Teaching Accelerated Computing and Deep Learning at a Large-Scale with the NVIDIA Deep Learning Institute

Bálint Gyires-Tóth
Budapest University of Technology and Economics
Budapest, Hungary
toth.b@tmit.bme.hu

Işıl Öz
Izmir Institute of Technology
Izmir, Turkey
isiloz@iyte.edu.tr

Joe Bungo
Deep Learning Institute, NVIDIA Corporation
Austin, Texas
jbungo@nvidia.com

ABSTRACT

Researchers and developers in a variety of fields have benefited from the massively parallel processing paradigm. Numerous tasks are facilitated by the use of accelerated computing, such as graphics, simulations, visualisations, cryptography, data science, and machine learning. Over the past years, machine learning and in particular deep learning have received much attention. The development of such solutions requires a different level of expertise and insight than that required for traditional software engineering. Therefore, there is a need for novel approaches to teaching people about these topics. This paper outlines the primary challenges of accelerated computing and deep learning education, discusses the methodology and content of the NVIDIA Deep Learning Institute, presents the results of a quantitative survey conducted after full-day workshops, and demonstrates a sample adoption of DLI teaching kits for teaching heterogeneous parallel computing.

KEYWORDS

Accelerated Computing, Deep Learning, Artificial Intelligence, NVIDIA Deep Learning Institute

1 INTRODUCTION

Research and development have been transformed by the advancement of accelerated computing (AC). At present, the computational power of a single workstation is comparable to the power of a supercomputer of the past. Furthermore, the top supercomputer of today has broken the exascale barrier [23]. Due to the growing amount of data available, the significant enhancements in accelerated computing, and novel scientific results, deep learning (DL) [9] has become the most powerful tool for modeling real-world processes based on observations. In a neural network, the trainable parameters realized as a computational graph, are capable of learning various high- and low-level abstractions of the process being modeled, which is also referred to as feature learning. The modeling is performed hand in hand with the feature learning part in order to align the 'best' features with the 'best' model. Deep neural networks are scaling well – if more data is available, than a larger model can usually achieve better accuracy [6]. A robust hardware and software architecture for deep learning is capable of supporting the computational requirements. Aside from the ability to model speech [20] and vision [25] functions, deep learning is among the basic techniques for natural language processing [3], predictive maintenance [21], and anomaly detection [17], just to name a few areas. Professionals who are skilled in developing accelerated computing and deep learning solutions are in great demand. In these fields typically Pi or comb-shaped skills [10] are needed. A good understanding of fundamentals, programming skills, and project experience are essential even for a junior, which slows down the learning curve [8] compared to traditional education in software engineering. Besides higher education (HE), reskill [4] and upskill offerings of tech giants (like NVIDIA, Google, Amazon Web Service, Microsoft, etc.) and of vocational education training (VET) providers are among the possible options. Our paper discusses the main challenges in accelerated computing and deep learning education, demonstrates the methodology that was implemented in two universities based on the NVIDIA Deep Learning Institute (DLI) materials, and presents and discuss the results of the delivered contents.

2 EDUCATION

2.1 Accelerated Computing Education

Accelerated computing enables speed-up in program executions by leveraging hardware resources [5]. While instruction-level parallelism implemented in earlier superscalar processors provides performance optimizations and often does not need specific code modifications, leveraging multiple cores in a parallel system requires significant programming effort. Understanding the massively parallel execution and resource utilization in heterogeneous platforms with many-core GPUs requires expertise in architecture-aware programming.

While it is possible to introduce accelerated computing concepts in high-level directive-based programming models like OpenACC or OpenMP [2], teaching fine-grained programming based on low-level programming models like CUDA [7] or Pthreads can be an option to enable more parallelism opportunities for performance improvements in target executions.

For teaching heterogeneous computing, there are efforts to introduce parallel programming in different stages of undergraduate and graduate university education [18, 19]. Besides formal university courses, Massive Open Online Courses (MOOC)-style platforms enable people to learn about diverse topics by maintaining online

https://doi.org/10.22369/issn.2153-4136/14/1/4

courses. This solution seems promising as MOOC serves high-quality content from various qualified instructors and provides cloud infrastructure with software and hardware setup.

2.2 Deep Learning Education

Teaching deep learning can be approached in a variety of ways. Among the most common methods are:

Bottom-up: Generally, fundamentals such as probability theory, algebra, data analysis, and machine learning are taught first. Based on these concepts, backpropagation, stochastic gradient decent (SGD) and its variants, regularization techniques and traditional and modern neural architectures are described. Programming tasks and deep learning applications follow the fundamental components. Due to the fact that learning the fundamentals takes a considerable amount of time, this approach is usually taught in HE institutions as BSc and MSc programs. A combination of MOOC courses can also follow this approach.

Top-down: In order to gain practical experience as early as possible, the education begins with high-level programming examples. Following the first impression and the experience of success, participants are instructed on the fundamentals in greater detail. Depending on the length of the educational program, the depth of fundamentals may vary. In shorter courses, in MOOC courses, as well as in multi-semester programs for higher education, top-down approaches can be effectively incorporated.

Application-oriented: It is similar to the top-down approach, however it is geared towards a specific application domain, such as speech, computer vision, natural language processing, predictive maintenance, etc. Furthermore, the fundamentals are briefly discussed, mostly. Essentially, the goal is to gain knowledge about how to use DL tools in order to solve some specific problems. Application-oriented deep learning education are usually done in one to few-days trainings, workshops and boot camps.

Project-based [22] and on-the-job training: This focuses on some specific problem, which is often related to a real-world project. This approach allows corporate employees to gain deep learning experiences while working on their primary duties. In this case, not only the modeling but the data collection, preparation, feature engineering, and evaluation might be included in the training. In order to conduct a project-based or on-the-job training, senior deep learning experts are needed as instructors, who understand the problem, identify potential pitfalls, assist the employees in finding a solution (in which the expert is also involved), and evaluate that solution appropriately. A bootcamp or consultation service can be implemented using this approach.

3 METHODOLOGY

In this paper, we describe how NVIDIA Deep Learning Institute offerings help people to dive into AC and DL, and we also discuss, how these contents can be integrated into the academia. NVIDIA is a hardware and software platform company focusing on graphics processing units (GPUs) for the gaming and professional markets (including Artificial Intelligence), as well as system-on-a-chip units (SoCs) for the mobile computing and automotive market. Providing high quality software tools and educational materials is essential for NVIDIA in order to assist their customers. As for the former, it is provided by NVIDIA researchers and developers, while the latter is provided by NVIDIA Deep Learning Institute (DLI). NVIDIA DLI offers resources for diverse learning needs – from learning materials to self-paced and live training to educator programs—giving individuals, teams, organizations, educators, and students what they need to advance their knowledge in AI, accelerated computing, accelerated data science, graphics and simulation, and more. NVIDIA DLI has various offerings, as follows.

3.1 Self-Paced Courses

DLI offers online self-paced courses, where interested individuals follow the online materials from NVIDIA infrastructure on their own and receive certificates upon successful completion. Through accessing content on the latest technology trends prepared by experienced instructors and domain experts, and gaining hands-on experience with GPU-accelerated servers in the cloud, they learn to build deep learning, accelerated computing, and data science applications for a variety of industries. DLI offers self-paced courses in Deep Learning, Accelerated Computing Fundamentals, Data Science, Graphics and Simulation, Infrastructure, and Networking. The courses are in different lengths, from one- to eight-hours. Due to the various lengths, these courses are flexible to be integrated into university classes. For instance, after introducing the theory of Long Short-Term Memory (LSTM) in a bottom-up approach, including a DLI self-paced course on 'Modeling Time Series Data with Recurrent Neural Networks in Keras' [15] as a 2-hour-long practice helps students to have a hands-on experience with a real-world dataset. As the hardware and software infrastructure are already available, it is a great benefit to educators as well.

3.2 Instructor-led Workshops

For developers, data scientists, and engineers, live instructor-led workshops are taught by DLI-certified instructors with deep learning or accelerated computing expertise. The workshops may take place virtually or in-person with both models leveraging NVIDIA's online compute resources. Course materials include hands-on experience in a variety of concepts and levels. While some basic courses are instructor-led versions of the self-paced courses, there are many other advanced and domain-focused courses. By having a specific content, instructor-led workshops can be categorized as 'application-oriented' (see Section 2.2 for details). In addition to the actual applications, a broad theoretical overview is often presented as well, so the attendees can decide where to further their knowledge. DLI's instructor-led workshops cover five major areas:

Deep Learning Fundamentals teach how to use deep learning for computer vision, transformer-based natural language processing (NLP), conversational AI applications, recommendation systems, and multi-GPU setups.

Deep Learning by Industry describes how deep learning and AI can be applied to various industry domains such as industrial inspection, intelligent video analytics, anomaly detection, and predictive maintenance.

Accelerated Computing focuses on programming CUDA with C/C++ and Python on single and multiple nodes, as well as how to accelerate applications with OpenACC.

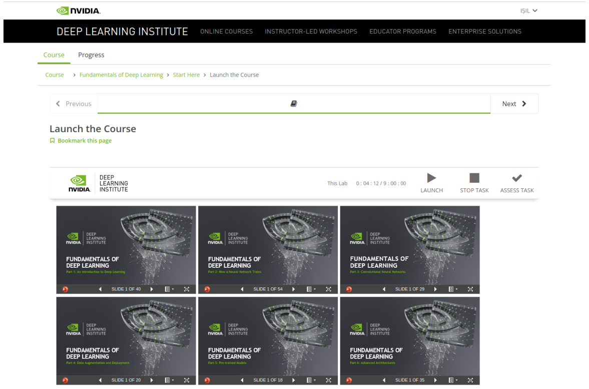

NVIDIA
IŞIL
DEEP LEARNING INSTITUTE
ONLINE COURSES
INSTRUCTOR-LED WORKSHOPS
EDUCATOR PROGRAMS
ENTERPRISE SOLUTIONS
Course
Progress
Course › Fundamentals of Deep Learning › Start Here › Launch the Course
Previous
Next
Launch the Course
Bookmark this page
NVIDIA DEEP LEARNING INSTITUTE
This Lab 0 : 04 : 12 / 9 : 00 : 00
LAUNCH
STOP TASK
ASSESS TASK
FUNDAMENTALS OF DEEP LEARNING
Part 1: An Introduction to Deep Learning
SLIDE 1 OF 40
FUNDAMENTALS OF DEEP LEARNING
Part 2: How a Neural Network Trains
SLIDE 1 OF 54
FUNDAMENTALS OF DEEP LEARNING
Part 3: Convolutional Neural Networks
SLIDE 1 OF 29
FUNDAMENTALS OF DEEP LEARNING
Part 4: Data Augmentation and Deployment
SLIDE 1 OF 20
FUNDAMENTALS OF DEEP LEARNING
Part 5: Pre-trained Models
SLIDE 1 OF 18
FUNDAMENTALS OF DEEP LEARNING
Part 6: Advanced Architectures
SLIDE 1 OF 35

Figure 1: DLI workshop main page with slides and link to the cloud (via Launch Task).

Accelerated Data Science covers data science techniques accelerated with GPUs using Rapids.AI, and libraries such as cuDF, cuML, cuGraph, and more.

Networking introduces important concepts in building multi-GPU and multi-node systems.

Instructor-led workshops are offered by Deep Learning Institute for both individuals and teams from academia and industry. While public workshops are available for everyone, DLI University Ambassadors deliver free workshops for students and lecturers by utilizing hands-on course materials and GPU-accelerated workstations in the cloud. It is possible either to request a workshop from NVIDIA or to attend a scheduled workshop by registering for the course. Once registered for the offered workshop, an event code is sent to the participant via e-mail, and s/he can join the course from https://courses.nvidia.com/dashboard by creating an account in the system. After logging into the system, as seen in Figure 1, the participant can reach presentation slides, which the instructor explains during the workshop. Additionally, cloud-based GPU resources are available via Jupyter Notebook and JupyterLab interfaces. The participant can view both brief explanations and small examples, where he can execute code segments and modify them to get hands-on experience. In the meantime, he can access the workstation via the terminal to compile, execute, and modify the source files provided as part of the workshop. As the final part of the workshop, an assessment is given to demonstrate the information gained from the workshop and receive a certificate if the participant successfully completes the assessment. A typical assessment includes a hands-on programming goal, combining the main concepts taught in the workshop and testing the skills learned in the course. Moreover, some courses include only multiple-choice questions and require a minimum number of correct answers from the participant. While it is possible to attempt the assessment just at the end of the workshop, the participant can postpone the assessment evaluation and certification process. After completion of the workshop, the participants are asked to provide feedback about the workshop to evaluate both the content and the instructor.

The feedback form asks the following questions:

- How likely is it that you would recommend this course to a friend or colleague? (0..10)
- How would you rate these aspects of your learning experience? (1..5 and N/A)
  - Overall experience
  - Registration and login
  - Navigating the course
  - Launching hands-on content
- Did the course material meet your expectations? (1..5 and N/A)
  - Hands-on exercises were helpful in my learning objectives
  - Level of difficulty was as expected
  - Quality of content was as expected
  - The content of the course was interactive
  - Prerequisite information was useful
- How would you rate these aspects of your instructor-led session? (1..5 and N/A)
  - Instructor presentation skills
  - Instructor knowledge
  - TA knowledge
  - Pacing of course
  - Pre-event communication
- Anything else you'd like to tell us? (open ended question)

Teaching assistants (TAs) are involved depending on the number of participants. There should be one teaching assistant per 20 attendees as a general guideline. TAs are mainly helping in the chat. In case of a complex question, the TA will take the attendee into a breakout room for direct assistance. In this paper, we investigate the feedbacks of the following DLI workshops organized in Hungary by NVIDIA DLI and the Budapest University of Technology and Economics:

- Fundamentals of Deep Learning (FDL) [14]
- Building Transformer-Based Natural Language Processing Applications (NLP) [12]
- Building Conversational AI Applications (CAI) [11]

There were three different target groups (even within a group, the participants varied between two workshops):

- BSc group: These workshops were delivered as a part of a beginner level deep learning class (4 ECTS) at a university for BSc students.
- MSc group: The students were attending to a Human-Computer Interaction class (5 ECTS) at a university in their MSc studies.
- Mixed group: including BSc, MSc and PhD students, educators and non-profit researchers.

Participation in the workshop and passing the assessment were required for the BSc group to complete their course at the university. For the MSc group, passing the assessment was among the tasks to be exempted from the exam. Participants from mixed groups were invited to attend workshops (although it was not mandatory), and they were encouraged to pass the assessment to earn the certificate so they can add it to their CV. Participation in all workshops was free of charge, but only non-profit research institute and university staff and students were permitted to attend.

3.3 Teaching Kits

In order to assist educators in incorporating deep learning and accelerated computing into university courses, DLI offers downloadable

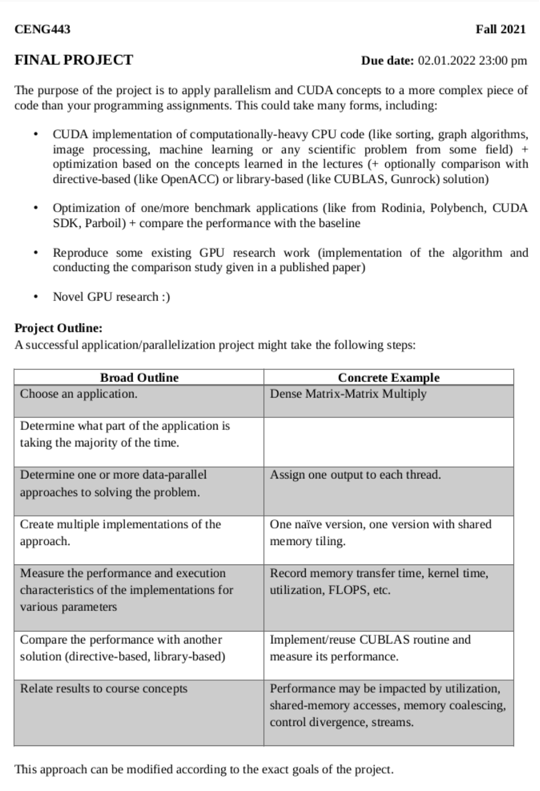
Table 1: Weekly Course Topics and Accelerated Computing Teaching Kit Modules.

Course Topic	Teaching Kit Module
Parallelism	Module 17 - Computational Thinking For Parallel Programming
Introduction to CUDA	Module 2 - Introduction to CUDA C
CUDA Threads	Module 3 - CUDA Parallelism Model
CUDA Memory	Module 4 - Memory and Data Locality
Tiling	Module 4 - Memory and Data Locality
Convolution	Module 8 - Parallel Computation Patterns (Stencil)
Parallel Patterns	Module 9 - Parallel Computation Patterns (Reduction) + Module 10 - Parallel Computation Patterns (Scan)
CUDA Performance	Module 6 - Memory Access Performance
Dynamic Parallelism	Module 23 - Dynamic Parallelism
CUDA Libraries	Module 25 - Using CUDA Libraries
CUDA CNN	–

teaching kits that include course materials that were co-developed with different university faculties. Each kit, freely available for the instructors world-wide, includes lecture slides and hand-on lab exercises with sample solutions. Additionally, the Teaching Kits Program provides free access for instructors and students to GPU-accelerated workstations in the cloud, either through Amazon's AWS program offering credits or online self-paced DLI courses. (mentioned in Section 3.1). The students can access GPU resources for hands-on exercises or larger-scale projects, and earn certificates that demonstrate their expertise in the subjects.

In the computer engineering department at Izmir Institute of Technology in Turkey, the Heterogeneous Parallel Programming course has been offered based on the Accelerated Computing teaching kit. The semester-long technical elective course covers GPU hardware, CUDA basics, advanced CUDA features, and parallel application development topics. While the content is updated each year, the main concepts and the corresponding teaching kit modules are presented in Table 1.

While the slides from the teaching kit are utilized in the specific modules, lab exercises and quiz questions are not used since there is no lab session or quiz in the course. Instead, self-developed programming assignments and midterm/final questions are designed for the course assessment and evaluation. Additionally, a final term project is assigned to the students, where *Project Guidelines* document of the Teaching Kit is utilized for defining the purpose, outline, and grading rubric of the project (The definition document at 2020-2021 term is given in Figure 2). The students are expected to propose and implement a complete CUDA application, conduct an experimental study, and perform a comparative analysis by comparing different CUDA implementations with other programming models, like OpenACC or other libraries.



CENG443 **Fall 2021**

FINAL PROJECT **Due date:** 02.01.2022 23:00 pm

The purpose of the project is to apply parallelism and CUDA concepts to a more complex piece of code than your programming assignments. This could take many forms, including:

- CUDA implementation of computationally-heavy CPU code (like sorting, graph algorithms, image processing, machine learning or any scientific problem from some field) + optimization based on the concepts learned in the lectures (+ optionally comparison with directive-based (like OpenACC) or library-based (like CUBLAS, Gunrock) solution)
- Optimization of one/more benchmark applications (like from Rodinia, Polybench, CUDA SDK, Parboil) + compare the performance with the baseline
- Reproduce some existing GPU research work (implementation of the algorithm and conducting the comparison study given in a published paper)
- Novel GPU research :)

Project Outline:
A successful application/parallelization project might take the following steps:

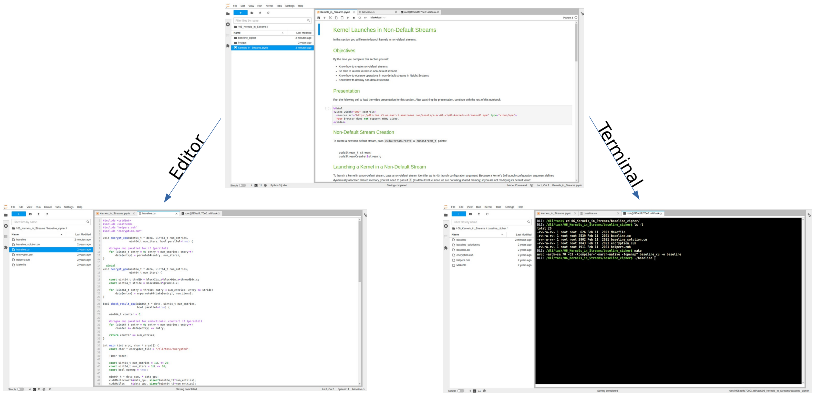
Broad Outline	Concrete Example
Choose an application.	Dense Matrix-Matrix Multiply
Determine what part of the application is taking the majority of the time.	
Determine one or more data-parallel approaches to solving the problem.	Assign one output to each thread.
Create multiple implementations of the approach.	One naïve version, one version with shared memory tiling.
Measure the performance and execution characteristics of the implementations for various parameters	Record memory transfer time, kernel time, utilization, FLOPS, etc.
Compare the performance with another solution (directive-based, library-based)	Implement/reuse CUBLAS routine and measure its performance.
Relate results to course concepts	Performance may be impacted by utilization, shared-memory accesses, memory coalescing, control divergence, streams.

This approach can be modified according to the exact goals of the project.

Figure 2: Final Term Project Definition at the Heterogeneous Parallel Programming Course.

3.4 Hardware and software infrastructure

In order to conduct research, development, and education in AC and DL, a specific hardware and software infrastructure is required. In terms of hardware, the most critical component is access to GPU(s), since they are not commonly found in personal computers. Further, the appropriate software stack is required, which includes drivers for the GPU(s) and the programming environment, frameworks, and modules relevant to the topic. Integrated development environments (IDEs) should also be easily accessible to users. Setting up an appropriate hardware and software environment for AC and DL education can be time-consuming and costly. Since one of the main goals of DLI courses is to provide hands-on programming exercises that are to be executed on GPU-based architectures, NVIDIA provides access to the participants NVIDIA GPU enabled cloud environment with all necessary software components installed. The software stack is built in separate Docker images [1], and the IDE



Kernel Launches in Non-Default Streams

In this section you will learn to launch kernels in non-default streams.

Objectives

By the time you complete this section you will:

- Know how to create non-default streams
- Be able to launch kernels in non-default streams
- Know how to observe operations in non-default streams in Nsight Systems
- Know how to destroy non-default streams

Presentation

Run the following cell to load the video presentation for this section. After watching the presentation, continue with the rest of this notebook.

```
%%html
<video width="800" controls>
  <source src="https://dli-lms.s3.us-east-1.amazonaws.com/assets/s-ac-01-v1/06-kernels-streams-01.mp4" type="video/mp4">
  Your browser does not support HTML video.
</video>
```

Non-Default Stream Creation

To create a new non-default stream, pass `cudaStreamCreate` a `cudaStream_t` pointer:

```
cudaStream_t stream;
cudaStreamCreate(&stream);
```

Launching a Kernel in a Non-Default Stream

To launch a kernel in a non-default stream, pass a non-default stream identifier as its 4th launch configuration argument. Because a kernel's 3rd launch configuration argument defines dynamically allocated shared memory, you will need to pass it `0` (its default value since we are not using shared memory) if you are not modifying its default value:

```
#include <cstdint>
#include <iostream>
#include "helpers.cuh"
#include "encryption.cuh"

void encrypt_cpu(uint64_t * data, uint64_t num_entries,
                 uint64_t num_iters, bool parallel=true) {

    #pragma omp parallel for if (parallel)
    for (uint64_t entry = 0; entry < num_entries; entry++)
        data[entry] = permute64(entry, num_iters);
}

__global__
void decrypt_gpu(uint64_t * data, uint64_t num_entries,
                 uint64_t num_iters) {

    const uint64_t thrdID = blockIdx.x*blockDim.x+threadIdx.x;
    const uint64_t stride = blockDim.x*gridDim.x;

    for (uint64_t entry = thrdID; entry < num_entries; entry += stride)
        data[entry] = unpermute64(data[entry], num_iters);
}

bool check_result_cpu(uint64_t * data, uint64_t num_entries,
                      bool parallel=true) {

    uint64_t counter = 0;

    #pragma omp parallel for reduction(+: counter) if (parallel)
    for (uint64_t entry = 0; entry < num_entries; entry++)
        counter += data[entry] == entry;

    return counter == num_entries;
}

int main (int argc, char * argv[]) {
    const char * encrypted_file = "/dli/task/encrypted";

    Timer timer;

    const uint64_t num_entries = 1UL << 26;
    const uint64_t num_iters = 1UL << 10;
    const bool openmp = true;

    uint64_t * data_cpu, * data_gpu;
    cudaMallocHost(&data_cpu, sizeof(uint64_t)*num_entries);
    cudaMalloc    (&data_gpu, sizeof(uint64_t)*num_entries);
```

```
DLI: /dli/task$ cd 06_Kernels_in_Streams/baseline_cipher/
DLI: /dli/task/06_Kernels_in_Streams/baseline_cipher$ ls -l
total 20
-rw-rw-rw- 1 root root  626 Feb 11  2021 Makefile
-rw-rw-rw- 1 root root 2539 Feb 11  2021 baseline.cu
-rw-rw-rw- 1 root root 2802 Feb 11  2021 baseline_solution.cu
-rw-rw-rw- 1 root root 1043 Feb 11  2021 encryption.cuh
-rw-rw-rw- 1 root root 1911 Feb 11  2021 helpers.cuh
DLI: /dli/task/06_Kernels_in_Streams/baseline_cipher$ make
nvcc -arch=sm_70 -O3 -Xcompiler="-march=native -fopenmp" baseline.cu -o baseline
DLI: /dli/task/06_Kernels_in_Streams/baseline_cipher$ ./baseline
```

Figure 3: Sample module interfaces in Fundamentals of Accelerated Computing workshop.

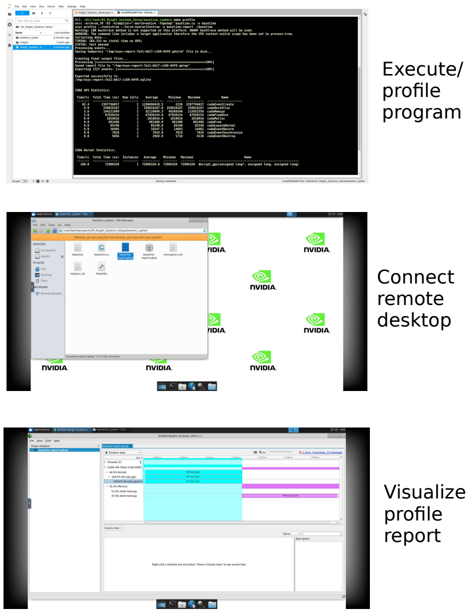
is primarily a web-based application (Jupyter Notebook and Lab, https://jupyter.org/).

The participant can utilize the cloud resources presented as Jupyter notebooks, which can be accessed by graphical notebook interface, graphical console IDE, or terminal screen. While graphical interfaces are more useful for Python-based courses like Fundamentals of Deep Learning, terminal provides more practical interface like Fundamentals of Accelerated Computing, which may require frequent source code modification and low-level analysis. Figure 3 presents one module (*Kernels_In_Streams*) and possible user interfaces in *Fundamentals of Accelerated Computing* workshop to access the module components. While the main Jupyter Notebook interface provides guidance about the module, the participant can edit the source code in the editor interface or modify/compile/execute in a terminal screen. Additionally, the courses that include visual performance analysis, based on NVIDIA Nsight Systems tool [16], offer remote desktop access, which has running Nsight Systems instance inside the JupyterLab environment. The participants can connect this desktop environment and visually profile their executions by observing performance behavior of the different code versions to see the effects on performance. Figure 4 demonstrates the phases for using remote Nsight Systems tool in DLI infrastructure:

(1) Executing the program in the terminal with *profile* option (provided in Makefile),
(2) Connecting the remote desktop and observing the report file generated at the end of the program execution,
(3) Visualizing the profile report at Nsight Systems Tool, which is already installed and configured in the remote desktop environment.

3.5 University Ambassador program

The DLI University Ambassador Program [13] enables qualified educators to teach free instructor-led courses for the academia, including university and non-profit research lab staff, students, and researchers. They are also allowed to run paid corporate workshops.



```
DLI: /dli/task/04_Nsight_Systems_Setup/baseline_cypher$ make profile
nvcc -arch=sm_70 -O3 -Xcompiler="-march=native -fopenmp" baseline.cu -o baseline
nsys profile --stats=true --force-overwrite=true -o baseline-report ./baseline
Warning: LBR backtrace method is not supported on this platform. DWARF backtrace method will be used.
WARNING: The command line includes a target application therefore the CPU context-switch scope has been set to process-tree.
Collecting data...
TIMING: 164.333 ms (total time on GPU)
STATUS: test passed
Processing events...
Saving temporary "/tmp/nsys-report-7e11-b617-c169-84f0.qdstrm" file to disk...

Creating final output files...
Processing [==============================================================100%]
Saved report file to "/tmp/nsys-report-7e11-b617-c169-84f0.qdrep"
Exporting 1727 events: [==================================================100%]

Exported successfully to
/tmp/nsys-report-7e11-b617-c169-84f0.sqlite


CUDA API Statistics:

 Time(%)  Total Time (ns)  Num Calls    Average      Minimum    Maximum            Name
 -------  ---------------  ---------  ------------  ---------  ---------  ------------------
    82.4       2397796857          2  1198898428.5       2220  2397794637  cudaEventCreate
     8.9        258018167          1   258018167.0  258018167   258018167  cudaHostAlloc
     5.6        164221899          2    82110949.5   48389340   115832559  cudaMemcpy
     3.0         87939154          1    87939154.0   87939154    87939154  cudaFreeHost
     0.0          1018616          1     1018616.0    1018616     1018616  cudaMalloc
     0.0           881488          1      881488.0     881488      881488  cudaFree
     0.0            65240          1       65240.0      65240       65240  cudaLaunchKernel
     0.0            38495          2       19247.5      14093       24402  cudaEventRecord
     0.0             7019          1        7019.0       7019        7019  cudaEventSynchronize
     0.0             5856          2        2928.0       1718        4138  cudaEventDestroy



CUDA Kernel Statistics:

 Time(%)  Total Time (ns)  Instances   Average    Minimum   Maximum                          Name
 -------  ---------------  ---------  ----------  --------  --------  ------------------------------------------------------
   100.0         72986320          1  72986320.0  72986320  72986320  decrypt_gpu(unsigned long*, unsigned long, unsigned long)
```

Figure 4: Nsight Systems Tool in remote desktop.

By completing the instructor certification process, educators affiliated with an academic institution are certified as University Ambassadors. For each workshop, DLI instructors must pass a multi-stage examination in order to become certified in the specific content. Teaching assistants are selected by the instructors. This program has several benefits: free DLI instructor certification, online ready-made workshop materials, free access to online GPU resources, and expense reimbursement for travel and catering expenses for instructor-led workshops. See [13] for detailed information about this program.

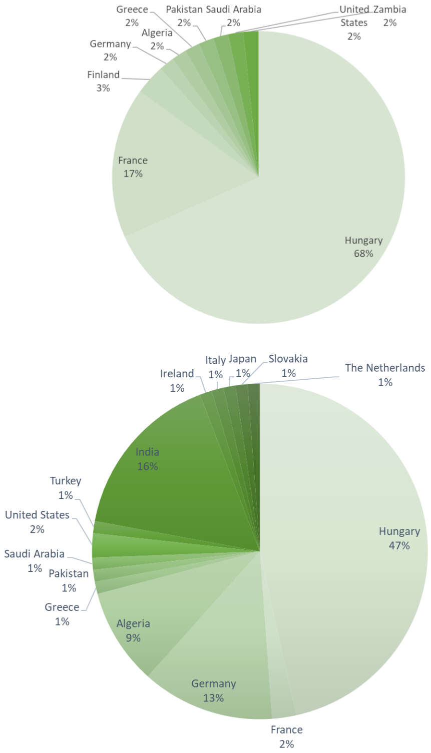


Figure 5: Country of origin of the attendees in the mixed group deliveries, on the top: FDL with 60, on the bottom: NLP with 86 attendees.

4 EVALUATION AND RESULTS

4.1 Instructor-led Deep Learning Workshops

Altogether we held 2 FDL, 1 NLP, and 3 CAI workshops in the autumn and spring semesters of 2021/2022 academic year, according to Section 3.2. All of these workshops were ran by an associate professor with 10+ years of machine learning, and 8+ years of deep learning research, development, and education experience. The number of participants of the examined workshops was as follows:

- BSc group: one FDL and one CAI were delivered in-class for 30 (22 from Hungary, 7 from the USA, 1 unknown) and 26 (22 from Hungary, 4 from the USA) students, respectively. These workshops were delivered as a part of a beginner level deep learning class at a Hungarian university.
- MSc group: two CAI were delivered online for 13 (12 from Hungary, 1 from Romania) and 38 (37 from Hungary, 1 from the USA) attendees. The students were attending to a Human-Computer Interaction class at a Hungarian university.
- Mixed group: one FDL and one NLP were delivered online for 60 and 86 attendees, respectively. The attendees' country of origin are shown in Fig. 5. These workshops were advertised in various channels in the EMEA region, including AI-related mailing list in Hungary, LinkedIn groups, and NVIDIA DLI academic partners.

Figure 6, 7, 8 show the results of the feedback forms.

Learning experience. The overall impression of the attendees was 4 or above. A weak but clear trend can be inspected that the more knowledgeable the audience was, the higher they scored the overall experience (4 and a little bit below for the BSc, 4 and a little bit above for the MSc, and around 4.5 for the Mixed group). Interestingly, similar trend is shown for the other questions (Registration, Navigation, Launch Time), however, those aspects are not directly correlated to hard skills, knowledge, and experience. There are two possible explanations for this. On the one hand, juniors are more likely to get frustrated than senior experts. There were more seniors in Mixed than in BSc and MSc groups, since it included PhD students, researchers, and educators in addition to BSc and MSc students. On the other hand, participants of mixed groups were attending the workshop on their own initiative and during their free time, so they recognized the value of the material more than university students, for whom the content was part of their course work.

Meeting the expectations. In all groups, meeting the learning objectives scored 4 or above – with the Mixed group scoring the highest. In spite of having different groups and different contents, the difficulty of the materials was considered to be similar. It reinforces that NVIDIA DLI's efforts to maintain a dense information content in the courses, but in a manner that is digestible in a full-day workshop are successful. Similar scores can be inspected for the 'clear prerequisites'. The quality of the content was scored better by more advanced groups (MSc and Mixed), and it scored 4 for the BSc group, too. In interactivity, similar weak trend can be inspected, as before. It is interesting that within the same groups FDL scored higher than the more advanced NLP and CAI content, regarding interactivity. This can be mainly the cause of the course content: When introducing deep learning for the first time, more interactions are involved in the workshop. When discussing advanced topics like NLP or CAI, the participants are considered to be more advanced, thus information content is superior to interaction.

Instructor, teaching assistants, course pace. Feedback about the instructor showed similar patterns as the previous two categories. The instructor's presentation skills and knowledge were judged quite similar by distinct groups. Interestingly, among all questions the feedback on the teaching assistant's (TA's) knowledge scored the lowest overall. The workshop TAs were all PhD candidates specialized in deep learning, they had teaching and consultation experience, and they had earned the certificate of the particular workshop in advance. The relatively lower scores (<4) may be the result of different expectations of the TAs (e.g. expecting more help in the self-paced parts of the workshop) and/or the way TAs interacted with the audience degraded the participants' experience (chat, generally).

The statistics of successful certificates are shown in Table 2. Due to the requirement to earn the certificate in order to complete the deep learning course at the university, it is understandable why the majority of attendees completed the assessment successfully in the BSc groups. In case of the MSc groups a smaller percentage of the class earned the certificate – in this case the certificate was not required, but was among the options to be exempted from the exam. In the case of the Mixed-FDL similar percentage of the group passed the assessment successfully. For Mixed-NLP the percentage dropped significantly, to 44%. The possible cause for this could be

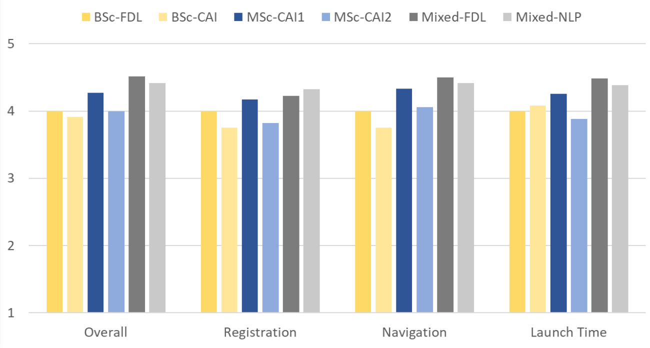


Figure 6: Results of the feedback form on the learning experience.

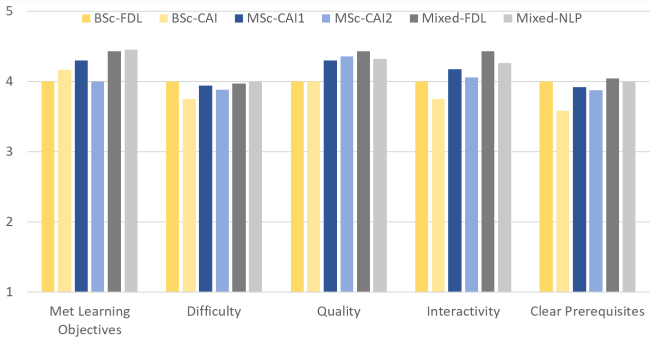


Figure 7: Results of the feedback form on the course meeting the expectations.

the timing of the workshop: this one was held in 13 December, right before the holiday season, when students and educators are also overloaded with exams, and researchers with finalizing projects at the end of the year – which allow them less time to completely participate in a full day workshop and complete its assessment.

Table 2: Percentage of participants who have obtained a certificate by completing the assessment in the given workshop.

Workshop	Percentage
BSc-FDL	94%
BSc-CAI	93%
MSc-CAI1	77%
MSc-CAI2	64%
Mixed-FDL	69%
Mixed-NLP	44%

4.2 Adopting Accelerated Computing Teaching Kit

Each year 10-20 students are registered in the Heterogeneous Parallel Programming course, and in average 60-80% of them can get a passing grade. Table 3 presents the statistics about the course in the four years. It presents the number of students in terms of enrolled in the course, failed (got F) from the course, and received the highest letter grade, AA. Additionally, *Course Evaluation* column demonstrates the average score of the evaluation survey (out of 5), where the number in parenthesis represents the score for the question about the demonstration of the course content based on the quality of the course material and effective examples. While general feedback appreciates the effort in the course, term *2020-2021* demonstrates a negatively different result with relatively low scores in the course evaluation. Since the course is taught virtually that term, we think that student involvement could not be achieved as in the face-to-face semesters. It is also remarkable that *2021-2022* evaluation results are the highest even though the number of students is not large. Since *CUDA Libraries* and *CUDA CNN* are emphasized that year, we think that the students were able to see the power of CUDA programming model and real scenarios that they can apply the methods and, as a result evaluated the course as more efficient.

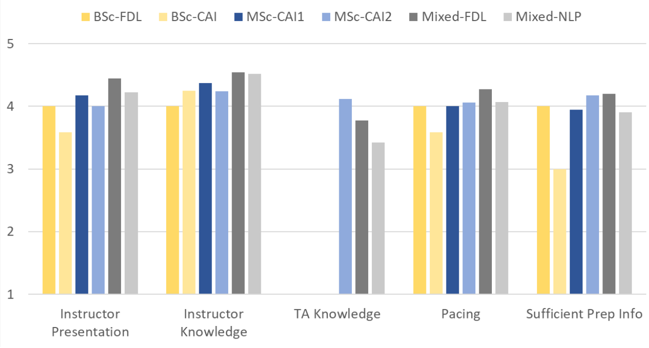


Figure 8: Results of the feedback form on instructor, teaching assistant and course pace (in BSc-FDL, BSc-CAI and MSc-CAI1 there were no teaching assistant.

Table 3: Heterogeneous Parallel Programming course statistics.

Term	#Students Taken	#Students Failed	#Students w/ AA	Course Evaluation
2021-2022	11	2	1	4.23 (4.27)
2020-2021	12	4	5	2.84 (2.89)
2019-2020	20	8	3	3.94 (3.94)
2018-2019	16	6	1	3.79 (3.60)

For the student evaluation, programming tasks were assigned to the students to demonstrate their comprehension of the concepts introduced throughout the semester. Additionally, the final project tests their skills at defining parallel programming problems, optimizing performance by considering GPU hardware and CUDA programming model features, and performing a comparison study to evaluate the effectiveness of their methods. The sample project topics in 2021-2022 semester were as follows: Perlin and fractal noise, Gaussian Jordan elimination, Dijkstra's shortest path algorithm, Convolution operations from the PolyBench benchmark. The

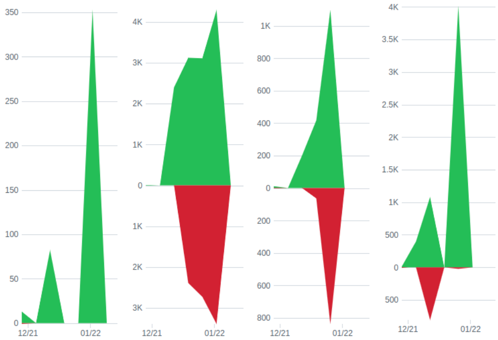


Figure 9: Number of additions and deletions per week in the GitHub projects.

students created GitHub repositories and updated their code during the semester based on a few deadlines. Figure 9 presents the code frequency in terms of additions and deletions in sample GitHub projects. In the two-month period, there are peaks at two specific points representing the deadlines. While most of the projects include basic CUDA implementations, one project is extended as a conference paper and presented at a national conference by the student [24].

5 SUMMARY

In this paper the primary challenges of accelerated computing and deep learning education was introduced, the offerings of NVIDIA Deep Learning Institute were discussed and instructor-led full day workshops and teaching kits were evaluated. The feedback form filled after the workshops revealed that in case of all examined content the overall satisfaction with the learning experience were between 3.9...4.5 (out of 5). The results also showed us, that more experienced groups scored various aspects higher (e.g. overall impression, quality of the content, interactivity, impressions about the instructor, etc.). No significant difference in difficulty was observed between beginner and advanced workshops, based on the feedback scores. Surprisingly, experienced teaching assistants received rather lower scores (between 3.4..4.3) compared to other questions in the feedback forms.

Based on the course evaluation questions and the implementation of the term projects, we can conclude that the adoption of Teaching Kits was a success.

It is our overall impression and conclusion that the content created by NVIDIA DLI can be easily and successfully integrated into related university courses for smaller and larger groups. DLI content can even be implemented in classes that are not directly related to AC or DL (e.g. the Human-Computer Interaction MSc course) with a great learning experience – based on our findings.

ACKNOWLEDGMENTS

The authors are thankful for the support of NVIDIA Deep Learning Institute. The work reported in this paper has been partly supported by the the European Union project RRF-2.3.1-21-2022-00004 within the framework of the Artificial Intelligence National Laboratory.

REFERENCES

[1] Carl Boettiger. 2015. An introduction to Docker for reproducible research. *ACM SIGOPS Operating Systems Review* 49, 1 (2015), 71–79.

[2] Rohit Chandra, Leo Dagum, David Kohr, Ramesh Menon, Dror Maydan, and Jeff McDonald. 2001. *Parallel programming in OpenMP.* Morgan kaufmann.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[4] Rebecca Fiebrink. 2019. Machine learning education for artists, musicians, and other creative practitioners. *ACM Transactions on Computing Education (TOCE)* 19, 4 (2019), 1–32.

[5] John L Hennessy and David A Patterson. 2011. *Computer architecture: a quantitative approach.* Elsevier.

[6] Joel Hestness, Sharan Narang, Newsha Ardalani, Gregory Diamos, Heewoo Jun, Hassan Kianinejad, Md Patwary, Mostofa Ali, Yang Yang, and Yanqi Zhou. 2017. Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409* (2017).

[7] David B Kirk and W Hwu Wen-Mei. 2016. *Programming massively parallel processors: a hands-on approach.* Morgan kaufmann.

[8] Karen Kreeger. 2003. The learning curve. *Nature Biotechnology* 21, 8 (2003), 951–952.

[9] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *nature* 521, 7553 (2015), 436–444.

[10] Linda Macaulay, Claire Moxham, Barbara Jones, and Ian Miles. 2010. Innovation and Skills. In *Handbook of Service Science.* Springer, 717–736.

[11] NVIDIA. 2022. Building Conversational AI Applications. Retrieved September 8, 2022 from https://www.nvidia.com/en-us/training/instructor-led-workshops/building-conversational-ai-apps/

[12] NVIDIA. 2022. Building Transformer-Based Natural Language Processing Applications. Retrieved September 8, 2022 from https://www.nvidia.com/en-us/training/instructor-led-workshops/natural-language-processing/

[13] NVIDIA. 2022. DLI University Ambassador Program. Retrieved September 8, 2022 from https://www.nvidia.com/en-us/training/educator-programs/university-ambassador-program/

[14] NVIDIA. 2022. Fundamentals of Deep Learning. Retrieved September 8, 2022 from https://www.nvidia.com/en-us/training/instructor-led-workshops/fundamentals-of-deep-learning/

[15] NVIDIA. 2022. Modeling Time Series Data with Recurrent Neural Networks in Keras. Retrieved September 8, 2022 from https://courses.nvidia.com/courses/course-v1:DLI+L-FX-24+V1/

[16] NVIDIA. 2022. NVIDIA Nsight Systems. Retrieved September 8, 2022 from https://developer.nvidia.com/nsight-systems

[17] Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. 2021. Deep learning for anomaly detection: A review. *ACM Computing Surveys (CSUR)* 54, 2 (2021), 1–38.

[18] Apan Qasem and David P. Bunde. 2022. Heterogeneous Computing for Undergraduates: Introducing the ToUCH Module Repository. In *SIGCSE 2022: The 53rd ACM Technical Symposium on Computer Science Education, Providence, RI, USA, March 3-5, 2022, Volume 2.* ACM, 1201. https://doi.org/10.1145/3478432.3499152

[19] Apan Qasem, David P. Bunde, and Philip Schielke. 2021. A module-based introduction to heterogeneous computing in core courses. *J. Parallel Distributed Computing* 158 (2021), 56–66. https://doi.org/10.1016/j.jpdc.2021.07.011

[20] Samik Sadhu, Di He, Che-Wei Huang, Sri Harish Mallidi, Minhua Wu, Ariya Rastrow, Andreas Stolcke, Jasha Droppo, and Roland Maas. 2021. Wav2vec-c: A self-supervised model for speech representation learning. *arXiv preprint arXiv:2103.08393* (2021).

[21] Oscar Serradilla, Ekhi Zugasti, Jon Rodriguez, and Urko Zurutuza. 2022. Deep learning models for predictive maintenance: a survey, comparison, challenges and prospects. *Applied Intelligence* (2022), 1–31.

[22] Daniel Spikol, Emanuele Ruffaldi, Giacomo Dabisias, and Mutlu Cukurova. 2018. Supervised machine learning in multimodal learning analytics for estimating success in project-based learning. *Journal of Computer Assisted Learning* 34, 4 (2018), 366–377.

[23] Top500.org. 2022. ORNL's Frontier First to Break the Exaflop Ceiling. Retrieved August 11, 2022 from https://www.top500.org/news/ornls-frontier-first-to-break-the-exaflop-ceiling

[24] Burak Topçu and Işıl Öz. 2022. Performance Evaluation of CUDA Optimizations for Convolution Operations. In *Yüksek Başarımlı Hesaplama Konferansı (BAŞARIM).* https://indico.truba.gov.tr/event/50/attachments/231/457/BASARIM2022_Proceedings.pdf

[25] Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. 2022. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *International Conference on Machine Learning.* PMLR, 23965–23998.