COMPUTING edge

- Embedded Systems
- Mobile Computing
- Automation
- Ethics

JANUARY 2021

www.computer.org

75 YEARS IEEE COMPUTER SOCIETY

IEEE

IEEE COMPUTER SOCIETY JOBS BOARD

Evolving Career Opportunities Need Your Skills

Explore new options—upload your resume today

www.computer.org/jobs

Changes in the marketplace shift demands for vital skills and talent. The **IEEE Computer Society Jobs Board** is a valuable resource tool to keep job seekers up to date on the dynamic career opportunities offered by employers.

Take advantage of these special resources for job seekers:

- JOB ALERTS
- TEMPLATES
- WEBINARS
- CAREER ADVICE
- RESUMES VIEWED BY TOP EMPLOYERS

No matter what your career level, the IEEE Computer Society Jobs Board keeps you connected to workplace trends and exciting career prospects.

75 YEARS IEEE COMPUTER SOCIETY

IEEE

COMPUTING edge

IEEE COMPUTER SOCIETY computer.org

STAFF

Circulation: *ComputingEdge* (ISSN 2469-7087) is published monthly by the IEEE Computer Society. IEEE Headquarters, Three Park Avenue, 17th Floor, New York, NY 10016-5997; IEEE Computer Society Publications Office, 10662 Los Vaqueros Circle, Los Alamitos, CA 90720; voice +1 714 821 8380; fax +1 714 821 4010; IEEE Computer Society Headquarters, 2001 L Street NW, Suite 700, Washington, DC 20036.

Postmaster: Send address changes to *ComputingEdge*-IEEE Membership Processing Dept., 445 Hoes Lane, Piscataway, NJ 08855. Periodicals Postage Paid at New York, New York, and at additional mailing offices. Printed in USA.

Editorial: Unless otherwise stated, bylined articles, as well as product and service descriptions, reflect the author's or firm's opinion. Inclusion in *ComputingEdge* does not necessarily constitute endorsement by the IEEE or the Computer Society. All submissions are subject to editing for style, clarity, and space.

IEEE Computer Society Magazine Editors in Chief

JANUARY 2021 • VOLUME 7 • NUMBER 1

COMPUTING edge

Embedded Systems

Mobile Computing

Automation

Ethics

Departments

Magazine Roundup

The IEEE Computer Society's lineup of 12 peer-reviewed technical magazines covers cutting-edge topics ranging from software design and computer graphics to Internet computing and security, from scientific applications and machine intelligence to visualization and microchip design. Here are highlights from recent issues.

Computer

Real-Time Systems Implications in the Blockchain-Based Vertical Integration of Industry 4.0

The Industrial Internet of Things (IIoT) is expected to attract significant investments for industry. In this new environment, blockchain presents immediate potential in applications of the IIoT, offering several benefits to industrial cyber-physical systems. Read more in this article from the September 2020 issue of *Computer*.

Computing in Science & Engineering

Data Cyberinfrastructure for End-to-End Science

Large-scale scientific facilities provide a broad community of researchers and educators with open access to instrumentation and data products generated from geographically distributed instruments and sensors. This article from the September/October 2020 issue of *Computing in Science & Engineering* discusses key architectural design, deployment, and operational aspects of a production cyberinfrastructure for the acquisition, processing, and delivery of data from large scientific facilities—using experiences from the National Science Foundation's Ocean Observatories Initiative. This paper also outlines new models for data delivery and opportunities for insights in a wide range of scientific and engineering domains as the volumes and variety of data from facilities grow.

IEEE Annals of the History of Computing

IBM's World Citizens: Valentim Bouças and the Politics of IT Expansion in Authoritarian Brazil

This article from the July–September 2020 issue of *IEEE Annals of the History of Computing* analyzes the politics of IBM's expansion in interwar Brazil. It does so by focusing on Valentim Bouças, IBM's first representative in Brazil and an outstanding figure in IBM trade press materials for the rapid pace at which he grew IBM's Brazilian operations. Grounded in the earliest recorded moment in which IBM was first threatened with expulsion from Brazilian markets, this article analyzes how IBM, led by Bouças, regained Brazilian markets and expanded its operations in the country through the political negotiations it entered into. This article analyzes how IBM, as a US-based multinational IT firm, first installed itself in Brazil's interwar authoritarian regime, helping restructure Brazilian administrative and financial apparatuses to its advantage.

IEEE Computer Graphics and Applications

Spatialized Audio in a Custom-Built OpenGL-Based Ear Training Virtual Environment

Interval recognition is an important part of ear training—the key aspect of music education. Once trained, the musician can identify pitches, melodies, chords, and rhythms by listening to music segments. In a conventional setting, the tutor would teach a trainee the intervals using a musical instrument, typically a piano. However, with the emergence of new technologies such as virtual reality (VR) and areas such as edutainment, this and similar trainings can be transformed into more engaging, more accessible,

customizable (virtual) environments, with the addition of new cues and bespoke progression settings. In this article from the September/October 2020 issue of *IEEE Computer Graphics and Applications*, the authors describe a VR ear training system for interval recognition.

IEEE Intelligent Systems

Proxy Experience Replay: Federated Distillation for Distributed Reinforcement Learning

Traditional distributed deep reinforcement learning (RL) commonly relies on exchanging the experience replay memory (RM) of each agent. Since the RM contains all state observations and action policy history, it may incur huge communication overhead while violating the privacy of each agent. This article from the July/August 2020 issue of *IEEE Intelligent Systems* presents a communication-efficient and privacy-preserving distributed RL framework, coined federated reinforcement distillation (FRD). In FRD, each agent exchanges its proxy experience RM (ProxRM), in which policies are locally averaged with respect to proxy states clustering actual states. To provide FRD design insights, the authors present ablation studies on the impact of ProxRM structures, neural network architectures, and communication intervals.

IEEE Internet Computing

Fog Computing as Privacy Enabler

Despite broad discussions on privacy challenges arising from fog computing, the authors of this article from the July/August 2020 issue of *IEEE Internet Computing* argue that privacy and security requirements might actually drive the adoption of fog computing. They present four patterns of fog computing fostering data privacy and the security of business secrets, complementing existing cryptographic approaches. Their practical application is illuminated on the basis of three case studies.

IEEE micro

PurpleDrop: A Digital Microfluidics-Based Platform for Hybrid Molecular-Electronics Applications

Molecular manipulation and analysis are the cornerstone of life sciences. With the recent advances in molecular data storage and computing, it has become an increasingly exciting and viable alternative for the post-CMOS scaling era. Widespread use of molecular manipulation/analysis and data storage/computing requires a scalable and low-cost platform for hybrid molecular-electronics systems. This enables us to build on the best of what molecular and electronics systems can offer. In this article from the September/October 2020 issue of *IEEE Micro*, the authors present PurpleDrop, a full-stack digital microfluidic platform for hybrid molecular-electronic systems in multiple domains, and focus on DNA data storage as a use case.

IEEE MultiMedia

Wall Screen: An Ultra-High Definition Video-Card for the Internet of Things

Eight-k ultra-high definition (UHD) is paving the way for the next-generation video systems. In the audio-visual industry, besides delivering a more immersive experience, it is a means to smooth spatial artifacts during video sampling. In the medical industry, it may provide surgeons with increased reality in surgeries. Nevertheless, researchers are struggling to meet the high throughput required by this resolution, and hardware solutions miss the flexibility required for on-demand updates. In this context, the authors of this article from the July–September 2020 issue of *IEEE MultiMedia* propose an 8-k video

card based on a hybrid platform endowed with "soft" programmable logic and "hard" processors.

Leveraging IoTs and Machine Learning for Patient Diagnosis and Ventilation Management in the Intensive Care Unit

Future healthcare systems will rely heavily on clinical decision support systems (CDSS) to improve the decision-making processes of clinicians. To explore the design of future CDSS, the authors of this article from the July–September 2020 issue of *IEEE Pervasive Computing* developed a research-focused CDSS for the management of patients in the intensive care unit that leverages Internet of Things devices capable of collecting streaming physiologic data from ventilators and other medical devices. They then created machine-learning models that could analyze the collected physiologic data to determine if the ventilator was delivering potentially harmful therapy and if a deadly respiratory condition, acute respiratory distress syndrome (ARDS), was present.

Plundervolt: How a Little Bit of Undervolting Can Create a Lot of Trouble

Historically, fault injection was the realm of adversaries with physical access. This changed when research revealed that remote attackers could use software to inject faults. Plundervolt is a new software-based attack on Intel's trusted execution technology (SGX). Plundervolt can break cryptography and inject memory-safety bugs into secure code. Read more in this article from the September/October 2020 issue of *IEEE Security & Privacy*.

Neural Distributed Ledger

The neural distributed ledger solution presented in this article from the September/October 2020 issue of *IEEE Software* adopts a ledger-of-ledgers approach to perform the interconnection of multiple ledgers. Beyond inter-ledger operability, it also enables the development of custom blockchain-based solutions, providing controlled costs and scalability while retaining decentralization and security.

Predictive Maintenance for Infrastructure Asset Management

Optimal maintenance is one of the key concerns for asset-intensive industries in terms of reducing downtime and occurring costs. The advancement of data-driven technologies, affordable computing powers, and growing amounts of data introduced a paradigm called predictive maintenance (PdM). PdM seeks to find out an optimal moment for the maintenance of an asset, where no early intervention leads to undue extra cost, and no late maintenance activity poses a safety risk. With the instrumentation of the cyber-physical system on assets, PdM transforms a typical structure into a smart structure that can send warnings in cases of near failure states. However, several practical challenges hamper the adoption of PdM solutions within industries. This article from the September/October 2020 issue of *IT Professional* outlines a typical PdM modeling framework and its key components.

Editor's Note

What's Next for Embedded Systems?

Embedded systems are becoming pervasive—with applications in consumer electronics, autonomous vehicles, and the Internet of Things (IoT)—but they still have a long way to go to reach peak efficiency, dependability, and interoperability. In this *ComputingEdge* issue, two articles from *Computer* examine the past, present, and future of embedded systems, including research opportunities and challenges.

In "Embedded Artificial Intelligence: The ARTEMIS Vision," the authors describe the current state of the art in embedded and cyber-physical systems and contemplate the improvements needed to achieve embedded intelligence. In "Developing IoT Systems: It's All About the Software," the authors focus on embedded systems in the IoT, arguing that developing software for these systems should be simpler and more collaborative.

Embedded systems are common in mobile devices, which have become essential tools for communication, business, and daily life. *IEEE Internet Computing*'s "6G Vision: An AI-Driven Decentralized Network and Service Architecture" discusses the future of mobile computing beyond 5G networks. *IEEE Computer Graphics and Applications*' "Turning a Smartphone Selfie Into a Studio Portrait" presents an algorithm for automatically removing lighting artifacts from flash photos taken on mobile devices.

Image editing is just one of many previously manual tasks that can now be automated. *IEEE Intelligent Systems*' "Intent Classification for Dialogue Utterances" looks at automated customer service systems that use intent classification to determine the reason the customer is contacting the organization. *IT Professional*'s "White Learning: A White-Box Data Fusion Machine Learning Framework for Extreme and Fast Automated Cancer Diagnosis" proposes a framework for automated breast cancer detection.

This *ComputingEdge* issue concludes with two articles about ethics in computing and technology. The author of *IEEE Software*'s "Is Your Software Valueless?" argues that software design should consider values such as compassion, social responsibility, and justice—not just business needs like cost and performance. In *IEEE MultiMedia*'s "Shaping Our Common Digital Future," the author urges multimedia researchers and practitioners to approach their work with social good in mind.

EDITOR: Dimitrios Serpanos, ISI/ATHENA and University of Patras, serpanos@computer.org

COLUMN: CYBER-PHYSICAL SYSTEMS

This article originally appeared in
Computer
vol. 53, no. 11, 2020

Embedded Artificial Intelligence: The ARTEMIS Vision

Dimitrios Serpanos, *ISI/ATHENA and University of Patras*
Gianluigi Ferrari, *University of Parma*
George Nikolakopoulos, *Lulea University of Technology*
Jon Perez, *Ikerlan*
Markus Tauber, *University of Applied Sciences Burgenland*
Stefan Van Baelen, *IMEC*

Advances in embedded and cyberphysical systems have disrupted numerous application domains. We examine the requirements and challenges of these technologies, which present significant opportunities for interdisciplinary research.

Embedded computing has brought significant advances in application domains ranging from home appliances and health systems to environmental monitoring and from smart factories to autonomous transportation (cars, trains, ships, and airplanes) and smart cities. Embedded computing systems constitute the cyber part of cyberphysical systems (CPSs). Autonomous CPSs are commonly used in processes of increasing complexity that are designed and implemented with single-processor systems (for example, a patient's insulin pump) or distributed, interconnected processing nodes (for example, autonomous vehicles). Autonomous CPSs have also become increasingly connected to the Internet of Things (IoT), which includes specialized networks, such as the Industrial IoT, Internet of Vehicles, and others.

Clearly, a hierarchy of CPSs is emerging, where simple autonomous systems are interconnected to create higher-level autonomous systems that, in turn, are interconnected to provide even more complex systems and applications. For example, a CPS for an autonomous car's cruise control is part of an autonomous car—a more complex distributed CPS—that may be a node of a network of autonomous vehicles (a fleet) managed through a cloud application.

The pervasiveness of embedded systems and the increasing deployment of CPSs lead to an emerging infrastructure that spans globally and enables the development of new applications and services that were infeasible or inconceivable in the recent past. The immediate availability of operational data as well as computational power in conjunction with artificial intelligence (AI) techniques provides significant opportunities for systems and services worldwide. To achieve this vision, CPSs need to be efficient, scalable, and extensible in terms of both hardware and software.

The adoption of CPSs in various application domains leads to strong constraints on their design and implementation. More precisely, CPS technologies are quite demanding for the purpose of satisfying strong application and operational environment requirements, including real-time constraints, safety and security, continuous operation, scalability, extensibility, autonomy, power consumption, and internetworking. Although CPSs typically abide by several of these requirements, the application domains, ranging from manufacturing to transport and from health to power, impose different constraints on each

Digital Object Identifier 10.1109/MC.2020.3016104
Date of current version: 21 October 2020

 Published by the IEEE Computer Society

specification. For example, industrial production systems have stricter requirements for real-time constraints and continuous operation than home automation systems, while they have more relaxed stipulations for power consumption relative to autonomous, mobile health-monitoring systems.

RECENT DEVELOPMENTS ENABLE THE CREATION OF AUTONOMOUS SYSTEMS THAT ARE SELF-AWARE AND ADAPTIVE TO DYNAMIC ENVIRONMENTS.

EMBEDDED INTELLIGENCE

The significant recent technological advances, including the revolution in AI, lead to the increased "intelligence" of computational systems and, especially, of CPSs. This happens during both the design phase and operation in the field. Autonomous and semiautonomous systems have been a reality for a long time in controlled environments, for example, robots in manufacturing lines, but recent developments enable the creation of autonomous systems that are self-aware and adaptive to dynamic environments, such as efficient and safe self-driving vehicles.

Embedded intelligence requires the development of efficient and effective technologies for embedded systems and CPSs in all application domains. A presentation of the related key technologies appears in the core circle of Figure 1, which presents the vision of the ARTEMIS Industrial Association, the largest European organization focusing on embedded systems, CPSs, and related technologies.[1]

The quest for embedded intelligence requires efficient embedded systems and CPSs, with effective processors, coprocessors, memories, network subsystems, special-purpose circuits, operating systems, programming environments, and so forth. Considering that these systems operate in resource-constrained environments, depending on the application domain, efficient tools are necessary for design space

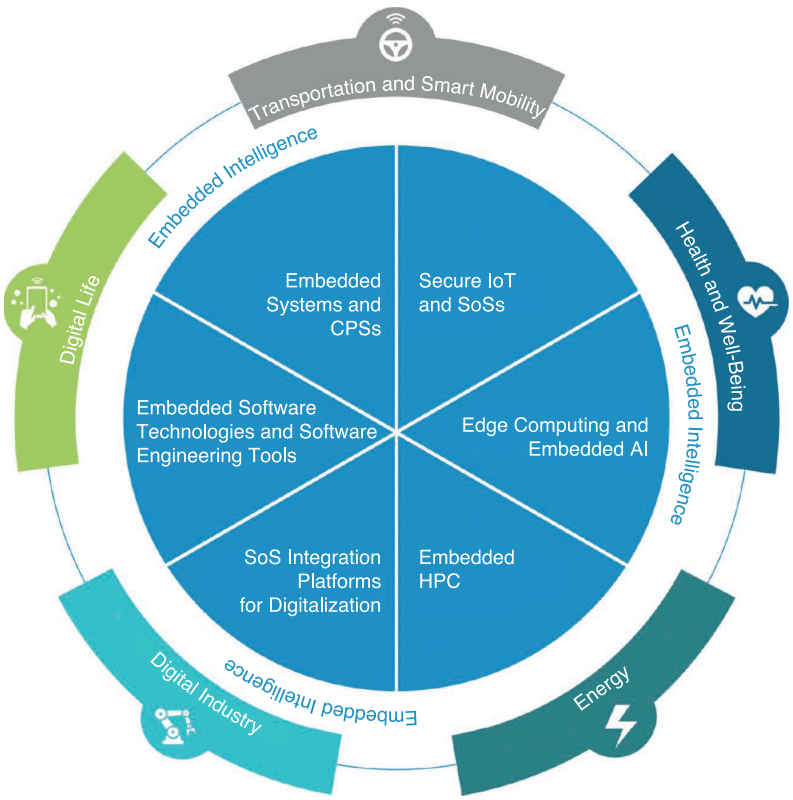


FIGURE 1. Embedded intelligence. (Source: https://artemis-ia.eu/key-technologies.html; used with permission.) SoSs: systems of systems; HPC: high-performance computing.

exploration that combines hardware and software so as to identify appropriate, effective designs. This is in to the technologies required for efficient and cost-effective systems.

The increasing computational capabilities of CPS nodes lead to powerful distributed systems that implement complex processes with high performance and reliability. The traditional model—where edge nodes collect information and transmit the data to centralized systems (or the cloud) for processing and actuation feedback—is rapidly changing to a model of powerful interconnected nodes that execute sophisticated processing locally and send information and event data only as required to central nodes. This augments performance and supports real-time processing, due to increased local node processing and a reduced centralized processing load, and it improves reliability and security, as a result of local storage and less data transmission, while saving bandwidth and reducing network complexity. Edge computing, coupled with AI methods, enables faster processing and decisions near data sources. In particular, it strongly supports the evolution of autonomous CPSs that are self-aware and adaptable to dynamic environments without sacrificing their interconnectivity and orchestration for higher-level complex applications.

THE EVOLUTION OF "SMART" EDGE SYSTEMS ENABLES HIGHLY DEMANDING DISTRIBUTED APPLICATIONS AND SERVICES IN SEVERAL DOMAINS.

The evolution of "smart" edge systems enables highly demanding distributed applications and services in several domains; for example, aerial autonomous vehicles enable services from fleet management to border surveillance. The requirements for increasing functionality and efficiency for hyperconverged infrastructure at the edge and those for smart sensors—for example, smart cameras—lead to a need for high-performance computing (HPC) architectures at the CPS level, where embedded vision systems, virtual reality, data fusion, and AI constraints are representative examples of the need for sophisticated, embedded HPC architectures.

The dramatic penetration of CPSs in increasing domains, from avionics to agriculture and from manufacturing to health, is disruptive. The rapidly growing number of embedded platforms constitutes a strong enabler of new business opportunities and models that have become feasible. More importantly, such models and opportunities are multiplying, and it is certain that new, unforeseen services will appear in the future. The ability to organize CPSs in domains, develop applications on them, and manage them effectively requires designs that can efficiently synthesize in large systems, effectively and seamlessly, providing necessary special-purpose computational infrastructures at will. Technologies that integrate systems and enable building systems of systems (SoSs) are fundamental in this direction. Software technologies and appropriate software architectures, such as service-oriented ones, are necessary to address the needs for evolving systems and platforms and for enabling novel services and business models.

Safety is a fundamental property of CPSs, considering their role in multiple processes, including health, manufacturing, and transportation (planes, ships, trains, and vehicles). Safety engineering for CPSs is a cornerstone of the emerging Industry 4.0 and Society 5.0 concepts based on CPSs. Safety requires the mitigation of both accidental failures and cyberattacks on computational and network resources and operations. Security mechanisms are required to protect the data on which safety mechanisms rely. It is imperative to develop methods and mechanisms to build overall safe and secure CPSs, not only individually but in dynamic interconnections when building SoSs, where collective properties need to be attained based on the safety and security properties of individual systems. Safety is a necessary property not only from the technological point of view but also the social one since it is key to the acceptance and adoption of CPS technologies in society.

Exploiting the preceding technologies, CPSs achieve embedded intelligence employed in all application domains, such as digital industry, transportation, health, and so on. Such application domains are included in the outer cycle of Figure 1, where the list is not exhaustive but descriptive at a high level of

abstraction, indicating priority areas for European industry. Specialized domains include smart agriculture, supply chain management, and border surveillance, to name a few.

RESEARCH CHALLENGES

Embedded intelligence presents several research challenges in its core technologies. In the remainder of this article, we describe several illustrative difficulties. We remark, however, that these are not exhaustive, and, although we organize them according to the core technologies of Figure 1, several of them are cross domain.

The diverse and increasing CPS application domains with strong functional and nonfunctional requirements, such as safety, security, real-time constraints, and low power consumption, drive the new generations of embedded computing systems that exploit multicore devices and advanced virtualization technology. New multicore devices that have novel architectures with effective memory structures (including distributed shared memories), high-performance coprocessors (such as graphics processing units, tensor cores, and programmable cells on field-programmable gate array components), and on-chip diagnosis and thermal management components provide a continuous challenge that targets the development of low-cost and power-efficient devices that offer the necessary performance and connectivity for increasingly demanding environments.[2]

The integration of these components, as well as smart sensors, creates significant research obstacles at all fronts, from semiconductor design to dependable system architectures. The challenges extend to the development of integrated development environments (IDEs) and tools that support the cost-efficient and dependable growth of CPSs, enabling design and design management at the appropriate abstraction levels to manage heterogenous languages and computing platforms, real-time guarantees, dependability constraints, and so forth. New models of computing, such as approximate, neuromorphic, and AI, provide promising results in several domains, including the cyberphysical interface. The efficient inclusion of AI processing components in embedded devices (for application efficiency and system dependability, among others) poses significant difficulties at both the design and IDE levels. Moreover, the strong progress at all fronts of embedded systems and CPS design creates a significant challenge to standardization and certification efforts, especially for safety-related systems that include AI subsystems.

The increasing interconnection and integration of independent, dedicated CPSs to form a higher-level single system while maintaining continuous operation independently of the collaborative system leads to the concept of SoSs.[3,4] Platforms that integrate SoSs for digitalization pose major challenges, such as the effective integration of SoSs at the appropriate middleware layer, thus enabling direct interaction among component systems and minimizing complexity while ensuring upgradability, scalability, and extensibility. Appropriate assessment metrics need to be identified to evaluate the performance of integrated systems, especially with respect to single systems. Platform definitions, in terms of functionality and supported (hardware and software) components, are required together with specifications for the cyberenvironment where digitalization takes place; the scalability and interoperability of the platform are key aspects.

APPROPRIATE ASSESSMENT METRICS NEED TO BE IDENTIFIED TO EVALUATE THE PERFORMANCE OF INTEGRATED SYSTEMS, ESPECIALLY WITH RESPECT TO SINGLE SYSTEMS.

In the context of SoS integration, the concepts of the fog, cloud, and IoT, together with the upcoming 5G technology, pose several major difficulties for effective and efficient communication architectures. All these paradigms foster the integration of SoSs in unprecedented ways, supporting a physical and logical network hierarchy of multiple levels of cooperating nodes. Nevertheless, it is necessary to automatically orchestrate different devices and layers, enabling resource sharing and interactions between nodes at the same layer and at different layers in the hierarchy. To meet the specific requirements of integrating SoSs, the combination of heterogeneous communication

and application protocols plays a key role. The IoT ecosystem is a perfect illustrative example of the need for communication protocol interoperability.[5]

Safety and security constitute a significant challenge to CPSs. Inherently, intelligent embedded systems and CPSs collect data, which is often sensitive, and make decisions based on that information. Privacy and data integrity form fundamental requirements of these systems to protect information appropriately and secure correct decisions. Safe and secure systems require new security-by-design and safety-by-design approaches that minimize attack and failure surfaces. The difficulties increase when considering CPSs with AI components, where data integrity, as well as algorithmic correctness, is a strong requirement. The inclusion of the cloud with CPS applications and services constitutes a challenge by itself, considering the current open problems of cloud security. The challenge of safe and secure CPSs expands to standardization and certification efforts in view of the legal and social aspects of the emerging CPSs that range from critical infrastructures to autonomous vehicles.

Because CPSs are computing systems with complex software components (in addition to hardware), software engineering and tools for embedded software play an important role in the development of efficient, safe, and reliable systems. Methods for software and system verification, testing for high-level properties (for example, safety and security), runtime verification, software synthesis, and software maintenance and management become increasingly important under the constraints for continuous fail-safe and real-time operation.[6] Advanced system and software management operations, such as the runtime confirmation of certification compliance due to multiple stakeholders, constitute significant process-dependent challenges.[7] Virtualization software has become a fundamental requirement for CPSs, leading to a strong need for mechanisms and tools for the efficient virtualization of constrained and heterogeneous microprocessor and multicore platforms.

New languages and tools for safe application development across distributed middleware frameworks and virtualized distributed platforms are required. Furthermore, the efficient integration of AI components in systems, especially for non-AI expert developers, is a significant growing challenge that requires fresh approaches to modular design and AI process specification. This is also part of the software lifecycle management, which is especially demanding in CPSs; agile methodologies, continuous integration, DevOps, and reconfigurability in real-time distributed and/or safety-critical systems require novel techniques for the constrained CPS environment. Updating CPS software in the field is a characteristic example that demonstrates the need for methods that guarantee CPS properties, such as safety, security, and real-time operation, in contrast to traditional software updating methods. The new software not only needs to be verified or tested appropriately when developed but inserted in a way that enables real-time updates and nondisruptive continuous operation without violating functional and nonfunctional properties.

The advances in embedded and CPS technologies, coupled with the growth of the IoT, cloud computing, and AI, have led to disruptive growth models in application domains ranging from manufacturing to energy and from transportation to health. The increasing adoption of these systems in everyday operations places significant requirements on these systems, which are application and process dependent, creating significant new opportunities in interdisciplinary research.

REFERENCES

1. ARTEMIS Industry Association. Accessed: Aug. 10, 2020. [Online]. Available: https://artemis-ia.eu
2. J. Perez Cerrolaza et al., "Multi-core devices for safety-critical systems: A survey," *ACM Comput. Surv.*, vol. 53, no. 4, pp. 1–38, July 2020. doi: 10.1145/3398665.
3. M. W. Maier, "Architecting principles for systems-of-systems," *Syst. Eng.*, vol. 1, no. 4, pp. 267–284, 1998. doi: 10.1002/(SICI)1520-6858(1998)1:4<267::AID-SYS3>3.0.CO;2-D.
4. P. Azzoni, "*From Internet of Things to System of Systems: Market analysis, achievements, positioning and future vision of the ECS community on IoT and SoS*," ARTEMIS-IA, Eindhoven, The Netherlands, White Paper, Apr. 2020. [Online]. Available: https://artemis-ia.eu/news/artemis-whitepaper-from-the-internet-of-things-to-system-of-systems.html
5. S. Cirani, G. Ferrari, M. Picone, and L. Veltri, *Internet*

of Things: Architectures, Protocols and Standards. Chichester, U.K.: Wiley, 2018.
6. M. T. Khan, D. Serpanos, and H. E. Shrobe, "ARMET: Behavior-based secure and resilient industrial control systems," *Proc. IEEE*, vol. 106, no. 1, pp. 129–143, 2018. doi: 10.1109/JPROC.2017.2725642.
7. A. Bicaku, M. Tauber, and J. Delsing, "Security standard compliance and continuous verification for Industrial Internet of Things," *Int. J. Distrib. Sensor Netw.*, vol. 16, no. 6, 2020. doi: 10.1177/1550147720922731.

DIMITRIOS SERPANOS is director of the Industrial Systems Institute/ATHENA and a professor of electrical and computer engineering at the University of Patras. He is the chair of the ARTEMIS Scientific Council; a Senior Member of IEEE; and a member of the ARTEMIS Scientific Council, ACM, AAAS, and NYAS. Contact him at serpanos@computer.org.

GIANLUIGI FERRARI is an associate professor at the University of Parma, Italy, and a member of the ARTEMIS Scientific Council. Contact him at gianluigi.ferrari@unipr.it.

GEORGE NIKOLAKOPOULOS is a professor at Lulea University of Technology, Sweden, and a member of the ARTEMIS Scientific Council. Contact him at george.nikolakopoulos @ltu.se.

JON PEREZ is principal researcher at Ikerlan, Spain, and a member of the ARTEMIS Scientific Council. Contact him at jmperez@ikerlan.es.

MARKUS TAUBER is a professor at the University of Applied Sciences Burgenland, Austria, and a member of the ARTEMIS Scientific Council. Contact him at markus.tauber @fh-burgenland.at.

STEFAN VAN BAELEN is project manager at IMEC, Belgium, and a member of the ARTEMIS Scientific Council. Contact him at stefan.vanbaelen@imec.be.

EDITOR: Trevor Pering, Google, peringknife@google.com

This article originally appeared in
Computer
vol. 53, no. 4, 2020

COLUMN: THE IOT CONNECTION

Developing IoT Systems: It's All About the Software

Thomas Kubitza, Patrick Bader, and Matthias Mögerle, *ThingOS GmbH*
Albrecht Schmidt, *Ludwig Maximilian University of Munich*

FROM THE EDITOR

Not surprisingly, much of the attention regarding the Internet of Things (IoT) is placed on the things themselves, following the visceral nature of the physical world; however, the software often proves to be the make-or-break glue holding the things together. In this article, the authors explore the (software) tools and techniques necessary to make the development of IoT systems amenable to an actual design process and therefore a broader base of application developers. —*Trevor Pering*

For the Internet of Things to move from the lab to the real world, software and application development must be simplified, and collaboration must increase. We designed an operating system and a development environment that facilitate the process and supports a range of users.

The Internet of Things (IoT) is widely discussed in research as well as in industry, and everyone agrees on its huge potential. However, there is a big difference between the concepts, studies, and prototypes shown at academic conferences and the products and services deployed in industry. Academic research has proposed interactive and networked versions of everyday objects during the past 30 years, including Internet-enabled coffee cups.[1] In industry, the IoT is more like a natural evolution of factory and office automation. Commercially successful connected things are essentially computers in different form factors and with tailored software (for example, watches and smart TVs and speakers).

MAKING SMART THINGS IS STILL HARD

Why are the ideas for enhancing things by adding Internet connectivity not catching on? Why are there plenty of scientific publications and no products? Simply put, it remains difficult to develop and deploy IoT solutions; there are challenges in hardware, software, and business models.[2] Embedded devices with a web stack are inexpensive (for instance, the ESP8266 costs under US$1), but software development for individual instances and customers is not. What else is still hard? The list includes

- physical embedding, systems integration, and network connectivity
- software implementation and application development
- security, data protection, and robustness in productive environments.

In our experience, cost-effective software development for the IoT is the core challenge. Custom software development is the primary driver for cost when realizing IoT visions in industry. Weiser[4] foresaw that hardware would be ubiquitous (and cheap) but not

Digital Object Identifier 10.1109/MC.2020.2972658
Date of current version: 9 April 2020

 Published by the IEEE Computer Society

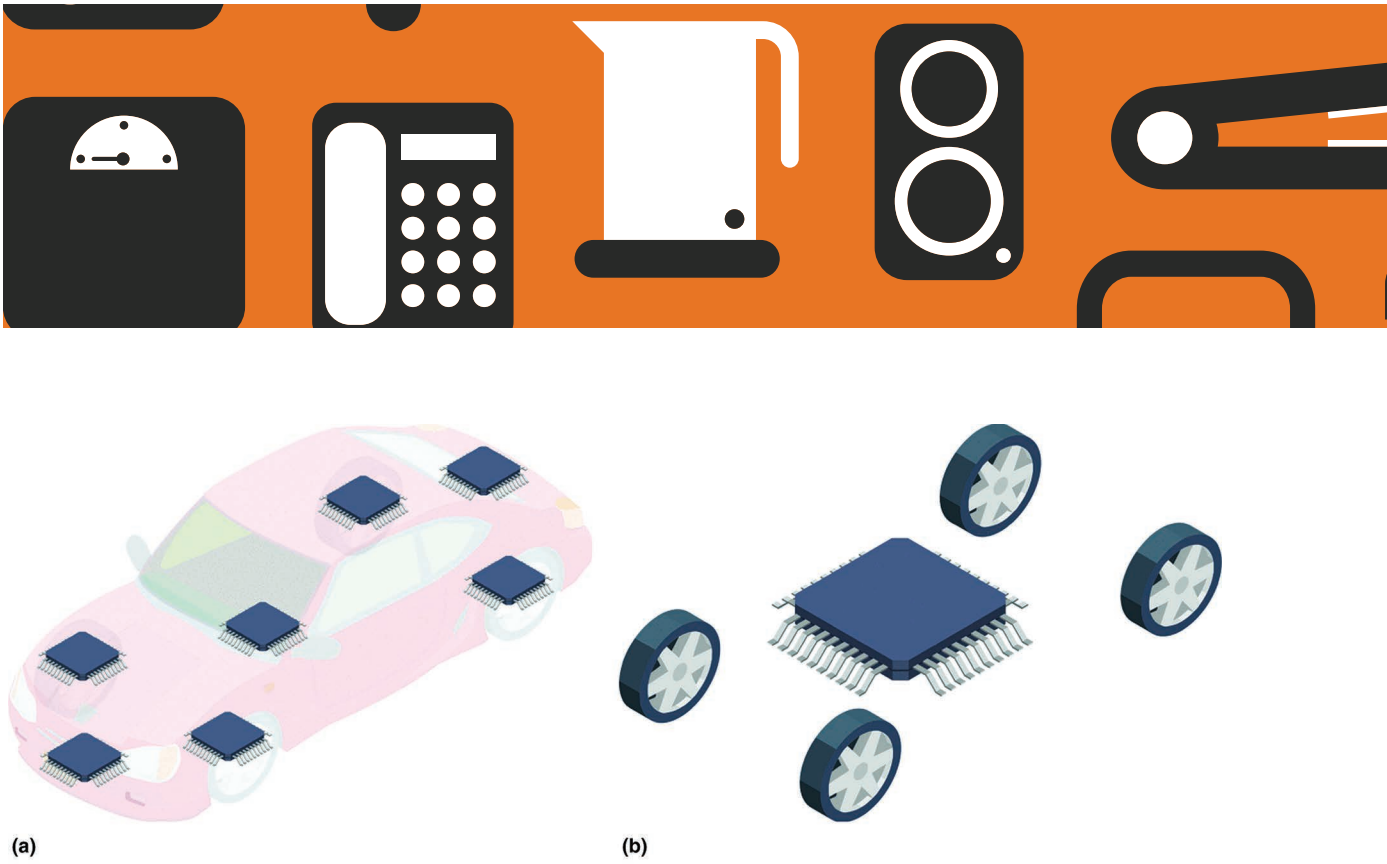


FIGURE 1. (a) The thing-centric evolutionary approach versus the (b) computing-centric revolutionary approach. (Source: Freepik.com.)

the development challenge. Empowering developers to create IoT solutions beyond prototyping (of toy-like applications) has received little attention and is largely ignored in research. IoT requirements outside the lab are diverse, and existing conventional solutions are the benchmark.

YOUR FOCUS: THING OR COMPUTER

We see two main ways of developing IoT solutions:

- *Thing-centric (evolutionary) approach*: You have a thing, you are good at making it (for example, a car), and now you add computing, sensors, actuators, and networking to provide new functionality (for instance, automated parking or self-driving) [Figure 1(a)].
- *Computing-centric (revolutionary) approach*: You have a computing platform with sensors, actuators, and networking that provides functionality that previously came from a thing. You aim to provide a superior performance and experience [Figure 1(b)].

There is no right or wrong approach. Combining both is helpful to create the best possible IoT device. Traditional manufacturers think thing-centrically, whereas computing professionals and disruptive developers favor the computing-centric approach. In both directions, software development is the key to creating the added value.

CHALLENGES IN THE REAL WORLD

Relevant applications on early smartphones (before iOS and Android) usually came preinstalled from the handset manufacturers. Application development was complicated, the access to hardware functions was limited, and applications worked well only on a small set of devices. Currently, the IoT seems similar. IoT applications are written for one specific environment (for example, smart kitchens, workshops with smart tools, or production monitoring). It is not realistically possible to deploy one software in an environment with different hardware or network infrastructures. Changes to the infrastructure or functionality lead to major alterations in the software and, often, the redevelopment of the applications. Developments

are costly; hence, making the IoT a viable business case requires generalization.

Our vision is to transfer the idea of smartphone apps to physical environments, such as rooms, floors, and whole buildings and production environments. An app that runs in one kitchen should run in any kitchen, independent of the specific devices supporting various brands. The key challenges are

THE INTERNET OF THINGS IS WIDELY DISCUSSED IN RESEARCH AS WELL AS IN INDUSTRY, AND EVERYONE AGREES ON ITS HUGE POTENTIAL.

- abstracting from devices, specific hardware, and networking protocols so that IoT applications handle resources they way classical operating systems (OSs) do
- striking a balance between custom development that is specific to an environment and the provision of prefabricated generic applications
- enabling increasing levels of customization, starting with apps that run in any environment and progressing to customer-specific applications and new drivers
- supporting operation with local connectivity only.

WHAT MAKES A THING "SMART?"

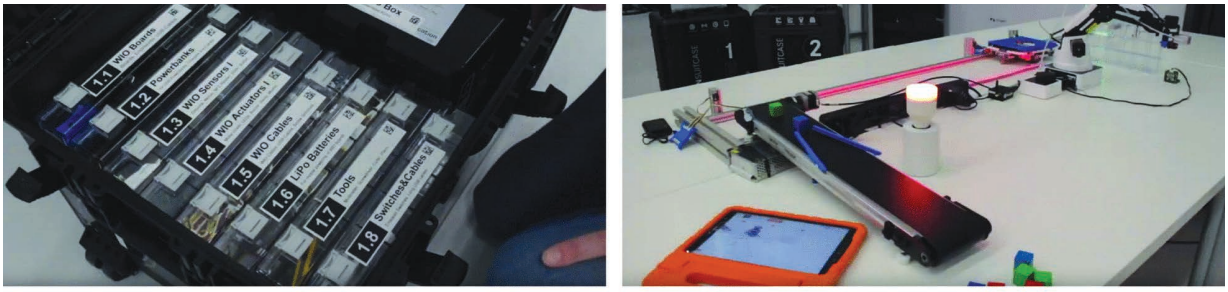
The obvious answer to the question of what makes a thing smart is processing, networking, sensors, and actuators. Creating smart things always occurs in the context of creating things. Designing embodiment is essential. In the following, we share our experience in the domain of smart furniture outside the lab. One thing is typically made up of several others: a table includes legs, a top, and connectors, and a cupboard consists of a body, boards, doors, and hinges. Which part do you make smart? Where do you add sensing and actuation, processing, and networking? For furniture, the connectors (including hinges, locks, and so forth) and accessories (such as handles) are the components that scale. For example, IKEA sells hundreds of different kitchen cupboards, wardrobes, and cabinets, but many of them share the same hinges. These components are also good for detecting interactions.

To decide where to put "smartness," domain experts have to experiment, exploring the added value that sensing and actuation can provide in different places. The research is hands-on, not theoretical. Developers and designers want to try different technologies and explore how they impact functions. This leads us from design thinking to design making.

FUNCTIONAL EXPLORATION: DESIGN MAKING

Design thinking has become a key activity when envisioning new products and services. Rapidly creating ideas has become very popular, and the barrier to enter is low. In design thinking, all participants feel that they can contribute to the solution. The typical outcomes are innovative concepts as minimalistic prototypes. In real-world environments, such as factory floors and hospitals, it is difficult to transform these ideas into a functional system that can be thoroughly evaluated and serve as a product. Design thinking is powerful for inspiration and effectively pruning "bad" ideas, but for positive evaluations, functional prototypes are required.

With design making, we empower teams to go beyond the idea and concept stages. The process and goals are similar, but they aim to create fully functional prototypes that enable realistic evaluations. We support prototyping and evaluation through technology and a facilitator. The design-making box includes a set of typical components (specific to use cases, for example, factory automation, smart buildings, health applications, and furniture) that include hardware, wireless networking, software, and business logic (see Figure 2). A facilitator supports the team during prototyping, helping to put the computing technology into the thing. In our experience, most participants can easily program an IoT system by describing what functionality they want to a person but hesitate to use any programming interface (no matter how simple). The key is that the functionality can be created immediately. Participants program by telling their ideas to the facilitator, who performs the coding during the conversation. Typically, the functionality is not complex, but it is required to experience the smart object during the evaluation.



(a) (b)

FIGURE 2. (a) A prototyping box for factory automation with different components and (b) a created prototypical model system developed in a design-making workshop that could be transferred with the same IoT software to a full-scale deployment. For a 2-min video showing this in detail, visit https://www.youtube.com/watch?v=Wyoz0Z-5gf0.

A person facilitating the development lowers the barrier to entry and levels the ability to program in interdisciplinary teams. He or she provides support to envision, prototype, and try ideas for new products and services through quick iterations (20 or more in a single-day workshop are implemented and tested). It is crucial to create functional prototypes and enable teams to understand the required steps and trajectory to move the mock-ups toward useful services or products. Once experienced, this process empowers teams to fundamentally rethink their products.

COMPUTER SCIENTISTS ARE NOT UNIVERSAL EXPERTS

The design-making approach jump-starts interdisciplinary innovation. The provided tools are crucial since they limit what participants will imagine and try. Our guiding principle is to have no barriers to entry and, at the same time, enable complex issues to be implemented. Traditional embedded-systems development tools can be used only by software developers. In research, it seems that computer scientists create smart things, but what they actually make are proofs of concept for smart things. Creating real products poses many challenges for which skills beside computer science are essential.

THINGOS: OS AND DEVELOPMENT SUPPORT

During our research and work with customers, we realized that IoT OSs and development support are closely linked and that tight integration is important. This insight was new for us in the IoT context, but looking back at UNIX and C, it is not surprising. We have to acknowledge that potential development partners are experts at envisioning and making things but have little knowledge of computer science and are often uninterested in learning about software development. With the ThingOS platform, we created an OS and

EMPOWERING DEVELOPERS TO CREATE IOT SOLUTIONS BEYOND PROTOTYPING (OF TOY-LIKE APPLICATIONS) HAS RECEIVED LITTLE ATTENTION AND IS LARGELY IGNORED IN RESEARCH.

web-based environment to ease IoT development for interdisciplinary teams; see Figure 3 and Kubitza.[3] The most important design choices were

- creating a technology-neutral abstraction layer for sensors, actuators, and network protocols
- requiring no installation for the IoT-application developer
- using JavaScript as the programming language to open the IoT to the web-development community
- providing an integrated development environment (IDE) that is adaptive and context-aware
- working with IoT-application templates as always-functional starting points
- having an IoT app store, enabling the quick deployment of applications

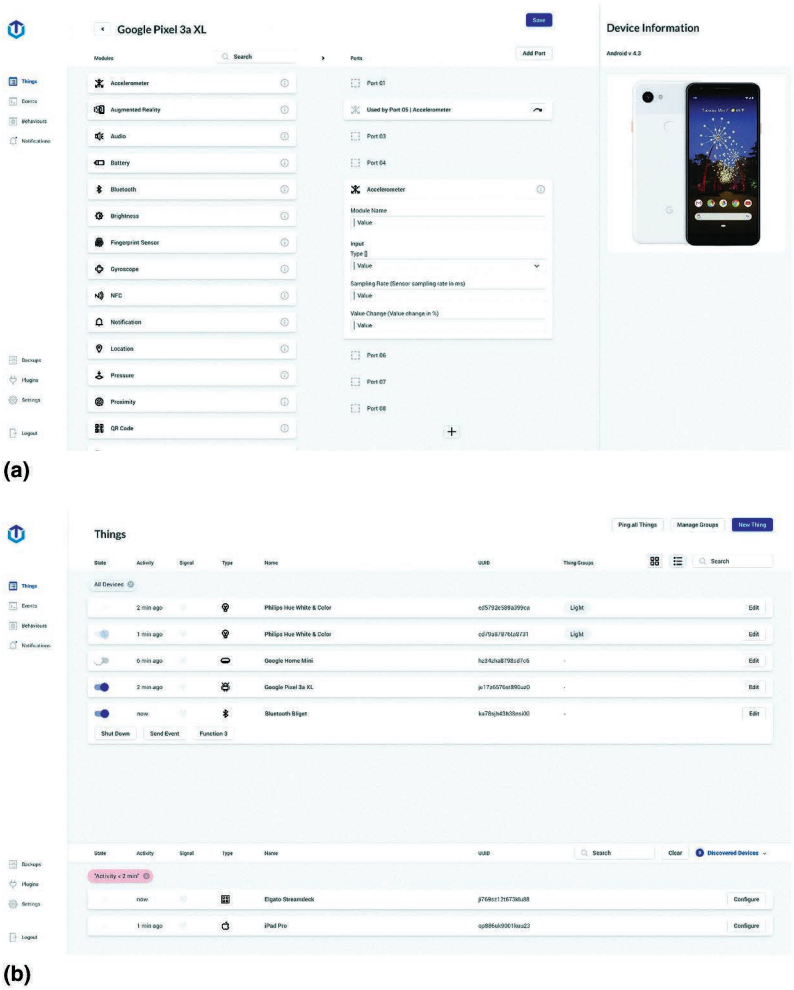


FIGURE 3. (a) The development view once a phone is added to the IoT system. Every property can be addressed by other IoT components through JavaScript calls. (b) All active and programmable devices in the IoT environment are shown. The "edit" button opens a dialog box to manipulate them.

› providing security in the background.

These decisions were guided by our goal to broaden the set of people who can create IoT applications. One key was to eliminate the installation of tool chains, drivers, and IDEs, which are typical for embedded development. Using JavaScript enabled us to include web and app developers who have a hard time with traditional IoT-development environments but are keen on novel applications and services. The development environment's user interface (see Figure 3) supports different views and levels of abstract. The abstraction layer that understands the capabilities of sensors, actuators, devices, and protocols facilitates transferring applications between different environments. The IDE also supports the exploration of devices that enter the vicinity and are programmable.

The typical steps for programming an IoT-System are

› connect to the local network provided by one of the ThingOS devices
› open the online IDE based on web technologies in the browser
› interactively explore the available devices (optional)
› chose an application template for the required functionally and run it
› configure the functionally to the specific environment and use case (optional)
› add functionality by programming components (optional)
› provide applications to others through the app store (optional).

To move the IoT from research to real-world deployment, software and application development has to be simplified for a broad base of programmers. Enabling efficient collaboration between the people who make things and those who make software is key. Making a clever physical design of a thing will massively ease the development of its functionality (for instance, the placement of sensors and the physical shape and size). Development tools must support collaboration. The IoT on a case-by-case basis will not work commercially, hence

1. consider, explore, and select which part of the thing to make smart to create the most viable product and gain from efficiencies of scale
2. make applications that are independent

of the actual hardware to distribute software-development costs across many instances.

In our work, we have achieved this by extending design thinking into design making, through an OS that supports abstraction in heterogeneous settings and by creating an environment to support a broad range of developers.

REFERENCES

1. M. Beigl, H.-W. Gellersen, and A. Schmidt, "Mediacups: Experience with design and use of computer-augmented everyday artefacts," *Comput. Netw.*, vol. 35, no. 4, pp. 401–409, 2001. doi: 10.1016/S1389-1286(00)00180-8.
2. B. Begole, *Ubiquitous Computing for Business, Video Enhanced Edition: Find New Markets, Create Better Businesses, and Reach Customers around the World 24-7-365*. Upper Saddle River, NJ: FT Press, 2011.
3. T. Kubitza and A. Schmidt, "meSchup: A platform for programming interconnected smart things," *Computer*, vol. 50, no. 11, pp. 38–49, 2017. doi: 10.1109/MC.2017.4041350.
4. M. Weiser, "The computer for the 21st century," *Sci. Amer.*, vol. 265, no. 3, pp. 94–105, 1991. doi: 10.1038/scientificamerican0991-94.

THOMAS KUBITZA is a doctoral candidate at the University of Stuttgart, Germany, and chief executive officer of ThingOS. Contact him at Thomas.Kubitza@thingos.io.

PATRICK BADER is a doctoral candidate at the University of Stuttgart, Germany, and chief technology officer of ThingOS. Contact him at Patrick.Bader@thingos.io.

MATTHIAS MÖGERLE is the chief operating officer of ThingOS. Contact him at Matthias.Moegerle@thingos.io.

ALBRECHT SCHMIDT is a computer science professor at the Ludwig Maximilian University of Munich in Germany. Contact him at Albrecht.Schmidt@ifi.lmu.de.

EDITOR: Schahram Dustdar, dustdar@dsg.tuwien.ac.at

DEPARTMENT: INTERNET OF THINGS, PEOPLE, AND PROCESSES

This article originally appeared in IEEE Internet Computing vol. 24, no. 4, 2020

6G Vision: An AI-Driven Decentralized Network and Service Architecture

Xiuquan Qiao, *State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications*

Yakun Huang, *Beijing University of Posts and Telecommunications*

Schahram Dustdar, *Vienna University of Technology*

Junliang Chen, *Beijing University of Posts and Telecommunications*

Recently, following the rapid commercial deployment of 5G networks, next-generation mobile communication technology (6G) has been attracting increasing attention from global researchers and engineers. 6G is envisioned as a distributed, decentralized, and intelligent innovative network. However, existing application provisioning is still based on a centralized service architecture, ubiquitous edge computing, and decentralized AI technologies have not been fully exploited. In this article, we analyze the problems faced by existing centralized service provisioning architecture, and propose design principles for a decentralized network and service architecture for a future 6G network. Finally, we discuss several open research problems to inspire readers to address these.

Due to the large number of commercial applications of 5G networks worldwide, potential 6G technologies are attracting attention from both academia and industry. Although 5G has achieved significant improvements in terms of communication performance, it remains difficult to meet demand for more intelligent communication in terms of information speed, multidomain coverage, artificial intelligence (AI), and security.[1] Recently, several governments have launched 6G projects to explore the requirements and key technologies of the next-generation mobile communication network. However, existing visions and discussions of 6G mainly focus on innovative wireless communication technologies, mobile edge computing (MEC) and AI,[2] and there is a lack of deep and innovative insights into networking and service provisioning mechanisms. It is therefore necessary to create a blueprint for a disruptive service provisioning mechanism in future 6G networks.

Following the continuous evolution of 5G networks, 6G is envisioned as an ultrabroadband, ultra-low-delay, full-dimensional coverage (terrestrial, aerial, space, and maritime domains) ubiquitous intelligent network with native AI and security. 6G will seamlessly integrate communication, computing, control, caching, sensing, positioning, and imaging features to support various Internet of Everything (IoE) applications. In contrast to 5G, 6G will evolve from "human-machine-thing" interactions to "human-machine-thing-genie" interaction,[3] and will become a highly autonomous and intelligent ecosystem. Several innovative applications will become reality, such as holographic communications, brainwave-machine interaction applications, tactile mixed reality (MR) experiences (including vision, hearing, smell, taste, and touch), and high-precision manufacturing.[4] With the continuous maturity of AI technology and the rapid reduction in related hardware costs, increasing numbers of devices will have native

Digital Object Identifier 10.1109/MIC.2020.2987738
Date of current version 9 September 2020.

 Published by the IEEE Computer Society

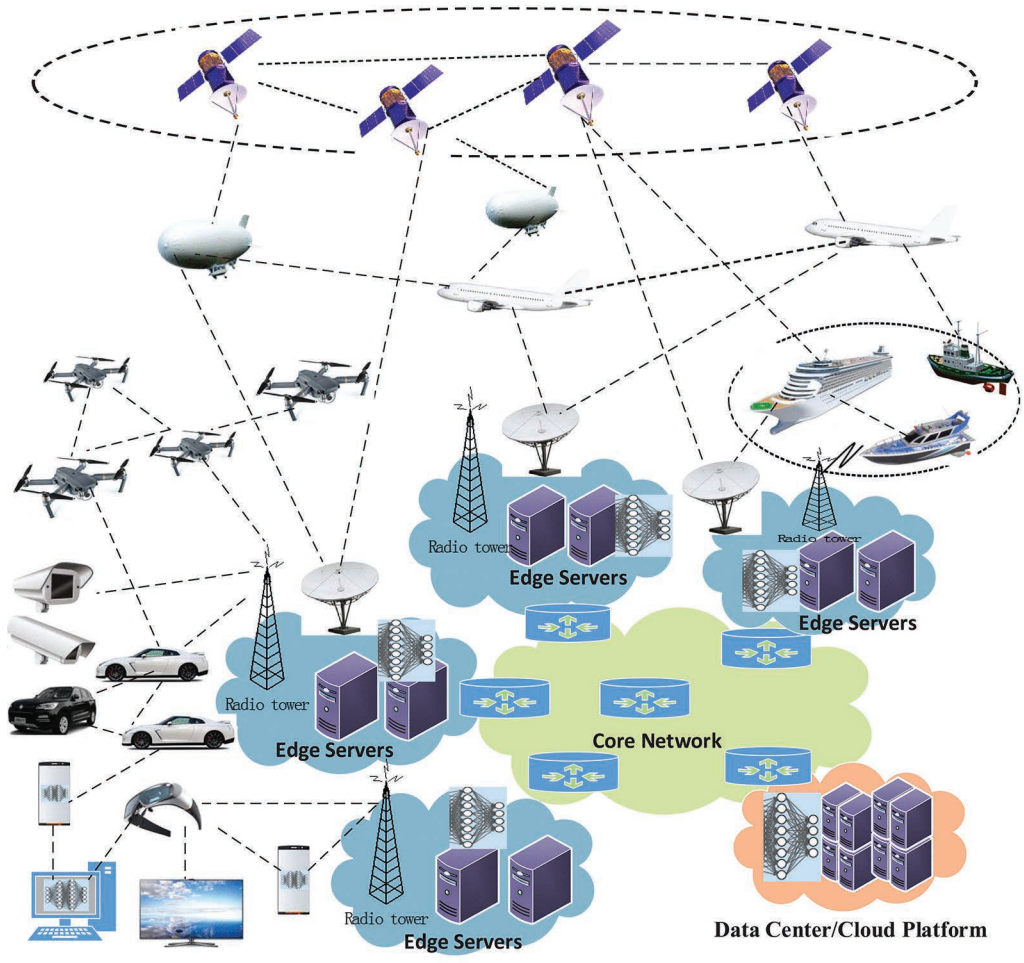


FIGURE 1. Vision for 6G: Integrated, ubiquitous, intelligent, and decentralized.

AI functions, such as smartphones, AR/VR glasses, smartwatches, headsets, TVs, loudspeakers, and vehicle-mounted devices. Based on the user's movements, these ubiquitous devices will dynamically and autonomously collaborate with each other to achieve better user experience. 6G will be a highly dynamic, autonomous, decentralized, and intelligent network in which network nodes will collaborate autonomously and dynamically, user data will be stored in the network in a decentralized way, and services will migrate on demand. This new ubiquitous, decentralized, AI-driven flat 6G network needs a corresponding new decentralized service provisioning mechanism.

However, although MEC has pushed computing closer to the user, and the device-to-device (D2D) model has enabled nearby mobile devices to communicate with each other directly, the 5G network remains a centralized network and service provisioning architecture in terms of the data storage and access, service running mechanism, and application protocols used. The service provisioning mechanism for the existing 5G network has not changed a great deal compared with the 4G network. We therefore need to design a novel service provisioning mechanism to meet the tremendous shift toward decentralization in the future 6G network.

In this article, we first explain the decentralized trends of the 6G network and its new characteristics. We then analyze and discuss the problems faced by existing centralized service provisioning. Finally, we propose some design principles for a future decentralized 6G service provisioning mechanism, and discuss open research issues related to this new decentralized computing paradigm.

6G VISION AND ITS NEW CHARACTERISTICS

Although 6G is not yet the subject of a global consensus, some potential new characteristics and trends have been widely discussed. In this section, we present a comprehensive vision of the future 6G network from multiple perspectives, as shown in Figure 1.

1. *Network coverage view:* With the expansion of human activities, the existing closed and vertical dedicated networks and terminals cannot meet the demand for ubiquitous mobile communication anytime and anywhere. Unlike the previous 1G to 5G networks, 6G will extend mobile communication coverage in an unprecedented way from terrestrial areas to the aerial, space, and maritime domains. A ubiquitous, integrated, multidimensional, and full-coverage mobile communication network will be available anywhere in the 6G era. Everything (including real-world objects and digital objects in virtual worlds) will be able to be connected with everything else, and a new IoE distributed ecosystem will be established based on this all-round connectivity.
2. *Capability convergence view:* With the enhancement of terminal capabilities, the large-scale deployment of MEC infrastructures[5] and the widespread applications of IoE, communication is no longer the only goal of the 6G network. The convergence of communication, computing, control, storage, and sensing capabilities will be the new trend in the 6G network. Based on these capabilities, increasing numbers of terminals and network nodes will become intelligent, autonomous information processing entities and act as both information producers and consumers.
3. *Interaction space view:* Based on the characteristics of eMBB, mMTC, and uRLLC, 5G has begun to support "human-machine-thing" interactions, which bridge the domains of cyber space, physical space, and society. 6G will further deepen and expand interaction spaces. Following the advances in wireless brain–computer interaction (BCI) technologies, the use of consciousness-based communication and control will create some new application scenarios. For example, brain–computer interfaces may be used to interact with ambient smart devices such as XR glasses, TV sets, or loudspeakers. 6G will also evolve from current "human-machine-thing" interactions to "human-machine-thing-consciousness" interactions. The real world and virtual world will be perfectly integrated, and the era of augmented reality (AR)/MR is imminent,[6] in which physical and digital objects coexist and interact in real time (i.e., dual worlds).
4. *AI view:* In the initial design phase of the 5G network, AI technologies were not sufficiently mature to act as an enabling technology. However, following rapid advances in big data, cloud computing, neural network, and dedicated chip technologies in recent years, AI technologies have begun to be employed in 5G network management, smart mobile phones, and various applications in a patchwork way. AI is considered the most innovative enabling technique for 6G, and will be an innate feature of the network from the application layer to the physical layer. In the 6G era, end devices with various AI capabilities will seamlessly collaborate with a variety of edge and cloud resources.[7] With the maturity of AI technology and the reduction in AI hardware costs, the number of smart end devices in use in daily life will constantly increase. Decentralized and collaborative AI services among distributed end devices and network nodes will also become a trend in 6G.
5. *Network architecture view:* As mobile communication networks have been updated from 1G to 5G over the decades, they have gradually evolved from a closed dedicated network to an open converged network based on general IT technologies. Network architectures are becoming increasingly flat, and the original customized hardware appliances used for each network function have been replaced by general IT devices and software platforms. This is particularly true for the 5G network, in which software-defined networks, network function virtualization, and network slicing technologies

have been fully employed. To a certain extent, network carriers can utilize different software and processes in a flexible way on top of standard high-volume servers, switches, and storage devices, and can customize different virtual networks in order to meet demand arising from differentiated application scenarios such as high bandwidth, low latency, or massive numbers of connections. In addition, MEC and D2D communication technologies promote the migration of computing and service processing capability from the cloud platform to the network edges.[8] With enhancements to smart user and network equipment, edge or fog computing will become as important as cloud computing. Increasing numbers of local communication clusters will be formed dynamically and autonomously, and applications will be processed both directly and locally. The network edge will be highly decentralized, and will take on some of the functions of the core network and cloud platform. In this model, the network edge is no longer simply an access network, but comprises a large number of ubiquitous, autonomous local networks that can integrate communication, computing, control, storage, and sensing capabilities. The network edges and core networks will have a more peer-to-peer structure, and in general, the network architecture will be much flatter and more flexible.

6. *Application architecture view:* Based on the above analysis, it can be seen that 6G will be a ubiquitous, distributed, decentralized, and intelligent innovative network. The existing application provisioning architecture mainly adopts B/S or C/S architectures, which were originally designed for centralized network. Clients often interact with centralized, specific application servers, and database servers in order to deal with user requests. In contrast, 6G will become more decentralized due to the long-term evolution of the 5G network. The application provisioning architecture of 6G will also therefore change significantly to cater to this shift. In the future 6G network, peer-to-peer and ad hoc networks will become more pervasive and popular, and current cloud-based serverless application provisioning architecture will gradually evolve toward a decentralized peer-to-peer application provisioning architecture. User data will be stored on a decentralized peer-to-peer network, and business processing logic will be divided into stateless and independent fine-grained services that can be migrated and run on any network node on demand.

EXISTING CENTRALIZED APPLICATION PROVISIONING MECHANISM AND RELATED ISSUES

Since the aforementioned vision for 6G and its new characteristics differ markedly from the existing 5G and 4G mobile networks, it is necessary to analyze and discuss the problems faced by current centralized service provisioning. After 40 years of development, the existing centralized application provisioning architecture has gradually become unsuitable for the application development needs of 6G.

1. *Limitations on B/S or C/S application architecture:* Most existing applications employ B/S or C/S application provisioning architecture, which was originally designed for the era of thin clients and powerful servers. An application is provided by a collaboration between user devices and edge/cloud servers. In a centralized architecture, applications are highly dependent on dedicated cloud servers, and information storage and business logic are all provided by servers. This architecture gives rise to the high computing, storage, and bandwidth costs on the server side. With the emergence of MEC in 5G, some of these application functions can now be offloaded to edge servers, and a "terminal+edge+cloud" collaboration computing architecture is being developed. However, 5G applications are only beginning to allow for distributed computing, let alone a decentralized computing model. Following significant advances in hardware and software, the capability of 6G terminals will be further improved, and some tasks will be processed by the local user terminal or by

collaborating with ambient devices or edge/cloud servers. It is therefore necessary to explore a novel application architecture to support this ubiquitous decentralized computing paradigm.

2. *Disadvantages of the centralized data model:* In the existing centralized application architecture, data are generally stored in specific cloud servers or terminal devices, and are cached on edge servers or CDN networks. Data storage and access are all controlled by a centralized authority such as Yahoo, Facebook, or YouTube. This centralized data model results in some potential problems such as censorship, privacy, data leakage, and data control rights. For example, if the central point is hacked, the entire user database is at risk. In addition, the trust problem of the centralized authority is often challenged. In fact, some Internet service providers use the data for their own benefit, such as selling it to advertising companies, meaning that the privacy and security of user data are not well protected.
3. *End-to-end application protocols:* Due to the use of a centralized data storage and service operation mechanism, most of the existing application protocols are based on an end-to-end communication model rather than a peer-to-peer model, and client requests need to be routed to dedicated application servers to be processed. Existing application protocols (such as HTTP) were originally designed for B/S or C/S application architectures, and are not suitable for this new dynamic and opportunistic form of connectivity and the ubiquitous edge and decentralized computing paradigm of the 6G network. In the era of 6G, application protocols will enable data access and service coordination on a peer-to-peer basis over a ubiquitous distributed network.
4. *Tight coupling of user data with specific applications:* Following the rapid development of the mobile Internet, increasing numbers of people rely on service provisioning by a few Internet giants such as Yahoo, Google, Facebook, Twitter, and WeChat, and the centralization of information has become more pronounced, with services and content being gradually aggregated by a few Internet oligarchs. This model of centralized information organization creates many information islands, and in this paradigm, users have no right to control their data. User data are tightly coupled with specific apps, and data utilization across different apps is often restricted due to commercial competition. These centralized information islands have gradually come to hinder the free dissemination of information.
5. *Shortcomings of centralized AI:* In recent years, due to the development of powerful cloud computing capability and big data, AI has become increasingly widespread. However, existing AI is mainly organized using a centralized application model. More specifically, massive training datasets are very valuable assets to enterprises. Training datasets and the creation and training of models are also controlled by a small number of large organizations, which increases the gap between large companies with access to large, labeled datasets, and smaller companies. At the same time, the centralization of model training requires the transmission of data from end devices to cloud servers, often resulting in high transmission and computing costs and giving rise to user privacy protection issues. Furthermore, current AI models are always deployed on either cloud/edge servers or end devices using a centralized operation model, without allowing for the efficient utilization of resources such as ubiquitous distributed network nodes.

DECENTRALIZED APPLICATION PROVISIONING ARCHITECTURE FOR 6G

Decentralized Application Provisioning Mechanism

Based on the above analysis, we expect that the application provisioning mechanism in 6G will change significantly from the existing centralized application mechanism, as shown in Figure 2. Some design principles for a future decentralized 6G application provisioning mechanism are presented below.

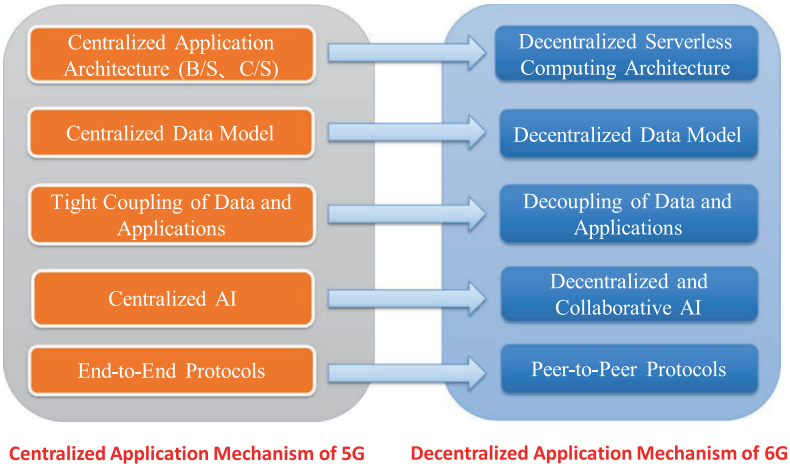


FIGURE 2. Evolution of 6G decentralized application provisioning.

1. *Decentralized serverless computing architecture:* In the future 6G network, the communication, computing, and storage capabilities of network nodes will be greatly enhanced. The traditional client–server boundary will be eliminated, and each network node (including various terminals, base stations, gateways, routers, servers, etc.) will act not only as an information publisher but also an information consumer. In 6G, the decentralization of the network infrastructure will be realized, and the whole network will become a service running environment. A microkernel-based distributed operating system will become popular, and this will be adaptively deployed on various types of hardware, including smartphones, AR/VR glasses, smart displays, wearable devices, in-car entertainment systems, and other IoT devices. The service environment will gradually expand from the existing cloud infrastructure to the network edge and ubiquitous end devices. The overall business logic will be composed of multiple fine-grained micro services; these will not be deployed on dedicated servers, and will be able to migrate to any network node on demand. Front-end client applications will resolve the application description file and invoke the related service components directly.
2. *Decentralized data model:* With the large-scale deployment of edge/fog computing, it is now possible to establish a ubiquitous decentralized storage infrastructure to address the problems faced by the existing centralized cloud storage model. Compared with the centralized data model, data will no longer be stored on specific servers, but will be distributed over a peer-to-peer network. This decentralized data model promises even greater advantages, such as efficient scalability, reliability, privacy, and data immutability. Since all the data are distributed among the different network nodes, decentralized data networks are better able to withstand massive user requests distributed between the nodes, since the pressure of these requests no longer falls on a few computers but on the network as a whole. This scheme can also deal with DDoS attacks more effectively. In addition, a decentralized data model can reduce dependence on the infrastructure of specific Internet giants, and will facilitate the disintermediation of the mobile Internet.
3. *Decoupling of data and applications:* In order to return data control rights to the users themselves, it is necessary to decouple user data from specific silo applications. In the future 6G network, user-generated data such as videos, social media posts, health data and tracking information, etc., will be completely controlled by the users themselves. These data will be stored in a decentralized P2P network in which users have the right to authorize certain applications to manipulate their data, and decide which users to share these data with. This new mechanism will facilitate information sharing and dissemination among different apps. For example, a user's profile information can be shared by different apps, thus avoiding the need for each app system to save a copy of the user's information. This scheme can also avoid data leakage by third-party application providers.
4. *Decentralized and collaborative AI:* In the 6G

era, each network node can store and process data and can autonomously communicate and seamlessly interact with other ambient devices. Following the development of ubiquitous computing infrastructure, existing centralized AI will gradually evolve to a decentralized and collaborative model. In contrast to traditional centralized AI, in which all data samples are uploaded to dedicated cloud servers, the decentralized approach will train a model across multiple decentralized edge devices or servers with local data samples; this will be done without sharing data, and instead parameters will be simply exchanged between these local models at a certain frequency to generate a global model. This approach can efficiently avoid the need for transmission and centralized storage of training data, and can also address several critical issues such as data privacy, data security, data access rights, and heterogeneity of data. In addition, aided by advances of lightweight model technology, AI models can be deployed on any devices, from mobile phones to massive numbers of IoT devices. AI will be able to run, train, and even make decisions on local devices in this type of decentralized network. The autonomous collaboration of multiple network nodes is controlled by a distributed group of intelligent agents, which will be able to solve complex planning and decision making problems.

Comparison of Centralized and Decentralized Application Solutions

In Figure 3, we use a mobile search application as an example to illustrate the significant changes in existing centralized and future decentralized application provisioning mechanisms.

Figure 3(a) shows the existing centralized Google mobile search application mechanism. It can be seen that the application is mainly processed via a collaboration between the mobile browser and dedicated cloud application servers, and the network is only responsible for the transmission of information. When the user inputs the Google URL, the mobile browser will query the IP address corresponding to the URL using the DNS service and sends the Web page request to Google's dedicated cloud server, which returns the search page. After the user inputs some search content ("Titanic" in this example), the browser will send this search HTTP request to the Google search server, which generates the results pages. The user may then click a link to to play "Titanic" on the YouTube website. The browser will obtain the content from the YouTube cloud servers or nearby CDN networks.

In contrast to this existing centralized mechanism, a decentralized mechanism will be very different in terms of the data storage, server architecture, and communication protocols used, as shown in Figure 3(b). There are no dedicated cloud application and database servers, and the whole network acts as a decentralized communication, computing, and storage infrastructure. The browser will get the search Web page from a distributed file system in a peer-to-peer way, using a distributed hash table. Segments of Web pages may be located on nearby mobile phones, PCs, edge, or cloud server nodes. After the user inputs the search content "Titanic," the AI-enhanced browser can process and analyze the search input by itself, using a lightweight AI model for natural language processing, and can collaborate with ambient AI-enhanced devices to generate the search results pages. When the user clicks the YouTube link in the search results page, the media player will fetch content segments from a distributed peer-to-peer network.

CONCLUSION AND OPEN ISSUES FOR FUTURE RESEARCH

Decentralization has become a likely trend for a future 6G network. In this article, we mainly focus on the potential disruptive changes to application provisioning mechanisms in the 6G era. By analyzing the issues faced by the existing centralized infrastructure, we propose some insights for a decentralized application provisioning mechanism for the future 6G network.

However, until now, there have been no comprehensive discussions of 6G from this perspective. Several issues are still open, and we describe these here to provide readers with inspiration to address these issues.

1. *Decentralized operating system for ubiquitous computing:* In view of the IoE application scenarios envisaged for 6G, it will be necessary to develop a decentralized operating system for

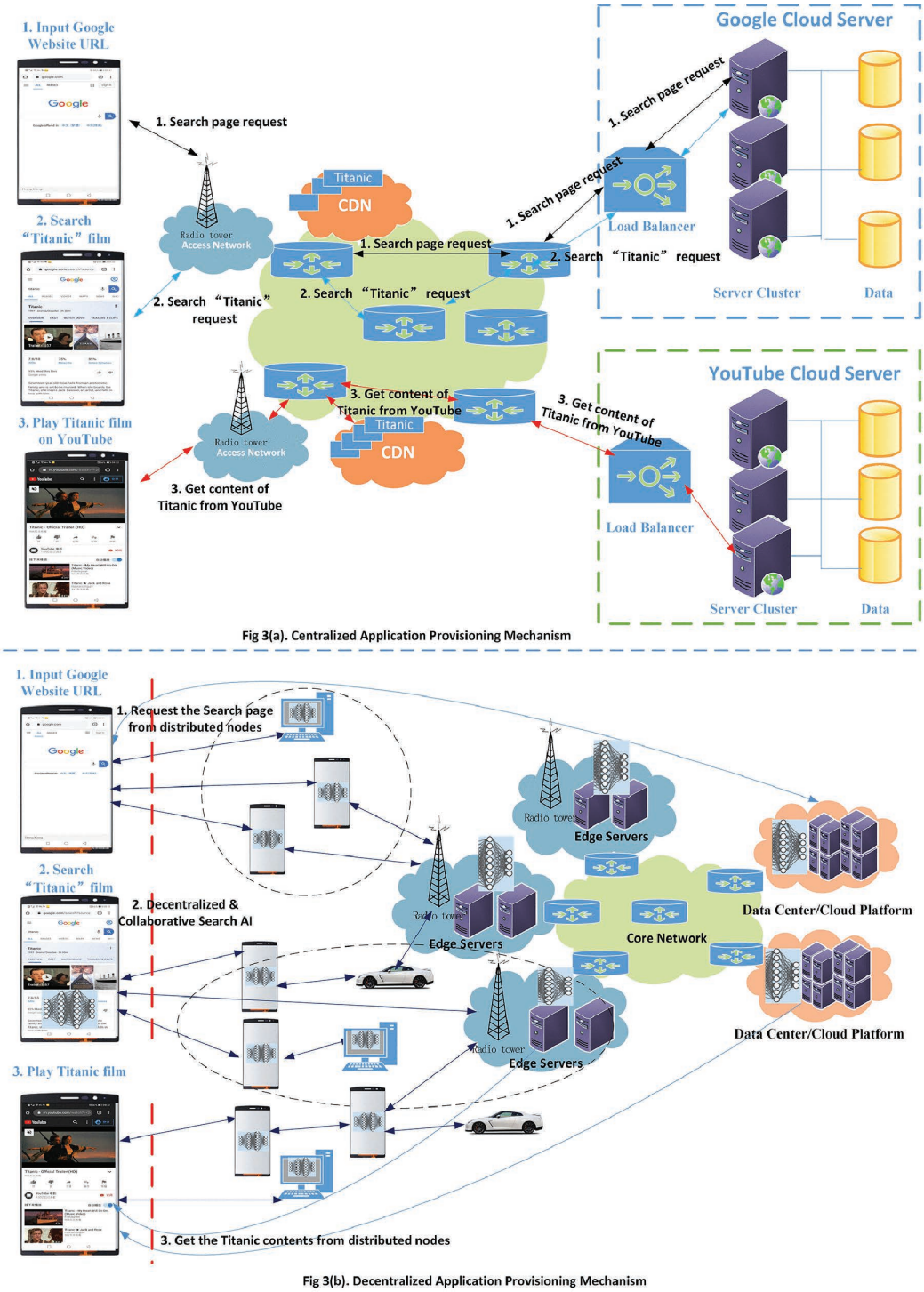


FIGURE 3. Differences between existing centralized and future decentralized application provisioning approaches.

a dynamic, autonomous, collaborative network, which can efficiently enable peer-to-peer communication, decentralized data storage and access, on-demand service migration and deployment, and flexible adaptation of heterogenous devices such as servers, mobile phones, TV sets, vehicle-mounted systems, and other IoT devices.

2. *Collective decision making by decentralized AI:* Decentralized AI has become one of the most promising trends for the next phase of AI. With the aid of D2D and MEC, decentralized and collaborative AI services among distributed network nodes will become an important enabling technology for 6G. The issues of how to integrate these scattered AI capabilities over distributed nodes and find the optimal combination of services to provide the best experiences for users are worthy of in-depth study and exploration. This will involve coordination between and decisions by multiple intelligent agents, and thus constitutes a collective decision-making issue.
3. *Disruptive influences of the decentralization network and service model:* A decentralization model will bring about disruptive impacts on existing application provisioning mechanisms in terms of business models, products, services, and ecosystem roles. It will inevitably weaken the authority of central entities and will affect the commercial interests of the existing Internet giants. At the same time, it will also create impacts on the infrastructure governance of telecom network operators. The problem of how to effectively activate and coordinate multiple stakeholders (individual users and other enterprises) to participate in the provisioning of network resources in a future 6G ecosystem is a newly emerging issue. It is therefore necessary to explore the strong potential influence on the operation of network infrastructure.

ACKNOWLEDGMENT

This work was supported by the National Key Research and Development Project under Grant 2018YFE0205503.

REFERENCES

1. Z. Zhang et al., "6G wireless networks: Vision, requirements, architecture, and key technologies," *IEEE Vehicular Technol. Mag.*, vol. 14, no 3, pp. 28–41, Sep. 2019.
2. K. B. Letaief, et al., "The roadmap to 6G: AI empowered wireless networks," *IEEE Commun. Mag.*, vol. 57, no. 8, pp. 84–90, Aug. 2019.
3. P. Zhang, et al., "Technology prospect of 6G mobile communications," *J. Commun.*, vol. 40, no. 1, pp. 141–148, 2019.
4. B. Zhang, et al., "6G technologies: key drivers, core requirements, system architectures, and enabling technologies," *IEEE Vehicular Technol. Mag.*, vol. 14, no. 3, pp. 18–27, Sep. 2019.
5. M. Gusev and S. Dustdar, "Going back to the roots—The evolution of edge computing, an IoT perspective," *IEEE Internet Comput.*, vol. 22, no. 2, pp. 5–15, Mar./Apr. 2018.
6. X. Q. Qiao, et al., "Web AR: A promising future for mobile augmented reality—State of the art, challenges, and insights," *Proc. IEEE*, vol. 107, no. 4, pp. 651–666, Apr. 2019.
7. X. Wang and Y. Han, "In-edge AI: Intelligentizing mobile edge computing, caching and communication by federated learning," *IEEE Netw.*, vol. 33, no. 5, pp. 156–165, Sep.–Oct. 2019.
8. X. Q. Qiao, et al., "A new era for web AR with mobile edge computing," *IEEE Internet Comput.*, vol. 22, no. 4, pp. 46–55, Jul./Aug. 2018.

XIUQUAN QIAO is a full professor with State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China. His main research interests focus on 5G/6G networks, augmented reality, edge computing, and services computing. Contact him at qiaoxq@bupt.edu.cn.

YAKUN HUANG is a Ph.D. candidate at Beijing University of Posts and Telecommunications. Contact him at hyk_it @foxmail.com.

SCHAHRAM DUSTDAR is a full professor of computer science (Informatics) with a focus on Internet Technologies heading the Distributed Systems Group, TU Wien, Austria. He is a member of the Academia Europaea: The Academy of Europe. He is a Fellow of the IEEE. Contact him at dustdar @dsg.tuwien.ac.at.

JUNLIANG CHEN is a professor of Beijing University of Posts and Telecommunications, Beijing, China. He is a member of both the Chinese Academy of Sciences and the Chinese Academy of Engineering. He is a Senior Member of the IEEE. Contact him at chjl@bupt.edu.cn.

EDITOR: Mike Potel, potel@wildcrest.com

DEPARTMENT: APPLICATIONS

This article originally appeared in IEEE Computer Graphics and Applications vol. 40, no. 1, 2020

Turning a Smartphone Selfie Into a Studio Portrait

Nicola Capece, *Università della Basilicata, Dipartimento di Matematica, Informatica ed Economia*

Francesco Banterle, Paolo Cignoni, and Fabio Ganovelli, *Consiglio Nazionale delle Ricerche Istituto di Scienza e Tecnologie dell'Informazione*

Ugo Erra, *Università della Basilicata, Dipartimento di Matematica, Informatica ed Economia*

We introduce a novel algorithm that turns a flash selfie taken with a smartphone into a studio-like photograph with uniform lighting. Our method uses a convolutional neural network trained on a set of pairs of photographs acquired in a controlled environment. For each pair, we have one photograph of a subject's face taken with the camera flash enabled and another one of the same subject in the same pose illuminated using a photographic studio-lighting setup. We show how our method can amend lighting artifacts introduced by a close-up camera flash, such as specular highlights, shadows, and skin shine.

Photographs taken with mobile devices are nowadays predominant on the Internet, including the web-based services dedicated to professional photography such as Flickr and 500px. This is due to the steady improvement of built-in digital cameras in smartphones, which has made them a default choice of many for taking pictures. Under favorable lighting conditions, smartphone picture quality has reached that of digital reflex cameras, but smartphones are not able to capture artifact-free images in low-light conditions. This is due to their sensors' size, a constraint that is not straightforward to solve because of the little room available in modern phones. Therefore, taking pictures in low light often triggers the camera flash, which is typically a low-power LED flash mounted side by side with the camera lens that produces several artifacts. Selfies are one of the most common forms of photographs taken with a smartphone. This practice consists of taking a picture of one's face by holding the phone in one hand or by using a so-called "selfie stick." Also, selfies are often low-light flash photographs, which is an unfavorable combination that produces images with specular highlights, sharp shadows, and flat and unnatural skin tones. In this article, we explore the possibility of turning flash selfies into studio portraits by employing a convolutional neural network (CNN). Doing so is a challenge for three reasons. First, it involves handling both global and local discriminant features, e.g., skin tone and highlight, respectively. Second, it needs to match how humans expect an image to look when a flash is not used. Finally, these two requirements have to be met in the domain of human faces, where people are very good at detecting any type of inconsistencies.

Smartphone flash selfies share several common traits for a well-defined subdomain of photos that our proposal is able to address: they are three-quarter or front single-face portraits, taken at close range, with a single flash colocated with the camera lens. Our approach is based on CNN training with a set of pairs of portraits (see Figure 1): one image with smartphone flash and one with photographic studio lighting (a "ground truth" image). Each pair is taken as simultaneously as possible, to keep the pose of the subject similar. The flash correction problem applies

Digital Object Identifier 10.1109/MCG.2019.2958274
Date of current version 6 January 2020.

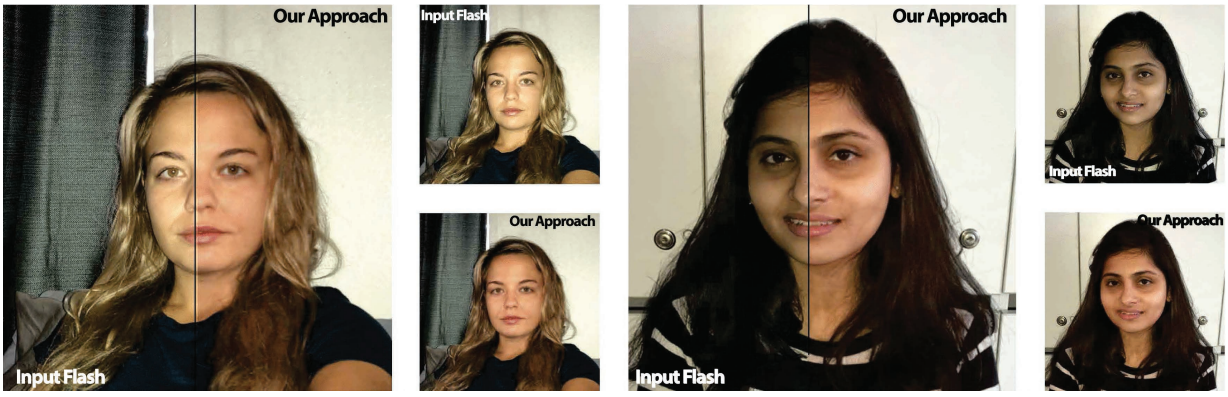


FIGURE 1. Two examples from our results. The split images show a comparison between the input and the output of our algorithm.

to wider application domains than just selfies, but generating the collection of images needed for this broader purpose would be an arduous undertaking, with hundreds or thousands of images required. After training our CNN with these image pairs, our model can be used to give a studio-lighting appearance to a broad range of real-world smartphone flash selfie images.

DEEP FLASH

We developed an encoder–decoder CNN based on two subnetworks: the first network performs the encoding of the input flash image to create a deep feature map representation; the second network recreates the image starting from the encoder's output while removing the flash defects. We use Visual Geometry Group's VGG-16[8] network to perform the image encoding that consists of 16 layers: the first 13 are convolutional layers and the last 3 are fully connected layers. We used only the convolutional layers of VGG-16, which are structured into five groups: the first two groups consist of two layers and the last three groups consist of three layers, as shown in Figure 2. There are three operations performed by each CNN layer: several parallel convolutions, nonlinear activation using Rectified Linear Unit (ReLU) functions, and max pooling operations.[9]

The decoding task was performed using a decoder component that is based on Eilertsen *et al.*'s U-Net-based approach.[5] The output produced from VGG-16 represents an input for the decoder after a further convolution operation. We use batch normalization after each convolution to normalize the output distribution of each layer in order to provide a valid input for the next layer. To this end, batch normalization constrains the activation function input to have unit variance and mean zero. After each batch normalization, the output tensor crosses through the next activation function, which introduces a nonzero gradient for the next inputs. Decoder layers consist of operations such as convolutions, batch normalization, deconvolutions, and concatenations.

To reduce the "vanishing gradients" problem that affects very deep neural networks, we employed a residual learning network-based approach (https://en.wikipedia.org/wiki/Residual_neural_network). The vanishing gradients problem concerns the backpropagation phase, where an inverse crossing of the network is performed to update the weights through the gradient of the error function. When a network is composed of many layers, the weight updating can be reduced so much from the last to the first layers that the updates in the first network layers become inefficient, thereby stopping the training. A way to solve this problem is to concatenate each block's output of the VGG-16 with its counterpart in the decoder using concatenation layers (see Figure 2). Another reason for our use of residual learning is that it recovers information lost through the convolutions of the encoding phase, helping the decoder in reconstructing the output image. Our encoder–decoder structured neural network is also able to recreate similar input images of faces taken in a different RGB lighting mode. Finally, we use deconvolutional layers to reconstruct the output image, starting from the VGG-16 output tensor.

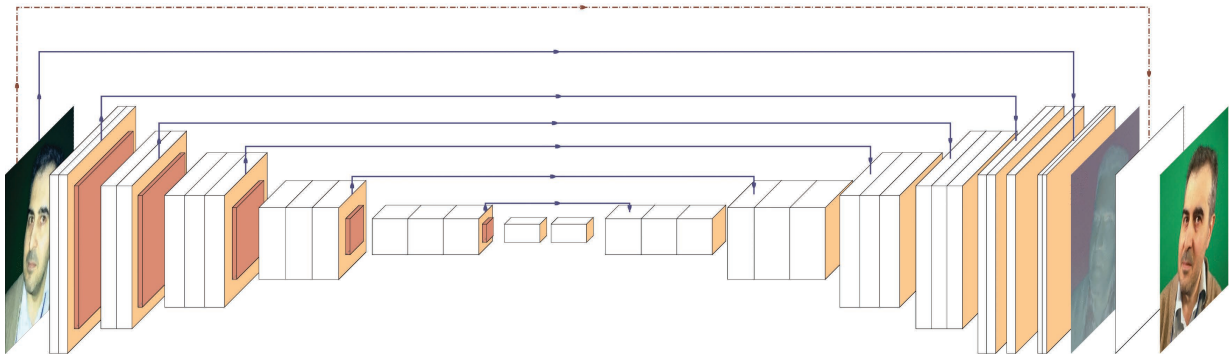


FIGURE 2. Architecture of our encoder–decoder, with its typical U-shape. The first 13 blocks represent the convolutional layers of VGG-16, which perform the image encoding. The second part reconstructs the output image and has several convolutional and deconvolutional layers. Arrows show the shortcut connections to the blue blocks of the decoder from their counterparts in the encoder.

Training

To minimize the loss function, we train our neural network using an algorithm called an Adam Optimizer (https://en.wikipedia.org/wiki/Stochastic_gradient_descent), which is a stochastic gradient descent technique with a special way of managing its learning rate. As the initial configuration of the Adam Optimizer, we set the learning rate to 10^{-5}, set the ϵ value (useful for avoiding divisions by zero) to 10^{-8}, and set the minibatch size to 4. The choice of minibatch size value is due to the GPU capability and the input image dimension. To compensate for the limited amount of available training data and to increase the generalization level, we use transfer learning (https://en.wikipedia.org/wiki/Transfer_learning), a technique that extends learning achieved in one domain to related problems.

In particular, the weights of the VGG-16 were initialized through a pretrained model originally used for face recognition.[10] This model was trained through a dataset of 2.6 million faces belonging to 2600 identities, using an NVIDIA Titan Black GPU. Our decoder weights were initialized using a truncated normal distribution, which ensures that the weight's initialization has unit standard deviation and mean zero. This approach avoids dissolution or increasing of the gradient, decreasing the probability of introducing critical errors during training. For the initialization of the last decoder layer, we use Xavier (http://proceedings.mlr.press/v9/glorot10a/glorot10a.pdf), which ensures that the signal passing through the neural network is propagated accurately and that the weights are neither too small nor too large.

Encoding

To implement our solution, we chose an encoding that decouples the high-frequency details such as hair or facial features from low-frequency details such as global skin tone. To this end, we employed the well-known bilateral filter (https://en.wikipedia.org/wiki/Bilateral_filter), which is a nonlinear filter that is frequently used to smooth images while preserving edges. Such filtering was used on both the flash and uniformly lit (ground truth) images of our dataset before training our neural network. Once the filter was applied, the flash-filtered image was used as the input to our neural network and the target was the difference between the filtered flash image and filtered ground truth image normalized to [0, 1]. The use of this type of encoding reduces artifacts such as blur due to the small misalignment of facial expressions between the flash and ground truth images, closed/open eye, lips position, and other facial landmarks.

Loss Function

The method described in the previous section allows us to preserve the low frequencies from the original nonfiltered image for use in subsequent steps. We minimize the distance between the low frequencies of the input and ground truth as follows:

FLASH PHOTOGRAPHY

Flash photography has been previously used to add details to photographs in low-light conditions, which typically suffer from high noise. In two concurrent works, Petschnigg *et al.*[1] and Eisemann and Durand[2] proposed transferring the ambient lighting from flash photographs with low ISO, which implies low noise, into nonflash photographs of the same subjects or scene, with reduced noise. Other works[3] have developed this idea further by removing over- or under-illumination at a given flash intensity, reflections, highlights, and attenuation over depth. Removing or reducing unwanted reflections in pictures can also be obtained by the approach proposed by Zhang *et al.*,[4] an end-to-end learning technique for single-image reflection separation with perceptual losses and a customized exclusion loss.

Eilertsen *et al.*[5] proposed an approach to obtain high dynamic range (HDR) images from low dynamic range images based on the U-Net architecture (https://en.wikipedia.org/wiki/U-Net) originally developed as a CNN for biomedical image segmentation. Similarly, Chen *et al.*[6] showed that U-Nets can be used successfully to de-Bayer images captured at low-light conditions and high ISO, which typically exhibit considerable noise. They extensively studied different approaches to processing such real-world noisy low-light images. For example, they tested a variety of architectures, loss functions [e.g., L1 (least absolute deviations), L2 (least square errors), and the structural similarity index (SSIM)], and different color inputs.

Aksoy *et al.*[7] presented a large-scale collection of pairs of images with ambient light and flash light of the same scene. These images were obtained by casual photographers using their smartphone cameras, and consequently, the dataset covers a wide variety of scenes. The dataset was provided for future work on high-level tasks such as semantic segmentation or depth estimation. Unlike their dataset, whose objective is to provide matching between two images under uncontrolled lighting conditions, our dataset aims to change the lighting scheme by converting images from flash lighting to a controlled photograph studio light one.

$$L(y_d, t_d) = \frac{1}{3N}\sum_i \Big((y_{di} - \mathbb{E}[y_{di}]) - (t_{di} - \mathbb{E}[t_{di}]) \Big)^2 \quad (1)$$

where

$$\begin{aligned} y_{di} &= \mathrm{BL}(x_i, \sigma_s, \sigma_r) - 2y_i + 1 \\ t_{di} &= \mathrm{BL}(x_i, \sigma_s, \sigma_r) - 2t_i + 1. \end{aligned} \quad (2)$$

Equation (1) is our objective (loss) function to be minimized, in which N represents the number of pixels, $\mathrm{BL}(x_i, \sigma_s, \sigma_r)$ is the CNN input, x_i is the flash image, y_i is the predicted difference of the CNN, and $t_i = \mathrm{BL}(x_i, \sigma_s, \sigma_r) - \mathrm{BL}(o_i, \sigma_s, \sigma_r)$, with o_i being the ground truth.

We normalized our target difference images into the range [0,..., 1] in order to avoid negative values affecting the CNN convergence given its activation functions. Then, values are remapped into the range [–1, 1] and subtracted from the input values. We performed the mean subtraction for each color channel of each image pixel by pixel, but only in the evaluation phase of the objective function. This operation was performed to distribute in a balanced manner the weights of each image across the training. In this way, each image gives the same contribution to the training, no more or less important than the others. In contrast to the classical method of subtracting from each image the mean computed across the whole training dataset, we subtracted the mean for each image to remove the average brightness from each pixel. This operation can be performed because our image domain consists of stationary data for which the lighting parameters are well defined and always the same for both the input and the output. For further details, see the article by Capece *et al.*[9]

DATASET CREATION

In order to produce a training set for our network, we acquired pairs of photographs of the same subject using the camera of a Google Nexus 6 smartphone at full resolution (i.e., 13 MP). We captured one photograph of the pair using only the flash of the smartphone, and the other using a studio-like set of lamps that provide uniform illumination. In postprocessing,

we aligned each pair using the MATLAB Image Processing Toolbox™ to minimize misalignments. These are caused by a delay of about 400 ms between the two shots due to switching on and off the studio lamps. We set the nonflash image as the misaligned one and the flash image as our reference, and then ran a tool for affine alignment, i.e., translation, rotation, scale, and shear. Since photographs in each pair have different lighting conditions, we had to use a multimodal optimizer to align two images using intensity-based registration. In a few cases, misalignments persist when one image has the subject with open eyes and the other with closed eyes or vice versa. In such cases, the worst images were removed from the dataset.

After the alignment step, we identified the subject face by running a simple face recognition API (https://github.com/ageitgey/face_recognition). This outputs a bounding box for a photograph that we used to crop each image, which is finally downsampled to 512 × 512. During our acquisition process, we managed to collect 495 pairs of photographs. These pairs represent 101 people (both females and males) in different poses. In order to have a larger dataset, we augmented this set using three common techniques:

- five rotations from −20° to +20° around the center of the face bounding box, using a 10° step;
- cropping the image to the face bounding box and rescaling to original image size;
- flipping images horizontally.

These operations increased the original dataset by a factor of 20, obtaining a training set of 9.900 images at a 3120 × 4160 resolution (13 MP).

RESULTS AND DISCUSSION

We trained our CNN on pairs of 512 × 512 images for about five days on an NVIDIA Titan Xp GPU, performing 62 epochs and about 458 000 backpropagation iterations. We interrupted the training when the value of the loss function computed on 1500 images reached a low level of approximately 0.0042.

We calculated the accuracy of the result as

$$\mathrm{acc} = 100 - \left(\frac{100}{3w(I)h(I)} \sum_i \sum_c | I_c - \tilde{I}_c | \right) \quad (3)$$

FIGURE 3. Results of our approach on real selfie images of the dataset provided by Aksoy *et al.*[7] Such a dataset, as well as the training dataset, consists of images taken to approximate real selfie images using a smartphone and smartphone flash at a similar distance and angle of a real selfie. The first column represents the input of our neural network, the middle column represents our result, and the last column represents the no-flash ambient image.

where $I = t_d, d = y_d, w(I) = \mathrm{width}(I)$, and $h(I) = \mathrm{height}(I)$. After the training step, we obtained an accuracy value of 96.2%. In the test phase, we evaluated our approach using 740 test images, obtaining a loss-test value of 0.0045 and an accuracy value of 96.5%. The evaluation was done on the dataset provided by Aksoy *et al.*,[7] on images such as those shown in Figures 1, 3, and 4.

Comparison With Reconstructed Ground Truth Images

One key idea of our technique is to train the CNN to learn the difference between the bilateral filtered target and input images. The output of the pipeline then subtracts the CNN prediction from the original input image. This allows us to preserve the high-frequency

FIGURE 4. Our approach can be used on people with different features and ethnicities. Although the flash highlights remain evident in the lenses of people with glasses, they do not affect our approach to the rest of the image.

detail, even though the exact ground truth cannot be reconstructed exactly, even with a zero-loss function.

But of greater concern are the misalignments due to pose changes between the flash and nonflash photos, which would otherwise dominate when computing the input and output image differences. For these reasons, we introduce a preconditioning operator on the ground truth:

$$\overline{o}_i = x_i - 2t_i + 1 \tag{4}$$

where t_i is the target difference. This operator represents the reconstructed ground truth in which some of the high frequencies lost through the bilateral filter and not recoverable were not considered.

We show an excerpt of the validation data in Figure 5. Note how hair, beard, and skin color are lost in the flash photo and restored in our results. Also, shadows and highlights due to the flash are mitigated.

We evaluated the data by employing the SSIM; see Figure 6 for comparisons between the ground truth images and reconstructed predictions.

Comparisons

We compared our results against three other approaches in the literature (see Figure 6). The first approach is HDRNet by Gharbi *et al.*[11] and is based on the use of a CNN combined with bilateral grid processing and local affine color transforms. HDRNet is designed to learn the effect of any image operator and, hence, is a suitable candidate to remove flash artifacts from photographs. The second approach is Pix2Pix by Isola *et al.*,[12] which is based on a "conditional" Generative Adversarial Network (cGAN) for which image generation is conditional on the type of image. This type of neural network was investigated as a general-purpose solution to image-to-image translation problems. Isola *et al.* tested their cGAN on different tasks such as photo generation and semantic segmentation. The third approach is the style transfer method proposed by Shih *et al.*[13] in which a multiscale local transfer approach is applied to portraits.

We feel our technique compares favorably with these other methods in that it makes the lighting uniform removing the flash highlights without introducing problems such as altering geometries and blur effect.

CONCLUSION

In photography, glare is a common issue that causes shiny highlights, especially in portraits. In the majority of cases, glare is a defect, and the subjects seem to be greasy. Glare can be removed from the face manually with a complicated and uncertain result process that requires photo-editing skills.

This article proposes a technique that is able to dramatically increase the quality of smartphone flash selfies by turning them into portraits with studio-like lighting. The approach is able to automatically remove flash lighting artifacts such as hard shadows and highlights by using a regression model based on supervised learning.

These results confirm the capabilities that learning-powered computational photography is able to reach in lighting control and suggest promising new developments in other contexts such as relighting for

FIGURE 5. Validation dataset samples where the SSIM was computed through central images and top-right images. Such a dataset, as well as the training dataset, consists of images taken to approximate real selfie images using a smartphone and smartphone flash at a similar distance and angle of a real selfie. Top-left images are the flash images taken with the smartphone; bottom-left images are the filtered images with bilateral filter; centered images are the results of our approach; top-right images are reconstructed ground truth images, and finally, the bottom-right images are the original ground truth images.

better presentation of objects or for advanced shading removal in photogrammetric reconstructions.

REFERENCES

1. G. Petschnigg, R. Szeliski, M. Agrawala, M. Cohen, H. Hoppe, and K. Toyama, "Digital photography with flash and no-flash image pairs," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 664–672, Aug. 2004.
2. E. Eisemann and F. Durand, "Flash photography enhancement via intrinsic relighting," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 673–678, Aug. 2004.
3. A. Agrawal, R. Raskar, S. K. Nayar, and Y. Li, "Removing photography artifacts using gradient projection and flash-exposure sampling," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 828–835, Jul. 2005.
4. X. Zhang, R. Ng, and Q. Chen, "Single image reflection separation with perceptual losses," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, Jun. 2018, pp. 4786–4794.
5. E. Gabriel, K. Joel, D. Gyorgy, M. Rafał, and U. Jonas, "HDR image reconstruction from a single exposure using deep CNNs," *ACM Trans. Graph.*, vol. 36, no. 6, 2017, Art. no. 178.

FIGURE 6. Example of comparisons among our approach and HDRNet, Pix2Pix, and Style Transfer. Note that our approach keeps the geometric features of the input images; this makes the lighting uniform.

6. C. Chen, Q. Chen, J. Xu, and V. Koltun, "Learning to see in the dark," in *Proc. IEEE/CVF Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 3291–3300.

7. Y. Aksoy, et al., "A dataset of flash and ambient illumination pairs from the crowd," in *Proc. Eur. Conf. Comput. Vision*, 2018, pp. 644–660.
8. K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations*, May 7–9, 2015.
9. N. Capece, F. Banterle, P. Cignoni, F. Ganovelli, R. Scopigno, and U. Erra, "Deepflash: Turning a flash selfie into a studio portrait," *Signal Process.: Image Commun.*, vol. 77, pp. 28–39, 2019.
10. O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proc. Brit. Mach. Vision Conf.*, Jan. 2015, vol. 1, pp. 41.1–41.12.
11. M. Gharbi, J. Chen, J. T. Barron, S. W. Hasinoff, and F. Durand, "Deep bilateral learning for real-time image enhancement," *ACM Trans. Graph.*, vol. 36, no. 4, 2017, Art. no. 118.
12. P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *Conf. Comput. Vision Pattern Recognit.*, 2017.
13. Y. Shih, S. Paris, C. Barnes, W. T. Freeman, and F. Durand, "Style transfer for headshot portraits," *ACM Trans. Graph.*, vol. 33, no. 4, 2014, Art. no. 148.

NICOLA CAPECE is currently a Research Fellow with the Computer Graphics Lab, Università della Basilicata, Dipartimento di Matematica, Informatica ed Economia, Potenza, Italy. Contact him at nicola.capece@unibas.it.

FRANCESCO BANTERLE is currently a Researcher with the Consiglio Nazionale delle Ricerche Istituto di Scienza e Tecnologie dell'Informazione (ISTI-CNR), Pisa, Italy. Contact him at francesco.banterle@isti.cnr.it.

PAOLO CIGNONI is currently the Research Director and Head of the Visual Computing Lab, Consiglio Nazionale delle Ricerche Istituto di Scienza e Tecnologie dell'Informazione (ISTI-CNR). Contact him at paolo.cignoni@isti.cnr.it.

FABIO GANOVELLI is currently a Full Time Researcher with Consiglio Nazionale delle Ricerche Istituto di Scienza e Tecnologie dell'Informazione (ISTI-CNR). Contact him at fabio.ganovelli@isti.cnr.it.

UGO ERRA is currently an Assistant Professor with the Università della Basilicata, Dipartimento di Matematica, Informatica ed Economia, Potenza, Italy, where he is the founder and the Head of the Computer Graphics Lab. Contact him at ugo.erra@unibas.it.

Contact department editor Mike Potel at potel@wildcrest.com.

EDITOR: E. Cambria, Nanyang Technological University

DEPARTMENT: AFFECTIVE COMPUTING AND SENTIMENT ANALYSIS

This article originally appeared in IEEE Intelligent Systems vol. 35, no. 1, 2020

Intent Classification for Dialogue Utterances

Jetze Schuurmans and Flavius Frasincar, *Erasmus University Rotterdam*

In this work, we investigate several machine learning methods to tackle the problem of intent classification for dialogue utterances. We start with bag-of-words *in combination with* Naïve Bayes. *After that, we employ* continuous bag-of-words *coupled with* support vector machines *(SVM). Then, we follow* long short-term memory *(LSTM) networks, which are made bidirectional. The best performing model is hierarchical, such that it can take advantage of the natural taxonomy within classes. The main experiments are a comparison between these methods on an open sourced academic dataset. In the first experiment, we consider the full dataset. We also consider the given subsets of data separately, in order to compare our results with state-of-the-art vendor solutions. In general we find that the SVM models outperform the LSTM models. The former models achieve the highest macro-F1 for the full dataset, and in most of the individual datasets. We also found out that the incorporation of the hierarchical structure in the intents improves the performance.*

Customer interaction is at the center of many organizations. In order to help customers efficiently, one could automate the interaction between the organization's representative and a customer. Customers usually contact the organization with a specific request or query. In order to help a customer, the intention of the customer needs to be classified.[8] Intent classification tries to answer the question why the customer contacted the organization and what the customer wants to achieve. The interaction can partly or fully be automated using a dialogue system,[27] which uses intent classification. The classification can also be used to help the human representatives, namely, by using intent classification to direct the incoming messages to the representative that has the right expertise. Due to its importance for dialogue handling,[25] intent classification needs to be done properly. Therefore, this research focuses on improving the existing practice of intent classification for dialogue utterances.

In order to classify intents of customers, a dialogue system needs to analyze the incoming messages. The messages are called utterances, or acts-of-speech. In our case they are typed messages in English, roughly the length of a sentence. The classification of the intent is made per utterance. We analyze the case where possible intents are disjoint. In other words, each incoming message belongs to only one class. However, some intents might be very similar and belong to a common category, or in other words to a group of intents. We explore the possibility of extending the classifier with knowledge about the inherent hierarchy of intents.

RELATED WORK AND SCIENTIFIC RELEVANCE

Previous studies have proposed several classification algorithms for short texts, starting with parsimonious text classifiers, such as bag-of-word (BoW) with Naïve Bayes (NB) and continuous bag-of-word (CBoW) with support vector machines (SVM).[24] The performance of NB is limited by the vocabulary in the training set. SVM

Digital Object Identifier 10.1109/MIS.2019.2954966
Date of publication 22 November 2019; date of current version 18 March 2020.

can circumvent this by using word embeddings, trained on an external corpus. However, with both approaches, word order is lost. To account for complex dependencies between words in the representation of an utterance *recurrent neural networks* (RNNs) were introduced.[6] Most recently, LSTMs and their simplification gated recurrent unit (GRU) have been used for intent classification[17] and emotion detection,[16] respectively, in dialogues. Attentive LSTMs[10] are less useful here as the classified text is rather short in nature.

Flat classifiers need to distinguish between all classes at once. When there is a large number of classes, this can become difficult. Instead, *hierarchical classification* can be used. A hierarchical classifier tries to incorporate the hierarchical structure of the class taxonomy. Hierarchical classification was first used for text classification by Koller and Sahami.[15] They used a local classifier per parent node for training, at each node selecting a subset of features relevant for that step in the classification process. A similar hierarchical structure with an SVM at every node was used for speech-act classification.[13] Ono *et al.* used a form of local classifier per level, where they tried the lowest level (leaf nodes) first.[19] If the uncertainty is too high, they move up in the hierarchical level. Hierarchical classifiers have been used for intent classification in Web[12] and platform[11] searches. For chatbots, multi-intent classification was researched by Rychalska *et al.*[21]

We contribute to the existing literature in two ways. First, we apply hierarchical intent classification on dialogue utterances (in multiclass classification as apposed to multilabel). Second, we present performances of machine learning classifiers, alongside the black box models used by Braun *et al.*[2]

METHODOLOGY

In this section, we discuss the methods used to classify intents. Each method is a combination of an utterance representation and a classification algorithm. We start with a formalization of the problem. Then follow the flat classifiers. Finally, we discuss the hierarchical classifier.

Intent Classification

The classification of an intent is answering the question: What is the customer trying to accomplish? In intent classification, the utterance $d \in X$ *of* a dialogue is given, where X is the utterance space; a fixed set of predefined intents $C = \{c_1,..., c_J\}$ and a training set D of labeled dialogue utterances $\{d_i, c_i\}_{i=1}^{N}$ where $(d,c) \in X \times C$. We consider the one-off problem or in other words single label classification, where each d corresponds to one element of C. For example,

(d,c) = ("What software can I use to view epub documents?", "Software Recommendation").

Flat Classifiers

BOW-NB: The first model we discuss is the BoW representation with multinomial NB. This model is the baseline in our experiments. Each utterance is represented by the set of word counts that occur in the utterance. Therefore, word order is neglected. The way we implement NB is as follows. First, we start by removing the stop words. Second, we use lemmatization. Although the combination of unigram and bi-gram is advised,[24] we do not have enough bi-gram counts. Therefore, we only use uni-grams. We handle zero counts with Laplace smoothing.

An advantage of NB is its efficiency during training time, as it only needs to pass through the data once. However, the downside of NB is the conditional independence assumption, stating that terms and the signal they carry are independent of each other given in the class. Furthermore, the model uses the positional independence assumption, stating the position of a word does not matter. Most importantly NB cannot handle unseen words.

CBOW-SVM: Second, we discuss CBoW as an input for SVM. CBoW uses continuous word representations called word embeddings. This gives the SVM classifier the advantage to pick up signals from similar in meaning, yet unseen, words. We use three word embeddings: Word2Vec,[18] GloVe,[20] and FastText.[1]

The CBoW representation is comparable to the conventional BoWs representation, since both lose the information of the order of terms. However, using word embeddings gives CBoW an advantage over traditional BoWs. Namely, CBoW can pick up signals from previously unseen words. CBoW gives us the additional advantage that the input for the classification algorithm is a fixed dimensional vector, independent of the length of the utterance or vocabulary. This is

a desirable feature for SVMs. There are two forms of CBoW we consider. One takes the sum of the embedding vectors of the respective terms, whereas the other takes the average

$$CBoW_{\text{sum}}(t_1, \ldots, t_k) = \sum_{i=1}^{k} v(t_i)$$
$$CBoW_{\text{ave}}(t_1, \ldots, t_k) = \frac{1}{k}\sum_{i=1}^{k} v(t_i) \quad (1)$$

where each feature t_i corresponds to a word and has an associated vector $v(t_i)$.

The intuition behind CBoW is as follows. The summation of word vectors creates a path in the word embedding space. The resulting vector (from the origin to the end of the path) should capture a mathematical representation of the overall meaning of the utterance. Adding more words with the same meaning might spread the cluster of the representations of a given intent, possibly making the classification harder. When the average is taken, the overall length of this path is normalized with respect to the number of words in the utterance.

SVMs are a classification method that uses a kernel function to find decision boundary between two classes that has a maximum margin in a latent space. We consider both the *Linear* and *Radial Basis Function* kernels. Since we allow for misclassifications in the training set, a cost parameter *C* is added to give a penalty to these violations. In order to determine which kernel and hyperparameters to use, we use twofold cross validation with stratified sampling.

Inherently, SVMs are binary classifiers. Several attempts have been made to create a multiclass SVM scheme.[9] We use the one-against-one[14] scheme, as it performed as one of the best in the comparison of Hsu and Lin.[9] During testing we use Max Wins voting,[4] where the class with the highest number of votes is chosen as final prediction. Since we are dealing with unbalanced class distributions, we use class weights in the SVM.

LSTM: The key feature of RNNs is that they can process sequential data, giving them the possibility to model word dependencies. Parameter sharing enables the recurrent network to pick up signals from longer sequences than dense neural networks, and to take inputs of arbitrary length and learn general patterns across them. There are several types of RNN architectures,[22] we consider the *tail* model. The tail model constructs a hidden state by passing the complete sequence and using the last hidden state as input for the classification layer. Alternatives such as the *pooling* or *hybrid pooling* do not consistently outperform the more parsimonious tail model.[22]

Gated RNNs are the most compelling sequence models used in practice. These include networks based on the LSTM[7] and GRU.[3] Gated RNNs are based on the idea of creating paths through time that have derivatives that neither vanish nor explode. This is done by learning connection weights, and the ability to forget the old state, from the data. We choose to use LSTMs over GRUs due to the extra flexibility offered by the controls for the update and output of the state.

Bidirectional LSTM (BiLSTM) was created to model dependencies on the next time step in the sequence.[5] They are a combination of a recurrent module that passes the sequence forward through a memory block and a recurrent module that passes the sequence backwards through a different memory block. The tail model uses a concatenation of the final two hidden states as input for the last layer.

Following similar work, the network is trained using the Adam optimizer.[17] We calculate updates from the gradients based on batches of training utterances. We use backpropagation through time[26] to update recurrent components. Gradient clipping is used in order to deal with exploding gradients and we found that capping the gradients at five works well. We use the following regularizers: early stopping, ensembles, and weight noise. A popular way of creating weight noise is by applying dropout. We use dropout only at the nonrecurrent connections.[28] The hyperparameters of the LSTM model are the size of the input dimension, and the size of the state variable. Both are determined by twofold cross validation using stratified sampling.

Hierarchical Classifiers

Hierarchical classification can be considered as a classification that takes the hierarchical structure of the taxonomy of classes into account, as opposed to a flat classifier, which only takes the final classes into account. By imposing the hierarchical structure, the model does not need to learn the separation between a large number of classes. It can now focus

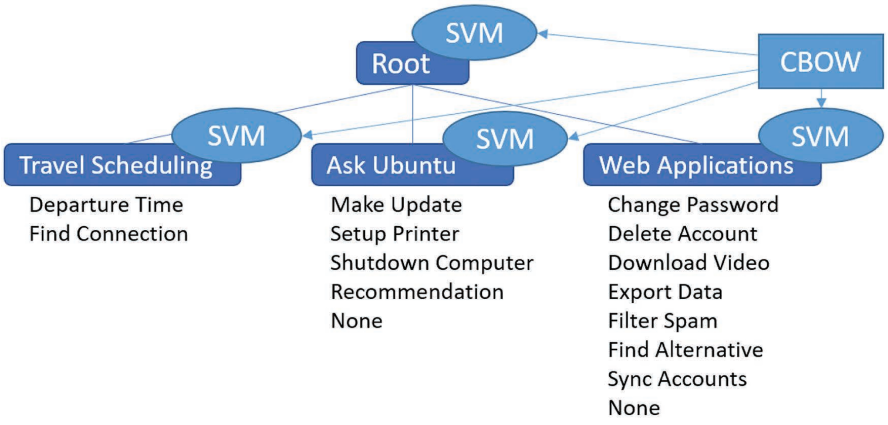


FIGURE 1. Hierarchy of the classes with a local hierarchical classifier per parent node.

on classifying subclasses within a category. The taxonomy can be formalized as a *tree* or a *directed acyclical graph*,[23] we consider the case where the taxonomy is a tree due to the nature of our data.

Our goal is to reduce the number of classes considered based on the natural taxonomy, therefore we use a local classifier per parent node. This local hierarchical classifier has a flat classifier at every parent node, which means that the number of classifiers that need to be constructed scales directly with the number of parent nodes. During training of a classifier at any given parent node, only the observations belonging to its children are considered. After training each individual classifier, the local classifier can be used for inference. During testing, the classification starts at the root node. The outcome of the root node determines which next classifier should be considered. The outcome of this classifier selects the next classifier to be used. This is repeated until a leaf node is predicted, this then becomes the final prediction of the local classifier.

Performance Measure

We measure the performance with the macro-F1 score. The F1 score is a harmonic mean of the precision and recall for each intent. We value both and do not want a linear tradeoff between them. We are interested in the performance on all classes equally, independent of the number of test observations. Therefore, we aggregate the measures by means of the macro average.

DATASET AND EXPERIMENTS

We use the dataset curated by Braun *et al.*,[2] available at https://github.com/sebischair. It consists of two corpora, distinguished by the way they were gathered. There is the Chatbot Corpus on Travel Scheduling, and the StackExchange Corpus on Ask Ubuntu and Web Applications. In this section, we discuss the experimental setups on this dataset.

Complete Dataset: We start with the complete set that includes all three subsets. This gives us the opportunity to select the best overall model, based on the macro-F1 score. The concatenation of the three subsets imposes a hierarchy in the taxonomy of intents. This allows us to compare hierarchical classifiers with flat classifiers. The class hierarchy is depicted in Figure 1.

Individual Datasets: In this experiment, we consider the subsets of the data separately. This gives us the possibility to compare our methods with the classifiers used by Braun *et al.*[2] They use the Natural Language Understanding solutions of LUIS, Watson Conversation, API.ai, and RASA.

RESULTS

Complete Dataset: The results of the different classifiers on the complete dataset are reported in Table 1. The best performing flat classifier is the SVM model, this is independent of the type of word embedding or the method used to aggregate the word embeddings. We select this classifier as candidate for the hierarchical classifier. When adding hierarchy to the models, we find varying results. The baseline model clearly improves when taking the taxonomy of classes into account, while adding the local hierarchy to the SVMs

comes with mixed results. For the FastText embeddings it is a clear improvement, whereas for the GloVe embeddings it is not. Overall, the best hierarchical SVM outperforms the best flat SVM.

With regard to the utterance representation, we find that averaging is better than summing the word embeddings, as SVM with $CBoW_{ave}$ performs better in the flat classification and the best hierarchical classifier uses also averages. Furthermore, we note that the bidirectional component in the BiLSTMs does not capture more information, as the LSTM performs better than the BiLSTM. Together with the fact that the SVM outperforms the LSTM, this indicates that taking the word order into account is not relevant in this dataset. This is likely due to the short utterance length.

Individual Datasets: The macro-F1 for the individual datasets are in Table 2. We note that it is hard to interpret the comparison with Braun *et al.*,[2] as most of the methods used are black boxes.

In the Travel Scheduling dataset, the (Bi)LSTM with Word2Vec embeddings performs the best. The SVM with Word2Vec and $CBoW_{ave}$ and BiLSTM perform equally well as the intent classifiers of LUIS and RASA. We note that the relatively high performance of our baseline, NB, indicates that this is a relatively easy set to classify.

The Ask Ubuntu set provides a slightly harder classification task. In this set, the intent classifier of Watson outperforms the other vendor solutions as well as all our models. From our models, the SVM with FastText with $CBoW_{ave}$ is the best performing model. We note that all RNNs are performing worse than the NB baseline.

The final subset is on Web Applications. The Web Applications data proves to be more difficult, this is likely due to the fact that it has very few training observations (an average of less than four training observations per intent). Here, we see that our best performing model is the SVM with FastText and $CBoW_{ave}$. Together with the Word2Vec $CBoW_{ave}$ and the GloVe $CBoW_{sum}$ it outperforms the vendor solutions. Furthermore, we note that the BiLSTM is the best performing recurrent network, just as in the Travel Scheduling and Ask Ubuntu sets. One can note that, on the complete dataset the LSTM performed better than BiLSTM, as the LSTM has an edge in differentiating between the three types of datasets.

TABLE 1. Macro-F1 for the test set on the complete dataset.

Model	Flat	Hierarchical	Model	Flat
NB	.541	.614		
SVM FastText average	.689	**.782**	LSTM FastText	.605
SVM FastText sum	.657	.642	BiLSTM FastText	.569
SVM GloVe average	**.752**	.654	LSTM GloVe	.586
SVM GloVe sum	.680	.658	BiLSTM GloVe	.575
SVM Word2Vec average	.705	.703	LSTM Word2Vec	.543
SVM Word2Vec sum	.673	.706	BiLSTM Word2Vec	.502

TABLE 2. Macro-F1 score for the individual subsets.

	Travel Scheduling	Ask Ubuntu	Web Applications
NB	.959	.726	.502
SVM FastText average	.958	**.812**	**.771**
SVM FastText sum	.968	.800	.658
SVM GloVe average	.946	.805	.591
SVM GloVe sum	.957	.729	.692
SVM Word2Vec average	.979	.742	.698
SVM Word2Vec sum	.946	.742	.680
LSTM FastText	.968	.644	.465
BiLSTM FastText	.979	.646	.549
LSTM GloVe	.945	.665	.546
BiLSTM GloVe	.979	.667	.635
LSTM Word2Vec	**.989**	.631	.395
BiLSTM Word2Vec	**.989**	.710	.443
LUIS	**.979**	.743	**.690**
Watson	.968	**.819**	.630
API.ai	.931	.782	.628
RASA	**.979**	.708	.494

CONCLUSION

In general, we find that the SVM models outperform the LSTM models. They achieve the highest macro-F1 for the full dataset, they are also able to handle the scenario of the individual datasets. With regard to taking advantage of the hierarchical structure in the

intents, we find that the SVM with averaged FastText embeddings significantly benefits from the hierarchy and outperforms all other models. Using word embeddings as utterance representation yields a better performance than using a count-based method, however, taking word order into account does not. In general, we see better results when we take the element wise average of the word embeddings, as apposed to the sum, indicating that correcting for the length of the utterance is useful. Finally, we note that our models improve on the NB baseline. Furthermore, they are on par with or improve on the performance of the black box methods used by Braun *et al.*[2]

Future Research

There are different opportunities for future work, we discuss a few below. We start with several options with respect to the hierarchy, followed by data augmentation and transfer learning.

The type of hierarchical model considered is a local hierarchical classifier per parent node. Alternatively, a global hierarchical classifier could be constructed by modifying a flat classifier to take the taxonomy into account at once. The intermediate certainties could be exploited by the dialogue system, with specific follow-up questions.

In order to deal with the limited number of training observations, future work could look into data augmentation or transfer learning. Data augmentation could be used by interchanging one or multiple random words with their synonyms. Alternatively, transfer learning can be used. One could take a subset of intents, starting with two intents, training the classifier and using the inferred weights as initialization when learning to classify with an additional intent added to the problem.

REFERENCES

1. P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Trans. Assoc. Comput. Linguistics*, vol. 5, pp. 135–146, 2017.
2. D. Braun, A. Hernandez-Mendez, F. Matthes, and M. Langen, "Evaluating natural language understanding services for conversational question answering systems," in *Proc. 18th Ann. SIGdial Meeting Discourse Dialogue*, 2017, pp. 174–185.
3. K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder–decoder approaches," in *Proc. SSST-8, 8th Workshop Syntax Semantics Structure Statistical Transl.*, 2014, pp. 103–111.
4. J. H. Friedman, "Another approach to polychotomous classification," Statistics Dept., Stanford Univ., Stanford, CA, USA, Tech. Rep., 1996. [Online]. Available: http://statweb.stanford.edu/~jhf/ftp/poly.pdf
5. A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," *Neural Netw.*, vol. 18, no. 5/6, pp. 602–610, 2005.
6. M. Henderson, B. Thomson, and S. Young, "Word-based dialog state tracking with recurrent neural networks," in *Proc. 15th Ann. Meeting Special Interest Group Discourse Dialogue*, 2014, pp. 292–299.
7. S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
8. N. Howard and E. Cambria, "Intention awareness: Improving upon situation awareness in human-centric environments," *Human-Centric Comput. Inf. Sci.*, vol. 3, no. 1, 2013, Art. no. 9.
9. C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 415–425, Mar. 2002.
10. D. Hu, "An introductory survey on attention mechanisms in NLP problems," in *Proc. Intelligent Systems Conference* (ser. Advances in Intelligent Systems and Computing), vol. 1038. Berlin, Germany: Springer, 2019, pp. 432–448.
11. J. Hu, G. Wang, F. Lochovsky, J.-T. Sun, and Z. Chen, "Understanding user's query intent with wikipedia," in *Proc. 18th Int. Conf. World Wide Web*, 2009, pp. 471–480.
12. B. J. Jansen, D. L. Booth, and A. Spink, "Determining the informational, navigational, and transactional intent of web queries," *Inf. Process. Manage.*, vol. 44, no. 3, pp. 1251–1266, 2008.
13. S. Kang, Y. Ko, and J. Seo, "Hierarchical speech-act classification for discourse analysis," *Pattern Recognit. Lett.*, vol. 34, no. 10, pp. 1119–1124, 2013.
14. S. Knerr, L. Personnaz, and G. Dreyfus, "Single-layer learning revisited: A stepwise procedure for building and training a neural network," in *Neurocomputing*. Berlin, Germany: Springer, 1990, pp. 41–50.
15. D. Koller and M. Sahami, "Hierarchically classifying

documents using very few words," Stanford InfoLab, Tech. Rep., 1997. [Online]. Available: http://ilpubs.stanford.edu:8090/291/

16. N. Majumder, S. Poria, D. Hazarika, R. Mihalcea, A. Gelbukh, and E. Cambria, "DialogueRNN: An attentive RNN for emotion detection in conversations," in *Proc. 33rd AAAI Conf. Artif. Intell.*, 2019, vol. 33, pp. 6818–6825.
17. L. Meng and M. Huang, "Dialogue intent classification with long short-term memory networks," in *Proc. Nat. CCF Conf. Natural Lang. Process. Chin. Comput.*, 2017, pp. 42–50.
18. T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. 27th Ann. Conf. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.
19. K. Ono, R. Takeda, E. Nichols, M. Nakano, and K. Komatani, "Toward lexical acquisition during dialogues through implicit confirmation for closed-domain chatbots," in *Proc. Workshop Chatbots Conversational Agent Technologies*, 2016, pp. 272–279.
20. J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2014, pp. 1532–1543.
21. B. Rychalska, H. Glabska, and A. Wroblewska, "Multi-intent hierarchical natural language understanding for chatbots," in *Proc. 5th Int. Conf. Soc. Netw. Anal., Manage. Secur.*, 2018, pp. 256–259.
22. L. Shen and J. Zhang, "Empirical evaluation of RNN architectures on sentence classification task," 2016. [Online]. Available: https://arxiv.org/abs/1609.09171
23. C. N. Silla and A. A. Freitas, "A survey of hierarchical classification across different application domains," *Data Mining Knowl. Discovery*, vol. 22, no. 1/2, pp. 31–72, 2011.
24. S. Wang and C. D. Manning, "Baselines and bigrams: Simple, good sentiment and topic classification," in *Proc. 50th Ann. Meeting Assoc. Comput. Linguistics*, 2012, pp. 90–94.
25. C. Welch, V. Perez-Rosas, J. Kummerfeld, and R. Mihalcea, "Learning from personal longitudinal dialog data," *IEEE Intell. Syst.*, vol. 34, no. 4, pp. 16–23, Jul./Aug. 2019.
26. P. J. Werbos, "Backpropagation through time: What it does and how to do it," *Proc. IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.
27. T. Young, E. Cambria, I. Chaturvedi, H. Zhou, S. Biswas, and M. Huang, "Augmenting end-to-end dialogue systems with commonsense knowledge," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 4970–4977.
28. W. Zaremba, I. Sutskever, and O. Vinyals, "Recurrent neural network regularization," 2014, [Online]. Available: https://arxiv.org/abs/1409.2329

JETZE SCHUURMANS received the Master of Science degree from Erasmus School of Economics, Erasmus University Rotterdam, Rotterdam, The Netherlands. After his studies in Econometrics, he worked for Captain AI, enabling autonomous shipping with artificial intelligence. Contact him at jetzeschuurmans@gmail.com.

FLAVIUS FRASINCAR is currently an Assistant Professor with Erasmus University Rotterdam, Rotterdam, The Netherlands. He received the Ph.D. degree in information systems from the Eindhoven University of Technology, Eindhoven, The Netherlands. His research interests include Web information systems, personalization, machine learning, and the Semantic Web. Contact him at frasincar@ese.eur.nl.

EDITOR: Jinan Fiaidhi, jinan.fiaidhi@lakeheadu.ca

This article originally appeared in
IT Professional
vol. 21, no. 5, 2019

DEPARTMENT: EXTREME AUTOMATION

White Learning: A White-Box Data Fusion Machine Learning Framework for Extreme and Fast Automated Cancer Diagnosis

Tengyue Li and Simon Fong, *University of Macau*
Lian-Sheng Liu, *Hospital of Guangzhou University of TCM*
Xin-She Yang, *Middlesex University*
Xingshi He, *Xi'an Polytechnic University*
Jinan Fiaidhi and Sabah Mohammed, *Lakehead University*

In machine learning, deep learning operates as a black-box, where the representation of prior knowledge is difficult to understand by human users and learning systems designers, although they provide high prediction accuracy. On the other hand, traditional learning techniques such as Bayesian networks usually learn from separate datasets with the ease of representing prior knowledge and causality information, which makes them remarkably good white-box inference making technology but with a moderate level of prediction accuracy. In this article, we are presenting a framework that combines the two worlds. It uses a fusing black-box learning that carries the advantage of high level of accuracy with the white-box representation capabilities of the traditional techniques with the prior knowledge, and causality information are readily available as a direct acyclic graph. We can call this combination *White Learning*. We define white learning as the systemic fusion of machine learning models based on an interpretable Bayes network that explains the relations among the attributes and class, as well as a deep learning that has the superior classification success rate. A case of loosely coupled white learning model, which uses an incremental version of Naïve Bayes network and deep learning, is tested on breast cancer diagnosis. Our experimental results show that it is possible to create a loosely coupled white-learning model that can do both accurate prediction and data relation reasoning.

Despite the recent momentum of AI-enabled cancer detection that has attracted lots of research output from academia, commercial bodies such as Google, for example, have developed a deep learning tool in October 2018 that is able to detect metastasized breast cancer with 99% accuracy. One key component of the cancer diagnostic information system is the core machine learning algorithm. According to Google: "*While Google LYNA (LYmph Node Assistant) achieved significantly higher cancer detection rates than had been previously reported, an accurate algorithm alone is insufficient to improve pathologists' workflow or improve outcomes for breast cancer patients. For patient safety, these algorithms must be tested in a variety of settings to understand their strengths and weaknesses. Furthermore, the actual benefits to pathologists using these algorithms had not been previously explored and must be assessed to determine whether or not an algorithm actually improves efficiency or diagnostic accuracy.*" That implies a deep learning algorithm, although fast may not be the only criteria in learning. There are more qualities that need to be explored before deep learning or machine learning technology in general can put into live applications that concern the life and death of patients.

Digital Object Identifier 10.1109/MITP.2019.2931415
Date of current version 11 September 2019.

 Published by the IEEE Computer Society

The deep learning model is often uninterpretable. Most people are using it without knowing the internal working mechanism on how and why the results were generated.

White Learning Framework

A framework of white learning is proposed in this article, which embraces three categories of white learning models where various levels of hybridization of Bayesian networks and neural networks are fused. At the algorithm level, Bayesian networks and neural computing are integrated tightly as a whole or partial redesigned entity of computing logics. Elements of Bayesian networks and neural computing co-exist in the design of program codes. This level of integration often requires a high extent of intellectual innovation, especially if the new hybrid after coupling the white-and-black box learning models would outperform either one of the original two models. On the other hand, loosely coupled models are those which run almost independently of each other; exemplars are those ensembles from which the results of the best performing model out of many are taken as the final results. These models are often taken as they are, without any modification in their codes. Their executions are totally self-contained. In some cases, the training/testing data may be modified throughout the process, improving its quality as output from one of the two models to the input of the other. A compromise between tightly and loosely coupled models is called a semicoupled model in our framework. This group of fusion model has the following features—the designs of the Bayesian network (BN) and neural network models that remain basically unchanged. However, the parameters, input variables, configurations, and/or execution controls of either one of the two models are influenced by the other model. During the machine learning operation, in addition to improved data, there is information passing between the two models with an objective of enhancing the performance of a model by receiving support from the other model. Figure 1 shows a framework in which three categories of hybridization are observed from the literature for combining white-and-black box learning together.

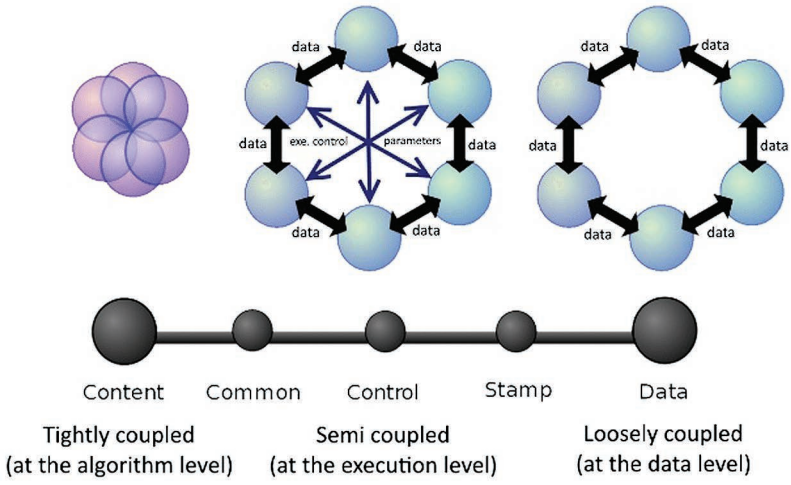


FIGURE 1. Framework of loosely-semi-and-tightly coupled models for white learning.

One of the most important tightly coupled white learning models is the Bayesian artificial neural network. This Bayesian neural network naturally can solve the model overfitting problem. Out of an entire distribution and possible neural network of different sizes, Monte Carlo simulation is often used in the model selection. This classical concept of using Bayesian probability distribution to grow a neural network dates back to the 1990s[1] and developed in the paper by Freitas[2] with application cases of financial stock prediction and recently applied for predicting and explaining cases of Alzheimer's disease.[3] Lately, researchers from Intel AI lab extended this principle for deep learning. Rohekar *et al.*[4] proposed a hierarchical network structure where the conditional independences between the depth and internetwork-layer connectivity can be controlled. The network structure, hence, can grow according to the BN structure learning. To this end, instead of learning for the classification power, it learns for obtaining a generative graph and then into a class-conditional discriminative graph in which the conditional-dependence relations among the variables are preserved. Automatically, the optimal network depth is, hence, determined, resulting in a compact deep learning model that has a good level of accuracy in image classification. Chaturvedi *et al.*[5] proposed a Bayesian deep CNN, which uses Gaussian networks for learning features in a CNN.

A semicoupled white learning model in our white learning framework is characterized by building two or more machine learning models in the form of Bayesian or similar white box decision tree model, and deep learning or neural network and the like, by transfer learning. Information or prior knowledge learnt from one model is passed onto the other model in such a way that the learned outcomes could benefit the construction of another model. In a semicoupled white learning model, either control, knowledge, or messages, which are related to shaping up the machine learning model, are passed from one model to another. The Ph.D. thesis by Krakovna[6] is focused on designing a new semicoupled white learning system, which combines recurrent neural networks and the hidden Markov model (HMM). In particular, Viktoriya attempts to train an HMM based on a long-short-term-memory neural network (LSTM). The HMM is first trained, followed by another training given by the state distributions from a small LSTM for making up the shortcoming of HMM's performance. The work shows that HMM and LSTM can learn complementary information about the feature; hence, feature selection is cooperatively applied together. Choi *et al.*[7] have built a Hybrid BN for predicting breast cancer prognosis, which has a certain level of accuracy by neural network and interpretation by the BN. First, the neural network and BN are independently built. The confidence value of the neural network output node is sent to the BN model as an additional input node. Therefore, the confidence value of the neural network output node is used as an additional input node in the hybrid BN model. A hybrid BN model, which uses the confidence value of the neural network output node, is used to predict the survival rate of a breast cancer patient. On a similar note, Antal *et al.*[8] infuse prior knowledge into the neural network from BN. It is done by encoding the prior knowledge in the format of a BN, which is called a belief network in his proposed methodology. Then, the prior knowledge is used to estimate information prior for the neural network. Two transformation methods are proposed: The first technique generates data samples according to the most probable parameterization of the Bayesian belief network, then combining the generated data and the original training data in the Bayesian learning of an NN. The second technique transforms probability distributions over belief network parameters into the statistical distributions over NN parameters. At the data level, where white box BN and black box NN are established and function independently, the only connection is of the passing data. This is called loosely coupled white learning in our framework. Pang *et al.*[9] proposed an ensemble, which consists of four popular individual classifiers which are trained individually using backpropagation NN, evolutionary NN, SVM, and decision tree. Then, the learning results from each classifier are combined and put in a combined BN. In other words, the predicted class variable is put at the last column, then the outputs of each classifier are put as the values of features, for forming up a new training dataset. The dataset then is used to train a combined BN. In this process, all the classifiers are unchanged; they are constructed and operate independently of each other. Similar work by Garg[10] was done by treating several BN independently and putting them together as an ensemble. The output of the BN that produces the highest level of accuracy is chosen as the final prediction. On one extreme, only the training dataset is shared, and no model is modified; Correa *et al.*[11] put several BN and NN together for execution. The model that offers the highest level of accuracy is taken as the predicted result.

EXPERIMENT

As a proof of concept for validating white-learning having an edge in accuracy performance and interoperability for cancer diagnosis, a novel white learning model is proposed with experimentation. The proposed model is similar to misclassified recall (MR),[12,13] which was proposed earlier for enhancing the accuracy level for life-science disease prediction and IoT extreme automation data stream mining. In the context of white learning, MR works by placing a BN cascading by a CNN. So, at the end of training/testing, the model results in two classifiers. One is representing white box by BN, which explains the posterior probabilities and causality about how a testing instance results in one of the hypotheses. The other one focuses on predicting the outcome with certainty, which is taken from the accuracy level of that particular prediction with a value capped between 0% and 100%. This is a typical case of white learning that is loosely coupled between BN and CNN deep learning, having the best of both white-and-black box learning. MR works by first

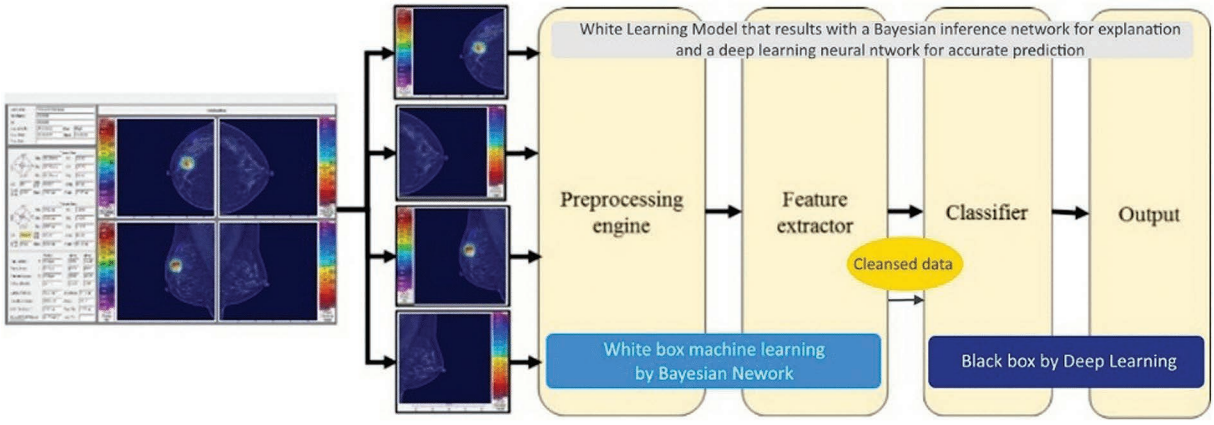


FIGURE 2. White learning model for breast cancer detection.

filtering the training dataset by an updatable BN that is believed to arrive at a high speed incrementally in an extreme automation scenario such as a cloud-based large-scale AI prediction online system. The algorithm which is used in the experimentation, as the preprocessing filter and white-box learner called *NaiveBayesUpdateable,* is available in R[14] and Java.[15] The quality of the training data would be improved because the misclassified instances would be eliminated, and the prior probabilities of those noisy data would not be included in the BN. Subsequently, the filtered training data would be passed onto the CNN for deep learning, in such a way that overfitting will be unlikely to occur because the noisy instances are removed in advance. As a result, the CNN structure shall be kept optimal, as is the prediction performance.

In the experiment, breast cancer data are simulated with an increase of artificially instilled random noise, from 0% to 20% with an increment of 5% at each step. The data are anonymized and composed of 1000 records of patient's cells information—six cells are selected from mammogram by a human expert, which are deemed to be most prominent as salient features describing the severity of the alleged disease. Each of the six cells is described by the aspects of radius, texture, perimeter, area, smoothness, compactness, concavity, concave points, symmetry, and fractal dimension of the cell. In total, there are 60 attributes. The white learning methodology in the case breast cancer detection is shown in Figure 2, where it has both explanatory part and predictive part of the machine learning model using BN and CNN, respectively.

Figures 3 and 4 show the comparison of white learning, NaiveBayesUpdatable, and deep learning for accuracy and model training/testing times, respectively.

In Figure 3, an interesting phenomenon is observed—the deep learning curve fluctuates greatly when the error level reaches 6% and beyond. That shows the CNN became very unstable, though the performance of CNN is known to be very sensitive to parameter settings. When the environment is noise-free, the performance for all the learning algorithms is at around 90% accuracy. At a noise level 6%, the accuracy of deep learning dipped to about 38%, which is even worse than at the maximum noise level 20% with accuracy of 41%. Among the three learning algorithms, deep learning is the worst, and white learning is the best. The differences of performance in magnitudes are best seen by the trend-lines, which are added to each of the curve of the learning algorithm. It can be observed that the gradients of the trend-lines for deep learning, *NaiveBayesUpdatable,* and white learning are –2.4843, –1.1614, and –1.2979, respectively. In general, the deeper the dive means the faster the accuracy will degenerate under noise. BN is most sustainable under noise in terms of the rate of accuracy degradation. At the best case, where there is no noise, the accuracy rates for deep learning, *NaiveBayesUpdatable,* and white learning are 96.1765%, 93.8235%, and 96.4706%, respectively. At the worst case of our simulation, the accuracy rates of the same are 41.1765%, 70.8824%, and 72.3529%, respectively. White learning demonstrates its superiority as a hybrid in

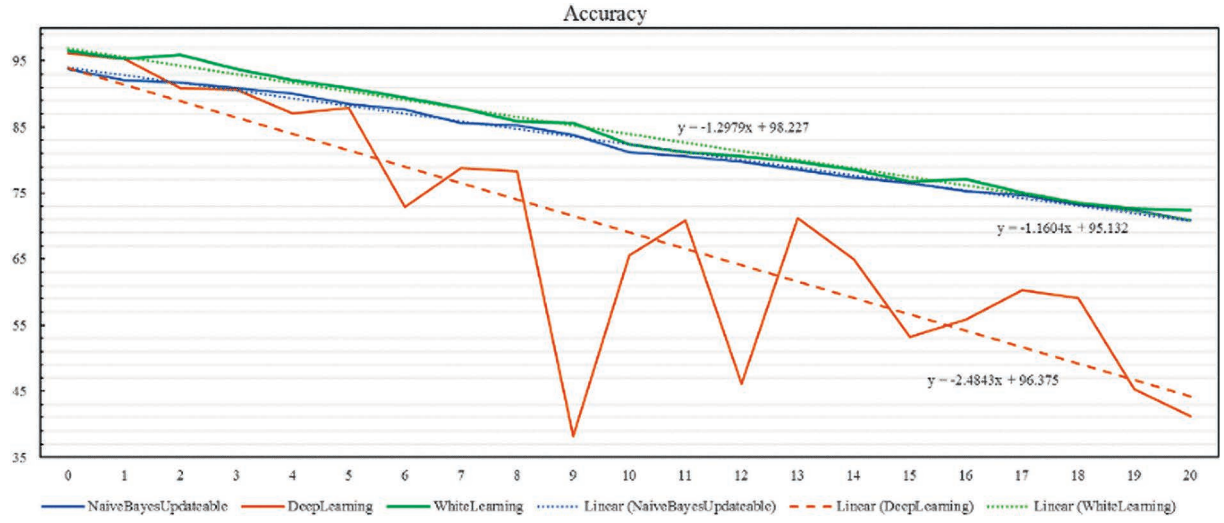


FIGURE 3. Comparison of white learning, *NaiveBayesUpdatable*, and deep learning in terms of accuracy.

FIGURE 4. Comparison of white learning, *NaiveBayesUpdatable,* and deep learning in terms of model training/testing times.

comparison to the individual NB and DL. The training time for White learning is shorter than DL as well, implying its suitability for extreme automation.

CONCLUSIONS

Deep learning as a data modeling tool is hard to be understood of how its predicted result came about from its inner working. It is generally known as a black box and is not interpretable. Often in medical applications, physicians need to understand why a model predicts a result. On the other hand, BN is a probabilistic graph with nodes representing the variables, and the arcs present the conditional dependences between the variables. Prior knowledge and reasoning of how

a predicted outcome come about via examining the probability distribution associate with each node and the dependencies among them can be made possible. In this article, a white learning framework is proposed, which advocates three levels of fusing the black-box deep learning and white-box BN, that offers both predictive power and interpretability. A case of breast cancer classification is conducted in an experiment. From the results, it is observed that white learning, which combines black-box and white-box machine learning, has an edge in performance over individually BN alone or deep learning alone. The white learning framework has the benefits of interpretability and high predictive power, making it suitable for critical decision-making task where a reliable prediction is as important as knowing how the outcome is predicted. The predicted output, which is generated from white learning, can be traced back via the conditional probability at each node. It is, hence, anticipated that in the future, especially for medical domain, white learning, which has the benefits of both black-box and white-box learning, would be highly valued and raised in popularity.

REFERENCES

1. C. E. Rasmussen, "A practical Monte Carlo implementation of Bayesian learning," in *Proc. 8th Int. Conf. Neural Inf. Process. Syst.*, Denver, CO, USA, Dec. 1995, pp. 598–604.
2. J. F. G. de Freitas, "Bayesian methods for neural networks," Ph.D. dissertation, Eng. Dept., Cambridge Univ., Cambridge, U.K., 2009.
3. W. Fruehwirt, et al., "Bayesian deep neural networks for low-cost neurophysiological markers of Alzheimer's disease severity," in *Proc. Conf. Neural Inf. Process. Syst., Mach. Learn. Health Workshop*, Dec. 2019, pp. 1–6.
4. R. Y. Rohekar, S. Nisimov, Y. Gurwicz, G. Koren, and G. Novik, "Constructing deep neural networks by Bayesian network structure learning," in *Proc. 32nd Conf. Neural Inf. Process. Syst.*, Montreal, QC, Canada, 2018, pp. 1–12.
5. I. Chaturvedi, E. Cambria, S. Poria, and R. Bajpai, "Bayesian deep convolution belief networks for subjectivity detection," in *Proc. IEEE 16th Int. Conf. Data Mining Workshops*, Barcelona, Spain, Dec. 12–15, 2016, pp. 916–923.
6. V. Krakovna, "Building interpretable models: From Bayesian networks to neural networks," Ph.D. dissertation, Graduate School Arts Sci., Harvard Univ., Cambridge, MA, USA, 2016.
7. J. P. Choi, T. H. Han, and R. W. Park, "A hybrid Bayesian network model for predicting breast cancer prognosis," *J. Korean Soc. Med. Informat.*, vol. 15, no. 1, pp. 49–57, 2009.
8. P. Antal, G. Fannes, D. Timmerman, Y. Moreau, and B. De Moor, "Bayesian applications of belief networks and multilayer perceptrons for ovarian tumor classification with rejection," *Artif. Intell. Med.*, vol. 29, pp. 39–60, 2003.
9. S. Pang, Y. Jia, R. Stones, G. Wang, and X. Liu, "A combined Bayesian network method for predicting drive failure times from SMART attributes," in *Proc. Int. Joint Conf. Neural Netw.*, Jul. 24–29, 2016, pp. 4850–4856.
10. A. Garg, V. Pavlovic, and T. S. Huang, "Bayesian networks as ensemble of classifiers," in *Proc. Object Recognit. Supported by User Interaction for Serv. Robots*," Aug. 11–15, 2002, pp. 779–784.
11. M. Correa, C. Bielza, and J. Pamies-Teixeir, "Comparison of Bayesian networks and artificial neural networks for quality detection in a machining process," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 7270–7279, Apr. 2009.
12. S. Fong, J. Li, W. Song, Y. Tian, R. K. Wong, and N. Dey, "Predicting unusual energy consumption events from smart home sensor network by data stream mining with misclassified recall," *J. Ambient Intell. Humanized Comput.*, vol. V9, no. I4, pp. 1197–1221, Aug. 2018.
13. D. Wang, S. Fong, R. K. Wong, S. Mohammed, J. Fiaidhi, and K. K. L. Wong, "Robust high-dimensional bioinformatics data streams mining by ODR-ioVFDT," *Sci. Rep.*, vol. 7, 2017, Art. no. 43167, 2017.
14. 2019. [Online]. Available: https://rdrr.io/github/aschersleben/NBCD/man/update.nb2.html
15. 1995. [Online]. Available: http://weka.sourceforge.net/doc.dev/weka/classifiers/bayes/NaiveBayesUpdateable.html

TENGYUE LI is currently working toward the Ph.D. degree at the University of Macau, Macau, China. She is also the Head of Data Analytics and Collaborative Computing Laboratory, Zhuhai Institute of Advanced Technology, Chinese Academy of Science, Zhuhai, China. She is leading and managing the laboratory in R&D as well as technological transfer and

incubation. She is an Entrepreneur with experience in innovative I.T. contests, with her award-winning team at the Bank of China Million Dollar Cup competition. Her latest winning work includes the first unmanned supermarket in Macau enabled by the latest sensing technologies, face recognition, and e-payment systems. She is also the Founder of several Online2Offline dot.com companies in trading and retailing both online and in an office environment. Contact her at mb75436@umac.mo.

SIMON FONG is currently an Associate Professor with the Computer and Information Science, Department of the University of Macau, Macau, China, and an Adjunct Professor with the Faculty of Informatics, Durban University of Technology, Durban, South Africa. He is a Cofounder of the Data Analytics and Collaborative Computing Research Group, the Faculty of Science and Technology. His research interests include the areas of data mining, data stream mining, big data analytics, meta-heuristics optimization algorithms, and their applications. He received the Graduate degree from La Trobe University, Melbourne, VIC, Australia, with a First-Class Honors B.Eng. degree in computer systems and the Ph.D. degree in computer science in 1993 and 1998, respectively. Contact him at ccfong@umac.mo.

LIAN-SHENG LIU is a deputy chief physician of the First Affiliated Hospital of Guangzhou University of Traditional Chinese Medicine, as well as a secretary of the Department of Imaging. He serves as a member of the Head and Neck Study Group of the Radiation Society of Guangdong Province, Chinese Medical Association. From 1996 to 2004, he was with the Imaging Center of Luzhou Hospital (Cangzhou People's Hospital of Jiangxi Province). He graduated from Jinan University in 2007, majoring in imaging medicine and nuclear medicine. His research direction is mainly on head and neck imaging. He was awarded the "Southern Guangdong Excellent Graduate Student" and the "First Prize of Jinan University Outstanding Graduate Student" in 2007.

XIN-SHE YANG is a Reader in Computational Modelling and Simulation with the School of Science and Technology, Middlesex University. He was the winner of the Shaanxi Province Distinguished Talented Professorship award from Xi'an Polytechnic University in 2011. He is an Adjunct Professor with Reykjavik University, Iceland, and a Guest Professor of both Harbin Engineering University and Shandong University, China. He is also a Visiting Professor at Megatrend University Belgrade, Serbia. Before he joined Middlesex University, he was a Senior Research Scientist at the U.K.'s National Physical Laboratory and Cambridge University after receiving the D.Phil. degree in applied mathematics from the University of Oxford.

XINGSHI HE graduated from the Mathematics Department of Shannxi Normal University, China, with the bachelor's and master's degrees in 1982 and 1987, respectively. He did teaching and research on probability and statistics with the Mathematics Department of Shannxi Normal University from 1982 to 1985. He is now a Professor with the School of Science, Xi'an Polytechnic University, and a director of the National Business Statistics Association and a standing director of the Shaanxi Industrial and Applied Mathematics Society. His research interests are in the areas of statistics, mathematical modeling, big data analytics, meta-heuristics optimization algorithms, and their applications.

JINAN FIAIDHI is a Full Professor in computer science and the Graduate Coordinator with the BioTech Ph.D. program, Lakehead University, Thunder Bay, ON, Canada. She is an Adjunct Research Professor with the University of Western Ontario and the Editor in Chief of the IGI Global International Journal of Extreme Automation and Connectivity in Healthcare. She is also the Chair of Big Data for eHealth with the IEEE Communications Society. Contact her at jfiaidhi @lakeheadu.ca.

SABAH MOHAMMED is a Full Professor with the Department of Computer Science and supervisor of the Smart Health FabLab, Lakehead University, Thunder Bay, ON, Canada. He is also an Adjunct Professor with the University of Western Ontario, London, ON, Canada, and is the Chair of Smart and Connected Health with the IEEE Communications Society. Contact him at mohammed@lakeheadu.ca.

EDITOR: Tim Menzies, North Carolina State University, tim@menzies.us

DEPARTMENT: REDIRECTIONS

This article originally appeared in
IEEE Software
vol. 36, no. 3, 2019

Is Your Software Valueless?

Jon Whittle

Software development ignores human values. As a society, we rely on software systems that neither align with nor respect our core values, such as transparency, gender diversity, social justice, and personal integrity. The past 50 years of software engineering have focused on functionality, cost, safety, availability, and security. But what about broader human values (Figure 1) such as compassion, social responsibility, and justice? The way we design software fundamentally influences society, yet human values—which we would all claim to care about—have been a side concern in software engineering. (See "Where Are the Values in Software?")

Surely it is high time that we fundamentally reimagine the way we design software. Rather than focusing only on a narrow set of concerns, we should embed all human values into software design. If we don't, we, as software engineers, will inadvertently create a society that nobody wants.

Admittedly, embedding human values into software is difficult. Even where there is a willingness, and managerial support, to think about values, it proves challenging. Software development defaults back to values that are relatively easy to deal with—accessibility, usability, and availability—for which there are more clearly defined guidelines or tools. Thus, running tools on a graphical user interface that check whether a color scheme is readable by the color-blind population is common but does not help with addressing broader human values. There is a mismatch between what the software development community values—typically automation, productivity, and quality—and broader societal values (Figure 2). There is also often an assumption that the latter naturally leads to the former, that we are improving people's lives by automating things using quality software.

Why do values in software matter? The issue of values, and more narrowly ethics, in computing is receiving renewed attention because there are doomsday predictions of discriminatory artificially intelligent systems taking over the world. However, the real problems are much more mundane than are those of sentient machines. The interface for a simple human resources (HR) recruitment system is one example. Most job interviewers are encouraged, if not mandated, to consider periods of time away from work, e.g., parental leave. Yet HR recruitment systems do not prominently display such information; to discover it, interviewers must search through potentially hundreds of curriculum vitaes. If such a system was designed with the value of gender equality in mind, the interface would be designed very differently. We have known for decades that design is not values neutral, but software engineers have failed to understand this. We examined the last four years of papers published in the top software engineering conferences and journals; we found that only 16% of papers considered values at all. Of those, a significant majority focused on values of security and privacy.

IF SUCH A SYSTEM WAS DESIGNED WITH THE VALUE OF GENDER EQUALITY IN MIND, THE INTERFACE WOULD BE DESIGNED VERY DIFFERENTLY.

But does the software industry care about nonfinancial values? A naive view might argue that it does not, that it is all about the bottom line. However,

Digital Object Identifier 10.1109/MS.2019.2897397
Date of publication: 16 April 2019

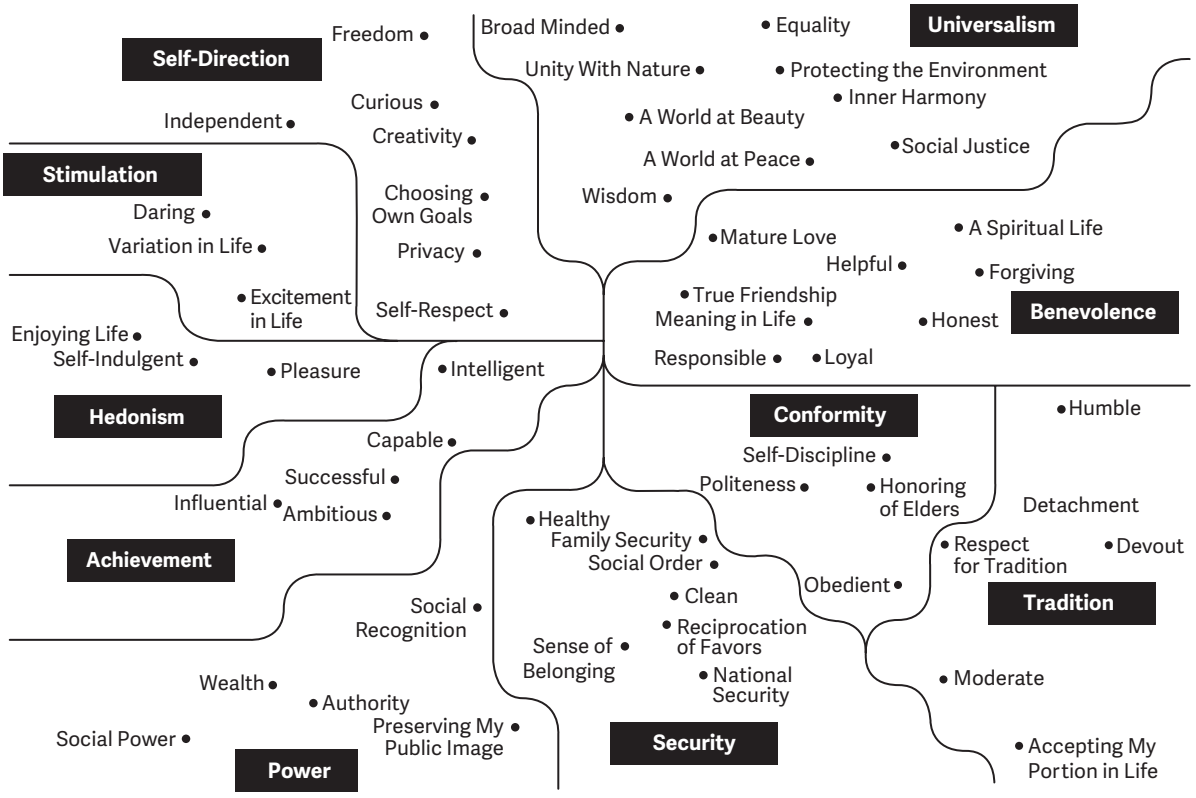


FIGURE 1. A number of theories have attempted to define values, such as the Schwartz Theory of Basic Values.[1] The 10 universal values are boxed and subdivided into finer-grained values. Values closer to each other (as defined by the bullet points) are complementary, and those farther apart are in conflict. Note that values are different from ethics: ethics are culturally agreed-upon moral principles, while values make no moral judgment, so, for example, making money is a perfectly acceptable value. (Adapted from Schwartz[8].)

companies have at least claimed to care about values for a long time. Ever since Jim Collins' and Jerry Porras' book *Built to Last*,[2] which found that a key determinant of a company's success is a strong values statement, organizations have put a lot of effort into defining their corporate values. A study by Maitland[3] found that 86 of Financial Times Stock Exchange 100 Index companies have public values statements, with values such as corporate integrity, respect, and honesty topping the list. Clearly, companies implement these with various levels of seriousness, but many, if not all, do take them seriously and have managerial mechanisms to create a values culture. However, there is no way to trickle these values down into the software that we build.

At Monash University, Melbourne, Australia, we recently conducted two case studies with software companies to improve our understanding of their approaches to values. At least for these companies, values are explicitly talked about during software development: this usually takes the form of a values document used hiring decisions, training new staff, performance appraisals, and strategic decision making. Sometimes there are more sophisticated ways to create an open culture, ones in which software developers can talk about and honestly question company values. These include forums for stepping back from day-to-day concerns, having someone designated as a critical friend, clarity in hiring practices, and training programs. However, even in those cases in which values are considered, the approach is limited to creating a values-driven culture as opposed to having it engrained into the act of software development. When companies consider values in developing software, it is during business analysis and requirements engineering only; values are easily forgotten later.

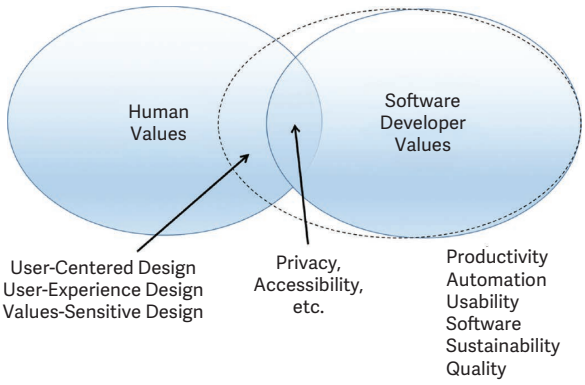


FIGURE 2. A software developer's values versus human values: software engineers value technical concepts such as productivity, automation, usability, and quality, with the assumption that these traits naturally lead to broader human values. This is a naive view, however; human values and software developer values rarely coexist. Methods such as user-centered design, user-experience design, and values-sensitive design[4] take a broader view, but we are a long way from full alignment between human and developer values.

WHERE ARE THE VALUES IN SOFTWARE?

Value is an overloaded term. Values-based methods are well known in human–computer interaction (HCI) and information systems[4,9,10] but are nonexistent in software engineering. HCI and information systems do not deal with the business of actually building software, so although they could apply in the early stage of software engineering, they offer little guidance as to how to handle values in the more technical stages of development. The word *value* is often referred to in agile methods, but then the focus is only on business value. Similarly, Boehm's value-based software engineering[11] deals almost entirely with economic value. Some emerging works in software engineering take a more human-values approach, such as GenderMag[12] for discovering gender bias in software, but this is still very early.

Therefore, the current state, at least for software companies that appreciate values, is a reliance on organizational culture. However, there is very little, if anything, to support technical work. This is not all bad news, however. Many existing software development approaches can be adapted easily to work with values. Therefore, while a revolution in the mind-set of developers is necessary, more of an evolution could suffice from a process point of view. For example, agile development methods lend themselves naturally to thinking about values. In their current form, values will not immediately receive attention, but through a designated values guardian in a Scrum team, user stories could easily become values stories, and measurement approximations, such as T-shirt sizing, would be useful for dealing with the inherent complexity of a nuanced concept such as values. Furthermore, we would go a long way by introducing well-established participatory design techniques into user-experience and user-centered design approaches. Participatory design methods are good at ensuring that end-user values are taken into account but have suffered from a lack of clarity and a refusal to simplify.[5]

More generally, there could be values versions of successful tricks that the software industry has used. Imagine a values manifesto with the beauty and simplicity of the agile manifesto, making it clear to developers that values are important. Or there could be a values maturity model that helps organizations to self-assess their values culture, such as level 0, ad-hoc consideration of values; level 1, a clear, published corporate values statement but no real way to implement it; level 2, some processes to deal with values; level 3, proactive and structured ways to ensure that values are considered at all stages of the software lifecycle; and level 4, software tools to support values. You get the idea. Requirements engineering methods could be applied easily to refine what values mean. After all, one of the biggest challenges in instilling values in software is that values are, by definition, vague concepts. However, specifying values in concrete terms, in the context of an actual project, plays to the strengths of requirements engineering methods. Also, well-accepted technical methods could be adapted to look at software development through a values lens: take A/B testing, for example,

which could be used to test out how different software versions impact values.

It is time that the software industry takes human values seriously, but not just for the greater social good. Violations of human values can have serious negative financial consequences for the economy. In a sample of Internet security breaches, Cavusoglu[6] found an average market-capitalization loss of US$1.65 billion for the companies affected. In the Volkswagen (VW) emissions scandal,[7] software designers deliberately contradicted the company's corporate value of responsible thinking, a decision that led to the resignation of the chief executive officer, a 30% drop in VW's stock price, and a 25% drop in sales within one year. Therefore, value violations are big business. Software researchers and practitioners must respond by doing what they do best: creating methods for handling such problems before a catastrophe hits.

ACKNOWLEDGMENTS

The ideas in this article benefited from discussions with Waqar Hussain and Davoud Mougouei.

REFERENCES

1. S. Schwartz, "Basic human values: Theory, measurement, and applications," *Revue Francaise de Sociologie*, vol. 47, no. 4, pp. 929–985, 2006.
2. J. Collins and J. Porras, *Built to Last: Successful Habits of Visionary Companies*. New York: HarperBusiness, 1997.
3. S. Walker, "*The values most valued by UK plc*," Maitland. Accessed on: Feb. 20, 2019. [Online]. Available: http://www.maitland.co.uk/wp-content/uploads/2015/10/20151001-Maitland-Values-Report.pdf
4. B. Friedman, D. Hendry, and A. Borning, "A survey of value sensitive design methods," in *Foundations and Trends in Human Computer Interaction*. Boston: Now Publishers, 2017.
5. J. Whittle, "How much participation is enough? A comparison of six participatory design projects in terms of outcomes," in *Proc. Participatory Design Conf.*, 2014, pp. 121–130.
6. H. Cavusoglu, B. Mishra, and S. Raghunathan, "The effect of Internet security breach announcements on market value: Capital market reactions for breached firms and Internet security developers," *Int. J. Electron. Commerce*, vol. 9, no. 1, pp. 70–104, 2004.
7. B. Georgievski and A. Alqudah, "The effect of the Volkswagen scandal: A comparative case study," *Res. J. Finance Accounting*, vol. 7, no. 2, pp. 54–57, 2016.
8. S. Schwartz, "I've built a good mousetrap and people come to use it," *Psychologist*, vol. 31, pp. 56–59, June 2018.
9. C. Knobel and G. Bowker, "Values in design," *Commun. ACM*, vol. 54, no. 7, pp. 26–28, 2011.
10. G. Cockton, "Value-centred HCI," in *Proc. NordCHI*, 2004, pp. 149–160.
11. S. Biffl, A. Aurum, B. Boehm, H. Erdogmus, and P. Grunbacher, *Value-Based Software Engineering*. New York: Springer, 2006.
12. M. Burnett et al., "GenderMag: A method for evaluating software's gender inclusiveness," *Interacting Computers*, vol. 28, no. 6, pp. 760–787, 2016.

JON WHITTLE is the dean of the faculty of information technology at Monash University. Contact him at Jon.Whittle@monash.edu.

EDITORIAL

This article originally appeared in
IEEE MultiMedia
vol. 27, no. 3, 2020

Shaping Our Common Digital Future

Susanne Boll, *University of Oldenburg*

The global spread of a vicious disease in our interconnected world is threatening the health and livelihoods of millions of people. Beyond the immediate effects of the disease on individuals, families, and communities, we can anticipate the long-term impact on whole societies and economies. Our lives are changing not only because of the coronavirus pandemic, but also because of climate change and environmental damage. These are the defining crises of our time, and they are shining a harsh spotlight on the intractable socioeconomic inequalities long plaguing the world's people. We cannot meet these challenges only on a local or national scale. Global crises require a global response.

MULTIMEDIA IN TIMES OF THE PANDEMIC

We have only begun to understand the importance of multimedia communication in the face of a pandemic. Electronic products and services, especially interactive ones, that combine text, sound, video, etc., quickly proved essential socially and economically when global quarantine became necessary. Previous multimedia research and existing tools have contributed a "safety net" of sorts to allow continuation of at least some education, business, and government communication.

Research in multimedia over the past decades has contributed to understanding, interpreting, transporting, delivering, and interactively presenting multimedia experiences across many domains. Software and tools rooted in this field support multimedia networking and streaming, interactive video conferencing, and communication and interaction on social media. Now, physical events and meetings, including those of global leaders, have by necessity become virtual. Multimedia research has thus become mainstream and usable for everyone. Confined by stay-at-home orders, we have found tools to connect, to keep in touch, to work and learn. Even when this disease is brought under control, however, our daily lives will never be what they were. Furthermore, the crisis has starkly exposed long-troubling, deep social and economic inequalities. Consequently, the questions now are how will this pandemic transform our future work life and educational systems, and how can we use this transformation to level the playing field, to address inequalities wasting so much human potential around the world?

Multimedia technologies are already building blocks for many application domains much needed in these days: Health care, education, additive manufacturing, logistics, crisis management, and many more. So, we could sit back and be satisfied—or we could understand our field from a philanthropic angle and help shape our common digital future, positively and inclusively.

MULTIMEDIA FOR OUR COMMON DIGITAL FUTURE

The challenges of the day have been well framed by the United Nations, when in 2015 they decided on a 2030 Agenda for Sustainable Development. This agenda "is a plan of action for people, planet and prosperity" and forms "universal goals and targets which involve the entire world, developed and developing countries alike."[1] The agenda addresses

Digital Object Identifier 10.1109/MMUL.2020.3017875
Date of current version 28 August 2020.

17 sustainability goals and describes actionable objectives, from ending poverty to ensuring access to clean water and clean energy, to education and decent work for all. In meeting these Sustainable Development Goals (SDGs), digitization will play an important, even transformative, role.

Recently, the German Advisory Council on Global Change published their Flagship report, "Towards Our Common Digital Future."[2] This excellent, comprehensive report describes the enormous potential digitization holds for our common digital future: "Digital change is epochal and opens the door to a new era of human development." The report frames digitization as an opportunity to shape the digital societies of the future and lays out how to shape the "Great Transformation" to address sustainability goals. The advisory council not only sees digital technologies as important for this transformation but also emphasizes the necessity to link digitization and sustainability.

Technology and science will play an important role in this common digital future, and so can the field of multimedia. However, technological advances alone are not necessarily a sure-fire success. We witnessed several examples in the last decade showing that the narrative of an always positive use of digitization cannot be told anymore—digital technologies can be used not only for the good of humans but also to their detriment. It is on us to actively shape this change for the better, for all of us.

MULTIMEDIA—WHERE TO GO?

Multimedia can be a rich source for addressing many global challenges. Here, we focus on the potential of multimedia to advance progress toward selected SDGs.

The Sustainable Development Goal 3: *Good Health and Well-being* focuses on the severe inequalities worldwide that leave much of the world's population struggling just to survive, much less experience good health and a sense of well-being. It is time to mount a concerted global effort to alleviate this condition. Multimedia can be instrumental to implementing global solutions. Multimedia researchers have already contributed to significant advances in personal health, from multimedia signals to a new generation of future personal digital health technology.[3] Multimedia can act as an accelerator for understanding personal health and supporting the individual in gaining and maintaining good health.[4] Current developments have only begun to unfold the potential to better understand, diagnose and predict courses of disease, and to contribute dramatically to universal health solutions.

The Sustainable Development Goal 4: *Quality Education:* aims to ensure inclusive and equitable, quality education and to promote lifelong learning opportunities for all. While the field of multimedia has been working for some time on interactive digital education and social media for learning,[5] the pandemic has given digital education a boost. It revealed the gap between digital technologies and digital education. The challenge is to integrate these new technologies into our learning contexts and curricula and use them to provide high quality education to *everyone*.

The Sustainable Development Goal 8: *Decent work and economic growth* aims to promote inclusive and sustainable economic growth, full and productive employment, and decent work for all. Digitization, along with multimedia and interactive technologies, will be the driving force of the workplace of the future. Widespread transformation of the workplace will require that people accept and want to use digital technologies.[6] Participatory design work can result in new technologies conducive to learning, to inclusion, and to access for the transformed job market of the future.

The Sustainable Development Goal 10: *Reduced Inequalities* aims at reducing inequality within and among countries. In our field, social media usage and social media coverage are studied to understand political information and disinformation on social media and how news is perceived on social media around the globe.[7]

For example, existing work has investigated the role of social media in political engagement and the technologies in play dispensing political information and mediating political engagement. We must develop technologies that allow people to "freely express themselves, access trustworthy information, engage in meaningful deliberation, and organize themselves without fear of being commoditized, manipulated, monitored, or harassed by authorities."[8]

Currently these SDGs are mapped to national research agendas. In Germany, for example, you will

now find different SDG objectives to be addressed in different calls for grant proposals.

We need to discuss and identify how the field of multimedia can contribute to a positive digital future for all of us. What does this mean for researchers and practitioners in higher education institutions, research institutes, and companies, and what can be our personal contribution to society to shape a digital future for the betterment of all?

REFERENCES

1. United Nations. Transforming Our World. The 2030 Agenda for Sustainable Development, 2015. [Online]. Available: https://sustainabledevelopment.un.org/content/documents/21252030%20Agenda%20for%20Sustainable%20Development%20web.pdf
2. German Advisory Council on Global Change. Flagship Report: Towards Our Common Digital Future. 2019. [Online]. Available: https://www.wbgu.de/fileadmin/user_upload/wbgu/publikationen/hauptgutachten/hg2019/pdf/wbgu_hg2019_en.pdf
3. P. Cesar, V. Singh, R. Jain, N. Sebe, and N. Oliver, "New signals in multimedia systems and applications," *IEEE MultiMedia*, vol. 25, no. 1, pp. 12–13, Jan.–Mar. 2018.
4. S. Boll, J. Meyer and N. E. O'Connor, "Health media: From multimedia signals to personal health insights," *IEEE MultiMedia*, vol. 25, no. 1, pp. 51–60, Jan.–Mar. 2018.
5. Q. Li, R. W. H. Lau, E. Popescu, Y. Rao, H. Leung, and X. Zhu, "Social media for ubiquitous learning and adaptive tutoring [guest editors' introduction]," *IEEE MultiMedia*, vol. 23, no. 1, pp. 18–24, Jan.–Mar. 2016.
6. O. Bode, S. Boll, S. Falk, M. Koch, S. Riedel, and V. Wulf, "Arbeitswelten der Zukunft gestalten! (Shaping the working worlds of the future!)," *i-com*, vol. 17, no. 3, pp. 265–266, 2018.
7. A. El Ali, T. C. Stratmann, S. Park, J. Schöning, W. Heuten, and S. C. J. Boll, "Measuring, understanding, and classifying news media sympathy on twitter after crisis events," in *Proc. CHI Conf. Human Factors Comput. Syst. Assoc. Comput. Machinery*, New York, NY, USA, 2018, Paper 556.
8. D. Gayo-Avello, "Social media, democracy, and democratization," *IEEE MultiMedia*, vol. 22, no. 2, pp. 10–16, Apr.–Jun. 2015.

Conference Calendar

IEEE Computer Society conferences are valuable forums for learning on broad and dynamically shifting topics from within the computing profession. With over 200 conferences featuring leading experts and thought leaders, we have an event that is right for you. Questions? Contact conferences@computer.org.

FEBRUARY

27 February

- CGO (IEEE/ACM Int'l Symposium on Code Generation and Optimization), virtual
- HPCA (IEEE Int'l Symposium on High-Performance Computer Architecture), virtual

MARCH

9 March

- SANER (IEEE Int'l Conf. on Software Analysis, Evolution and Reengineering), virtual

22 March

- ICSA (IEEE Int'l Conf. on Software Architecture), Stuttgart, Germany
- MIPR (IEEE Int'l Conf. on Multimedia Information Processing and Retrieval), Tokyo, Japan
- PerCom (IEEE Int'l Conf. on Pervasive Computing and Communications), Kassel, Germany

27 March

- IEEE VR (IEEE Conf. on Virtual Reality and 3D User Interfaces), Lisbon, Portugal

28 March

- ISPASS (IEEE Int'l Symposium on Performance Analysis of Systems and Software), Stony Brook, USA

APRIL

12 April

- ICST (IEEE Conf. on Software Testing, Verification and Validation), virtual

14 April

- COOL Chips (IEEE Symposium on Low-Power and High-Speed Chips and Systems), Tokyo, Japan

19 April

- ICDE (IEEE Int'l Conf. on Data Engineering), Chania, Greece

21 April

- SELSE (IEEE Workshop on Silicon Errors in Logic - System Effects), Los Angeles, USA

25 April

- VTS (IEEE VLSI Test Symposium), San Diego, USA

28 April

- MetroCAD (Int'l Conf. on Connected and Autonomous Driving), Detroit, USA

MAY

10 May

- CCGrid (IEEE/ACM Int'l Symposium on Cluster, Cloud and Internet Computing), Melbourne, Australia
- ICFEC (IEEE Int'l Conf. on Fog and Edge Computing), Melbourne, Australia

15 May

- BigDataSecurity (IEEE Int'l Conf. on Big Data Security on Cloud), New York, USA

17 May

- IPDPS (IEEE Int'l Parallel and Distributed Processing Symposium), Portland, Oregon, USA

18 May

- RTAS (IEEE Real-Time and Embedded Technology and Applications Symposium), Nashville, USA

22 May

- ISCA (ACM/IEEE Int'l Symposium on Computer Architecture), Valencia, Spain

23 May

- ICCP (IEEE Int'l Conf. on Computational Photography), Haifa, Israel
- ICSE (IEEE/ACM Int'l Conf. on Software Engineering), Madrid, Spain
- SP (IEEE Symposium on Security and Privacy), San Francisco, USA

25 May

- ISMVL (IEEE Int'l Symposium on Multiple-Valued Logic), Nur-Sultan, Kazakhstan

JUNE

1 June

- ISORC (IEEE Int'l Symposium

 2469-7087/21

on Real-Time Distributed Computing), Daegu, South Korea

7 June

- BCD (IEEE/ACIS Int'l Conf. on Big Data, Cloud Computing, and Data Science Eng.), Macao
- CBMS (IEEE Int'l Symposium on Computer-Based Medical Systems), Aveiro, Portugal
- WoWMoM (IEEE Int'l Symposium on a World of Wireless, Mobile and Multimedia Networks), Pisa, Italy

14 June

- ARITH (IEEE Int'l Symposium on Computer Arithmetic), virtual

20 June

- SERA (IEEE/ACIS Int'l Conf. on Software Engineering Research, Management and Applications), Kanazawa, Japan

21 June

- CSF (IEEE Computer Security Foundations Symposium), Dubrovnik, Croatia
- DSN (IEEE/IFIP Int'l Conf. on Dependable Systems and Networks), Taipei, Taiwan

26 June

- CSCloud (IEEE Int'l Conf. on Cyber Security and Cloud Computing), Washington, DC, USA

JULY

5 July

- ICME (IEEE Int'l Conf. on Multimedia and Expo), Shenzhen, China
- SERVICES (IEEE World Congress on Services), Chicago, USA

7 July

- ICDCS (IEEE Int'l Conf. on Distributed Computing Systems), Washington, DC, USA
- SNPD (IEEE/ACIS Int'l Conf. on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing), Taichung, Taiwan

12 July

- COMPSAC (IEEE Computers, Software, and Applications Conf.), Madrid, Spain

27 July

- SMC-IT (IEEE Int'l Conf. on Space Mission Challenges for Information Technology), virtual

AUGUST

23 August

- SCC (IEEE Space Computing Conf.), virtual

SEPTEMBER

7 September

- CLUSTER (IEEE Int'l Conf. on Cluster Computing), Portland, Oregon, USA

20 September

- eScience (IEEE Int'l Conf. on eScience), Innsbruck, Austria

OCTOBER

13 October

- FIE (IEEE Frontiers in Education Conf.), Lincoln, Nebraska, USA
- ICIS Fall (IEEE/ACIS Int'l Fall Conf. on Computer and Information Science), Xi'an, China

29 October

- IPCCC (Int'l Performance Computing and Communications Conf.), Austin, USA

NOVEMBER

15 November

- ASE (IEEE/ACM Int'l Conf. on Automated Software Engineering), Melbourne, Australia

DECEMBER

20 December

- MCSoC (IEEE Int'l Symposium on Embedded Multicore/Many-Core Systems-on-Chip), Singapore

Get Published in the *IEEE Open Journal of the Computer Society*

Submit a paper today to the premier open access journal in computing and information technology.

Your research will benefit from the IEEE marketing launch and 5 million unique monthly users of the IEEE *Xplore*® Digital Library. Plus, this journal is fully open and compliant with funder mandates, including Plan S.

Submit your paper today!
Visit www.computer.org/oj to learn more.

75 YEARS IEEE COMPUTER SOCIETY | IEEE