Нижегородский государственный университет им. Н.И. Лобачевского
Институт информационных технологий, математики и механики

Компиляторы для глубоких нейросетевых моделей

Обзор тензорных компиляторов

При поддержке компании YADRO

Сысоев А.В.

Содержание

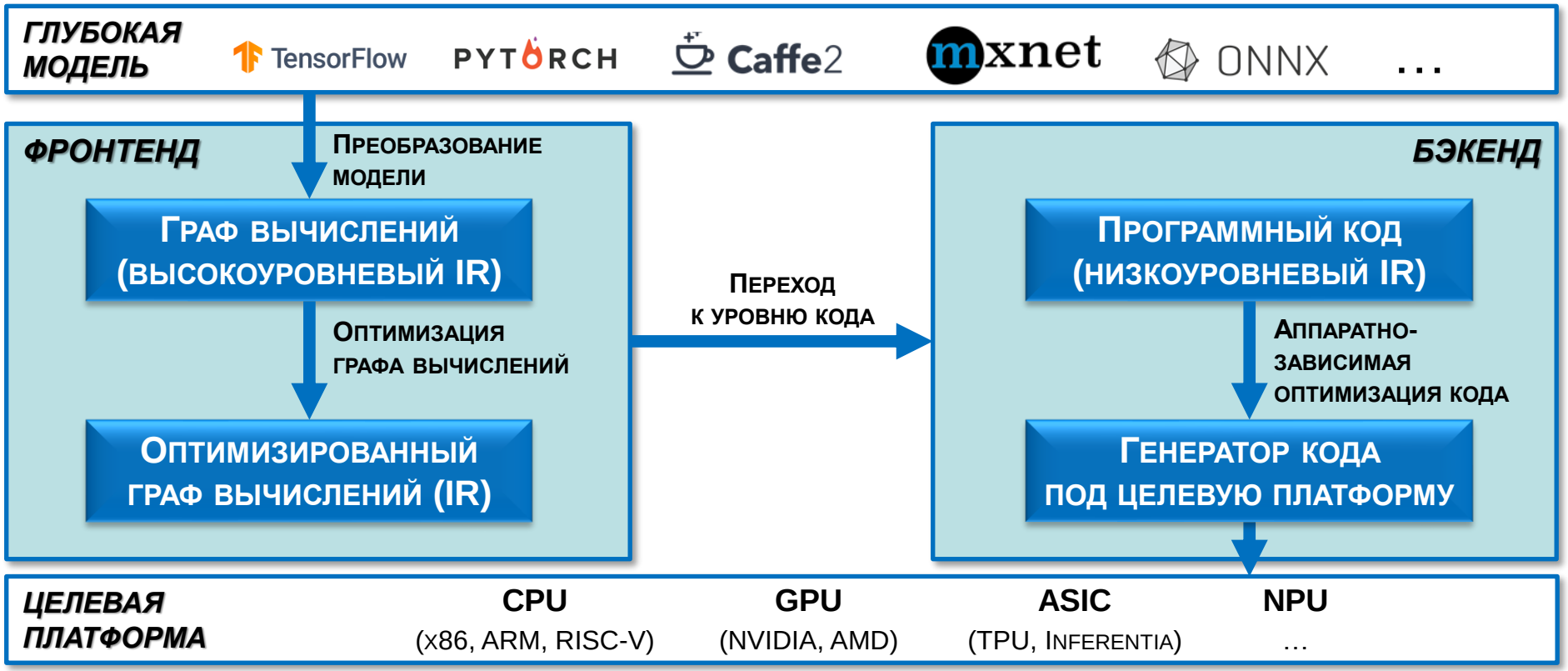
НАЗНАЧЕНИЕ ТЕНЗОРНЫХ КОМПИЛЯТОРОВ

Назначение тензорных компиляторов

- ***Тензорный компилятор*** – отдельный инструмент или компонент какого-либо фреймворка, который обеспечивает преобразование и оптимизацию глубокой нейросетевой модели в исполняемый формат под конкретное устройство

- В процессе преобразования тензорные компиляторы используют различные варианты высокоуровневого (типично, граф вычислений) и низкоуровневого ***промежуточного представления (intermediate representation, IR)***

Общая архитектура тензорных компиляторов



ОБЗОР ТЕНЗОРНЫХ КОМПИЛЯТОРОВ

Обзор тензорных компиляторов

- На данный момент ML-сообществом разрабатываются десятки проектов, являющихся или содержащих тензорные компиляторы

- Существующие проекты перечислены по ссылке [https://github.com/merrymercy/awesome-tensor-compilers]

- Все рассмотренные далее тензорные компиляторы, в целом, подчиняются представленной ранее архитектуре

Обзор тензорных компиляторов. Техническая информация

	ОС	Платформы	Интерфейсы
Glow	macOS, Linux	CPU	Python, C++
XLA	Windows, Linux, macOS	GPUs, CPUs, ML accelerators	Python
NNFusion	Linux	CPU, GPU	Python
OpenVINO* (nGraph)	Windows, Linux, macOS	CPU, GPU, NPU, FPGA компании Intel, ARM CPU, RISC-V CPU**	Python, C++
Apache TVM	Windows, Linux, macOS	CPU, GPU	Python, C++

* OpenVINO – самостоятельный инструментарий, включающий разнообразный функционал (от реализации эффективного вывода до анализа производительности вывода)

** Поддержка RISC-V в OpenVINO находится в процессе разработки (оптимизированы некоторые операции)

План рассмотрения каждого тензорного компилятора…

- ***Ключевые особенности***
  - Как разработчики позиционируют свой инструмент?
- ***Общая информация***
  - Где найти инструмент и информацию о нем?
  - Какая лицензия?
  - Какие языки программирования используются в разработке?
- ***Интеграция в известные фреймворки глубокого обучения***
  - В какие фреймворки для вывода глубоких моделей можно интегрировать тензорный компилятор, чтобы ускорить вычисления?
- ***Поддерживаемые форматы глубоких моделей***
  - Какие форматы хранения моделей поддерживаются для запуска вывода?

План рассмотрения каждого тензорного компилятора

- ***Встроенная поддержка понижения точности весов моделей и/или запуск моделей с соответствующими весами***
  - Существуют ли встроенные инструменты для понижения точности весов моделей (переход от весов в формате fp32 в fp16, int8 или int16) и последующего запуска моделей?
- ***Интеграция высокопроизводительных библиотек для базовых вычислительных примитивов (kernel libraries)***
  - Использует ли тензорный компилятор сторонние библиотеки, содержащие оптимизированные реализации базовых вычислительных операций под конкретное аппаратное обеспечение?
- ***Возможность запуска моделей с динамической формой входных данных***
  - Поддерживается ли оптимизация и вывод моделей, которые могут принимать на вход и обрабатывать данные произвольной формы (например, тензора с вариабельным размером входной пачки данных)?

GLOW

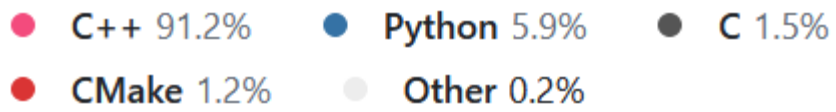
Glow…

- ***Glow (Graph-Lowering)*** – компилятор машинного обучения, который ускоряет работу фреймворков глубокого обучения на различных аппаратных платформах

- ***Ключевые особенности:***
  - Glow предназначен для использования в качестве бэкенда для высокоуровневых фреймворков глубокого обучения
  - Позволяет разработчикам оборудования и исследователям сосредоточиться на создании аппаратных ускорителей следующего поколения, которые могут поддерживаться фреймворками глубокого обучения, такими как PyTorch

Glow…

- ***Статья:*** Rotem N., et al. Glow: Graph Lowering Compiler Techniques for Neural Networks. – 2019. – [https://arxiv.org/abs/1805.00907].
- ***Web-site:*** https://ai.meta.com/tools/glow
- ***Репозиторий:*** https://github.com/pytorch/glow
- ***Лицензия:*** Apache-2.0 license
- ***Статус:*** активная разработка
- ***Сообщество:***
  - Более 680 форков репозитория
  - Более 240 контрибьюторов
- ***Языки разработки:***



Glow

- ***Интеграция в известные фреймворки глубокого обучения***
  - PyTorch (в процессе разработки, можно собрать из исходников)
- ***Поддерживаемые форматы глубоких моделей***
  - Caffe2, ONNX, PyTorch, TensorFlow Lite
- ***Встроенная поддержка понижения точности весов моделей и/или запуск моделей с соответствующими весами***
  - INT8 / INT16 (согласно [https://github.com/pytorch/glow/blob/master/docs/Quantization.md])
- ***Интеграция высокопроизводительных библиотек для базовых вычислительных примитивов (kernel libraries)***
  - Отсутствует
- ***Возможность запуска моделей с динамической формой входных данных***
  - Отсутствует (согласно [https://arxiv.org/pdf/2002.03794])

XLA

XLA…

OpenXLA

- ***XLA (accelerated linear algebra)*** – компилятор с открытым исходным кодом, оптимизирует модели глубокого обучения в формате популярных фреймворков для эффективного выполнения на CPUs, GPUs и ускорителях машинного обучения

- ***Ключевые особенности:***
  - *Производительность*. XLA сокращает время выполнения операций, уменьшает накладные расходы на память, реализует агрессивное распространение констант, и другие техники оптимизации
  - *Использования памяти*. XLA анализирует использование памяти, устраняя промежуточные буферы хранения
  - *Переносимость*. XLA упрощает написание новой серверной части для нового оборудования, чтобы большая часть моделей машинного обучения могла работать на этом оборудовании без изменений

XLA…

OpenXLA

- ***Web-site:*** https://openxla.org/xla
- ***Репозиторий:*** https://github.com/openxla/xla
- ***Лицензия:*** Apache-2.0 license
- ***Статус:*** активная разработка
- ***Сообщество:***
  - Более 340 форков репозитория
  - Более 600 контрибьюторов
- ***Языки разработки:***

XLA…

OpenXLA

- ***Интеграция в известные фреймворки глубокого обучения***
  - TensorFlow, Keras [https://openxla.org/xla/tf2xla]
  - PyTorch [https://github.com/pytorch/xla] (Google TPUs, NVIDIA GPUs)
  - JAX [https://github.com/jax-ml/jax]
- ***Поддерживаемые форматы глубоких моделей***
  - TensorFlow, Keras (пример [https://huggingface.co/docs/transformers/en/tf_xla])
- ***Встроенная поддержка понижения точности весов моделей и/или запуск моделей с соответствующими весами***
  - INT8 / INT16
  - Исполнение квантованных операций в экспериментальном режиме (согласно [https://pytorch.org/xla/master/quantized_ops.html])

XLA

OpenXLA

- ***Интеграция высокопроизводительных библиотек для базовых вычислительных примитивов (kernel libraries)***
  - Eigen, MKL, cuDNN, TensorRT
- ***Возможность запуска моделей с динамической формой входных данных***
  - Имеется

NNFUSION

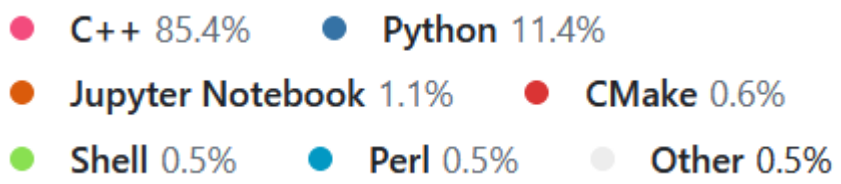
NNFusion…

- ***NNFusion (Rammer)*** – гибкий и эффективный компилятор глубоких нейронных сетей, который может генерировать высокопроизводительный исполняемый код на основании описания модели

- ***Ключевые особенности:***
  - Полнофункциональный механизм оптимизации
  - Поддержка популярных форматов, включая TensorFlow и ONNX, в качестве входных моделей
  - Поддержка настраиваемой оптимизации
  - Поддержка устройств: CPUs, NVIDIA GPUs, ROCm GPUs
  - Поддержка параллельного обучения через проект SuperScaler [https://github.com/microsoft/SuperScaler]

NNFusion…

- ***Статья:*** Ma L., et al. RAMMER: enabling holistic deep learning compiler optimizations with rtasks // In Proc. of the 14th USENIX Conf. on Operating Systems Design and Implementation (OSDI'20). Article 50. – 2020. – P. 881-897.
- ***Web-site:*** https://www.microsoft.com/en-us/research/project/deep-learning-compiler-and-optimizer
- ***Репозиторий:*** https://github.com/microsoft/nnfusion
- ***Лицензия:*** MIT license
- ***Статус:*** разработка заморожена (последние коммиты летом 2022 года)
- ***Сообщество:***
  - Более 150 форков репозитория
  - 23 контрибьютора
- ***Языки разработки:***



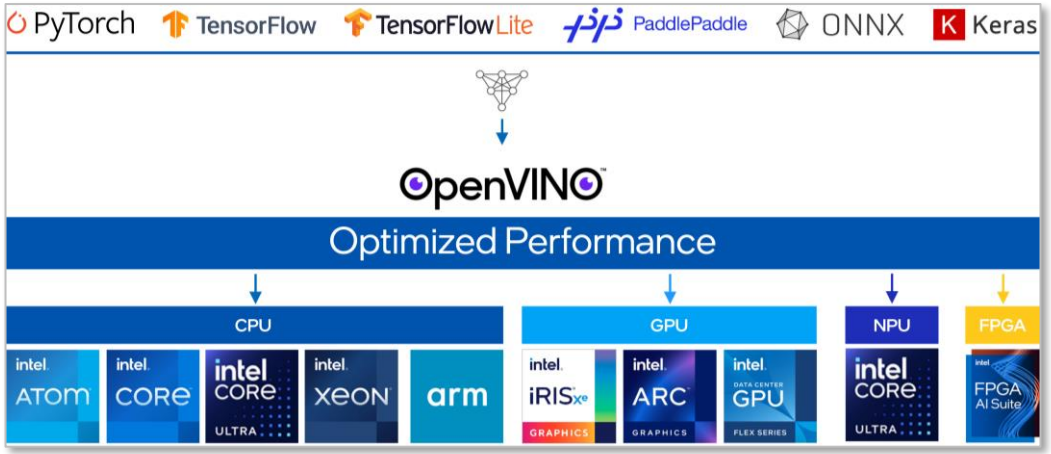
NNFusion

- ***Интеграция в известные фреймворки глубокого обучения***
  - Отсутствует
- ***Поддерживаемые форматы глубоких моделей***
  - ONNX, TensorFlow frozen (.pb)
- ***Встроенная поддержка понижения точности весов моделей и/или запуск моделей с соответствующими весами***
  - Не поддерживается
- ***Интеграция высокопроизводительных библиотек для базовых вычислительных примитивов (kernel libraries)***
  - cuBlas, cuDNN, mkl, Eigen, CBLAS (исходя из CMake-файлов и исходных кодов)
- ***Возможность запуска моделей с динамической формой входных данных***
  - Информация отсутствует

OPENVINO

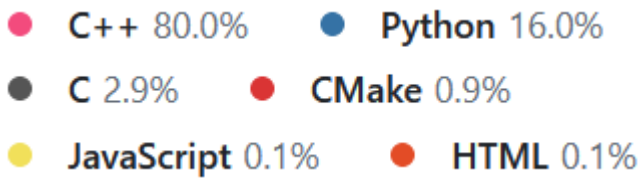
OpenVINO…

- ***OpenVINO*** – набор инструментов с открытым исходным кодом для оптимизации и развертывания моделей глубокого обучения от облака до периферии
- ***Ключевая особенность:***
  - Поддержка всего «парка» устройств Intel, ARM; RISC-V в процессе разработки



OpenVINO…

- ***Web-site:*** https://docs.openvino.ai
- ***Репозиторий:*** https://github.com/openvinotoolkit/openvino
- ***Лицензия:*** Apache-2.0 license
- ***Статус:*** активная разработка
- ***Сообщество:***
  - Более 2 100 форков репозитория
  - Более 500 контрибьюторов
- ***Языки разработки:***



OpenVINO…

- ***Интеграция в известные фреймворки глубокого обучения***
  - ONNX Runtime [https://onnxruntime.ai/docs/execution-providers/OpenVINO-ExecutionProvider.html]
  - PyTorch [https://docs.openvino.ai/2024/openvino-workflow/torch-compile.html]
- ***Поддерживаемые форматы глубоких моделей***
  - PyTorch
  - TensorFlow
  - TensorFlow Lite
  - PaddlePaddle
  - ONNX
  - Keras

OpenVINO

- ***Встроенная поддержка понижения точности весов моделей и/или запуск моделей с соответствующими весами***
  - INT8 / FP16 (с использованием Neural Network Compression Framework, NNCF)
- ***Интеграция высокопроизводительных библиотек для базовых вычислительных примитивов (kernel libraries)***
  - Intel® oneAPI Deep Neural Network Library (oneDNN)
- ***Возможность запуска моделей с динамической формой входных данных***
  - Имеется (при запуске вывода можно явно установить размеры входного тензора)

APACHE TVM

Apache TVM…

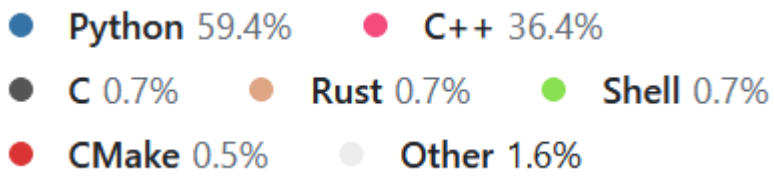
- ***Apache TVM (Tensor Virtual Machine)*** – тензорный компилятор с открытым исходным кодом, специализируется на ***выполнении*** нейронных сетей на CPUs, GPUs и ускорителях машинного обучения

- ***Основные возможности:***
  - Компиляция моделей глубокого обучения в развертываемые (исполняемые на устройстве) модули
  - Предоставление инфраструктуры для автоматического создания и оптимизации моделей на большом количестве бекэндов* для обеспечения высокой производительности

*Бекэнд – программно-аппаратное окружение.

Apache TVM…

- ***Статья:*** Chen T., et al. TVM: An Automated End-to-End Optimizing Compiler for Deep Learning. – 2018. – [https://arxiv.org/abs/1802.04799].
- ***Web-site:*** https://tvm.apache.org
- ***Репозиторий:*** https://github.com/apache/tvm
- ***Лицензия:*** Apache-2.0 license
- ***Статус:*** активная разработка
- ***Сообщество:***
  - Более 3 400 форков репозитория
  - Более 900 контрибьюторов
- ***Языки разработки:***



Apache TVM…

- ***Интеграция в известные фреймворки глубокого обучения***
  - ONNX Runtime [https://onnxruntime.ai/docs/execution-providers/community-maintained/TVM-ExecutionProvider.html]
- ***Поддерживаемые форматы глубоких моделей***
  - PyTorch, ONNX, TensorFlow Lite, Caffe, MXNet, TensorFlow (tf2onnx), Keras
  - Конвертация и компиляция моделей с помощью встроенных инструментов или сторонних пакетов
- ***Встроенная поддержка понижения точности весов моделей и/или запуск моделей с соответствующими весами***
  - INT8 (подробнее в [https://arxiv.org/pdf/2006.10226v1], провалидирована для некоторых классов моделей)
  - Обеспечивается запуск квантованных моделей (согласно документации [https://tvm.apache.org/docs/how_to/deploy_models/deploy_prequantized.html])

Apache TVM

- ***Интеграция высокопроизводительных библиотек для базовых вычислительных примитивов (kernel libraries)***
  - BLAS, MKL (blas), cuBLAS, cuDNN, cutlass
- ***Возможность запуска моделей с динамической формой входных данных***
  - Имеется (осуществляется через Virtual Machine API)

Резюме

	Glow	XLA	OpenVINO	Apache TVM
Форматы моделей	Caffe2, ONNX, PyTorch, TensorFlow Lite	TensorFlow, Keras	PyTorch, ONNX, TensorFlow, TensorFlow Lite, PaddlePaddle, Keras	PyTorch, ONNX, Caffe, MXNet, TensorFlow, TensorFlow Lite (tf2onnx), Keras
Интеграция в фреймворки	PyTorch (в разработке)	TensorFlow, Keras, PyTorch, JAX	ONNX Runtime, PyTorch	ONNX Runtime
Понижение точности весов	INT8 / INT16	INT8 / INT16 (запуск в эксп. р.)	INT8 / FP16 (NNCF)	INT8
Интеграция сторонних библиотек	–	Eigen, MKL, cuDNN, TensorRT	oneDNN	BLAS, MKL (blas), cuBLAS, cuDNN, cutlass
Динамическая форма входных данных	–	+	+ (указывается перед выводом)	+ (Virtual Machine API)

Выводы

- Несмотря на то, что область тензорных компиляторов является относительно молодой, существует некоторое количество разработок, которые отличаются доступным функционалом
- Из активно-разрабатываемых в настоящее время тензорных компиляторов одним из лидеров является Apache TVM:
  - Наибольшее по охвату сообщество (по данным github)
  - Поддержка наиболее популярных платформ
  - Поддержка большого числа форматов глубоких моделей
- Apache TVM более детально рассматривается далее в курсе

Заключение

- Несмотря на разнообразие тензорных компиляторов, все они решают одну и ту же задачу – обеспечивают оптимизацию и преобразование глубокой нейросетевой модели в исполняемый формат под конкретное вычислительное устройство

Литература

1. Awesome Tensor Compilers [https://github.com/merrymercy/awesome-tensor-compilers].
2. Xing Y., et al. An In-depth Comparison of Compilers for Deep Neural Networks on Hardware // IEEE International Conference on Embedded Software and Systems (ICESS) – 2019. – P. 1-8.
3. Mingzhen Li, et al. The Deep Learning Compiler: A Comprehensive Survey // IEEE Transactions on Parallel and Distributed Systems, 32. – 2021. – P. 708-727.

Авторский коллектив

- Мееров Иосиф Борисович, к.т.н., доцент, зав. каф. ВВиСП Института ИТММ ННГУ им. Н.И. Лобачевского
- Кустикова Валентина Дмитриевна, к.т.н., доцент каф. ВВиСП Института ИТММ ННГУ им. Н.И. Лобачевского
- Родимков Юрий Александрович, младший научный сотрудник каф. ВВиСП Института ИТММ ННГУ им. Н.И. Лобачевского
- Сысоев Александр Владимирович, к.т.н., доцент каф. ВВиСП Института ИТММ ННГУ им. Н.И. Лобачевского