Internet of Things, Ad Hoc and Sensor Networks Technical Committee Newsletter

(IoT-AHSN TCN)

Volume 1, No. 14

June, 2021

CONTENTS

PREFACE

The IEEE ComSoc Ad Hoc and Sensor Networks Technical Committee (IoT-AHSN TC) sponsors papers, discussions, and standards on all aspects of IoT, ad hoc and sensor networks. It provides a forum for members to exchange ideas, techniques, and applications, and share experience among researchers. Its areas of interest include systems and algorithmic aspects of sensor and ad hoc networks, networking protocols and architecture, embedded systems, middle-ware and information management, novel applications, flow control and admission control algorithms, network security, reliability, and management. In an attempt to make all the TC members as well as the IoT-AHSN worldwide community aware of what is going on within our main areas of concerns, this newsletter had been set up. The newsletter aims at inviting the authors of successful research projects and experts from all around the world with large vision about IoT-AHSN-related research activities to share their experience and knowledge by contributing in short news.

The fourteenth issue of the IoT-AHSN TC Newsletter focuses on the theme "Machine Learning for Internet of Things". Specifically, this issue includes 6 news articles: i) Automatic Street Parking Sign Reading; ii) On the Discovery of Orthogonal Memory Representation for Incremental Learning Enabled Pattern Recognition in IoT; iii) IoT Device Energy Consumption Measurement Using a Machine Learning Model; iv) Tradeoff between Complexity and Performance: Machine Learning Driven Resource Allocation in IoT; v) EMG Sensor Based Finger Movement Detection; vi) Understanding Human Factors for Machine Learning Algorithm Design over Edge IoT. We thank the contributors for their efforts to help make the IoT-AHSN TC Newsletter a success. We hope that the methods/approaches presented in this issue could significantly benefit researchers and application developers who are interested in IoT and ad hoc/sensor networks.

Newsletter Co-Editors
Qiang Ye (Dalhousie University, Canada)
Moez Esseghir (University of Technology of Troyes, France)
Lu Lv (Xidian University, China)

TC OFFICERS AND NEWSLETTER EDITORS

TC Officers

Name	Affiliation	Email
Soumaya Cherkaoui (Chair)	Université de Sherbrooke, Canada	soumaya.cherkaoui@usherbrooke.ca
Jiajia Liu (Vice Chair)	Xidian University, China	liujiajia@xidian.edu.cn
Sharief Oteafy (Secretary)	DePaul University, USA	soteafy@depaul.edu

Newsletter Editors

Name	Affiliation	Email
Qiang Ye (Editor in Chief)	Dalhousie University, Canada	qye@cs.dal.ca
Moez Esseghir (Technical Editor)	University of Technology of Troyes, France	moez.esseghir@utt.fr
Lu Lv (Technical Editor)	Xidian University, China	lulv@xidian.edu.cn

Automatic Street Parking Sign Reading

Jiayu Li, Putthida Samrith, Nicole Guobadia, Juhua Hu, Wei Cheng
Tacoma School of Engineering and Technology, University of Washington, USA
{jiayu7, psamrith, nguob, juhuah, uwcheng}@uw.edu

***Abstract*—As the number of transportation usage is increasing day by day, autonomous driving and parking have been investigated for years, where machine learning and IoT technologies are actively involved. To address the unsolved problem of automatically reading street parking signs in real time, so as to determine an appropriate place to park, we propose a new smart street parking system consisting of parking sign detection on roadsides, text detection and recognition in parking signs, special symbol detection in parking signs, and accurate interpretation of parking rules. With the innovation of Internet of Things (IoTs), we build a mobile application as a communication channel for human users to engage with our smart street parking system. Our application allows a user to take pictures of street parking signs and get an immediate response of what the parking signs show. The preliminary results illustrate that our system can successfully generate accurate street parking rules from different types of street parking signs including the scenario of multiple parking signs stacked at one place, and have higher accuracy than existing state-of-the-art methodologies like Google OCR.**

***Index Terms*—Street Parking, Computer Vision, IoT**

I. Introduction

In terms of transportation, the world was home to around 670 million vehicles in 1996, which was only 342 million vehicles in 1976. If this staggering rate of growth continues as doubling every 20 years, we can expect to see around 2.8 billion vehicles on the planet in 2036 [1]. As the world continues to urbanize, we expect to see more autonomous vehicles running on roads. These smart vehicles with the robust control system [2] can automatically avoid pedestrians, change lanes, maintain the safe distance between vehicles, and handle emergencies by employing computer vision technologies to perceive the environment and make appropriate predictions for decisions. Although machine learning and IoT technologies have been widely used to assist smart vehicles to interact with the real world, there is still an open problem in smart street parking, which is how to automatically find an appropriate parking spot through correctly detecting, recognizing, analyzing, and understanding the street parking signs on roadsides.

In this project, we aim to build an efficient and reliable street parking sign reading system, which accepts raw street images that are either photos taken by human users or image frames extracted from camera videos. To achieve this, the following procedure is proposed: 1) detect parking sign(s) from a street image, 2) detect and recognize text in the parking sign, 3) detect special symbols (i.e., no-parking symbol, handicap, etc.) in the parking sign, 4) generate accurate parking rules based on results obtained from the previous two stages. Besides, we aim to build a reliable system to interact with human users or autonomous cars based on this procedure.

II. Street Parking Sign Reading System

In this section, we describe the detailed techniques involved in each component of our proposed street parking sign reading system.

a) Parking Sign Detection: Given a raw image, the first step of this system is to detect parking signs, where bounding-box (BB) coordinates of each parking sign in the image are expected to be generated to crop parking sign only images. Apparently, this is an object detection problem, which can be divided into two categories: one-stage (e.g., [3], [4]) and two-stage (e.g, [5], [6]) approaches. Two-stage methods firstly generate regions of interests as candidates that are then sent to another neural network for object classification and BB regression. One-stage methods, accept the input image and predict the class probabilities and BB coordinates simultaneously. Therefore, one-stage approaches are less time-consuming, which is desired in our real-time system.

Although object detection has been extensively studied, only Irshad et al. [7] studied street parking sign detection using a google street view dataset collected in San Francisco. However, their parking signs are too small to recognize text that can be used in our system. Therefore, we build and annotate our own parking sign detection data, which contains 2,064 parking sign images from 10 states in US. Currently, RetinaNet50 [3] is chosen due to our preliminary empirical results summarized in Table I, where 1,832 images were used to fine-tune the parking sign detection model based on the pre-trained weights from ImageNet, and 232 images were used for test. RetinaNet50 has relatively better detection performance according to the mean Average Precision (mAP, $AP_{0.75}$ means AP for Intersection over Union (IoU) > 0.75) and higher inference speed according to the processing Frames Per Second (FPS). Yolov5, an upgraded version of Yolov3 [8] has been recently released. Therefore, a comprehensive comparison will be conducted with more collected training data.

TABLE I
Performance comparison of parking sign detection

Model	Backbone	mAP	$AP_{0.75}$	FPS
Faster_RCNN [6]	ResNet50	0.827	0.970	14.20
Yolov3 [8]	-	0.825	0.975	**18.79**
RetinaNet [3]	ResNet50	**0.849**	**0.981**	17.00

b) Text Detection and Recognition: To understand a parking sign, the information mostly comes from the text on the parking sign. A number of approaches for word detection and recognition in natural scene images have been

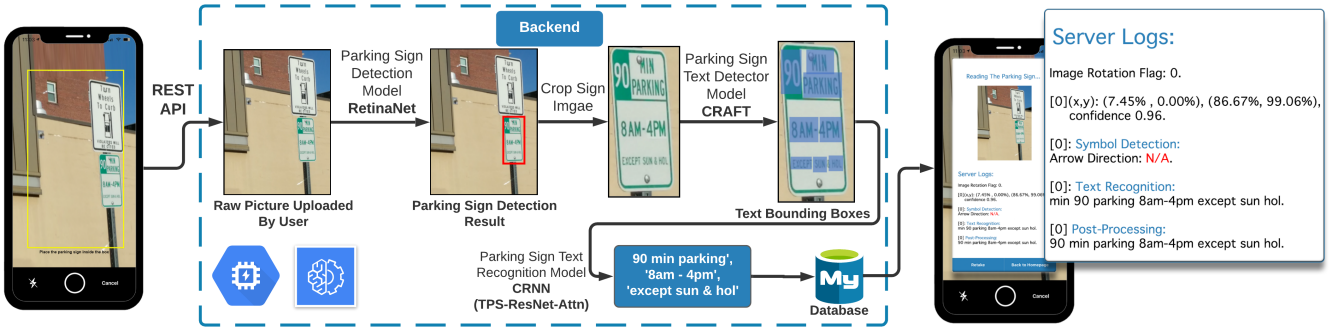

Fig. 1. An example of street parking sign interpretation procedure.

proposed [9], [10]. In our system of sign text detection, we adopt the CRAFT [11] framework. Unlike other word-level localization methods, character-level awareness CRAFT has better performance for curved and deformed texts, which often happens for parking sign images. Thereafter, the characters are recognized by a state-of-art approach, CRNN [12].

The first challenge of this component is that nobody has built any text detection or recognition training data for parking sign images before. Therefore, we use the cropped parking signs from our parking sign dataset and manually annotated 9,157 text BBs, for which all characters are well annotated. The second challenge is that to correctly interpret a parking sign, we need to figure out an accurate order of all text BBs within a parking sign. Therefore, for the first time, we propose a reorder function based on the IOU metric and the tilted angle of the parking sign as shown in Fig. 1. Currently, our framework provides an accuracy of 90.82% compared to Google OCR API that provides an accuracy of 87.18%, whose performance can be further improved by collecting more training data.

c) Special Symbol Detection: Besides text, other symbols such as 'no parking symbol', 'handicap', and 'arrow' are essential to avoid a parking ticket. No previous work has been done for this. Currently, we treat the symbol detection problem for each symbol as a binary/multi classification problem, e.g., with symbol and without symbol. Although an F1-score near 99% and an inference speed near 140 FPS can be reached using Xception [13] for each symbol, this will contribute too many models for the whole system. Moreover, symbol detection and text BB detection can be combined into one to save both the time and space, which is under development.

d) System Implementation: In our initial implementation, we build a mobile application that allows users to take a picture of any parking signs and get an immediate interpretation of the parking sign as in Fig. 1. Our application supports both iOS (version 13.0 and above) and Android (version 8.0 and above) platforms. After a user takes a picture, the App will send the image in portrait mode with the original resolution to our back-end server to interpret. The application communicates with the server through REST API. Currently, the system returns a server log including rotation flag, parking sign BB, symbols, text, and re-ordered result for development and improvement.

III. Conclusions

We propose an automatic street parking sign reading system. To the best of our knowledge, this is the very first machine learning and IoT based solution to the parking sign reading problem. We build a new parking sign detection, text detection and recognition, and symbol detection training data. Advanced machine learning techniques are extended to accurately interpret a parking sign on the roadside. Besides, an application is successfully built to take a user's input for testing and improvement, as well as enlarging the parking sign image database. Currently, although some models are small enough to be embedded in small devices like mobile phones, some are still large and we have too many models. Therefore, each image has to be sent to the server for interpretation, which will slow down the procedure and downgrade the user experience. Therefore, we will compress large models, combine some models into one, and integrate the system to other IoT devices, such as dash-cam in the future.

References

[1] J. C. Maxwell, *A treatise on electricity and magnetism*. Oxford: Clarendon Press, 1873, vol. 1.
[2] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.
[3] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *ICCV*, 2017, pp. 2980–2988.
[4] A. Bochkovskiy, C.-Y. Wang, and H. Liao, "Yolov4: Optimal speed and accuracy of object detection," *arXiv*, vol. abs/2004.10934, 2020.
[5] R. Girshick, "Fast R-CNN," in *ICCV*, 2015, pp. 1440–1448.
[6] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *NIPS*, 2015, pp. 91–99.
[7] H. Irshad, Q. Mirsharif, and J. Prendki, "Crowd sourcing based active learning approach for parking sign recognition," *arXiv preprint arXiv:1812.01081*, 2018.
[8] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.
[9] K. Wang, B. Babenko, and S. Belongie, "End-to-end scene text recognition," in *ICCV*, 2011, pp. 1457–1464.
[10] L. Neumann and J. Matas, "Real-time scene text localization and recognition," in *CVPR*, 2012, pp. 3538–3545.
[11] Y. Baek, B. Lee, D. Han, S. Yun, and H. Lee, "Character region awareness for text detection," in *CVPR*, 2019, pp. 9365–9374.
[12] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *TPAMI*, vol. 39, no. 11, pp. 2298–2304, 2017.
[13] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *CVPR*, 2017, pp. 1251–1258.

On the Discovery of Orthogonal Memory Representation for Incremental Learning Enabled Pattern Recognition in IoT

Yongxin Liu, Jian Wang, Shuteng Niu, Houbing Song
Security and Optimization for Networked Globe Laboratory (SONG Lab)
Embry-Riddle Aeronautical University
Daytona Beach, FL 32114 USA
Email: LIUY11,WANGJ14,SHUTENGN@my.erau.edu; h.song@ieee.org

***Abstract*—Deep Learning (DL) has been utilized pervasively in the Internet of Things (IoT). One typical application of DL in IoT is smart decision making using massive data and learning components. However, learning components in smart systems, have to evolve to adapt to operational variations, such a paradigm is termed as Incremental Learning (IL). Conventional IL schemes can not provide satisfying performance when historical data are not available, this phenomenon is termed Catastrophic Forgetting (CF). In this paper, we address the CF problem from a new perspective, minimizing the concept confusion or interference that causes forgetting using Orthogonal Memory Representation (OMR). Firstly, we formally prove that to minimize concept confusion in large-scale classification scenarios, concept representation vectors should be orthogonally distributed. Secondly, we show that the conventional IL schemes can lead to low topological maturity and high concept confusion of DNN models. Finally, we propose a new orthogonal Channel Separation Enabled Incremental Learning (CSIL) scheme and demonstrate its effectiveness using large-scale signal records for NDI in aviation wireless communication systems. Data and code available at IEEE Dataport (DOI: 10.21227/1bxc-ke87) and https://github.com/pcwhy/CSIL.**

I. Introduction

The Internet of Things (IoT) is providing applications and services that would otherwise not be possible [1]–[3]. Intelligent decision making is of great significance in IoT [4]–[7]. A typical way to implement smart decision functionality in IoT is by integrating learning-enabled components through Deep Learning (DL) and Deep Neural Networks (DNNs). One typical application of DNNs in IoT is the non-cryptographic identification of IoT devices through their wireless signals [8]–[10]. However, DNN models in IoT need to continuously evolve as new classes are being added. This scheme is termed as Lifelong or Incremental Learning (IL).

Non-Incremental Learning (Non-IL) requires periodic re-training, which requires substantial memory and long training time. In IoT, zero memory for historical data are preferred for efficient and continuous evolving. The absence of historical data results in *Catastrophic Forgetting (CF)*, in which DNNs' performance degrades significantly after training on new tasks. IL has become an emerging topic in machine learning, however, some works require notorious efforts to train task-related generative models for knowledge replay, while others still require storing data exemplars.

In this paper, we explored the topological properties of latent space in the final classification layer of DNN. We formally prove that, to minimize confusion, the concept representation vectors should be orthogonally distributed. We then invented an enhanced IL scheme, the Channel Separation Enabled Incremental Learning (CSIL), and we introduced orthogonal relationships in latent spaces between concepts in different tasks (learning stages). The proposed framework has been evaluated in massive signal recognition. The most recent advancement of neuroscience [11] also suppports our findings from a biological perspective.

II. Methodology

A. Optimal separation of concepts in the latent space

Our previous work has shown that the last layer of DNNs can be described as a weighted and biased nearest neighbour matching process using cosine similarity [12], and each class or concept can be modeled using a unit vector. Intuitively, if the concepts' representative vectors (fingerprints) are distantly separated, we will have less chance to confuse them. To quantify the separation, we use the sum of the mutual cosine distances of all concepts' fingerprints to quantify the optimality of the separation of concepts. We find that the converging point of the summed mutual cosine distance of all concepts' fingerprints is a predictable constant: $-\frac{C}{2}$, where C is the number of concepts (classes). The complete proof of this theorem is given in [13]. When such a value is reached, the separation of concepts are maximized in the latent space, indicating the lowest degree of conflict or interference. We will use the term *Degree of Conflict (DoC)*. For example, if we have 10 classes, the optimal value of DoC will be $\frac{-10}{2} = 5$ when a DNN model is converged. The training of DNN models also minimizes the DoC as shown in [14].

B. Proof of orthogonality

We define that the averaged cosine distance between N classes is $\overline{D_0}$, then $\overline{D_0}$ can be estimated as: $\overline{D_0} = \frac{-N}{2} / \frac{N(N-1)}{2} = -\frac{1}{N-1}$. If N becomes larger, we will have:

$\overline{D_0} \approx 0$. Therefore, we prove that in a DNN model, the concepts' representation vectors distribute in a mutually orthogonal manner, especially when we have more concepts. An advantage of orthogonal concept organization is that when new concepts are inserted incrementally and orthogonally, there will not be any influence on the DoC. This is the key motivation of our new incremental learning strategy.

C. Channel Separation enabled Incremental Learning (CSIL)

We proposed the Channel Separation Enabled Incremental Learning (CSIL), as depicted in Figure 1. Intuitively, the merits of this approach are: a) we let the concepts learned at different stages to automatically use their task-specific channels in the feature embedding layer. b) We force the concepts learned from different stages to be orthogonally separated. c) CSIL model is trained without historical data and we applied elastic weight consolidation and knowledge distillation to prevent forgetting. A complete demonstration of the model is presented in [14].

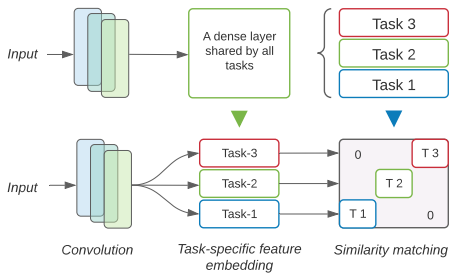

Fig. 1. Channel separation for incremental learning

III. Performance Evaluation

We use real-world aviation communication signal dataset to verify our solution and compare the CSIL algorithm with other IL algorithms that do not require historical data. The overall performance is given in Figure 2, our proposed algorithm CSIL yields the best performance in incremental learning. A comparison of the metric, the Degree of Conflict (DoC), of all devices' fingerprints during incremental learning, is given in Figure 3. The propose method, CSIL, yields the lowest DoC.

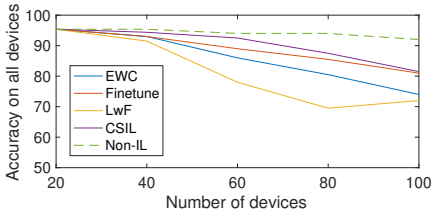

Fig. 2. Comparison of incremental learning strategies for signal identification

IV. Conclusion

In this paper, we provide a new metric, Degree of Conflict (DoC), to measure the degree of confusion within the latent concept representation of DNN models. We formally prove that, to minimize confusion, the concept representation vectors should be orthogonally distributed. We then invented an enhanced IL scheme, the Channel Separation Enabled Incremental Learning (CSIL), based on the orthogonal memory organization. The effectiveness of the proposed framework has been demonstrated in massive signal recognition. We believe the orthogonal memory organization methods can be generalized to other domains, such as virus detection or medical image classification.

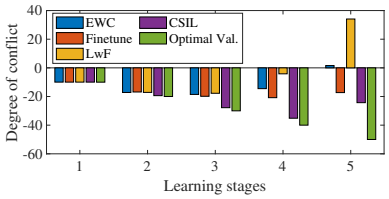

Fig. 3. Comparison of Degree of Conflict among IL algorithms

Acknowledgment

This research was partially supported by the National Science Foundation under Grant No. 1956193.

References

[1] S. Jeschke, C. Brecher, H. Song, and D. Rawat, *Industrial Internet of Things: Cybermanufacturing Systems*. Cham, Switzerland: Springer, 2017.

[2] Y. Sun, H. Song, A. J. Jara, and R. Bie, "Internet of things and big data analytics for smart and connected communities," *IEEE Access*, vol. 4, pp. 766–773, 2016.

[3] R. Taheri, M. Shojafar, M. Alazab, and R. Tafazolli, "Fed-iiot: A robust federated malware detection architecture in industrial iot," *IEEE Transactions on Industrial Informatics*, 2020.

[4] G. Dartmann, H. Song, and A. Schmeink, *Big Data Analytics for Cyber-Physical Systems: Machine Learning for the Internet of Things*. Elsevier, 2019.

[5] H. Song, D. B. Rawat, S. Jeschke, and C. Brecher, *Cyber-physical systems: foundations, principles and applications*. Morgan Kaufmann, 2016.

[6] H. Song, R. Srinivasan, T. Sookoor, and S. Jeschke, *Smart cities: foundations, principles, and applications*. John Wiley & Sons, 2017.

[7] Z. Chu, P. Xiao, M. Shojafar, D. Mi, J. Mao, and W. Hao, "Intelligent reflecting surface assisted mobile edge computing for internet of things," *IEEE Wireless Communications Letters*, 2020.

[8] Y. Liu, J. Wang, J. Li, S. Niu, and H. Song, "Machine learning for the detection and identification of internet of things (iot) devices: A survey," *arXiv preprint arXiv:2101.10181*, 2021.

[9] H. Song, G. Fink, and S. Jeschke, *Security and privacy in cyber-physical systems*. Wiley Online Library, 2017.

[10] I. Butun, P. Österberg, and H. Song, "Security of the internet of things: Vulnerabilities, attacks, and countermeasures," *IEEE Communications Surveys Tutorials*, vol. 22, no. 1, pp. 616–644, 2020.

[11] A. Libby and T. J. Buschman, "Rotational dynamics reduce interference between sensory and memory representations," *Nature Neuroscience*, pp. 1–12, 2021.

[12] Y. Liu, J. Wang, J. Li, H. Song, T. Yang, S. Niu, and Z. Ming, "Zero-bias deep learning for accurate identification of internet of things (iot) devices," *IEEE Internet of Things Journal*, 2020.

[13] Y. Liu, J. Wang, J. Li, S. Niu, and H. Song, "Formal proofs of orthogonality for class-incremental learning for wireless device identification in iot," *Techrxiv Preprint:10.36227/techrxiv.14559765.v1*, 5 2021.

[14] ——, "Class-incremental learning for wireless device identification in iot," *IEEE Internet of Things Journal*, 2021.

IoT Device Energy Consumption Measurement Using A Machine Learning Model

Saroj Kumar Panda
Department of Computer Science
St. Francis Xavier University
Canada
x2019fpn@stfx.ca

Man Lin
Department of Computer Science
St. Francis Xavier University
Canada
mlin@stfx.ca

***Abstract*—Internet of Things (IoT) devices are deployed in the field to collect data from the environment, which are then processed in IoT applications to make intelligent decisions and take appropriate actions. These devices are often powered by batteries, which have a limited lifetime. Energy consumption is one of the most vital issues for these IoT devices. In recent years, the energy consumption problem has continued to receive a lot of attention from the IoT community in applying various techniques to measure and reduce energy consumption while still meeting the computational demand. The question is how to continually monitor the energy consumption of these IoT devices in a cost-effective manner. One cost-effective way is to use low-cost external power meters such as power meter plugs or USB power meters to supply power to the IoT device and read the energy consumption information from the power meter. The readings have to be collected manually, even some of these devices provide Bluetooth applications to display the energy in a separate screen on a mobile or a PC, as these proprietary devices do not expose any API to communicate with them directly. In this work, we propose a framework that integrates a machine learning model to monitor the energy consumption of any IoT device that can be powered through a power meter by pro-grammatically reading the energy information from the Bluetooth application screen.**

***Index Terms*—IoT, Power Meter, Energy Consumption.**

I. Introduction

Internet of Things (IoT) is an emerging domain that provides ordinary tiny physical devices to be connected to the internet, collect, process, and share data that is changing the way people interact with things around them. IoT devices often collect a huge amount of data from the environment that are processed by the IoT applications. These applications may run on the IoT device or on nearby servers (Edge Servers) or on the servers in the cloud. In the process of collecting, processing and offloading data to the Edge or Cloud servers, the IoT devices consume a huge amount of energy.

These IoT devices are often powered by batteries, which have a limited lifetime. Therefore, energy optimization is one of the key research areas for these devices. Many energy optimization solution techniques have been proposed. But, most of these solutions use formula-based energy estimations to validate their results and often only estimate processor energy consumption for their solution. Note that processor energy consumption may differ significantly from system energy consumption of the IoT device, which also includes energy consumption for other components such as memory, sensors, and for running other software components on the system, etc. Hence, to validate the effectiveness of any solution to the energy optimization problem, there is a need to accurately measure and monitor system energy consumption for the whole IoT device cost-effectively.

In this work, we propose a framework that integrates a **computer vision** software component and a **Machine Learning** model with a Power Meter to monitor the energy consumption of any IoT device that can be powered through the Power Meter.

II. System Architecture and Results

The System Architecture is schematically illustrated in Fig. 1. The IoT device under observation is powered through an USB Power meter, UM25C [1], which provides a Windows Bluetooth application to display the instantaneous information sent from the power meter. The Power meter monitors the power drawn by the IoT device, computes the energy consumption every half a second and sends that information to the Bluetooth application, which then displays that in the text field **Energy(mWh)** as shown in the **image** within the ***Smart Energy Meter*** application in Fig. 1. The background application, ***Smart Energy Meter***, in our framework extracts the energy information from the display image to compute the energy consumption.

A. Software Components

The **computer vision** based framework for computing energy consumption consists of two software components namely ***Smart Energy Meter***, and the ***Energy Interface*** as shown in Fig. 1.

1) ***Energy Interface:*** This interface runs on the IoT device as part of the entity/application that needs to measure the device energy consumption. It provides a persistent TCP/IP connection to the Windows machine running the ***Smart Energy Meter*** application and two commands to be sent to that application. To measure the energy for any experiment we will send the ***Start*** command before starting the experiment and the ***Stop*** command at the end of the experiment to receive the energy consumption measurement.

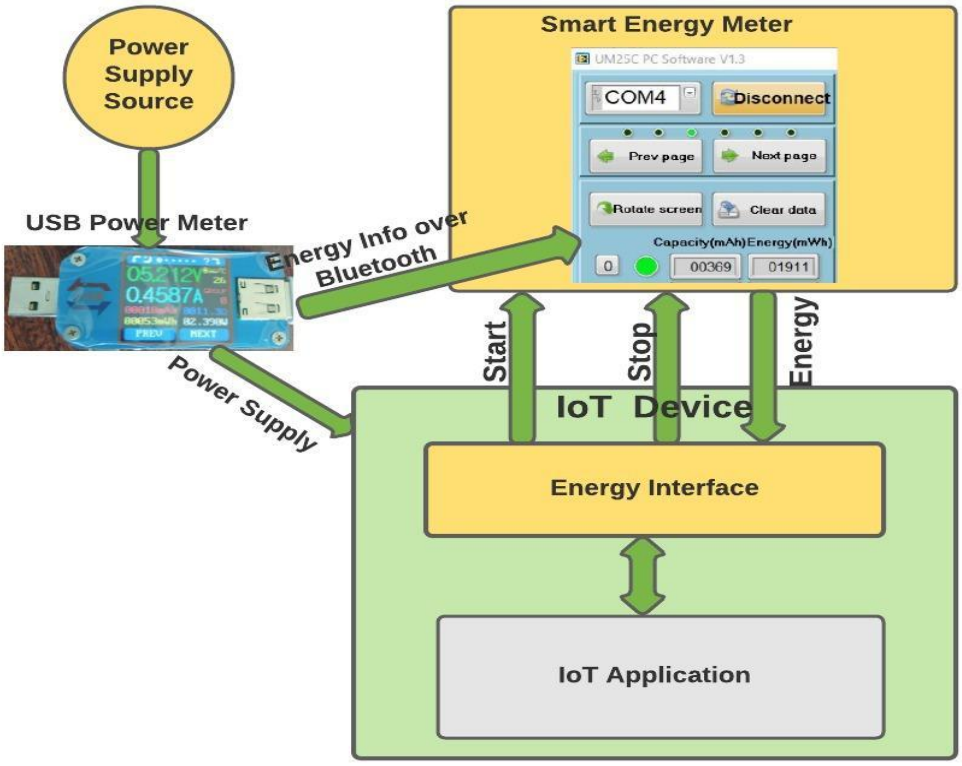

Fig. 1. System Architecture of IoT Device Energy Measurement

2) ***Smart Energy Meter:*** This is the heart of the framework. This background application captures the computer screen that includes the window displaying the energy information from the UM25C power meter upon receiving the commands ***Start*** and ***Stop*** from the IoT device. After the screens are captured, it uses **computer vision** algorithms to process the images in order to locate the energy fields and extract the energy fields from those images as separate images.
The extracted images are then processed through the **computer vision** algorithms to scale up the image size, binarize (black-and-white) the image, and remove noise from them. The processed images are passed to Tesseract [2], an optical character recognition (OCR) engine to extract the digits and hence the energy readings. The difference between the two readings is returned as the energy consumption of the experiment to the IoT device. We use Tesseract 4 with a new OCR engine, which uses a **neural network** system based on **Long short-term memory (LSTM)** models and has higher accuracy than the legacy OCR engine. The new OCR engine has in-built trained **LSTM** models for many languages and also provides options to train a new model from scratch or fine-tune an existing model. For our framework, the trained **LSTM** model for the English language suffices our requirement.

B. Results

We used the framework to measure energy consumption of two devices, Raspberry Pi 4 Model B [3] and Nvidia Jetson Nano 2GB [4] that are used as IoT devices for many IoT applications. We used MNIST Image classification application as the test application and run the experiment multiple times. The energy measurement framework accurately returned the energy consumption for each experiment consistent with the previous results for each of those devices.

III. Conclusion

The experiment results show that the proposed framework can accurately and continually measure energy consumption of any IoT device that can be powered through UM25C USB Power Meter.

References

[1] G. Liu, "Um25c documents and software," https://www.mediafire.com/folder/q2b8h079hpywq/UM25.
[2] Google, "Tesseract, an optical character recognition (ocr) engine," https://opensource.google/projects/tesseract.
[3] R. P. Foundation, "Raspberry pi 4 model b specifications," https://www.raspberrypi.org/products/raspberry-pi-4-model-b/specifications/.
[4] N. Corporation, "Jetson nano 2gb developer kit," https://developer.nvidia.com/embedded/jetson-nano-2gb-developer-kit.

Tradeoff between Complexity and Performance: Machine Learning Driven Resource Allocation in IoT

Li Ping Qian*, Chao Yang*, Yuan Wu‡, and Bin Lin†
*College of Information Engineering, Zhejiang University of Technology, Hangzhou, China
‡State Key Laboratory of Internet of Things for Smart City, University of Macau, Macau, China.
†Department of Communication Engineering, Dalian Maritime University, Dalian, China
Email: lpqian@zjut.edu.cn; cyang_zjut@163.com; yuanwu@um.edu.mo; binlin@dlmu.edu.cn

***Abstract*—The increase in the number of mobile devices and the expansion of smart cities require the Internet of Things (IoT) networks to provide a larger coverage and serve more users, which significantly increases the complexity of communication networks. Machine learning (ML), as a discipline to study how to make machines realize human behavior, provides a promising approach to solve the increasingly complicated IoT problems, and ML has been widely used in various applications, including natural language processing, machine vision, pattern recognition, and etc. In this newsletter, we introduce some of our recent researches on the application of ML in IoT networks, including the unmanned aerial vehicle (UAV) assisted Internet of Vessels (IoVs) systems, energy harvesting (EH) relay aided IoT networks, as well as mobile edge computing (MEC) systems.**

I. Introduction

The Internet of Things (IoT) has been considered as one of the fundamental technologies for enabling future smart cities as well as various smart applications. With the explosive growth in the number of IoT devices, including smart sensors, wearable devices, and mobile terminals, the IoT networks have become more and more complicated. Meanwhile, thanks to the deep penetration of B5G/6G cellular networks in recent years, there have been many emerging paradigms of advanced IoT systems/technologies, e.g., Narrowband IoT, Internet of Vehicles, Satellite-based IoT, unmanned aerial vehicle assisted IoT. The huge number of IoT devices, as well as the aforementioned advanced IoT systems, necessitate careful and comprehensive management and resource allocation, which, however, has imposed significant challenges to conventional optimization-based schemes. Machine Learning (ML), which exploits knowledge discovery/extraction from data to enable machine-type self-learning and strategic optimization, has provided a promising approach for large-scale management and resource allocation for massive IoT systems [1] - [5] (and see a comprehensive overview in [6]).

In the remainder of this newsletter, we will introduce our recent work which exploits DRL for the trajectory optimization in marine IoT networks, resource allocation for energy-harvesting (EH) assisted IoT networks, non-orthogonal multiple access (NOMA) enabled NB-IoT networks, and exploits meta-learning for dynamic offloading optimization in mobile edge computing (MEC) networks.

II. Learning-Driven IoT Network Performance Optimization

ML has provided promising solutions for the efficient management and resource allocation for large-scale IoT networks. In this section, we briefly introduce some of our recent efforts on machine learning for IoT.

A. UAVs' trajectory optimization in NOMA-based maritime IoT networks

The complex and challenging marine environments have imposed difficulties in deploying IoT networks along with their management. To meet the long-distance communication requirements and to deal with the bottleneck of the limited energy resources, we proposed a non-orthogonal multiple access (NOMA) based unmanned aerial vehicle (UAV)-assisted Internet of Vessels (IoVs) system, and the system model is shown in Fig. 1. Vessels offload their computation tasks to the moving UAV via NOMA transmission and complete their tasks locally as well as remotely at the edge-servers co-located with the UAVs synchronously.

To improve the energy efficiency of offloading and computing, we focus on minimizing the total energy consumption by jointly optimizing the IoVs' offloaded workload, computation resource allocation as well as the trajectory of the UAV. Therefore, we propose a two-layered algorithm for solving it efficiently. Specifically, the top-layered algorithm is proposed to solve the problem of optimizing the UAV trajectory based on DRL, and the KKT-based underlying algorithm is proposed to optimize the multi-domain resource allocation problem.

B. Time scheduling and power allocation in EH relay communication networks

The number of mobile devices is increasing at an unimaginable rate with the development of the IoT. Using renewable energy to maintain the operation of the IoT network can effectively reduce greenhouse gas emissions. EH relays not

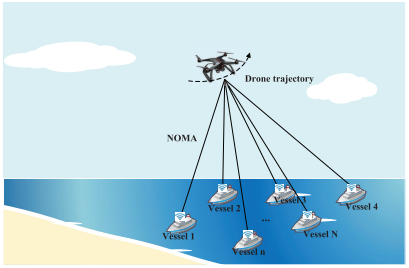

Fig. 1. A NOMA-based UAV-assisted IoV System.

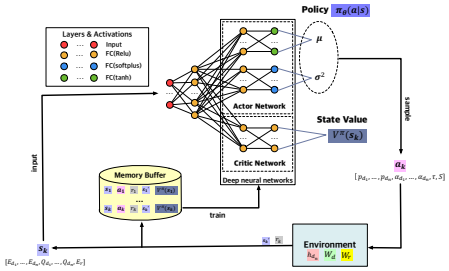

Fig. 2. The framework of the proposed DRL-based algorithm.

only effectively exploits the green energy from natural, but also increases the coverage of the network.

Therefore, we have studied an EH relaying communication network, where a source node transmits its data to the destination node through an EH relay node [7]. Considering the uncertainty of energy arrival and the channel fading, we have designed a time scheduling and power allocation scheme based on DRL to maximize the throughput. While obtaining 90% of the performance of the KKT-based offline algorithm, the DRL-based algorithm has lower computational complexity and faster convergence speed than the offline algorithm.

C. Resource allocation and SIC ordering in NOMA-based EH relay aided NB-IoT networks

As a solution for B5G/6G networks to enable spectrum-efficient access of a large number of mobile users, NOMA has been envisioned as a more effective solution for supporting massive IoT connections than the conventional orthogonal multiple access (OMA). Due to successive interference cancellation (SIC), the decoding order has a great influence on the transmission rate of the devices. Therefore, we consider balancing the transmission rate of the device by optimizing the SIC ordering.

We proposed a NOMA-based EH relay aided network [8] where the devices transmit their data to the EH relay node via NOMA transmission, and the EH relay node forwards the data to the sink node. We take the throughput-based proportional fairness as the optimization objective. For the formulated non-convex optimization problem, we design a DRL-based algorithm to perform SIC ordering and resource allocation, with the detailed framework of the proposed algorithm shown in Fig. 2. Simulation results have demonstrated that the proposed algorithm can obtain 85% of the performance of the offline algorithm with lower computational complexity.

D. Dynamic Computation Task Offloading for Mobile Edge Computing Networks

The past decades have witnessed the growing development of MEC which enables various computation-intensive and latency-sensitive mobile services in future wireless networks. Via MEC, mobile devices can actively offload part of their computation tasks to nearby edge-servers, thereby solving the problem of insufficient computing resources and improving energy efficiency.

To this end, we considered a MEC network that includes dynamic computing tasks in [9]. In particular, taking into account that solving dynamic edge computing tasks requires a lot of training data and time, we have designed a dynamic computing task offloading algorithm based on meta-learning. Compared with other deep learning algorithms, meta-learning requires less training data. Numerical simulations demonstrate that our proposed algorithm can obtain 99% accuracy of the CNN method with only 10 sets of training data.

III. Conclusion and Future Work

ML provides a promising approach for enabling management and resource allocations for massive IoT networks. In this newsletter, we introduced our recent research on applying ML algorithms to perform the optimization with path planning, resource allocation, and computation offloading in IoT networks. In our future work, we are committed to working with ML algorithms to solve distributed resource management for IoT networks via distributed multi-agent deep learning schemes.

References

[1] Z. Li, M. Xu, J. Nie, J. Kang, W. Chen, and S. Xie, "Noma-enabled cooperative computation offloading for blockchain-empowered internet of things: A learning approach," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2364–2378, 2021.

[2] R. Ding, F. Gao, and X. S. Shen, "3D UAV trajectory design and frequency band allocation for energy-efficient and fair communication: A deep reinforcement learning approach," *IEEE Wireless Communication Letter*, vol. 19, no. 12, pp. 7796–7809, 2020.

[3] W. Shi, J. Li, H. Wu, C. Zhou, N. Cheng, and X. Shen, "Drone-cell trajectory planning and resource allocation for highly mobile networks: A hierarchical drl approach," *IEEE Internet of Things Journal*, 2020.

[4] H. Ye, G. Y. Li, and B.-H. Juang, "Power of deep learning for channel estimation and signal detection in ofdm systems," *IEEE Wireless Communication Letter*, vol. 7, no. 1, pp. 114–117, 2018.

[5] Y. Hua, R. Li, Z. Zhao, H. Zhang and X. Chen, "GAN-Based Deep Distributional Reinforcement Learning for Resource Management in Network Slicing," *2019 IEEE Global Communications Conference (GLOBECOM)*, Waikoloa, HI, USA, 2019, pp. 1-6.

[6] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y. Liang, and D. I. Kim, "Applications of deep reinforcement learning in communications and networking: a survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 4, pp. 3133–3174, Fourth Quarter 2019.

[7] L. P. Qian, A. Feng, X. Feng and Y. Wu, "Deep RL-Based Time Scheduling and Power Allocation in EH Relay Communication Networks," *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, Shanghai, China, 2019, pp. 1-7.

[8] L. P. Qian, C. Yang, H. Han, Y. Wu, and L. Meng, "Learning driven resource allocation and sic ordering in eh relay aided NB-IoT networks," *IEEE Communication Letter*, 2021.

[9] L. Huang, L. Zhang, S. Yang, L. P. Qian, and Y. Wu, "Meta-learning based dynamic computation task offloading for mobile edge computing networks," *IEEE Communication Letter*, vol. 25, no. 5, pp. 1568–1572, 2021.

EMG Sensor Based Finger Movement Detection

Mohammadbagher Fotouhi[†*], Ghazaleh Jowkar[†], Tongjue Wang[†], Juhua Hu[†], Payman Arabshahi[*], Wei Cheng[*†]
[†] *Tacoma School of Engineering and Technology*
[*] *Department of Electrical and Computer Engineering*
University of Washington, USA
{mfotouhi, tw66, juhuah, uwcheng}@uw.edu, ghazal.jowkar@gmail.com, paymana@ece.uw.edu

***Abstract*—In this article, we introduce our pilot research on finger movement detection based on the data collected by the Electromyography (EMG) sensor that is placed on the forearm to sense the muscle movements. The EMG sensor is battery powered and connects to the smartphone app through Bluetooth. The size of the EMG sensor is similar to a regular adhesive bandage for minor wound care. As it can be easily covered by sleeve, if we can successfully use them to accurately detect each individual finger movement, there will be many interesting applications such as playing a visual music instrument. We studied a machine learning model for processing the unfiltered EMG signals generated by the muscle movement when we move each individual finger. The current average accuracy of five finger movement detection is about 87%.**

***Index Terms*—Signal Processing, Feature extraction, Dimension reduction, Machine learning.**

I. Introduction

A critical component of most recent human-machine interaction (HCI) devices is Myoelectric control systems, a system that receives the Electromyography (EMG) signal originated from muscle movement. Most of the existing studies are focused on EMG signal based gesture detection [1]. Though using EMG signals for accurately detecting individual finger movements is more challenging than gesture detection, it is a necessary precursor to control a prosthetic hand for the tasks such as typing. The success of finger movement detection can also extend the EMG application to the areas such as virtual music instrument play, secure communication (sending morse code), authentication, and key pairing.

Fig. 1 shows the nano EMG sensor we used to capture the muscle signal when an individual finger is moving and the Android app for data collection. Each individual finger has two movement pattens. One is closed for half a second and open for half a second, the other is closed for one second and open for one second. We target on detecting both which finger is moving and what is the movement patten.

To achieve the goal, we conduct research in two steps: (1) Extract each movement from the raw signal, via which we can detect the movement patten, and (2) Individual finger detection from each extracted movement signal. In the rest of this article, we will introduce these research followed by the evaluation results.

II. Movement Extraction

The EMG signals are acquired from epidermal electronic systems. The collected EMG signals are noisy and distorted.

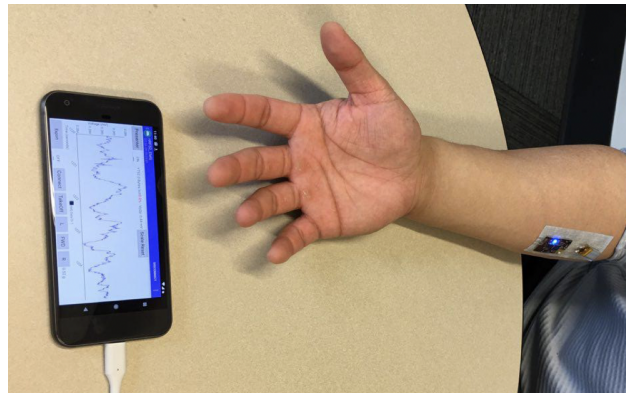

Fig. 1: EMG Sensor and Data Collection

As the signal's quality and the accuracy of extracted movements significantly affect the extracted features, which will be used for individual finger detection in the next step, we first send the signals to a high pass filter to eliminate the noise caused by electrode-skin impedance and continuous body movements. We use Butterworth high pass filter [2] with a corner frequency of 20 Hz [2]. Then, to extract each finger movements from the filtered signals, we designed a finger movement detection system, whose working flow diagram is shown in Fig. 2.

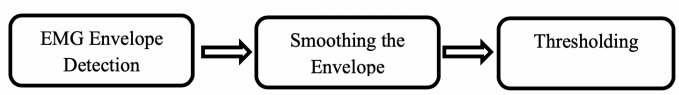

Fig. 2: Movement Extraction Work Flow

The EMG Envelope Detection step returns the upper envelopes of the filtered EMG sequence. The envelope is the magnitude of the analytic signal computed by Hilbert function. We, then, have the signal without any ripple. There are still some sharp variations in the envelope of the EMG signal. It can cause an error in duration calculations. Therefore, the signal envelope is smoothed in the second step. The Smooth function will smooth the data in the column vector by using a moving average filter. In the step of Thresholding, the samples turn to 1 when they are higher than threshold, otherwise they turn to

0. The threshold value is heuristically set by the average value of the EMG signal sequence. By multiplying the thresholded sequence as a duration window, we can access to 'Closed Fingers' and 'Open Hands' samples separately. We can, then, detect the movement patten.

III. Individual finger detection

To detect which finger is moving, we first extract features from each movement signal. These features should contain the most descriptive information about the signal and their size should be reduced in dimension compared to the input signal as a whole. We extracted 17 features from the signal.

We, then, use a machine learning based classifier to identify the individual finger based on the extracted features. Six popular machine learning algorithms are examined for classification including Convolutional Neural Network (CNN) [3], Deep Neural Network (DNN) [3], k-Nearest Neighbor (KNN) [4], Logistic regression (LR) [5], Quadratic Discriminant Analysis (QDA) [6], and XGBoost [7]. We focus on finding out a selection of classification system elements (i.e., feature set, classifier, window characteristics, dimensionality reduction method) for the best performance of individual finger detection.

We performed three rounds of data collection on our testers, giving us three data sets to study and develop a model specified to our testers. These data were first prepossessed for movement extraction then used to train and test a series of classification systems, each of them consists of a different combination of system element choices. Considering the computation power difference of the finger detection devices, we designed a single-layer classifier and a two-layer classifier for less powerful systems and powerful systems, respectively.

1) Single Layer Classifier: By using StratifiedKFold to split the whole data set with a fixed random seed (random state = 1), we ran fivefold cross-validation on each model. We applied Principal Component Analysis (PCA), which is used for dimension reduction, to transform the data set into a new subspace, which makes it more separable. After components are received, we feed the data to the classifiers to do cross-validation. Note that we removed the features that are zeroed out before running PCA. We compared the result of different classifiers with/without using PCA in Table I.

TABLE I: Single Layer Classifier Performance Comparison

5 class classifier	5 fold cross validation total accuracy(%)	
Classifier	Without Using PCA	Using PCA
XGBoost	75.4	77.8
LR	78.03	79.5
CNN	79.6	79.03
DNN	79.9	79.03
QDA	70	79.1
KNN	70.2	69.9

By using PCA, we can combine the information to make them more separable and thus yield better results in classification [8]. Through a series of evaluations, the results and the distribution of the data showed us that the quality and the distribution of the data generally have more impacts on the classification's accuracy than different classifiers.

2) Two-layer classifier: From the confusion matrix of different classifiers and the evaluation results, we observed that the Middle finger has the lowest detection accuracy. To further improve the performance, we designed a two-layer classifier. We have a binary classifier between the Middle finger and the rest of the fingers at the first layer. At the second layer, we have a 4-class classifiers for detecting Thumb, Index, Ring, and Pinkie. TableII shows the average detection accuracy among the three data-set while using different classifiers. We can observe that the best two-layer classifier is DNN with PCA at binary layer and CNN without PCA at 4-class layer. The best average accuracy among three data sets is about 87%

TABLE II: Two-Layer Classifier Performance Comparison

Binary Classifier	5 fold cross validation		4 class classifier	5 fold cross validation	
Classifier	Without Using PCA	Using PCA	Classifier	Without Using PCA	Using PCA
XGBoost	82.8	83.06	XGBoost	84.8	86.23
LR	85.7	85.8	LR	86.7	85.8
CNN	86	84.3	CNN	87.2	86.9
DNN	80.4	87.3	DNN	80.4	85.9
QDA	77.5	75.6	QDA	80.3	83.1
KNN	79.3	79.4	KNN	81.2	80.9

IV. Conclusion and future work

In this article, we briefly introduced our pilot work on detecting finger movement via using nano EMG sensors. We designed an effective system to detect individual finger movement and its movement patten. The current best model is a two-layer classifier, where DNN with PCA at binary layer and CNN without PCA at 4-class layer. Since the classifiers' performance vary along with the change of the quality of received signal, in the future, we will work on improving the quality of the captured signals. Besides, we will focus on using voting models, such as Ensemble, to improve the detection accuracy.

References

[1] W.-H. Yeo, Y.-S. Kim, J. Lee, A. Ameen, L. Shi, M. Li, S. Wang, R. Ma, S. H. Jin, Z. Kang *et al.*, "Multifunctional epidermal electronics printed directly onto the skin," *Advanced Materials*, vol. 25, no. 20, pp. 2773–2778, 2013.

[2] C. Castellini and P. van der Smagt, "Surface emg in advanced hand prosthetics," *Biological cybernetics*, vol. 100, no. 1, pp. 35–47, 2009.

[3] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural networks*, vol. 61, pp. 85–117, 2015.

[4] N. S. Altman, "An introduction to kernel and nearest-neighbor nonparametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175–185, 1992.

[5] A. Y. Ng and M. I. Jordan, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes," in *Advances in neural information processing systems*, 2002, pp. 841–848.

[6] A. Silva and A. Stam, "Discriminant analysis." 1995.

[7] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[8] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

Understanding Human Factors for Machine Learning Algorithm Design over Edge IoT

Guoming Tang* and Kui Wu†
*Peng Cheng Laboratory, Shenzhen, China
†Department of Computer Science, University of Victoria, B.C., Canada

***Abstract*—With the help of AI-powered mobile applications such as Siri and adaptive video streaming, people nowadays are increasingly dependent on mobile phones for daily communication, study, and business. Naturally, understanding human behavior and accommodating it in the machine learning algorithm design are critical to the success of edge IoT systems. We thoroughly investigate the low-battery anxiety (LBA), which is an important human factor but often ignored in algorithm design due to the difficulty of quantification in real-world. Without a better understanding of LBA, it would be difficult to precisely validate energy saving and management techniques in various machine learning applications in terms of alleviating LBA and enhancing users' Quality of Experience (QoE). To fill the gap, we conduct an investigation over $2000+$ mobile users, look into their feelings and reactions towards LBA, and quantify their anxiety degree during the draining of battery power. We also discuss how our LBA model can provide valuable references for the design, evaluation, and improvement of machine learning algorithms in mobile applications and services.**

I. Introduction

The low-battery anxiety (LBA) refers to one's fear of losing mobile phone battery power, especially when it is at a low level (20% for example). According to a survey conducted by LG in 2016, nine out of ten mobile were users more or less affected by the LBA [1]. Based on our survey over $2000+$ mobile users in 2019, the ratio of users suffering from LBA hits 92%.

Evidences have shown that, LBA could cause negative effects on mobile users' emotion, behavior and even health. Those who severely suffer from the LBA were reported to behave strangely, e.g., head home immediately, ask chargers from strangers, secretly "borrow" other's charger, or stop answering incoming calls [1]. Also, the LBA is said to potentially harm our social relationships, and a large proportion of surveyed users got blamed for not speaking to their family members, friends or colleagues due to the low battery. Considering the billions of smartphone users all over the world, the impact of LBA is profound. It is an urgent call to consider human factors, LBA in particular, in the design of intelligent algorithms running on battery-powered IoT devices (e.g., smartphone and smartwatch) as well as on edge access point (e.g., edge server and WiFi access point). Nevertheless, so far there are no quantitative LBA models available for this purpose.

II. Needs to Learn More about LBA

Understanding users' behavior when facing LBA has significant practical implication. For example, for the mobile OS designers, LBA study could provide references and inspirations for more user-friendly human-battery interfaces. More specifically, it is important to incorporate users' low-battery anxiety as a QoE metric when designing the mobile OS. As another important observation, mobile video streaming services are booming recently, but it is challenging for service providers to retain customers. According to our survey, nearly half of the mobile users would give up watching videos when the battery level drops below 10%. Thus, for mobile service providers, LBA may impact the customer retention rate as well as the revenue.

Great efforts have been devoted to saving energy and prolonging the battery lifetime of mobile devices, mainly targeting at the components of computing, communicating, and display. Nevertheless, very few of them take LBA into consideration and how effectively they would perform at alleviating LBA remains largely unanswered.

We are thus motivated to quantitatively investigate the LBA among large-scale mobile users. A quantitative profiling of LBA population's psychology and behavior could result in more efficient and even new solutions to LBA-related problems. For instance, quantitative metrics of LBA can be either leveraged to evaluate the effectiveness of anxiety relieving approaches, or incorporated into user QoE enhancement strategies for better IoT services.

III. LBA Survey and Quantifications

A. A Survey over 2000+ Mobile Users

To investigate the severity of LBA and quantify its impacts on mobile users, we carefully designed a questionnaire [2], including questions on participants, the models of mobile phones, satisfaction of phone batteries, self-assessment of LBA, and so on. By extensively distributing it over a popular mobile social network platform, we were able to collect $2,071$ feedbacks. After the data cleansing, we eventually obtained $2,032$ effective answers.

According to the feedback, it is surprising to see that 92% of the participants suffer from LBA, more or less. This is consistent with the LG survey, but the percentage is even higher. Particularly, over 34% of them firmly admitted their suffering of LBA, 6% of which claimed "severely suffering".

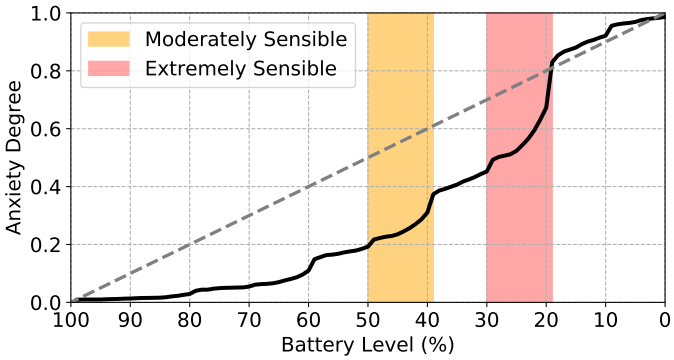

Fig. 1. The LBA curve (the black solid line) obtained from the survey data of $2,032$ mobile users, with sensible regions highlighted in orange and red.

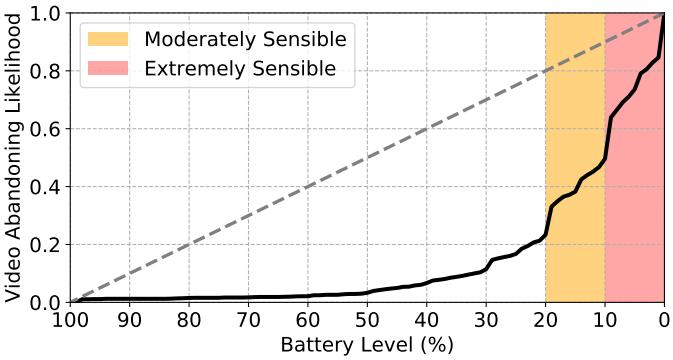

Fig. 2. The VAL curve (the black solid line) extracted from the survey data of $2,032$ mobile users, with sensible regions highlighted in orange and red.

B. Quantifying LBA and Its Impact on Video Watching Behavior

1) LBA Model: To quantify the anxiety degree of mobile users at different battery levels, we set a particular question in the questionnaire: *at what battery level will you charge your mobile phone, when it is possible?* The answer provides us with an angle to infer at which energy level the user begins to worry about the battery life, i.e., experiencing the low-battery anxiety. With all the answers of participants, we are able to extract an empirical *LBA curve* by a reversed cumulative histogram approach (refer to our technique report [2] for more details). The extracted LBA curve is shown in Fig. 1, from which we have the following findings.

- The anxiety degree does not linearly increase with the decrease of battery level. The curve shows that, the mobile user gets more sensible to the battery level as the energy drains.
- Two sensible regions of the users' anxiety can be found, named *moderately sensible* and *extremely sensible* regions as illustrated in Fig. 1, both corresponding to about 10% battery level drop. In the moderately sensible region, the 10% battery level dropping leads to about 20% anxiety degree increases, while in the extremely sensible region the number is about 40%. The occur of the extremely sensible region is most probably due to the battery user interface (e.g., the battery icon's color turns yellow or red) and the low-battery warning message.

2) VAL Model: It is well known that video playing could consume a large portion energy of the mobile system. Thus, we further investigate and quantify the impact of LBA on mobile users' video watching behavior. In the questionnaire, we ask the participants to answer: *at what battery level will you give up watching a video you are interested in, when you are browsing the WeChat Moment or Weibo*[1]*?* The feedback to this question directly shows how the mobile users value the battery power versus attractive videos, thus indicates how the LBA impacts the behavior of video watching. Thus, following the same way to extracting LBA curve, we were able to extract the *video abandoning likelihood (VAL) curve*, as illustrated in Fig. 2. From another perspective, the VAL curve depicts the (approximate) proportion of mobile users, among the whole population, that would give up watching attractive videos at specific battery levels. Based on the curve, we have the following observations.

- The video abandoning likelihood does not linearly increase with the draining of battery power. In contrast, the curve is way below the linear trend (the grey dashed line in the figure), indicating that mobile users generally value attractive videos much more than the battery energy.
- When the battery level is above 20%, the user's video watching behavior seems not much affected by the battery level. Nevertheless, when the battery level drops below 20%, the video abandoning likelihood rises up quickly; when the battery energy is left around 10%, nearly half of users would give up watching videos, no matter how attractive they are.
- Two sensible regions of the video abandoning likelihood curve can be found: *moderately sensible region* for battery level in $[10\%, 20\%]$ and *extremely sensible region* for battery level in $(0, 10\%]$.

IV. Applications of LBA Model in Machine Learning Algorithm Design for IoT Services

1) QoE Enhancement for Mobile Services: QoE measurement and enhancement are critical problems in mobile services. The LBA measurements (given by the LBA curve) could be a useful knob in the QoE-aware design and enhancement of mobile services and applications. Recently, we took advantage of the quantified LBA model for mobile user QoE optimization and designed an intelligent low-power video streaming service solution at the Internet edge [3]. In this work the quantified LBA plays a critical role in reducing mobile users' anxiety and increasing customer retention rate in mobile video streaming.

2) Pricing Policy Making in Crowdsourcing: Thanks to the advanced Internet technology and expanding popularity

[1]The *WeChat Moment* and *Weibo* are currently the two biggest mobile social network platforms, where millions of fresh and popular videos are shared.

of mobile phones, people can easily share information with each other through mobile networks, leading to the burgeoning of crowdsourcing [4]. However, as crowdsourcing consumes mobile battery power, given the same payment, people without LBA would be more willing to participant than those with severe LBA. Accordingly, the pricing strategy should be distinguished for different users by referring to their undergoing LBA.

3) Range Anxiety Quantification of EV Drivers: Electric vehicle (EV) drivers are exposed to the so-called range anxiety, i.e., the fear of being left in the middle of a trip due to battery depletion [5]. As range anxiety of EV and low-battery anxiety of cellphone have some similarity, our methodology to quantify the LBA could be leveraged to quantify range anxiety. The result could then be employed in emotion and behavior analysis of EV drivers.

V. Conclusions Remarks

Through a survey investigation over 2000+ mobile users, we quantified the anxiety degree of mobile users under different battery levels (w.r.t. the LBA curve), as well as the impacts on mobile video watching behavior (w.r.t. the VAL curve). The findings could serve as valuable guidance for algorithm design in IoT services and applications involving human behavior.

References

[1] LG, "'low battery anxiety' grips 9 out of ten people," https://www.lg.com/us/PDF/press-release/LG_Mobile_Low_Battery_Anxiety_Press_Release_FINAL_05_19_2016.pdf, May. 2016.

[2] G. Tang, K. Wu, Y. Wu, H. Liao, D. Guo, and Y. Wang, "Quantifying low-battery anxiety of mobile users and its impacts on video watching behavior," *arXiv preprint arXiv:2004.07662*, 2020.

[3] G. Tang, K. Wu, D. Guo, Y. Wang, and H. Wang, "Alleviating low-battery anxiety of mobile users via low-power video streaming," in *International Conference on Distributed Computing Systems (ICDCS)*. IEEE, 2020, pp. 998–1008.

[4] X. Hu, Z. Ning, K. Zhang, N. Edith, K. Bai, and F. Wang, "Crowdsourcing for mobile networks and iot," *Wireless Communications & Mobile Computing*, vol. 2018, pp. 1–2, 2018.

[5] J. Neubauer and E. Wood, "The impact of range anxiety and home, workplace, and public charging infrastructure on simulated battery electric vehicle lifetime utility," *Journal of power sources*, vol. 257, pp. 12–20, 2014.