Manifesto from Dagstuhl Perspectives Workshop 16152

Tensor Computing for Internet of Things

Edited by

Evrim Acar[1], Animashree Anandkumar[2], Lenore Mullin[3], Sebnem Rusitschka[4], and Volker Tresp[5]

1 **University of Copenhagen, DK**
evrim@life.ku.dk
2 **University of California – Irvine, US**
a.anandkumar@uci.edu
3 **University of Albany – SUNY, US**
lmullin@albany.edu
4 **Siemens AG – München, DE**
sebnem.rusitschka@siemens.com
5 **Siemens AG – München, DE**
volker.tresp@siemens.com

Abstract

"The fundamental laws necessary for the mathematical treatment of large part of physics and the whole of chemistry are thus completely known, and the difficulty lies only in the fact that application of these laws leads to equations that are too complex to be solved." – Dirac 1929

The digital world of Internet of Things (IoT) will provide a high-resolution depiction of our physical world through measurements and other data - even high-definition "video," if you consider streaming data frames coming from a myriad of sensors embedded in everything we use. This depiction will have captured our interactions with the physical world and the interactions of digitally enhanced machines and devices. Tensors, as generalizations of vectors and matrices, provide a natural and scalable framework for handling data with such inherent structures and complex dependencies. Scalable tensor methods have attracted considerable amount of attention, with successes in a series of learning tasks, such as learning latent variable models, relational learning, spatio-temporal forecasting as well as training and compression of deep neural networks.

In a Dagstuhl Perspectives Workshop on Tensor Computing for IoT, we validated the fundamental suitability of tensor methods for handling the massive amounts of data coming from connected cyber-physical systems (CPS). The multidisciplinary discourse among academics, industrial researchers and practitioners in the IoT/CPS domain and in the field of machine learning and tensor methods, exposed open issues that need to be addressed to reap value from the technological opportunity. This Manifesto summarizes the immediate action fields for advancement: IoT Tensor Data Benchmarks, Tensor Tools for IoT, and the evolution of a Knowledge Hub. The activities will also be channeled to create best practices and a common tensor language across the disciplines.

In a not so distant future, basic infrastructures for living will be mainly data-driven, automated by digitally enhanced devices and machines. The tools and frameworks used to engineer such systems will ensure production-ready machine learning code which utilizes tensor-based, hence better interpretable, models and runs on distributed, decentralized, and embedded computing resources in a robust and reliable way. We conclude the manifesto with a strategy how to move towards this vision with concrete steps in the identified action fields.

Perspectives Workshop April 10–13, 2016 – http://www.dagstuhl.de/16152

Tensor Computing for Internet of Things, *Dagstuhl Manifestos*, Vol. 7, Issue 1, pp. 52–68
Editors: Evrim Acar, Animashree Anandkumar, Lenore Mullin, Sebnem Rusitschka, and Volker Tresp
Dagstuhl Manifestos
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

2012 ACM Subject Classification Computing methodologies → Machine learning, Computer systems organization → Embedded and cyber-physical systems, Hardware → Emerging tools and methodologies

Keywords and phrases Distributed Systems, Real-time and embedded systems, Signal processing systems, Learning, Multiagent systems

Digital Object Identifier 10.4230/DagMan.7.1.52

Executive Summary

Cyber-physical systems (CPS), or the more consumerized Internet of Things (IoT) is a new wave of embedding affordable computing and communication into our previously mechanized world to enable for example adaptive energy efficient buildings fueled by renewable energy sources and connected to smart power grids, factory automation that yields flexible manufacturing and zero down-time connected to adaptive global supply chains, multi-modal on-demand public transportation facilitated by car-sharing and even self-driving cars in the near future.

After years of industrial research, we can pinpoint with confidence that all of the above scenarios of IoT have the following common requirements emerging from a set of common characteristics, i.e., they all require the extraction of actionable information for near real-time automation from multidimensional, spatio-temporal data. This data is only partially stochastic, as much as humans are involved as the users and operators. But mostly the data comes from a human-engineered, but mechanically, increasingly digitally, automated network such as electricity networks, supply chains/networks, transportation networks – commonly referred to as flow networks. The digitalization of these flow networks is what we refer to as CPS or IoT. Such digitalization includes ever more precise sensors, cheaper embedded computing, ubiquitous connectivity, combined with massive amounts of historical data and easy-to-spawn compute clusters in global data centers.

In April 2016, Dagstuhl hosted a Perspectives Workshop on Tensor Computing for the Internet of Things by bringing together academic researchers from the tensor community, distributed computing and machine learning as well as industrial researchers and practitioners from the IoT/CPS domain. The goal of the workshop was to explore the tensor representations and tensor computing as the basis for the machine learning solutions needed to turn massive amounts of IoT/CPS data into useful and actionable information. Tensors, as generalizations of vectors and matrices, provide a natural representation for data with many axes of variation, e.g., multidimensional, spatio-temporal data. The workshop validated the suitability of tensor-based computation for handling data coming from IoT/CPS and concluded with a vision that tensors would be a crucial part of a bigger computational machinery supporting the domain experts of IoT/CPS in the near future and supporting the machines and devices in IoT/CPS in the long term. This manifesto discusses the immediate action areas, i.e., IoT Tensor Benchmark Data & Infrastructure, Tensor Tools for IoT, and Tensor Learn – a knowledge hub, to move towards this vision, and concludes with strategic steps to be taken within the three action areas.

The manifesto is intended for government and industry funding agencies as well as academic and industrial researchers. The manifesto will draw the attention of funding agencies to the open issues needed to be addressed for utilizing the massive amounts of IoT/CPS data from a data science perspective by pointing to tensor computing as a crucial tool. The manifesto will also address to academic and industrial researchers by emphasizing

the open research directions in tensor computing as well as in its use for production-level development and deployment in IoT/CPS.

Table of Contents

1 Introduction

In April 2016, Dagstuhl hosted a Perspectives Workshop on Tensor Computing for the Internet of Things [2]. The prior year, industrial researchers had formulated the challenges of gaining insights from multi-dimensional sensory data coming from large-scale connected energy, transportation networks or manufacturing systems. The sheer amount of streaming multi-aspect data was prompting us to look for the most suitable techniques from the machine learning community: multi-way data analysis.

The workshop focused on the Internet of Things (IoT), i.e. devices, which have the capability to sense, communicate, and even control or interact with their environments. These devices are increasingly becoming parts of complex, dynamic, and distributed systems of electricity or mobility networks, hence our daily lives. Various sensors enable these devices to capture multiple aspects of their surroundings in real-time. For example, phasor measurement units capture transient dynamics and evolving disturbances in the power system in high-resolution, in a synchronized manner, and in real-time. Another example is traffic networks, where a car today can deliver about 250 GB of data per hour from connected electronics such as weather sensors within the car, parking cameras and radars. Experts estimate that the IoT will consist of almost 50 billion objects by 2020 [36], which will trigger the Era of Exascale computing necessitating the management of heat and energy of computing in concert with more and more complex processor/network/memory hierarchies of sensors and embedded computers in distributed systems.

Crucial for the extraction of relevant information is the format in which the raw data from such systems is represented. Crucial for the practicability of information extraction in IoT is which and how operations are used guaranteeing various attributes of resource use and management. Tensors can be viewed as both multidimensional data structures and as multilinear operators. The goal of the workshop was to explore tensor representations and computing as the basis for machine learning solutions for the IoT. Tensors are algebraic objects which describe linear and multilinear relationships, and can be represented as multidimensional arrays. They often provide a natural and compact representation for multidimensional data. In the recent years, tensor and machine learning communities - mainly active in the data-rich domains such as neuroscience, social network analysis, chemometrics, knowledge graphs etc. - have provided a solid research infrastructure, reaching from the efficient routines for tensor calculus to methods of multi-way data analysis, i.e., tensor decompositions, to methods for consistent and efficient estimation of parameters of the probabilistic models.

Some tensor-based models have the intriguing characteristic that if there is a good match between the model and the underlying structure in the data, the models are much better interpretable than alternative techniques. Their interpretability is an essential feature for the machine learning techniques to gain acceptance in the rather engineering heavy fields of automation and control of cyber-physical systems (CPS). Many of these CPS show intrinsically multilinear behavior, which is appropriately modeled by tensor methods and tools for controller design can use these models. The calibration of sensors delivering data and the higher resolution of measured data will have an additional impact on the interpretability of models.

Various presentations on tensor methods by established researchers at the workshop from different application domains assured us that tensor methods are reaching a maturity tipping point. However, knowledge of usage characteristics of tensor models is scattered. Discussions of the currently independent perspectives on the usage of tensor methods showed a potential for convergence, which we want to leverage through the action areas we are describing in this Dagstuhl Manifesto. During our discussions based on the presentations of the IoT industrial researchers, it quickly became clear that we would need benchmark challenges for cyber-physical systems and benchmark data in order to be able to replicate the successes in machine learning for object recognition and natural language understanding.

The tensor computing community will equally benefit from the new types of data, requirements, and multi-aspect characteristics of IoT, which can lead to techniques that increase success rates of previous applications of tensor methods, as was the case with the challenges of social network data analysis leading to better tensor models/algorithms that can analyze data sets with missing entries, now used in many other fields in addition to social network analysis. Additionally, as opposed to standardized machine learning techniques, tensor computing currently lacks a common language and the homogeneity to flexibly exchange models. Hence, a hub platform bringing data and domain knowledge of cyber-physical systems together with various practitioners of tensor computing would enhance increasing coherence of terms, best practices in data acquisition and structuring methods as well as model benchmarking, cataloging, and exchange of methods.

In the following the Manifesto describes the three action fields of Benchmarks, Tools, and Knowledge Hub, when put together, will make tensors a crucial part of a bigger computational machinery. This machinery will enable first domain experts of IoT/CPS and at a future time also the machines and devices in IoT/CPS to create efficient and sustainable infrastructures for life. We conclude the Manifesto with this vision and a strategy how to move into the right direction now.

2 IoT Tensor Benchmark Data & Infrastructure

Availability of benchmark data has been one of the reasons behind the recent advances in machine learning, e.g. large collections of high-resolution imagery for image recognition in computer vision tasks - or large corpus of written and spoken text for applications that need natural language processing. Although the special - multi-relational - structure of data is at the heart of tensor decompositions, there are no dedicated benchmark tensor data sets. Benchmark data typically is chosen to shed light on an algorithm's critical performance aspects and compare it to other algorithms. Well-known problems with this approach are the problem-specificity and that the computational performance and scalability remain still untested for larger real-world problem data sets.

IoT may indeed bring with it the much needed tensor data in a benchmarkable environment for tensor computing. Until now most effective data sets are known to be from chemometrics, telecommunications networks, neuroscience and social networks. Chemometrics data mainly represent a "closed" environment, e.g. the make-up of a fluid consisting of multiple components with different spectra. Application of tensor decompositions allows for interpretable factorization and analysis results in such closed environments. Cyber-physical systems are made up of such closed environments, which connected to each other build wider networks/systems. Examples are IoT data sets on home energy usage, in which the multi-aspect measurements of power parameters at the home breaker box capture the varying characteristic spectra of all the electrical appliances contributing to a home's energy usage, a "closed" environment. Hence, the application of tensor decompositions to the problem of so called non-intrusive load monitoring should yield similar interpretable analytic results as in chemometrics applications. Furthermore a local power grid network consists of multiple such closed environments, and links to other local grids to make up the bigger power distribution and transmission system.

Similar connectivist view of other CPS domains - such as in manufacturing with factories and supply chains, or in mobility with connected vehicles and multimodal transportation systems, etc. - and the promising nature of tensor methods motivates researchers and potential data providers to organize so called "IoT Tensor Data Challenges," which will

- accommodate larger data sets on real-world problems of IoT/CPS,
- curate for high-accuracy and high-resolution sensor data,
- from a "closed" environment such that factorization yields interpretable results, as well as
- have the potential to capture larger networks in the data

through the inherent connectivity of the IoT/CPS data challenge. The issues with current benchmark data collections should be addressed by standardizing the "IoT Tensor Data Benchmark Infrastructure" with following research & development aspects:

- The infrastructure should enable users to filter problems based on technical similarity, e.g. spatio-temporal problems, multi-class predictions etc. The infrastructure should also enable to browse across others' implementation of algorithms and compare effectively.
- In addition to prediction accuracy, key performance indicators of benchmarking for IoT/CPS applications are interpretability, computational resource consumption, robustness in stream processing and potentially in highly distributed settings.
- The interface to the infrastructure should also enable users to access metadata and analyze metadata to understand how the algorithms perform, e.g. computation cost per training, per prediction, etc.
- Additionally, the interface should enable users to easily understand how different tensor models and algorithms perform in different scenarios.

The organization of IoT Tensor Data Challenges will require coordinated efforts of this community and their extended network. Whereas especially data from industrial partners will be handled with care, and confidentially if required, as to lower the barriers to providing data for the challenge. The design and development of the IoT Tensor Data Benchmark Infrastructure requires an open and iterative approach, which will be improved with every data challenge.

3 Tensor Tools for IoT

Data in many disciplines contains more than two axes of variation, e.g., spatial, temporal and spectral dimensions of multi-channel electroencephalography (EEG) signals represented in both time and frequency domains [3, 27], and can be represented as a multi-way array, also referred to as a higher-order tensor. Exploiting the low-rank structure and capturing the underlying patterns in such higher-order data sets are crucial in some domains in order to extract information from complex data sets. Therefore, tensor factorizations, i.e., extensions of matrix factorizations to multi-way data, have proved useful in a variety of applications, in particular, in chemometrics, neuroscience, signal processing and data mining [4, 23, 34, 32].

In this section we discuss and identify open research questions in two parts: (a) regarding models and algorithms and (b) regarding development for and deployment of these models and algorithms in IoT/CPS.

Models and algortihms. Tensor factorizations have become a popular data mining tool in the last decade. Inter-disciplinary conferences of the tensor community such as TRICAP (Three-way Methods in Chemistry and Psychology) and TDA (Tensor Decompositions and Applications) as well as workshops sponsored by AIM (American Institute of Mathematics) and NSF (National Science Foundation) have played a key role in promoting and advancing the field by bringing experts from different fields together to identify and solve issues in tensor computing. Significant efforts have been invested in developing tensor factorization models, building algorithms and finding the right tensor models for applications of interest. Among the variety of tensor factorization approaches, the CANDECOMP/PARAFAC (CP) model [17, 11] has proved useful in applications, where the goal is to capture the underlying factors uniquely and use them for interpretation. As a result of its uniqueness properties leading to easily interpretable models, CP has been successfully used in neuroscience, chemometrics, social network analysis and signal processing applications. The CP model has strong assumptions about the underlying structure of the multi-way data, i.e., each slice of the tensor should have the same factors but in different proportions. If there is a good match between the data and the CP model, it is possible to summarize the data in a compact, unique and meaningful way. If the data does not follow a CP model, more flexible tensor factorization models such as a Tucker model [37] can be used for exploratory data analysis. Also, in particular, when the goal is data compression, Tucker-based approaches have proved to be effective. In addition to CP and Tucker models, there are many tensor models (see surveys/books on tensor factorizations [4, 23, 34, 16, 32]), which may be preferred depending on the goal of the application and the underlying structure of the data sets of interest.

While the analysis of data emerging from IoT/CPS applications will benefit from the expertise of the tensor community, new types of data and requirements of the applications will also call for further developments in tensor computing. The CPS/IoT systems are real-time, distributed, networked, and show dynamic behavior. The data coming from the sensors embedded into these systems is streaming, noisy, both high-frequency and high-volume, both sparse and dense. We have identified the following open problems in tensor computing as the challenges to primarily focus on in order to make tensor computations effective tools in IoT/CPS applications:

- Developing efficient streaming tensor models that can analyze real-time data,
- Building algorithms scalable to high-volume data (for both sparse and dense),
- Developing efficient distributed models and algorithms,
- Automating the building blocks of tensor modeling, e.g., model order selection, model selection, to decrease expert inputs in the analysis of IoT/CPS data,
- Uncertainty quantification of model parameters for tensor factorizations,
- Introducing new visualization methods in order to increase the interpretability of tensor factorizations,
- Developing data fusion models and their streaming versions that can jointly analyze coupled heterogeneous data sets, i.e., data sets in the form of matrices and higher-order tensors,
- Building tensor factorization models that can incorporate prior knowledge such as the connectivity structure (topology) of IoT systems,
- Forming a common tensor computing language to facilitate the exchange of expertise.

Development and Deployment. During the workshop we also had the opportunity to exchange on trends in tensor tools and emerging frameworks, which focus on development and deployment support for production-level code. Tensor tools have come a long way since the first version of Tensor Toolbox for Matlab over a decade ago [6].

Whilst new tools for Matlab have emerged with more focus on modularity, documentation and getting users from other research fields up to speed on using tensors [38], more specialized implementations such as Tensor Trains [31] or a distributed version of Tucker computations [22] are increasingly being shared on github as open source. Open source does speed up research immensely since code and papers are instantly accessible to investigate, learn from, and build upon. We believe that this trend will also assist in disseminating and in creating the common tensor computing language across disciplines. Especially, when data scientists will start adopting and porting some of these tensor-based models and algorithms for use in their favorite programming language and numerical computation libraries.

Matlab is popular with mathematicians and scientists. However, data scientists and machine learning researchers rarely use Matlab. Instead for the longest time Theano [9], a numerical computation library in Python, has been the most popular open source framework. Theano's focus has been deep learning and efficient computations utilizing GPUs. Since November 2015, Google open sourced their numerical computation library called Tensorflow [1], which since then gained considerably in popularity. Tensorflow has Python bindings, whilst the core is written in C++. Tensorflow aims to enable the creation of maintainable production-ready code, which runs on distributed machines, hence highly targeted towards industrial data scientists and applications which deal with massive amounts of data that no single analytics machine can handle effectively. In this latter category of industry-focused tools another framework worth mentioning exists: Deeplearning4J [21]. Deeplearning4J is also a distributed deep learning framework suitable for major companies and large government organizations, which to date still heavily rely on Java or a JVM-based system. Both Tensorflow and Deeplearning4J are designed for use with distributed data management and processing systems such as open source Hadoop and Spark [35] or in the case of Tensorflow also naturally with Google's proprietary cluster scheduling system called Borg [15].

Researchers in the intersection of tensor computing and machine learning have been implementing and open sourcing tensor methods in Python [28] for use in Python projects, or in Scala [18] for use with Spark, or in Julia [5], a language designed to address the needs of high-performance numerical analysis and computational science while also being effective for general-purpose programming, just to name a few. This is a typical sign of the search for a dominant design in this newly converging field of machine learning and tensor computing. A potential research & development direction is to create an abstraction layer. A well-designed API would allow to build tensor-based learning models by clipping together high-level building blocks of tensor decompositions and similar methods. The abstraction layer would be placed on top of numerical computation libraries like Tensorflow or Deeplearning4J etc. TensorLab has such a layer built-in but currently it is only on top of MATLAB.

At this point it is hard to predict, which languages and frameworks will prevail after more experience has been gained in the intersection of machine learning and tensor computing. Yet, the domain of IoT/CPS additionally demands the code deployment to be lightweight and the programming language to be robust and efficient for embedded processors. Java is inherently cross-platform, there is an embedded variant and OSGi suitable for some IoT application classes. However, in CPS domains where the insights gained from tensor decompositions shall translate into controller actions and other near real-time optimization, performance will be the crucial factor. Whilst C++ as a systems programming language seems to be the natural choice, it must be noted that C++ is difficult to optimize and maintain.

The skill set that can break down tensor-based machine learning models and algorithms - even if only for inference - into reliable, high-performant, embedded code is very rare. This realization is a definitive call for developing of frameworks that support the developers. Tensorflow is the only framework, at the time of this writing, which is used in production and supports direct deployment of trained models in embedded and mobile devices [39]. During research for the compilation of the Manifesto, we also found a new machine learning framework called Leaf [26] written in Rust, which is an up and coming safe and parallel systems programming language that is easy to write and deploy. Interestingly, the initial performance benchmarks affirm our discussions that Tensorflow may be too memory-intensive for embedded environments. Ironically, Leaf's development concluded in May 2016 due to the rapidly increasing popularity of Tensorflow.

One very important realization which is just beginning to surface in the research community is that all of these frameworks depend on the same low-level libraries such as BLAS for efficiently performing linear algebraic routines. BLAS is a library from the 70s, which has added so-called levels over the years for vector operations (Level 1) for matrix-vector operations (Level 2) and for matrix-matrix operations (Level3). BLAS Level 1 operations are computed in linear time, Level 2 in quadratic and Level 3 operations are computed in cubic time. Tensor operations have traditionally been implemented in terms of BLAS operations, e.g. Matrix Multiplication, incurring both a performance and a storage overhead because tensors must be flattened to use matrix-matrix operations and this procedure is repeated multiple times depending on the model/algorithm, the dimension of the data as well as layout of caches and processors of the hardware. This typical memory blowup problem might have been a niche problem until now, but the more data is being processed and the faster analytics result are being expected, the more critical it will become [8] [25]. A promising abstraction we came across during the research after our Workshop is BLIS [40]. The BLIS framework is not a single library or static API, but rather a nearly-complete template for instantiating high-performance BLAS-like libraries.

At the hardware level most of the frameworks again depend on the same abstractions for translating the algebraic routines onto machine instruction sets through libraries such as CUDA and OpenCL. Whilst CUDA is a software layer that gives direct access to the GPU's virtual instruction set and parallel computational elements for NVIDIA hardware, OpenCL aims to deliver comparable abstraction across heterogeneous platforms consisting of central processing units (CPUs), graphics processing units (GPUs), digital signal processors (DSPs), field-programmable gate arrays (FPGAs) and other processors or hardware accelerators. In the application domain of IoT/CPS we have heterogeneous architectures across hierarchies of processors, memory, and network. In our discussions surrounding the workshop, we even questioned traditional processor architectures with hardware managed cache hierarchy, a design principal also from the 70s.

Indeed we are starting to see more innovation even at the processor level, because the cost of moving data across hardware-managed memory layers starts to dwarf the useful computation with that data. This difference was not significant in the early days of computing, and was remedied by scaling techniques via increasing processor clock frequencies and now increasing the number of cores integrated on a single chip. However, the difference in energy used for moving data to the computation versus the energy used for the computation itself becomes very costly when we have machine learning from massive amounts of data. The cost increase is exponential when tensor operations on multidimensional data are necessary. Google, accompanying the open sourcing of their Tensorflow framework for machine learning, unveiled the Tensor Processing Unit (TPU) [10], a custom application-specific integrated circuits (ASIC) built specifically for machine learning. TPUs "only" utilize a clever trick for optimizing performance per watt by allowing the chip to be more tolerant of reduced computational precision, which means it requires fewer transistors per operation. Others redesign processors from the ground up such as the NEO chip from REX Computing [12]. The design of NEO relies on a range of hardware simplifications which are focused on exposing low level functionality. Once a feature exists in software, the reshaping of the tensor could be fused with internal layout of data and packing operations, requiring no explicit reshaping operations or additional workspace and memory.

In summary, we believe three R&D directions will crystallize in the following years in the intersection of mass data-driven machine learning, tensor computing, and IoT/CPS:

- High-level building blocks of tensor decompositions to be used on top of lower level numerical computation libraries
- Basic multilinear algebraic libraries with optimized tensor operations for the currently heterogeneous processor architectures
- New processor architectures redesigned to fundamentally improve balance between extreme efficiency and reconfigurability

As a Tensor Computing for IoT community we will closely follow and co-develop in these R&D directions to also feed in the IoT/CPS requirements for reliability, safety and robustness in highly distributed systems.

4 Tensor Learn - Knowledge Hub

In a recent publication [33], co-authored by two of our participants, the authors state that "After two decades of research on tensor decompositions and applications, the senior co-authors still couldn't point their new graduate students to a single 'point of entry' to begin research in this area." There is this need to provide a comprehensive and deep overview to young researchers and practitioners that will enable them to start developing related algorithms and applying them also to IoT/CPS.

At the same time, another one of our participants has been recently recognized for the two decades of dedication to transforming the process and food industry through actionable insights gained by applying and refining tensor decomposition techniques on multi-way chemometrics data collected in manufacturing facilities. There is this reward for both researchers, industrial practitioners, as well as the society and organizations supporting them - especially "in a time when there is a flood of data, but not the resources to draw out valuable and socially beneficial information from it" [14].

In the few months since the Dagstuhl workshop, one of the organizers joined Amazon's Machine Learning team as principal research scientist, one organizer was called upon as an advisor for the development of a new embeddable chip to disrupt exascale computing, and yet another started her company to enable clean electricity usage and exchange at zero-marginal cost through data-driven automation. These industrial activities signal not only the renaissance but also to some extent the viability of tensor methods for dealing with massive amounts of data coming from an increasingly digitalizing world.

By reviving the tensor decomposition application fields through varied challenges and high-quality data from IoT/CPS, and by creating a focal point of knowledge consolidation and dissemination, we believe that we can considerably shorten the time for breakthrough research in socially beneficial fields such as energy, mobility, cities, and manufacturing to name a few. Through digitalization these areas will be main sources of massive amounts of data coming from high-precision sensors at higher speeds given the advances in communication and computing infrastructures. Many of the established businesses in these areas, especially small and medium enterprises, which do not have the resources for R&D but face the same data deluge, will highly benefit from educational and open source resources available through this international knowledge hub.

As an initial step towards creating the knowledge hub "Tensor Learn," two of our participants organized a workshop co-located with NIPS [24]. The workshop aimed to draw the attention to this recent renaissance of tensor methods in machine learning, availability of

new tensor numerical computation frameworks, and point towards open research questions. In order to make the extension from workshop to Knowledge Hub, we aim to:
- host IoT Tensor Data Challenges and
- call for multidisciplinary discourse on the tensor applications for machine learning
- whilst establishing the common tensor computing language to facilitate such discourse.

5 Vision & Strategy

In the short-term, tensors will be a crucial part of a bigger computational machinery supporting the domain experts of IoT/CPS due to the ability of tensor frameworks to capture and represent the multi-aspect information within raw data sets and streams. Further along the line, also connected machines and devices will be supported by the same machinery to carry on tasks in dynamic, (near) real-time environments along-side domain experts. Data scientists, data engineers and system engineers are already building pieces of this computational machinery.

We as a community will have reached a first milestone when we eventually can qualify the most heard phrase: "It depends on the data", e.g. through recipes and best practices. For example, in IoT/CPS data is always analyzed over time and space. In IoT, prediction (trending) is very important to detect anomalies that deviate from the prediction; e.g. anomalies in massive streams of IP traffic data coming from interconnected routers, or coming from interconnected machines in factories, or in the future from sensorized streets accommodating self-driving cars. In CPS, additionally the control aspect comes into play: Once connected machines and devices recognize objects and can classify those, then they can learn through reinforcement within safe parameters how to interact with their multi-dimensional environment.

Scalable tensor methods have attracted considerable amount of attention, with successes in a series of learning tasks, such as learning latent variable models, relational learning, spatio-temporal forecasting as well as training [19] and compression [20] of deep neural networks. As a community we want to pave the way towards successful application of these methods in IoT/CPS. Our milestones on this way are to:
- Showcase suitability of tensor methods on real-world data coming from IoT/CPS that have the inherent structures and complex dependencies that result from the networked nature of IoT/CPS.
- Identify the new research problems that dynamic, (near) real-time, and/or safety-critical systems of energy, mobility, factories expose – especially w.r.t. deployment and performant, robust computing.
- Motivate our multidisciplinary network to take on these research problems by contributing to tensor tools and frameworks for production-level development and deployment in IoT/CPS.

In the following we depict the strategic and tactical steps within the three action areas:

Develop IoT Tensor Data Challenge & Infrastructure to be plugged into Tensor Learn knowledge hub by
- Communicating the potential and curating data from
  - open data initiatives of cities and regulated governmental bodies through our extended network [7]
  - crowd-sourced open infrastructure data like opengridmap [29] (power system), openstreetmap [30] (mobility) and

    - open environmental sensing from open data APIs of hardware providers developers, e.g. Enphase solar inverter cloud API, or data on public blockchains, e.g. solar power generation data logged into Electricchain [13]
  - Forming partnerships with hardware/sensor providers/users who will benefit from tensor decomposition for improving interpretability of sensor data analytics and for compressed sensing and at the same time can explore how machine learning systems improve with the availability of high-accuracy and high-resolution data.
  - Applying for an international research grant that allows us to work together on curating the data and to create and host a benchmarking infrastructure, to extract and share best practices discovered through the challenges.

Open Source contributions to available tensor tools alongside tutorials, recipes and best practices of applications of these tensor methods listed along side the completed data challenges/benchmarks on the Tensor Learn knowledge hub. The established communities of available frameworks that we are extending can become sponsors and partners of the researchers and practitioners of Tensor Computing for IoT.

Promote and position Tensor Learn as a knowledge hub that started as a workshop co-located with NIPS in order to advance the multidisciplinary discourse between tensor computing and its applications in machine learning. In the same manner we will co-locate further workshops with renown IoT/CPS conferences of IEEE, ACM, and the International Federation of Automation and Control (IFAC). Along the way gathering significant curated data challenges and benchmarks for typical tasks in multi-aspect IoT/CPS that can be automated through machine learning in a reliable and interpretable way by utilizing tensor methods.

6 Participants

- Evrim Acar (University of Copenhagen, DK)
- Kareem Aggour (General Electric – Niskayuna, US)
- Animashree Anandkumar (University of California – Irvine, US)
- Rasmus Bro (University of Copenhagen, DK)
- Ali Taylan Cemgil (Boğaziçi University – Istanbul, TR)
- Edward Curry (National University of Ireland – Galway, IE)
- Lieven De Lathauwer (KU Leuven, BE)
- Hans Hagen (TU Kaiserslautern, DE)
- Souleiman Hasan (National University of Ireland – Galway, IE)
- Denis Krompaß (Siemens AG – München, DE)
- Gerwald Lichtenberg (HAW – Hamburg, DE)
- Benoit Meister (Reservoir Labs, Inc. – New York, US)
- Lenore Mullin (University of Albany – SUNY, US)
- Morten Mørup (Technical University of Denmark – Lyngby, DK)
- Axel-Cyrille Ngonga-Ngomo (Universität Leipzig, DE)
- Ivan Oseledets (Skoltech – Scolkovo, RU)
- Renato Pajarola (Universität Zürich, CH)
- Vagelis Papalexakis (Carnegie Mellon University, US)
- Christine Preisach (SAP SE – Walldorf, DE)
- Achim Rettinger (KIT – Karlsruher Institut für Technologie, DE)
- Sebnem Rusitschka (Siemens AG – München, DE)
- Volker Tresp (Siemens AG – München, DE)
- Bülent Yener (Rensselaer Polytechnic Institute – Troy, US)

References

1 Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Gregory S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian J. Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Józefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Rajat Monga, Sherry Moore, Derek Gordon Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul A. Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda B. Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *CoRR*, abs/1603.04467, 2016. URL: http://arxiv.org/abs/1603.04467, `arXiv:1603.04467`.

2 Evrim Acar, Animashree Anandkumar, Lenore Mullin, Sebnem Rusitschka, and Volker Tresp. Tensor computing for internet of things (dagstuhl perspectives workshop 16152). *Dagstuhl Reports*, 6(4):57–79, 2016. `doi:10.4230/DagRep.6.4.57`.

3 Evrim Acar, Canan Aykut-Bingöl, Haluk Bingol, Rasmus Bro, and Bülent Yener. Multiway analysis of epilepsy tensors. In *Proceedings 15th International Conference on Intelligent Systems for Molecular Biology (ISMB) & 6th European Conference on Computational Biology (ECCB), Vienna, Austria, July 21-25, 2007*, pages 10–18, 2007. `doi:10.1093/bioinformatics/btm210`.

4 Evrim Acar and Bülent Yener. Unsupervised multiway data analysis: A literature survey. *IEEE Trans. Knowl. Data Eng.*, 21(1):6–20, 2009. `doi:10.1109/TKDE.2008.112`.

5 Yun-Jhong Wu Alex Williams. A julia implementation of tensor decomposition algorithms, 2016. (accessed January 13, 2017). URL: https://github.com/JuliaTensors.

6 Brett W Bader and Tamara G Kolda. A preliminary report on the development of matlab tensor classes for fast algorithm prototyping. Technical report, Technical Report SAND2004–3487, Sandia National Laboratories, Livermore, CA, 2004.

7 Laure Le Bars, Edward Curry, Thomas Hahn, and Milan Petkovi? Big data value association, 2015. (accessed January 13, 2017). URL: http://www.bdva.eu/.

8 Muthu Manikandan Baskaran, Benoît Meister, Nicolas Vasilache, and Richard Lethin. Efficient and scalable computations with sparse tensors. In *IEEE Conference on High Performance Extreme Computing, HPEC 2012, Waltham, MA, USA, September 10-12, 2012*, pages 1–6. IEEE, 2012. `doi:10.1109/HPEC.2012.6408676`.

9 Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, James Bergstra, Ian J. Goodfellow, Arnaud Bergeron, Nicolas Bouchard, David Warde-Farley, and Yoshua Bengio. Theano: new features and speed improvements. *CoRR*, abs/1211.5590, 2012. URL: http://arxiv.org/abs/1211.5590, `arXiv:1211.5590`.

10 Google Cloud Platform Blog. Google supercharges machine learning tasks with tpu custom chip, 2016. (accessed January 13, 2017). URL: https://cloudplatform.googleblog.com/2016/05/Google-supercharges-machine-learning-tasks-with-custom-chip.html.

11 J. Douglas Carroll and J.-J. Chang. Analysis of individual differences in multidimensional scaling via an N-way generalization of "Eckart-Young" decomposition. *Psychometrika*, 35:283–319, 1970.

12 REX Computing. The rex neo architecture, 2015. (accessed January 13, 2017). URL: http://www.rexcomputing.com/.

13 eletricchain. Electricchain, 2016. (accessed January 13, 2017). URL: http://www.electricchain.org/.

14 University of Copenhagen Faculty of Science. Food professor rasmus bro receives the first nils foss excellence prize, 2016. (accessed January 13, 2017). URL: http://www.science.ku.dk/english/press/news/2016/food-professor-rasmus-bro-receives-the-first-nils-foss-excellence-prize/.

15 Google. Tensorflow: An open source software library for numerical computation using data flow graphs, 2015. (accessed August 16, 2016). URL: https://www.tensorflow.org/.

16 Lars Grasedyck, Daniel Kressner, and Christine Tobler. A literature survey of low-rank tensor approximation techniques. *GAMM-Mitt.*, 36(1):53–78, 2013.

17 Richard A. Harshman. Foundations of the PARAFAC procedure: Models and conditions for an "explanatory" multi-modal factor analysis. *UCLA working papers in phonetics*, 16:1–84, 1970.

18 Furong Huang. A spectral (third order tensor decomposition) learning method for learning lda topic model on spark, 2016. (accessed January 13, 2017). URL: https://github.com/FurongHuang/SpectralLDA-TensorSpark.

19 Majid Janzamin, Hanie Sedghi, and Anima Anandkumar. Beating the perils of non-convexity: Guaranteed training of neural networks using tensor methods. *CoRR abs/1506.08473*, 2015.

20 Yong-Deok Kim, Eunhyeok Park, Sungjoo Yoo, Taelim Choi, Lu Yang, and Dongjun Shin. Compression of deep convolutional neural networks for fast and low power mobile applications. *CoRR*, abs/1511.06530, 2015. URL: http://arxiv.org/abs/1511.06530, `arXiv:1511.06530`.

21 Alicia Klinvex, Grey Ballard, Tamara Kolda, Woody Austin, and Hemanth Kolla. Deeplearning4j: Open-source distributed deep learning for the jvm, 2014. (accessed January 13, 2017). URL: https://deeplearning4j.org/index.html.

22 Alicia Klinvex, Grey Ballard, Tamara Kolda, Woody Austin, and Hemanth Kolla. Tuckermpi on gitlab: Open-source software for distributed tucker computations on large-scale dense data, 2017. (accessed January 13, 2017). URL: https://gitlab.com/tensors/TuckerMPI.

23 Tamara G. Kolda and Brett W. Bader. Tensor decompositions and applications. *SIAM Review*, 51(3):455–500, 2009. `doi:10.1137/07070111X`.

24 Anima Anand Kumar, Maximilan Nickel, Rong Ge, Yan Liu, and Rose Yu. Tensor learn workshop@nips - learning with tensors: Why and how?, December 10, 2016. (accessed January 13, 2017). URL: http://tensor-learn.org/.

25 Devin Matthews. Blas for tensors: What, why, and how?, 2015. (accessed January 13, 2017). URL: https://www.cs.utexas.edu/users/flame/BLISRetreat2015/slides/Devin_BLISRetreat_2015.pdf.

26 Michael Hirn Maximilian Goisser. Open machine intelligence framework for hackers, 2014. (accessed January 13, 2017). URL: https://github.com/autumnai.

27 Fumikazu Miwakeichi, Eduardo Martinez-Montes, Pedro A. Valdés-Sosa, Nobuaki Nishiyama, Hiroaki Mizuhara, and Yoko Yamaguchi. Decomposing EEG data into space-time-frequency components using parallel factor analysis. *NeuroImage*, 22:1035–1045, 2004.

28 Maximilian Nickel. scikit-tensor: Python library for multilinear algebra and tensor factorizations, 2013. (accessed August 16, 2016). URL: https://github.com/mnick/scikit-tensor.

29 Open grid map, 2014. (accessed January 13, 2017). URL: http://opengridmap.com/.

30 Open street map, 2004. (accessed January 13, 2017). URL: https://www.openstreetmap.org.

31 Ivan Oseledets. Tt-toolbox (tt=tensor train) version 2.2.2, 2016. (accessed January 13, 2017). URL: https://github.com/oseledets/TT-Toolbox.

32 Evangelos E. Papalexakis, Christos Faloutsos, and Nicholas D. Sidiropoulos. Tensors for data mining and data fusion: Models, applications, and scalable algorithms. *ACM TIST*, 8(2):16:1–16:44, 2017. `doi:10.1145/2915921`.

33 Nicholas D. Sidiropoulos, Lieven De Lathauwer, Xiao Fu, Kejun Huang, Evangelos E. Papalexakis, and Christos Faloutsos. Tensor decomposition for signal processing and machine learning. *CoRR*, abs/1607.01668, 2016. URL: http://arxiv.org/abs/1607.01668, `arXiv:1607.01668`.

34 Age K. Smilde, Rasmus Bro, and Paul Geladi. *Multi-way Analysis with Applications in the Chemical Sciences*. Wiley, 2004.

35 Apache Spark. Apache spark: A fast and general engine for large-scale data processing, 2012. (accessed August 16, 2016). URL: http://spark.apache.org/.

36 Statista. Internet of things (iot): number of connected devices worldwide from 2012 to 2020 (in billions), 2015. (accessed August 16, 2016). URL: http://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/.

37 Ledyard R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31:279–311, 1966.

38 N Vervliet, O Debals, L Sorber, M Van Barel, and L De Lathauwer. Tensorlab 3.0. *available online*, 2016. URL: http://www.tensorlab.net.

39 Pete Warden. How to quantize neural networks with tensorflow, 2016. (accessed January 13, 2017). URL: https://petewarden.com/2016/05/03/how-to-quantize-neural-networks-with-tensorflow/.

40 Field G. Van Zee and Robert A. van de Geijn. BLIS: A framework for rapidly instantiating BLAS functionality. *ACM Trans. Math. Softw.*, 41(3):14:1–14:33, 2015. doi:10.1145/2764454.