MAD24-403

Linaro Connect
MADRID 2024 | MAY 14-17 2024

Optimising oneDAL & oneDNN for Arm CPUs to accelerate AI workloads

Ragesh Hajela
Software Engineering Manager
MONAKA Software R&D (HPC AI) Unit
Fujitsu Research of India, Bengaluru

About Me

Ragesh Hajela

Software Engineering Manager
Fujitsu Research of India Pvt Limited (FRIPL)
Email ID: ragesh.hajela@fujitsu.com

Introduction

Leading AI software R&D at Fujitsu Research of India, Bengaluru, focussing on Arm HPC Framework Engineering with Machine Learning based Computational Data Science.

2023 — Software Engineering Manager, Fujitsu Research of India
- Leading ML Framework Engineering at FUJITSU-MONAKA SW R&D Unit
- Working with Advanced Technology Development Unit (ATDU, Japan)

2021 — AI Frameworks Engineer at Intel Corporation
- OpenVINO Toolkit Inference Acceleration Software Development

2019 — Principal Machine Learning Engineer, ConcertAI
- AI Software Engineering for Healthcare AI Oncology Research

2013 — Member of Technical Staff, Amadeus Labs
- ML Software development for Travel Intelligence Platform

2011 — Software Engineer, Tata Consultancy Services
- C++ Programmer for Software Application Development

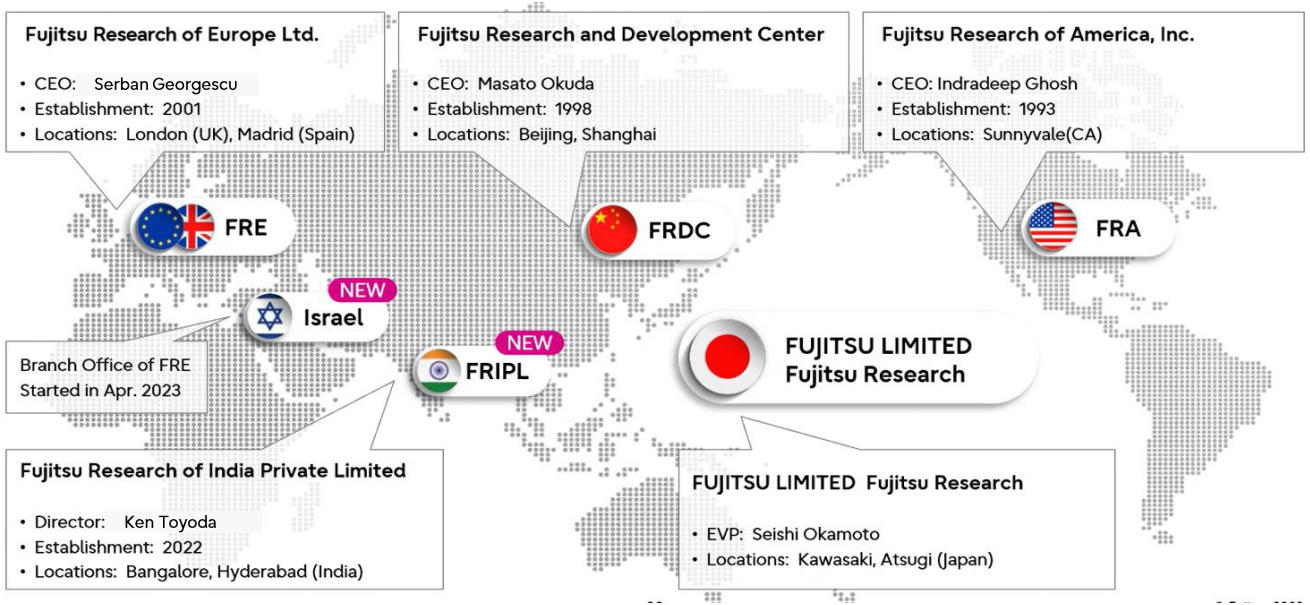
About Fujitsu

Our Team at FRIPL

- Expertise in AI Software Framework Engineering and Optimization with ML, DL, NLP, generative AI, LLMs, Real-time data processing & Cloud-based Data Security.
- **Open-source contributors** for accelerated software in HPC AI core technologies:

  Scikit Learn, BLAS, Math Libs, PyTorch, TensorFlow, Hugging Face, LLMs etc.

Fujitsu Research Locations

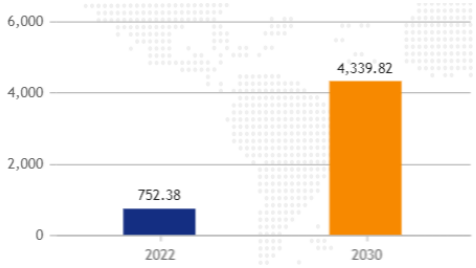


Outline of Presentation

- Introduction about AI workloads optimization
- Fujitsu’s contribution for Scikit Learn oneDAL porting on Arm
- Fujitsu’s recent advancements in oneDNN and JIT Kernel on Arm
- Conclusion and Future Work
- Resources and Acknowledgement

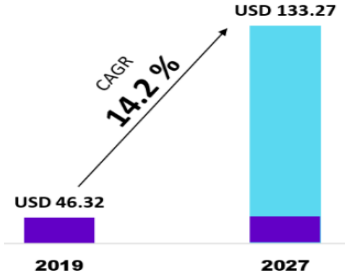
Project Vision

Data Processing increasing rapidly through Innovative Digital Transformation → Data Processing is growing demand for large scale data centers (DC) → Power Consumption is rising with expansion of DCs demand → High power consumption is increasing carbon footprint & global warming



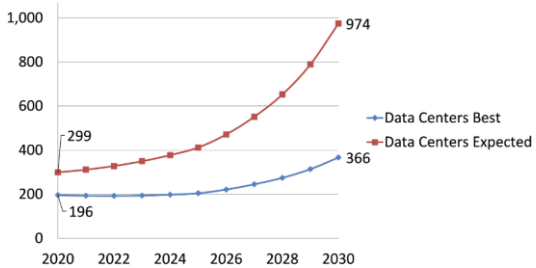
Market size USD 4339.82 Billion by 2030 CAGR ~23.6%

Source: Data Bridge Market Research



Demand for DC increasing CAGR ~14.2%

Source: Report Crux Market Research



DC power consumption is rising sharply

Source: Engineering & Applied Science Journal

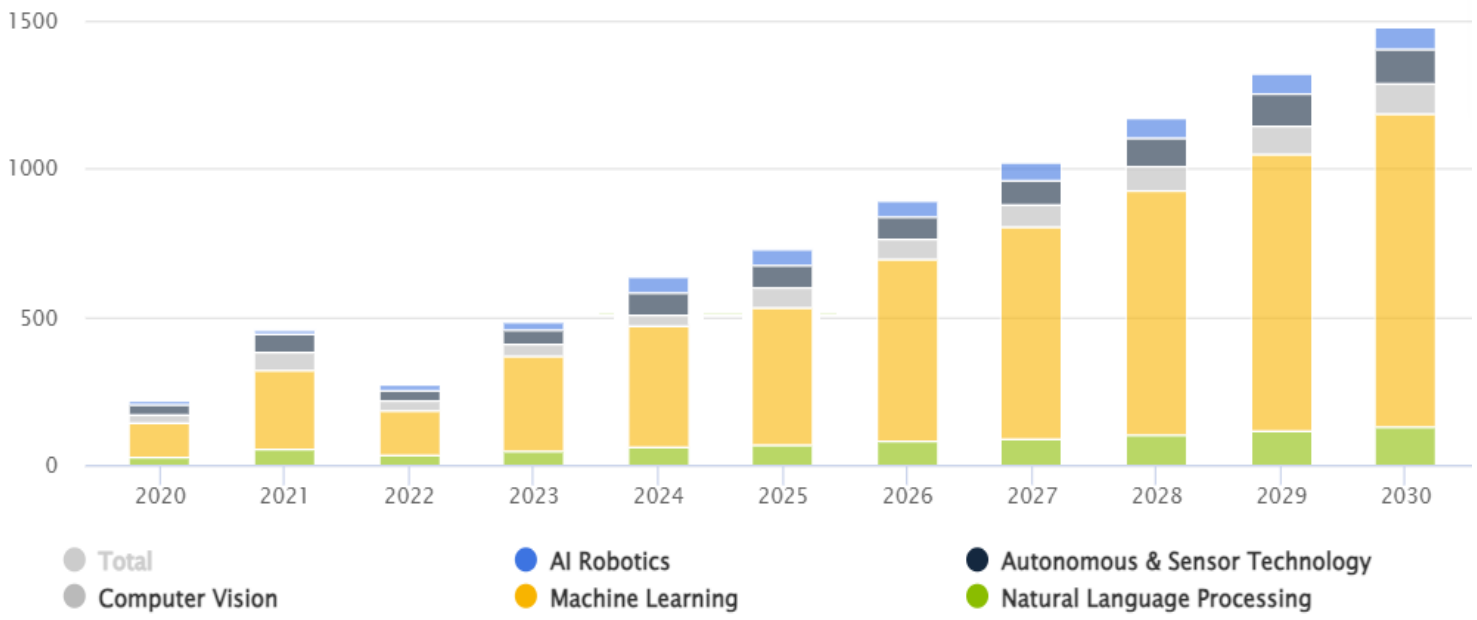
DCs account for world's highest CO2 emissions (~3.7%)

Source: International Energy Agency

Fujitsu's vision is to develop an energy-efficient AI software stack to enable maximised Arm CPU performance and support green data centres for sustainable digital transformation

AI Workloads in Data Centers

In data center operations, 80% workloads are machine learning applications. Our focus is on AI software acceleration to contribute towards low-power green data centres with Arm-based HPC technologies



Source: Statista Research Report (DC Market Size Analysis)

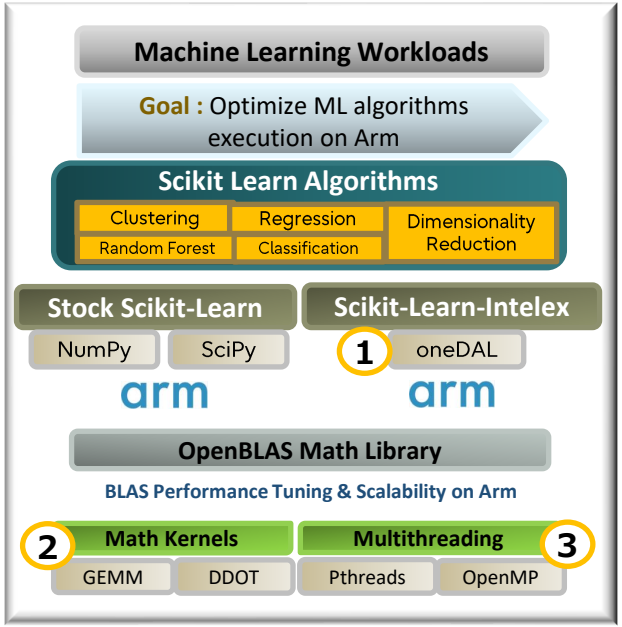
Linaro Connect
MADRID 2024 | MAY 14-17 2024

Fujitsu’s contribution for Scikit-Learn oneDAL Porting on Arm

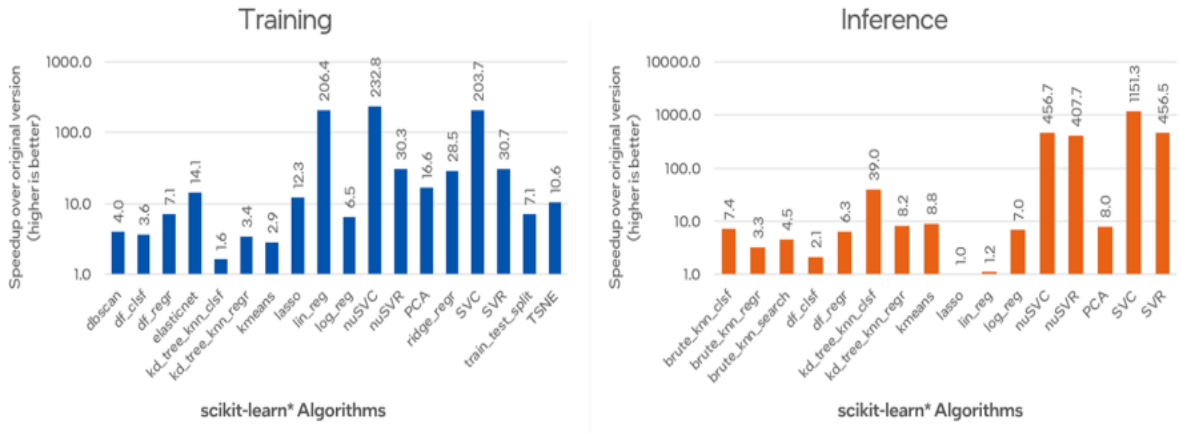
Machine Learning on Arm

Introduction to Scikit-Learn and oneDAL

Machine Learning Workload Optimization with Scikit-Learn Framework Enablement & Tuning

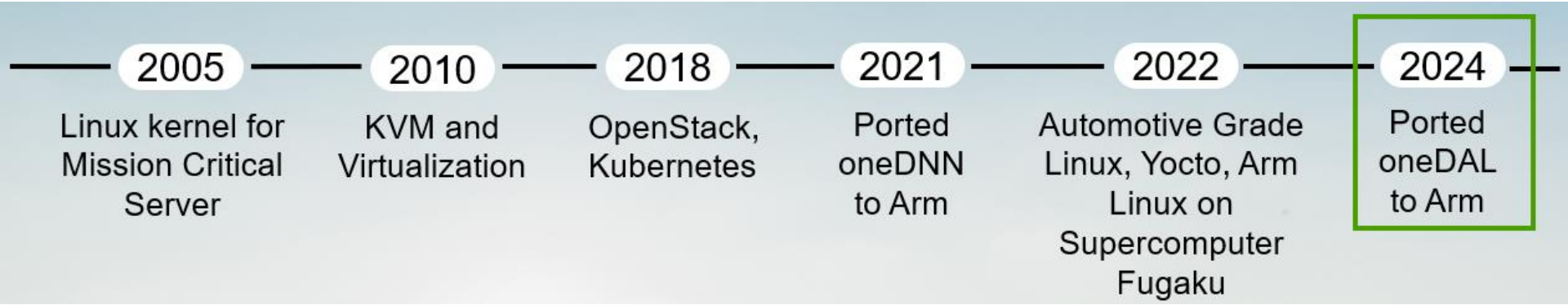


Intel® Extension for scikit-learn (scikit-learn-intelex) speeds up scikit-learn applications for x86 CPUs and GPUs across single- and multi-node configurations using oneDAL (Data Analytics Library)



Performance Benefits of Intel Extension for Scikit Learn (Combined Averages of FP32 & FP64 Workloads)
https://github.com/intel/scikit-learn-intelex

Fujitsu’s key contributions to OSS Community



Fujitsu ported oneDAL on Arm continuing long history of collaborating with open-source communities, via open-source development in mission-critical systems.

oneDAL PR (#2614) merged, raised by **Fujitsu**, to enable oneDAL multi architecture build with reference backend selection on Arm and optimized Scikit-Learn Algorithms

Partnership with Unified Accelerator (UXL) Foundation

- Build a multi-architecture multi-vendor software ecosystem for all accelerators
- Unify the heterogeneous compute ecosystem around open standards
- Build on and expand open-source projects for accelerated computing

Steering Committee Members

arm | FUJITSU | Google Cloud

Imagination | intel | Qualcomm

SAMSUNG | vmware by Broadcom

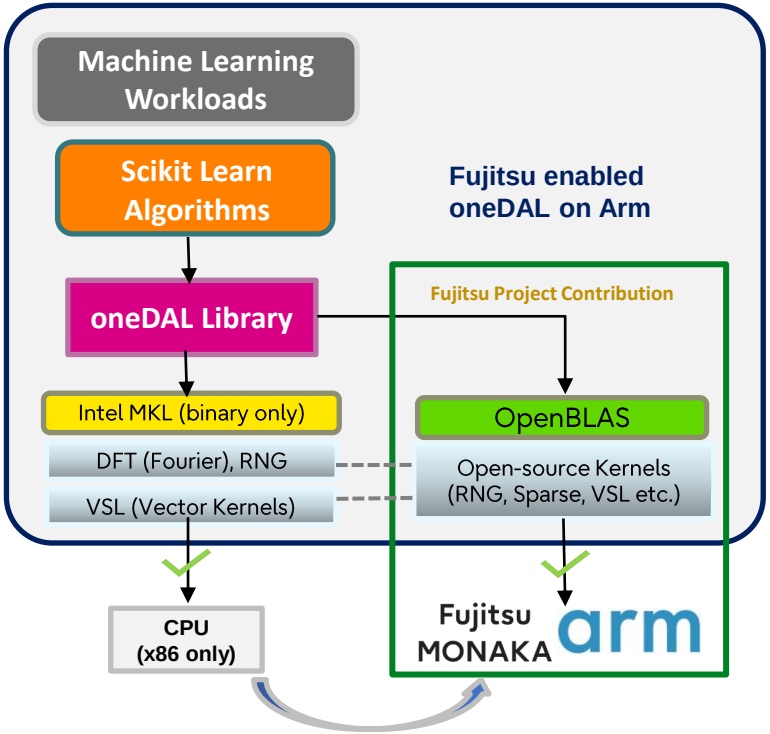
oneDAL Arm Porting Design

Scikit-Learn-Intelex
1 oneDAL arm

Historically, Intel's oneAPI Data Analytics Library (oneDAL) could only be compiled on x86 architecture due to Intel's Math Kernel Library (MKL) binary-only backend.

To accelerate ML workloads on Arm, Fujitsu replaced MKL calls with open-source function calls, and this resulted in oneDAL enablement on Arm.

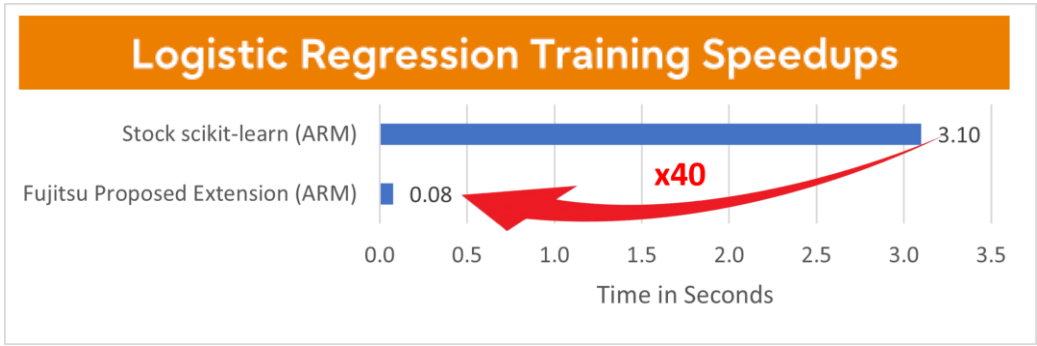
It is one of the first open-source contributions to UXL Foundation.



Performance Results

With SVE optimisation and oneDAL porting enhancements on ARM, our work showcases notable performance gains across multiple ML algorithms.

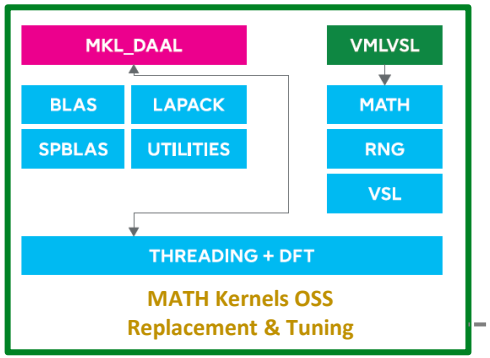
These graphs illustrate the training speedup of top two ML algorithms used by Fujitsu AutoML, which got a significant speedup of x31 in Random Forest and x40 in Logistic Regression.

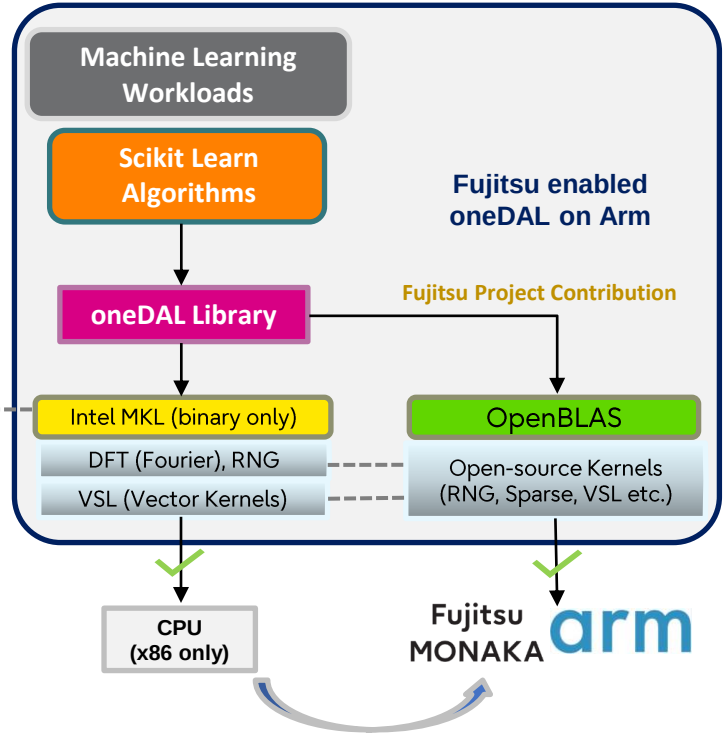


Results computed on AWS Graviton3 Arm-based CPU c7g.8xlarge 32-cores

Porting Methodology with Arm SVE

- oneDAL on x86 uses MKLFPK, with functionalities
- To support these functions on Arm, open-source optimised compute kernels from OpenBLAS are used as alternatives to leverage SVE on the Arm.





- Used reference backend & added makefile compiler options
- Added compiler macros throughout the code base to isolate x86 specific code chunks and handle it with Arm.

OpenBLAS GEMM 1-core Tuning

OpenBLAS GEMM Tuning PR (#4381) is merged. Updated SGEMM and DGEMM PQ param as per config of NEOVERSEV1 cache size

Update GEMM param for NEOVERSEV1 #4381

Code

Merged martin-frbg merged 1 commit into OpenMathLib:develop from darshanp4:issue_4323 on Dec 19, 2023

Conversation 3 Commits 1 Checks 63 Files changed 1 +4 −4

Performance for SGEMM improved by ~2-5% and DGEMM improved by ~2-12%

Source Code

param.h

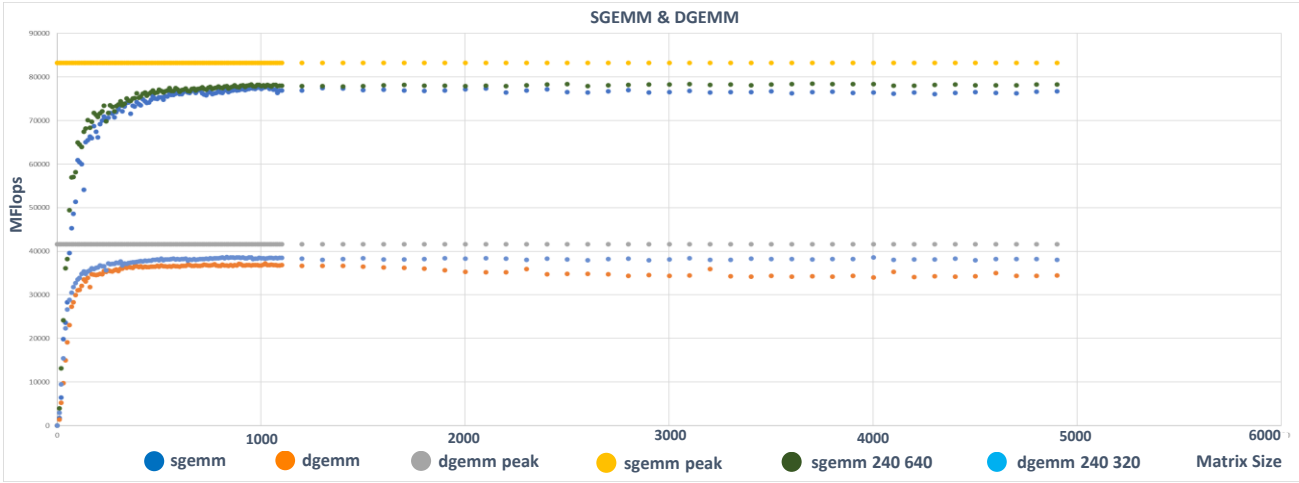
```
@@ -3396,13 +3396,13 @@ is a big de
#define ZGEMM_DEFAULT_UNROLL_N  4
#define ZGEMM_DEFAULT_UNROLL_MN 16

- #define SGEMM_DEFAULT_P 128
- #define DGEMM_DEFAULT_P 160
+ #define SGEMM_DEFAULT_P 240
+ #define DGEMM_DEFAULT_P 240
#define CGEMM_DEFAULT_P 128
#define ZGEMM_DEFAULT_P 128

- #define SGEMM_DEFAULT_Q 352
- #define DGEMM_DEFAULT_Q 128
+ #define SGEMM_DEFAULT_Q 640
+ #define DGEMM_DEFAULT_Q 320
#define CGEMM_DEFAULT_Q 224
#define ZGEMM_DEFAULT_Q 112
```

Performance Result

OpenBLAS/gemm.c benchmark on AWS Graviton3 single core



OpenBLAS GEMM 1-core Tuning

OpenBLAS GEMM Tuning PR (#4629) is merged. GEMM_PREFERED_SIZE is adjusted to 4 for double precision and 8 for single precision for Neoverse V1

GEMM_PREFERED_SIZE is modified small enough not to cause load imbalance and large enough to improve kernel efficiency

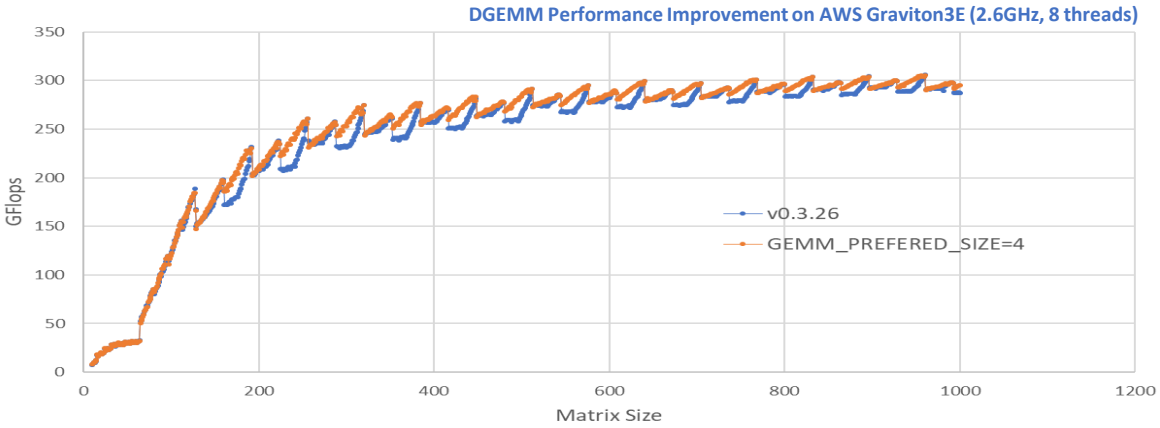
Source Code

param.h

```
@@ -3547,8 +3547,10 @@ is a big desktop

#if defined(XDOUBLE) || defined(DOUBLE)
#define SWITCH_RATIO            8
+#define GEMM_PREFERED_SIZE      4
#else
#define SWITCH_RATIO            16
+#define GEMM_PREFERED_SIZE      8
#endif
```

Performance Result



OpenBLAS GEMM multi-core Tuning

OpenBLAS GEMM multi-core tuning PR (#4655) Enhanced 2D thread distribution for improved parallel performance, noticeable with large matrices.

Expanding the scope of 2D thread distribution to improve multi-threaded DGEMM performance #4655

Code

Merged martin-frbg merged 1 commit into OpenMathLib:develop from yamazakimitsufumi:update_2d_thread_distribution 17 hours ago

Conversation 2 Commits 1 Checks 69 Files changed 1 +10 −0

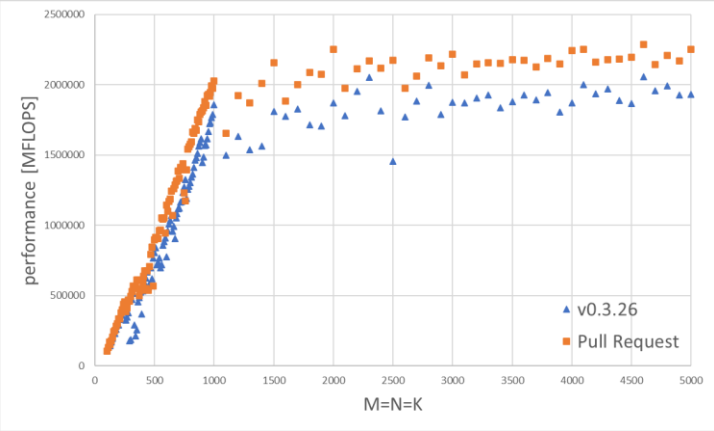
Performance improved by about 10% on Graviton3E (64 cores) & more than 20% on Xeon Platinum 8375C (32 cores x 2 sockets)

➢ Source Code

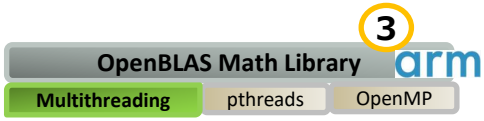
driver/level3/level3_thread.c

```
@@ -826,6 +826,16 @@ int CNAME(blas_arg_t *args, BLASLONG *range_m, BLASLONG *range_n, IFLOAT *sa, IF
     if (nthreads_m * nthreads_n > args -> nthreads) {
       nthreads_n = blas_quickdivide(args -> nthreads, nthreads_m);
     }
+    /* The nthreads_m and nthreads_n are adjusted so that the submatrix       */
+    /* to be handled by each thread preferably becomes a square matrix        */
+    /* by minimizing an objective function 'n * nthreads_m + m * nthreads_n'. */
+    /* Objective function come from sum of partitions in m and n.             */
+    /* (n / nthreads_n) + (m / nthreads_m)                                    */
+    /* = (n * nthreads_m + m * nthreads_n) / (nthreads_n * nthreads_m)        */
+    while (nthreads_m % 2 == 0 && n * nthreads_m + m * nthreads_n > n * (nthreads_m / 2) + m * (nthreads_n * 2)) {
+      nthreads_m /= 2;
+      nthreads_n *= 2;
+    }
```

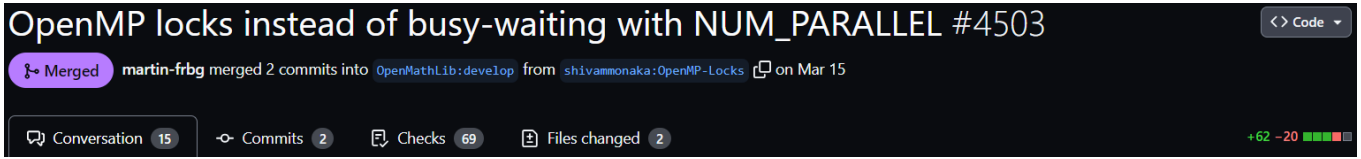
➢ Performance Result



OpenBLAS Threading Improvements

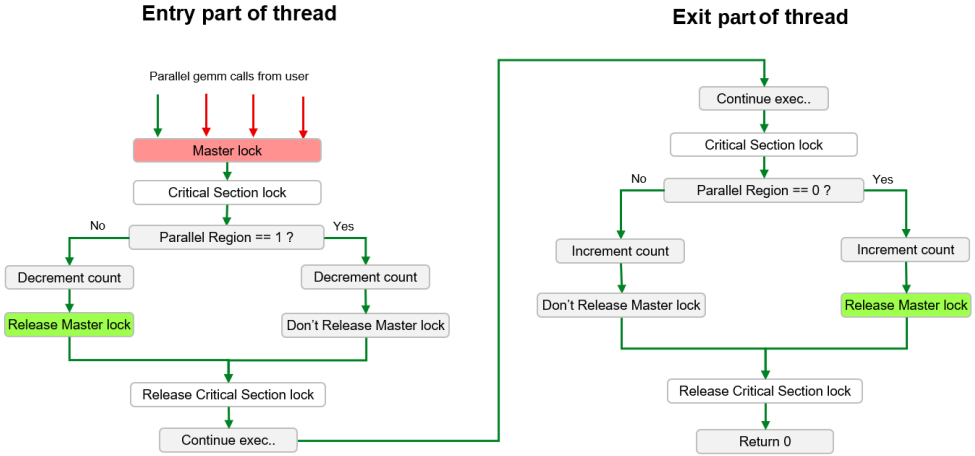


OpenBLAS OpenMP Tuning PR (#4503) is merged. Implemented OpenMP locking mechanism for refined parallel execution

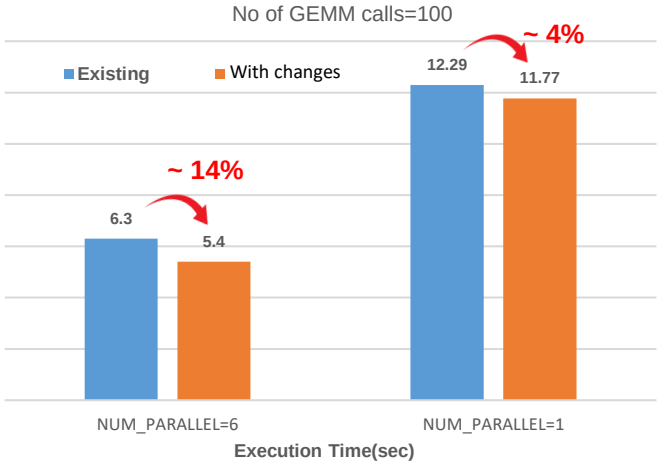


Locking design as per OpenMP specifications, consistent with Pthreads, Win32. Performance gain up to ~15% to default.

➢ Solution Design

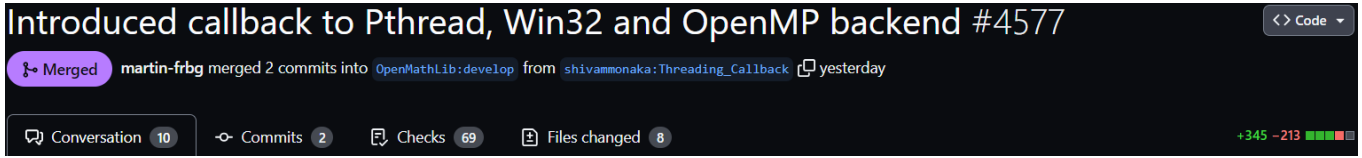


➢ Performance Result



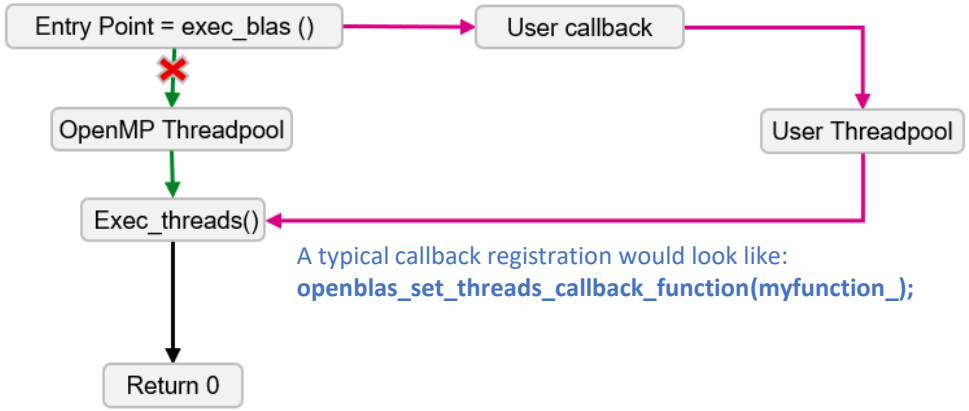
OpenBLAS Thread Composability

OpenBLAS Thread Composability PR (#4577) is merged. Enables OpenBLAS threading to be composable with caller multithreading with performance gain



Achieves thread composability in OpenBLAS with all threading backends e.g., OpenMP, Pthreads and win32

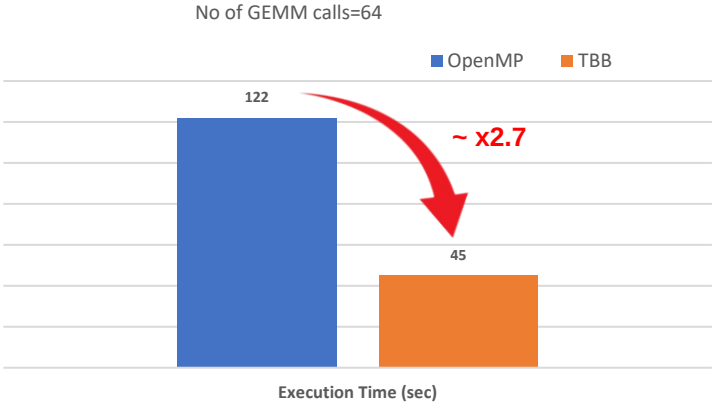
➢ Solution Design



A typical callback registration would look like:
openblas_set_threads_callback_function(myfunction_);

➢ Performance Result

Upto 2.7x speedup with TBB v/s OpenMP in nested parallelism scenario



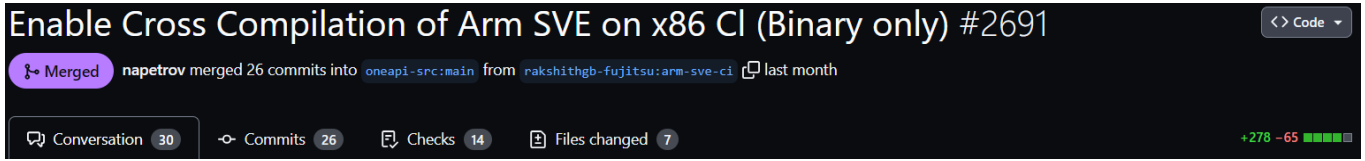
oneDAL Multi Architecture Enablement

➢ Contributions

Scikit-Learn-Intelex PR (#1771) is merged. This change successfully enables scikit-learn-intelex on Arm CPUs with BLAS reference backend.

oneDAL Makefile PR (#2672) is merged. Streamlined structure for enhanced platform support and maintainability for multi-arch builds.

oneDAL CI Support PR (#2691) Added Arm cross-compilation support for x86 CI system, simplified future additions of cross-compilers and architectures.



➢ Future Work

Block Size Optimization for Arm	Bazel Build Support for Arm	Algorithm and Kernel Tuning
Dynamic template dispatcher to identify architecture/ISA specific optimal block size	New architectures to support Bazel build system, starting with Arm	Further tuning of Machine Learning algorithms and computationally intensive Math Kernels

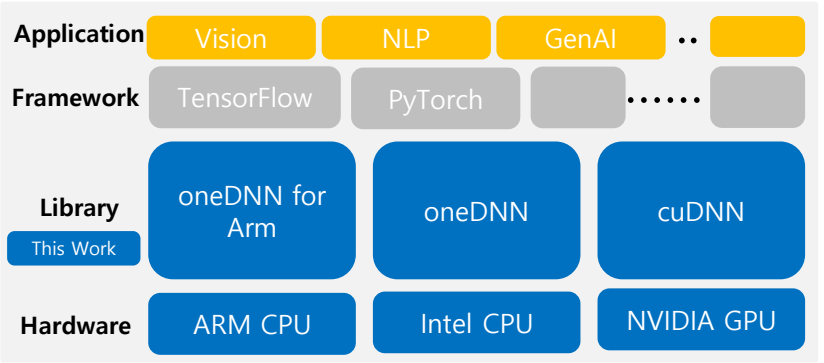
Linaro Connect
MADRID 2024 | MAY 14-17 2024

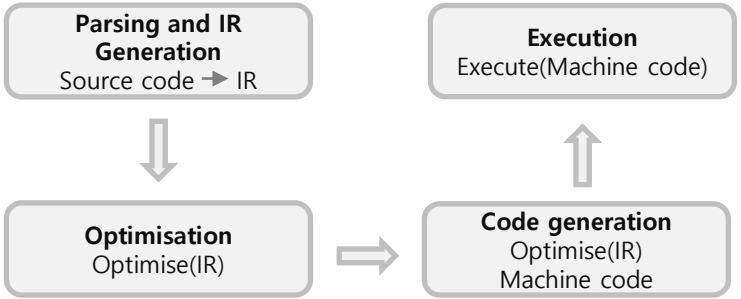
Fujitsu’s recent advancements in oneDNN and JIT Kernel on Arm

Deep Learning on Arm

Introduction – oneDNN and JIT

- To accelerate deep learning (DL) processes on the supercomputer Arm CPU's, oneDNN was ported and optimized by Fujitsu, Kawakami – San & this work was presented in Linaro Connect 2021.
- oneDNN is an open-source DL processing library developed by Intel for cross-platform architecture. oneDNN dynamically creates the execution code for the computation kernels, which are implemented at the granularity of arch. instructions using Xbyak, the Just-In-Time (JIT) assembler.
- Just-In-Time (JIT) compilation is a technique used in compiler design where the compiler translates source code into machine code at runtime, rather than ahead of time (AOT) as in traditional compilation.
- Major Advantages of JIT:





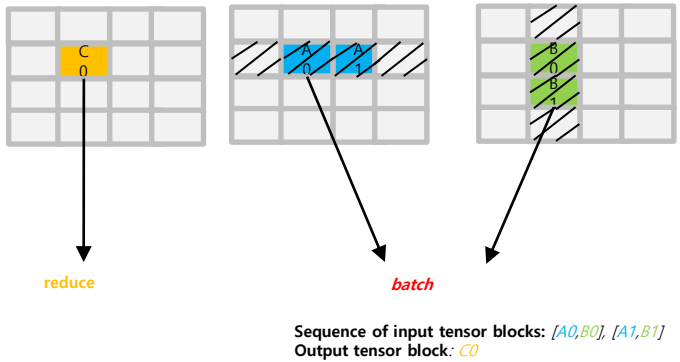
BRGEMM Contribution and Results

BRGEMM MatMul Enablement PR (#1818) is merged, expands Arm SVE support for matrix mul. & adds BRGEMM folder for aarch64

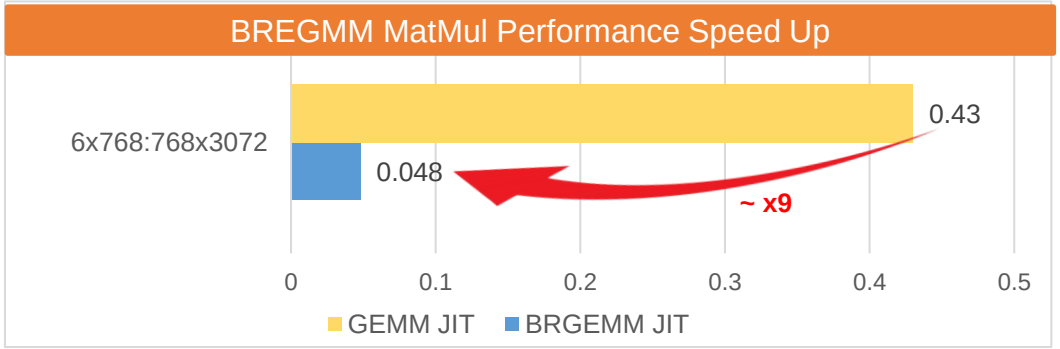
cpu: aarch64: Expand ARM SVE support for matrix multiplication #1818

Merged igorsafo merged 12 commits into oneapi-src:main from vineelabhinav:feature-sve-matmul 2 weeks ago

~9x performance gain observed as compared to oneDNN GEMM JIT Implementation



Sequence of input tensor blocks: [A0,B0], [A1,B1]
Output tensor block: C0



- **What is BRGEMM ? (Batch Reduced General Matrix Multiplication)**
  - Tensors are reduced into batches for multiplications
  - Broadcast the input matrix B values
  - Perform fused-multiply-add instruction (fma instruction) at once for multiple values

SoftMax with results

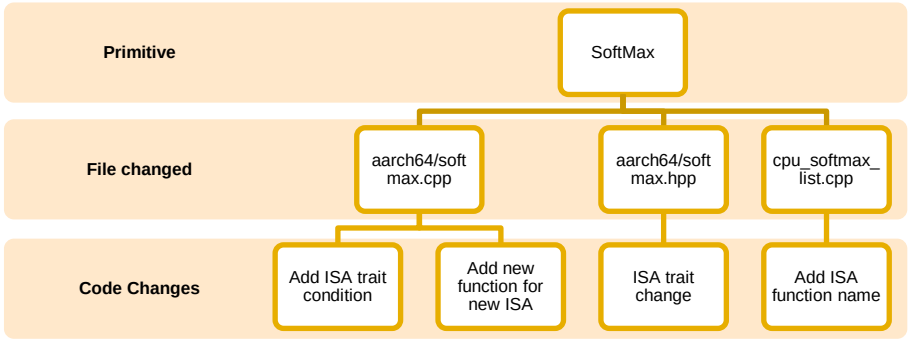
oneDNN Pooling JIT Kernel PR (#1786) is merged, expands support of SoftMax for multiple ISA

cpu: aarch64: Expand ARM SVE support in jit_uni_softmax #1786

Merged dzarukin merged 1 commit into `oneapi-src:main` from `deepeshfujitsu:aarch64-sve-jit-softmax` on Feb 6

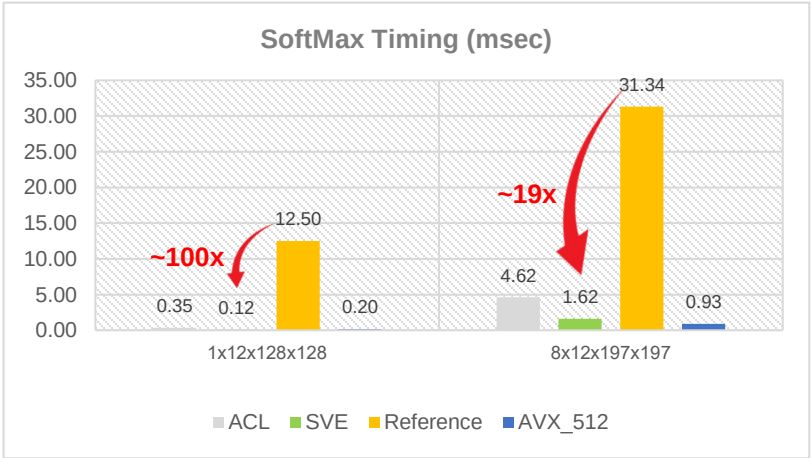
~100x performance gain observed as compared to oneDNN Reference implementation

Code Changes



Conditional Statement Modification : Added OR condition to use multiple ISA
Added new function for supporting SVE in different vector length.

Performance Result



Pooling with results

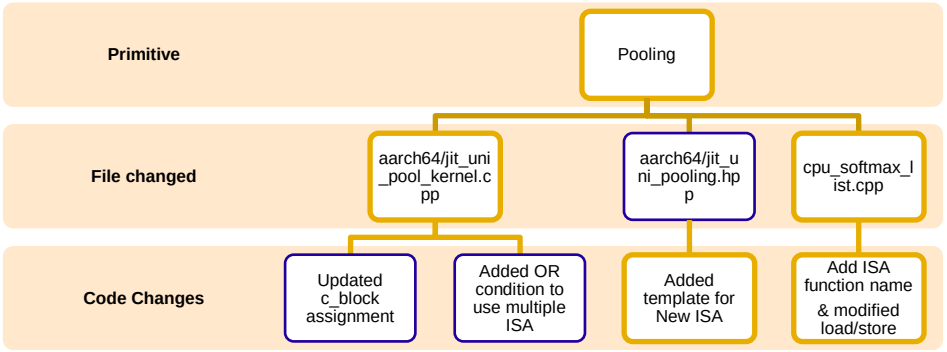
oneDNN Pooling JIT Kernel PR (#1850) is merged, expands support of Pooling for multiple ISA

cpu: aarch64: Expand ARM SVE support in jit_uni_pool_kernel #1850

Merged igorsafo merged 2 commits into oneapi-src:main from vishwascm:aarch64-sve-jit-pooling 2 weeks ago

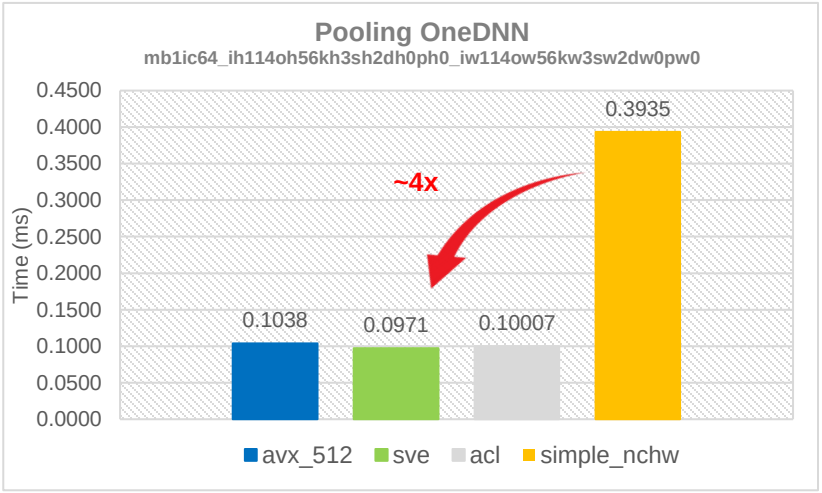
~4x performance gain observed as compared to current implementation in oneDNN

Code Changes



Conditional Statement Modification : Added OR condition to use multiple ISA. Modified load and store instructions to use predicate registers for correct ISA matching.

Performance Result



Concluding Remarks

Fujitsu Arm Processor "FUJITSU-MONAKA"

- Creating a new era of computing power is mandatory for the future society with massive data generation and processing
- Ever-increasing power in datacenters is critical, and the power efficiency in CPU (consists of 60%) would be the vital factor for a sustainable future
- Fujitsu shall utilize its Supercomputer success and technology for the solution

FUJITSU-MONAKA

- **Developing the new power efficient CPU "FUJITSU-MONAKA" for datacenters, which will be shipped in 2027**
- **Targeted for wide range of usage in the datacenter including AI and HPC, and contribute to the realization of carbon-neutral society**

Software Ecosystem for AI & HPC Computing

Focus on Application Performance

Focus on Ease-of-Use & Open-Source Software

Fujitsu microarchitecture | 3D many-core architecture | Confidential Computing

High-performance
- Cloud native 3D many-core design by Fujitsu-proven microarchitecture
- High memory bandwidths

Energy Efficient
- Leading-edge process technology
- Ultra low voltage operation

High Reliability
- Multiple VM Confidential Computing
- Mainframe class RAS for stable operation

Easy to Use
- Open & de-facto standard software stacks
- Fujitsu compiler technology
- Air-Cooling for easy deployment

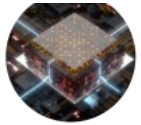
Fujitsu

UXL Unified Acceleration Foundation | arm | Linaro

Feature Development

Quality Enhancement

Technical Support



Open-Source Community

Linux Kernel | QEMU | LLVM | Python | oneDNN | Kubernetes | OpenStack | oneDAL

Software Distributors
Linux OS/System SW

Customers

Conclusion and Future Work

OpenMathLib/
OpenBLAS
oneDAL
arm

Contribution

- **Fujitsu successfully contributes to ML Acceleration** enabling oneDAL on ARM, showcasing significant AI-ML algorithm speedups. Focussing on BLAS and SVE optimization.

FUJITSU
oneDNN
arm

Contribution

- **Fujitsu continues contribution to DL Acceleration** BRGEMM based Convolution, Expansion of JIT kernels to Batch Normalization & Softmax with BF16 data type upcoming.

FUJITSU
MONAKA

Fujitsu Vision

- **FUJITSU-MONAKA aligns software acceleration** commitment with green data center goals and aims to democratize AI for sustainable digital transformation.

Collaboration

- **Fujitsu welcomes feedback & contributions from Linaro**, Fujitsu looks forward to actively collaborate with OSS community for accelerated Arm computing ecosystem.

Resources

FUJITSU-MONAKA Reference Links

- FUJITSU-MONAKA Next Arm Processor
- Democratizing the use of AI: FUJITSU - MONAKA

Scan the QR code to know more

oneDAL Pull Request Contributions

- Enable ARM(SVE) CPU support with reference backend #2614
- Enable build on ARM(SVE) #1771
- Makefile refactoring to factor out common build code #2672
- Enable Cross Compilation of Arm SVE on x86 Cl (Binary only) #2691

OpenBLAS Pull Request Contributions

- Update GEMM param for NEOVERSEV1 #4381
- Set GEMM_PREFERED_SIZE parameter for Neoverse V1 #4629
- Expanding the scope of 2D thread distribution to improve multi-threaded DGEMM performance #4655
- OpenMP locks instead of busy-waiting with NUM_PARALLEL #4503
- Introduced callback to Pthread, Win32 and OpenMP backend #4577

oneDNN Pull Request Contributions

- cpu: aarch64: Expand ARM SVE support for matrix multiplication #1818
- cpu: aarch64: Expand ARM SVE support in jit_uni_softmax #1786
- cpu: aarch64: Expand ARM SVE support in jit_uni_pool_kernel #1850

Acknowledgement

- Kentaro KAWAKAMI, Kouji KURIHARA, Takumi HONDA : Binary Translator to Accelerate Development of Deep Learning Processing Library for AArch64 CPU
- Mitsufumi YAMAZAKI : Update GEMM param for NEOVERSEV1 #4381
- Tetsuzou USUI : Set GEMM_PREFERED_SIZE parameter for Neoverse V1 #4629
- Mitsufumi YAMAZAKI : Expanding the scope of 2D thread distribution to improve multi-threaded DGEMM performance #4655
- Initial input for backend selection #2396

Acknowledgement

NEDO Project | "Technology Development of the Next Generation Green Data Center" for the "Green Innovation Fund Project/Construction of Next Generation Digital Infrastructure"

- NEDO is "New Energy and Industrial Technology Development Organization", a national research and development agency in Japan.
- Fujitsu has been selected for the national initiative along with NEC Corporation, AIOCORE Co., Ltd., KIOXIA Corporation, FUJITSU Optical Components Limited and KYOCERA Corporation.
- This presentation is based on results obtained from a project, JPNP21029 subsidized by the New Energy and Industrial Technology Development Organization (NEDO).


LinaroConnect
MADRID 2024 | MAY 12-17 2024
Thank you