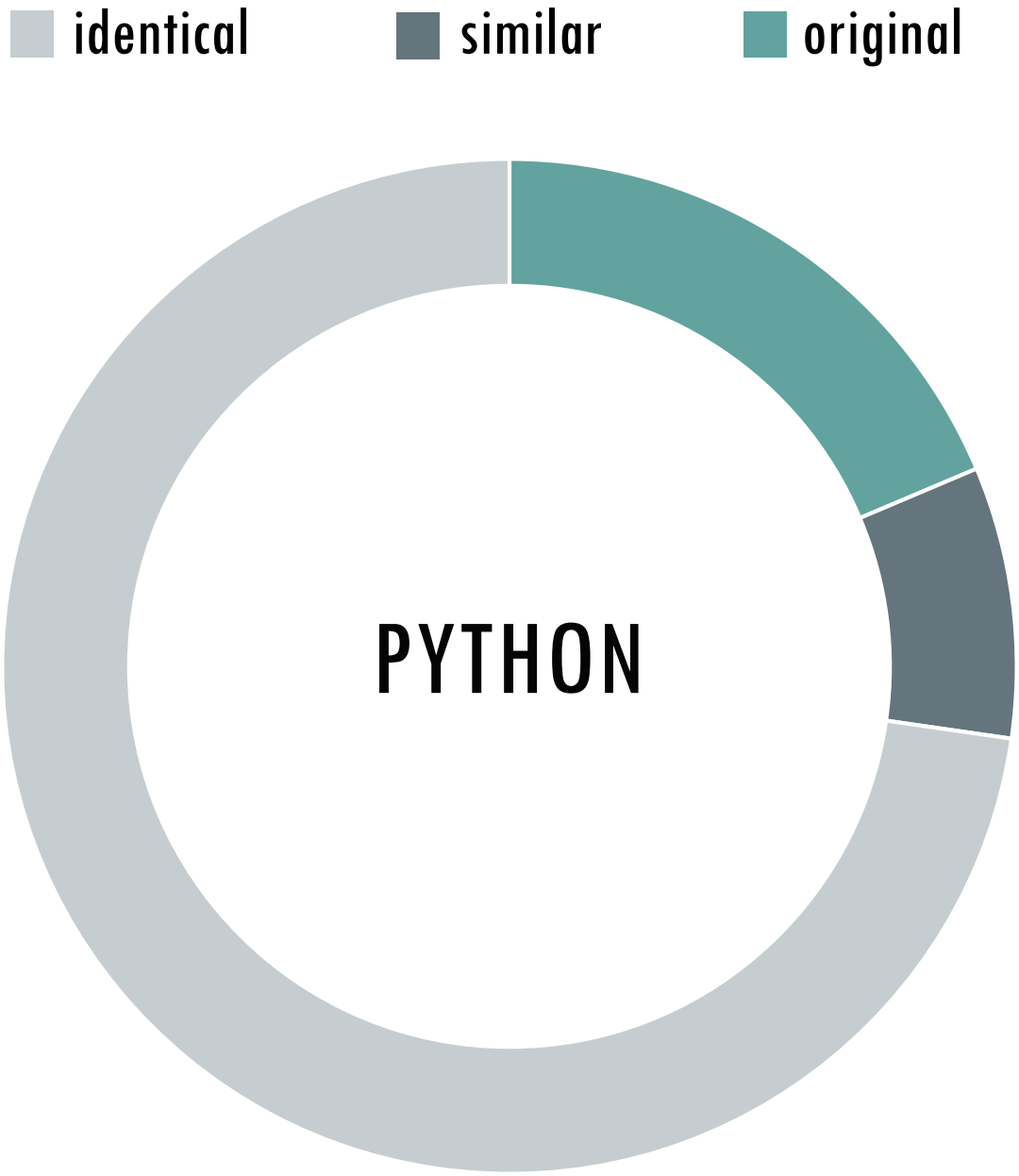


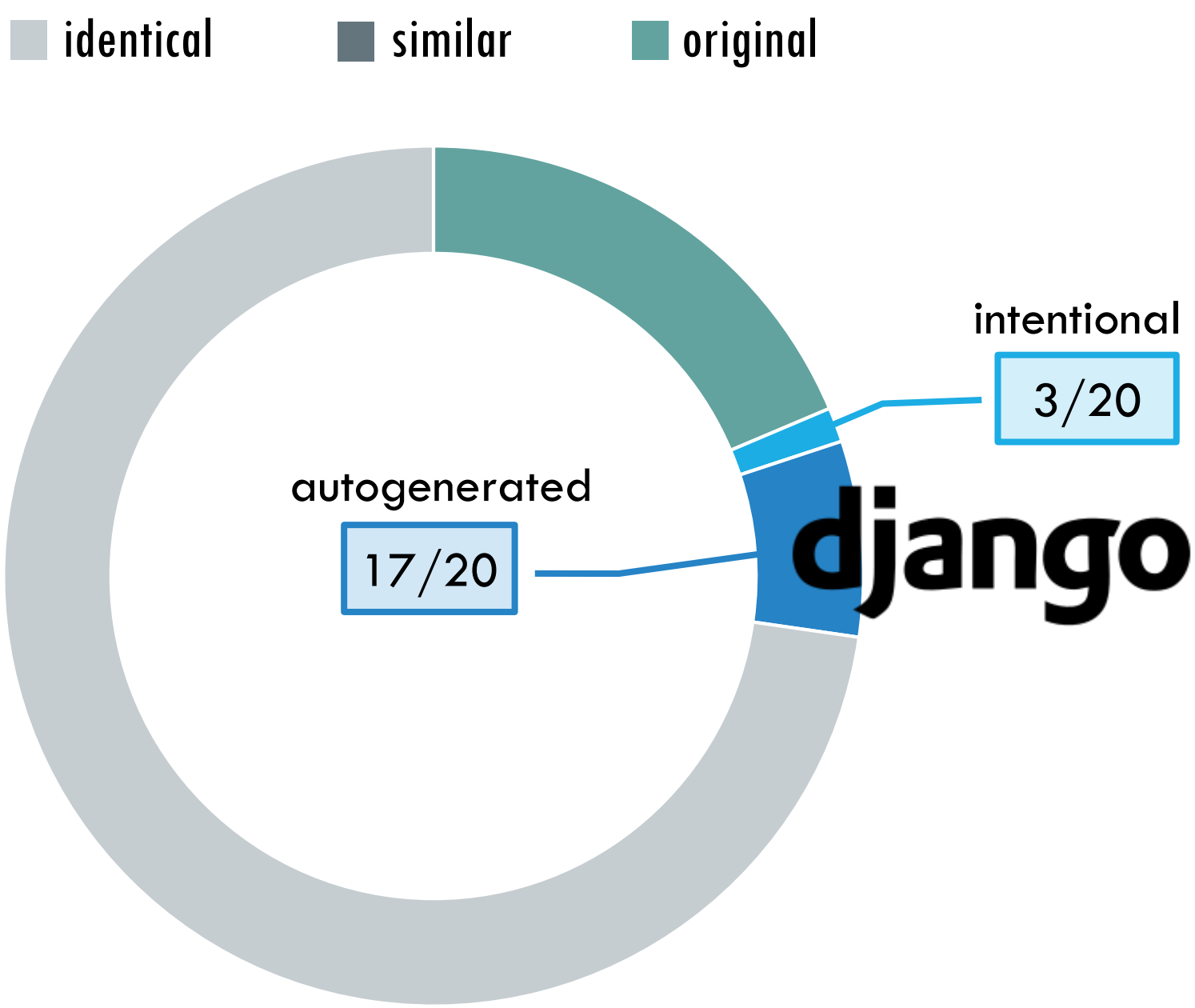
OkHttp


identical
similar
original
Java
2/20
unintentional
6/20
intentional
OkHttp
autogenerated
12/20
axis
Java

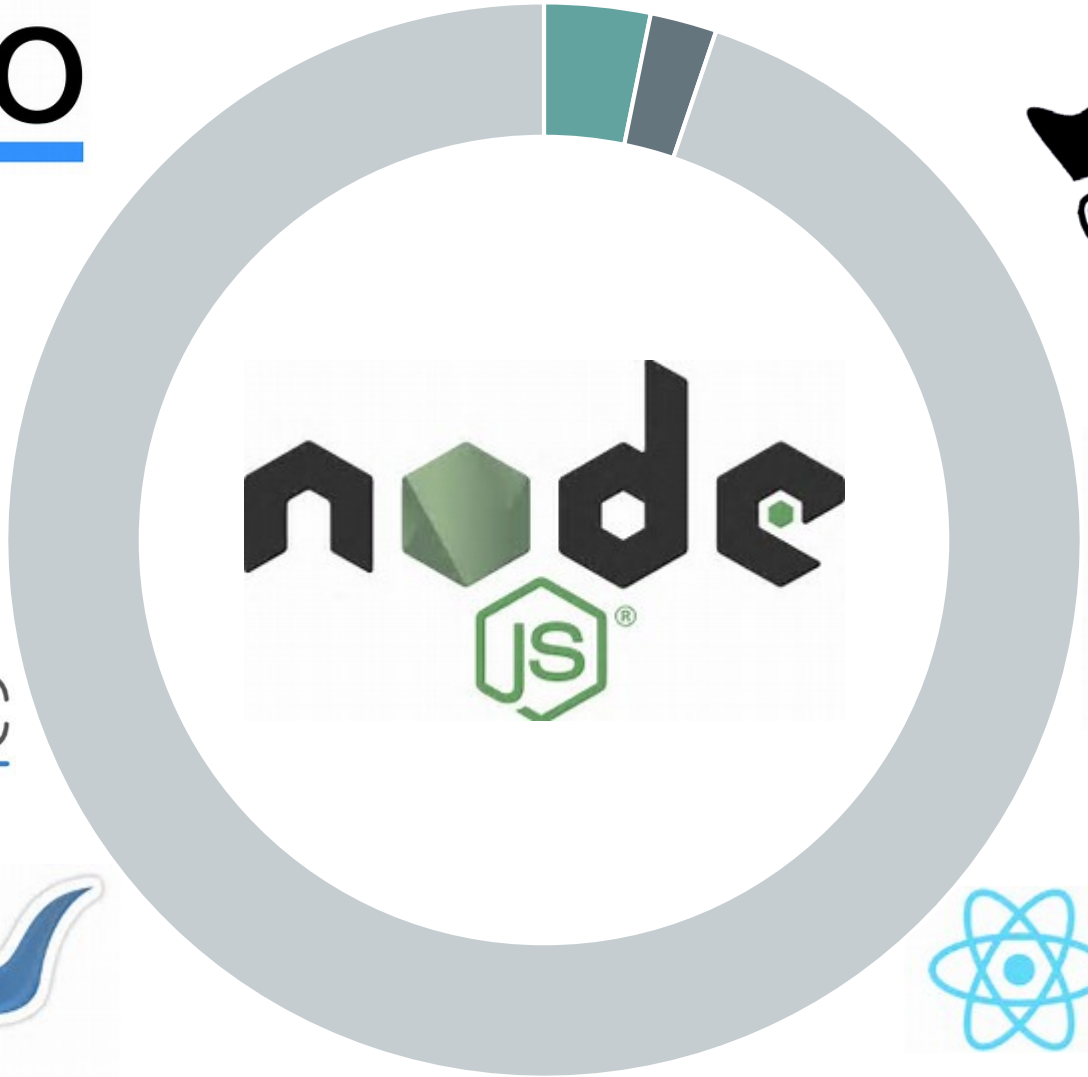

identical
similar
original
C/C++


identical
similar
original
boost
C++ LIBRARIES
12/20
intentional
PNG
5/20
unintentional
3/20
autogenerated
Qt
the FreeType Project


identical
similar
original
PYTHON


identical
similar
original
intentional
3/20
autogenerated
17/20
django

identical similar original



Lo

CHALK

express

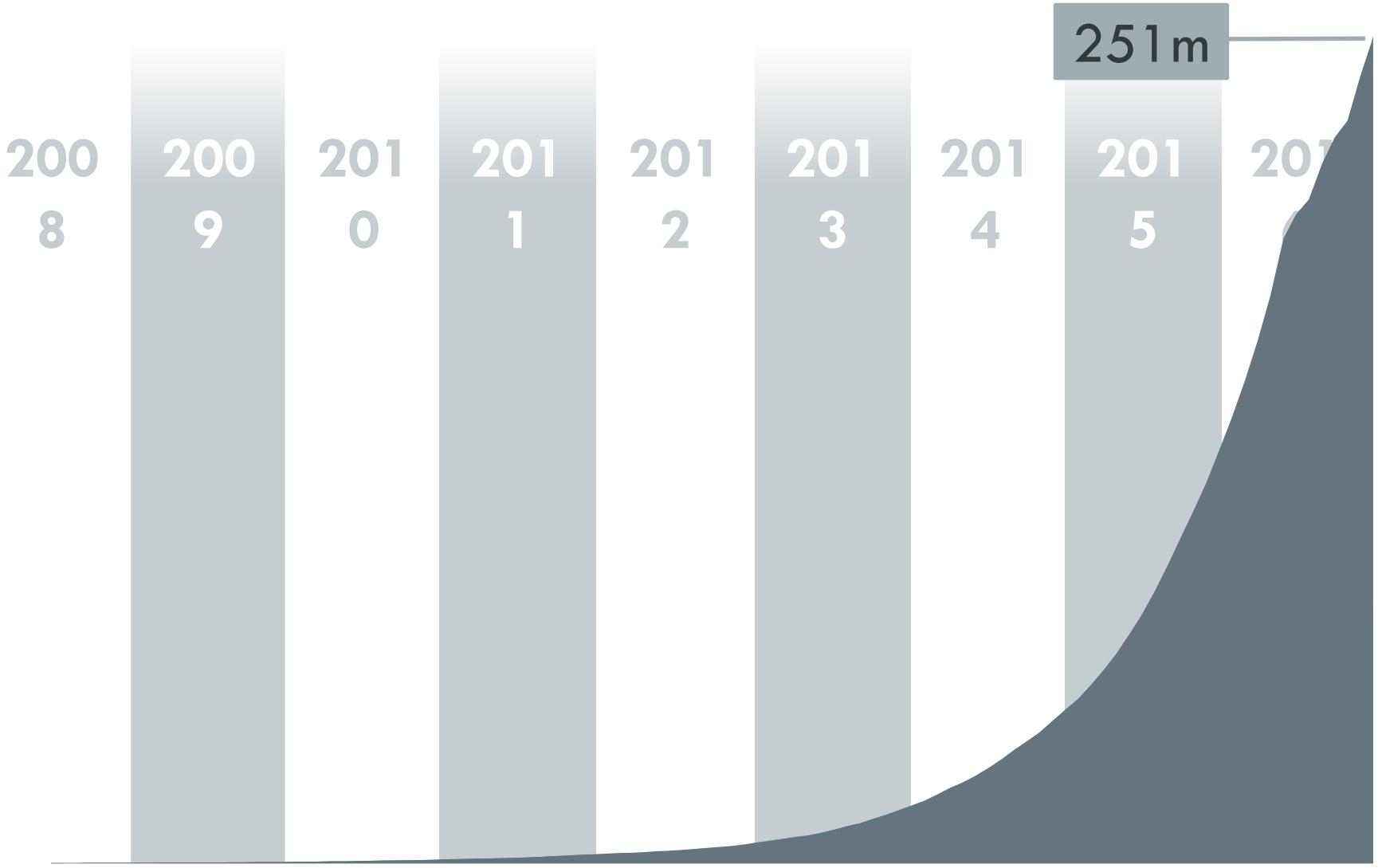
async

UNDERSCORE.JS

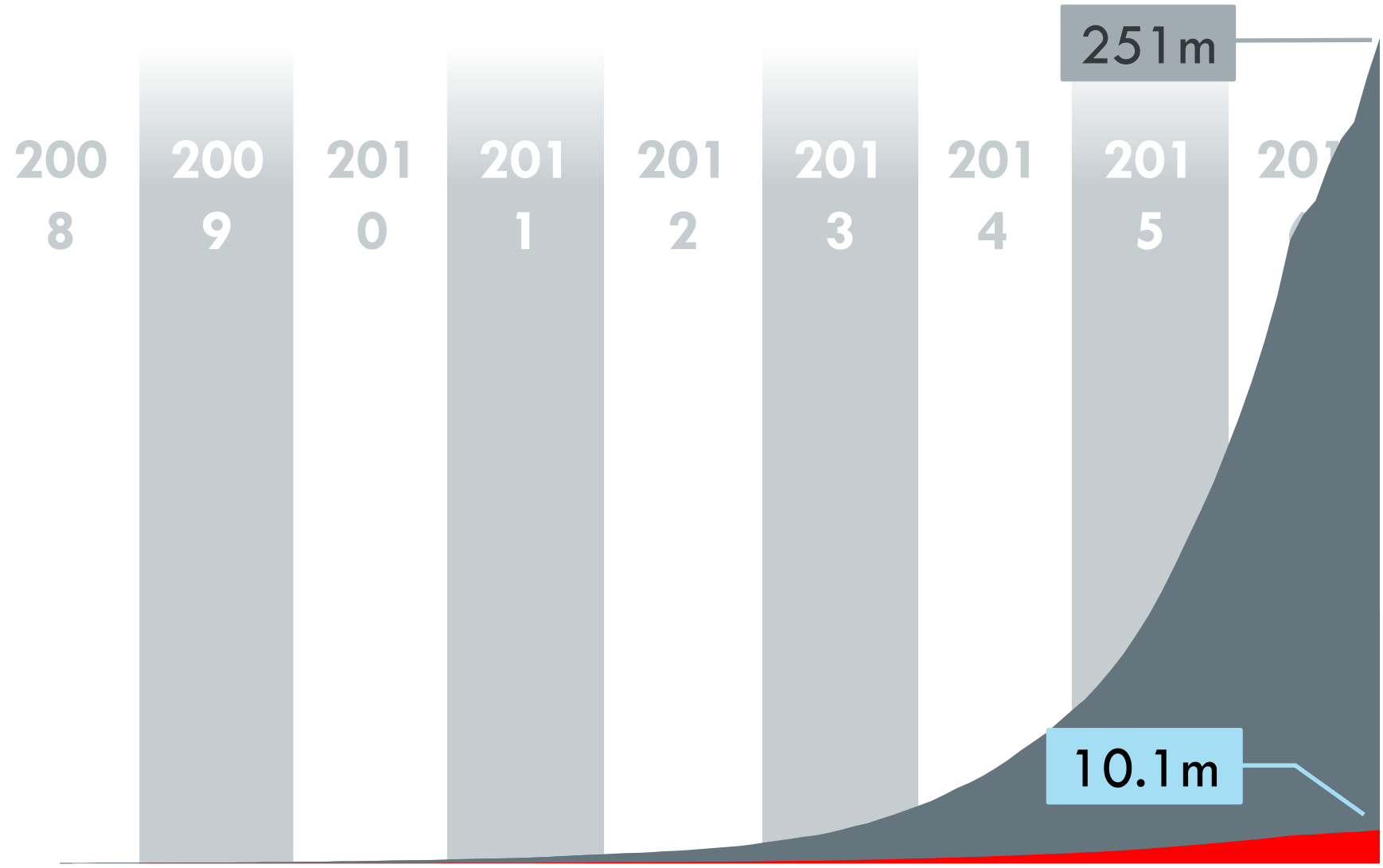
React

```
373860,30721,"build/gallery-debounce/gallery-debounce.js",12370
264560,28784,"node_modules/mocha/lib/browser/fs.js",875
373880,30721,"src/gallery-debounce/js/debounce.js",12372
264580,28784,"node_modules/mocha/browser/path.js",875
264600,28784,"node_modules/mocha/browser/progress.js",8298
373900,30721,"src/gallery-debounce/tests/unit/js/tests.js",12373
264680,28784,"node_modules/mocha/build/interfaces/bdd.js",8302
264700,28784,"node_modules/mocha/lib/interfaces/node_modules/exports.js",8303
373960,30721,"build/gallery-affix/gallery-affix-min.js",12376
264640,28784,"node_modules/mocha/lib/context.js",8300
264660,28784,"node_modules/mocha/lib/node_modules/foo/node_modules/hook.js",8301
373940,30721,"build/gallery-affix/gallery-affix-debug.js",12375
264720,28784,"node_modules/mocha/tests/interfaces/node_modules/index.js",8304
264520,28784,"node_modules/mocha/tests/browser/debug.js",8296
264460,28784,"lib/uri.js",8293
264540,28784,"node_modules/mocha/build/browser/events.js",8297
264620,28784,"node_modules/mocha/lib/browser/node_modules/tty.js",8299
264480,28784,"sanitizer.js",8294
264500,28784,"node_modules/mocha/index.js",8295
373920,30721,"build/gallery-affix/gallery-affix-coverage.js",12374
264740,28784,"tests/qunit.js",8305
264440,28784,"lib/html4.js",8292
264760,28784,"node_modules/mocha/lib/interfaces/tdd.js",8306
264420,26906,"js/script.js",8291
```

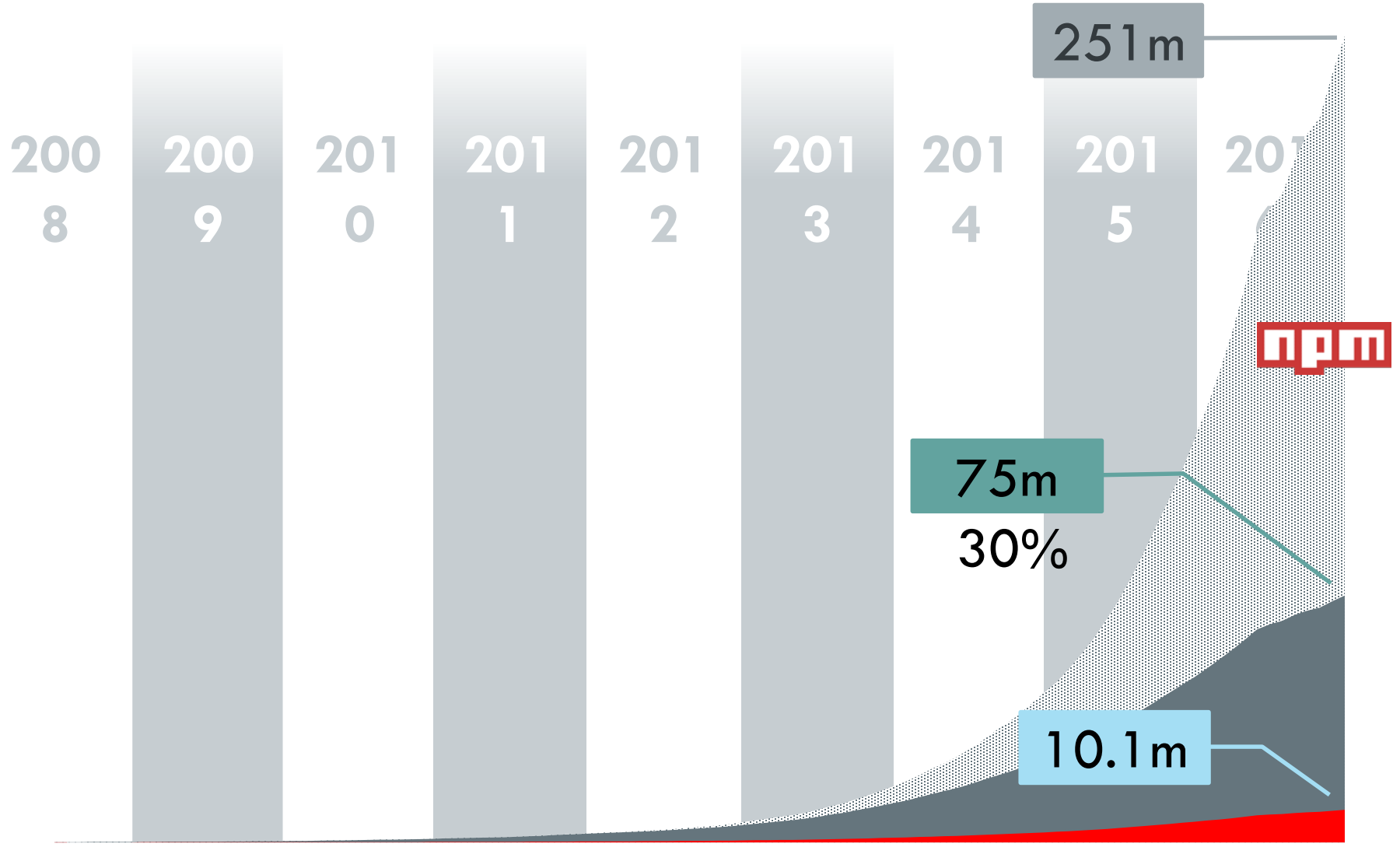
JAVASCRIPT FILES OVER TIME

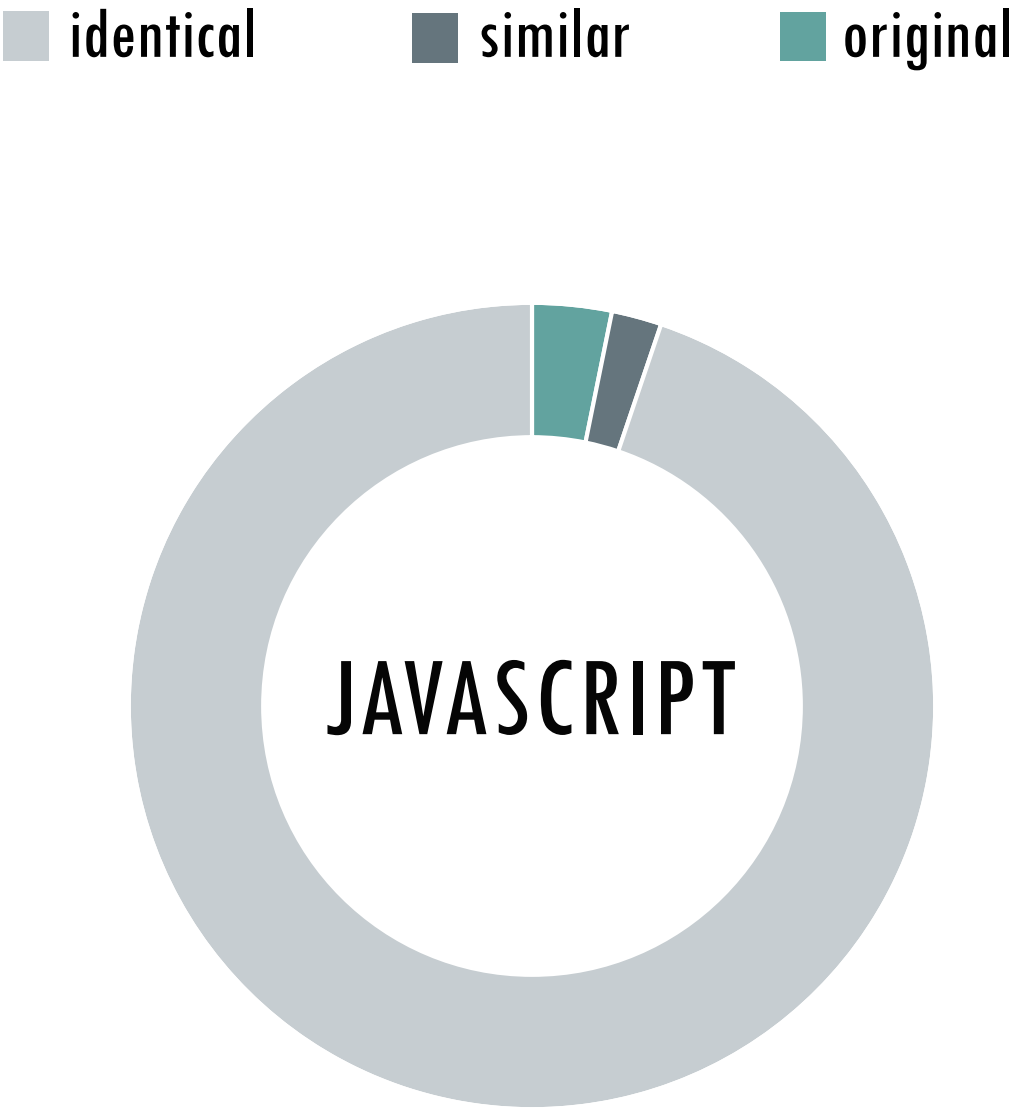


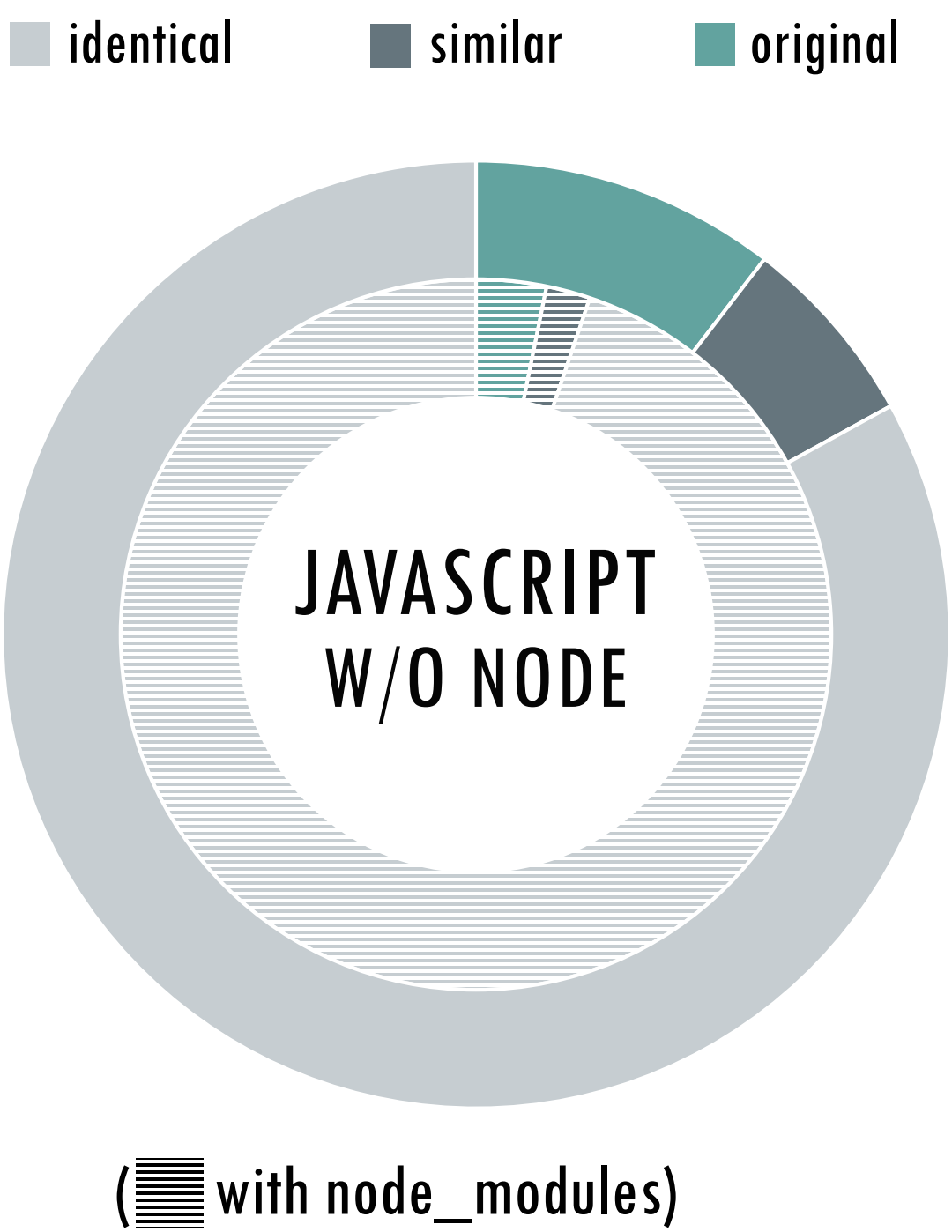
JAVASCRIPT FILES OVER TIME

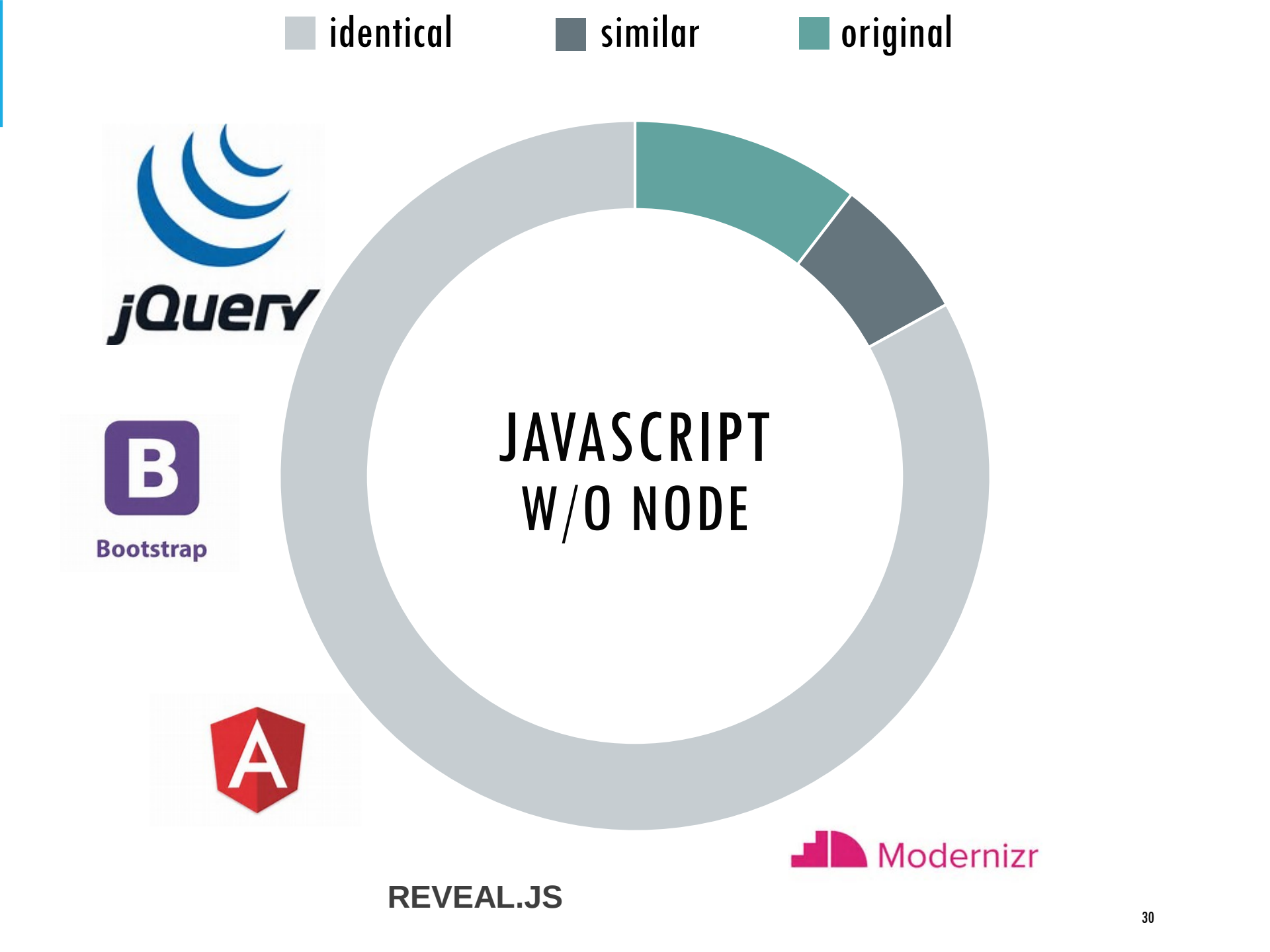


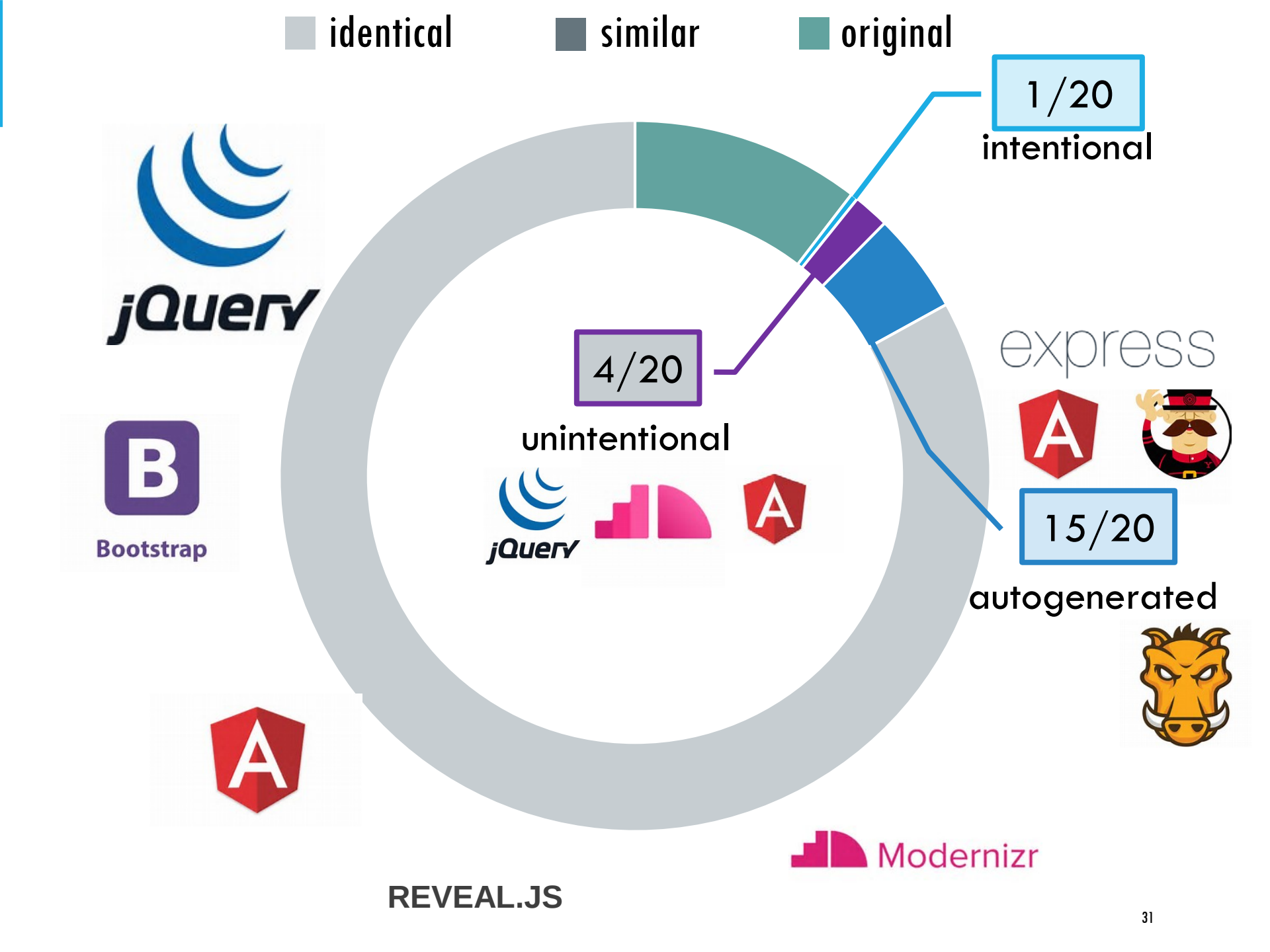
JAVASCRIPT FILES OVER TIME

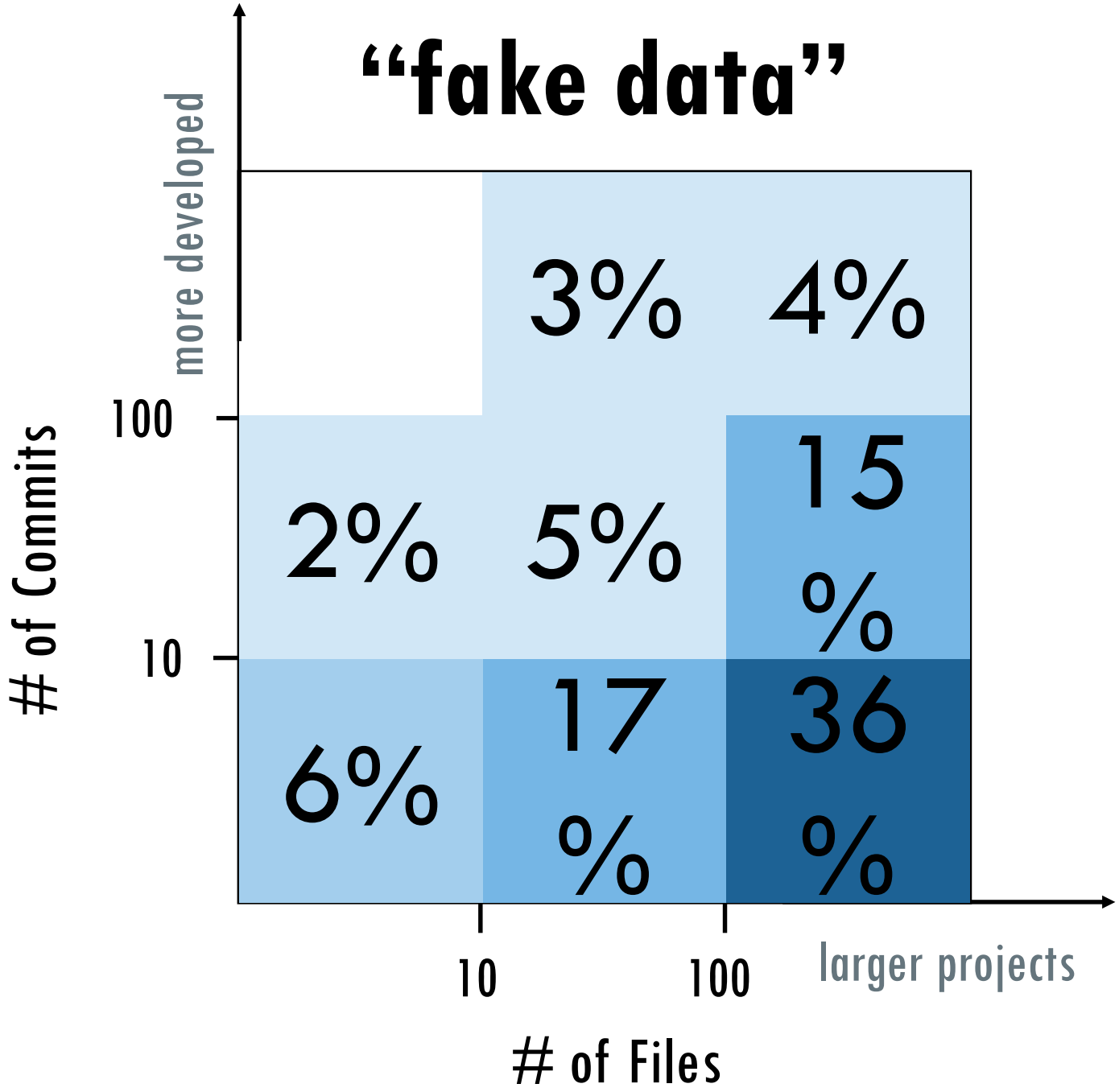



identical
similar
original
JAVASCRIPT

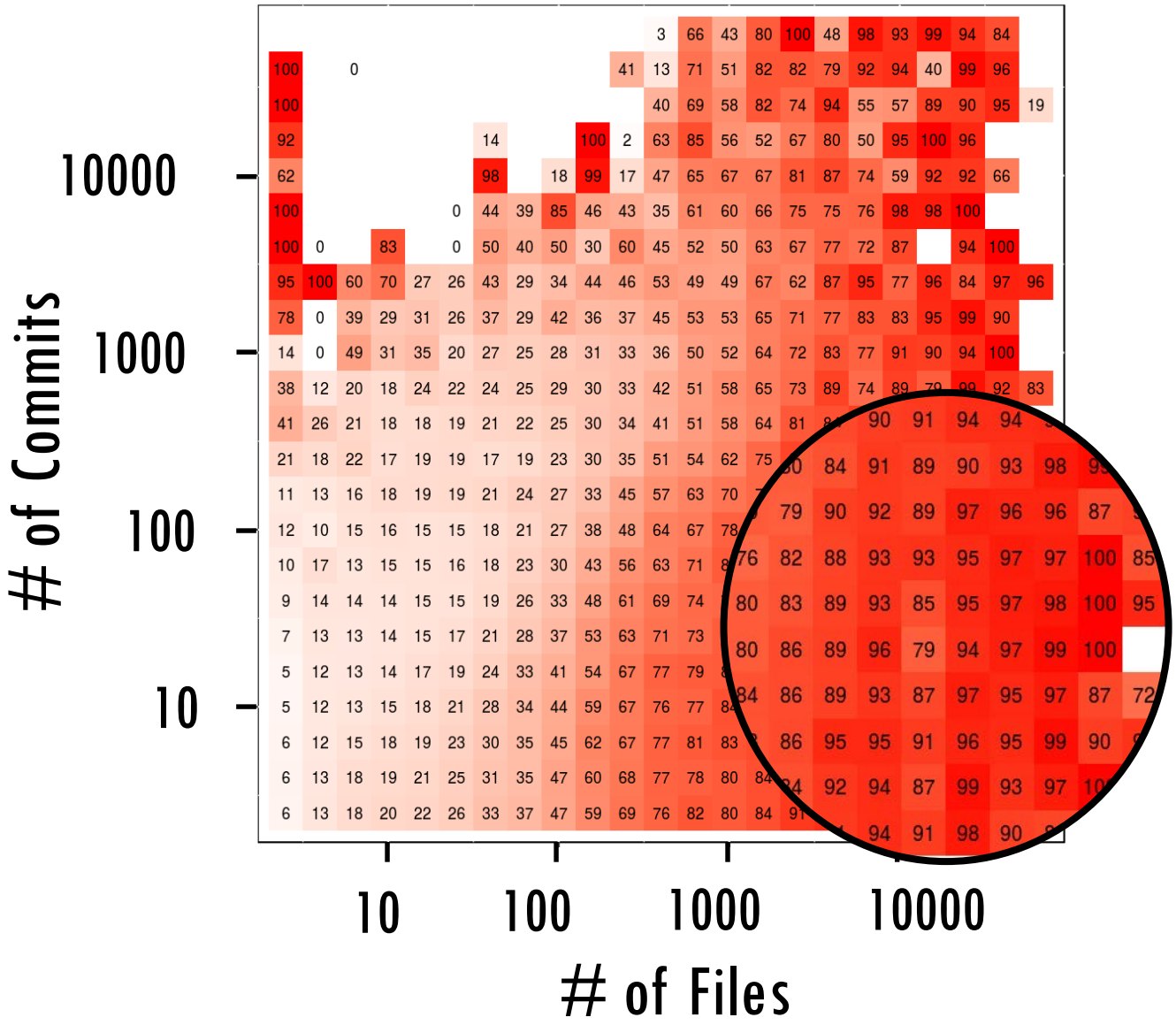

identical
similar
original
JAVASCRIPT
W/O NODE
(with node_modules)

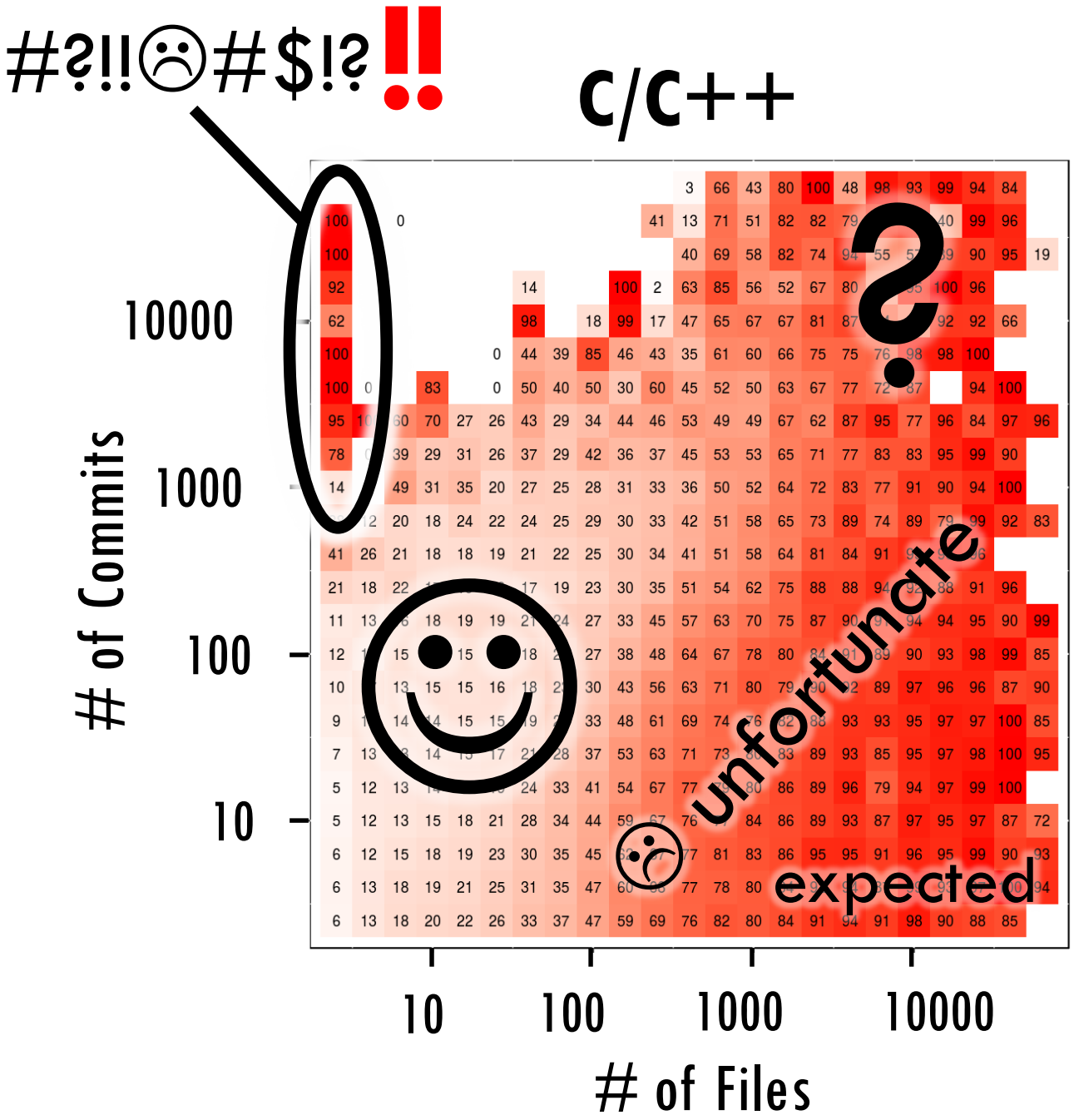

identical
similar
original
jQuery
Bootstrap
JAVASCRIPT
W/O NODE
Modernizr
REVEAL.JS


identical
similar
original
1/20
intentional
4/20
unintentional
15/20
autogenerated
jQuery
Bootstrap
express
Modernizr
REVEAL.JS

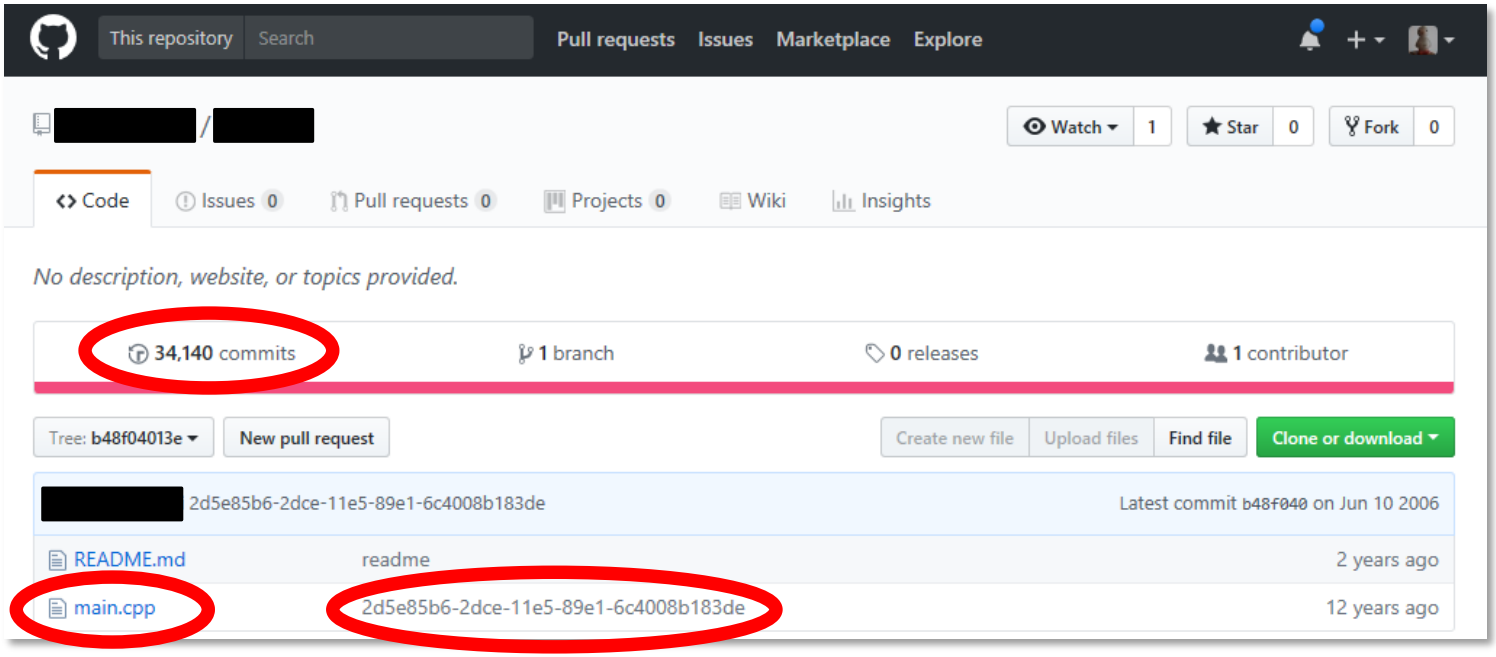

"fake data"
more developed
# of Commits
100
10
3%
4%
2%
5%
15
%
6%
17
%
36
%
10
100
larger projects
# of Files

C/C++




#?!!☹#$!?‼
C/C++
10000
1000
100
10
# of Commits
10
100
1000
10000
# of Files
unfortunate
expected

#?!!☹#$!?


This repository
Search
Pull requests
Issues
Marketplace
Explore
[redacted] / [redacted]
Watch 1
Star 0
Fork 0
Code
Issues 0
Pull requests 0
Projects 0
Wiki
Insights
No description, website, or topics provided.
34,140 commits
1 branch
0 releases
1 contributor
Tree: b48f04013e
New pull request
Create new file
Upload files
Find file
Clone or download
[redacted] 2d5e85b6-2dce-11e5-89e1-6c4008b183de
Latest commit b48f040 on Jun 10 2006
README.md
readme
2 years ago
main.cpp
2d5e85b6-2dce-11e5-89e1-6c4008b183de
12 years ago


This repository Search Pull requests Issues Marketplace Explore

[redacted] / [redacted]

Watch 100 Star 3,097 Fork 212

Code Issues 6 Pull requests 0 Projects 0 Wiki Insights

Makes you a Rockstar C++ Programmer in 2 minutes

143 commits 1 branch 0 releases 36 contributors MIT

Branch: master New pull request Create new file Upload files Find file Clone or download

[redacted] Merge pull request #86 from bryant1410/master ... Latest commit 87fe23a on Apr 17

examples	Added Fish example.	a year ago
images	added one more example	2 years ago
rockstar	Make the HELLO_WORLD variable name correct	a year ago
.gitignore	added README and images	2 years ago
LICENSE	initial import	2 years ago
MANIFEST.in	Add support for random commit messages	2 years ago
README.md	Fix broken Markdown headings	6 months ago
requirements.txt	Fixes #4	2 years ago
setup.py	Add support for random commit messages	2 years ago

README.md

Life at Google
@lifeatgoogle

Following

@iavins That's some super C++ shit on Github. Would you like to work for us?

Chris Lattner
@clattner_llvm

Following

@iavins I approve this

RETWEETS	FAVORITES
769	342

3:01 AM - 18 Jul 2015

learned

RETWEE
8856

11:37

Usag

Time is ve
couple of

Summary of pull requests, issues opened, and commits. Learn how we count contributions.

Less More

Contributions in the last year	Longest streak	Current streak
2,572 total	400 days	400 days
Jul 16, 2014 – Jul 16, 2015	June 12 – July 16	June 12 – July 16

```
from rockstar import RockStar
```

This repository Search Pull requests Issues Marketplace Explore

[redacted] / [redacted]

Watch 1 0 Star 3,097 Fork 212

Code Issues 6 Pull requests 0 Projects 0 Wiki Insights

Makes you a Rockstar C++ Programmer in 2 minutes

143 commits 1 branch 0 releases 36 contributors MIT

Branch: master New pull request Create new file Upload files Find file Clone or download

[redacted] Merge pull request #86 from bryant1410/master ... Latest commit 87fe23a on Apr 17

examples	Added Fish example.	a year ago
images	added one more example	2 years ago
rockstar	Make the HELLO_WORLD variable name correct	a year ago
.gitignore	added README and images	2 years ago
LICENSE	initial import	2 years ago
MANIFEST.in	Add support for random commit messages	2 years ago
README.md	Fix broken Markdown headings	6 months ago
requirements.txt	Fixes #4	2 years ago
setup.py	Add support for random commit messages	2 years ago

This repository | Search | Pull requests | Issues | Marketplace | Explore

[redacted] / [redacted]

Watch 1 | Star 0 | Fork 0

Code | Issues 0 | Pull requests 0 | Projects 0 | Wiki | Insights

No description, website, or topics provided.

67,889 commits | 1 branch | 0 releases | 1 contributor

Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

[redacted] Final commit

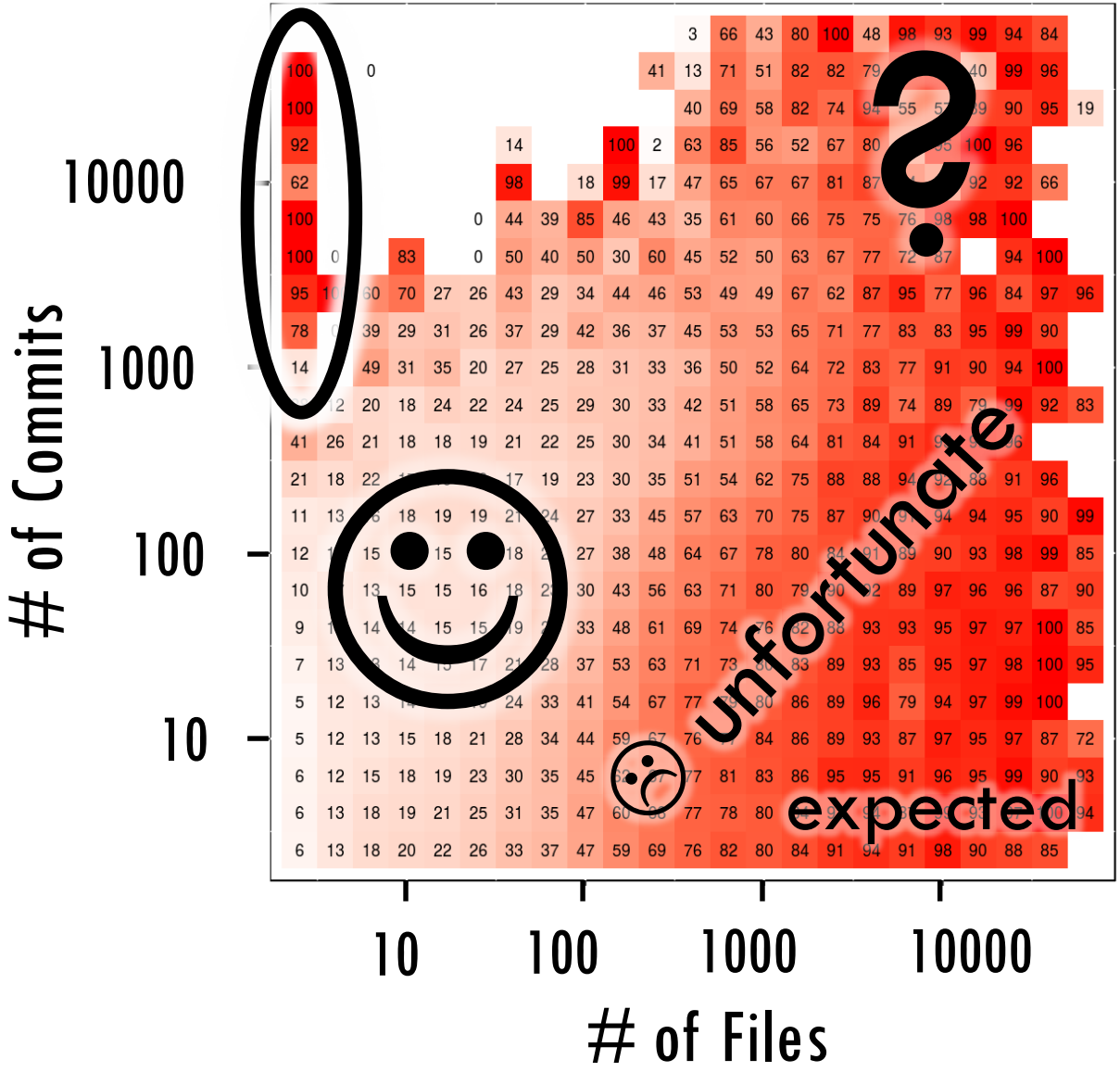
Latest commit ea2fa16 on Jul 20 2015

hello.java	Final commit	2 years ago
hello.swift	Final commit	2 years ago
helloWorld.c		
helloWorld.cpp		
helloWorld.html		
helloWorld.js	Final commit	2 years ago
hello_world.sql	Final commit	2 years ago
helloworld.php	Final commit	2 years ago
heloWorld.py	Final commit	2 years ago

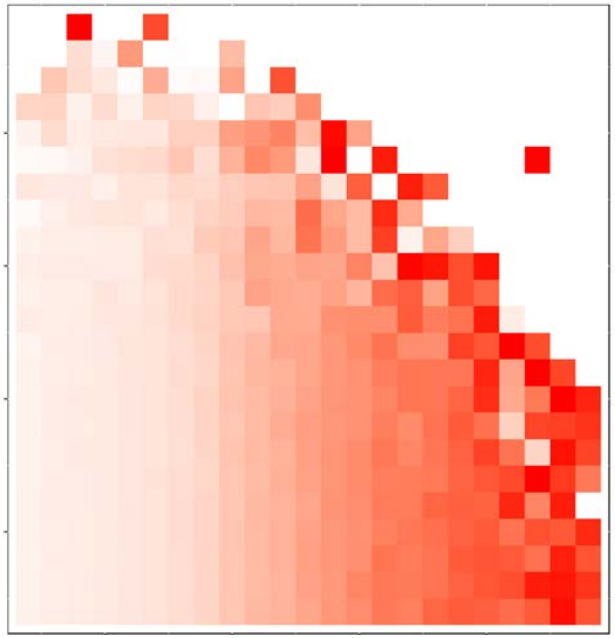
1 lines (1 sloc) | 39 Bytes

```
1  std::cout << 'Hello world' << std::endl
```

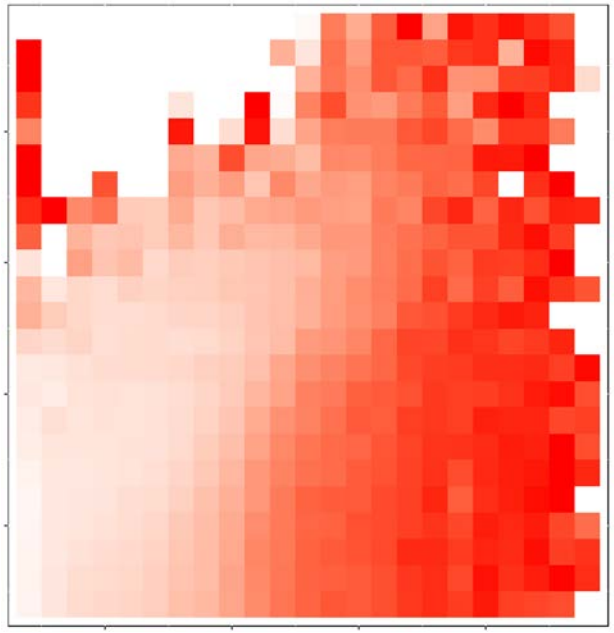
C/C++


10000
1000
100
10
# of Commits
10
100
1000
10000
# of Files
unfortunate
expected

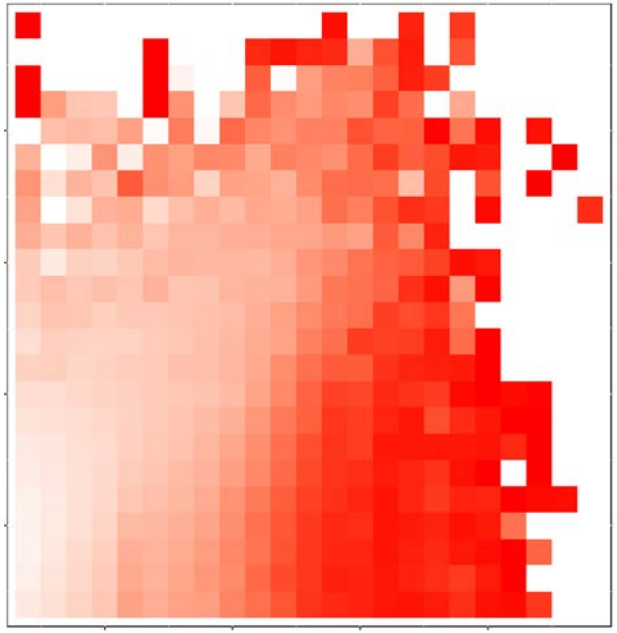
Java



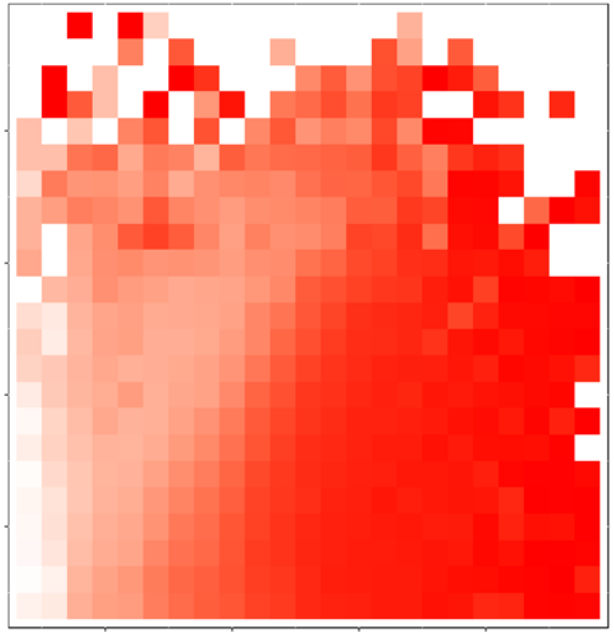
C/C++



Python



JavaScript

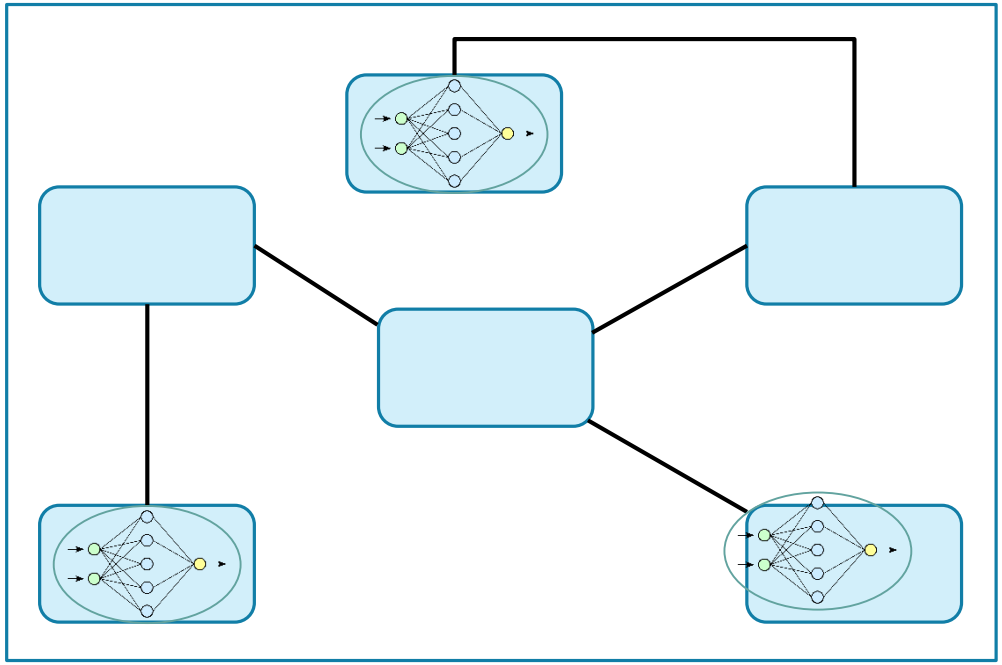


Farima **Farmahinifarahani** Crista **Lopes**

FINDING FUNCTIONAL CLONES IN DNN CLASSIFIERS

BRINC

DNN MODELS IN SOFTWARE



DNN FUNCTIONAL SIMILARITY: TRAINING SCRIPTS

Even if training scripts are available:

```
def train_my_model1():#train on mnist
    import keras
    (X_train,Y_train),(X_test,Y_test)=mnist.load_data()
    X_train=(X_train.reshape((60000,28*28))).astype('float32')/255
    Y_train=keras.utils.to_categorical(Y_train)
    dnn=keras.models.Sequential()
    dnn.add(keras.layers.Dense(512,activation='relu',input_shape=(28*28,)))
    dnn.add(keras.layers.Dense(10,activation='softmax'))
    dnn.compile(optimizer='rmsprop',loss='categorical_crossentropy',metrics=['
          accuracy'])
    dnn.fit(X_train,Y_train,epochs=10,batch_size=128)
```

```
def train_my_model2():#train on fashion_mnist
    import keras
    (X_train,Y_train),(X_test,Y_test)=fashion_mnist.load_data()
    X_train=(X_train.reshape((60000,28*28))).astype('float32')/255
    Y_train=keras.utils.to_categorical(Y_train)
    dnn=keras.models.Sequential()
    dnn.add(keras.layers.Dense(512,activation='relu',input_shape=(28*28,)))
    dnn.add(keras.layers.Dense(10,activation='softmax'))
    dnn.compile(optimizer='rmsprop',loss='categorical_crossentropy',metrics=['
          accuracy'])
    dnn.fit(X_train,Y_train,epochs=10,batch_size=128)
```

DNN MODELS: BLACK BOXES

Image from: https://expoundai.wordpress.com/2019/04/17/deep-learning-our-journey-begins/

DNN MODELS

Matrices of numbers: weights and biases

Learned through training

Their structure does not disclose any insights on the functions being implemented

Weight matrices can become very large!

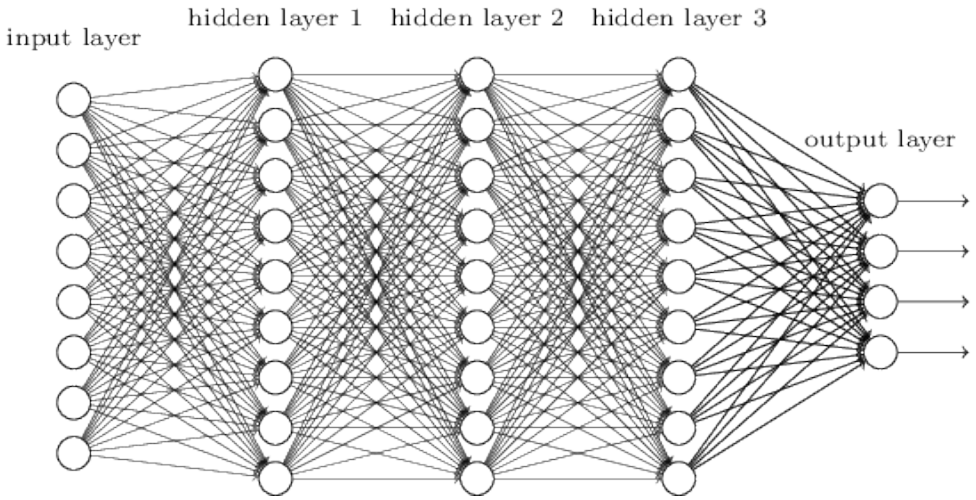


Image from: http://neuralnetworksanddeeplearning.com/chap5.html

DNN MODELS FUNCTIONAL SIMILARITY

Another solution:

- Input/output analysis

Given a canonical set of inputs, when the outputs of two models are largely the same, then the models are functionally similar

Requirement: possessing a set of canonical inputs

BRINC: FUNCTIONAL SIMILARITY

The main measure:

- Given two classifiers, over the same input data, how many times do they agree on their classifications?

The absence of canonical inputs:

- Generate random inputs
- But can't be unconstrained… (another talk)

BALANCED RANDOM INPUTS: SIMILARITY THRESHOLDS

A query model M_q with N output labels;

A dataset of random inputs d where M_q has balanced distribution of labels;

Any arbitrary comparable model can randomly agree with M_q on d for $^1/_n$ times

- This notion can help in defining the thresholds
- If two models agree for $^1/_n$ times, or less → they are not similar
- If the level of agreement is much higher (e.g., twice the level of chance) → something more than chance at play, indicator of models' similarity

VALIDATION: DATA COLLECTION

Searched for DNN classifiers on GitHub

Using GitHub code search API[1]

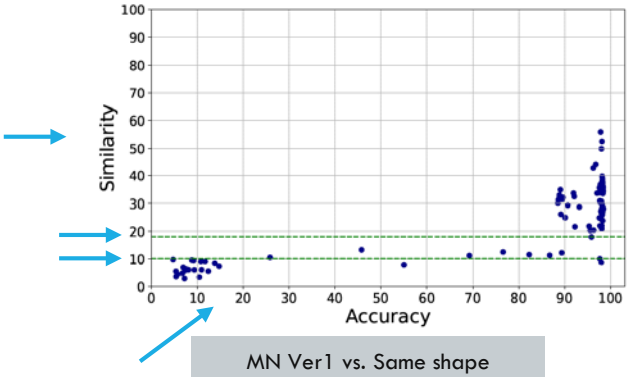
Search query: files with *.h5* extension

Obtained a list of 340,933 *.h5* files

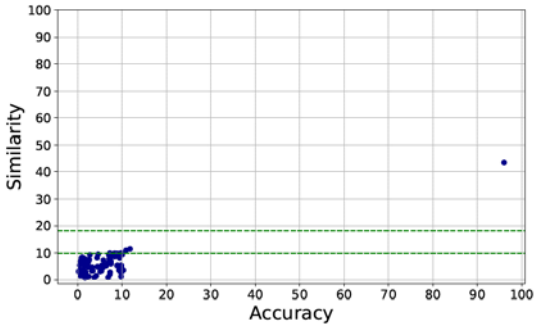
After download and filtering:

- 56,355 models clustered into 6,280 groups based on input and output shapes

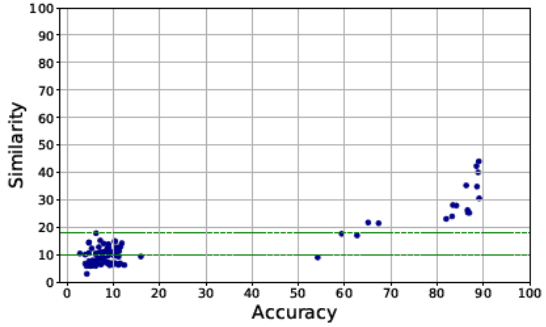
VALIDATION ON KNOWN QUERY MODELS: RESULTS OVERVIEW



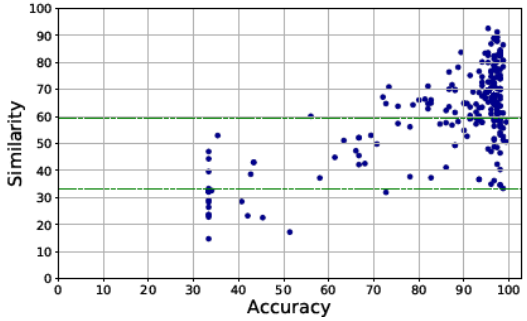
MN Ver1 vs. Same shape models



MN Rev Color vs. Same shape models

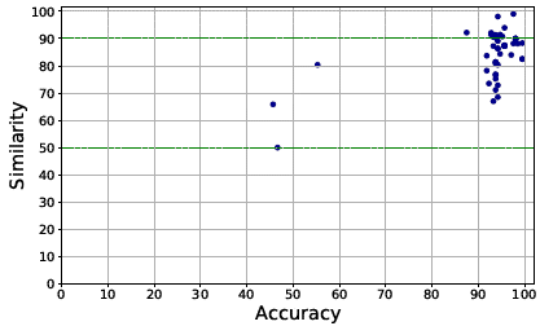


FMN vs. Same shape models

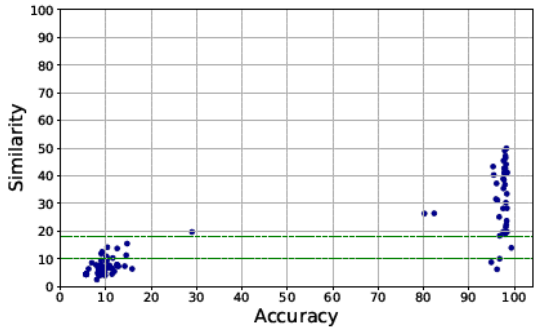


Iris vs. Same shape models

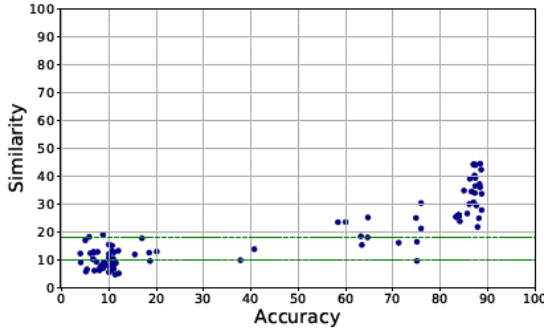
VALIDATION ON KNOWN QUERY MODELS: RESULTS OVERVIEW



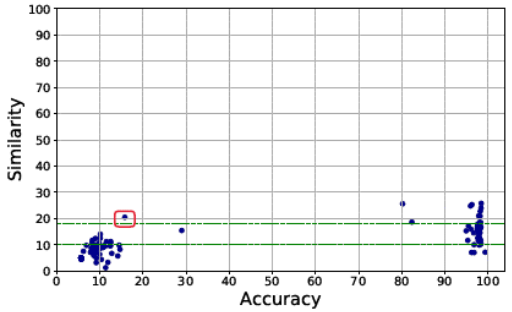
Sonar vs. Same shape models



MN Ver1 vs. Compatible shape models

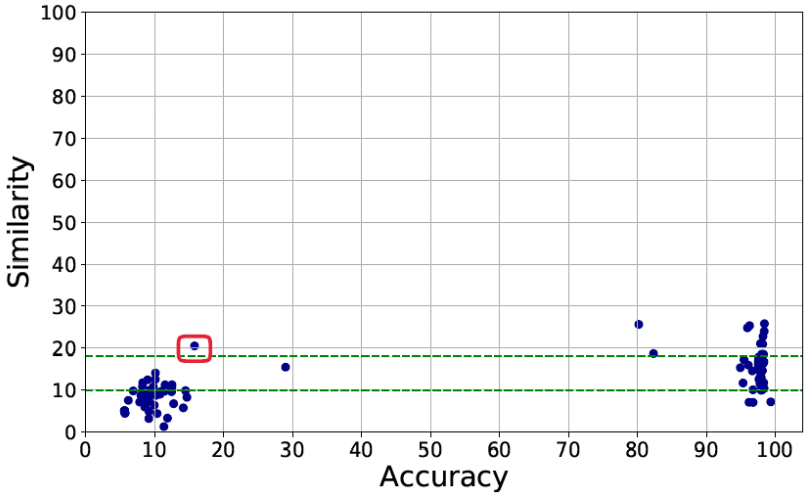


FMN vs. Compatible shape models



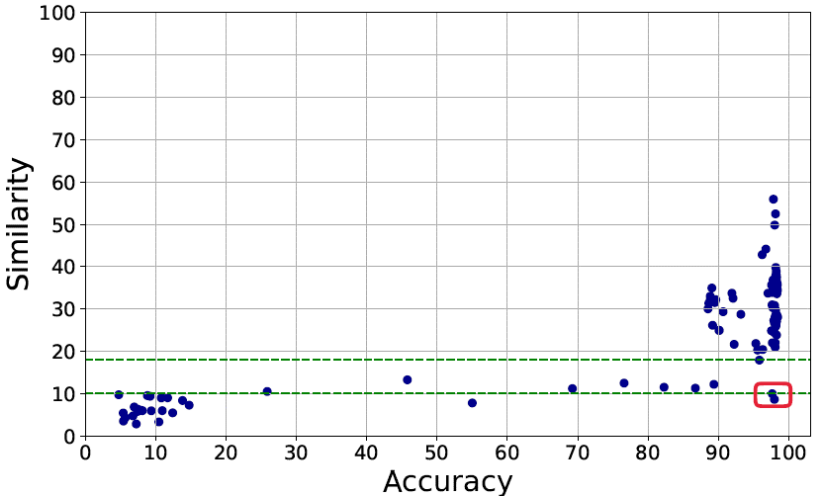
CNN vs. Compatible shape models

VALIDATION ON KNOWN QUERY MODELS: INTERESTING CASES



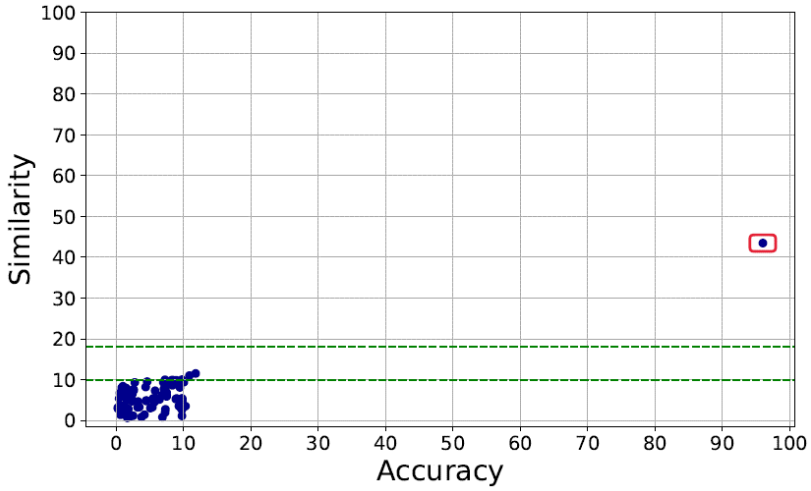
CNN vs. Compatible shape models

VALIDATION ON KNOWN QUERY MODELS: INTERESTING CASES



MN Ver1 vs. Same shape models

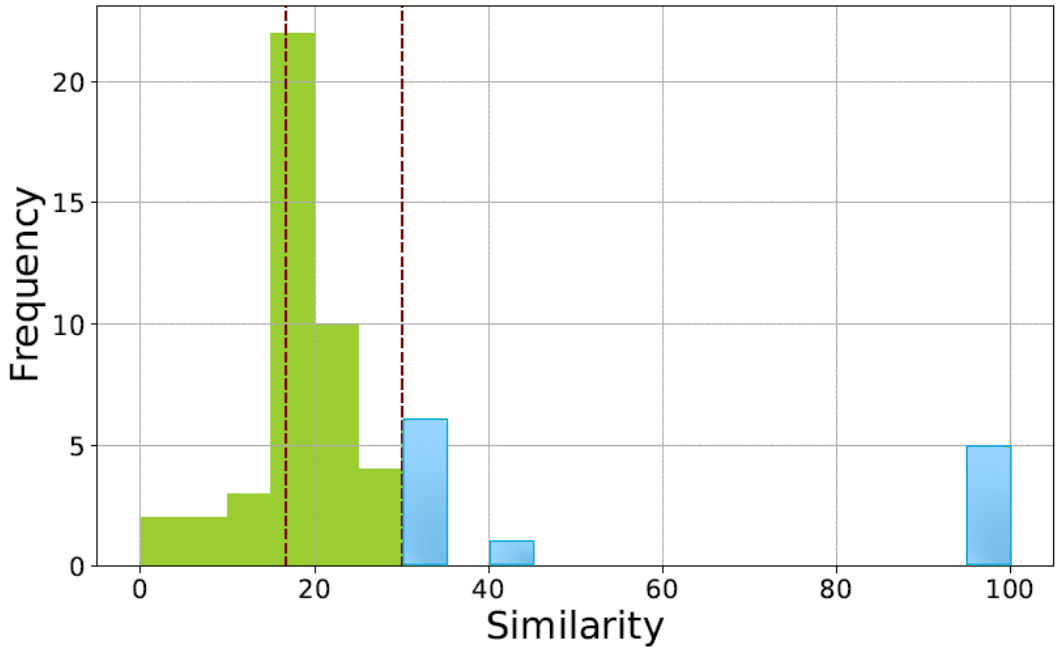
VALIDATION ON KNOWN QUERY MODELS: INTERESTING CASES



MN Rev Color vs. Same shape models

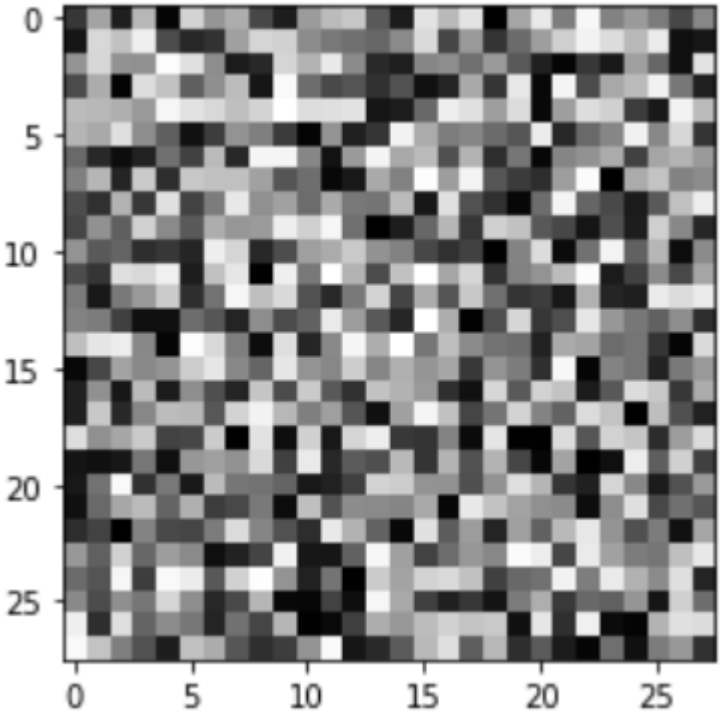
VALIDATION ON ARBITRARY QUERY MODELS

No data model (6 output classes)



SUMMARY

It is entirely possible to find clones of DNN classifiers without knowing anything about what they do or how they were trained! – just feed carefully crafted noise as input



THANK YOU!

clone
clone
clone
clone
clone
clone
clone
clone
clone
clone
clone
clone
clone
clone
clone
clone
clone
clone
clone
clone
clone

crista@tagide.com
@cristalopes