Treelite (dev, commit bd5529a) (#)

Go to version: latest

TABLE OF CONTENTS

Search...

Docs / Treelite : toolbox for decision tree deployment

Treelite : toolbox for decision tree deployment

Treelite is a flexible toolbox for efficient deployment of decision tree ensembles.

Star 62 Watch 22

You are currently browsing the documentation of a development version of treelite: commit **bd5529a**.

Why treelite?

Use machine learning package of your choice

Treelite accommodates a wide range of decision tree ensemble models. In particular, it handles both random forests (https://en.wikipedia.org/wiki/Random_forest) and gradient boosted trees (https://en.wikipedia.org

v: latest ▾

Treelite (dev, commit bd5529a) (#)

Go to version: latest

TABLE OF CONTENTS

/wiki/Gradient_boosting).

Treelite can read models produced by XGBoost (https://github.com/dmlc/xgboost/), LightGBM (https://github.com/Microsoft/LightGBM), and scikit-learn (https://github.com/scikit-learn/scikit-learn). In cases where you are using another package to train your model, you may use the flexible builder class (tutorials/builder.html).

Deploy with minimal dependencies

It is a great hassle to install machine learning packages (e.g. XGBoost, LightGBM, scikit-learn, etc.) on every machine your tree model will run. This is the case no longer: treelite will export your model as a **stand-alone prediction subroutine** so that predictions will be made without any machine learning package installed.

Compile and optimize your model for fast prediction

Treelite optimizes the prediction subroutine for faster prediction.

Depending on your use cases, simply compiling the prediction subroutine into machine code

V: latest ▾

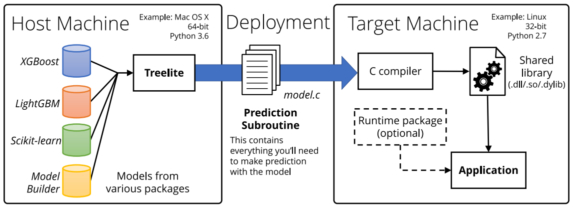
Treelite (dev, commit bd5529a) (#)

Go to version: latest

TABLE OF CONTENTS

(https://en.wikipedia.org/wiki/Machine_code) may boost the performance noticeably. (See the benchmark section below.) In addition, treelite supports additional optimizations (tutorials/optimize.html) that improve performance while preserving the ensemble model.

How treelite works



(Click to enlarge) (_static/deployment.png)

The workflow involves two distinct machines: **the host machine** that generates prediction subroutine from a given tree model, and **the target machine** that runs the subroutine. The two machines exchange a single C file that contains all relevant information about the tree model. Only the host machine needs to have treelite installed; the target machine requires only a working C compiler.

Quick start

v: latest ▾

Treelite (dev, commit bd5529a) (#)

Go to version: latest

TABLE OF CONTENTS

Install treelite from PyPI:

```
pip3 install --user treelite
```

Import your tree ensemble model into treelite:

```
import treelite
model = treelite.Model.load('my_model
```

Deploy a source archive:

```
# Produce a zipped source directory, co
# Run `make` on the target machine
model.export_srcpkg(platform='unix', to
                    pkgpath='./mymodel.zip', lib
                    verbose=True)
```

Deploy a shared library:

```
# Like export_srcpkg, but generates a sh
# Use this only when the host and targe
model.export_lib(toolchain='gcc', libpat
```

Make predictions on the target machine:

```
import treelite.runtime
predictor = treelite.runtime.Predictor('./
batch = treelite.runtime.Batch.from_npy
out_pred = predictor.predict(batch, verb
```

v: latest ▾

Read First tutorial (tutorials/first.html)

Treelite (dev, commit bd5529a) (#)

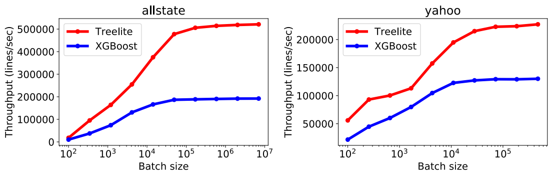
Go to version: latest

TABLE OF CONTENTS

for a more detailed example. See Deploying models (tutorials/deploy.html) for additional instructions on deployment.

> **Note**
>
> A note on API compatibility
>
> Since treelite is in early development, its API may change substantially in the future.

Benchmark



See the page Benchmark (benchmark.html) for details.

Contents

v: latest ▾

Treelite (dev, commit bd5529a) (#)

Go to version: latest

TABLE OF CONTENTS

Indices

- Index (genindex.html)
- Module Index (py-modindex.html)

Installation (install.html)

v: latest ▾

Treelite (dev, commit bd5529a) (#)

Go to version: latest

TABLE OF CONTENTS

v: latest ▾