

Characterizing the Occurrence of Dockerfile Smells in Open-Source Software: An Empirical Study

Yiwen Wu, Yang Zhang, Tao Wang, Huaimin Wang

Proceedings of the 2020 Institute of Electrical and Electronics Engineers Access(IEEE Access)
vol.8, pp34127–34139

発表者：辻 裕真

背景

- Dockerは最も人気なコンテナツールであり， 頻繁に利用されている
- コンテナはDockerfileにより作成され， アプリケーションとその依存関係， 実行環境などが記述される
- Dockerfileには多くのCode smells (Dockerfile smells)が存在し， プロジェクトの品質問題となっている
- プロジェクトにおけるDockerfile smellsを調査することで今後のDockerfileのベストプラクティスを提供することを目的とする

Dockerfile smells

- DL-smells
 - ▶ Docker公式ドキュメントで公開されているベストプラクティスに違反
- SC-smells
 - ▶ 基本的なシェルスクリプトのプラクティスに違反 (RUN命令)
- Dockerfile静的解析ツール (Dockerfile Linter) を用いて Dockerfile smellsを検出する

Dockerfile smells

Base Image	←	FROM debian	DL Smell: Always tag the version of an image explicitly
Maintainer Info	←	MAINTAINER David James<david@james.com>	DL Smell: MAINTAINER is deprecated
ENV. Variable	←	ENV USE_HTTP 1	
Comment	←	# Add proxy settings	
Copy script	←	COPY ./setenv.sh /tmp/	
Execute shell	←	RUN export node_version="0.10"	
Installing dependencies	{	RUN apt-get update RUN apt-get -y install nodejs="\$node_version"	SC Smell: node_version is referenced but not assigned (did you mean 'node version?') DL Smell: Avoid additional packages by specifying '--no-install-recommends'
Copy file	←	COPY package.json usr/src/app	
Installing software	{	RUN cd /usr/src/app RUN npm install node-static	SC Smell: Use 'cd ... exit' or 'cd ... return' in case cd fails. DL Smell: Use WORKDIR to switch to a directory DL Smell: Pin versions in npm. Instead of `npm install <package>` use `npm install <package>@<version>`
Open Port	←	EXPOSE 80000	DL Smell: Valid UNIX ports range from 0 to 65535
Start Process	←	CMD ["npm", "start"]	

FIGURE 1. Example of a Dockerfile illustrating instructions and smells.

Research Questions

RQ1 : Dockerfile smellsを持つプロジェクトはどの程度か

RQ2 : DL-smellsとSC-smellsの分布に違いはあるか

RQ3 : Dockerfile smellsはプロジェクトの特性に依存するか

データセット

- Google BigQueryを利用してGitHub上のDockerfileを含む16,959のプロジェクトとそのメタデータを取得
- プロジェクトで使用されているプログラミング言語を調査し、上位10件のプログラミング言語を使用している6,334のプロジェクトで調査を行う
 - ▶ Dockerfile smellsとプロジェクトが使用している言語との関係を調査するため
- 全てのプロジェクトの最新のDockerfileにLinterを使用してDockerfile smellsを検出する

分析結果

RQ1 : Dockerfile smellsを持つプロジェクトはどの程度か

- 6,334のプロジェクトから29,843のDockerfile smellsを検出
 - DL-smells 26,545 (88.9%) SC-smells 3,298 (11.1%)
- 5,309 (83.8%)のプロジェクトに少なくとも1つのDockerfile smellsが存在した

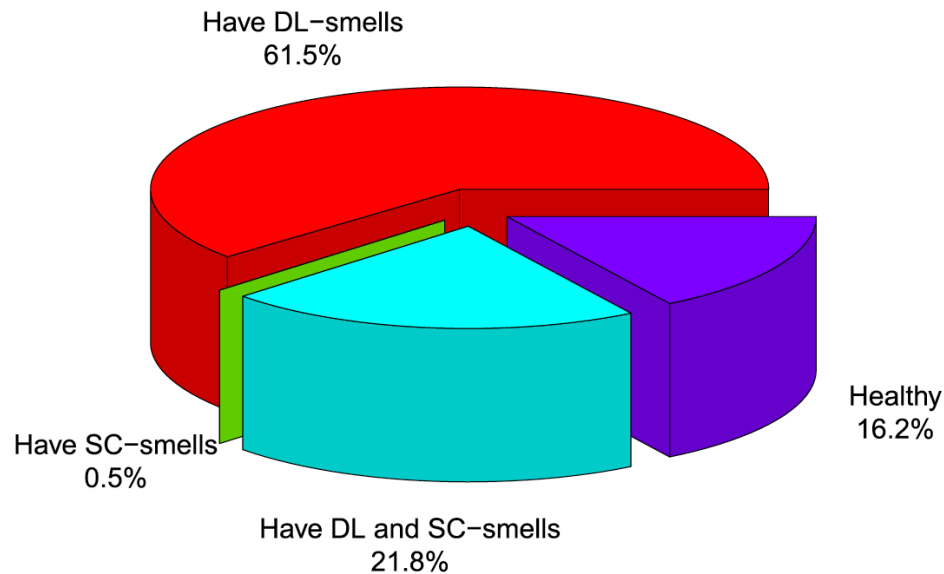


FIGURE 3. The distribution of projects w/wo Dockerfile smells.

分析結果

RQ1 : Dockerfile smellsを持つプロジェクトはどの程度か

- Dockerfile smellsを含むプロジェクトを調査
- Dockerfile内に平均:5.6（中央値:4.0）のsmellsが存在
- smellsの密度は平均:0.62（中央値:0.43）であった

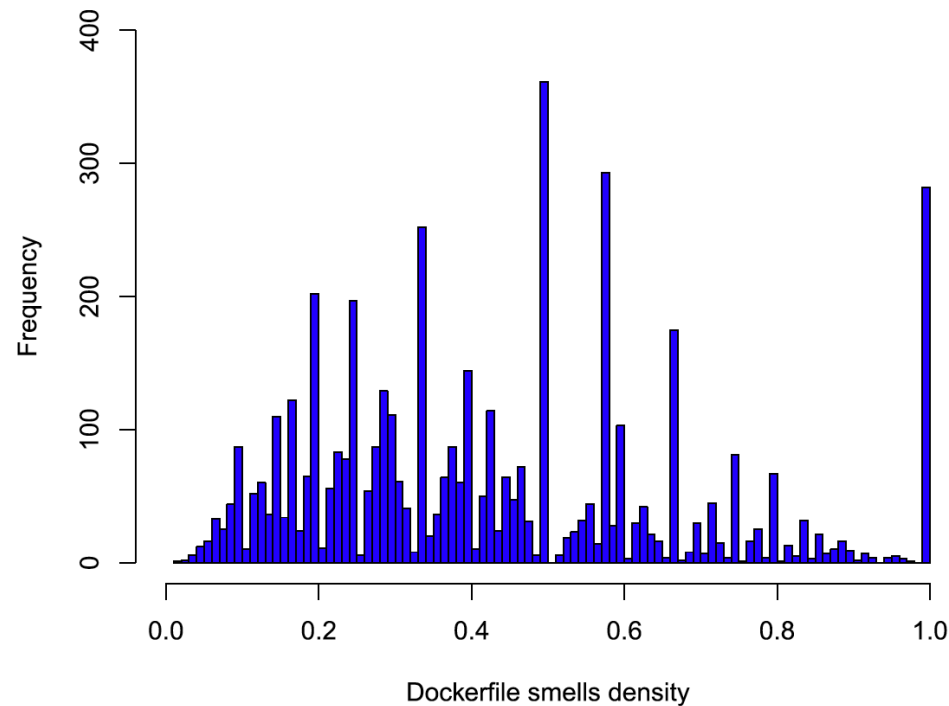


FIGURE 4. The distribution of Dockerfile smells density.

分析結果

RQ1 : Dockerfile smellsを持つプロジェクトはどの程度か

- プロジェクトのオーナータイプ, 言語との関係を調査

TABLE 2. Smells coverage in different project owner types.

Owner type	#projects	#S-projects	%S-projects	Avg. smells density
Organization	2,266	1,896	83.67%	61.42%
User	4,068	3,413	83.90%	61.58%

- オーナータイプによる違いはなかった

TABLE 3. Smells coverage in different programming languages.

Language	#projects↓	#S-projects	%S-projects	Avg. smells density
Shell	2,009	1,770	88.10%	62.63%
JavaScript	1,268	1,014	79.97%	56.81%
Python	973	816	83.86%	68.37%
Go	628	479	76.27%	70.98%
Java	364	305	83.79%	51.56%
Ruby	338	292	86.39%	59.72%
PHP	281	239	85.05%	51.18%
HTML	184	152	82.61%	60.13%
CSS	149	121	81.21%	61.98%
Makefile	140	121	86.43%	52.57%

- 言語ごとにsmellsを含む割合が異なっていた
- Shellを使用するプロジェクトの88%にsmellsが存在した

分析結果

RQ2 : DL-smellsとSC-smellsの分布に違いはあるか

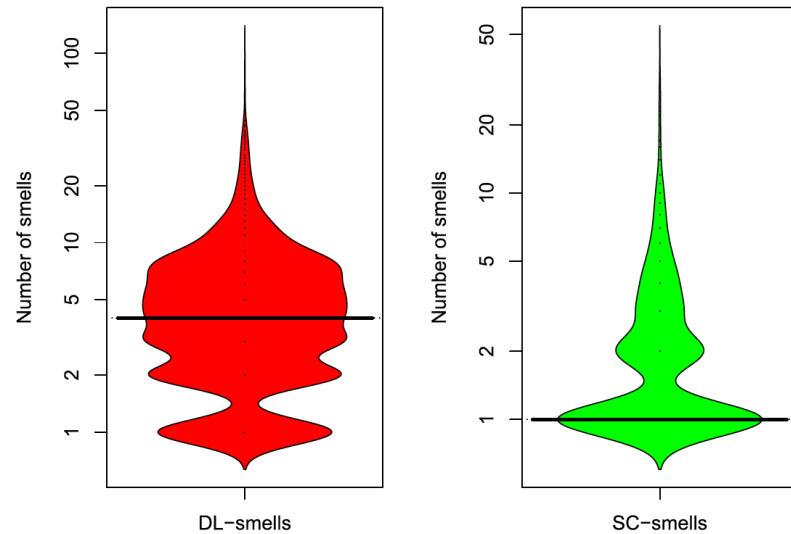


FIGURE 5. The distribution of DL-smells and SC-smells.

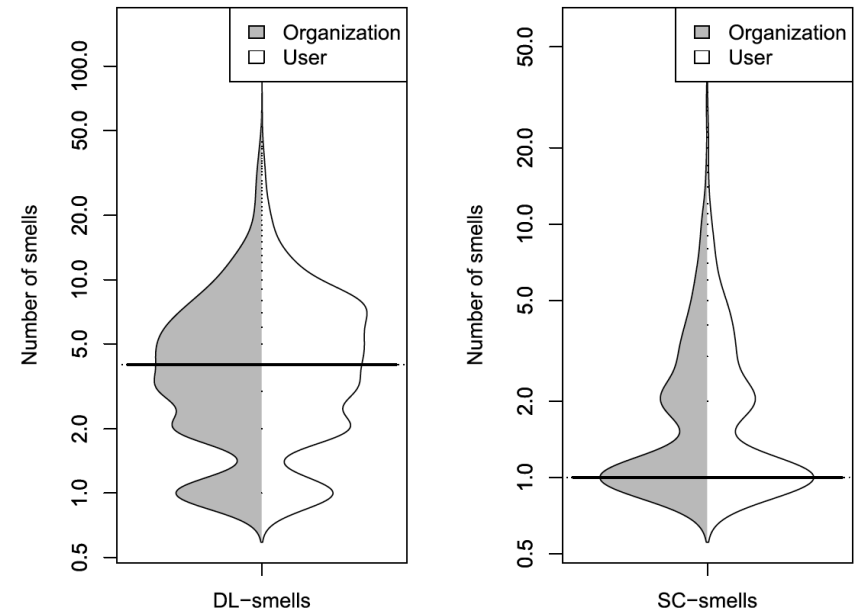


FIGURE 6. The distribution of smells in different project owner types.

- オーナータイプの違いによるsmellsの数は有意差があるが効果量は非常に小さかった (Wilcoxon検定, Cliff's d)

分析結果

RQ2 : DL-smellsとSC-smellsの分布に違いはあるか

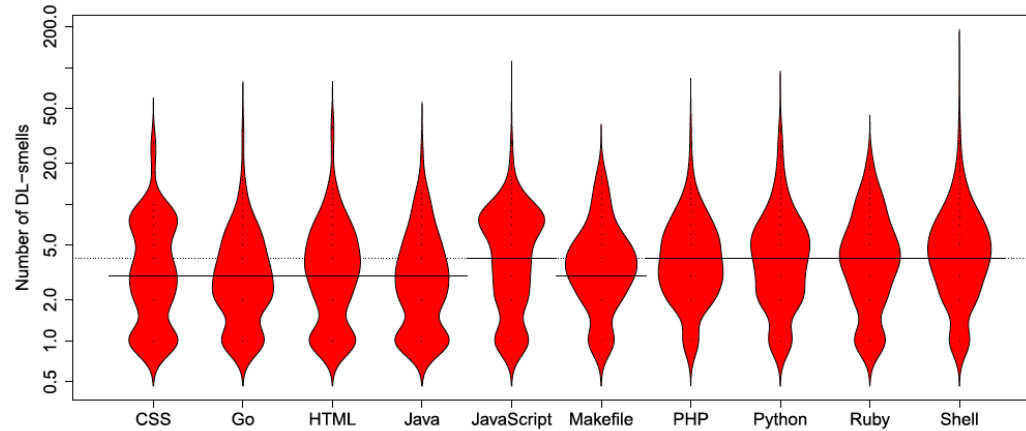


FIGURE 7. The distribution of DL-smells in different programming languages.

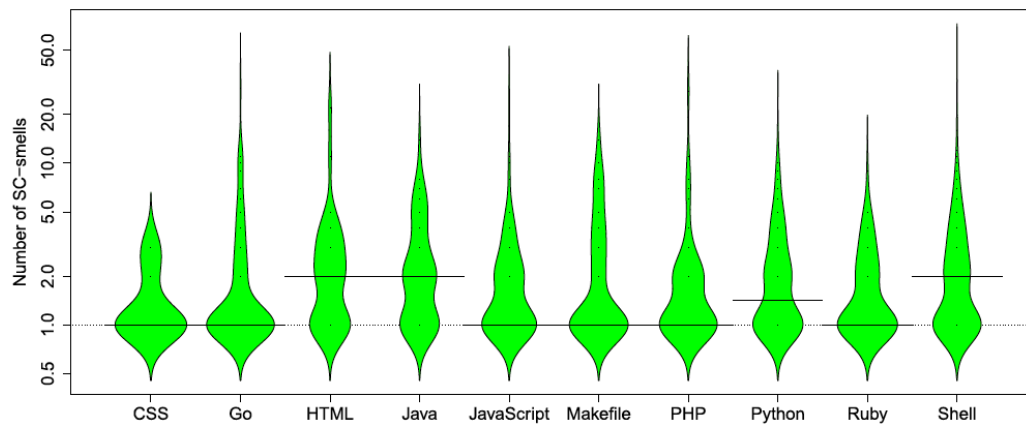


FIGURE 8. The distribution of SC-smells in different programming languages.

- 言語の違いによる smellsの数には有意差があった (Kruskal-Wallis検定)

分析結果

RQ2 : DL-smellsとSC-smellsの分布に違いはあるか

- DL-smellsとSC-smellsの共起性を調査

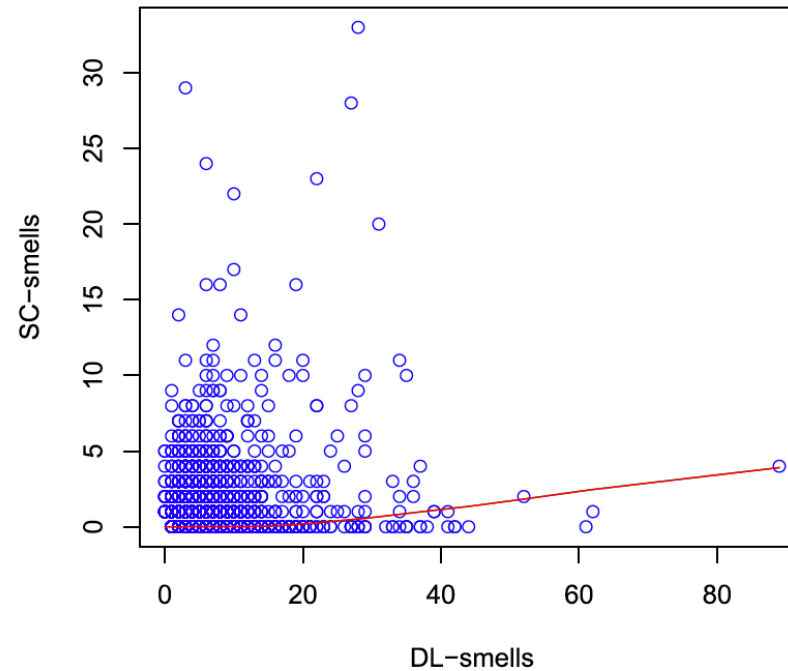


FIGURE 9. Co-occurrence between DL-smells and SC-smells (by volume).

- 2つのsmellsの間には正の相関があった (Spearmanの相関係数)

分析結果

RQ3 : Dockerfile smellsはプロジェクトの特性に依存するか

- プロジェクトの特性とsmellsの関係を調べるため3つの線形回帰モデルを作成

全てのsmellsのモデル

TABLE 6. Results of Overall smells model.

	Coef.	Err.	$Pr(< z)$	Sign.	Relationship
(Intercept)	-0.02	0.02	0.259		—
nInstructions	0.47	0.01	0.000	***	↗
nContributors	-0.02	0.01	0.053		—
nStars	-0.03	0.01	0.020	*	↘
githubAge	0.11	0.01	0.000	***	↗
ownerType=Organization	-0.03	0.02	0.170		—
language=CSS	-0.04	0.07	0.567		—
language=Go	-0.07	0.04	0.060		—
language=HTML	0.01	0.06	0.871		—
language=Java	-0.11	0.04	0.012	*	↘
language=JavaScript	0.05	0.03	0.049	*	↗
language=Makefile	0.02	0.07	0.807		—
language=PHP	-0.05	0.05	0.342		—
language=Python	0.09	0.03	0.001	**	↗
language=Ruby	0.02	0.05	0.644		—
language=Shell	0.07	0.02	0.170		—

*** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$

- 命令の数, プロジェクトの運用期間がsmellsの数に大きく関係していた
- スター数が多いほどsmellsは少なくなる傾向がある

まとめ

- DockerfileのCode smellsであるDockerfile smellsを調査した
- 83.8%のプロジェクトに少なくとも1つのDockerfile smellsが存在した
- オーナータイプによる違いはほとんどなかったが、言語の違いによるDockerfile smellsの発生には違いが見られた

所感

- 命令の数が多かったり古いプロジェクトであるほどsmellsが多いのは予想通りだった
- 言語の違いによるsmellsの発生が違ったのは興味深い
- 自分の研究でも言語ごとに発生しているエラーが異なっている可能性が高いので調査する必要がある

予備資料

DL-smellsのモデル

TABLE 7. Results of DL smells model.

	Coef.	Err.	$Pr(< z)$	Sign.	Relationship
(Intercept)	-0.02	0.02	0.356		—
nInstructions	0.42	0.01	0.000	***	↗
nContributors	-0.03	0.01	0.023	*	↘
nStars	-0.03	0.01	0.006	**	↘
githubAge	0.12	0.01	0.000	***	↗
ownerType=Organization	-0.06	0.02	0.018	*	↘
language=CSS	0.01	0.07	0.894		—
language=Go	-0.08	0.04	0.022	*	↘
language=HTML	-0.01	0.06	0.824		—
language=Java	-0.13	0.05	0.004	**	↘
language=JavaScript	0.11	0.03	0.000	***	↗
language=Makefile	-0.06	0.07	0.399		—
language=PHP	-0.04	0.05	0.442		—
language=Python	0.12	0.03	0.000	***	↗
language=Ruby	0.04	0.05	0.340		—
language=Shell	0.05	0.02	0.056		—

*** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$

SC-smellsのモデル

TABLE 8. Results of SC smells model.

	Coef.	Err.	$Pr(< z)$	Sign.	Relationship
(Intercept)	-0.02	0.02	0.366		—
nInstructions	0.34	0.01	0.000	***	↗
nContributors	0.01	0.01	0.664		—
nStars	0.01	0.01	0.586		—
githubAge	0.02	0.01	0.121		—
ownerType=Organization	0.06	0.03	0.012	*	↗
language=CSS	-0.16	0.07	0.027	*	↘
language=Go	0.03	0.04	0.504		—
language=HTML	0.07	0.06	0.243		—
language=Java	0.02	0.05	0.684		—
language=JavaScript	-0.15	0.03	0.000	***	↘
language=Makefile	0.23	0.07	0.001	**	↗
language=PHP	-0.04	0.05	0.435		—
language=Python	-0.03	0.03	0.320		—
language=Ruby	-0.06	0.05	0.190		—
language=Shell	0.08	0.03	0.001	**	↗

*** $p < 0.001$, ** $p < 0.01$, * $p < 0.05$