Internet of Things Meets Brain-Computer Interface: A Unified Deep Learning Framework for Enabling Human-Thing Cognitive Interactivity

Xiang Zhang, *Student Member, IEEE,* Lina Yao, *Member, IEEE,* Shuai Zhang, *Student Member, IEEE,* Salil S. Kanhere, *Member, IEEE,* Quan Z. Sheng, *Member, IEEE,* and Yunhao Liu, *Fellow, IEEE*

Abstract—A Brain-Computer Interface (BCI) acquires brain signals, analyzes and translates them into commands that are relayed to actuation devices for carrying out desired actions. With the widespread connectivity of everyday devices realized by the advent of the Internet of Things (IoT), BCI can empower individuals to directly control objects such as smart home appliances or assistive robots, directly via their thoughts. However, realization of this vision is faced with a number of challenges, most importantly being the issue of accurately interpreting the intent of the individual from the raw brain signals that are often of low fidelity and subject to noise. Moreover, pre-processing brain signals and the subsequent feature engineering are both time-consuming and highly reliant on human domain expertise. To address the aforementioned issues, in this paper, we propose a unified deep learning based framework that enables effective human-thing cognitive interactivity in order to bridge individuals and IoT objects. We design a reinforcement learning based Selective Attention Mechanism (SAM) to discover the distinctive features from the input brain signals. In addition, we propose a modified Long Short-Term Memory (LSTM) to distinguish the inter-dimensional information forwarded from the SAM. To evaluate the efficiency of the proposed framework, we conduct extensive real-world experiments and demonstrate that our model outperforms a number of competitive state-of-the-art baselines. Two practical real-time human-thing cognitive interaction applications are presented to validate the feasibility of our approach.

Index Terms—Internet of Things, Brain-Computer Interface, deep learning, cognitive.

✦

1 INTRODUCTION

IT is expected that by 2020 over 50 billion devices will be connected to the Internet. The proliferation of the Internet of Things (IoT) is expected to improve efficiency and impact various domains including home automation, manufacturing and industries, transportation and healthcare [1], [2], [3]. Individuals will have the opportunity to interact and control a wide range of everyday objects through various means of interactions including applications running on their smartphone or wearable devices, voice and gestures. Brain-Computer Interface (BCI) [1] is emerging as a novel alternative for supporting interaction bewteen IoT objects and individuals. BCI establishes a direct communication pathway between human brain and an external device thus eliminating the need for typical information delivery methods [4]. Recent trends in BCI research have witnessed the translation of human thinking capabilities into physical actions such as mind-controlled wheelchairs and IoT-enabled appliances [1], [5]. These examples suggest that the BCI is going to be a major aiding technology in human-thing interaction [6] [7].

BCI-based cognitive interactivity offers several advantages.

- Xiang Zhang, Lina Yao, Shuai Zhang, Salil S. Kanhere are with the School of Computer Science and Engineering, University of New South Wales, Kensington, NSW 2052, Australia.
  E-mail: xiang.zhang3, shuai.zhang@student.unsw.edu.au
  {lina.yao, salil.kanhere}@unsw.edu.au
- Quan Z. Sheng is with the Department of Computing, Macquarie University, Sydney, NSW 2109, Australia. E-mail: michael.sheng@mq.edu.au
- Yunhao Liu is with the Department of Computer Science and Engineering, Michigan State University, MI 48824, USA. E-mail: yunhao@cse.msu.edu

Manuscript under review.

1. The BCI mentioned in this paper refers to non-invasive BCI.

One is the inherent privacy arising from the fact that brain activity is invisible and thus impossible to observe and replicate [8]. The other is the convenience and real-time nature of the interaction, since the human only needs to think of the interaction rather than undertake the corresponding physical motions (e.g., speak, type, gesture) [9].

However, the BCI-based human-thing cognitive interactivity faces several challenges. While the brain signals can be measured using a number of technologies such as Electroencephalogram (EEG) [4], Functional Near-Infrared Spectroscopy (fNIR) [10], and Magnetoencephalography (MEG) [11], all of these methods are susceptible to low fidelity and are also easily influenced by environmental factors and sentiment status (e.g., noise, concentration) [12], [13], [14]. In other words, the brain signals generally have very low signal-to-noise ratios, and inherently lack sufficient spatial or temporal resolution and insight on activities of deep brain structures [7]. As a result, while current cognitive recognition systems can achieve about 70-80% accuracy, this is not sufficient to design practical systems. Second, data pre-processing, parameter selection (e.g., filter type, filtering band, segment window, and overlapping), and feature engineering (e.g., feature selection and extraction both in the time domain and frequency domain) are all time-consuming and highly dependent on human expertise in the domain [15].

To address the aforementioned issues, in this paper, we propose a unified Deep Learning (DL) [16] framework for enabling human-thing cognitive interactivity. As shown in Figure 1, our framework measures the user's brain activity (such as EEG, FNIRS, and MEG) through a specific brain signal collection equipment. The raw brain signals are forwarded to the cloud server
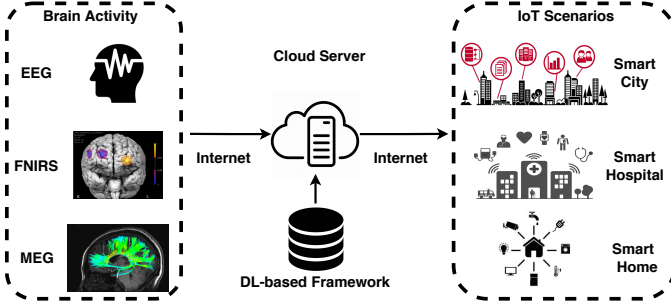
Fig. 1: Schematic diagram of cognitive IoT framework

via Internet access. The cloud server uses a person-dependent pre-trained deep learning model for analyzing the raw signals. The analysis results interpreted signals could be used for actuating functions in a wide range of IoT applicants such as smart city [17] (e.g., transportation control, agenda schedule), smart hospital [18] (e.g., emergency call, anomaly mentoring), and smart home [19], [20] (e.g., appliances control, assistive robot control).

The proposed unified deep learning framework aims to interpret the subjects' intent and decode it into the corresponding commands which are discernible for the IoT devices. Based on our previous study [7], for each single brain signal sample, the self-similarity is always higher than the cross-similarity, which means that the intra-intent cohesion of the samples is stronger than the inter-intent cohesion. In this paper, we propose a weighted average spatial Long Short-Term Memory (WAS-LSTM) to exploit the latent correlation between signal dimensions. The proposed end-to-end framework is capable of modeling high-level, robust and salient feature representations hidden in the raw human brain signal streams and capturing complex relationships within data. The main contributions of this paper are highlighted as follows:

- We propose a unified deep learning based framework to interpret individuals' brain activity for enabling human-thing cognitive interactivity. To our best knowledge, we are the very first work that bridging BCI and IoT to investigate end-to-end cognitive brain-to-thing interaction.
- We apply deep reinforcement learning, with designed reward, state, and action model, to automatically discover the most distinguishable features from the input brain signals. The discovered features are forwarded to a modified deep learning structure, in particular, the proposed WAS-LSTM, to capture the cross-dimensional dependency in order to recognize user's intention.
- We also present two operational prototypes of the proposed framework: a brain typing system and a cognitive controlled smart home service robot, which demonstrate the efficacy and practicality of our approach.

2 THE PROPOSED FRAMEWORK

In this section, we present the cognition detection framework in detail. The subjects' brain activity can be measured by a number of methods like EEG, fMRI, MEG. In this paper, we exploit EEG due to its unique features such as low-cost, low-energy, privacy, and portability. The proposed framework is depicted in Figure 2. The main focus of the approach is to exploit the latent dependency among different signal dimensions. To this end, the proposed framework contains several components: 1) replicate and shuffle processing; 2) the selective attention learning; 3) the

sequential LSTM-based classification. In the following, we will first discuss the motivations of the proposed method and then introduce the aforementioned components in details.

2.1 Motivation

How to exploit the latent relationship between EEG signal dimensions is the main focus of the proposed approach. The signals belonging to different cognitions are supposed to have different inter-dimension dependent relationships which contain rich and discriminative information. This information is critical to improve the distinctive signal pattern discovery.

In practice, the EEG signal is often arranged as 1-D vector, the signal is less informative for the limited and fixed element arrangement. The elements order and the number of elements in each signal vector can affect the element dependency. For example, the inter-dimension dependency in $\{0,1,2,3,4\}$ and $\{1,2,3,4,0\}$ are not reciprocal; similarly, $\{0,1,2,3,4\}$ and $\{0,1,1,2,3,4\}$ are not reciprocal. In many real-world scenarios, the EEG data are concatenated following the distribution of biomedical EEG channels. Unfortunately, the practical channel sequence, with the fixed order and number, may not be suitable for inter-dimension dependency analysis. Therefore, we propose the following three techniques to amend the drawback.

First, we replicate and shuffle the input EEG signal vector on dimension-wise in order to provide as much latent dependency as possible among feature dimensions (Section 2.2).

Second, we introduce a focal zone as a Selective Attention Mechanism (SAM), where the optimal inter-dimension dependency for each sample only depends on a small subset of features. Here, the focal zone is optimized by deep reinforcement learning which has been shown to achieve both good performance and stability in policy learning (Section 2.3).

Third, we propose the WAS-LSTM classifier by extracting the distinctive inter-dimension dependency (Section 2.4).

2.2 Data Replicate and Shuffle

Suppose the input EEG data can be denoted by $\mathbf{X} = \{(\mathbf{x}_i, y_i), i = 1, 2, \cdots I\}$ where $(\mathbf{x}_i, y_i)$ denotes the 1-D EEG signal, called one *sample* in this paper, and I denotes the number of samples. In each sample, the feature $\mathbf{x}_i \in \mathbb{R}^K$ contains K elements and the corresponding ground truth $y_i \in \mathbb{R}$ is an integer that denotes the sample's category. Different categories correspond to various brain activities. $\mathbf{x}_i$ can be described as a vector with K elements, $\mathbf{x}_i = \{x_{ik}, k = 1, 2, \cdots, K\}$.

To provide more potential inter-dimension spatial dependencies, we propose a method called *Replicate and Shuffle* (RS). RS is a two-step feature transformation method which maps $\mathbf{x}_i$ to a higher dimensional space $\mathbf{x}'_i$ with more complete element combinations:

$$\mathbf{x}_i \in \mathbb{R}^K \to \mathbf{x}'_i \in \mathbb{R}^{K'}, K' > K$$

In the first step (Replicate), we replicate $\mathbf{x}_i$ for $h = K'\%K + 1$ times where $\%$ denotes remainder operation. Then we get a new vector with length as $h * K$ which is not less than K'; in the second step (Shuffle), we randomly shuffle the replicated vector in the first step and intercept the first K' element to generate $\mathbf{x}'_i$. Theoretically, compared to $\mathbf{x}_i$, $\mathbf{x}'_i$ contains more diverse and complete inter-dimension dependencies.
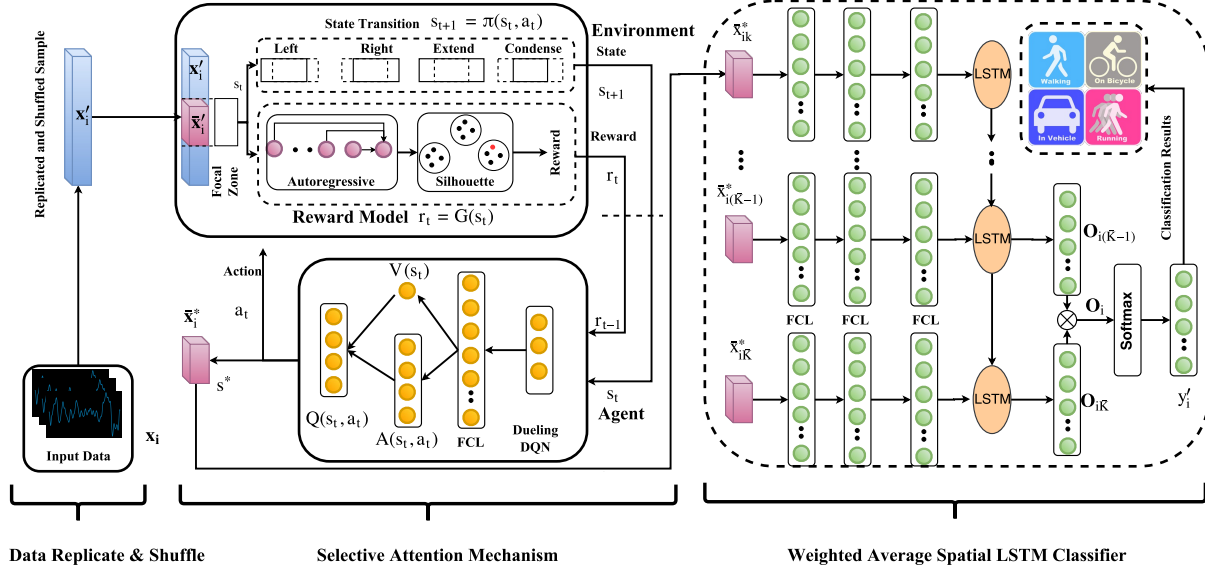
Fig. 2: Flowchart of the proposed framework. The focal zone $\bar{\mathbf{x}}_i$ is a selected fragment from $\mathbf{x}'_i$ to feed in the state transition and the reward model. In each step t, one action is selected by the state transition to update s_t based on the agent's feedback. The reward model evaluates the quality of the focal zone to the reward r_t. The dueling DQN is employed to find the optimal focal zone $\bar{\mathbf{x}}^*_i$ which will be feed into the LSTM based classifier to explore the inter-dimension dependency and predict the sample's label y'_i. FCL denotes Fully Connected Layer. The State Transition contains four actions: left shifting, right shifting, extend, and condense. The dashed line indicates the focal zone before the action while the solid line indicates the position of the focal zone after the action.

2.3 Selective Attention Mechanism

In the next process, we attempt to find the optimal dependency which includes the most distinctive information. But K', the length of $\mathbf{x}'_i$, is too large and is computationally expensive. To balance the length and the information content, we introduce the *attention mechanism* [21] to emphasize the informative fragment in $\mathbf{x}'_i$ and denote the fragment by $\bar{\mathbf{x}}_i$, which is called *focal zone*. Suppose $\bar{\mathbf{x}}_i \in \mathbb{R}^{\bar{K}}$ and $\bar{K}$ denotes the length of the focal zone. For simplicity, we continue to denote the k-th element by $\bar{\mathbf{x}}_{ik}$ in the focal zone. To optimize the focal zone, we employ deep reinforcement learning as the optimization framework for its excellent performance in policy optimization [22].

Overview. As shown in Figure 2, the focal zone optimization includes two key components: the *environment* (including state transition and reward model), and the *agent*. Three elements (the state s, the action a, and the reward r) are exchanged in the interaction between the environment and the agent. In the following we elaborate these three elements which are crucial to our proposed deep reinforcement learning model:

- The **state** $\mathcal{S} = \{s_t, t = 0, 1, \cdots, T\} \in \mathbb{R}^2$ describes the position of the focal zone, where t denotes the time stamp. In the training, s_0 is initialized as $s_0 = [(K' - \bar{K})/2, (K' + \bar{K})/2]$. Since the focal zone is a shifting fragment on 1-D $\mathbf{x}'_i$, we design two parameters to define the state: $s_t = \{start^t_{idx}, end^t_{idx}\}$, where $start^t_{idx}$ and end^t_{idx} separately denote the start index and the end index of the focal zone[2].
- The **action** $\mathcal{A} = \{a_t, t = 0, 1, \cdots, T\} \in \mathbb{R}^4$ describes which the agent could choose to act on the environment. In our case, we define 4 categories of actions for the focal

----

2. E.g., for a random $\mathbf{x}'_i = [3, 5, 8, 9, 2, 1, 6, 0]$, the state $\{start^t_{idx} = 2, end^t_{idx} = 5\}$ is sufficient to determine the focal zone as $[8, 9, 2, 1]$.

zone (as described in the **State Transition** part in Figure 2): left shifting, right shifting, extend, and condense. Here at time stamp t, the state transition only choose one action to implement following the agent's policy π: $s_{t+1} = \pi(s_t, a_t)$.

- The **reward** $\mathcal{R} = \{r_t, t = 0, 1, \cdots, T\} \in \mathbb{R}$ is calculated by the reward model, which will be detailed later. The reward model Φ: $r_t = \Phi(s_t)$ receives the current state and returns an evaluation as the reward.
- We employ the Dueling DQN (Deep Q Networks [23]) as the optimization **policy** $\pi(s_t, a_t)$, which is enabled to learn the state-value function efficiently. Dueling DQN learns the Q value $V(s_t)$ and the advantage function $A(s_t, a_t)$ and combines them: $Q(s_t, a_t) \leftarrow V(s_t), A(s_t, a_t)$. The primary reason we employ a dueling DQN to optimize the focal zone is that it updates all the four Q values at every step while other policy only updates one Q value at each step.

Reward Model. Next, we introduce the design of the reward model, which is one important contribution of this paper. The purpose of the reward model is to evaluate how the current state impacts our final target which refers to the classification performance in our case. Intuitively, the state which can lead to the better classification performance should have a higher reward: $r_t = \mathcal{F}(s_t)$. As a result, in the standard reinforcement learning framework, the original reward model regards the classification accuracy as the reward. $\mathcal{F}$ refers to the WAS-LSTM. Note, WAS-LSTM focuses on the spatial dependency between different dimensions at the same time-point while the normal LSTM focuses on the temporal dependency between a sequence of samples collected at different time-points. However, WAS-LSTM requires considerable training time, which will dramatically increase the optimization time of the whole algorithm. In this section, we propose an alternative method to calculate the reward: construct a

new reward function $r_t = \mathcal{G}(s_t)$ which is positively related with $r_t = \mathcal{F}(s_t)$. Therefore, we can employ $\mathcal{G}$ to replace $\mathcal{F}$. Then, the task is changed to construct a suitable $\mathcal{G}$ which can evaluate the inter-dimension dependency in the current state s_t and feed-back the corresponding reward r_t. We propose an alternative $\mathcal{G}$ composed by three components: the autoregressive model [24] to exploit the inter-dimension dependency in $\mathbf{x}'_i$, the Silhouette Score [25] to evaluate the similarity of the autoregressive coefficients, and the reward function based on the silhouette score.

The autoregressive model [24] receives the focal zone $\bar{\mathbf{x}}_i$ and specifies that how the last variable depends on its own previous values. Then, to evaluate how rich information is taken in the autoregressive coefficients, we employ silhouette score [26] to interpret the consistence of φ. The higher score ss_t indicates the focal zone in state s_t contains more inter-dimension dependency, which means $\bar{\mathbf{x}}_i$ is easier to be classified by the classifier in the next stage. At last, based on the $ss_t \in [-1, 1]$, we design a **reward function**:

$$r_t = \frac{e^{ss_t+1}}{e^2 - 1} - \beta \frac{\bar{K}}{K'}$$

The function contains two parts, the first part is a normalized exponential function with the exponent $ss_t + 1 \in [0, 1]$, which encourages the reinforcement learning algorithm to search the better s_t w that leads to a higher ss_t. The motivation of the exponential function is that: the reward growth rate is increasing with the silhouette score's increase[3]. The second part is a penalty factor for the focal zone length to keep the bar shorter and the β is the penalty coefficient.

In summary, the aim of focal zone optimization is to learn the optimal focal zone $\bar{\mathbf{x}}_i^*$ which can lead to the maximum reward. The optimization totally iterates $N = n_e * n_s$ times where n_e and n_s separately denote the number of episodes and steps [23]. ε-greedy method [27] is employed in the state transition.

2.4 Weighted Average Spatial LSTM Classifier

In this section, we propose Weighted Average Spatial LSTM classification for two purposes. The first attempt is to capture the cross-relationship among feature dimensions in the optimized focal zone $\bar{\mathbf{x}}_i^*$. The LSTM-based classifier is widely used for its excellent sequential information extraction ability which is approved in several research areas such as natural language processing [28], [29]. Compared to other commonly employed spatial feature extraction methods, such as Convolutional Neural Networks, LSTM less dependent on the hyper-parameters setting. However, the traditional LSTM focuses on the temporal dependency among a sequence of samples. Technically, the input data of traditional LSTM is 3-D tensor shaped as $[n_b, n_t, \bar{K}]$ where n_b and n_s denote the batch size and the number of temporal samples, separately. WAS-LSTM aims to capture the dependency among various dimensions at one temporal point, therefore, we set $n_t = 1$ and transpose the input data as: $[n_b, n_t, \bar{K}] \rightarrow [n_b, \bar{K}, n_t]$.

The second advantage of WAS-LSTM is that it could stabilize the performance of LSTM via moving average method [30]. Specifically, we calculate the LSTM outputs $\mathbf{O}_i$ by averaging the past two outputs instead of only the final one (Figure 2):

$$\mathbf{O}_i = (\mathbf{O}_{i(\bar{K}-1)} + \mathbf{O}_{i\bar{K}})/2$$

---

3. For example, for the same silhouette score increment 0.1, $ss_t : 0.9 \rightarrow 1.0$ can earn higher reward increment than $ss_t : 0.1 \rightarrow 0.2$.
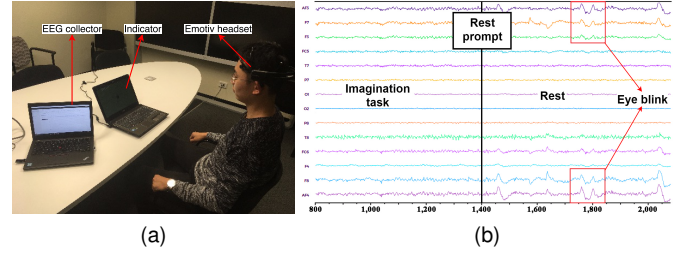

Fig. 3: EEG data collection: (a) collection scenario and (b) the gathered EEG signal

TABLE 1: IMAGERY ACTION, LABEL, AND CORRESPONDING COMMANDS IN CASE STUDIES

Imagery Action	Label	Typing Commands	Robot Commands
Upward	0	Up	Forward
Downward	1	Cancel	Turn Left
Leftward	2	Left	Grasp
Rightward	3	Right	Loose
Middle Cycle	4	Nothing	Nothing
Eye-closed	5	Confirm	Stop/Start

The predicted label is calculated by $y'_i = \mathcal{L}(\bar{\mathbf{x}}_i^*)$ where $\mathcal{L}$ denotes the LSTM algorithm. ℓ_2-norm (with parameter λ) is adopted as regularization to prevent overfitting. The sigmoid activation function is used on hidden layers. The loss function is cross-entropy and is optimized by the AdamOptimizer algorithm [31].

3 EXPERIMENTS

In this section, we design local real-world experiments to evaluate the efficiency and effectiveness of the proposed framework. First, the experimental setting is reported. Then, we compare our model with competitive state-of-the-art baselines and evaluate the performance in detail. Finally, we investigate the impact of crucial factors such as the framework latency and the reward model.

3.1 Experimental Setting

We conduct the EEG collection by using a portable and easy-to-use commercialized Emotiv Epoc+ headset. The headset contains 14 channels and the sampling rate is 128 Hz. The local dataset can be accessed from this link[4]. This experiment is carried out using 7 subjects (4 males and 3 females) aged from 23 to 26. During the experiment, the subject wearing the *Emotiv Epoc+*[5] EEG collection headset, faces the computer screen and focuses on the corresponding *hint* which appears on the screen (shown in Figure 3a). EEG signals are recorded when the subject is imaging certain actions (without any physical action). The certain actions contains: upward arrow, downward arrow, leftward arrow, rightward arrow, and a cycle. Beyond that, the EEG signals that the subject stays relaxation with eye closed are also recorded. In total, there are 6 categories of EEG signals. The imagery action associated with brain activities and the corresponding labels used in this paper are listed in Table 1. In summary, this experiment contains 241,920 samples with 34,560 samples for each subject. For each participant, the dataset is divided into a training set and

---

4. https://drive.google.com/open?id=0B9MuJb6Xx2PIM0otakxuVHpkWkk
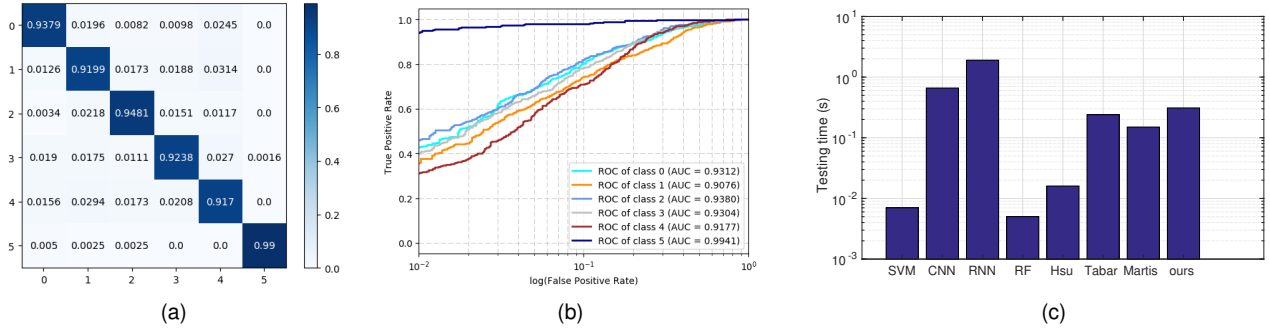5. https://www.emotiv.com/product/emotiv-epoc-14-channel-mobile-eeg/

Fig. 4: Recognition results: (a) confusion matrix, (b) ROC curves with AUC scores, and (c) latency

a testing set. The training set contains 31,104 samples and the testing set contains 3,456 samples. The classification results are evaluated by a number of metrics including accuracy, precision, recall, F-1 score, confusion matrix, ROC (Receiver Operating Characteristic) curve, AUC (Area Under Curve) score.

3.2 Overall Comparison and Analysis

In the training stage, based on the tuning experience, the hyper-parameters setting are listed as follows. In the selective attention learning: the order of autoregressive is 3; $\bar{K} = 128$, the Dueling DQN has 4 layers and the node number in each layer are: 2 (input layer), 32 (FCL), 4 ($A(s_t, a_t)$) + 1 ($V(s_t)$), 4 (output). The decay parameter $\gamma = 0.8$, $n_e = n_s = 50$, $N = 2,500$, $\epsilon = 0.2$, learning rate= 0.01, memory size = 2000, length penalty coefficient $\beta = 0.1$, and the minimum length of focal zone is set as 10. In the deep learning classifier: the node number in the input layer equals to the number of feature dimensions, three hidden layers with 164 nodes, two layers of LSTM cells (164 cells) and one output layer (6 nodes). The learning rate = 0.001, ℓ_2-norm coefficient $\lambda = 0.001$, forget bias = 0.3, batch size = 9, and iterate for 1000 iterations.

To demonstrate the efficiency of our approach, we compare our model with several competitive state-of-the-art methods:

- **Hsu** [32] extracts several potential features, including amplitude modulation, spectral power and asymmetry ratio, adaptive autoregressive model, and wavelet fuzzy approximate entropy (wfApEn), followed by a SVM classifier, to classify the binary motor imagery EEG signals.
- **Tabar et al.** [33] combine convolutional neural networks (CNN) and stacked Autoencoder (SAE) to automatically classify EEG data.
- **Martis et al.** [34] artificially extract several nonlinear features on different EEG frequency bands (including delta, theta, lower alpha, upper alpha, lower beta, upper beta and lower gamma) and forward to SVM with radial basis function kernel.

Table 2 shows the overall comparison between our approach with non-DL baselines, DL baselines, and the state-of-the-art models. RF denotes Random Forest, AdaB denotes Adaptive Boosting, LDA denotes Linear Discriminant Analysis. In addition, the key parameters of the baselines are listed here: Linear SVM ($C = 1$), RF ($n = 200$), KNN ($k = 3$). In LSTM, $n_{steps} = 5$, another set is the same as the WAS-LSTM classifier, along with the GRU (Gated Recurrent Unit [35]). The CNN contains 2 stacked

TABLE 2: OVERALL COMPARISON WITH THE STATE-OF-THE-ART BASELINES. DL DENOTES DEEP LEARNING.

Baselines	Methods	Metrics			
		Acc	Pre	Rec	F1-score
Non-DL	SVM	0.2569	0.2737	0.2569	0.2577
	RF	0.8041	0.8071	0.8041	0.8048
	KNN	0.8539	0.8563	0.8539	0.8544
	AB	0.2506	0.2039	0.2506	0.1557
	LDA	0.2595	0.2761	0.2595	0.2618
DL	LSTM	0.2609	0.2447	0.2348	0.2354
	GRU	0.2521	0.271	0.2696	0.2701
	CNN	0.725	0.724	0.7237	0.7238
The state-of-the-art	[32]	0.8965	0.9011	0.8926	0.8968
	[33]	0.7894	0.7938	0.8013	0.7975
	[34]	0.8891	0.8932	0.8765	0.8848
	WAS-LSTM	0.9026	0.9125	0.9003	0.9064
	Ours	**0.9363**	**0.9394**	**0.9398**	**0.9396**

convolutional layers (both with stride [1, 1], patch [2, 2], zero-padding, and the depth are 4 and 8, separately.), one pooling layer (stride [1, 2], zero-padding), and one fully connected layer (164 nodes). Relu activation function is employed in the CNN.

The observations in Table 2 show that our approach outperforms all the baselines by achieving the highest accuracy of **0.9363** on the 6-class classification. In addition, our model (SAM +WAS-LSTM) performs better than the solo WAS-LSTM, which demonstrates that the selective attention mechanism has a positive contribution to the classification. The confusion matrix, ROC curves, and AUC scores of the proposed framework are reported in Figure 4. We can observe that the last class, representing the eye-closed state, obtains the best performance compared to other 5 classes. This demonstrates that the eye-closed state is the easiest to be recognized, which is reasonable while all the other classes are in eye-open state and are easier to be interrupted by the environmental factors.

3.3 Impact of Key Factor

3.3.1 Latency

To design effective and real-world cognitive interactive applications, both the accuracy and latency of intent recognition are equally important. Subsequently, we compare the latency of the proposed framework with several typical state-of-the-art algorithms and the results are presented in Figure 4c. It is observed that our approach has competitive latency compared with other methods. The overall latency is less than 1 second. The deep learning based techniques in this work do not explicitly lead to extra latency.
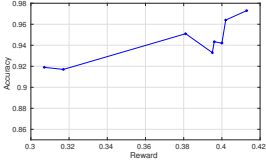
Fig. 5: The relationship between the classifier accuracy and the reward. The correlationship coefficient is 0.8258 while the p-value is 0.0115.
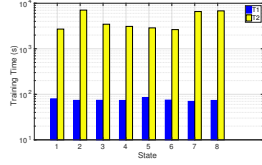
Fig. 6: Reward model training time in various states. T1 and T2 separately denote the training time in reward model $\mathcal{G}$ and $\mathcal{F}$.

3.3.2 Reward Model

Furthermore, we conduct extensive experiments to demonstrate the efficiency of the proposed reward model $\mathcal{G}$. First, we measure a batch of data pairs of the reward (represents the reward of $\mathcal{G}$) and the WAS-LSTM classifier accuracy (represents the reward of $\mathcal{F}$). The relationship between the reward and the accuracy is shown in Figure 5. The figure illustrates that the accuracy has an approximately linear relationship with the reward. The correlations coefficient is 0.8258 (with p-value as 0.0115), which shows that the accuracy and reward are highly positive related. As a result, we can estimate $\arg\max_{\bar{\mathbf{x}}^*} \mathcal{F}$ by $\arg\max_{\bar{\mathbf{x}}^*} \mathcal{G}$. Moreover, another experiment is carried out to measure the single step training time of two reward models $\mathcal{G}$ and $\mathcal{F}$. The training times are marked as T1 and T2, respectively. Figure 6 qualitatively shows that T2 is much higher than T1 (8 states represent 8 different focal zones). Quantitatively, the sum of T1 over 8 states is $35,237.41$ seconds while the sum of T2 is 601.58 seconds. These results demonstrate that the proposed approach, designing a $\mathcal{G}$ to approximate and estimate the $\mathcal{F}$, saves $\mathbf{98.3\%} = 1 - 601.58/35237.41$ training time in focal zone optimization.

4 CASE STUDY

Inspired by the high accuracy and low latency of our proposed framework for human intent recognition, we proceed to develop two real-world cognitive IoT prototypes, namely, (1) a brain typing system (2) mind-controlled assistive robot for the smart home.

4.1 Brain Typing System

Due to the high intent recognition accuracy, we develop an online brain typing system to convert user's thoughts to texts. The video demo clip can be found at the given link[6]. The brain typing system (Figure 7a) consists of two components: the *pre-trained deep learning model* and the *online BCI system*. The pre-trained deep learning model, which is trained offline, aims to accurately recognize the user's typing intent in real time. The online system contains 5 components: the EEG headset, the client 1 (data collector), the server, the client 2 (typing command receiver), and the typing interface. The user wears the Emotiv EPOC+ headset which collects EEG signals and sends the data to client 1 through a Bluetooth connection. The raw EEG signals are transported to the server through a TCP connection.

Specifically, the typing interface (up right corner in Figure 7a) can be divided into three levels: the initial interface, the sub-interface, and the bottom interface. All the interfaces have similar structure: three *character blocks* (separately distributed in left, up,

---

6. https://youtu.be/Dc0StUPq61k

and down directions), a *display block*, and a *cancel button*. The display block shows the typed output and the cancel button is used to cancel the last operation. The typing system in total includes $27 = 3*9$ characters (26 English alphabets and the space bar) and all of them are separated into 3 character blocks (each block contains 9 characters) in the initial interface. Overall, there are 3 alternative selections and each selection will lead to a specific sub-interface which contains 9 characters. Again, the $9 = 3*3$ characters are divided into 3 character blocks and each of them is connected to a bottom interface. In the bottom interface, each block represents only one character.

In the brain typing system, there are 5 commands to control the interface: 'left', 'up', 'right', 'cancel', and 'confirm'. Each command corresponds to a specific motor imagery EEG category (as shown in Table 1). Since the user can hardly concentrate for a long time (usually, less than 10 seconds), the brain activity may represent none of the valid commands sometimes. Nevertheless, the proposed deep learning framework cannot distinguish the invalid brain activity, we leave one specific brain category to represent the invalid signal. If the individual's brain signal is not in any of the 5 valid categories, it is classified as the invalid category and the brain typing system will do nothing under this situation[7]. Moreover, based on the experiments results in Section 3.2, the eye-closed state has the highest precision and accuracy, therefore, we select this state as the confirmation command for the reason that 'confirmation' is the most crucial command in typing system. To type every single character, the interface is supposed to accept 6 commands. Consider typing the letter 'I' as an example. The sequence of commands to be entered is as follows: 'left' (choose the left block with characters $A \sim I$), 'confirm', 'right' (choose the right block with characters $G \sim I$), 'confirm', 'right' (choose the right block with characters I), 'confirm'.

In our practical deployment, the sampling rate of Emotiv EPOC+ headset is set as 128Hz, which means the server can receive 128 EEG recordings each second. Since the brainwave signal varies rapidly and is very easy to be affected by noises, the EEG data stream is sent to server *every half second*, which means that the server receives 64 EEG samples each time. The 64 EEG samples are classified by the deep learning framework and generate 64 categories of intents. we calculate the mode of 64 intents and regard the mode as the final intent decision. Furthermore, to achieve steadiness and reliability, the server sends the command to client 2 only if *three consecutive decisions remain consistent*. After the command is sent, the command list will be reset and the system will wait until the next three consistent decisions are made.

4.2 Cognitive Robot

Another important application for BCI-inspired Internet of Things is extending the orientation of smart homes by integrating the subject's intent and the real-world IoT objects to effectively control things of interest (TOIs).

To demonstrate the feasibility of the proposed framework, we report the second use case as implementing cognitive interactivity in an IoT-based smart home system. The IoT-based smart home is equipped with sensors, wherein IR sensors, ambient sound, heat, as well as contact sensors are mounted on furniture and used in the home environment in a non-intrusive manner. In our

---

7. Similarly, in the cognitive robot case, the robot will remain the previous state under the invalid command.
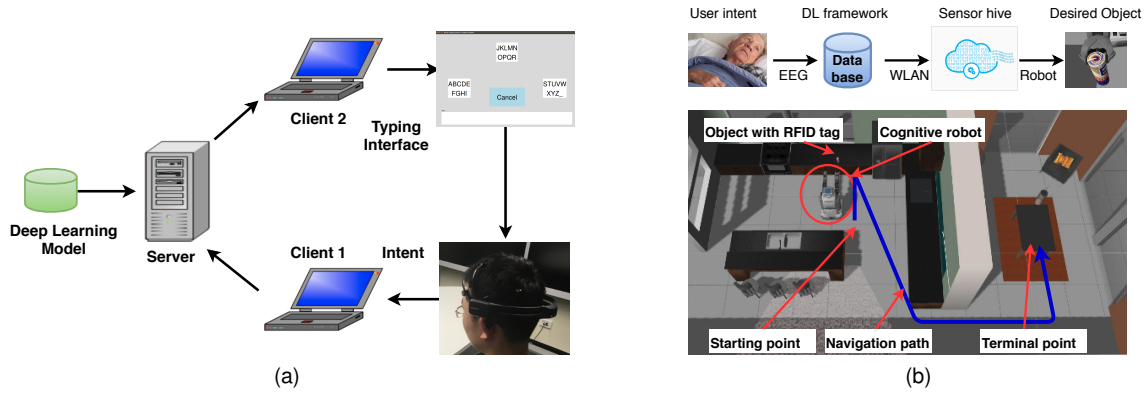
Fig. 7: Human-thing cognitive applications: (a) brain typing system and (b) cognitive robot in IoT scenario and WLAN denotes Wireless Local Area Network

case, within the smart home environment which is perceived by the embedded sensor-networks, a simulated robot is cognitively navigated to perform a routine task. In the specific scenario, the robot, learns user's intent from EEG recordings via the proposed framework, to take the IoT object (e.g., a can of beverage) from a table in the kitchen and put it in a table in the living room. The desired object is aggregate with RFID tag which helps to identify the location. The IoT scenario is depicted in Figure 7b and the demo can be found at here[8]. The user's intent is carried in the EEG recordings which are forwarded to the deep learning based framework for interpretation. The recognized intent is send to sensor hive through WLAN to navigate the robot to get the desired object. Starting from the table near the kitchen, the PR2 robot receives action commands (as shown in Table 1) and walks forward until the specific position with the auxiliary of RFID tag. Then, the robot grasps the object, turns back and walks along the path to the table in living room and unlooses hands to put the beverage on the table. The simulation result shows that the robot can 100% precisely grasp and unloose object according to the path planned in the subject's mind. The simulation platform is in Gazebo toolbox and the robot controlling program is powered by Robot Operating System (ROS). This case randomly selects some EEG raw data from Subject 1 dataset as simulation inputs.

5 DISCUSSION

This paper proposes a novel unified deep learning based framework for enabling human-thing cognitive interactivity. The efficiency and effectiveness are demonstrated by the experiments while the deployment feasible is demonstrated by the case studies. However, there are still several open challenges: 1) the experiment only contains 7 subjects limited by the practical conditions, a larger and more diverse dataset is necessary to illustrate the effects of the proposed model; 2) the SAM component with focal zone is designed to automatically explore the latent dimension sequence of the input EEG data, nevertheless, the employment of SAM increases the training time resulted from more iterations of the LSTM cell; 3) most importantly, the RS stage shuffles the order and replicate the number of input dimensions to discover the optimal order in order to the best performance, but the optimal order can not be guaranteed to appear after the RS, thus try more

times if the classification result is unsatisfactory; 4) the WAS-LSTM exploits the spatial information among EEG channels, thus a number of channels are necessary to provide enough spatial information. Based on our experience, the minimum amount of EEG channel is 8 to keep a competitive performance. The aforementioned challenges will be addressed in the future work.

6 CONCLUSION

We propose a unified deep learning framework to bridge Brain-Computer Interface and Internet of Things in order to enable cognitive interactivity. We propose WAS-LSTM to extract inter-dimension dependency among the input signal of the human brain activities which are selected by the selective attention mechanism. We conduct real-world experiments to evaluate the proposed framework and the results demonstrate that our model outperforms the state-of-the-art baselines. Furthermore, our experience in developing two case studies, namely the brain typing system and the cognitive robot, are reported in the paper. These case studies validate the feasibility of the proposed framework.

REFERENCES

[1] B. Jagadish, M. Kiran, and P. Rajalakshmi, "A novel system architecture for brain controlled iot enabled environments," in *e-Health Networking, Applications and Services (Healthcom), 2017 IEEE 19th International Conference on*. IEEE, 2017, pp. 1–5.

[2] L. Yao, Q. Z. Sheng, and S. Dustdar, "Web-based management of the internet of things," *IEEE Internet Computing*, vol. 19, no. 4, pp. 60–67, 2015.

[3] N. K. Tran, Q. Z. Sheng, M. A. Babar, and L. Yao, "Searching the web of things: State of the art, challenges, and solutions," *ACM Computing Surveys*, vol. 50, no. 4, pp. 55:1–55:34, 2017.

[4] A. Vallabhaneni, T. Wang, and B. He, "Brain—computer interface," in *Neural engineering*. Springer, 2005, pp. 85–121.

[5] A. Teles, M. Cagy, F. Silva, M. Endler, V. Bastos, and S. Teixeira, "Using brain-computer interface and internet of things to improve healthcare for wheelchair users."

[6] N. Elsayed, Z. S. Zaghloul, and M. Bayoumi, "Brain computer interface: Eeg signal preprocessing issues and solutions," *Brain*, vol. 169, no. 3, 2017.

[7] X. Zhang, L. Yao, Q. Z. Sheng, S. S. Kanhere, T. Gu, and D. Zhang, "Converting your thoughts to texts: Enabling brain typing via deep feature learning of eeg signals," in *PerCom 2018*.

[8] B. Nguyen, D. Nguyen, W. Ma, and D. Tran, "Investigating the possibility of applying eeg lossy compression to eeg-based user authentication," in *IJCNN 2017*. IEEE, 2017, pp. 79–85.

8. https://youtu.be/VZYX1095Vkc

[9] T. Nakamura, V. Goverdovsky, and D. P. Mandic, "In-ear eeg biometrics for feasible and readily collectable real-world person authentication," *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 3, pp. 648–661, 2018.

[10] M. A. Rahman and M. Ahmad, "Evaluating the connectivity of motor area with prefrontal cortex by fnir spectroscopy," in *ECCE*. IEEE, 2017, pp. 296–300.

[11] M. Iijima and N. Nishitani, "Cortical dynamics during simple calculation processes: a magnetoencephalography study," *Clinical Neurophysiology Practice*, vol. 2, pp. 54–61, 2017.

[12] S. P. Fitzgibbon, D. M. Powers, K. J. Pope, and C. R. Clark, "Removal of eeg noise and artifact using blind source separation," *Journal of Clinical Neurophysiology*, vol. 24, no. 3, pp. 232–243, 2007.

[13] M. A. Rahman and M. Ahmad, "A straight forward signal processing scheme to improve effect size of fnir signals," in *ICIEV*. IEEE, 2016, pp. 439–444.

[14] A. Tarkiainen, M. Liljeström, M. Seppä, and R. Salmelin, "The 3d topography of meg source localization accuracy: effects of conductor model and noise," *Clinical Neurophysiology*, vol. 114, no. 10, pp. 1977–1992, 2003.

[15] E. Haselsteiner and G. Pfurtscheller, "Using time-dependent neural networks for eeg classification," *IEEE transactions on rehabilitation engineering*, vol. 8, no. 4, pp. 457–463, 2000.

[16] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, p. 436, 2015.

[17] M. Angelidou, "Smart city planning and development shortcomings," *Tema. Journal of Land Use, Mobility and Environment*, vol. 10, no. 1, pp. 77–94, 2017.

[18] K. Dhariwal and A. Mehta, "Architecture and plan of smart hospital based on internet of things (iot)," *Int. Res. J. Eng. Technol*, vol. 4, no. 4, pp. 1976–1980, 2017.

[19] A. Al-Ali, I. A. Zualkernan, M. Rashid, R. Gupta, and M. Alikarar, "A smart home energy management system using iot and big data analytics approach," *IEEE Transactions on Consumer Electronics*, vol. 63, no. 4, pp. 426–434, 2017.

[20] L. Yao, Q. Z. Sheng, B. Benatallah, S. Dustdar, X. Wang, A. Shemshadi, and S. S. Kanhere, "Wits: an iot-endowed computational framework for activity recognition in personalized smart homes," *Computing*, pp. 1–17, 2018.

[21] P. Cavanagh *et al.*, "Attention-based motion perception," *Science*, vol. 257, no. 5076, pp. 1563–1565, 1992.

[22] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, p. 529, 2015.

[23] Z. Wang, T. Schaul, M. Hessel, H. Van Hasselt, M. Lanctot, and N. De Freitas, "Dueling network architectures for deep reinforcement learning," *arXiv preprint arXiv:1511.06581*, 2015.

[24] H. Akaike, "Fitting autoregressive models for prediction," *Annals of the institute of Statistical Mathematics*, vol. 21, no. 1, pp. 243–247, 1969.

[25] A. Laurentini, "The visual hull concept for silhouette-based image understanding," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 16, no. 2, pp. 150–162, 1994.

[26] L. Lovmar, A. Ahlford, M. Jonsson, and A.-C. Syvänen, "Silhouette scores for assessment of snp genotype clusters," *BMC genomics*, vol. 6, no. 1, p. 35, 2005.

[27] M. Tokic, "Adaptive ε-greedy exploration in reinforcement learning based on value differences," in *Annual Conference on Artificial Intelligence*. Springer, 2010, pp. 203–210.

[28] F. A. Gers and E. Schmidhuber, "Lstm recurrent networks learn simple context-free and context-sensitive languages," *IEEE Transactions on Neural Networks*, vol. 12, no. 6, pp. 1333–1340, 2001.

[29] M. Sundermeyer, R. Schlüter, and H. Ney, "Lstm neural networks for language modeling," in *Thirteenth Annual Conference of the International Speech Communication Association*, 2012.

[30] Z. C. Lipton, D. C. Kale, C. Elkan, and R. Wetzel, "Learning to diagnose with lstm recurrent neural networks," *arXiv preprint arXiv:1511.03677*, 2015.

[31] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[32] W.-Y. Hsu, "Assembling a multi-feature eeg classifier for left–right motor imagery data using wavelet-based fuzzy approximate entropy for improved accuracy," *IJNS*, vol. 25, no. 08, p. 1550037, 2015.

[33] Y. R. Tabar and U. Halici, "A novel deep learning approach for classification of eeg motor imagery signals," *Journal of neural engineering*, vol. 14, no. 1, p. 016003, 2016.

[34] R. J. Martis, J. H. Tan, C. K. Chua, T. C. Loon, S. W. J. YEO, and L. Tong, "Epileptic eeg classification using nonlinear parameters on different frequency bands," *Journal of Mechanics in Medicine and Biology*, vol. 15, no. 03, p. 1550040, 2015.

[35] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

Xiang Zhang is currently a Ph.D. student (since 2016) at School of Computer Science and Engineering, University of New South Wales (UNSW). He received the Master degree (in 2016) from Harbin Institute of Technology (HIT), China. His research interests mainly in Deep learning, Brain-Computer Interface (BCI), Internet of Things (IoT), and Human Activity Recognition.

Lina Yao received the PhD degree in computer science from the University of Adelaide, Australia. She is currently a lecturer in the School of Computer Science and Engineering, UNSW. Her research interests lie in machine learning and data mining with applications to the Internet of Things, Brain-Computer Interface (BCI), information filtering and recommending, and human activity recognition. She is a member of the IEEE and the ACM.
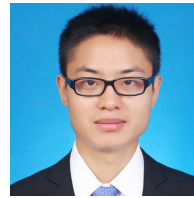
Shuai Zhang is a PhD student at the School of Computer Science and Engineering, University of New South Wales , as well as at Data61, CSIRO. He received a Bachelor degree from the School of Information Management, Nanjing University. His major research interests lie in the field of recommender systems, deep learning and internet of things. He is a student member of the IEEE and ACM.

Salil S. Kanhere received his Ph.D. from Drexel University. He is an associate professor in the School of Computer Science and Engineering at UNSW. His research interests include the Internet of Things, pervasive computing, crowd sourcing, sensor networks, and security. He has published 170 peer-reviewed articles and delivered over 20 tutorials and keynote talks. He is a Senior Member of ACM. He is a recipient of the Humboldt Research Fellowship.

Quan Z. Sheng is a full Professor and Head of Department of Computing at Macquarie University. Before moving to Macquarie, Michael spent 10 years at School of Computer Science, the University of Adelaide (UoA). Michael holds a PhD degree in computer science from the University of New South Wales (UNSW) and did his post-doc as a research scientist at CSIRO ICT Centre. From 1999 to 2001, Sheng also worked at UNSW as a visiting research fellow.

Yunhao Liu received his BS degree in Automation Department from Tsinghua University, and an MA degree in Beijing Foreign Studies University, China. He received an MS and a Ph.D. degree in Computer Science and Engineering at Michigan State University, USA. Yunhao is now MSU Foundation Professor and Chairperson of Department of Computer Science and Engineering, Michigan State University, and holds Chang Jiang Chair Professorship (No Pay Leave) at Tsinghua University.