

研究発表

Dockerfileのビルドエラーの原因分析

辻 裕真

背景

- Dockerなどのコンテナ技術はソフトウェアのインフラを構築する軽量仮想化技術として近年関心を集めている.
- そのコンテナを構成するDockerfileのビルドは成功率が他の言語と比べて低く, 約34%ものDockerfileがビルドに失敗する.[1]
- Dockerに関する研究はパフォーマンスに関わるものが多く, ビルドに関する研究はあまり行われていない.

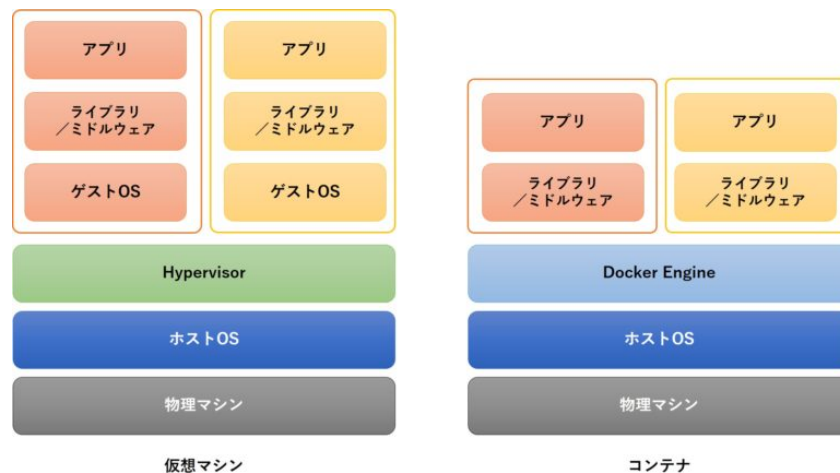
[1]Jürgen Cito, Gerald Schermann, John Erik Wittern, Philipp Leitner, Sali Zumberi and Harald C. Gall.
An Empirical Analysis of the Docker Container Ecosystem on GitHub.

目的

Dockerfileのビルドエラーの原因を明らかにする.

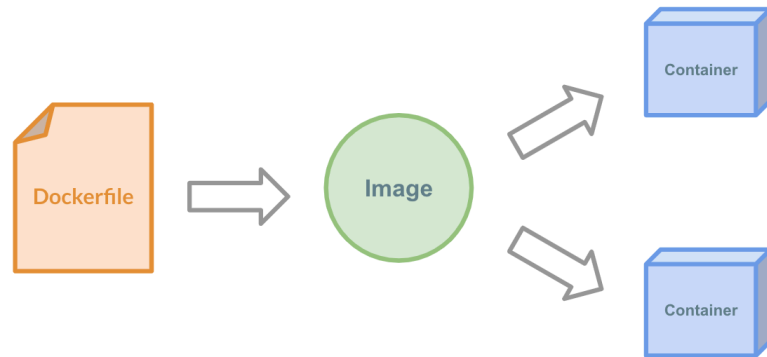
- ビルドログとDockerfileからビルドエラーの原因を調査し, どのような命令でエラーが起こったのか, よくエラーが起こる命令は何なのか, どう修正すればエラーが解消できるのかなどを明らかにしたい.
- その後の研究では, ビルドエラーが起こったDockerfileを自動で修正するシステムを開発したい.(機械学習など)

Docker(コンテナ)とは



- 別のホストOSをエミュレートする仮想マシンと異なり、コンテナは基盤となるカーネルをホストと共有するため、オーバーヘッドが大幅に削減され、高速に実行できるアプリケーション環境である。
- Dockerはコンテナを作成、実行するためのプラットフォームの1つである。

Dockerfile



Dockerfileの例

```
FROM centos

CMD echo "now running..."

RUN yum install -y httpd

ADD ./index.html /var/www/html/

EXPOSE 80

CMD ["/usr/sbin/httpd", "-D", "FOREGROUND"]
```

- コンテナの構成内容を記述しているテキスト形式のファイルである.
- コンテナのベースとなるイメージを指定したり, コマンドの使用, ファイルの追加などの命令を記述している.

Research Questions

RQ1 : Dockerfileのビルドエラーの原因は何か.

- Dockerfileをビルドし, 失敗した時にDockerfileのどこで, なぜエラーが起こったのかを調査し, 特徴ごとに分類する.

RQ2 : ビルドログやLinter(Dockerfile静的解析ツール)の結果から原因を予測することは可能か.

- Dockerfileをビルドした際のビルドログやLinterの結果をまとめて, エラーの特徴などと繋がりが無いかを調べて, 自動的にエラーの原因を特定する.

調査方法

- MSRのDockerのデータセット[2](GitHub上のDockerfileが含まれているプロジェクトのセット, プロジェクト数は16959)でプロジェクトをフォーク数, スター数で絞る(基準は未定).
- プロジェクト内のDockerfileが変更されたコミットを特定し, 各リビジョンでビルドテスト(docker build, Linter, ビルド時間の計測など)を行い, ビルドが失敗した場合は原因の特定を行い, 以降のリビジョンでどう修正されたかを調査する.

[2]<https://github.com/sealuzh/msr18-docker-dataset>

現在行っていること

- ・ データセットのフォーク数が上位のプロジェクトからビルドテストを行い、ビルド失敗の原因、修正方法などを調査している.
- ・ 現在はビルドテストを手動で行なっているので、自動で各リビジョンでビルドを行うスクリプトを作成中.

今後の作業

- ・ スクリプトを完成させ、自動的にビルドテストをどんどん行なっていく.
- ・ プロジェクト選択の基準を決める.(他の研究などを参考にしてみる)
- ・ 論文を書き始める.