ONNX

ONNX Model Zoo/Tutorials Sig Updates

Presenter:
Jacky Chen (Microsoft US)

Outlines

- Latest ONNX Model Zoo models
- Next generation of ONNX Model Zoo
  - New upload requirements
  - Bring more new models from MLAgility
  - Deprecate old models
  - Web interface for ONNX Model Zoo
- Roadmap

ONNX

ONNX Model Zoo

a collection of pre-trained, state-of-the-art models

Latest ONNX Model Zoo models

- 182 models in total
- New preprocessing model
- More quantized models (int8, qdq)
- Enhance CIs: codeql; validate JSON for ONNX Hub
- Weekly test version conversion from latest ONNX

Next generation of ONNX Model Zoo

- Motivations
  - Existing models are hard to reproduce with outdated script
  - Existing models are still using old opset versions (opset_version < 13)
  - Same model usually has few versions
  - No sufficient state-of-the-art models in the past 2 years
- Proposals
  - Propose new upload requirements
  - Utilize benchmark tool (MLAgility) to verify uploaded models
  - Bring more new models from transformers/keep single version of model
  - Deprecate old models
  - Have a new web interface

New upload requirements

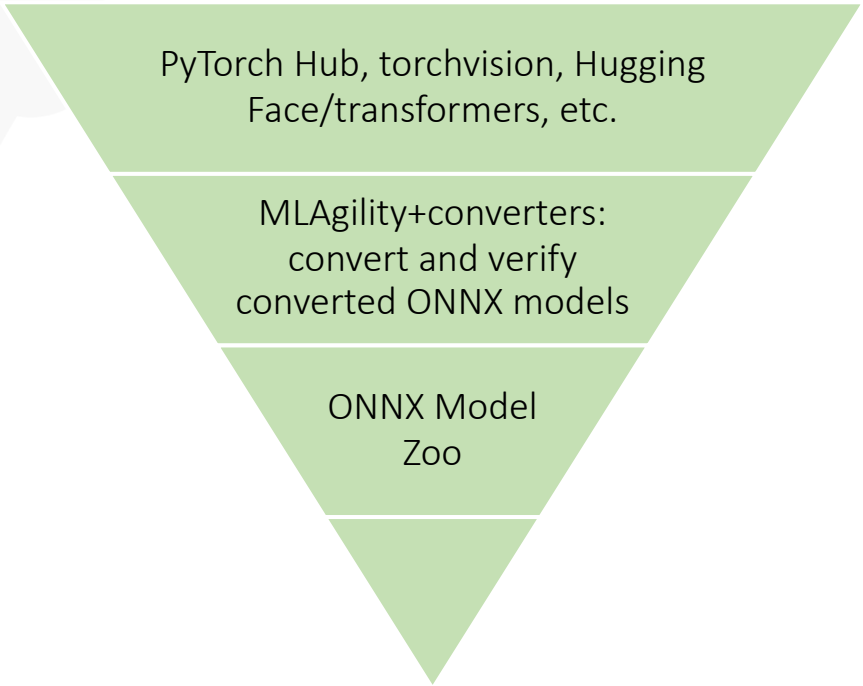
- Single .onnx file (git-lfs): For instance, bert-18.onnx
- test_data_set_0 (git-lfs): a directory containing the test data set
- README.md: a readme file describing the model and how to use
- LICENSE file: a standalone license file for the model. For instance, MIT
- (New) model.py: a python to reproduce .onnx model from original framework
- (New) requirements.txt: a text file listing all the required Python packages and their versions

New upload requirements: CI

- Under new directory: models/python/. e.g., models/python/bert-18/
- Model tags will be obtained from the main README.md
- CIs will help verification
  - Run onnx/onnxruntime on models and existing data
  - Rerun reproduction script to ensure the models is reproducible
  - Run MLAgility to check

Bring more new models from MLAgility

- MLAgility from Groq has great benchmark on a lot of state-of-the-art ONNX models from transformers, torch hub, torch vision
- Have a config file to run mlagility to get converted models and store them in ONNX Model Zoo
- Also replace existing old models

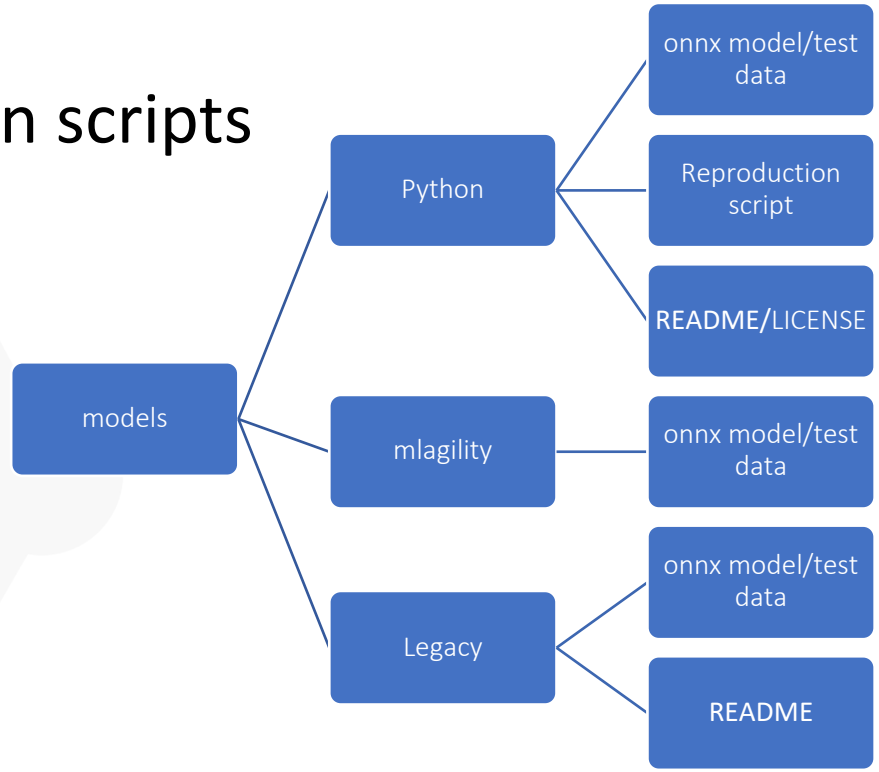


Deprecate old models

- All existing ONNX models (whose opset_version < 13) will be moved to directory models/legacy
- Users can still get them through onnx.hub
- Users are encouraged to use newer models with newer opset_version
- Will be removed once ONNX has sufficient new models

New model directory hierarchy

- models/python: new models with reproduction scripts
  - model/python/bert-18/bert-18.onnx with test_data_set
  - Reproduction script, README.md, License file
  - User facing
- models/mlagility: new models from mlagility
  - model/mlagility/bert-18/bert-18.onnx with test_data_set
  - The reproduction script will be found in groq/mlagility
  - We will bring more new models from there
  - Frequently update/verify these models
- models/legacy: old models with opset_version < 13



Web interface from ONNX Model Zoo

- Thanks Krishna from Groq for contributing web interface for model zoo
- If interested, feel free to join his later talk today

Roadmap (ONNX Model Zoo)

- Deal with legacy operators and models
- Introduce more state-of-the-art models
- Ensure models are reproducible
- Focus on base models and provide more detailed tutorials for optimization and quantization
- ONNX hub will support to download all kinds of models
- More frequently update opset_version in ONNX Model Zoo

Welcome to contribute!

- Discussion: join us on Slack in #onnx-modelzoo channel
- Help to review pull requests Upload new ONNX models

Files needed for PR

- ONNX Model file
- requirements.txt
- Reproduction Python script
- Test input/output data
- README.md
- LICENSE file

Model verification

- Ensure model is reproducible from provided script
- Check by onnx.checker/shape_inference
- ORT inference test on test data with CPU EP
- Verify by MLAgility

Q&A