Summary of
“A COMPARISON STUDY OF OPENSOURCE LICENSE CRAWLER”
by THOMAS WOLTER

Japan workgroup
Hiroyuki FUKUCHI

Japan workgroup

Paper

- Title: “A COMPARISON STUDY OF OPENSOURCE LICENSE CRAWLER”
- Author: THOMAS WOLTER
- Organization: Friedrich-Alexander-Universitat Erlangen-Nurnberg, Technische Fakultat, Department Informatik

- URL: https://osr.cs.fau.de/2019/08/07/final-thesis-a-comparison-study-of-open-source-license-crawler/

Agenda

- Background
- Selecting scanning tools
- Selecting project to be tested
- Evaluation criteria
- Result

Background

- Finding the appropriate licensing information often causes problem.
- There are no clear guidelines on where exactly the license text should be placed.
  - Root directory
  - COPYING, LICENSE file
  - README file
- Contain several licenses in different locations of the directory tree
  - Results in a conflict situation

Selecting scanning tools

- Collecting scanning tools:
  - Search by “license crawler”, “license identifier”, “license detector” via Google, GitHub
- Filter Criteria:
  - 1. A function to scan a given project for licensing information. This can be limited to the root directory or extend to the entire directory tree. Only looking at a single file however was considered to be insufficient.
  - 2. The scanning process is mostly automated and does not require much input beyond an initial directory name or input le.
  - 3. The output presents the results in a comprehensive manner. In order to make more in-depth comparisons in phase 2 of our benchmark we needed the crawlers to give details about their nds.
  - 4. The project is open sourced.

- Selected tools:
  - askalono
  - FOSSology
  - go-license-detector
  - Licensechecker
  - Licensee
  - scancode,

Evaluation Criteria

- There is no clear guideline for tool comparison
- Proposed method:
  - Phase 1: Number of found licenses
  - Phase 2: Difference between top two tools and human detection

Selecting projects for scanning

- Collecting projects:
  - the 1000 most starred projects on GitHub.
- Filter Criteria:
  - The project is actively working on developing code.
  - The project is providing more than just links to other resources.
  - It must be feasible that the project is implemented in other projects.

Used projects

Used projects:

AFNetworking-master	incubator-echarts-master	react-native-master
angular-master	jekyll-master	react-router-master
async-master	jQuery-File-Upload-master	redis-5.0
atom-master	julia-master	redux-master
axios-master	keras-master	requests-master
babel-master	kotlin-master	Rocket.Chat-master
bitcoin-master	laravel-master	rust-master
bootstrap-master	lodash-master	RxJava-2.x
brackets-master	lottie-android-master	scikit-learn-master
caddy-master	mermaid-master	SDWebImage-master
cpython-master	meteor-master	select2-master
d3-master	moment-master	serverless-master
discourse-master	node-master	shadowsocks-windows-master
express-master	normalize.css-master	slate-master
flask-master	nvm-master	socket.io-master
fullPage.js-master	nylas-mail-master	spring-boot-master
Ghost-master	oh-my-zsh-master	swift-master
gogs-master	parcel-master	tensorflow-master
grafana-master	pdf.js-master	three.js-master
gulp-master	pixi.js-master	vscode-master
hexo-master	preact-master	vue-master
html5-boilerplate-master	prettier-master	x64dbg-development
httpie-master	prometheus-master	yarn-master
hugo-master	quill-develop	you-get-develop
immutable-js-master	rails-master	zxing-master

Result

Phase 1: Total license found

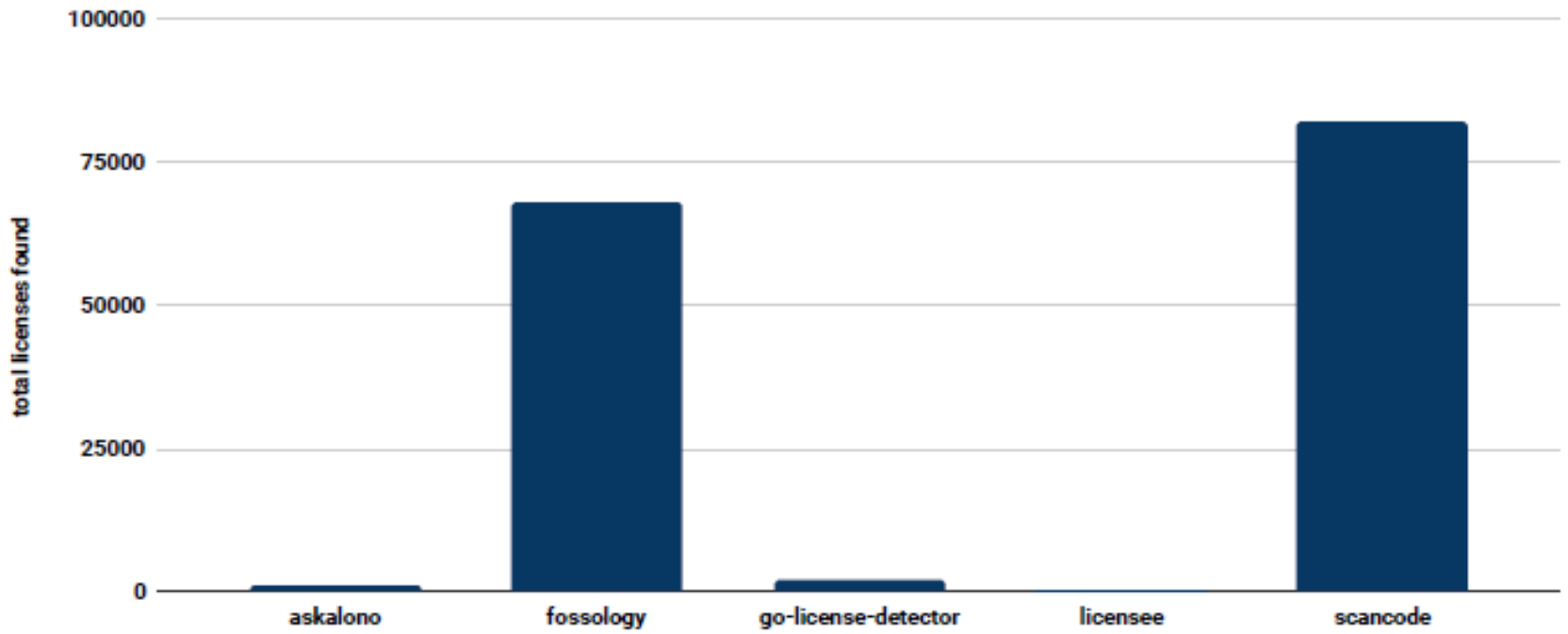

Figure 2.6: Total licenses found

Unique licenses found

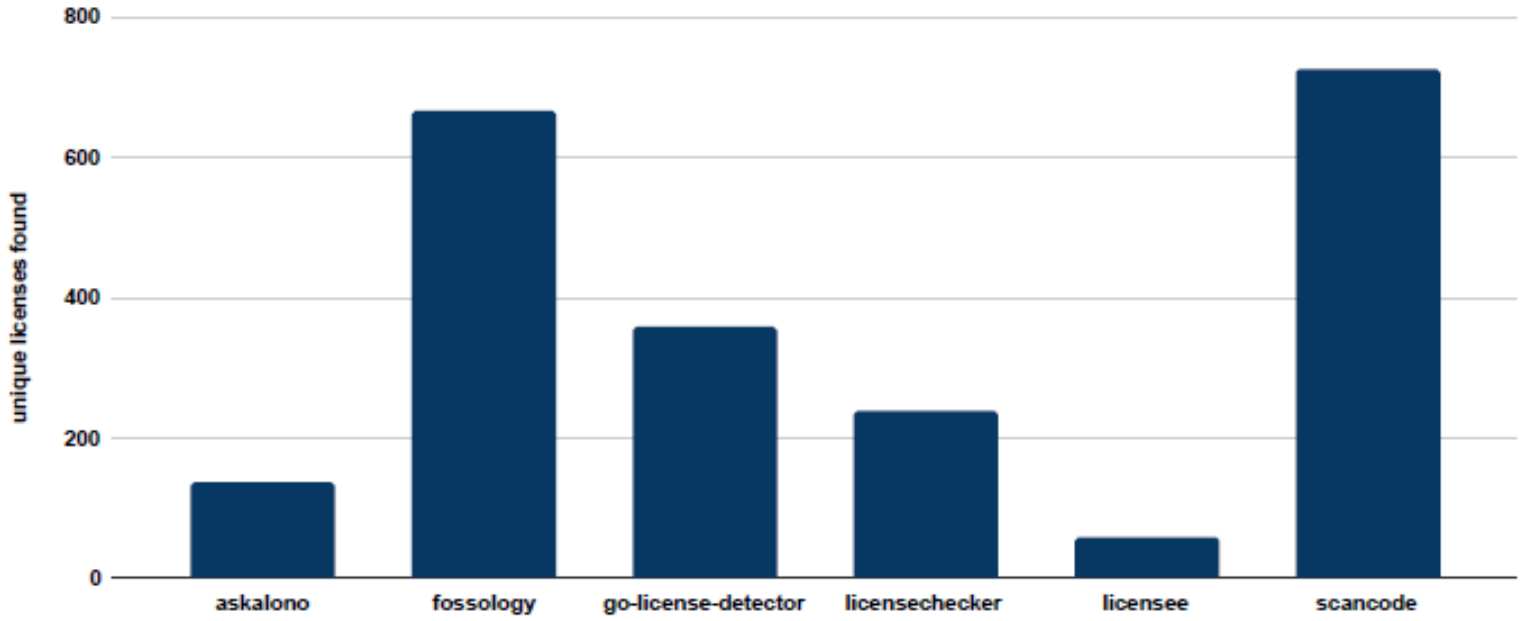

Figure 2.7: Unique licenses found

Time of scanning

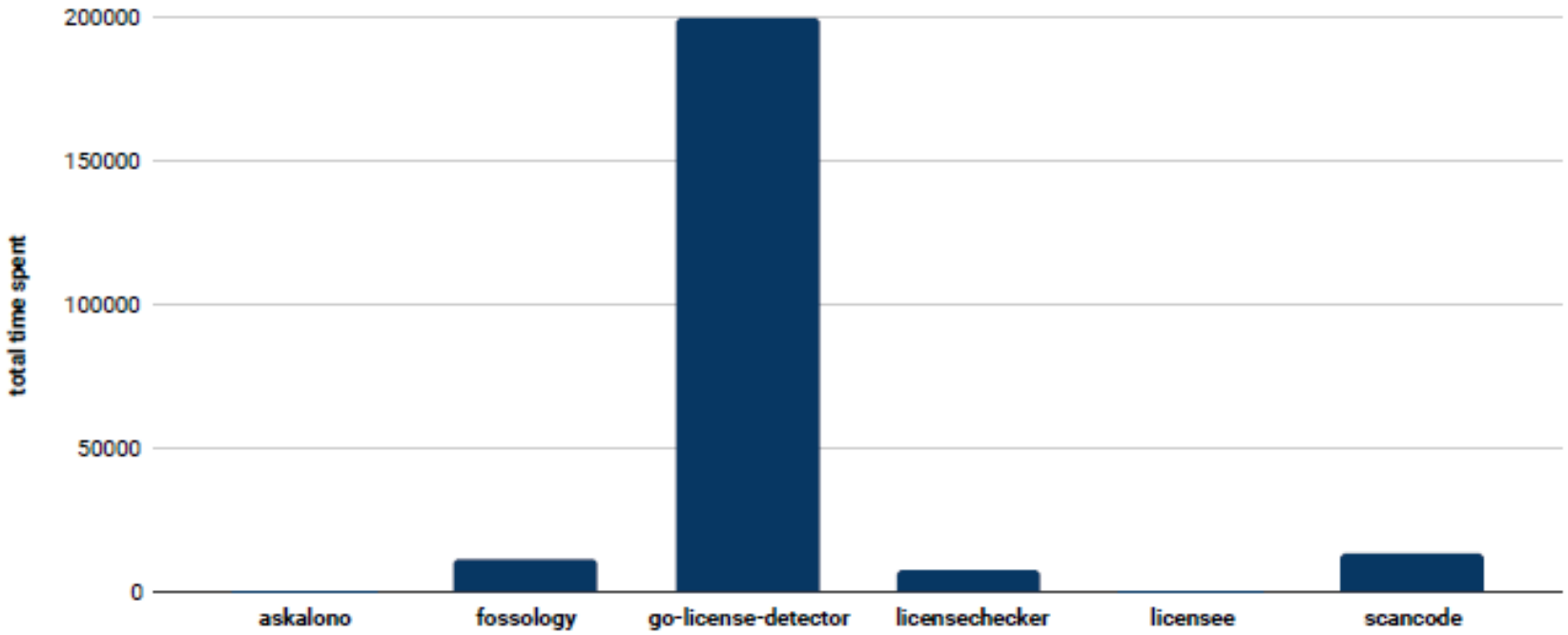

Figure 2.8: Total time spent scanning

Phase 1

- FOSSology and Scancode check every single file available and ultimately provide a better result.
- The other crawlers do not look at all files. They make preselection of files they deem likely to have relevant data.

Phase 2

- FOSSology and scancode
- 149,884 total license hits
  - Agreed: 124,756 (83.24%)
  - Conflict: 12,564
  - (Other?)

- Analysis of the difference between FOOSology and scancode and human detection
- 25 cases

25 conflict situations

- angular-master/packages/animations/browser/src/render/css keyframes/direct style player.ts
- angular-master/modules/benchmarks/e2e test/tree spec.ts
- bitcoin-master/src/qt/transactionlterproxy.h
- bitcoin-master/src/test/cuckoocache tests.cpp
- brackets-master/src/extensions/default/JavaScriptQuickEdit/unittest-les/jqueryui/ui/jquery.ui.tooltip.js
- brackets-master/src/nls/id/strings.js
- cpython-master/Lib/platform.py
- cpython-master/Lib/unittest/ init .py
- kotlin-master/core/script.runtime/src/kotlin/script/templates/annotations deprecated.kt
- kotlin-master/js/js.ast/src/org/jetbrains/kotlin/js/backend/ast/JsBreak.java
- node-master/deps/v8/test/message/fail/rest-param-object-setter-sloppy.js
- node-master/deps/v8/test/mjsunit/harmony/regexp-property-lu-ui3.js
- node-master/deps/v8/test/cctest/gay-precision.cc
- node-master/deps/v8/tools/unittests/testdata/testroot2/test/sweet/testcfg.py
- node-master/deps/v8/src/string-hasher.h
- node-master/deps/v8/src/compiler/type-narrowing-reducer.cc
- node-master/deps/icu-small/source/i18n/simpletz.cpp
- node-master/deps/icu-small/source/common/ubidiln.cpp
- node-master/deps/zlib/FAQ
- pdf.js-master/test/resources/reftest-analyzer.js
- prometheus-master/vendor/google.golang.org/api/CONTRIBUTORS
- Rocket.Chat-master/packages/rocketchat-ui/client/lib/Modernizr.js
- scikit-learn-master/sklearn/utils/multiclass.py
- swift-master/stdlib/public/core/SipHash.swift
- tensorow-master/tensorow/compiler/xrt/BUILD

	license present	license not present
license identified	12	20
license missed	6	1

Table 2.4: Confusion matrix: Scancode

	license present	license not present
license identified	14	7
license missed	4	0

Table 2.5: Confussion matrix: FOSSology

4 error categories

- License references
  - Cannot find LICENSE file, etc.
- Context
  - Misreading FAQ, etc.
- Incorrect version
  - GPL v2, v3 etc.
- False evaluation
  - Author cannot understand the reason to fail

References

- German, D. M., Manabe, Y. & Inoue, K. (2010).
- A Sentence-matching Method for Automatic License Identification of Source Code Files. In Proceedings of the IEEE/ACM International Conference on Automated Software Engineering(S. 437{446). ASE '10. Antwerp, Belgium: ACM.

- Lerner, J. & Tirole, J. (2005).
- The Scope of Open Source Licensing. Journal of Law, Economics, & Organization, 21 (1), 20{56.

- Rosen, L. (2005).
- Open Source Licensing: Software Freedom and Intellectual Property Law. Prentice Hall.

- Stewart, K. J., Ammeter, A. P. & Maruping, L. M. (2006).
- Impacts of License Choice and Organizational Sponsorship on User Interest and Development Activity in Open Source Software Projects. Information Systems Research, 17 (2), 126-144.

- Stol, K.-J. & Fitzgerald, B. (2018).
- The ABC of Software Engineering Research. ACM Trans. Softw. Eng. Methodol. 27 (3), 11:1-11:51.

- Vendome, C., Bavota, G., Penta, M. D., Linares-Vasquez, M., German, D. & Poshyvanyk, D. (2017).
- License usage and changes: a large-scale study on gitHub. Empirical Software Engineering, 22 (3), 1537-1577.