@MorrisRiedel

NEURAL ARCHITECTURE SEARCH WITH REINFORCEMENT LEARNING

PROF. DR. – ING. MORRIS RIEDEL, JUELICH SUPERCOMPUTING CENTRE (JSC) / UNIVERSITY OF ICELAND

HEAD OF HIGH PRODUCTIVITY DATA PROCESSING & CROSS-SECTIONAL TEAM DEEP LEARNING

23TH AUGUST, SUMMER SCHOOL ON BIG DATA & MACHINE LEARNING, TECHNICAL UNIVERSITY OF DRESDEN

ScaDS DRESDEN LEIPZIG

BBDC BERLIN BIG DATA CENTER

HELMHOLTZ RESEARCH FOR GRAND CHALLENGES

HAICU HELMHOLTZ ARTIFICIAL INTELLIGENCE COOPERATION UNIT

DEEP Projects

UNIVERSITY OF ICELAND
SCHOOL OF ENGINEERING AND NATURAL SCIENCES
FACULTY OF INDUSTRIAL ENGINEERING, MECHANICAL ENGINEERING AND COMPUTER SCIENCE

JÜLICH Forschungszentrum

JÜLICH SUPERCOMPUTING CENTRE

OUTLINE

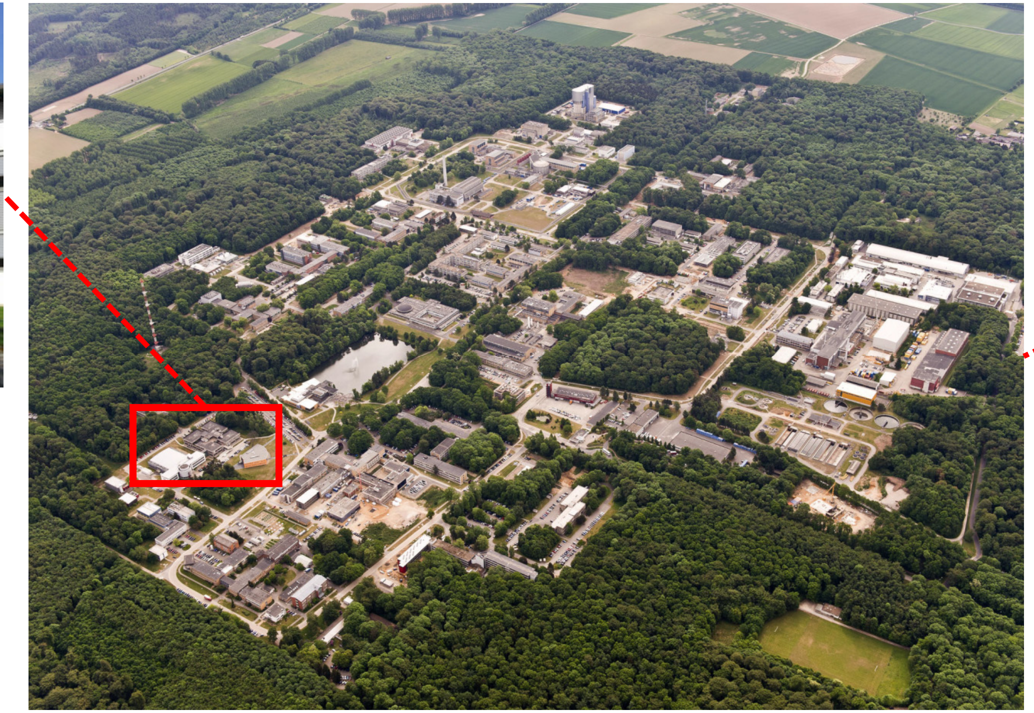
- Machine Learning & High Performance Computing (HPC)
  - @ Juelich Supercomputing Centre (JSC) & University of Iceland & Modular Supercomputing
- Machine Learning & Deep Learning Fundamentals
  - Learning approaches & Relationship HPC, Deep Learning & Big Data
- Motivation for Neural Architecture Search (NAS)
  - Growing Complexity of Machine Learning Model Parameters, Hyper-Parameters & Architectures
  - Traditional Search Approaches & Challenges using Remote Sensing Application Examples
- Neural Architecture Search Aproaches
  - Fundamentals & Overlap with Hyper-Parameter Optimization & Meta-Learning
  - Using Reinforcement Learning Techniques & Examples
- Summary

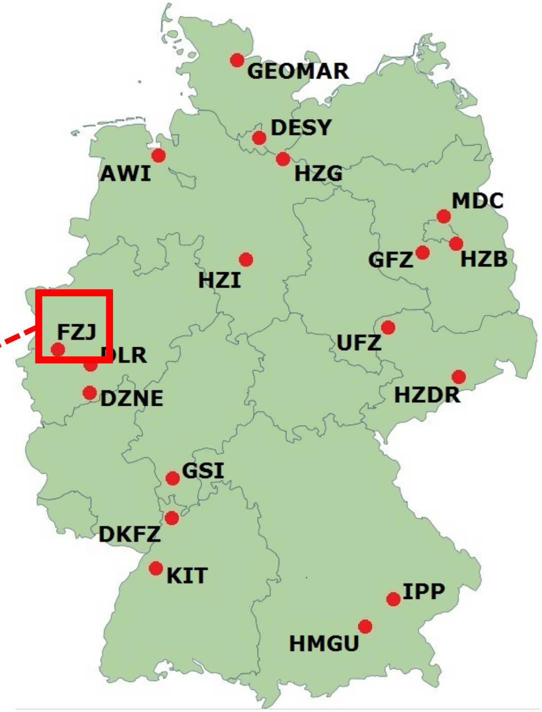
MACHINE LEARNING & HIGH PERFORMANCE COMPUTING

@ Juelich Supercomputing Centre (JSC) & University of Iceland & Modular Supercomputing

JUELICH SUPERCOMPUTING CENTRE (JSC)

Institute of Multi-Disciplinary Research Centre Juelich of the Helmholtz Association in Germany





HELMHOLTZ
RESEARCH FOR GRAND CHALLENGES

[1] Holmholtz Association Web Page

- Selected Facts
  - One of EU largest inter-disciplinary research centres (~5000 employees)
  - Special expertise in physics, materials science, nanotechnology, neuroscience and medicine & information technology (HPC & Data)

UNIVERSITY OF ICELAND

School of Engineering & Natural Sciences (SENS)

- Selected Facts
  - *Ranked among the top 300 universities in the world (by Times Higher Education)*
  - ~2900 students at the SENS school
  - Long collaboration with Forschungszentrum Juelich
  - *~350 MS students & ~150 PhD students*
  - *Many foreign & Erasmus students; english courses*

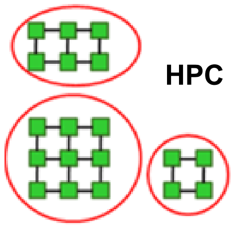
UNIVERSITY OF ICELAND
SCHOOL OF ENGINEERING AND NATURAL SCIENCES
FACULTY OF INDUSTRIAL ENGINEERING, MECHANICAL ENGINEERING AND COMPUTER SCIENCE

[2] University of Iceland Web page

UNDERSTANDING HIGH PERFORMANCE COMPUTING

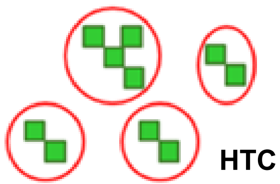
In Comparison with High Throughput Computing

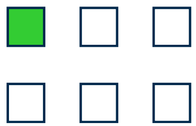
- **High Performance Computing (HPC) is based on computing resources that enable the efficient use of parallel computing techniques through specific support with dedicated hardware such as high performance cpu/core interconnections.**



network interconnection important

- **High Throughput Computing (HTC) is based on commonly available computing resources such as commodity PCs and small clusters that enable the execution of 'farming jobs' without providing a high performance interconnection between the cpu/cores.**





network interconnection less important!

HPC & DATA SCIENCE: A FIELD OF CONSTANT EVOLUTION

Perspective: Floating Point Operations per one second (FLOPS or FLOP/s)

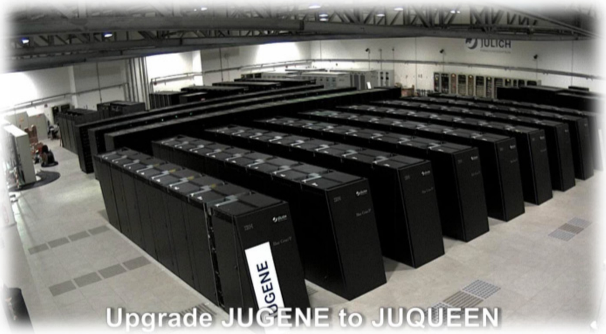
1.000.000 FLOP/s

~1984

- **1 GigaFlop/s = 10^9 FLOPS**
- **1 TeraFlop/s = 10^{12} FLOPS**
- **1 PetaFlop/s = 10^{15} FLOPS**
- **1 ExaFlop/s = 10^{18} FLOPS**

1.000.000.000.000.000 FLOP/s

~295.000 cores~2009 (JUGENE)



>5.900.000.000.000.000 FLOP/s
~ 500.000 cores
~2013 → end of service in 2018

GERMAN GAUSS CENTRE FOR SUPERCOMPUTING

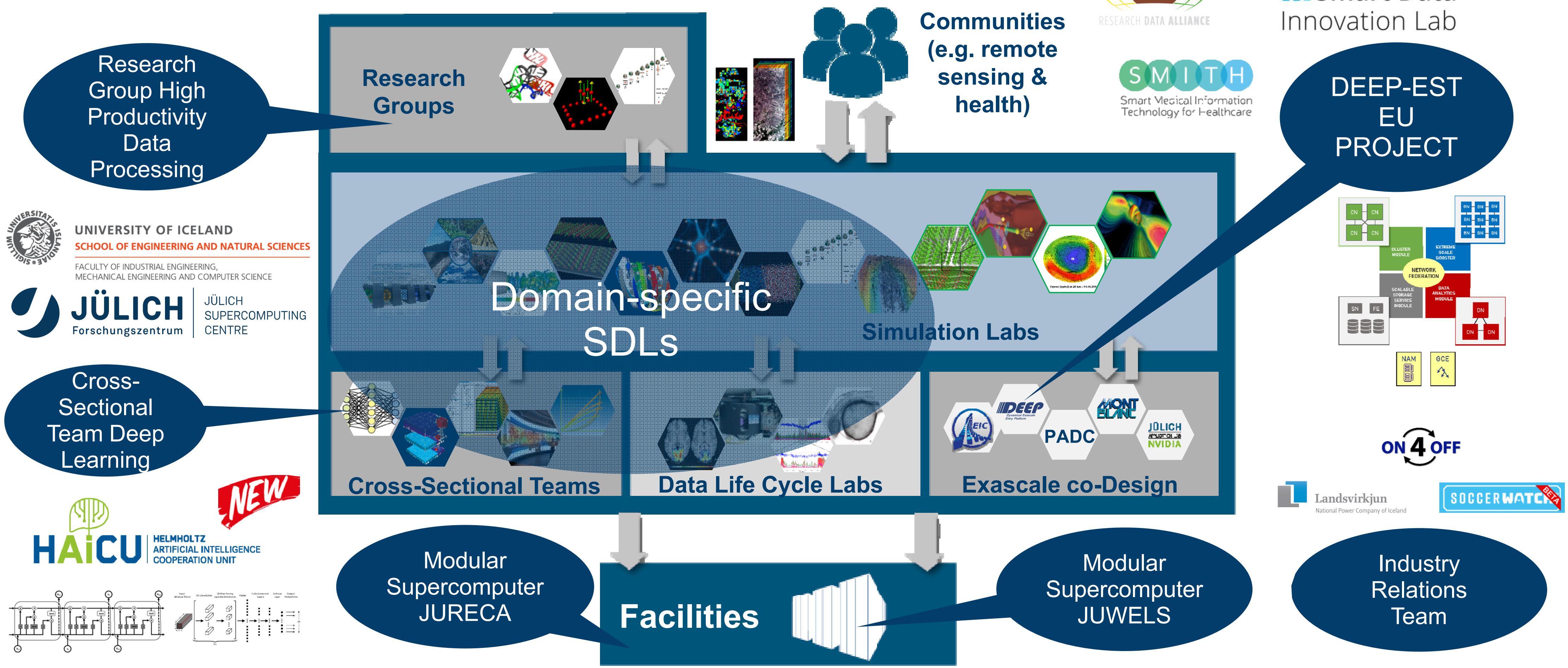
Alliance of the three national supercomputing centres HLRS (Stuttgart), JSC (Juelich) & LRZ (Munich)

[3] GCS Web page

- Supercomputer JUWELS @ JSC
  - Juelich Wizard for European Leadership Science (JUWELS)
  - Cluster architecture based on commodity multi-core CPUs
  - 2,550 compute nodes: two Intel Xeon 24-core Skylake CPUs & 48 accelerated compute nodes (4 NVIDIA Volta GPUs)
- Supercomputer SuperMUC @ LRZ
  - 155,000 cores
- Supercomputer Hazel Hen @HLRS
  - 185,088 compute cores
- GCS represents Germany in PRACE

JUELICH SUPERCOMPUTING CENTRE (JSC) OF FZJ

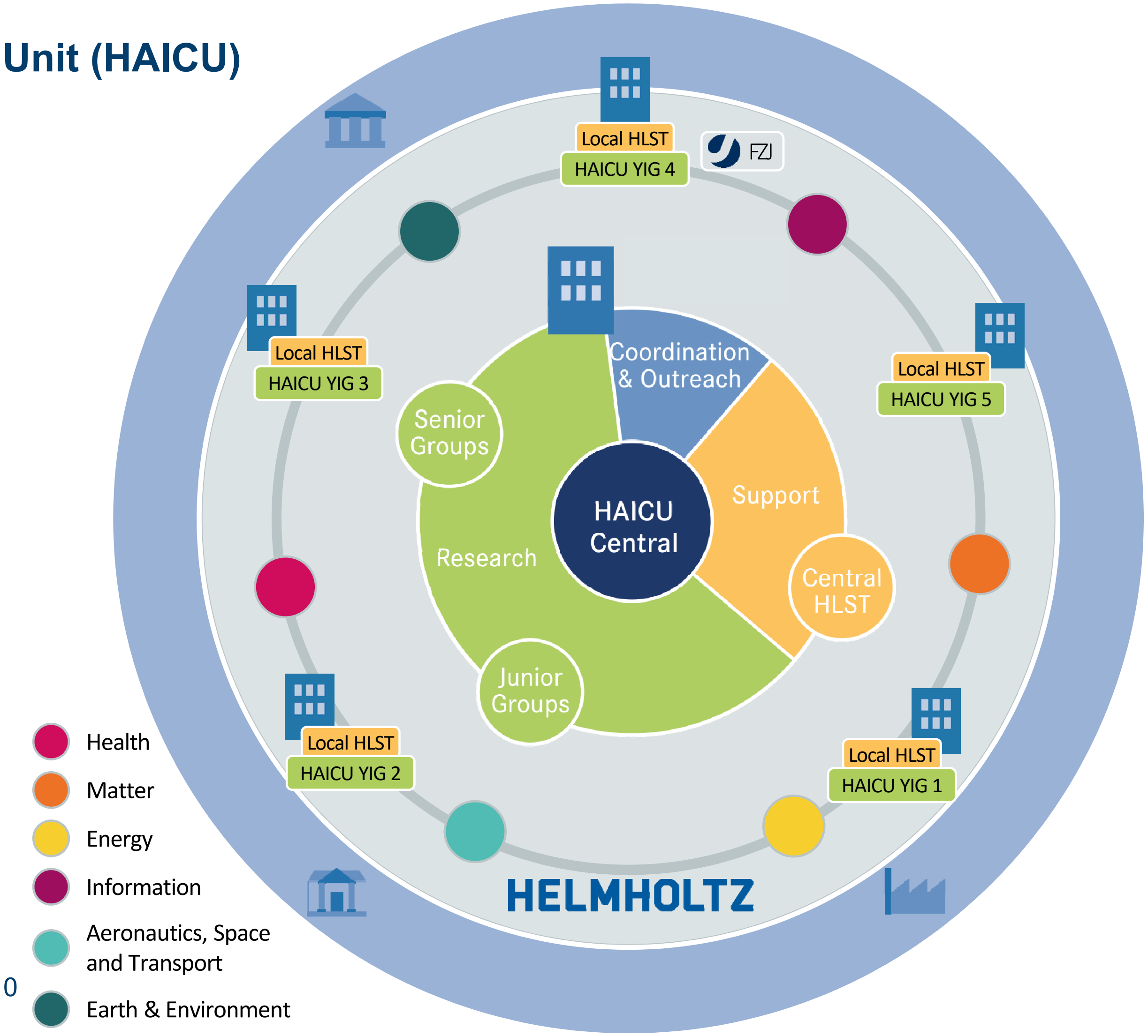
Simulation & Data Labs (SDL) using High Performance Computing (HPC)



AI COOPERATION IN HELMHOLTZ

Helmholtz Artificial Intelligence Cooperation Unit (HAICU)

- **Forschungszentrum Jülich (HAICU Local ‘Information‘)**
  - **Young Investigator Group at INM-1 (~3 FTEs)**
  - **High Level Support Team (HLST) at JSC (~ 5 FTEs)**
- Helmholtz Zentrum München (HMGU)
  (HAICU Central ‘Health‘)
- Karlsruhe Institute of Technology (KIT)
  (HAICU Local ‘Energy’)
- Helmholtz-Zentrum Geesthacht (HZG)
  (HAICU Local ‘Earth & Environment‘)
- Helmholtz-Zentrum Dresden Rossendorf (HZDR)
  (HAICU Local ‘Matter‘)
- German Aerospace Center (DLR)
  (HAICU Local ‘Aeronautic/Space & Transport‘)



NEW

HAICU HELMHOLTZ ARTIFICIAL INTELLIGENCE COOPERATION UNIT

~11.4 million € / year

[4] HAICU Web Page

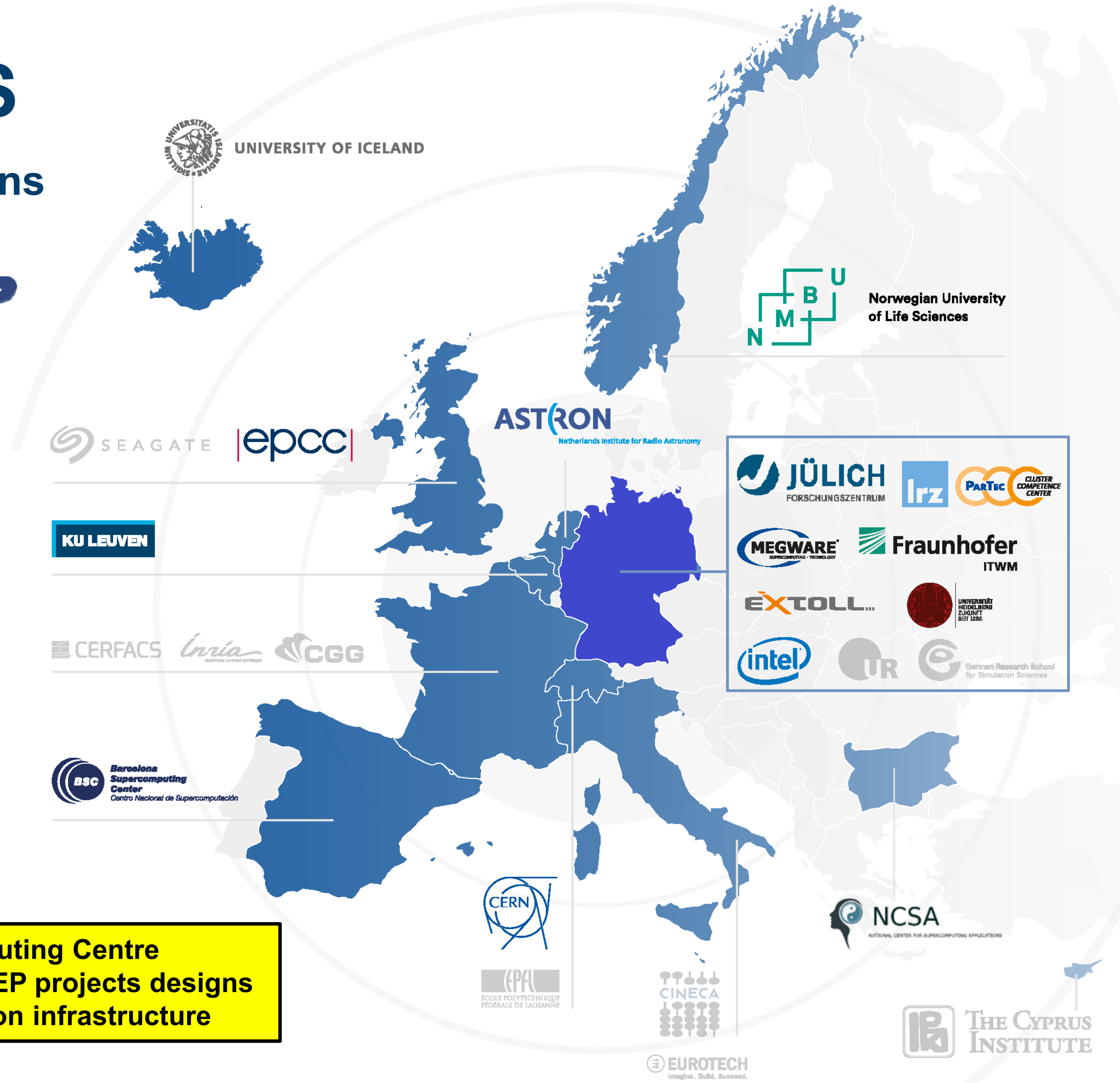
DEEP SERIES OF PROJECTS

EU Projects Driven by Co-Design of HPC Applications

- 3 EU Exascale projects
  DEEP, DEEP-ER, DEEP-EST
- 27 partners
  Coordinated by JSC
- EU-funding: 30 M€
  JSC-part > 5,3 M€
- Nov 2011 – Dec 2020

Strong collaboration with our industry partners Intel, Extoll & Megware

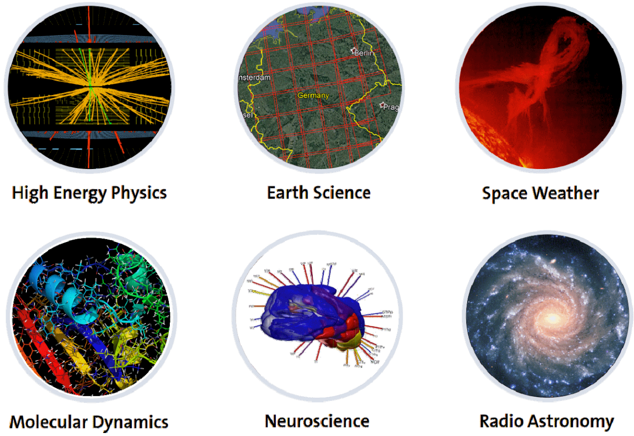
Juelich Supercomputing Centre implements the DEEP projects designs in its HPC production infrastructure

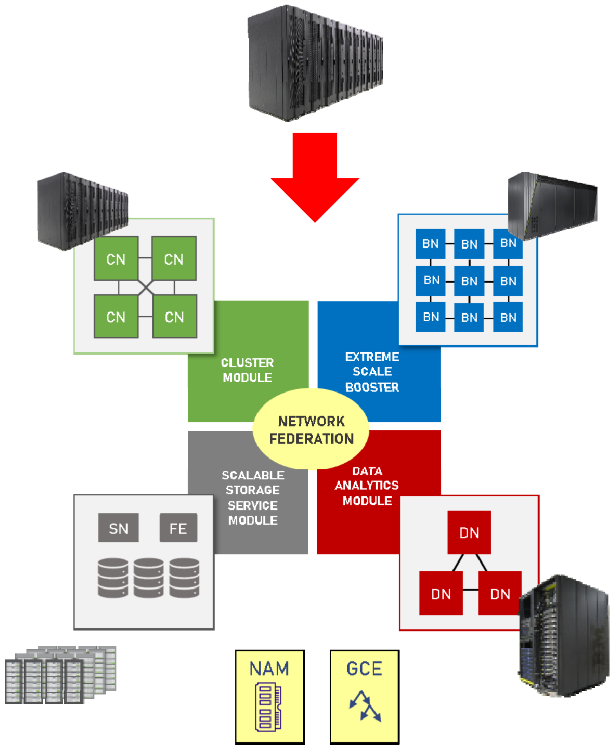


[5] DEEP Projects Web Page

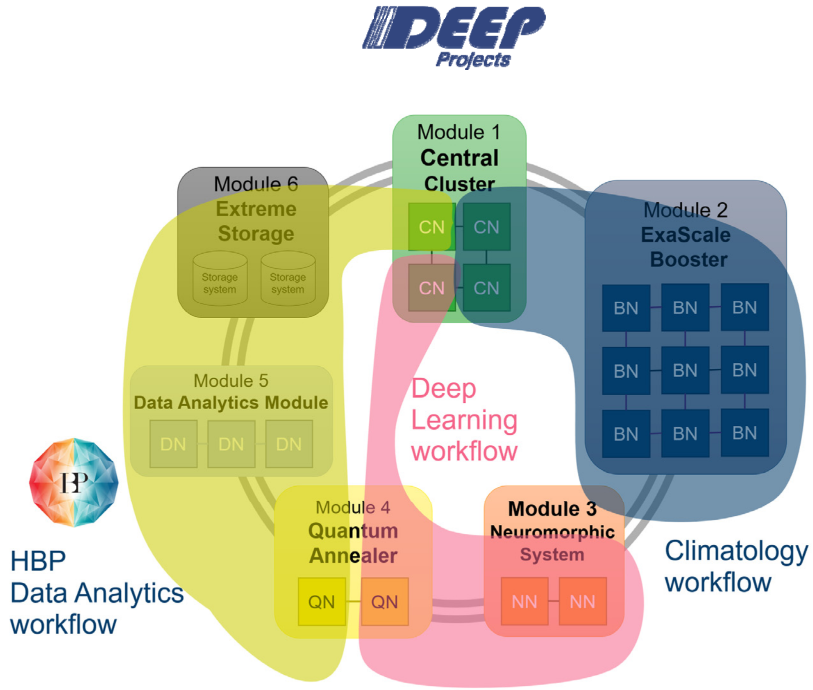
IMPACTS OF ARTIFICIAL INTELLIGENCE IN HPC DESIGN

Co-Design via Requirements from Machine/Deep Learning Applications & Innovative Simulation Sciences





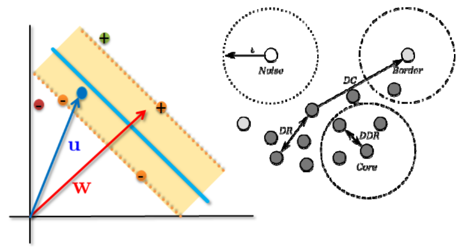
The modular supercomputing architecture (MSA) enables a flexible HPC system design co-designed by the need of different application workloads



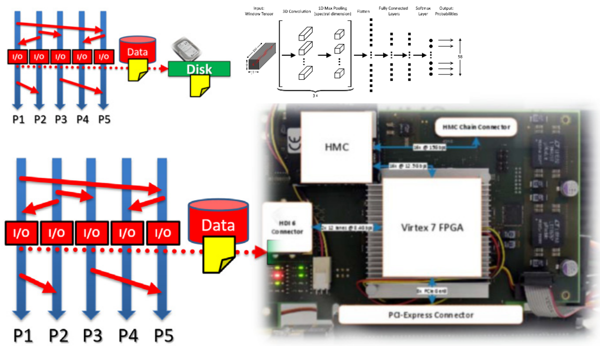
[5] DEEP Project Web page

DRIVING INNOVATIVE HPC FOR MACHINE LEARNING

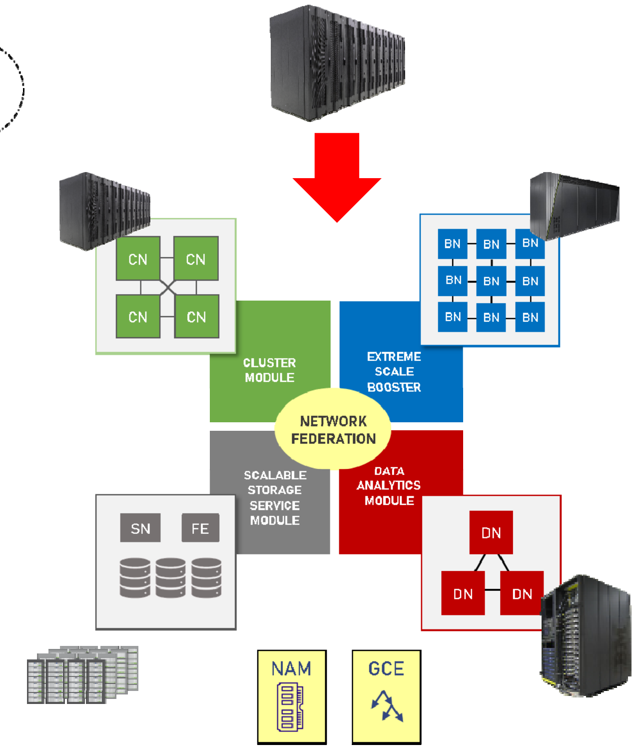
Co-Design of Innovative HPC Memory Designs and GPU/CPU Communications in Modular Supercomputing



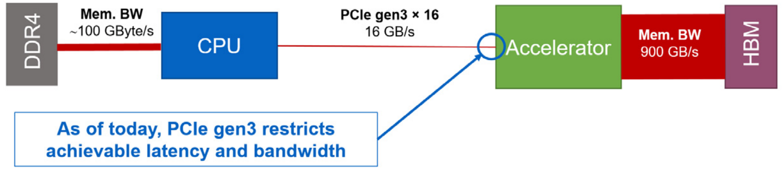
Explore Network Attached Memory (NAM)



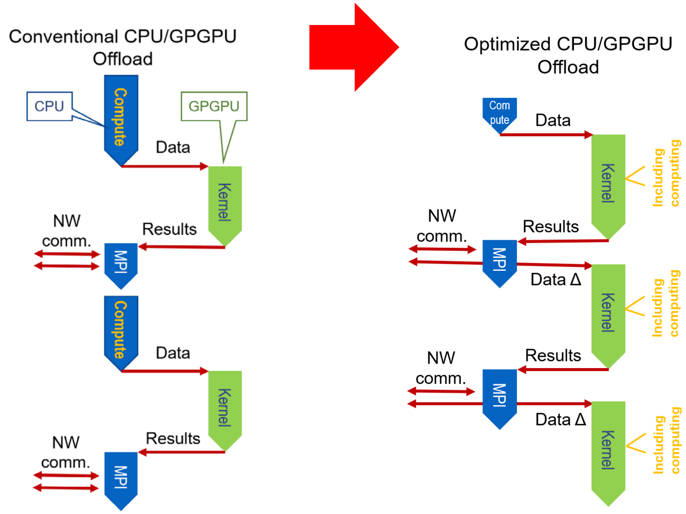
[26] E. Erlingsson, M. Riedel et al.,
IEEE MIPRO Conference, 2018



The modular supercomputing architecture (MSA) enables a flexible HPC system design co-designed by the need of different application workloads

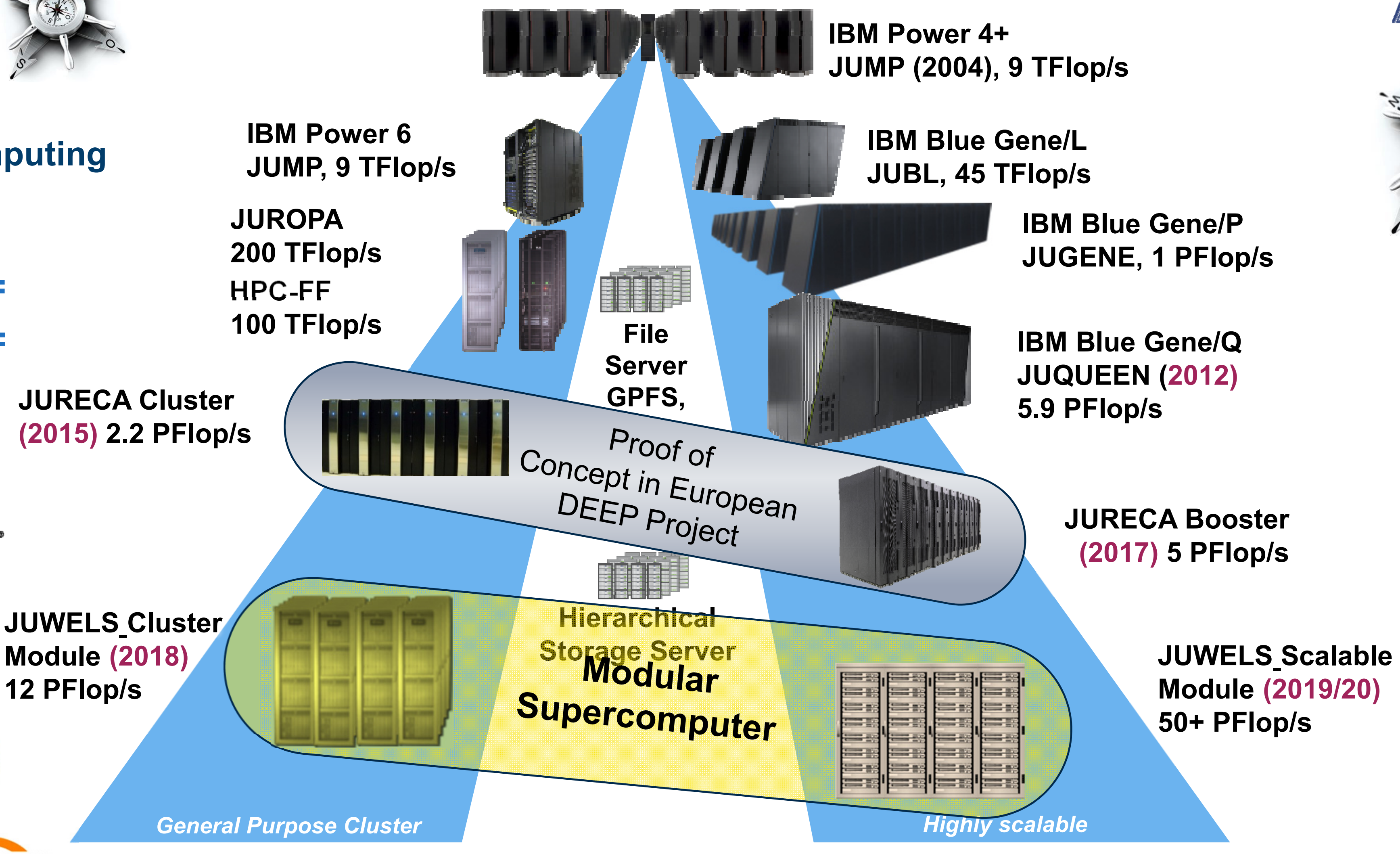


Explore more scalability with NVIDIA GPUDirect beyond one node compared to NVIDIA NVLink/NVSwitch ‘islands‘



JSC

Modular Supercomputing Roadmap



MACHINE LEARNING & DEEP LEARNING FUNDAMENTALS

Learning approaches & Relationship HPC, Deep Learning & Big Data

ARTIFICIAL INTELLIGENCE OVERVIEW

Terminology & Methods

Artificial Intelligence (AI)

A wide area of techniques and tools that enable computers to mimic human behaviour (+ robotics)

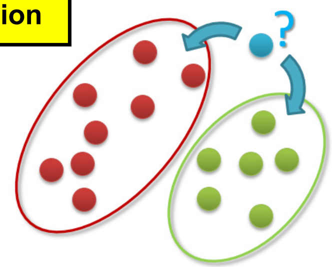
Machine Learning (ML)

Learning from data without explicitly being programmed with common programming languages

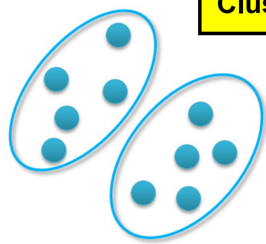
Deep Learning (DL)

Systems with the ability to learn underlying features in data using large neural networks

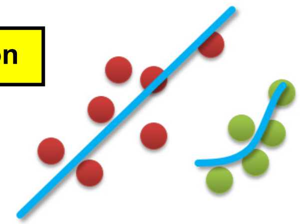
Classification



Clustering



Regression



LEARNING APPROACHES

What means learning from data?

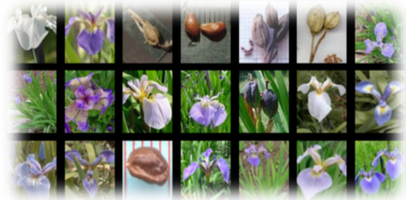
- **The basic meaning of learning is ‘to use a set of observations to uncover an underlying process or pattern in the dataset‘**
- **The three different learning approaches can be roughly categorized in supervised, unsupervised, and reinforcement learning**

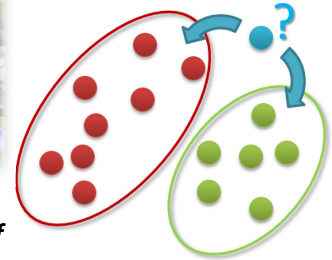
- Supervised Learning
  - Majority of methods follow this approach as groundtruth or labels exist to guide the learning best
  - Example: credit card approval based on previous customer applications
- Unsupervised Learning
  - Often applied before other learning → higher level data representation & data exploration process
  - Example: Coin recognition in vending machine based on weight and size
- Reinforcement Learning
  - Typical ‘human way‘ of learning
  - Example: Toddler tries to touch a hot cup of tea (again and again)

LEARNING APPROACHES

Supervised Learning

Classification





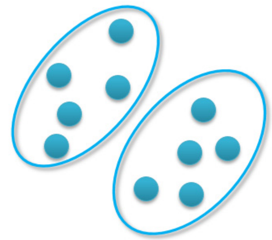
[6] Image sources: Species Iris Group of North America Database, www.signa.org

- Each observation of the predictor measurement(s) has an associated response measurement:
  - Input $\mathbf{x} = x_1, \ldots, x_d$
  - Output $y_i, i = 1, .., n$
  - Data $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$
- Goal: Fit a model that relates the response to the predictors
  - Prediction: Aims of accurately predicting the response for future observations
  - Inference: Aims to better understanding the relationship between the response and the predictors
  - Relatively straightforward to apply when the quality of labels are good

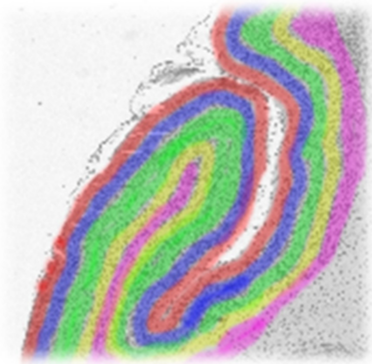
- **Supervised learning approaches fits a model that related the response to the predictors**
- **Supervised learning approaches are used in classification algorithms such as SVMs**
- **Supervised learning works with data = [input, correct output]**

LEARNING APPROACHES

Unsupervised Learning



Clustering



- Each observation of the predictor measurement(s) has no associated response measurement:
  - Input $\mathbf{x} = x_1, \ldots, x_d$
  - No output
  - Data $(\mathbf{x}_1), \ldots, (\mathbf{x}_N)$
- Goal: Seek to understand relationships between the observations
  - Clustering analysis: check whether the observations fall into distinct groups
  - Challenge: No response/output that could supervise our data analysis
  - Challenge: Clustering groups that overlap might be hardly recognized as distinct group

- **Unsupervised learning approaches seek to understand relationships between the observations**
- **Unsupervised learning approaches are used in clustering algorithms such as k-means, etc.**
- **Unupervised learning works with data = [input, ---]**

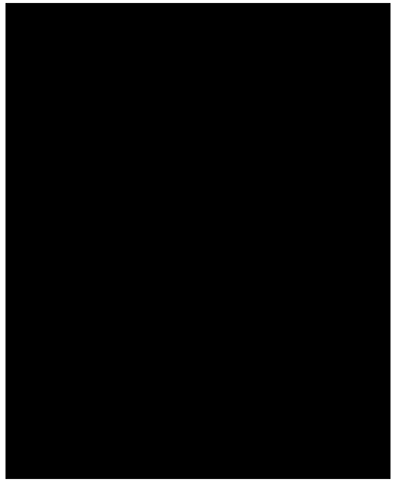
LEARNING APPROACHES

Reinforcement Learning

Learn to play games

- Each observation of the predictor measurement(s) has some associated response measurement:
  - Input $\mathbf{x} = x_1, \ldots, x_d$
  - Some output & grade of the output
  - Data $(\mathbf{x}_1), \ldots, (\mathbf{x}_N)$
- Goal: Learn through iterations
  - Guided by output grade: check learning and compare with grade
  - Challenge: Iterations may require lots of CPU time (e.g. backgammon playing rounds)
  - Challenge: Usually considered as a complicated learning approach but with applications in gaming

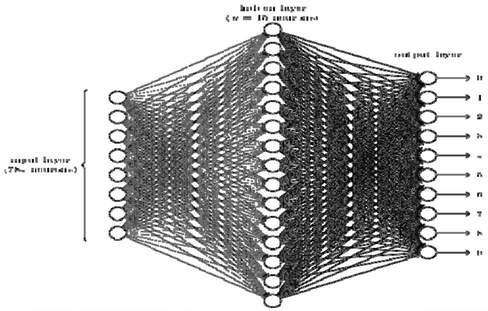
[7] Video source: Google DeepMind's Deep Q-learning playing Atari Breakout

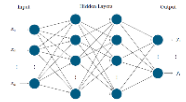


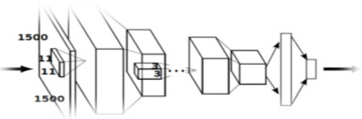
- **Reinforcement learning approaches learn through iterations using the grading output as guide**
- **Reinforcement learning approaches are used in playing game algorithms (e.g backgammon)**
- **Unupervised learning works with data = [input, some output, grade for this output]**

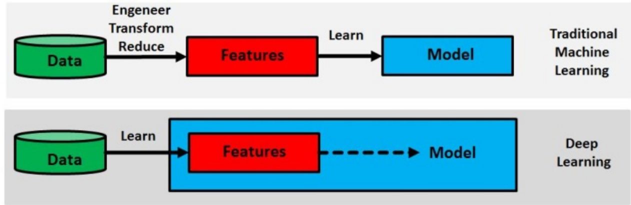
INNOVATIVE DEEP LEARNING TECHNOLOGIES

Short Introduction & Role of Cross-Sectional Team Deep Learning @ JSC







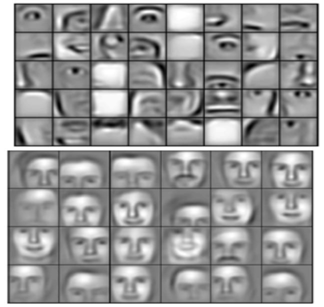


[8] M. Riedel, 'Deep Learning - Using a Convolutional Neural Network', Invited YouTube Lecture, six lectures, University of Ghent, 2017

[9] M. Riedel et al., 'Introduction to Deep Learning Models', JSC Tutorial, three days, JSC, 2019

JÜLICH Forschungszentrum | JÜLICH SUPERCOMPUTING CENTRE

HAICU HELMHOLTZ ARTIFICIAL INTELLIGENCE COOPERATION UNIT



[10] H. Lee et al., 'Convolutional Deep Belief Networks for Scalable Unsupervised Learning of Hierarchical Representations'

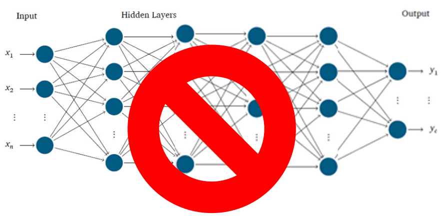
Cross-Sectional Team Deep Learning

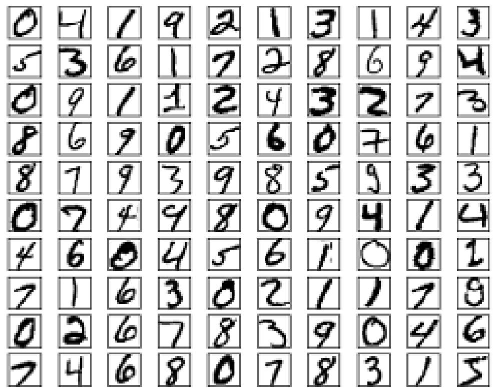
- **Provide deep learning tools that work with HPC machines (e.g. Python/Keras/Tensorflow)**
- **Advance deep learning applications and research on HPC prototypes (e.g. DEEP-EST, SMITH, etc.)**
- **Engage with industry (industrial relations team) & support SMEs (e.g. Soccerwatch, ON4OFF)**
- **Offer tutorials & application enabling support for commercial & scientific users (e.g. YouTube)**
- **Cooperate in a artificial intelligence network across Helmholtz Association (e.g. HAICU)**

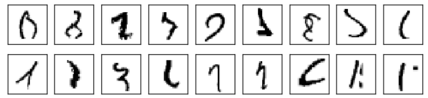
DEEP LEARNING TECHNIQUE EXAMPLE

Convolutional Neural Network (CNN) for Image Analysis

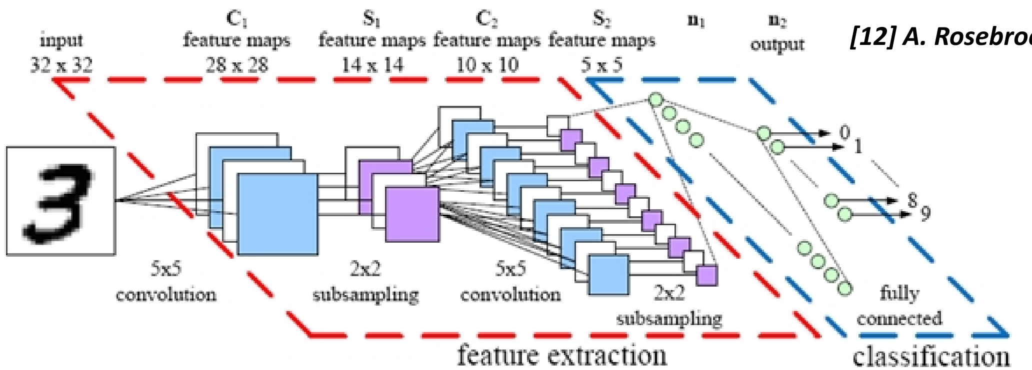
[11] Video Source: Neural Network 3D Simulation







- **Innovation via specific layers and architecture types**

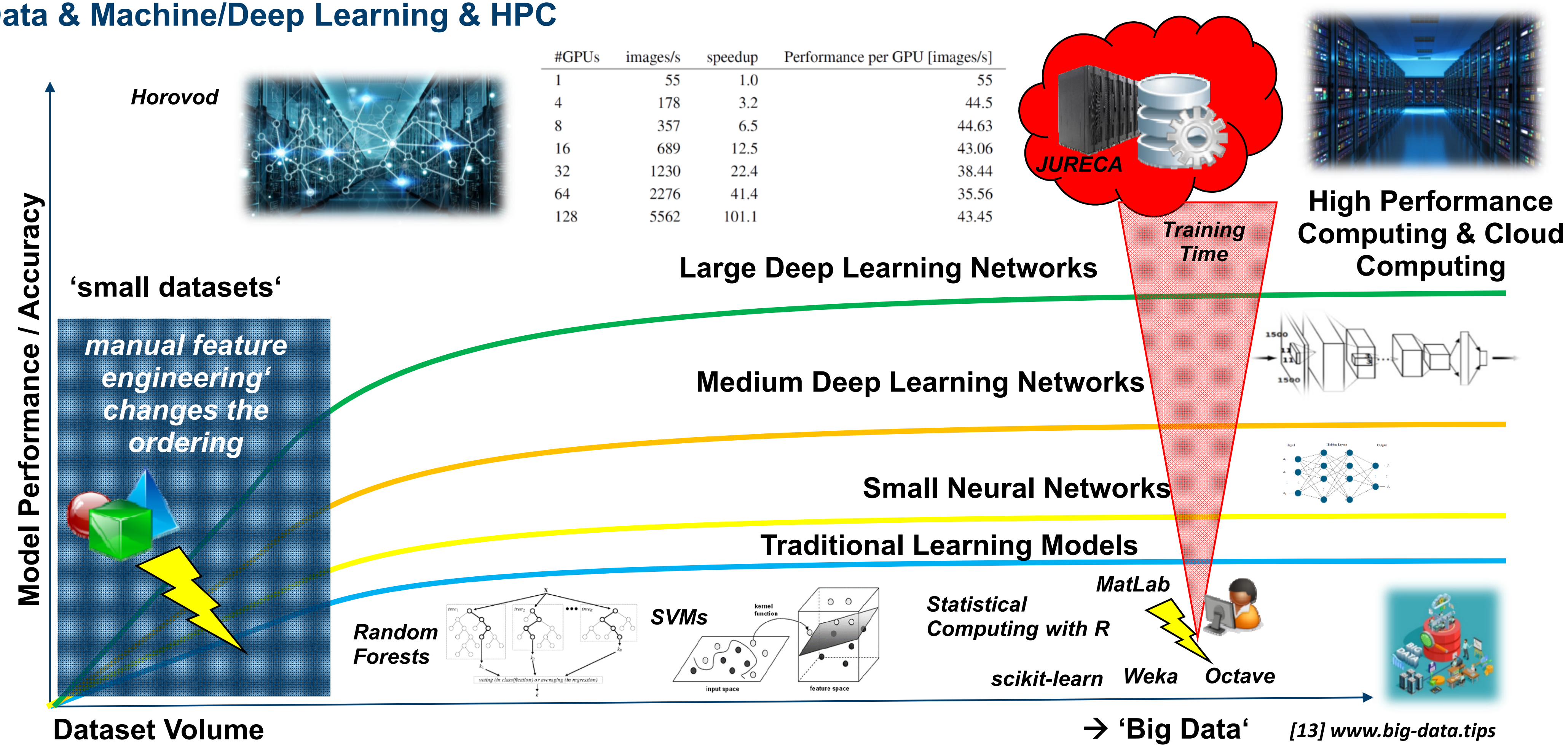


[12] A. Rosebrock

ARTIFICIAL INTELLIGENCE – COMPLEX RELATIONSHIPS

Big Data & Machine/Deep Learning & HPC

#GPUs	images/s	speedup	Performance per GPU [images/s]
1	55	1.0	55
4	178	3.2	44.5
8	357	6.5	44.63
16	689	12.5	43.06
32	1230	22.4	38.44
64	2276	41.4	35.56
128	5562	101.1	43.45



DEEP LEARNING APPLICATION EXAMPLE

Understanding the Different Factors that all Combined Provide new Chances – NOW

1952 Stochastic Gradient Descent
- Solving optimization problems

1958 Perceptron Learning Model
- Learning weights

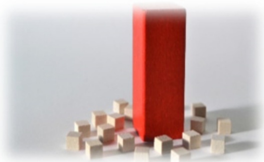
⋮

1985 'Backpropagation of Error' approch in learning
- Artificial Neural Networks

1995 Deep Convolutional Neural Networks
- Significant improvements in image analysis

⋮

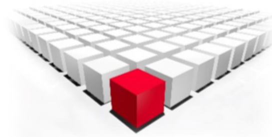
Impact in AI & HPC in industry & science



Big Data
- Large datasets
- Easy access
- More storage for less cost

Hardware
- More memory
- Graphical Processing Units (GPUs)
- HPC & parallel systems

[24] NVIDIA



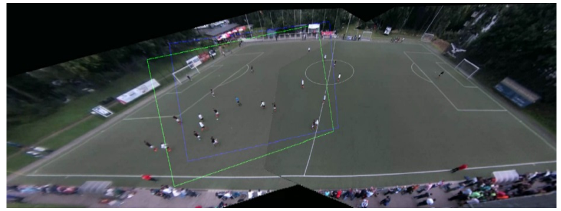
Software
- Scalable data science tools
- New learning models
- Open Source & free software packages

Keras

[25] Keras

TensorFlow

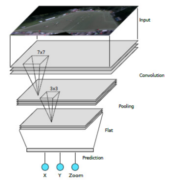
[26] TensorFlow



UNIVERSITÄT DUISBURG ESSEN

eXIST Existenzgründungen aus der Wissenschaft

[14] soccerwatch.tv



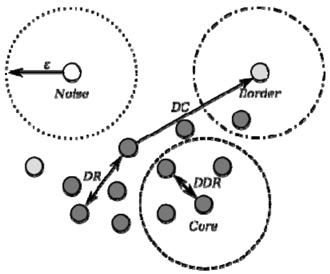
Combination: Start-up Example of my research group

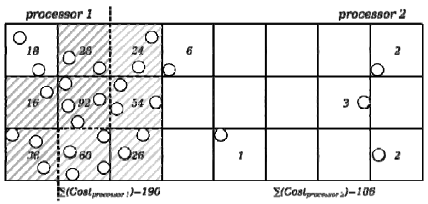
MOTIVATION FOR NEURAL ARCHITECTURE SEARCH (NAS)

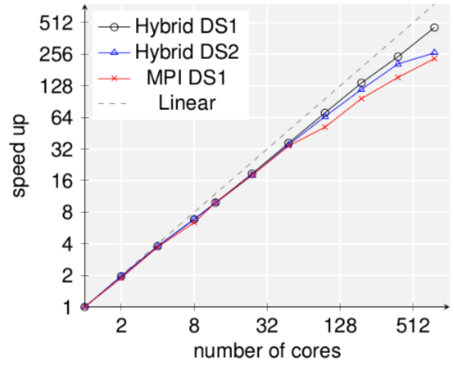
Growing Complexity of Machine Learning Model Parameters, Hyper-Parameters & Architectures

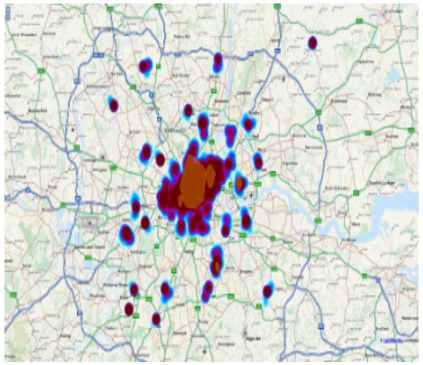
UNSUPERVISED LEARNING MODEL FOR CLUSTERING

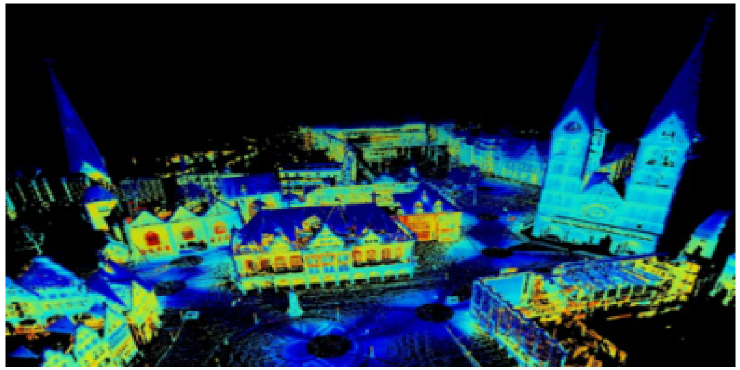
Example: Parallel & Scalable Density-based Spatial Clustering of Applications with Noise (DBSCAN)

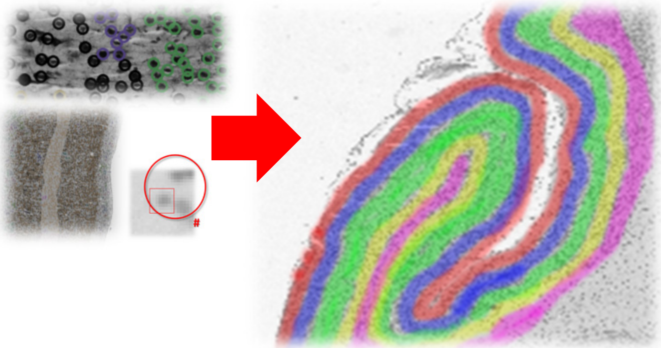


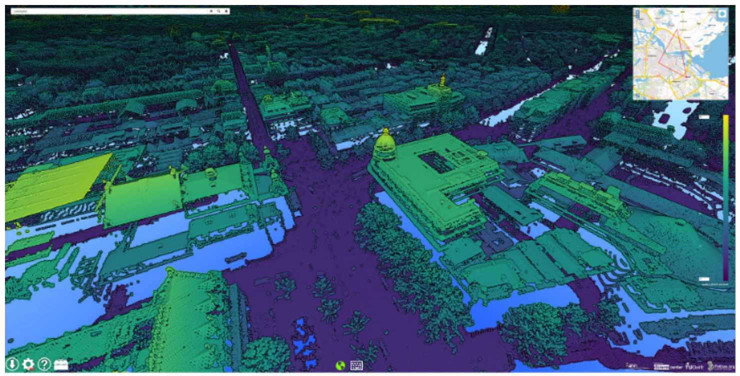










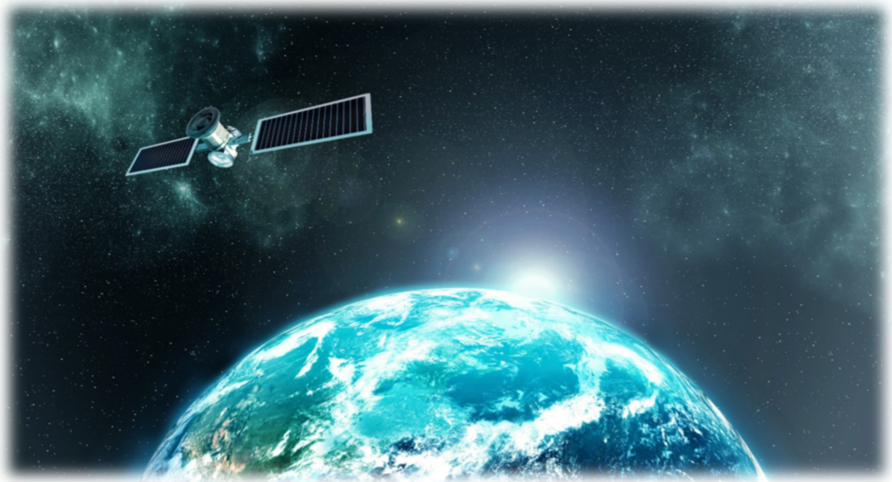


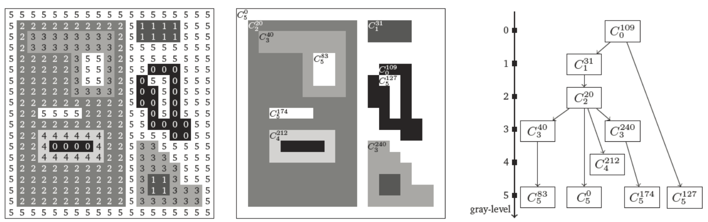
[15] M. Goetz and M. Riedel et al, Proceedings IEEE Supercomputing Conference, 2015

- Find right set of 2 parameters for application
- 1 Parameter: Minimum number of points
- 2 Parameter: Epsilon Neighbourhood
- Needs already HPC to be efficient in searching the right set of parameters, e.g. particle swarm optimization (evolutionary algorithm)

SUPERVISED LEARNING MODEL FOR CLASSIFICATION

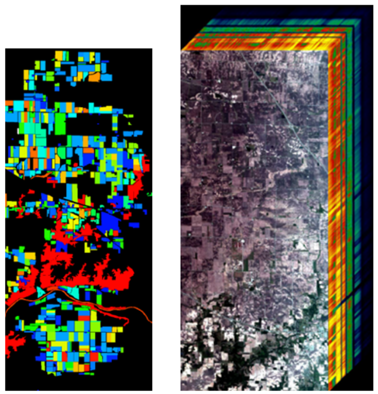
Example: Parallel and Scalable Support Vector Machine (SVM) – using Radial Basis Function (RBF) Kernel

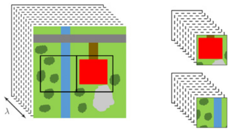


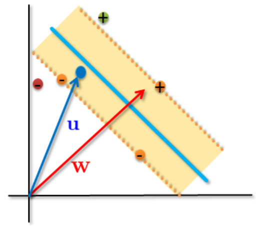


Parallel & Scalable Feature Engineering with Component Trees

[16] M. Goetz and M. Riedel et al., Journal of Transactions on Parallel and Distributed Systems, 2018







- **Find right set of 2 parameters for application**
- **1 Parameter: RBF Parameter Gamma**
- **2 Parameter: Cost of Error allowed for soft margin**
- **Needs already HPC to be efficient in searching the right set of parameters, e.g. gridsearch using partly also community experience in applications**

Scenario 'pre-processed data', 10xCV **serial**: accuracy (min)

γ/C	1	10	100	1000	10 000
2	48.90 (18.81)	65.01 (19.57)	73.21 (20.11)	75.55 (22.53)	74.42 (21.21)
4	57.53 (16.82)	70.74 (13.94)	75.94 (13.53)	76.04 (14.04)	74.06 (15.55)
8	64.18 (18.30)	74.45 (15.04)	**77.00** (14.41)	75.78 (14.65)	74.58 (14.92)
16	68.37 (23.21)	76.20 (21.88)	76.51 (20.69)	75.32 (19.60)	74.72 (19.66)
32	70.17 (34.45)	75.48 (34.76)	74.88 (34.05)	74.08 (34.03)	73.84 (38.78)

Scenario 'pre-processed data', 10xCV **parallel**: accuracy (min)

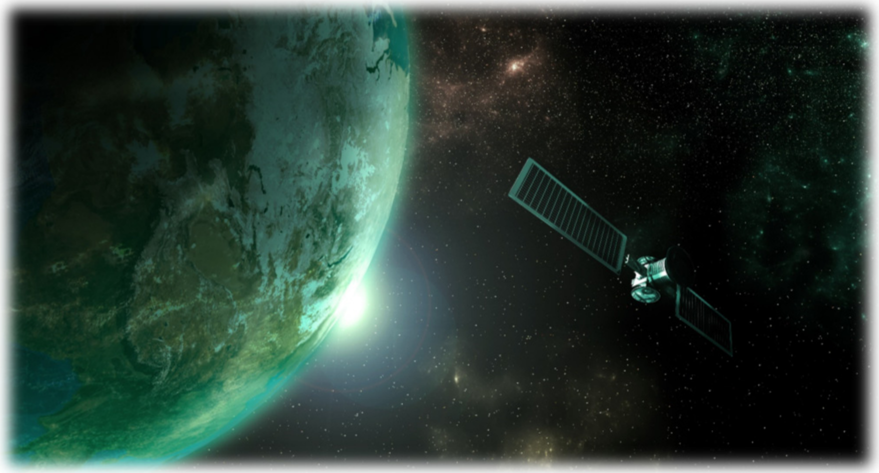
γ/C	1	10	100	1000	10 000
2	75.26 (1.02)	65.12 (1.03)	73.18 (1.33)	75.76 (2.35)	74.53 (4.40)
4	57.60 (1.03)	70.88 (1.02)	75.87 (1.03)	76.01 (1.33)	74.06 (2.35)
8	64.17 (1.02)	74.52 (1.03)	**77.02** (**1.02**)	75.79 (1.04)	74.42 (1.34)
16	68.57 (1.33)	76.07 (1.33)	76.40 (1.34)	75.26 (1.05)	74.53 (1.34)
32	70.21 (1.33)	75.38 (1.34)	74.69 (1.34)	73.91 (1.47)	73.73 (1.33)

First Result: best parameter set from 14.41 min to 1.02 min
Second Result: all parameter sets from ~9 hours to ~35 min

[17] G. Cavallaro and M. Riedel et al., Journal of Selected Topics in Applied Earth Observation and Remote Sensing, 2015

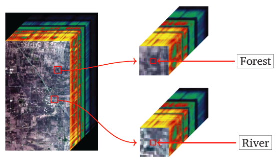
SUPERVISED LEARNING MODEL FOR CLASSIFICATION

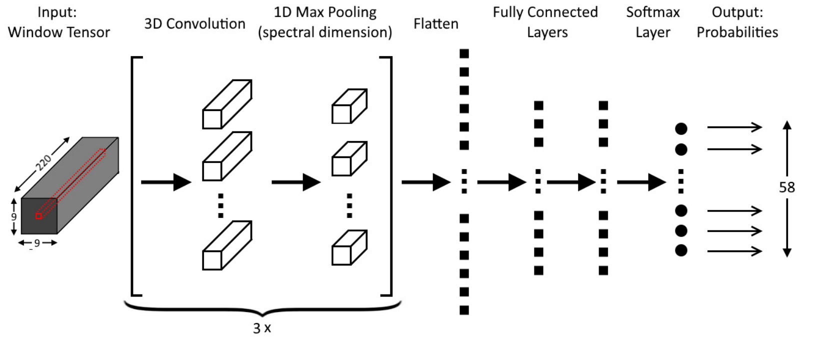
Example: Parallel & Scalable Deep Learning with Convolutional Neural Networks (CNNs)



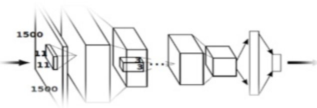
Using Convolutional Neural Networks (CNNs) with hyperspectral remote sensing image data

[18] J. Lange and M. Riedel et al., IGARSS Conference, 2018





Find Hyperparameters given rare labeled/annotated data in science (e.g. 36,000 vs. 14,197,122 images ImageNet)



Feature	Representation / Value
Conv. Layer Filters	48, 32, 32
Conv. Layer Filter size	(3,3,5), (3,3,5), (3,3,5)
Dense Layer Neurons	128, 128
Optimizer	SGD
Loss Function	mean squared error
Activation Functions	ReLU
Training Epochs	600
Batch Size	50
Learning Rate	1
Learning Rate Decay	5×10^{-6}

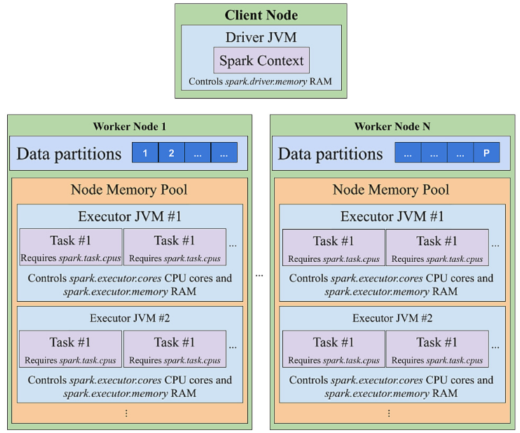
- What is the right optimization method?
- How many convolutional layers we need?
- How many neurons in dense layers?
- What is the right filter size?
- How do we train best?

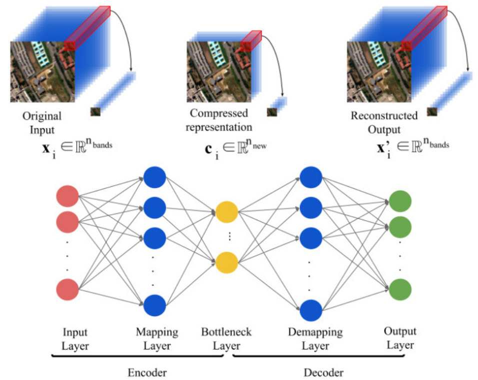
- **Find right set of hyper-parameters and the right neural network architecture is a manual time-consuming and error-prone process**
- **Needs urgently HPC, but a systematic and automated way is required as trying out all options of hyper-parameters and architectures is computationally infeasible**

SUPERVISED LEARNING MODEL FOR CLASSIFICATION

Example: Parallel & Scalable Deep Learning with Autoencoders

- Find right set of hyper-parameters and the right neural network architecture for autoencoder is a manual time-consuming and error-prone process
- Needs urgently HPC, but a systematic and automated way is required as trying out all options of hyper-parameters and architectures is computationally infeasible
- As resolutions of sensors becomes better and more data is available it is likely that the learning model will be increasingly complex in the future that in turn raises demands for automated architecture search and meta-learning approaches

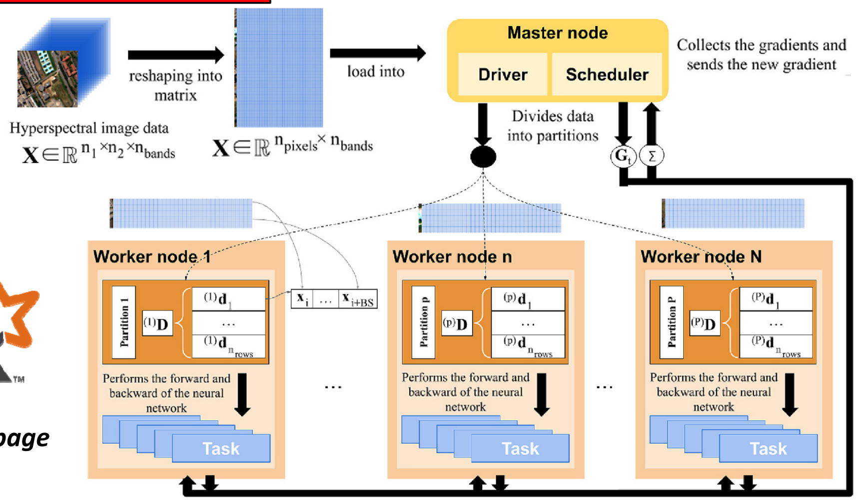




[19] J. Haut, G. Cavallaro and M. Riedel et al., IEEE Transactions on Geoscience and Remote Sensing, 2019

Using Autoencoder deep neural networks with Cloud computing & Apache Spark

[20] Apache Spark Web page

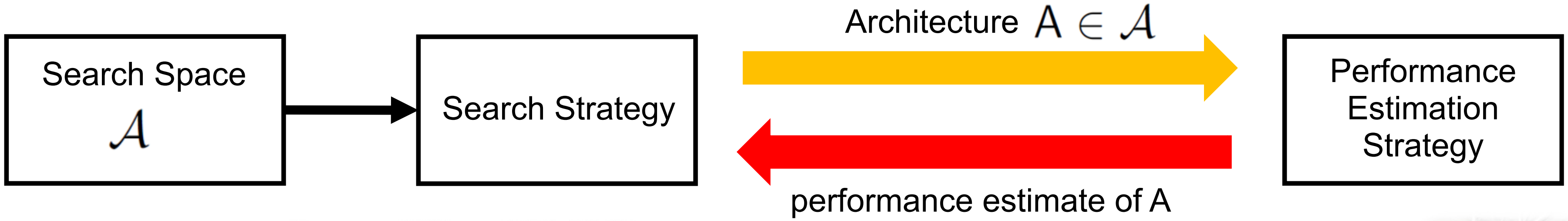


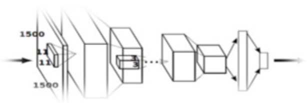
NEURAL ARCHITECTURE SEARCH (NAS)

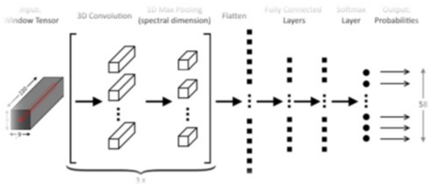
Fundamentals & Examples & Using Reinforcement Learning Techniques

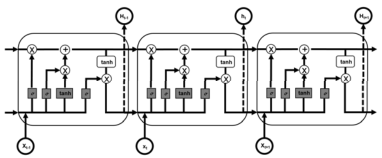
NEURAL ARCHITECTURE SEARCH (NAS) OVERVIEW

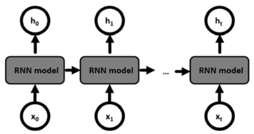
Methods for Automated Neural Network Architecture Engineering











[24] Neural Architecture Search Research

[21] T. Elsken et al., Neural Architecture Search: A Survey, Journal of Machine Learning Research, 2019

- **Employed neural networks architectures are often developed manually by human experts that is time-consuming and error-prone**
- **Deep learning success has been accompanied by a rising demand for architecture engineering, where increasingly more complex neural architectures are designed manually**
- **Neural Architecture Search (NAS) methods can be categorized in (a) search space, (b) search strategy, and (c) performance estimation strategy**
- **Automated Neural Architecture (NAS) search methods aim to solve this problem as a process of automating Architecture engineering**

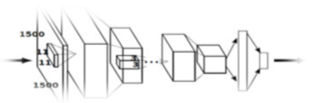
NEURAL ARCHITECTURE SEARCH

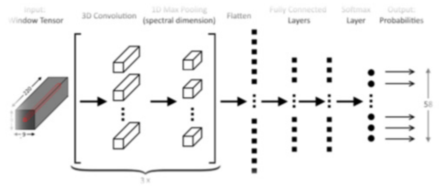
Understanding the Search Space & Using Prior Knowledge Example

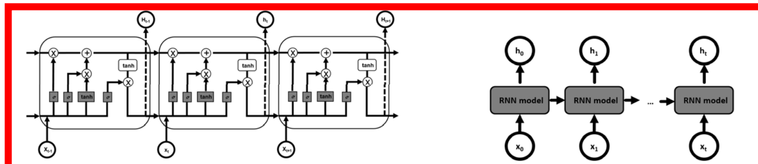
Search Space $\mathcal{A}$ → Search Strategy

Architecture $A \in \mathcal{A}$ → Performance Estimation Strategy

performance estimate of A

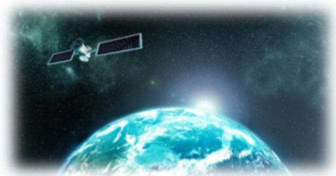


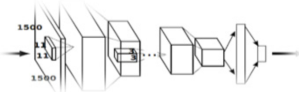


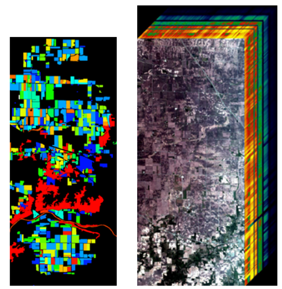


(using prior knowledge → are these useful?)

(using prior knowledge → typical property of neural network architecture for image analysis)





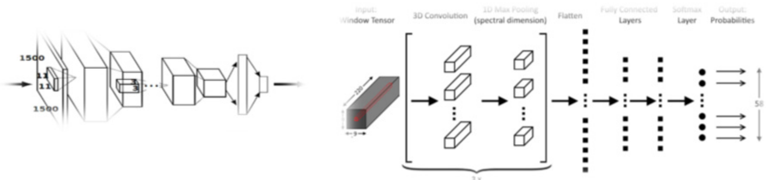


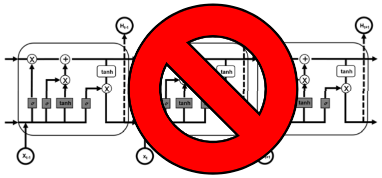
[21] T. Elsken et al., Neural Architecture Search: A Survey, Journal of Machine Learning Research, 2019

- **Search space defines which neural network architectures can be represented in principle**
- **Reduce the size of the search space to simplify the search by incorporating prior knowledge about typical properties of architectures**
- **Be aware that using prior knowledge also might introduce a human bias thus preventing finding novel neural network architectures**

NEURAL ARCHITECTURE SEARCH

Understanding the Search Space & Common Search Space Examples for CNNs

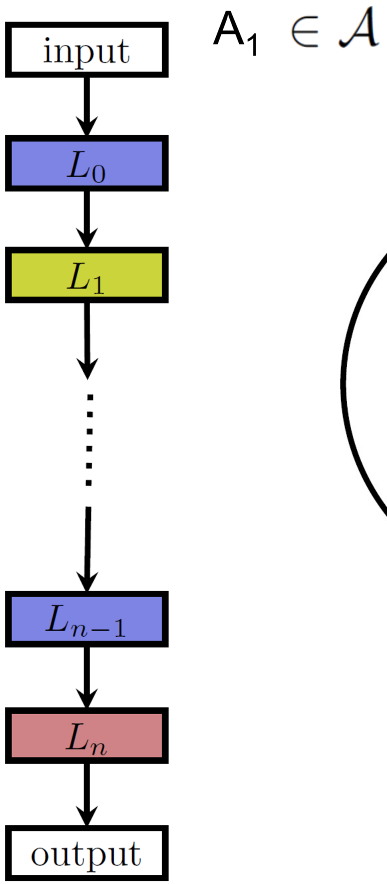


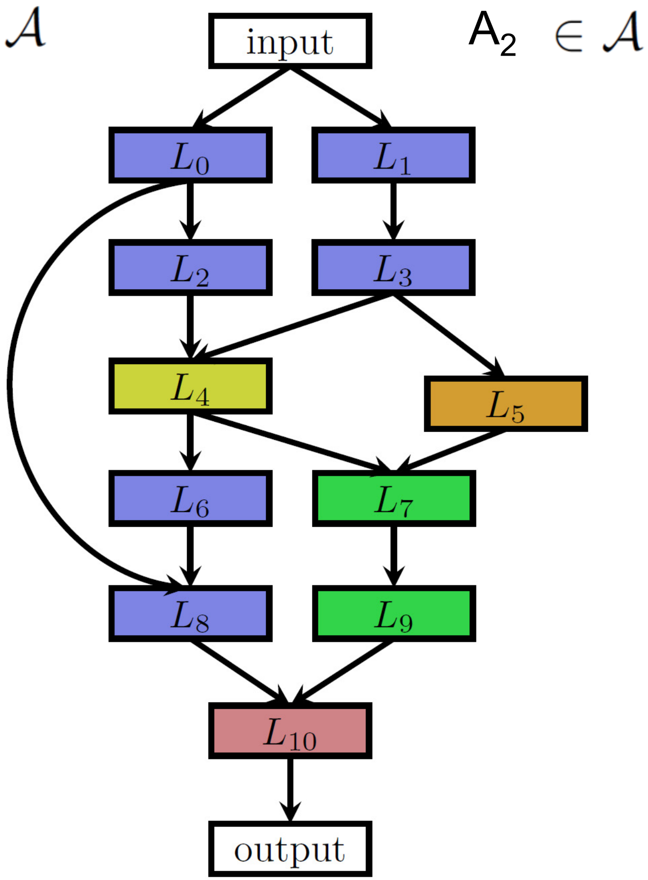


(using prior knowledge → e.g. here no time series)

[21] T. Elsken et al., Neural Architecture Search: A Survey, Journal of Machine Learning Research, 2019

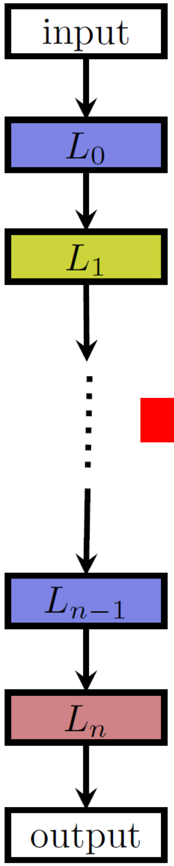
- **Each node in the graphs corresponds to a layer in a convolutional neural network (CNN), e.g. convolutional or pooling layer**
- **Different layer types are visualized by different colors**
- **Example Architecture A_1: Element of a chain-structured space**
- **Example Architecture A_2: element of a more complex search space with additional layer types and multiple branches and skip connections**

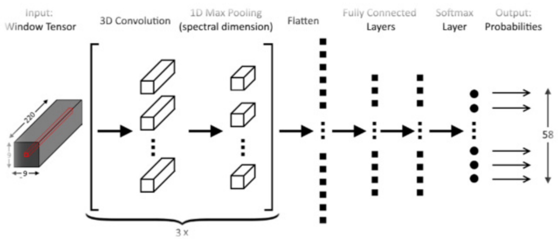


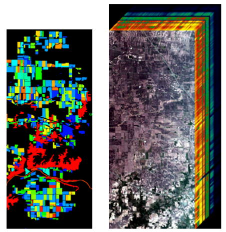


NEURAL ARCHITECTURE SEARCH

Understanding Layer Parameters & Complexity in Setting Parameter Values for Automated Search

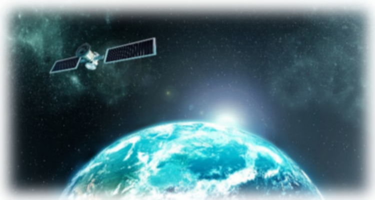






?

?



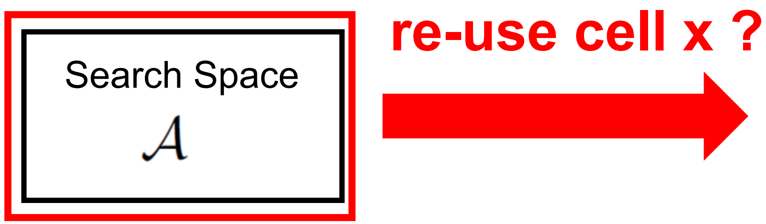
Layer Type	Layer Parameters	Parameter Values
Convolution (C)	$i \sim$ Layer depth $f \sim$ Receptive field size $\ell \sim$ Stride $d \sim$ # receptive fields $n \sim$ Representation size	< 12 Square. $\in \{1, 3, 5\}$ Square. Always equal to 1 $\in \{64, 128, 256, 512\}$ $\in \{(\infty, 8], (8, 4], (4, 1]\}$
Pooling (P)	$i \sim$ Layer depth $(f, \ell) \sim$ (Receptive field size, Strides) $n \sim$ Representation size	< 12 Square. $\in \{(5, 3), (3, 2), (2, 2)\}$ $\in \{(\infty, 8], (8, 4]$ and $(4, 1]\}$
Fully Connected (FC)	$i \sim$ Layer depth $n \sim$ # consecutive FC layers $d \sim$ # neurons	< 12 < 3 $\in \{512, 256, 128\}$
Termination State	$s \sim$ Previous State $t \sim$ Type	Global Avg. Pooling/Softmax

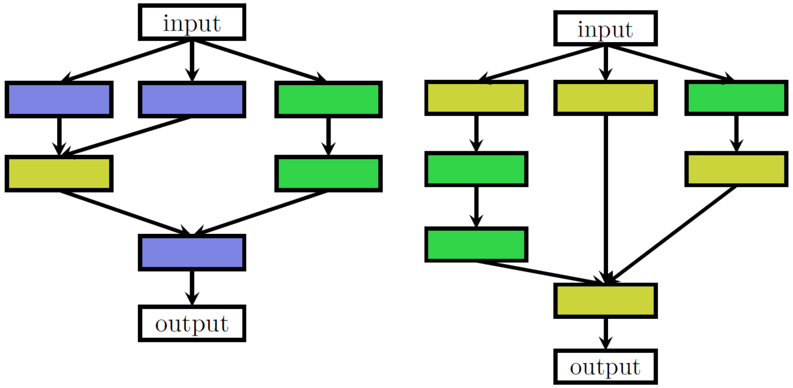
[21] T. Elsken et al., Neural Architecture Search: A Survey, Journal of Machine Learning Research, 2019

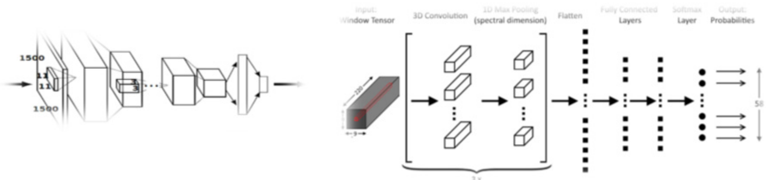
[25] B. Baker et al., 'Designing Neural Network Architectures using Reinforcement Learning', 2017

NEURAL ARCHITECTURE SEARCH

Understanding the Search Space & Cells instead of whole architectures



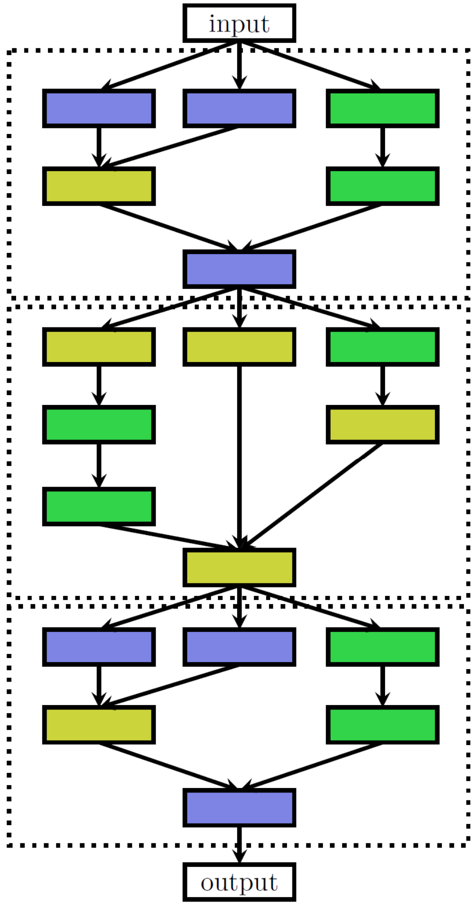




[21] T. Elsken et al., Neural Architecture Search:
A Survey, Journal of Machine Learning Research, 2019

- **Example illustrations of the cell search space with two different cells (not whole neural network architectures) → many design choices for the overall 'macro architecture' of the network**
- **Approach: a whole neural network architecture can be built by stacking the cells sequentially**
- **Complex approach: Cells can also be combined in a more complex approach: e.g., in multi-branch spaces by simply replacing layers**

- Simplicity: Search space is drastically reduced (less layers → ~ 7 x speed-up vs. full architectures in some recent work examples)
- Reusability: Architectures built from cells can more easily be transferred or adapted to other datasets
- Repitition: Creating architectures by repeating building blocks has proven a useful design (e.g. CNN with N x convolution, pooling layers, etc.)

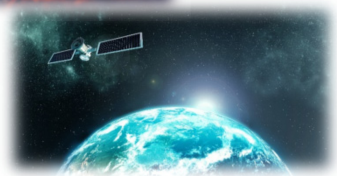


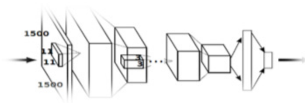
NEURAL ARCHITECTURE SEARCH

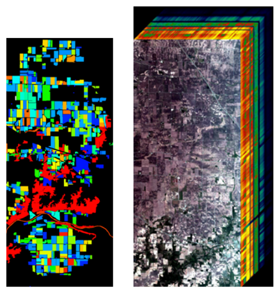
Computational Complexity & Number of Parameters of Known Architectures

Model	Depth	Parameters	Error rate (%)
Network in Network (Lin et al., 2013)	-	-	8.81
All-CNN (Springenberg et al., 2014)	-	-	7.25
Deeply Supervised Net (Lee et al., 2015)	-	-	7.97
Highway Network (Srivastava et al., 2015)	-	-	7.72
Scalable Bayesian Optimization (Snoek et al., 2015)	-	-	6.37
FractalNet (Larsson et al., 2016)	21	38.6M	5.22
with Dropout/Drop-path	21	38.6M	4.60
ResNet (He et al., 2016a)	110	1.7M	6.61
ResNet (reported by Huang et al. (2016c))	110	1.7M	6.41
ResNet with Stochastic Depth (Huang et al., 2016c)	110	1.7M	5.23
	1202	10.2M	4.91
Wide ResNet (Zagoruyko & Komodakis, 2016)	16	11.0M	4.81
	28	36.5M	4.17
ResNet (pre-activation) (He et al., 2016b)	164	1.7M	5.46
	1001	10.2M	4.62
DenseNet ($L = 40, k = 12$) Huang et al. (2016a)	40	1.0M	5.24
DenseNet($L = 100, k = 12$) Huang et al. (2016a)	100	7.0M	4.10
DenseNet ($L = 100, k = 24$) Huang et al. (2016a)	100	27.2M	3.74
DenseNet-BC ($L = 100, k = 40$) Huang et al. (2016b)	190	25.6M	3.46
Neural Architecture Search v1 no stride or pooling	15	4.2M	5.50
Neural Architecture Search v2 predicting strides	20	2.5M	6.01
Neural Architecture Search v3 max pooling	39	7.1M	4.47
Neural Architecture Search v3 max pooling + more filters	39	37.4M	3.65

- Further complexity in choosing HPC system features (e.g. Network Attached Memory, parallel file-systems, etc.)
- Computing Impact in various GPU architectures like Kepler, Pascal, Volta



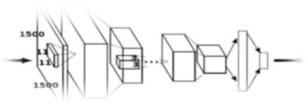


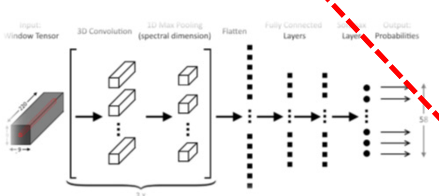


[23] B. Zoph et al., 'InstaNAS: Instance-aware Neural Architecture Search', 2018

NEURAL ARCHITECTURE SEARCH

Understanding the Search Strategy





[21] T. Elsken et al., Neural Architecture Search: A Survey, Journal of Machine Learning Research, 2019

- **Search strategy details how to explore the search space (which is often exponentially large or even unbounded)**
- **(a) it is desirable to find well-performing architectures quickly**
- **(b) premature convergence to a region of suboptimal architectures should be avoided**
- **Search strategy encompasses the classical exploration-exploitation trade-off: (a) vs (b)**

(different search strategies to explore the space of neural network architectures in addition to random search)

Reinforcement Learning

- Generation of an architecture can be considered as agent's action with space identical with neural architecture search space and rewards are used to guide the process

Gradient-Based Methods

- Optimize network weights & architecture by alternating gradient descent steps on training data for weights & validation data for architectural parameters

Bayesian Optimization

- Central idea of Bayesian optimization is to build a neural network architecture that can be updated and queried to drive optimization decisions, also assuming e.g. distributions (i.e., priors)

Evolutionary Methods

- Evolve a population of neural architecutres & in every evolution step a model from the population is sampled and serves as parent to generate offsprings & mutations

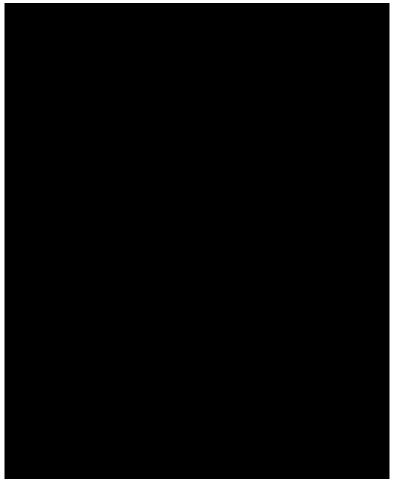
LEARNING APPROACHES – REVISITED FOR NAS

Reinforcement Learning

Learn to play games

- Each observation of the predictor measurement(s) has some associated response measurement:
  - Input $\mathbf{x} = x_1, \ldots, x_d$
  - Some output & grade of the output
  - Data $(\mathbf{x}_1), \ldots, (\mathbf{x}_N)$
- Goal: Learn through iterations
  - Guided by output grade: check learning and compare with grade
  - Challenge: Iterations may require lots of CPU time (e.g. backgammon playing rounds)
  - Challenge: Usually considered as a complicated learning approach but with applications in gaming

[7] Video source: Google DeepMind’s Deep Q-learning playing Atari Breakout

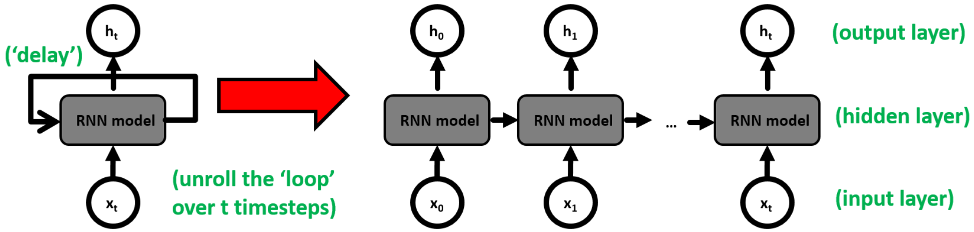


- **Reinforcement learning approaches learn through iterations using the grading output as guide**
- **Reinforcement learning approaches are used in playing game algorithms (e.g backgammon)**
- **Unupervised learning works with data = [input, some output, grade for this output]**

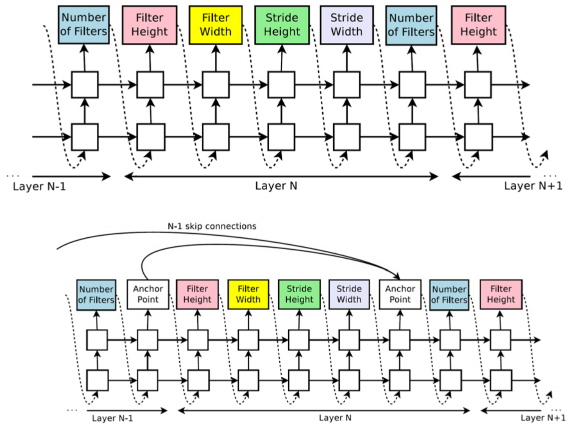
NEURAL ARCHITECTURE SEARCH

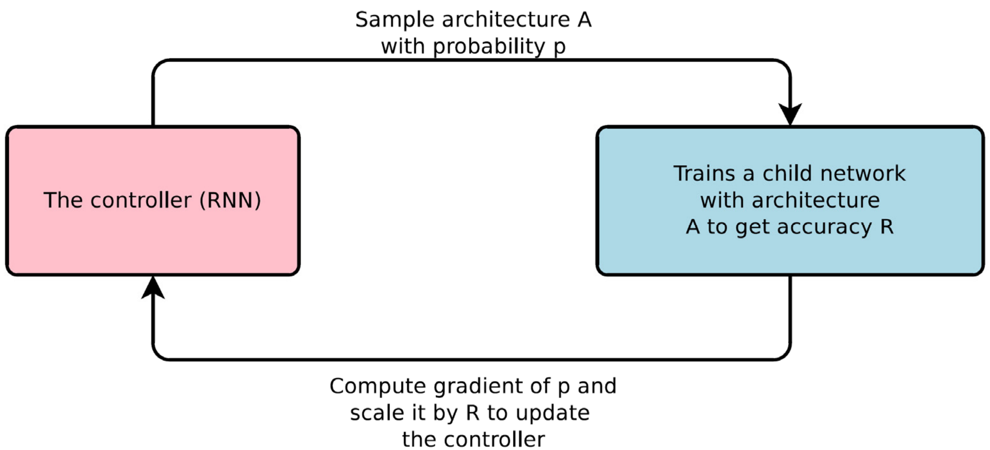
Using Reinforcement Learning Techniques - Understanding Controllers

- **Recurrent Neural Networks (RNNs) can handle sequence data**
- **Idea: CNNs are essentially a sequence of layers**
- **Controller (RNN) generete hyper-parameters as a sequence**



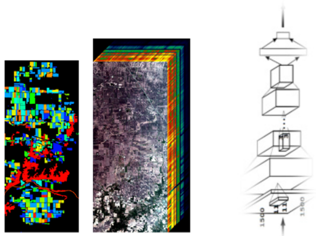
[23] B. Zoph et al., 'Neural Architecture Search with Reinforcement Learning', 2017

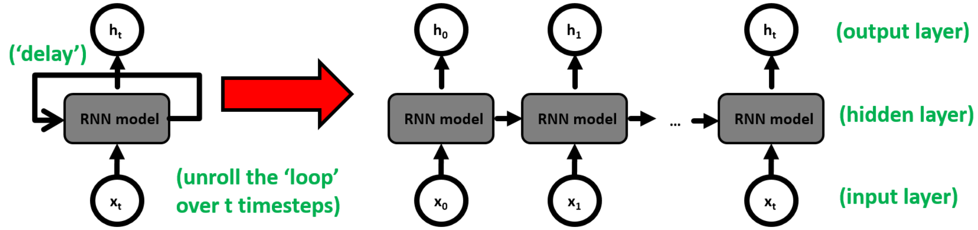


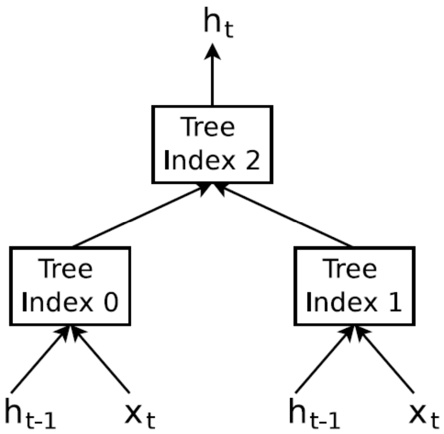


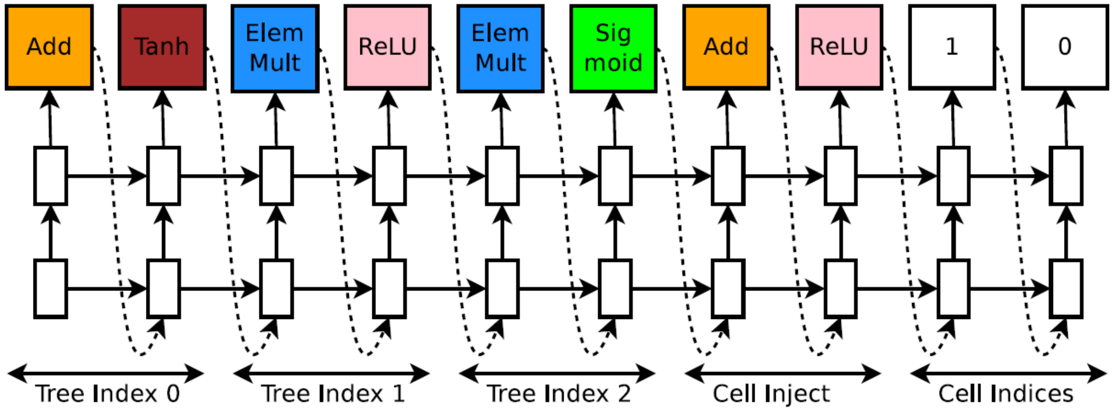
NEURAL ARCHITECTURE SEARCH

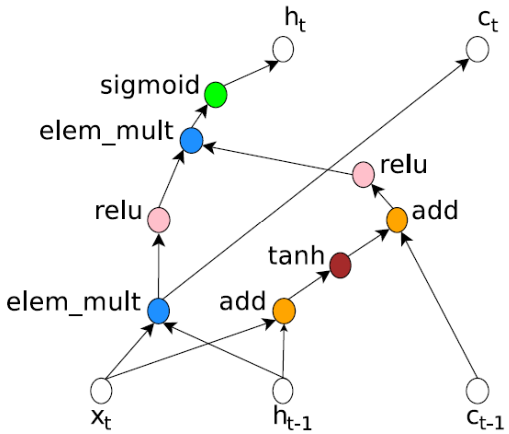
Using Reinforcement Learning Techniques – Understanding Neural Architecture vs. RNN Structure







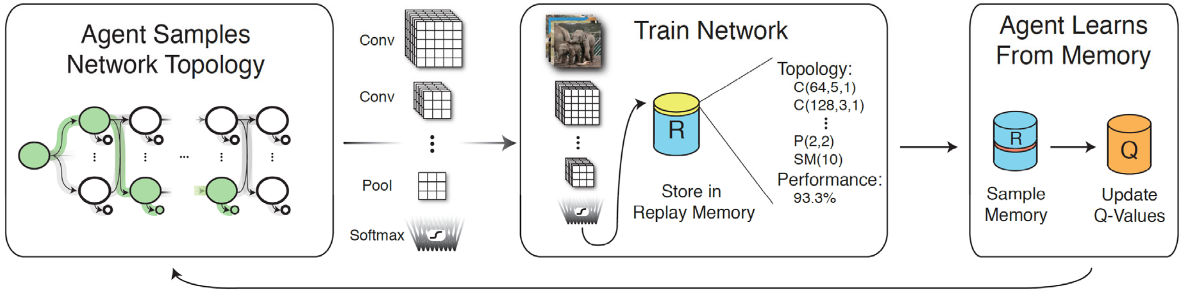




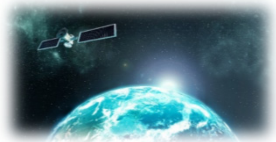
[23] B. Zoph et al., 'Neural Architecture Search with Reinforcement Learning', 2017

NEURAL ARCHITECTURE SEARCH

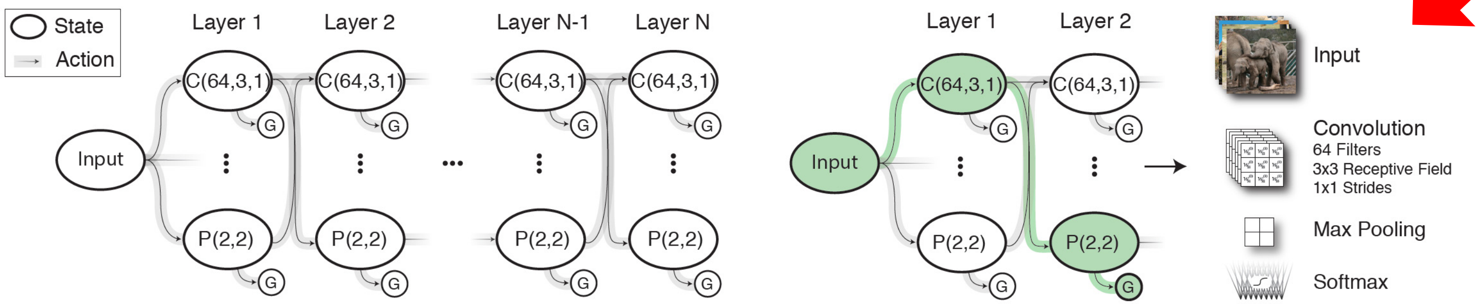
Using Reinforcement Learning Techniques – Understanding Agents

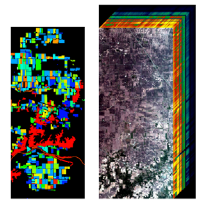


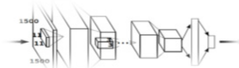
- **Generation of an architecture can be considered as agent‘s action with space identical with neural architecture search space and rewards are used to guide the process**



(peform distributed training of NAS)





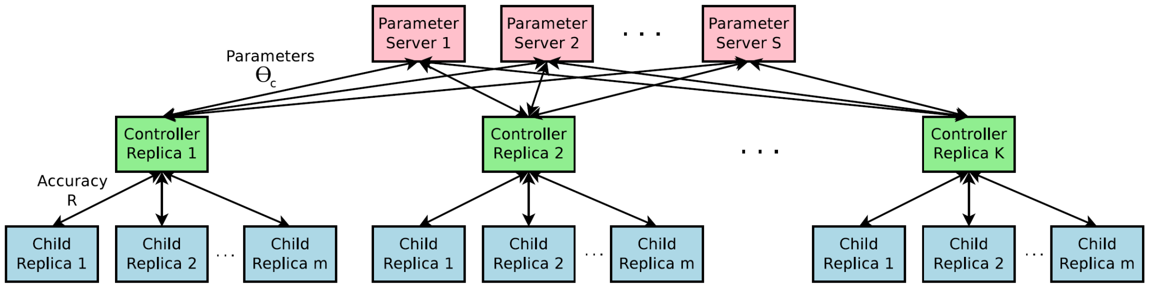


[25] B. Baker et al., ‘Designing Neural Network Architectures using Reinforcement Learning’, 2017

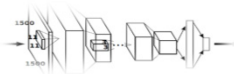
NEURAL ARCHITECTURE SEARCH

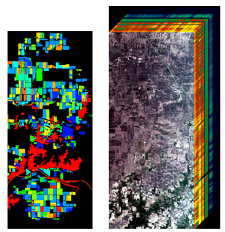
Using Reinforcement Learning Techniques – Understanding Distributed Training

- Distributed training for Neural Architecture Search can use a set of S parameter servers
- Parameter servers store and send parameters to K controller replicas
- Each controller replica then samples m architectures and run the multiple child models in parallel
- Accuracy of each child model is recorded to compute the gradients with respect to parameters
- In turn sent back to the parameter servers



(each controller replica
then samples m architectures)





[23] B. Zoph et al., 'Neural Architecture Search with Reinforcement Learning', 2017

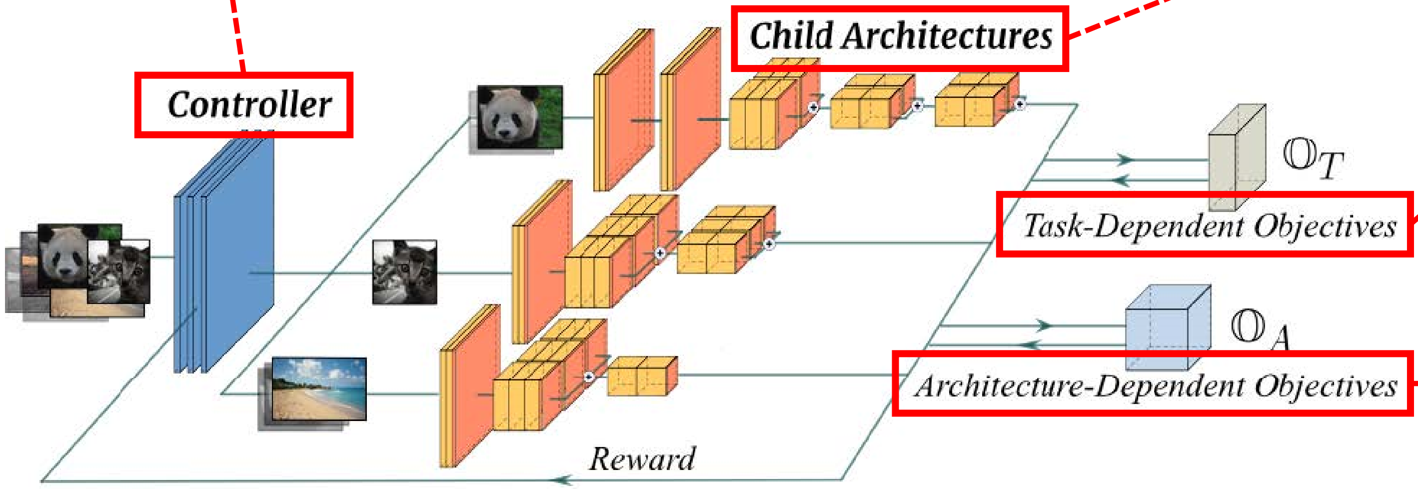
NEURAL ARCHITECTURE SEARCH

Using Reinforcement Learning Techniques and InstaNAS (multiple Neural Network Architecture Instances)

- Often a Recurrent Neural Network (RNN) technique that performs the agent steps

- Derived specific architectures that perform good for specific dataset samples

- E.g. what is the accuracy or error rate we obtain as metric to guide the search for specific architectures for specific dataset samples



- E.g. what is the latency of the network for a given dataset sample to guide the search for specific architectures that offer better latency by keeping accuracy(!)

[22] A.C. Cheng et al., 'InstaNAS: Instance-aware Neural Architecture Search', 2018

NEURAL ARCHITECTURE SEARCH

Understanding the Performance Estimation Strategy

Search Space $\mathcal{A}$ → Search Strategy

Architecture $A \in \mathcal{A}$ →

Performance Estimation Strategy

(accuracy? error-rate? latency?)

← performance estimate of A

(recent research focuses on developing methods that reduce the cost of these performance estimations)

Speed-up method	How are speed-ups achieved?
Lower fidelity estimates	Training time reduced by training for fewer epochs, on subset of data, downscaled models, downscaled data, ...
Learning Curve Extrapolation	Training time reduced as performance can be extrapolated after just a few epochs of training.
Weight Inheritance/ Network Morphisms	Instead of training models from scratch, they are warm-started by inheriting weights of, e.g., a parent model.
One-Shot Models/ Weight Sharing	Only the one-shot model needs to be trained; its weights are then shared across different architectures that are just subgraphs of the one-shot model.

[21] T. Elsken et al., Neural Architecture Search: A Survey, Journal of Machine Learning Research, 2019

- **Objective of neural architecture search is typically to find architectures that achieve high predictive performance on unseen data**
- **Performance Estimation refers to the process of estimating this performance and the usefulness of the architecture that has been 'found/explored'**
- **Simplest option: perform a standard training and validation of the architecture on data → unfortunately computationally expensive (even with HPC!)**
- **Simplest option thus limits the number of neural network architectures that can be explored or apply a number of speed-up methods**

SUMMARY

Neural Architecture Search (NAS) is a vibrant new research field

SUMMARY

Neural Architecture Search (NAS) is a vibrant new research field

- High Performance Computing & Machine Learning more intertwined today
  - GPUs can significantly speed-up the training of machine and deep learning models
- Recent Deep Learning models have tremendous success in many application areas
  - Pro: Manual feature engineering processes is often automated using automated feature learning
  - Contra: Employed neural network architectures are still often developed manually by human experts
  - Lessons learned: Manual time-consuming and error-prone process shifted to architecture engineering
- Automated Neural Architecture Search
  - Need since there is a growing number of fine-tuned architectures with a high number of hyper-parameters
  - Approaches differ in (a) search space, (b) search strategy, and (c) performance estimation strategy
  - Reinforcement Learning for NAS is just one of the possible search strategies, but a promising technique
  - Overlaps with meta-learning and hyper-parameter optimization approaches and subfield of AutoML

REFERENCES

REFERENCES (1)

- [1] Helmholtz Association Web Page,
  Online: https://www.helmholtz.de/en/
- [2] University of Iceland, School of Enginering and Natural Sciences,
  Online: https://english.hi.is/school_of_engineering_and_natural_sciences
- [3] GCS Web Page,
  Online: http://www.gauss-centre.eu/gauss-centre/EN/Home/home_node.html
- [4] HAICU Web Page,
  Online: http://www.haicu.de/
- [5] DEEP Projects Web Page,
  Online: http://www.deep-projects.eu/
- [6] Species Iris Group of North America Database,
  Online: http://www.signa.org
- [7] YouTube Video, Google DeepMind's Deep Q-Learning playing Atari Breakout,
  Online: https://www.youtube.com/watch?v=V1eYniJ0Rnk
- [8] Morris Riedel, 'Deep Learning - Using a Convolutional Neural Network', Invited YouTube Lecture, six lectures & exercises, University of Ghent, 2017
  Online: https://www.youtube.com/watch?v=gOL1_YIosYk&list=PLrmNhuZo9sgZUdaZ-f6OHK2yFW1kTS2qF

REFERENCES (2)

- [9] M. Riedel et al., 'Introduction to Deep Learning Models', JSC Tutorial, three days, JSC, 2019,
  Online: http://www.morrisriedel.de/introduction-to-deep-learning-models
- [10] H. Lee et al., 'Convolutional Deep Belief Networks for Scalable Unsupervised Learning of Hierarchical Representations',
  Online: http://doi.acm.org/10.1145/1553374.1553453
- [11] YouTube Video, 'Neural Network 3D Simulation',
  Online: https://www.youtube.com/watch?v=3JQ3hYko51Y
- [12] A. Rosebrock, 'Get off the deep learning bandwagon and get some perspective',
  Online: http://www.pyimagesearch.com/2014/06/09/get-deep-learning-bandwagon-get-perspective/
- [13] Big Data Tips – Big Data Mining & Machine Learning,
  Online: http://www.big-data.tips/
- [14] Deep Learning Start-Up Beispiel Deutschland,
  Online: https://soccerwatch.tv/
- [15] M. Goetz, C. Bodenstein, M. Riedel, 'HPDBSCAN – Highly Parallel DBSCAN', in proceedings of the ACM/IEEE International Conference for High Performance Computing, Networking, Storage, and Analysis (SC2015), Machine Learning in HPC Environments (MLHPC) Workshop, 2015,
  Online: http://dx.doi.org/10.1145/2834892.2834894
- [16] M. Goetz, G. Cavallaro, T. Geraud, M. Book, M. Riedel, 'Parallel Computation of Component Trees on Distributed Memory Machines', Journal of Transactions on Parallel and Distributed Systems, 2018

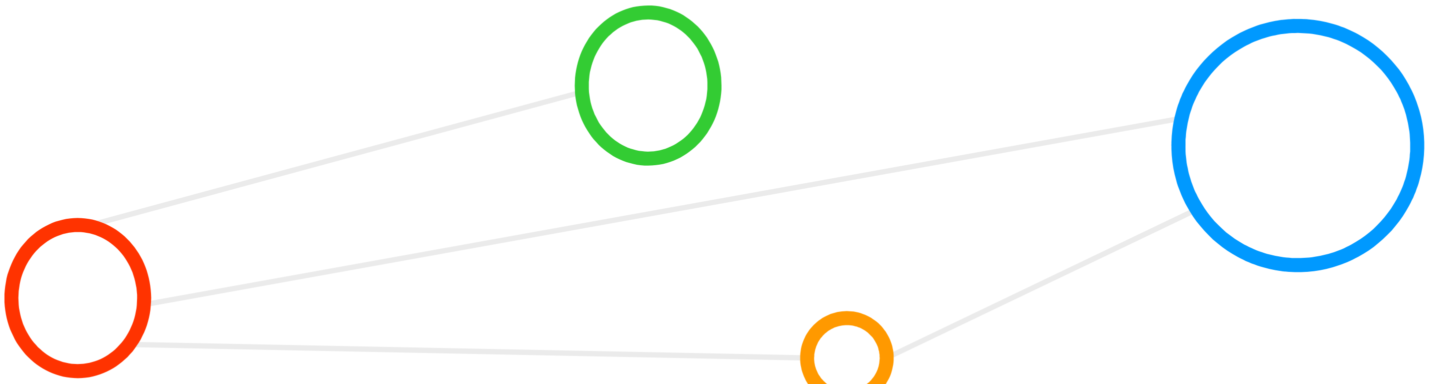
REFERENCES (3)

- [17] G. Cavallaro, M. Riedel, J.A. Benediktsson et al., 'On Understanding Big Data Impacts in Remotely Sensed Image Classification using Support Vector Machine Methods', IEEE Journal of Selected Topics in Applied Earth Observation and Remote Sensing, 2015, DOI: 10.1109/JSTARS.2015.2458855
- [18] J. Lange, G. Cavallaro, M. Goetz, E. Erlingsson, M. Riedel, 'The Influence of Sampling Methods on Pixel-Wise Hyperspectral Image Classification with 3D Convolutional Neural Networks', Proceedings of the IGARSS 2018 Conference
- [19] Haut, J.M., Gallardo, J.A., Paoletti, M.E., Cavallaro, G., Plaza, J., Plaza, A., Riedel, M.: Cloud Deep Networks for Hyperspectral Image Analysis, IEEE Transactions on Geoscience and Remote Sensing, PP(99):1-17, 2019, Online: https://doi.org/10.1109/TGRS.2019.2929731
- [20] Apache Spark,
  Online: https://spark.apache.org/
- [21] Elsken, T., Metzen, J.H., Hutter, F., Neural Architecture Search: A Survey, Journal of Machine Learning Research, 2019,
  Online: https://arxiv.org/pdf/1808.05377.pdf
- [22] Cheng, A.C, Lin, C.H., Juan, D.C., InstaNAS: Instance-aware Neural Architecture Search,
  Online: https://arxiv.org/abs/1811.10201
- [23] Zoph, B., Le, Q.V., 'Neural Architecture Search with Reinforcement Learning', 2017,
  Online: https://arxiv.org/pdf/1611.01578.pdf
- [24] Neural Architecture Search, Morris Riedel Webpage,
  Online: http://www.morrisriedel.de/neural-architecture-search

REFERENCES (4)

- [25] Baker, B., Gupta, O., Naik, N., Raskar, R., 'Designing Neural Network Architectures using Reinforcement Learning', Online: https://arxiv.org/abs/1611.02167
- [26] E. Erlingsson, G. Cavallaro, A. Galonska, M. Riedel, H. Neukirchen, 'Modular Supercomputing Design Supporting Machine Learning Applications', in conference proceedings of the 41st IEEE MIPRO 2018, May 21-25, 2018, Opatija, Croatia

ACKNOWLEDGEMENTS



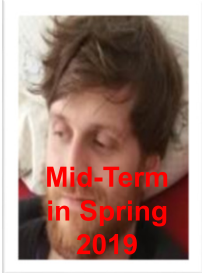
ACKNOWLEDGEMENTS

Previous & current members of the High Productivity Data Processing Research Group

PD Dr.
G. Cavallaro

Senior PhD
Student A.S. Memon

Senior PhD
Student M.S. Memon

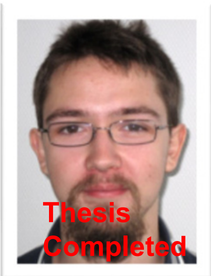


PhD Student
E. Erlingsson

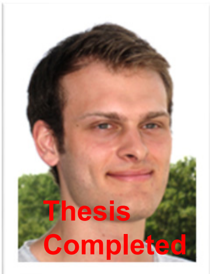
PhD Student
S. Bakarat

PhD Student
R. Sedona

Dr. M. Goetz
(now KIT)

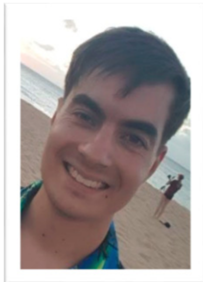


MSc M.
Richerzhagen
(now other division)



MSc
P. Glock
(now INM-1)

MSc
C. Bodenstein
(now
Soccerwatch.tv)



MSc Student
G.S. Guðmundsson
(Landsverkjun)

This research group has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 763558 (DEEP-EST EU Project)

THANKS

Talk shortly available under www.morrisriedel.de