Foundations and Trends® in Electronic Design Automation
Vol. XX, No. XX (2018) 1–67

DOI: 10.1561/XXXXXXXXXX

now
the essence of knowledge

Smart Healthcare

Hongxu Yin
hongxuy@princeton.edu
Electrical Engineering
Princeton University

Ayten Ozge Akmandor
akmandor@princeton.edu
Electrical Engineering
Princeton University

Arsalan Mosenia
arsalan@princeton.edu
Electrical Engineering
Princeton University

Niraj K. Jha
jha@princeton.edu
Electrical Engineering
Princeton University

Contents

Abstract

Internet-of-Things and machine learning promise a new era for healthcare. The emergence of transformative technologies, such as Implantable and Wearable Medical Devices (IWMDs), has enabled collection and analysis of physiological signals from anyone anywhere anytime. Machine learning allows us to unearth patterns in these signals and make healthcare predictions in both daily and clinical situations. This broadens the reach of healthcare from conventional clinical contexts to pervasive everyday scenarios, from passive data collection to active decision-making.

Despite the existence of a rich literature on IWMD-based and clinical healthcare systems, the fundamental challenges associated with design and implementation of smart healthcare systems have not been well-addressed. The main objectives of this article are to define a standard framework for smart healthcare aimed at both daily and clinical settings, investigate state-of-the-art smart healthcare systems and their constituent components, discuss various considerations and challenges that should be taken into account while designing smart healthcare systems, explain how existing studies have tackled these design challenges, and finally suggest some avenues for future research based on a set of open issues and challenges.

H. Yin, A. O. Akmandor, A. Mosenia and N. K. Jha. *Smart Healthcare*. Foundations and Trends® in Electronic Design Automation, vol. XX, no. XX, pp. 1–67, 2018.
DOI: 10.1561/XXXXXXXXXX.

1

Introduction

A rapidly aging population and the dramatic increase in the cost of in-hospital healthcare have led to the recognition of the importance of efficient healthcare systems [Nia et al., 2015] and fostered several novel research directions at the intersection of healthcare, data analytics, wireless communication, embedded systems, and information security. Implantable and Wearable Medical Devices (IWMDs), which facilitate non-invasive prevention, early diagnosis, and continuous treatment of medical conditions, are envisioned as key components of modern healthcare [Ghayvat et al., 2015, Mukhopadhyay, 2015, Mosenia et al., 2017b]. The computational power, energy capacity, and networking capabilities of IWMDs have improved significantly in the last decade while their sizes have decreased drastically. These technological advances have brought daily healthcare systems from a distant vision to the verge of reality. Furthermore, the emergence of Internet-of-Things (IoT) and the introduction of new computing/networking paradigms (such as Cloud computing and Fog computing), which make possible systems consisting of several heterogeneous sensing and computing devices, have revolutionized traditional healthcare and provided an opportunity to replace in-hospital medical systems with Internet-connected IWMD-

based systems, thus bringing us to the dawn of a new era of smart healthcare.

Smart healthcare does not have a unique definition. However, *our broad interpretation of smart healthcare is that besides **clinical** usage, it also utilizes IWMDs to gather, store, and process various types of physiological data during **daily** activities.* Smart healthcare systems may rely on wireless connectivity to take advantage of external resources, e.g., computational/storage resources available on nearby devices or the Cloud, or inform a clinician about the patient's medical condition. Hence, smart healthcare offers a proactive approach to early detection and even prevention of medical conditions. It even enables physicians and clinicians to assist patients in their home environment where they can be continuously monitored with the help of numerous Internet-connected healthcare systems. This reduces the need for institutionalization and hospitalization, and is especially beneficial to the disabled and elderly. It also has the potential to reduce healthcare costs significantly and enhance the quality of life of patients.

Since the introduction of the first IWMD (an implantable pacemaker) in 1958, several types of IWMDs have been developed and introduced in the market, ranging from sweat-analyzing devices [Gao et al., 2016] to Internet-connected multi-sensor continuous long-term health monitoring systems [Nia et al., 2015, Pantelopoulos and Bourbakis, 2010]. However, despite a rich body of literature on IWMD-based and clinical healthcare systems (see [Pantelopoulos and Bourbakis, 2010], [Mosenia et al., 2017b], and [Musen et al., 2014] for a comprehensive survey), the fundamental challenges associated with design and implementation of smart healthcare systems have not yet been well-addressed. The main goals of this article are to define the scope of smart healthcare and investigate state-of-the art smart healthcare systems, their constituent components, their design considerations, and how existing studies have tackled these challenges. In particular, we do the following.

- We present a novel framework for smart healthcare, which aims to support both in-patient and out-patient health monitoring and discuss and compare clinical and daily healthcare.
- We describe several emerging smart healthcare systems, including

IBM Watson [High, 2012], Open mHealth [Estrin and Sim, 2010], Health Decision Support System (HDSS) [Yin and Jha, 2017], Stress Detection and Alleviation system (SoDA) [Akmandor and Jha, 2017], and an energy-efficient system for continuous health monitoring of a patient's medical condition over the long term [Nia et al., 2015].

- We discuss several considerations and challenges that should be taken into account while designing smart healthcare systems.
- We describe five research trends for addressing these design considerations, including compact deep neural networks and compressive sensing to drastically reduce computation energy and storage, and MedMon, OpSecure, and SecureVibe to enhance security of healthcare systems.
- Finally, we discuss several future research directions, including the need to obtain medical datasets and machine learning models for them, standardization and infrastructure, and the promising role that Fog computing can play in smart healthcare.

The rest of the article is organized as follows. In Chapter 2, we present a smart healthcare framework that enables exploitation of the rapid clinical-to-daily healthcare expansion. In Chapter 3, we analyze five emerging systems that act as enablers of smart healthcare. In Chapter 4, we discuss associated design considerations and challenges in these systems, including efficiency, security, accuracy, cost, responsiveness, maintainability, scalability, reliability and fault tolerance. In Chapter 5, we describe five emerging research trends that address some of these challenges. In Chapter 6, we discuss open issues and future research directions. Finally, we conclude in Chapter 7.

2

What is Smart Healthcare?

Modern healthcare saves human lives and improves the quality of life. The average life expectancy has increased by five years in the past two decades [Salomon et al., 2013]. The wide-ranging impact of healthcare on billions of people around the globe has spurred enormous interdisciplinary research efforts and remarkable innovations. However, for decades, healthcare has been confined to clinics/hospitals. It has failed to utilize patient data obtained from the daily context, thus missing out on the ability to catch a disease in its early stages. Recent years have seen such deficiencies beginning to get addressed by advances in daily healthcare enabled by IWMDs, i.e., Wearable Medical Sensors (WMSs) and Implantable Medical Devices (IMDs). The possibility of daily healthcare monitoring, in conjunction with conventional clinical healthcare, promises to usher in a new era of smart healthcare.

In this chapter, we present a novel smart healthcare framework that captures the rapid clinical-to-daily healthcare expansion. This framework defines the scope of smart healthcare, thus helps unify various fragmented healthcare tasks under one umbrella.

2.1 The Smart Healthcare Framework

Depending on where healthcare takes place, smart healthcare can be divided into two major parts: (i) daily healthcare, depicted on the left of Fig. 2.1, and (ii) clinical healthcare, depicted on the right of Fig. 2.1. These two parts are separated by the clinical boundary.

The upper section of Fig. 2.1 summarizes the five major ***tasks*** that need to be carried out by smart healthcare:

- Disease prevention: (i) daily prevention
- Disease diagnosis: (ii) daily and (iii) clinical diagnosis
- Disease treatment: (iv) clinical and (v) daily treatments

These five ***tasks*** fall under three categories that correspond to the three most critical challenges of modern healthcare: prevention, diagnosis, and treatment of human diseases. Each ***task*** constitutes a vibrant research field that includes challenging research topics such as fitness tracking, daily disease diagnosis, physician variance reduction, treatment plan selection, and precision medicine (summarized in bullet points in Fig. 2.1). These five ***tasks*** need to be carried out in a sequential and circular manner, as indicated by the arrows in Fig. 2.1. This directed loop is referred to as the smart healthcare ***loop***.

We call healthcare systems 'smart' when they have a decision-making ability. This ability is enabled by data analytics, as shown in the lower section of Fig. 2.1. Information distillation starts with various data types of interest that may assist decision-making. Data types vary from physiological and environmental readings in the daily context to physician observations and laboratory test results in the hospital/clinic. These data must be efficiently captured, processed, and securely transmitted to the upper levels of the healthcare system, assisted by machine learning engines, such as WEKA [Hall et al., 2009] and TensorFlow [Abadi et al., 2016], to extract health inferences. These health inferences form an integral part of the ***tasks*** in the ***loop***.

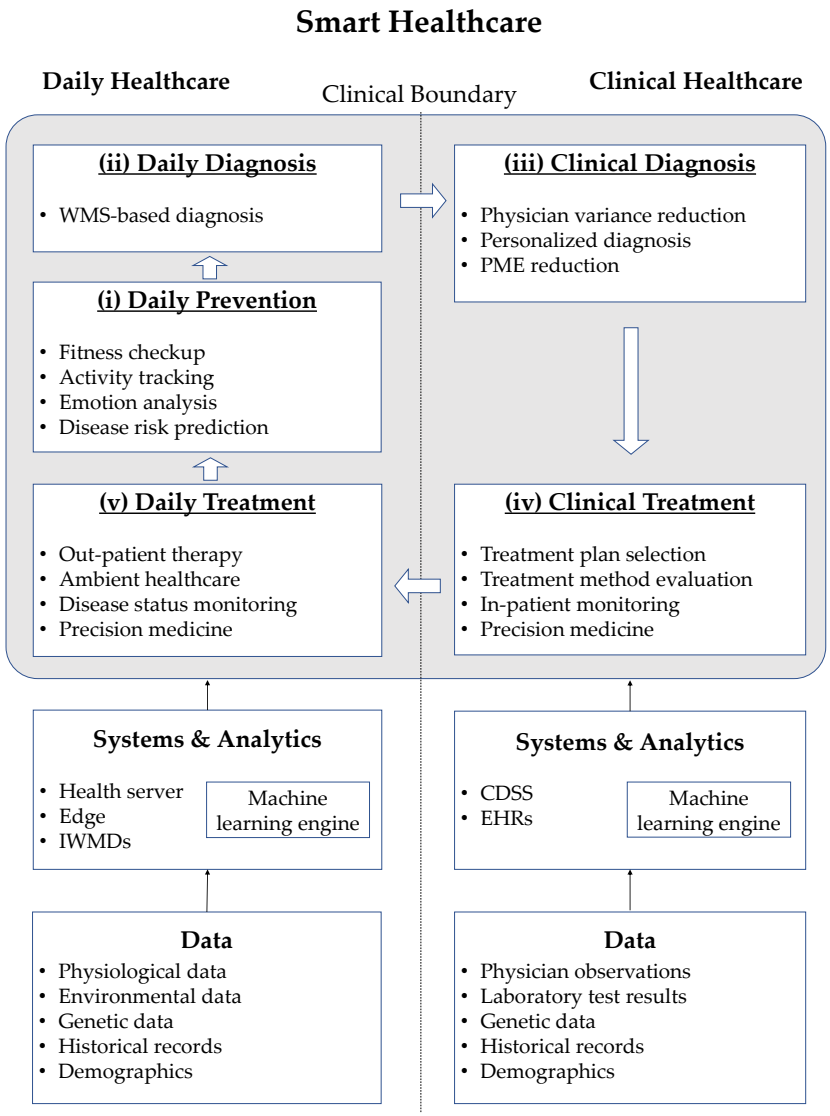


Figure 2.1: The smart healthcare framework.

Next, we zoom into clinical healthcare and then daily healthcare and discuss them in detail.

2.2 Clinical Healthcare

Despite remarkable progress over the past few decades, the clinical healthcare system in the U.S. is still far from being optimal. For example, a recent study [Makary and Daniel, 2016] shows that Preventable Medical Errors (PMEs) accounted for more than 251,000 deaths in 2013, making it the third leading cause of death in the U.S. hospitals after heart disease and cancer. This is substantially higher than the 98,000 deaths due to preventable medical errors mentioned in the 1999 IOM report [Kohn et al., 2000].

Computerized information systems, e.g., Clinical Decision Support Systems (CDSSs) and Electronic Health Records (EHRs), provide physicians and healthcare providers with intelligently filtered clinical suggestions, thus can greatly improve the quality of clinical healthcare. More than 66% of EHR-based CDSSs have been shown to significantly improve clinical practice in the long run [Hunt et al., 1998]. As a result, more hospitals and clinics are adopting CDSSs and EHRs to assist physicians. This was aided by the Health Information Technology for Economic and Clinic Health Act of 2009 that was accompanied by a $27B federal disbursement.

The sharp increase in the amount of patient-specific clinical data provides a fertile resource for machine learning algorithms to derive healthcare inference. Rapid algorithmic advancements in machine learning have even enabled super-human clinical decision-making performance. For example, a deep Convolutional Neural Network (CNN) has been shown to perform on par with 21 board-certified dermatologists in skin cancer classification [Esteva et al., 2017]. CheXNet, a 121-layer CNN, was able to beat the average performance of four radiologists in pneumonia detection and analysis [Rajpurkar et al., 2017]. Deep Patient [Miotto et al., 2016] deploys a three-layer stack of auto-encoders to capture the regularities and dependencies in the aggregated EHRs of 700,000 patients. It uses the extracted rules for disease risk predic-

tion and achieves very high accuracy on 76,214 test patients with 78 diseases. With the help of massive parallel deep learning on Graphical Processing Units (GPUs), DeepBind analyzes millions of genome sequences (previously between 10,000 and 100,000) to identify causal disease variants [Alipanahi et al., 2015]. This speeds up exploration of relationships among DNA, key molecules in cells, and associated disease risks, thus assisting with the development of precision medicine [Leung et al., 2016].

However, clinical healthcare is still restricted to hospitals/clinics. It has very limited access to the daily health status of patients, a context in which most diseases actually develop and are treated [Estrin and Sim, 2010]. This can lead to many deficiencies. For example, daily health data form an extremely important and sometimes the only information source for physicians and CDSSs for making diagnostic decisions. Relying on self-reported symptom recalls from patients can be quite error-prone, given that symptoms often may not even be noticeable by an individual. These shortcomings point to the need to complement clinical healthcare in the daily scenario.

2.3 Daily Healthcare

As opposed to decades-long advancements in clinical innovations, daily healthcare is an emerging research field. It requires a steady, consistent, accurate, yet user-transparent, data acquisition mechanism. This has only been made feasible by recent advancements in low-power sensors and signal processing techniques.

The past decade has witnessed the deployment of many disruptive sensors in IWMDs, stationary sensors in house/office/gym, and embedded sensors in mobile phones. These sensors can consistently and persistently collect vast amounts of health-related data in the daily context to enable decision-making. This falls under an IoT paradigm where things communicate and cooperate with each other pervasively to achieve common goals [Atzori et al., 2010]. The IoT framework contains three hierarchical computation layers: sensor, edge, and the Cloud. In the context of healthcare, the three IoT layers sequentially transform

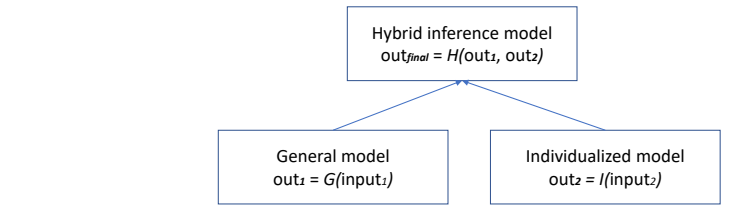
health-related data, such as physiological signals and environmental readings, into purposeful healthcare inferences, such as disease diagnosis and activity prediction, thus delivering smartness to daily healthcare. With the need to support billions of devices, the current IoT framework suffers from a limited sensor energy budget, constrained communication bandwidth, limited server storage, and a wide attack surface for malicious adversaries [Li et al., 2011, Halperin et al., 2008, Yin et al., 2015]. These shortcomings lead to significant design challenges, such as the need for efficiency and security. We discuss these design considerations in Chapter 4 and the techniques to address these challenges in Chapter 5.

Hitherto, two major approaches have been used to obtain healthcare inference along the IoT hierarchy:

- A top-down approach from the Cloud: this approach starts from the Cloud to obtain population-level inferences and extract general rules. The data amount that such an approach needs to tackle is large, typically terabytes (10^{12} B) or more. Consequentially, this approach suffers from high analysis costs for data collection, storage, and pre-processing. IBM Watson[1] uses this approach, as explained in detail in Chapter 3.

- A bottom-up approach from the sensor/edge: this approach starts with the user side to obtain individualized inferences. It typically assembles and analyzes the data from relatively smaller patient groups. As a result, it enables a fine-grained analysis that can lead to more accurate individualized models. The amount of data that needs to be analyzed is typically in the megabytes (10^6 B) to gigabytes (10^9 B) range, hence much smaller than the top-down approach. Hence, it reduces associated costs for machine learning model generation. Examples using this approach include Open mHealth [Estrin and Sim, 2010], HDSS [Yin and Jha, 2017] and SoDA [Akmandor and Jha, 2017], as explained in detail in Chapter 3.

[1]IBM Watson, https://www.ibm.com/watson.



Training data acquisition	Cloud	Sensor/edge
Model cost	High	Low
Data amount	Large (TB)	Small (MB)/medium (GB)
Target	General rules	Individual traits

Figure 2.2: A hybrid inference model for daily healthcare.

However, neither approach is mature yet. A top-down approach, which yields a generalized population-level model, may fail to cater to individual patients by not being able to accommodate his/her personal traits. This is also a fundamental research question faced in personalized diagnosis and precision medicine. On the other hand, the bottom-up approach, which yields an individualized model, is limited by the fact that analysis of data from a single patient group may fail to yield general rules applicable at the population level. This may lead to inference models that perform well within the group, but are not generalizable to other groups since the features may not be drawn from the same probability distribution.

To address the above concerns, we describe a potential inference model for smart healthcare as an ensemble of both generalized and individualized models. We refer to it as a ***hybrid inference model***, as shown in Fig. 2.2. The meta learner function $H(\cdot)$ accepts inputs from both the generalized function $G(\cdot)$ and individualized function $I(\cdot)$ as its base learners. This can effectively:

- tune a generalized model to each individual, given his/her personal physiological traits, and
- augment an individualized model with additional ground truths and larger knowledge base.

3

Emerging Smart Healthcare Systems

There is a need for ubiquitous healthcare to improve human well-being. Modern healthcare systems have become smart based on unprecedented advances in data analytics and increasingly pervasive based on rapid deployment of IoT.

In this chapter, we review several systems from the smart healthcare domain, including: (i) IBM Watson that extracts rules from the medical literature to answer health-related questions [High, 2012], (ii) Open mHealth that aims at daily chronic disease prevention based on data collected from mobile phone applications [Estrin and Sim, 2010], (iii) HDSS that enables disease diagnosis based on WMSs and machine learning ensembles [Yin and Jha, 2017], (iv) SoDA that focuses on continuous stress detection and alleviation via integration of WMS data with machine learning algorithms [Akmandor and Jha, 2017], and (v) an energy-efficient system that tackles continuous monitoring of a patient's medical conditions over the long term [Nia et al., 2015].

3.1 IBM Watson

IBM Watson is a cognitive system that combines deep Natural Language Processing (NLP), hypothesis generation, and dynamic learning to generate confidence-based responses [High, 2012]. It is capable of condensing information from an immense amount of unstructured and noisy data: scientific articles, textbooks, user manuals, guidelines, frequently asked questions, plans, laboratory notes, news, and proprietary data. The extracted knowledge base is stored as a Watson corpus. Watson generates a unique corpus per target domain. Due to its significant NLP capabilities, Watson has tackled a wide range of target domains, such as engineering, medicine, law, and finance [Chen et al., 2016].

Watson uses its corpus for question-answer style inferences. When a new question is raised, it (i) captures the main question features, (ii) acquires the candidate answers across hundreds of hypotheses generated by the corpus, (iii) compares answers through reasoning algorithms, and then (iv) selects the answer that has the highest confidence score. This top-down approach, i.e., starting with reading all available information in the target domain, yields amazing results but at a very high cost. For example, IBM Watson won against human champions in the Jeopardy Clash Knowledge Test in 2011. It read roughly 200M pages of content to acquire its corpus to prepare for this test. It had to rely on 90 IBM Power750 processors and 16 terabytes of RAM during the competition [Chandrasekar, 2014].

In the healthcare domain, Watson's better-than-human content reading capability enables it to answer health-related questions in both daily and clinical scenarios. It can scan and analyze content from a wide range of medical resources: scientific journals, patents, drug and disease related ontologies, clinical trials, EHRs, laboratory and imaging data, genomic data, claims data, and web social content [Chen et al., 2016]. This has led to three major applications of Watson in healthcare:

- Oncology: Watson can compare a patient profile with relevant clinical trials/records to evaluate and rank cancer treatment options. It cuts down time costs involved in reviewing the literature and EHRs dramatically. For example, it took Watson 10 min-

utes to finalize a treatment plan for a 76-year old brain-tumor patient, while this process took human experts roughly 160 hours to complete [Wrzeszczynski et al., 2017].

- Drug discovery: Watson can identify novel drug targets and new uses of existing drugs. For example, it successfully identified 15 new drug candidates for a malaria parasite from a drug candidate pool available from a pharmaceutical company [Chen et al., 2016]. To do so, it first checked the literature, identified relevant drugs that have been shown to be effective for malaria parasite mitigation, and then checked the candidate drug's similarity in terms of chemical structures and action mechanisms.

- Genomics: Watson can unearth new associations and relationships between genes, proteins, drugs, and diseases. It can also rank and predict the most likely driver mutation and the alteration type of DNA in a patient's tumor to enhance personalized treatments. This provides physicians with more therapeutic options.

3.2 Open mHealth

The Open mobile Health (mHealth) project is aimed at daily chronic disease prevention and management based on data collected from healthcare applications in mobile phones and devices [Estrin and Sim, 2010]. There were more than 13,000 health-related applications available on Apple's iPhone by 2012 [Localystics, 2012]. These applications enable patients to electronically record and track their vital physiological signs on a daily basis, thus enabling satisfactory user-centered outcomes. For instance, WellDoc is a mobile phone based diabetes management application that uses messages and prompts to track the glucose level of its users [Quinn et al., 2011]. A randomized controlled trial observed that WellDoc leads to a significant reduction in glycated hemoglobin among its users and a 20% reduction in clinical visits and emergency care usage [Quinn et al., 2011].

Open mHealth aims to standardize the fragmented mHealth applications deployed on mobile phones. It proposes open Application

Programming Interfaces (APIs) built around a minimal set of common communication protocols [Estrin and Sim, 2010, Chen et al., 2012]. This enables the sharing of data and application modules between various devices, across different operating systems, and over multiple chronic diseases.

InfoVis was the first standard data visualization tool developed in the Open mHealth project [Chen et al., 2012]. It accepts data inputs from lower-level lego-like reusable software modules: Data Processing Units (DPUs) and Data Visualization Units (DVUs). Each DPU and DVU performs a general-purpose task. Multiple DPUs and DVUs work collaboratively to deliver application-level functionalities.

Chen et al. have developed DPUs and DVUs for an mHealth application called Post-Traumatic Stress Disorder (PTSD) explorer that helps PTSD patients manage acute distress symptoms [Chen et al., 2012]. Mobile applications can offer unique standard care to PTSD patients, given that such patients seldom seek in-person consultation due to the attached stigma, logistical barriers, and hard-to-notice symptoms [Hoge et al., 2004]. The DPUs and DVUs enable a direct visualization of the self-reported PTSD checklist scores and blood glucose levels [Chen et al., 2012]. Given an open architecture based on shared APIs, these PTSD DPUs and DVUs can also be used for other applications, such as the visualization of self-reported chronic pain measurements from patients [Chen et al., 2012].

3.3 HDSS: Health Decision Support System

The Health Decision Support System (HDSS) enables disease diagnosis in both in- and out-of-clinic scenarios through the integration of WMS data to CDSSs [Yin and Jha, 2017]. HDSS has a multi-tier structure, starting with a WMS tier, backed by robust machine learning, that enables diseases to be tracked individually by a disease diagnosis module. It sequentially structures the information framework for daily health monitoring, initial clinical checkup, detailed clinical examination, and post-diagnostic decision support.

HDSS has two major parts to support daily and clinical healthcare:

(i) Pervasive Decision Support System (PHDS), shown on the left of Fig. 3.1 and (ii) PHDS-assisted CDSS (CDSS+), shown on the right of Fig. 3.1. PHDS acts on WMS data for daily disease diagnosis. CDSS+ tackles clinical diagnosis. HDSS has four major tiers. Tier-1 assists with daily health monitoring. The decision modules in Tier-1 are trained using clinical domain knowledge. This transmits physician expertise across the clinical boundary, and can thus help individuals without professional medical training to effectively track their diseases. When an alert is raised at Tier-1, it passes symptom records, stored as disease-onset records, across the clinical boundary. Tier-2 provides immediate decision support to physicians for an incoming patient based on basic clinical measurements. Tier-3 entails a more detailed diagnostic analysis based on detailed laboratory measurements. Finally, Tier-4 provides post-diagnostic treatment, prescription, and lifestyle suggestions. Higher-level tiers gather more information than lower-level ones, but at higher time and energy costs. HDSS operates across these tiers in a sequential and closed-loop manner, as indicated by the large arrow behind the four tiers in Fig. 3.1.

HDSS deploys various Transitions (T) to facilitate the flow of information among the various tiers, as depicted by indexed arrows in Fig. 3.1. When an alert is raised at Tier-1, a transition T_{IN} passes disease-onset records across the clinical boundary. At Tier-2, HDSS aggregates the data with additional physician insights, and then passes the data to the diagnosis engine through T_1. The diagnosis engine contains libraries accessible by machine learning engines, such as WEKA [Hall et al., 2009] and TensorFlow [Abadi et al., 2016]. If a diagnosis requires further laboratory measurements, T_2 transfers HDSS to Tier-3. Otherwise, $T_{2'}$ transfers HDSS to Tier-4. In either case, diagnostic suggestions are immediately available to physicians. When T_2 occurs, Tier-3 reaches the diagnosis engine through T_3. The engine orders appropriate laboratory tests via T_4, after which test results are fed back to the diagnosis engine via T_5. Diagnosis at Tier-3 consumes more time and expense compared to Tier-2. For example, it can take 12-16 hours to acquire a blood test report (even longer for tests like computed tomography and functional magnetic resonance imaging). However, Tier-3 is still the most

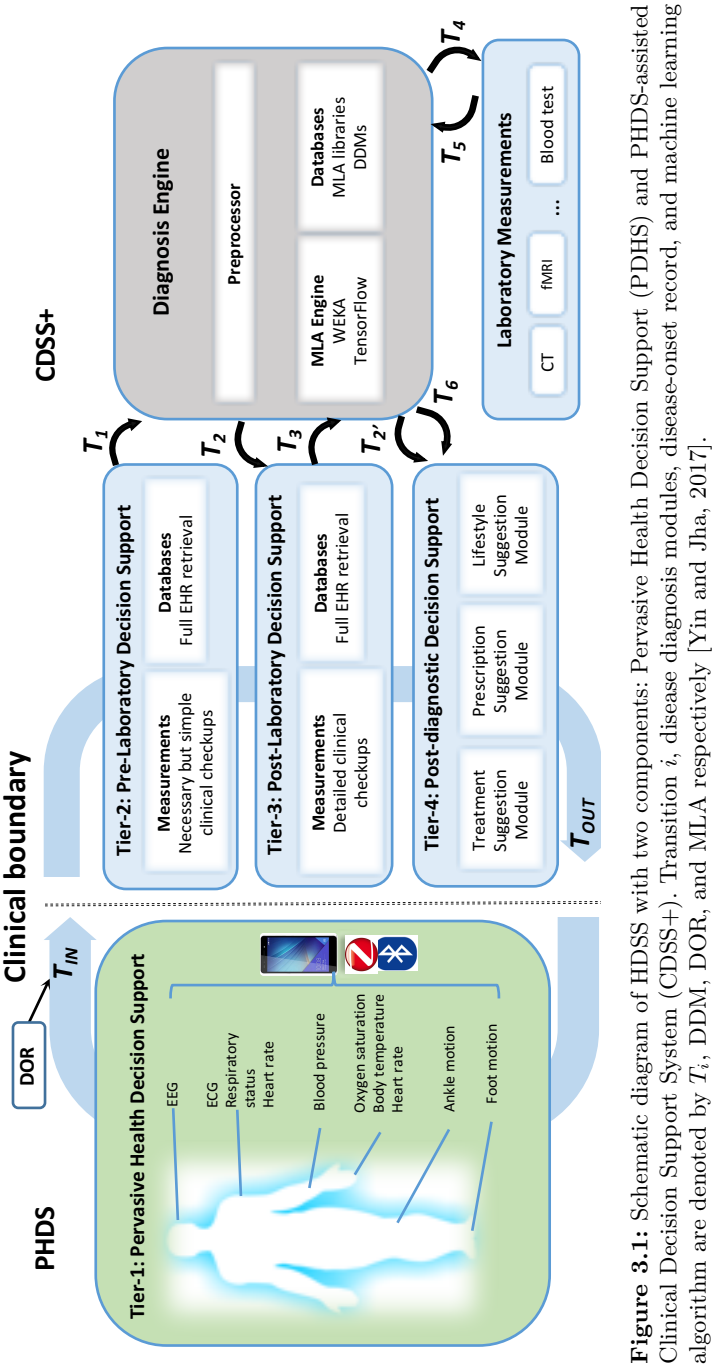


Figure 3.1: Schematic diagram of HDSS with two components: Pervasive Health Decision Support (PDHS) and PHDS-assisted Clinical Decision Support System (CDSS+). Transition i, disease diagnosis modules, disease-onset record, and machine learning algorithm are denoted by T_i, DDM, DOR, and MLA respectively [Yin and Jha, 2017].

important tier, since it needs to finalize a diagnosis based on all available clinical information. Tier-4 is reached through $T_{2'}$ or T_6 for post-diagnostic suggestions. A final T_{OUT} indicates completion of the clinical visit.

HDSS relies on disease diagnosis modules for disease monitoring. Each such module specifies the unique and necessary information framework components for the diagnosis of a target disease. Thus, one only needs to modify the disease diagnosis module of the target disease to update or evaluate the diagnostic rules instead of restructuring the entire HDSS. Multiple disease diagnosis modules derive disease ***signatures*** in parallel to monitor various diseases simultaneously. To differentiate one disease from another, HDSS adopts the International Statistical Classification of Diseases and Related Health Problems (ICD) coding system for disease diagnosis module indexing. The ICD code is maintained by the World Health Organization. Its latest version, ICD-10, has two coding sub-systems: ICD-10-CM for disease categorization and ICD-10-PCS for in-patient procedure identification. HDSS uses ICD-10-CM for disease indexing. ICD-10-CM currently contains 69,000 human disease codes allocated to 20 disease categories [Quan et al., 2005].

Yin and Jha demonstrate the feasibility of disease diagnosis through HDSS by generating disease diagnosis modules for arrhythmia, type-2 diabetes, breast cancer, urinary bladder disorder, renal pelvis origin nephritis, and hypothyroid disease based on University of California at Irvine (UCI) datasets [Lichman, 2013, Czerniak and Zarzycki, 2003]. They experiment with eight supervised machine learning algorithms and six ensemble methods for disease diagnosis module training. Table 3.1 summarizes their names, abbreviations, along with brief descriptions. An ensemble method specifies the rules for a meta learner to make a final decision based on predictions from its base learners. In general, ensemble methods boost machine learning algorithm performance.

The disease diagnosis module performance is summarized in Table 3.2. The rows indexed with *Type* list the machine learning models that achieve the highest diagnostic accuracies. The rows indexed with *Obj.* list the performance objective that varies between binary classifications (B) at Tier-1 to multi-class classification of k classes (M-k) at

Table 3.1: Machine learning algorithms and ensemble methods [Yin and Jha, 2017]

Name	Abbr.	Descriptions
Naive Bayes	NB	Bayes theorem based probabilistic learner
Bayes network	BN	Network driven, conditional tables at nodes
k-nearest neighbor	IB-k	Similarity analysis with k closest instances
Best-first decision tree	BFTree	Tree with binary splits on features
J48	J48	Pruned or unpruned decision tree
Decision table	DT	A decision table based majority learner
Support vector machine	SVM	Support vector based linear separator
Multilayer perceptron	MLP	Back propagation based neural network
Stacker	ST	Combiner based stacking of base learners
AdaBoost (Booster)	ADA	Weighted decision of weak classifiers
DECORATE (Voter)	DEC	Voting through diversified base learners
Bagger	BAG	Training with sampled subsets
Random tree	RT	Bagging on tree sampling features
Random forest	RF	Bagging on tree sampling instances/features

Tier-2 and Tier-3. The rows indexed with ACC summarize the highest diagnostic accuracies at the corresponding diagnostic tiers. These values are comparable or better than relevant work from the literature [Suryakumar et al., 2013, Cao et al., 2016, Arif and Basalamah, 2012, Jadhav et al., 2011]. What is noteworthy is that HDSS obtains high diagnostic accuracies even at Tier-1, in which the data can only be collected from WMSs.

To evaluate the scalability of HDSS, Yin and Jha conducted a literature review over the last 10-year span of diseases whose biomedical datasets are not yet publicly available. They discuss seven representative works that verify the applicability of machine learning to diagnosis and treatment of malaria [Das et al., 2013], sleep apnea [Khandoker et al., 2009], Parkinson's disease [Tahir and Manap, 2012], respiratory malfunction [Palaniappan et al., 2014], seizure [Tzallas et al., 2009], skin lesion [Korotkov and Garcia, 2012], and prenatal/perinatal defections [Cerqueira et al., 2014].

Fig. 3.2 summarizes the current scope of HDSS over the ICD-10-CM categories. It contains three major sections. The lightest section includes the four ICD-10-CM categories covered by the disease diagnosis modules in Table 3.2. The light gray section highlights the categories that contain at least one verified machine learning model, but on private datasets. HDSS could be applied to these categories as well once these datasets are made public. Finally, the dark gray section lists the open categories

Table 3.2: Performance summary of HDSS on UCI biomedical datasets [Yin and Jha, 2017]

	Disease → **DDM ↓**	Arrhythmia	Diabetes type-2	Breast cancer	Urinary bladder disorder	Renal pelvis origin nephritis	Hypothyroid
	ICD-10-CM	I49.9	E11-*	C50-*	N32.0	N12	E03.9
Tier-1	Type	RF+F	NB+F	–	RT	NB	RF
	Obj.	B	B	–	B	B	B
	ACC	85.9%	77.6%	–	99.6%	93.7%	94.8%
Tier-2	Type	BAG(BN)+F	NB+F	–	RF+F	RT	ADA(BFTree)
	Obj.	M-16	B	–	B	B	B
	ACC	77.4%	77.6%	–	100% (100%[3])	99.9% (100%[3])	94.8%
Tier-3	Type	BAG(BN)+F	DEC(BN)	BAG(BN)+F	–	–	J48+F
	Obj.	M-16	B	B	–	–	B
	ACC	77.4% (78.9%[1])	78.9% (76.5%[2])	97.0% (95.5%[2])	–	–	99.3%

+F: feature filtering; DDM: disease diagnosis module; [1]:[Jadhav et al., 2011]; [2]:[Cao et al., 2016]; [3]:[Arif and Basalamah, 2012]. Abbreviations for the machine learning algorithms are summarized in Table 3.1.

where biomedical datasets and machine learning models are sparse. This offers opportunities to further broaden the scope of HDSS.

3.4 SoDA: Stress Detection and Alleviation System

Stress is linked with various health problems, ranging from cardiovascular diseases [Schubert et al., 2009] to sleep disorders [McEwen, 2004] and cancer [Irie et al., 2001]. Reducing the risk of these serious health problems requires keeping stress under control. Akmandor and Jha have introduced a system called SoDA to address this problem [Akmandor and Jha, 2017].

SoDA is an automatic stress detection and alleviation system that collects physiological signals using WMSs and performs machine learning inferences on them. Its main components are shown in Fig. 3.3. It uses machine learning inferences to provide continuous and user-/situation-oriented stress level tracking (green path) and coaching (red path).

The detailed flow of the operations in SoDA is shown in Fig. 3.4. As a first step, SoDA collects physiological data from the WMSs. It then processes the collected data to remove artifacts, i.e., outliers, baseline wander, power-line interference, and muscle noise, and extracts informative features. After inputting the feature values to a previously-

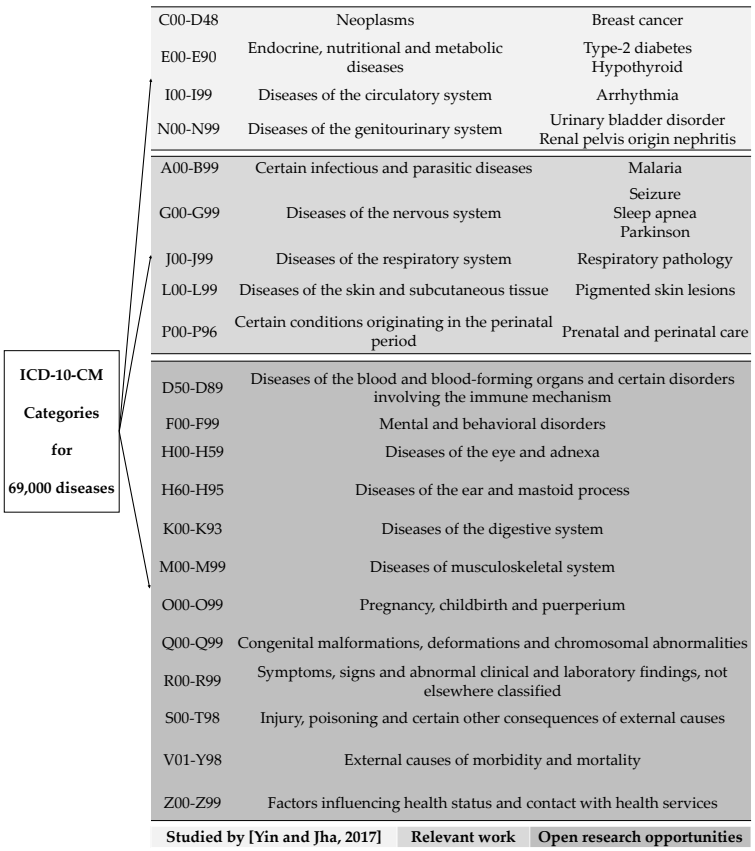


Figure 3.2: Coverage of HDSS on the ICD-10-CM disease categories based on the generated disease diagnosis modules and analysis of related work [Yin and Jha, 2017].

trained machine learning model (classification block in Fig. 3.4), SoDA determines if the epoch under question corresponds to 'stressed' or 'not stressed'. If the decision is 'not stressed', SoDA bypasses the stress alleviation protocol and performs operations on the upper path in Fig. 3.4. However, if the decision is 'stressed', SoDA activates the stress alleviation protocol through the lower path in Fig. 3.4. Using the stress alleviation protocol, outlined in Algorithm 1, SoDA guides the user to perform stress-reducing therapies and analyzes the extracted feature values from the collected physiological data. Depending on the tendency of the feature values, SoDA either terminates the stress alleviation

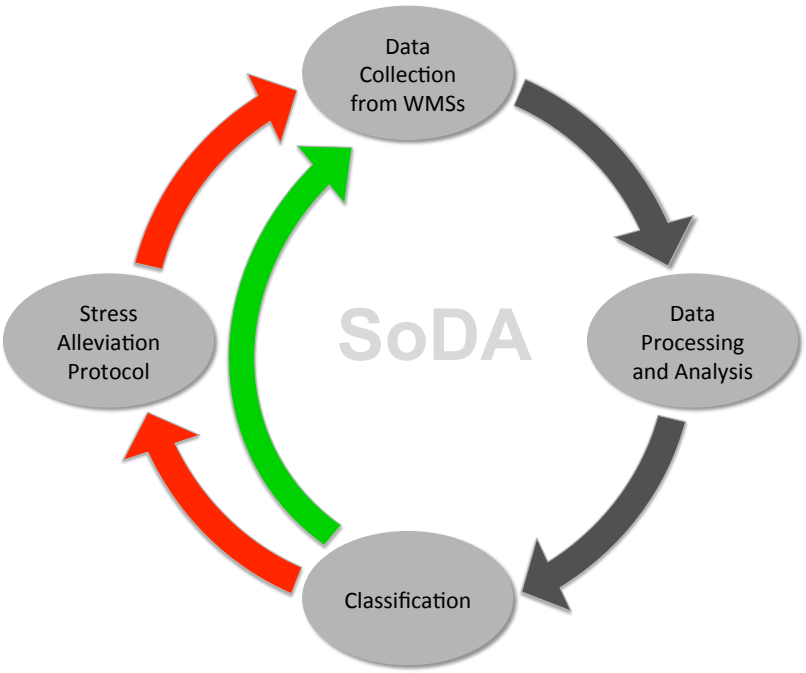


Figure 3.3: Main components of SoDA [Akmandor and Jha, 2017].

protocol or continues it with the same or the next stress-reducing therapy for a predefined time period.

SoDA has two operating models: 'generalized' and 'individualized'. In the 'generalized' model, the machine learning model is obtained using WMS data from a large group of individuals. Thus, the 'generalized' model can be used immediately. On the other hand, the 'individualized' model is obtained from the WMS data of only the individual in question. Since the model parameters are adjusted according to the specific user's data, the 'individualized' model requires extra training time. However, it provides higher classification accuracy during stress tracking and coaching.

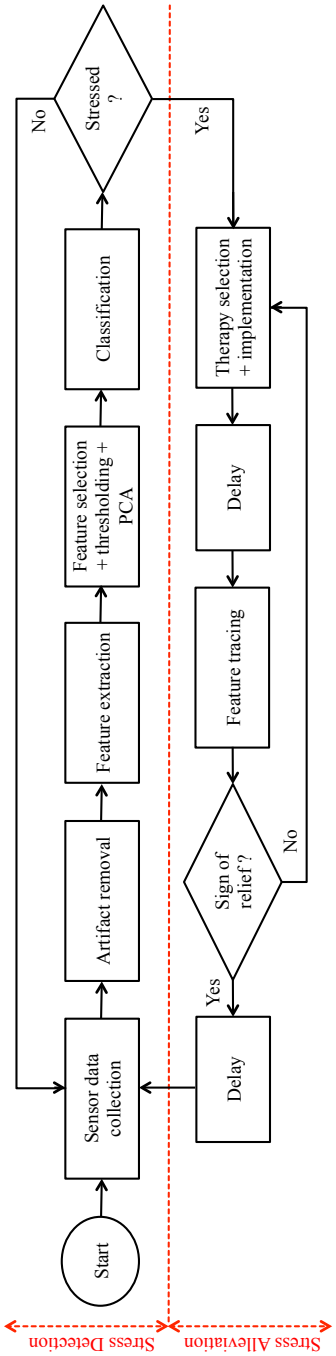


Figure 3.4: Flow of WMS data processing/analysis operations in the stress detection and alleviation stages [Akmandor and Jha, 2017].

Algorithm 1 Stress alleviation protocol [Akmandor and Jha, 2017]

Given: *therapySet*, set of the stress alleviation techniques.

therapy $\leftarrow$ *null*, $k \leftarrow 0$, $flag \leftarrow 0$
for $i = 1, ..., length(therapySet)$
    therapy $\leftarrow$ *therapySet(i)*
    Delay (30*sec.*)
    Compute selected N feature values
    Compute k, number of features showing stress relief
        if $k \geq N/2$
            Delay (30*sec.*)
            Compute selected N feature values
            Compute k
                if $k \geq N/2$
                    $flag \leftarrow 1$
                    return
                end
        end
end
if $flag = 0$, *none of the stress alleviation techniques is effective*
Give warning to the user
return
end

Akmandor and Jha use the following WMSs for stress detection and alleviation: Electrocardiogram (ECG), Galvanic Skin Response (GSR), Respiration rate (RESP), Blood Pressure (BP), and Blood Oximeter (BO). The data are collected from 32 participants. They include a total of eight stress-inducing epochs: four with and four without stress-reducing therapies. The collected WMS data are subjected to artifact removal, feature extraction, feature selection, and Principal Component Analysis (PCA). The final feature vectors are provided as inputs to machine learning models for decision-making. Thus, the stress inferences are done in a user-transparent fashion.

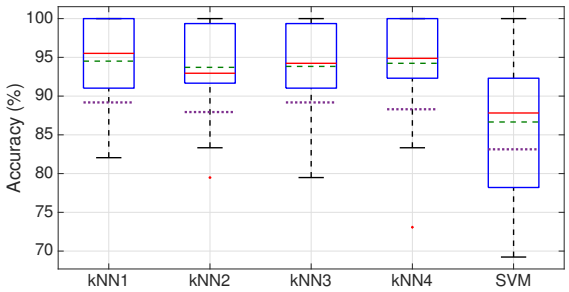
Following data processing and feature extraction, stress detection performance is analyzed on two feature sets (Table 3.3). SoDA uses

Table 3.3: Selected feature sets [Akmandor and Jha, 2017]

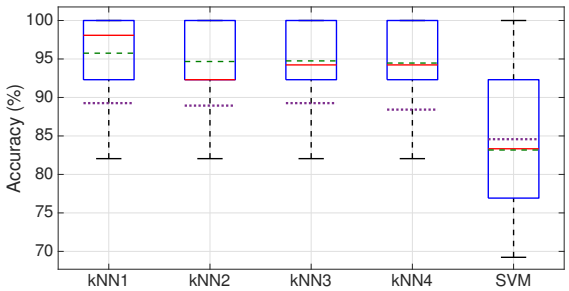
	Feature	Sensor	Set
1	ECG-derived respiration rate	ECG	I, II
2	Mean of skin conductance amplitude	GSR	I, II
3	Standard deviation of skin conductance amplitude	GSR	I, II
4	Sum of amplitudes of skin conductance responses above the threshold (continuous decomposition analysis)	GSR	I
5	Mean of tonic activity	GSR	I
6	Maximum positive deflection	GSR	I
7	Mean of respiration duration	RESP	I
8	RMS of respiration signal	RESP	I, II
9	Median of respiration duration	RESP	I
10	Mean of blood oxygen level	BO	I, II
11	Mean of systolic blood pressure	BP	I, II
12	Variance of systolic blood pressure	BP	I, II
13	Mean of diastolic blood pressure	BP	I, II
14	Mean of mean arterial pressure	BP	I, II
15	Variance of mean arterial pressure	BP	I, II

k-Nearest Neighbor (k-NN) and Support Vector Machine (SVM) Radial Basis Function (RBF) as classifiers. The 'individualized' model has the following average stress classification accuracies for the two feature sets (see Fig. 3.5(a) and Fig. 3.5(b)): 94.5-95.8%, 93.7-94.7%, 93.8-94.8%, 94.2-94.5%, and 86.7-83.2% for the k-NN ($k = 1$), k-NN ($k = 2$), k-NN ($k = 3$), k-NN ($k = 4$), and SVM RBF classifiers, respectively. For the analyses in the stress alleviation stage, Akmandor and Jha compare feature values with and without stress therapy and find that therapy results in a considerable improvement in stress levels.

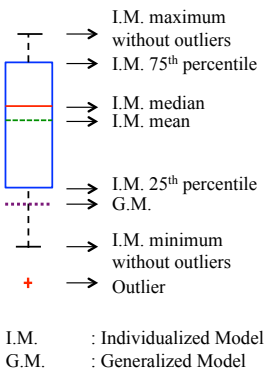
In summary, SoDA is an automatic and user-friendly stress level coach that continuously tracks physiological signals to detect stress and guides the user whenever needed. Due to its high stress detection accuracy and efficient stress alleviation, SoDA is a promising technology



(a)



(b)



(c)

Figure 3.5: Stress detection accuracy statistics of the machine learning algorithms for the 'individualized' and 'generalized' models with feature set (a) I, (b) II, and (c) definitions of the boxplot parameters [Akmandor and Jha, 2017].

for prevention and treatment of stress and stress-related health problems.

3.5 Energy-efficient Health Monitoring System

Traditional medical monitors, e.g., bedside ECG and oxygen saturation monitoring systems, gather, store, and transmit data with no (or minimal) on-device processing. Furthermore, such systems are commonly powered from the electrical outlet, as opposed to state-of-the-art IWMDs that rely on energy-constrained batteries. Recent advances in signal processing, low-power electronics, communication protocols, and, in particular, design of low-power radio-frequency transmission modules, have enabled wireless connectivity to even the most energy-constrained medical devices, allowing them to form a Wireless Body Area Network (WBAN) [Ullah et al., 2012, Ko et al., 2010]. WBAN-based continuous health monitoring systems rely on a network of medical sensors. They gather, process, and store various types of physiological data and offer a holistic approach to prevention and early detection of diseases. Indeed, continuous health monitoring systems have garnered ever-increasing attention in recent years and are envisioned as fundamental components of smart healthcare systems.

There exist several key challenges in the design and development of such systems. To maximize system acceptance and user convenience, IWMDs must be small and passive, i.e., collect physiological data with minimum user involvement. These requirements impose significant limits on the storage and battery capacities of each sensor. Nia et al. [Nia et al., 2015] comprehensively examine the energy and storage requirements of continuous personal health monitoring systems. To enable energy-efficient continuous health monitoring, they first conduct a thorough literature review of IWMDs, summarize their common resolution, sampling rate, and transmission rate, and analyze a health monitoring system consisting of eight sensors that continuously gather and transmit raw physiological data to a base station for further processing (Fig. 3.6). Table 3.4 summarizes the resolution, sampling rate, and maximum wireless data transmission rate of various sensors.

Nia et al. have developed several analytical models to characterize

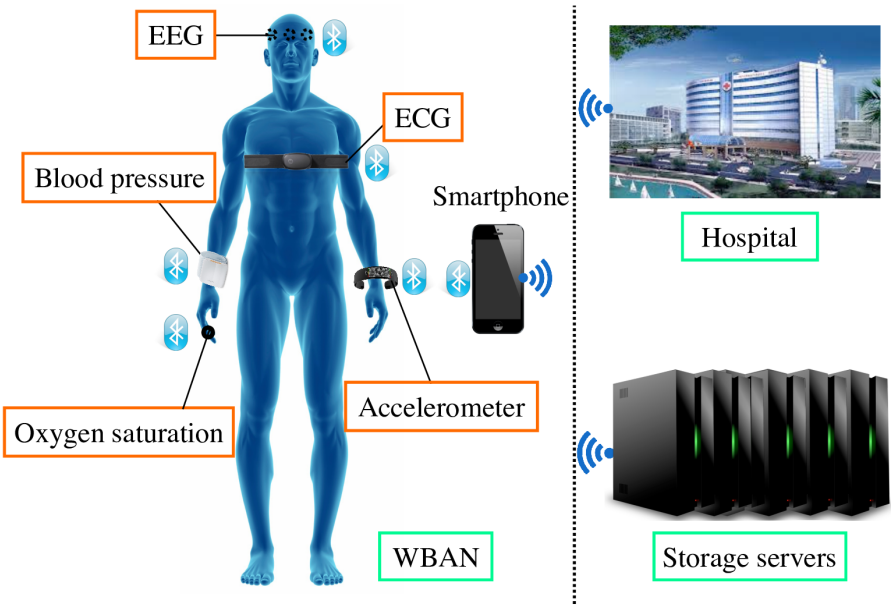


Figure 3.6: A personal healthcare system [Nia et al., 2015]. EEG: Electroencephalogram.

such a system and highlight a significant gap between the storage/energy requirements for long-term continuous monitoring and the capabilities of already-in-use IWMDs utilized in health monitoring systems. To minimize the energy consumption of each sensor, they analyze three lightweight on-sensor computation techniques, namely, sample aggregation, anomaly-driven transmission, and Compressive Sensing (CS).

Table 3.5 summarizes key characteristics of all schemes discussed in [Nia et al., 2015]. They compare their proposed schemes with a baseline scenario in which all sensor are continuously collecting and transmitting the data to a base station without any on-sensor computation. As discussed in [Nia et al., 2015], in addition to the storage and battery capacity of a sensor, latency and extensibility requirements are among the key considerations for choosing the suitable computation/transmission approach for each sensor. Next, we describe

Table 3.4: Resolution, sampling rate, and maximum transmission rate [Nia et al., 2015]

Sensor	Resolution (bits/sample)	Sampling rate (Hz)	Transmission rate (bits/s)
Heart rate	10	2-8	80
Blood pressure	16	0.001-100	1600
Oxygen saturation	8	0.001-2	16
Temperature	8	0.001-1	8
Blood sugar	16	0.001-100	1600
Accelerometer	12	2-400	4800
ECG	12	100-1000	12000
EEG	12	100-1000	12000

Table 3.5: Comparison of different schemes [Nia et al., 2015]

Scheme	Energy	Storage	Latency	Extensibility
Baseline	Very high	High	Low	High
Aggregation	Very high	High	Varies	High
Anomaly-driven	Low	Low	Low	Low
CS-based	Very low	Low	Low	Low

these two requirements in further detail.

Latency: In [Nia et al., 2015], latency is defined as the time interval between the occurrence of a significant event (e.g., an anomaly) and the response provided by medical devices or physicians. Tolerable latency significantly depends on the application of the sensor and the patient's condition. For example, a health monitoring system that monitors a healthy subject may use sample aggregation to offer a routine medical check by collecting and sending medical information to physicians or hospitals at long intervals (e.g., once a day). In contrast, a monitoring system that is used to monitor a subject with a history of serious disease (e.g., high blood glucose) should detect any changes in the medical condition (e.g., any rapid rise in blood glucose) immediately and cannot use sample aggregation.

Extensibility: Extensibility is an essential design requirement where the implementation takes future modifications into account. High extensibility implies that applications of a biomedical sensor can be extended in the future with a minimum level of effort. In general, schemes that do not perform on-sensor computation are more extensible since they can be changed without modifying the sensor.

Takeaway: Performing on-sensor computation can significantly reduce the total energy consumption of the sensor, e.g., compressive sensing can provide up to three orders-of-magnitude improvement in energy and storage. In addition to significantly increasing the battery lifetime of IWMDs, this approach provides another key benefit: designers can take advantage of the energy saved [Nia et al., 2015] to enable the implementation of security-enhancing technologies (in particular, strong encryption schemes that are commonly avoided in IWMDs due to their significant energy overhead) on IWMDs.

4

Design Considerations

The increasing functional complexity of smart healthcare systems raises inherent design challenges: efficiency, security, accuracy, cost, responsiveness, maintainability, scalability, reliability, and fault tolerance. Due to the human-critical nature of smart healthcare systems, any shortcomings in these measures may lead to adverse consequences, ranging from system impracticality to life-threatening situations [Akmandor and Jha, 2018]. In this chapter, we summarize and explain the key design considerations of smart healthcare systems in order to: (i) pinpoint where improvements can be made to existing systems, and (ii) provide guidance for future system design. The key design considerations are:

- **Efficiency:** Smart healthcare systems have to be both energy- and storage-efficient. They rely on devices located at various positions on/in the body and in the environment to capture health information of the users. If the system frequently runs out of battery energy or requires external energy resources, the practicality of the system is negatively impacted. As an example, in some IMDs like pacemakers, a battery change requires surgery with its attendant risks [Halperin et al., 2008]. Data compression techniques (e.g., compressive sensing with direct computations

[Shoaib et al., 2015, 2014], compressed signal processing [Lu et al., 2016], etc.) are effective in promoting energy efficiency. Compression reduces the data size. This leads to lower computational energy needs. A Hierarchical Inference Model (HIM) that exploits the intrinsically sensor/edge-grouped IoT data structure (e.g., data collected from one WMS is spatially separated from data collected from another WMS) can also improve efficiency without impacting accuracy. For example, Yin et al. show that HIM can reduce run-time system transmission loads by 3.2-60.0× with classification accuracy change in the -0.4%-$+6.7\%$ range for eight inference models derived for seven IoT applications, which include stress monitoring, chemical gas classification, and the diagnosis of heart disease, renal pelvis disorder, urinary bladder disorder diagnosis, hypothyroid, and type-2 diabetes [Yin et al., Submitted]. Moreover, flexible electronics offer energy harvesting potential for wearable IoT applications. As demonstrated by Jokic and Magno [Jokic and Magno, 2017], thin-film flexible photovoltaic panels have the potential to enable continuous long-term health monitoring by harvesting energy. This technology reduces or eliminates the need for battery change or recharge. Smart healthcare systems also need to support efficient memory utilization. The collected data, parameters needed for signal processing, and inference models need significant amounts of storage. This may make inference slower and limit integration of more sensors/devices. Therefore, energy and storage requirements of smart healthcare systems need to be carefully analyzed.

- **Security:** Smart healthcare systems extract sensitive health information from the user. As a result, security is of utmost importance in such systems. However, most smart healthcare systems overlook security. Previous studies have exposed vulnerabilities of various medical devices, e.g., insulin pump [Li et al., 2011], pacemaker [Halperin et al., 2008], physiological side channels [Nia et al., 2016], etc., to security attacks. These attacks may range from a loss of privacy to life-threatening consequences. Therefore, smart healthcare systems should be designed to be resistant to possible

security attacks.

- **Accuracy:** Smart healthcare systems are called 'smart' because of their decision-making capability. This capability emanates from inference. Since smartness enables disease diagnosis, treatment decisions, therapy duration, and raising of alerts [Yin and Jha, 2017, Akmandor and Jha, 2017], accuracy of the inference has a direct impact on the health of the user. If the system provides false alerts, diagnosis, or treatment suggestions, and steers the users/physician in the wrong direction, the reliability of the system decreases. This may lead to loss of confidence in the system and hence its discontinuation. Therefore, accuracy plays a critical role in utilization, quality, and effectiveness of smart healthcare systems.

- **Cost:** With lowered cost, smart healthcare systems become more widely deployed in daily and therapeutic medical applications. This has the potential to decrease dependency on hospitals/clinics, increase early diagnosis of the medical conditions, and bend national healthcare costs downwards. Thus, the cost aspect of such systems needs to be analyzed comprehensively at the design stage. Flexible Hybrid Electronics (FHE) is a promising technology for satisfying this goal. FHE integrates flexible electronics with silicon technology. It benefits from flexible technology in terms of low-cost manufacturing and flexible substrates, while preserving computational and storage advantages of traditional silicon technology [Gupta et al., 2017, Huang et al., 2015].

- **Responsiveness:** Smartness imparts increased functionality to the healthcare systems, but at the cost of additional computational and storage resources. This may decrease the responsiveness of the system (i.e., increase its latency). Since these systems provide treatment, diagnosis, guidance, and alerts when necessary, increased latency adversely impacts system functionality. If the system cannot provide the desired response on time, adding smartness to the system becomes unreasonable and disadvantageous. Therefore, system responsiveness too needs careful analysis.

- **Maintainability:** Smart healthcare systems require periodic maintenance. This includes both software and hardware updates [Akmandor and Jha, 2018]. The system should be readily adaptable to these updates. Thus, the system need not be discarded when a minor flaw is discovered. Energy harvesting also has a positive impact on maintainability. Since thin-film flexible photovoltaic panels [Jokic and Magno, 2017] not only address the energy-efficiency challenge, but also maintainability, they are a promising candidate for wearable healthcare systems.

- **Scalability:** Advancing sensor technology enables an increasing number of sensors and devices to be integrated into smart healthcare systems. This increased functionality broadens the application scope. Thus, system design should anticipate future expansion of system capability.

- **Reliability and fault tolerance:** Smart healthcare systems collect data from various sources. During data collection, signal processing or data transmission, errors might be introduced due to faults in different parts of the system. Reliability indicates the system's ability to withstand these faults. A higher reliability makes it more likely that the smart healthcare system will be adopted. Thus, continued operation in the presence of faults is very desirable. This can be ensured through built-in fault tolerance.

5

Innovations & Trends

In this chapter, we explain five innovative research trends that may help address the design considerations of smart healthcare systems. We first explain two approaches that can lead to substantial energy and storage efficiency improvements through: (i) compact deep neural networks in Section 5.1, and (ii) compressive sensing in Section 5.2. Then, we explain three approaches that address the security concerns: (i) MedMon for wireless communication channel monitoring of WBANs in Section 5.3, (ii) OpSecure for optical key exchange in Section 5.4, and (iii) SecureVibe for secure communication via a vibration side channel in Section 5.5.

5.1 NeST: Synthesizing Compact Deep Neural Networks

Neural Networks (NNs) have begun to have a pervasive impact on various healthcare applications. Their ability to distill intelligence from very large datasets through multi-layer abstraction can lead to superior or even super-human performance, as observed in the case of CheXNet and DeepBind (see Section 2.2). However, conventional NNs consume extensive memory and computation energy. As a result, most healthcare-

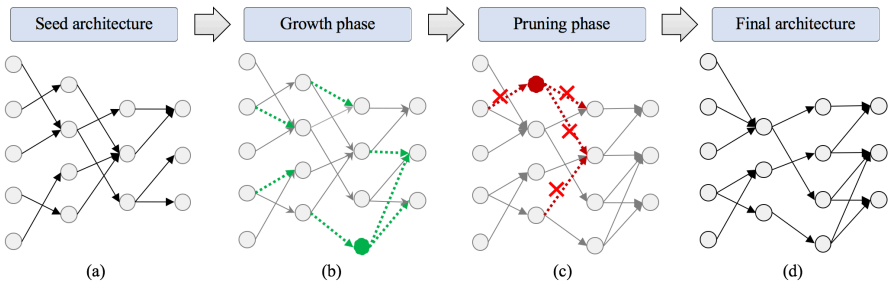


Figure 5.1: An illustration of the architecture synthesis flow in NeST [Dai et al., 2017].

oriented deep NNs are still confined to the clinical Cloud, and do not find place in mobile phones that can be used on a daily basis.

The problem of finding an accurate yet compact NN (or lightweight NN) for large applications has remained open for several decades. Conventional approaches search for optimal NNs through extensive trial-and-error. Such approaches are extremely inefficient. For a deep NN with millions of parameters, each training trial can easily consume tens or hundreds of hours even with the fastest GPUs. In addition, the generated NNs still suffer from substantial redundancy. For example, Han et al. show that without any accuracy degradation, the number of parameters in the well-known AlexNet [Krizhevsky et al., 2012] NN architecture for the ImageNet dataset [Deng et al., 2009] can be reduced by 9× [Han et al., 2015].

To address these problems, Dai et al. propose an NN Synthesis Tool (NeST) that can automatically generate accurate yet extremely compact NNs, given a target application dataset [Dai et al., 2017]. NeST can dramatically cut down on the memory cost, inference run-time, and energy consumption of deep NNs. It has the potential to enable incorporation of deep NNs on mobile phones, thus expanding the reach of such NNs from the clinical domain to daily healthcare.

The NeST methodology is depicted in Fig. 5.1. It starts with a very sparse seed NN architecture. It iteratively tunes this architecture with: (i) gradient-based growth and (ii) magnitude-based pruning of neurons and connections. The major techniques involved in these two phases

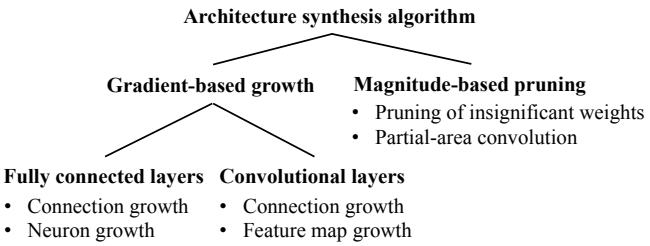


Figure 5.2: Major components of the NN architecture synthesis algorithm in NeST [Dai et al., 2017].

are summarized in Fig. 5.2. The growth phase utilizes the gradient information to add new connections, neurons, and feature maps. This allows the NN to easily adapt to the problem at hand. The pruning phase removes redundant connections and neurons. This drastically reduces the number of NN parameters, thus memory, and Floating-point Operations (FLOPs) per inference, thus computation cost. Finally, NeST yields accurate, yet very compact, NNs.

Dai et al. used NeST to synthesize compact NNs for the MNIST dataset[1] based on hints from LeNet-300-100 and LeNet-5 architectures [Lecun et al., 1998], and for the ImageNet dataset[2] based on hints from the AlexNet architecture. NeST delivers extremely compact NNs with the same or improved accuracies relative to the corresponding NN baselines:

- For LeNet-300-100, NeST reduces network parameters by 70.2× and FLOPs by 79.4×.
- For LeNet-5, NeST reduces the network parameters by 74.3× and FLOPs by 43.7×.
- For AlexNet, NeST reduces network parameters by 15.7× and FLOPs by 4.6×.

All these results constitute the current state-of-the-art [Dai et al., 2017].

[1] 28×28 handwritten digits, 60K instances for training and 10K for validation.

[2] ILSVRC-2012 image classification dataset, 1.2 million instances for training and 50K for validation.

5.2 Compressive Sensing: Reducing Computation Loads

Compressive sensing is another technique that can reduce energy and storage requirements of smart healthcare systems through computation load reduction. Compressive sensing is applicable when the data are sparse in a secondary basis and there is incoherence between the secondary basis and the random projection matrix it utilizes [Donoho, 2006, Candes and Tao, 2006]. The random projection matrix generally satisfies this incoherence property if its elements are sampled from the $\{+1, -1\}$ set that is uniformly distributed [Candes and Tao, 2006, Shoaib et al., 2015].

Random projection is a single matrix multiplication operation. Thus, it incurs little computational cost [Shoaib et al., 2015]. However, for traditional Nyquist-domain signal processing, the compressed signal needs to be reconstructed to enable machine learning inference on the user side. Reconstruction is very energy-intensive (requiring three-to-four orders of magnitude more energy than compression) due to the computation load required for the convex optimization problem that needs to be solved [Shoaib et al., 2015]. Since energy-intensive operations are not compatible with many smart healthcare systems because of the need for frequent battery recharge [Akmandor and Jha, 2018], Shoaib et al. and Lu et al. propose techniques that **do not** require signal reconstruction in the inference stage. These techniques are based on direct computations on compressively-sensed data [Shoaib et al., 2015, 2014] and Compressed Signal Processing (CSP) [Lu et al., 2016], respectively.

Direct computations on compressively-sensed data requires the derivation of compressed-domain signal processing operations. These operations enable both feature extraction and classification to be performed in the compressed domain, thus reducing energy consumption dramatically.

CSP carries out signal processing operations in the Nyquist domain, then performs a random projection. This minimizes the inner product error in the compressed domain relative to the Nyquist domain. This improves classification accuracy.

We discuss these methods in detail next.

5.2.1 Direct Computations on Compressively-sensed Data

In the Nyquist domain, the feature vector y corresponding to data epoch x is obtained using Eq. 5.1 based on a linear signal processing matrix $\mathbf{H}$ [Shoaib et al., 2015]. In the compressed domain, as the data are projected randomly with the help of a random projection matrix $\mathbf{\Phi}$, Shoaib et al. derive a compressed-domain equivalent, $\hat{\mathbf{H}}$, of matrix $\mathbf{H}$. $\hat{\mathbf{H}}$ can be used to obtain the feature vector $\hat{y}$ (Eq. 5.2). Ideally, as shown in Eq. 5.3, $\hat{y}$ and y should be equal to preserve inference performance.

$$y = \mathbf{H} \cdot x \tag{5.1}$$

$$\hat{y} = \hat{\mathbf{H}} \cdot \mathbf{\Phi} \cdot x \tag{5.2}$$

$$y = \hat{y} \Rightarrow \mathbf{H} \cdot x = \hat{\mathbf{H}} \cdot \mathbf{\Phi} \cdot x \Rightarrow \mathbf{H} = \hat{\mathbf{H}} \cdot \mathbf{\Phi} \tag{5.3}$$

For an N-dimensional input x, $\mathbf{H}$ is an $N \times N$ matrix, $\mathbf{\Phi}$ is an $M \times N$ matrix, where $M \ll N$, and $\hat{\mathbf{H}}$ is an $N \times M$ matrix. To derive $\hat{\mathbf{H}}$, Eq. 5.3 specifies $N \times M$ variables and $N \times N$ equations, thus leading to an overdetermined system [Shoaib et al., 2015]. To get around this problem, Shoaib et al. introduce a regularization term $\mathbf{\Theta}$. This transforms Eq. 5.3 to Eq. 5.4.

$$\mathbf{\Theta} \cdot y = \hat{y} \Rightarrow \mathbf{\Theta} \cdot \mathbf{H} \cdot x = \hat{\mathbf{H}} \cdot \mathbf{\Phi} \cdot x \Rightarrow \mathbf{\Theta} \cdot \mathbf{H} = \hat{\mathbf{H}} \cdot \mathbf{\Phi} \tag{5.4}$$

Shoaib et al. obtain solutions to Eq. 5.4 for both the square $\mathbf{H}$ and non-square $\mathbf{H}$ cases. When $\mathbf{H}$ is a square matrix, the solution can be exact or approximate (an approximate solution trades further energy efficiency for a slight degradation in classification accuracy). Algorithm 2 shows how to obtain $\hat{\mathbf{H}}$ for these cases.

Shoaib et al. apply their proposed technique to neural prosthesis spike sorting and EEG seizure detection. In the case of neural prosthesis, even with 54× fewer samples, the technique achieves system

Algorithm 2 Computation of the compressed-domain signal processing matrix $\hat{\mathbf{H}}$ [Shoaib et al., 2015]

Require: projection dimension K and matrices $\mathbf{\Phi}$ and $\mathbf{H}$
Ensure: $\mathbf{\Theta}$ and $\hat{\mathbf{H}}$ with $\mathbf{\Theta H} = \hat{\mathbf{H}}\mathbf{\Phi}$

Init: $N \leftarrow$ # cols($\mathbf{\Phi}$); $M \leftarrow$ # rows($\mathbf{\Phi}$); $L \leftarrow$ # rows($\mathbf{H}$)
if $L = N$ **then**
  $\mathbf{D^T} := \mathbf{\Phi H^{-1}}$; $\mathbf{USV^T} \leftarrow$ SVD($\mathbf{D}$); ▷ for $\theta_\mathbf{i} = \mathbf{D}\hat{\mathbf{h}}_\mathbf{i}$
  if $K = M$ **then**
    $\hat{\mathbf{H}} = \sqrt{(N/M)}\,(\mathbf{S^{-1}V^T})$; $\mathbf{\Theta} = \sqrt{(N/M)}\left(\hat{\mathbf{H}}\mathbf{\Phi H^{-1}}\right)$;
  else
    for $i = 1$ to K **do**
      $\mathbf{x_i} \sim N(0, \mathbf{I_M})/\sqrt{(K)}$; ▷ for $\hat{\mathbf{h}}_\mathbf{i} \sim N\left(0, \mathbf{VS^{-2}V^T}\right)$
      $\hat{\mathbf{h}}_\mathbf{i} = \mathbf{VS^{-1}x_i}$; $\theta_\mathbf{i} = \mathbf{Ux_i}$;
    end for
    $\mathbf{\Theta}$=$\sqrt{(N/M)}\left(\theta_1^T; \ldots; \theta_K^T\right)$; $\hat{\mathbf{H}}$=$\sqrt{(N/M)}\left(\hat{h}_1^T; \ldots; \hat{h}_K^T\right)$;
  end if
else
  $\mathbf{PQR^T} \leftarrow$ SVD($\mathbf{H}$); $\mathbf{VSU^T} \leftarrow$ SVD($\mathbf{\Phi}$);
  $\mathbf{\Theta} \sim N\,(0,1)\,/\sqrt{(NK/M)}$; ▷ ortho($\mathbf{\Theta}$) if $K > L$
  $\mathbf{B} = \mathbf{\Theta PQ}$; $\mathbf{A} = \mathbf{BR^TU}$; $\hat{\mathbf{H}} = \sqrt{N/M}\,(\mathbf{AS^{-1}V^T})$;
end if

performance comparable to Nyquist-domain processing. The accuracies corresponding to Spike Count (SC), neuron Firing-Rate (FR) estimation, and Coefficient of Variation (CV), which are the main system-level metrics in this application, are 98.63%, 98.56%, and 96.51% for the compressed domain as opposed to 98.97%, 99.69%, and 97.09% for the Nyquist domain, respectively. In the case of EEG based seizure detection, they need 21× fewer samples and achieve 94.43% sensitivity, 4.70s latency, and 0.1543 false alarms/h in the compressed domain. This result is comparable to 96.03% sensitivity, 4.59s latency, and 0.1471 false alarms/h in the Nyquist domain.

This method can significantly reduce both energy consumption and storage capacity in healthcare applications. For example, Nia et

al. [Nia et al., 2015] show that this method can offer up to 724× energy consumption reduction for ECG sensors for arrhythmia detection. It achieves storage savings of up to 19344× when EEG sensors are used for seizure detection.

5.2.2 Compressed Signal Processing

CSP [Lu et al., 2016] performs signal processing and feature extraction in the Nyquist domain. After the computation of the feature vector, CSP uses random projections to improve energy efficiency. As shown in Eq. 5.5, the $\tilde{\mathbf{H}}$ matrix provides this transformation and outputs the feature vector $\tilde{y}$. Lu et al. aim to minimize the difference between $\tilde{y}$ and y in order to preserve inference performance.

$$\tilde{y} = \tilde{\mathbf{H}} \cdot x \tag{5.5}$$

Lu et al. also evaluate their technique on neural prosthesis spike sorting and EEG seizure detection. In the case of neural prosthesis spike sorting, in the Nyquist domain, the average errors for SC, FR, and CV are found to be 4.00%, 4.00%, and 2.75%, respectively. In the compressed domain, they obtain average errors of 4.89% for SC, 4.90% for FR, and 3.42% for CV with 32× fewer samples. In the case of EEG seizure detection, compared to the Nyquist domain (100% sensitivity, 4.37s latency, and 0.12 false alarms/h), they obtain 100% sensitivity, 4.33s latency, and 0.22 false alarms/h with 32× fewer samples.

CSP provides comparable inference performance while compressing the data by 32×. This can be exploited for drastic energy and storage reductions, thus providing significant benefits to resource-constrained healthcare applications.

5.3 MedMon: Defending Against Wireless Attacks

Diabetes mellitus is one of the most important public health challenges of the 21st century. In 2015, more than 30.3M people in the U.S. lived

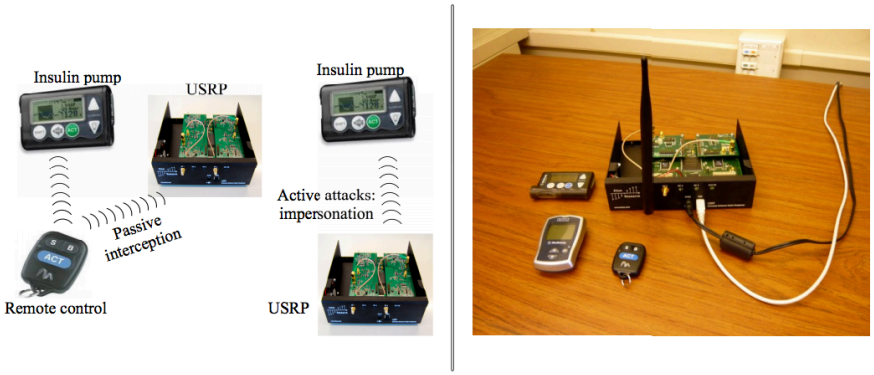


Figure 5.3: Security attacks and the experimental setup [Li et al., 2011].

with diabetes (9.4% of the population)[3]. Diabetic patients rely on effective monitoring and response systems, such as continuous glucose monitoring and insulin delivery, to avoid hyperglycemia (high blood glucose level) or hypoglycemia (low blood glucose level). These systems are the state-of-the-art for diabetes management, and have thus become increasingly popular among diabetic patients.

However, continuous glucose monitoring and insulin delivery systems have security flaws. For example, Li et al. successfully launched security attacks on a popular glucose monitoring and insulin delivery system [Li et al., 2011]. The attacks exploited unencrypted wireless communication channels that enable the glucose sensor, glucose meter, insulin pump, and a remote control to communicate with each other. As a result, the safety and privacy of users can be easily undermined.

Li et al. use a widely available off-the-shelf Universal Software Radio Peripheral (USRP) to launch the attacks, as shown in Fig. 5.3. A USRP can intercept radio communications within a target frequency band and generate wireless signals in that target band at different power with various modulation schemes. Their attacks can be categorized into two major groups:

[3]National Diabetes Statistics Report 2017, https://www.cdc.gov/diabetes/pdfs/data/statistics/national-diabetes-statistics-report.pdf.

- Passive attack: eavesdropping on the wireless communication. Li et al. use a USRP to intercept communications in the 915 MHz communication band between the remote control and insulin pump, as shown in Fig. 5.3. They first identify the channel modulation scheme as the on-off keying scheme (a presence of carrier indicates 1, otherwise 0). This allows them to decipher the 80-bit communication packets through reverse engineering: 4-bit device type indicator, 36-bit device PIN, 12-bit payload information, 12-bit system counter, 12-bit cyclic redundancy check string, and a constant 4-bit string '0101' at the end. The 12-bit payload string leaks patient information such as the existence of therapy and the glucose level, thus completely breaching user privacy.

  Li et al. show that a passive attack can be easily launched when the USRP is within a 7-8 meter range of the insulin pump.

- Active attack: impersonation and control of the insulin pump to alter therapy. After deciphering the 80-bit data packets, Li et al. generate legitimate-looking packets, which pass all the checks and are, hence, accepted by the insulin pump. They use the USRP to transmit these maliciously crafted packets to the insulin pump. These packets contain malicious commands, such as stop an insulin injection or inject a much higher or lower dose. This may lead to severe adverse consequences, such as hyperglycemia or hypoglycemia, thus endangering patient's life.

  Li et al. show that an active attack can be successfully launched even when the USRP is 20 meters away from the insulin pump. This is even farther than the original control range of the remote controller: 4.5 meters.

Li et al. propose two countermeasures against these attacks. The first method uses a cryptographic approach based on a pair of rolling code encoders embedded in both the remote control and insulin pump. The encryption mechanism prevents an attacker from accessing device PINs and data payloads, thus guards against both attacks. The second method transmits the command signal via the human body instead of through a wireless communication channel. This prevents attackers from

accessing the packets as long as they are not in very close proximity to the patient or touching the patient's skin. Both methods incur additional battery energy on the insulin pump.

Zhang et al. propose a medical security monitor called MedMon to detect wireless attacks on a WBAN with zero additional power on biomedical devices [Zhang et al., 2013]. MedMon is a non-invasive external wearable device that monitors the transmission traffic in and from medical devices. Whenever an anomaly is detected, MedMon raises an alert, and jams the malicious transaction before it alters the state of the target device. MedMon does not require any modifications to existing hardware or software, and is thus applicable to legacy healthcare systems.

MedMon captures two major types of anomalies in the transmission traffic within a WBAN:

- Physical anomalies: though carefully crafted, malicious transmissions may deviate significantly from legitimate ones in their physical signal characteristics. These characteristics include Received Signal Strength Indicator (RSSI), Time of Arrival (TOA), Differential Time of Arrival (DTOA), and Angle of Arrival (AOA). This allows MedMon to determine the relative position (angle and distance) of the signal transmitter to the medical device. This enables MedMon to verify the legitimacy of the transmitter, and hence the transmission, with very high confidence. To avoid false alarms, MedMon utilizes multiple threshold values for each of its target signal characteristics. This accommodates the unfixed relative positions of medical devices in a WBAN. The threshold values are configured when MedMon is calibrated and trained to learn normal behaviors of a target WBAN prior to its deployment for anomaly monitoring.

- Behavioral anomalies: carefully crafted malicious transmissions may be physically indistinguishable from a legitimate one, but contain different underlying information (command or data) to cause harm to the patients. These anomalies are referred to as behavioral anomalies. For example, an adversary can forge a

Figure 5.4: Upon identifying an attack, MedMon (a) just provides a warning in the passive mode, and (b) provides a warning and jams the communication in the active mode [Zhang et al., 2013]

> legitimate-looking packet and send it to the insulin pump to trigger large insulin doses. Such sophisticated attacks may in some cases be able to bypass the physical anomaly checking process. To defend against such attacks, MedMon compares the content of a newly arrived command/data with pre-stored legitimate historical records to determine its legitimacy.

MedMon checks physical and behavioral anomalies in a sequential manner. An incoming transmission is only deemed to be safe if no anomaly is detected in both steps, and only then granted access to the medical device. In response to a detected anomaly, MedMon can either raise a warning (passive defense mode) to the user or jam the transmission channel (active defense mode), as shown in Fig. 5.4.

Zhang et al. implement MedMon to defend against both the passive and active attacks on insulin pumps [Li et al., 2011, Zhang et al., 2013]. They use one USRP to launch the attack and another USRP to function as MedMon. The transmission band is centered at 916.68 MHz, while the jamming signal is centered at 916.87MHz. The attacker USRP injects malicious packets in the wireless channel, while MedMon tries to determine the command type, verify the device PIN, and jam the

communication channel, just in time, whenever an anomaly is detected.

MedMon achieves a 99.2% detection rate (0.8% false negative rate) over 250 attacks launched at random times from varying locations and at varying power levels. It achieves a 100.0% detection rate (0% false negative rate) when within 10-20 cm from the insulin pump in over 100 attacks. In all these attacks, MedMon successfully jams the channel before malicious packets alter the state of the insulin pump.

Takeaway: MedMon addresses security issues associated with the use of several already-in-market medical devices and sensors utilized in WBANs. Some companies may consider adding built-in security features to their future designs. However, at the current state of the technology, many in-market medical devices and sensors do not come with strong security solutions due to the cost, energy, and storage overheads of such solutions. MedMon can bring add-on security to non-secure medical devices while imposing no design change and energy/cost overheads.

5.4 OpSecure: Exchanging Keys via Light

To mitigate the risk of eavesdropping on wireless channels used to communicate with IMDs, the use of lightweight data encryption techniques has been suggested [Hu et al., 2013, Zhang et al., 2014]. However, due to limited on-IMD storage/energy resources, traditional encryption schemes cannot be implemented on IMDs. In particular, asymmetric encryption mechanisms are not suitable for IMDs since they would significantly degrade battery lifetime [Potlapally et al., 2006, Hu et al., 2013]. Several recent studies have proposed lightweight symmetric encryption mechanisms to prevent eavesdropping on IMDs and enhance the security of communication protocols utilized in such resource-constrained devices ([Strydis et al., 2008] summarize several such mechanisms). Symmetric cryptography requires a secret key that is shared between the two parties involved in the communication. Thus, symmetric key encryption can be utilized in IMDs only if a secure key exchange protocol is available.

Previous studies indicate that, to prevent both battery-draining and remote eavesdropping attacks, attack-resilient wakeup and key exchange protocols are desirable. Mosenia and Jha [Mosenia and Jha,

2017] have proposed attack-resilient wakeup and key exchange protocols for subcutaneous IMDs. They have developed practical ***key exchange and wakeup protocols based on visible light***. These protocols complement lightweight symmetric encryption mechanisms and prevent remote battery-draining attacks and security threats against insecure communication channels. They present a new communication channel for IMDs, called OpSecure, inspired by the observation that visible light can penetrate deep enough into the body to reach the IMD when the light source is in contact with the human body. However, the challenge is that visible light attenuates very fast in the body. Indeed, the proposed solution is intrinsically secure due to the proximity requirements imposed by the physical characteristics of visible light. OpSecure relies on two components: (i) a light source embedded in an external device (for example, a smartphone's flashlight) that modulates visible light to transmit data, and (ii) a light sensor in the IMD that can sense the visible light generated by the light source. They implement their protocols using the flashlight of smartphones. They implement an Android application on the smartphone to support two functionalities: waking up the IMD and transmitting a randomly-generated key (Fig. 5.5), as described next.

Wakeup protocol: To wake up the IMD, an authorized user must place the smartphone on the patient's body close to the IMD, e.g., on the patient's chest when the user wants to wake up a pacemaker, and press the wakeup button in the application. In this scenario, the wakeup button simply turns on the flashlight. The IMD periodically wakes up to check if a light source is on the body, i.e., it checks if the intensity of the light received by the light sensor embedded in the IMD is above a predefined threshold T. The presence of an on-body light source pointed at the IMD is interpreted as the presence of a trusted external device.

Key exchange protocol: To exchange a key with the IMD, an authorized user creates a random key, places the smartphone's flashlight on the patient's body, and presses the key exchange button. The smartphone exchanges the key with the IMD in four steps:
Step 1: The smartphone prepares a key packet as $Key_{packet} = Pre||K||Post$, where K is the randomly-generated key and Pre and

Figure 5.5: The smartphone generates a 4-bit key and transmits the key over OpSecure. The application allows the user to control both the key length (N) and transmission rate (R) [Mosenia and Jha, 2017].

Post are two predefined binary sequences that specify the beginning and end of the key.

Step 2: The smartphone uses on-off keying modulation to transmit Key_{packet}. To transmit bit "1", the smartphone turns the flashlight on for a predefined period. Similarly, it turns the flashlight off for the same period if a bit "0" is to be transmitted.

Step 3: The IMD recovers Key_{packet} by demodulating the light received at the light sensor. It then removes Pre and $Post$ from the packet and recovers the key K. It then creates and sends an acknowledgment packet to the smartphone. It encrypts a predefined message $M_{confirm}$ using K, encrypts the message with K, and transmits the encrypted message $C = ENC(M_{confirm}, K)$ to the phone over a Radio-Frequency (RF) communication channel (both the IMD and smartphone are assumed to support a bidirectional RF communication protocol).

Step 4: The smartphone checks if it can successfully decrypt the acknowledgment message. If so, it concludes that the key has been successfully transferred to the IMD.

In [Mosenia and Jha, 2017], Mosenia and Jha describe an IMD prototype that supports the above-mentioned protocols and use a beef-bacon body model, i.e., a human body model that consists of a thin layer of bacon on a thick layer of lean ground beef, to evaluate them in several realistic scenarios. In particular, they demonstrate that OpSecure adds minimal size/energy overheads to an IMD while significantly enhancing the security of the device. They show that OpSecure is robust against environmental noise, e.g., ambient light, and resilient against a variety of remote attacks, e.g., using a coherent laser to try to inject a malicious key. Their empirical results suggest that OpSecure can significantly enhance the security of subcutaneous IMDs in real-world scenarios, as opposed to prior approaches, such as acoustic-based [Halperin et al., 2008] communication channels that are vulnerable to remote eavesdropping and can be negatively affected by environmental noise [Halevi and Saxena, 2013].

5.5 SecureVibe: Exploiting the Vibration Side Channel

Kim et al. [Kim et al., 2015] introduce alternative vibration based protocols for RF module wakeup and key exchange, called SecureVibe. Fig. 5.6 shows its two-stage wakeup procedure. In the first stage, at regular time instances, the accelerometer enters the Motion-Activated Wakeup (MAW) mode. In this mode, the vibration is compared with a threshold to distinguish between actual vibration and body motion. In the case of body motion, since the previously set threshold has a higher value, the accelerometer does not become activated. Thus, the module shifts to the standby mode. In the case of vibration, the threshold is exceeded and the accelerometer takes measurements. In the measurement mode, to eliminate low-frequency components induced by the environment or patient movements, a high-pass filter is used. After filtering the accelerometer measurements, if there is vibration, the RF module is activated, otherwise it is returned to the standby mode.

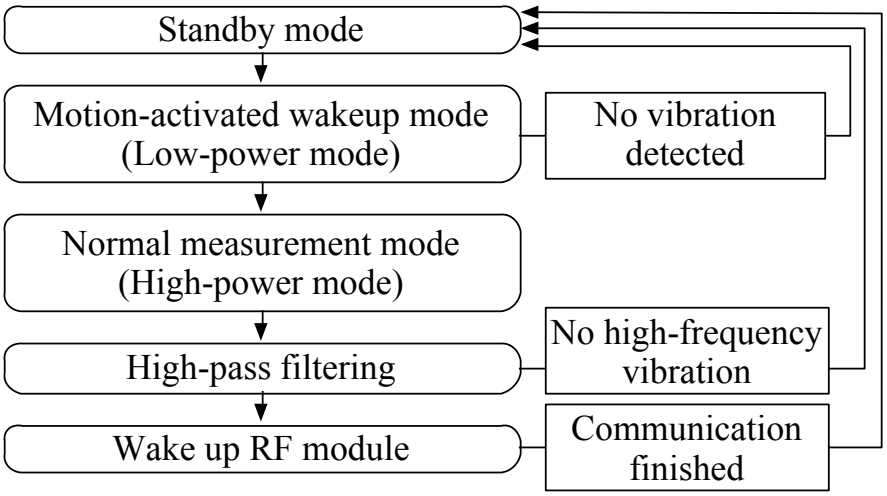


Figure 5.6: Block diagram of the two-stage wakeup of the RF module in SecureVibe [Kim et al., 2015].

SecureVibe uses the key exchange protocol shown in Fig. 5.7. As a first step, the External Device (ED) generates a random key. The medical device receives this key in the form of a vibration. After transforming the vibration to a bit string, the medical device encrypts a previously-defined message and sends it to the external device. After decrypting the received ciphertext, if the external device obtains the message correctly, the key received by the medical device is confirmed and the communication is carried out with the verified key. However, due to noise or other influences in the communication environment, when vibration is converted into bits, some bits may remain ambiguous. To address this problem, in the case of a small number of ambiguous bits, Kim et al. use random guessing. After guessing the random bits, the medical device encrypts the predefined fixed message and sends it to the external device along with the positions of the ambiguous bits. Using these positions, the external device searches through all possible combinations of bits that can be used to decrypt the ciphertext correctly. If the external device finds such a key, communication is established and the messages are encrypted with the verified key. However, if there are a large number

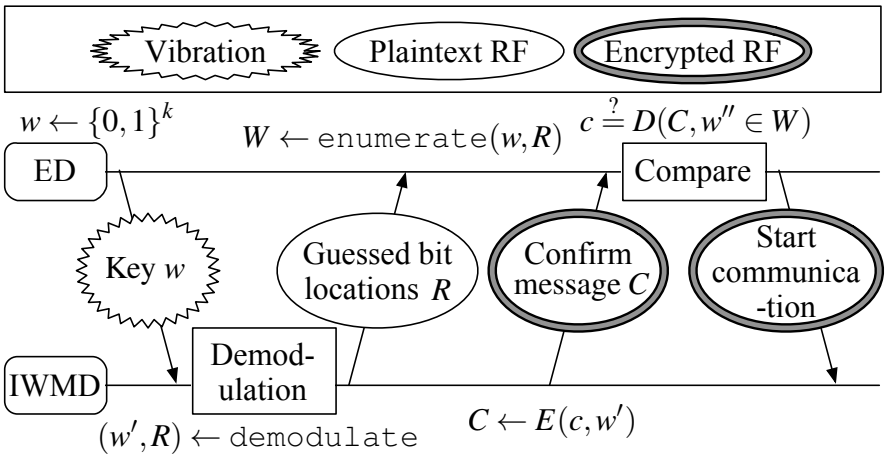


Figure 5.7: Key exchange protocol of SecureVibe [Kim et al., 2015].

of ambiguous bits, the procedure is repeated with a different random key.

To evaluate the wakeup vibration detection and key exchange protocols, Kim et al. include the effect of body movement. Fig. 5.8 shows the wakeup vibration and corresponding modes of the wakeup module. In the beginning, since no vibration is detected, the accelerometer stays in the standby mode. When the vibration exceeds the threshold, the accelerometer passes to the normal measurement mode. The normal measurement mode activates the high-pass filter. After de-noising the measured signal, since there is no vibration, the accelerometer enters the standby mode again. However, when there is wakeup vibration, even after high-pass filtering, the threshold is exceeded and the RF module is activated (see the final MAW span in Fig. 5.8). Fig. 5.9 shows the vibration waveform, amplitude gradient, and amplitude mean for the key exchange experiment that is carried out at 20 bits/second with a 32-bit key. As indicated by the orange triangle, the ninth bit is ambiguous and requires a random guess. However, once the external device obtains the position of the ambiguous bit, only two trials are required to find the key.

Kim et al. also analyze the resistance of SecureVibe to security

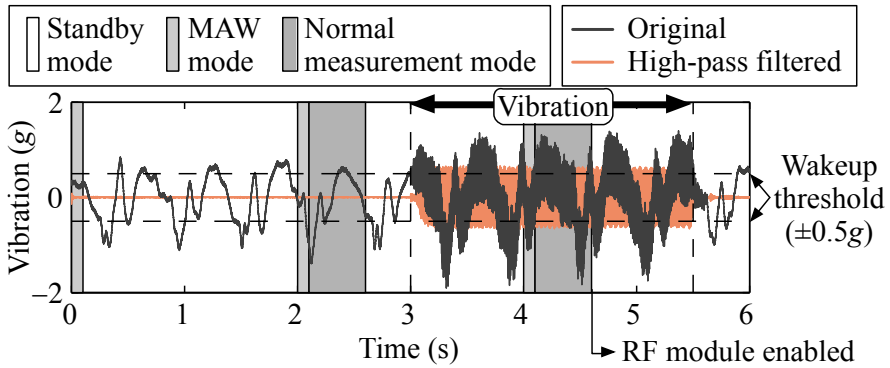


Figure 5.8: Wakeup vibration and various modes of the two-step RF wakeup module [Kim et al., 2015].

attacks. In the presence of direct attacks, they show that key exchange can only be done at most 10 cm away from the external device. Due to this short-range requirement, the attacks become easily recognizable by the patient. They also evaluate resistance to an acoustic eavesdropping attack. Due to the masking sound, eavesdropping on the vibration sound does not provide any valuable information about the key. In the case of differential attacks, again no valuable information can be obtained due to the short distance between the two sound sources. Therefore, the practicality and strength of the proposed system are verified in various ways.

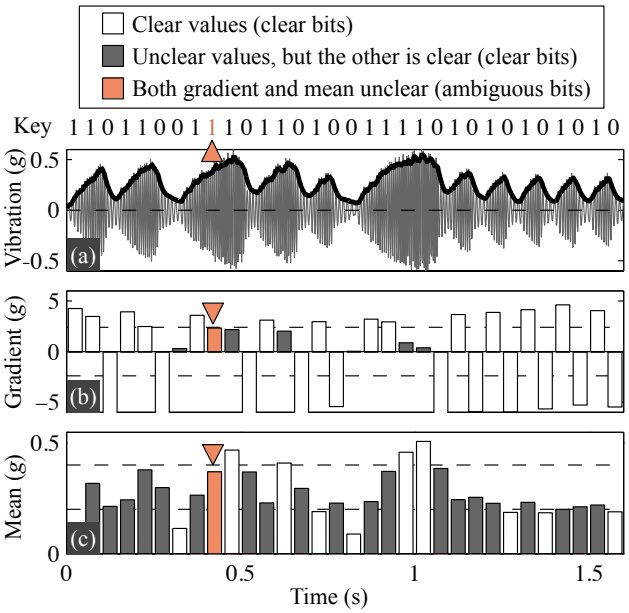


Figure 5.9: (a) Vibration waveform, (b) amplitude gradient, and (c) amplitude mean for the key exchange experiment at 20bps and with a 32-bit key [Kim et al., 2015].

6

Looking Forward

Despite remarkable progress, smart healthcare is still an emerging and booming field where a wide range of open research opportunities are available.

6.1 Unsatisfactory Datasets and Machine Learning Models

Healthcare data can be, or in most cases are, noisy, unstructured, time-correlated, large in volume, varying in amplitudes, packed with missing values, and most importantly, without proper labels. These shortcomings may prevent existing machine learning algorithms from delivering high classification accuracy. For example, SVM, tree-based algorithms, and Bayesian networks fail to scale well when the feature dimension increases for more challenging tasks, such as biomedical image classification and EHR analysis (NNs dominate in these fields). Deep NNs, though achieving astonishing accuracies, still have to rely on an immense well-labeled training dataset. This incurs substantial training time. Effective unsupervised learning methods, on the other hand, do not require labels but are not yet mature.

Currently, a lack of usable biomedical datasets still acts as a big

bottleneck to further advances. For most target applications, a comprehensive, clean (i.e., structured and pre-processed), well maintained, and preferably correctly labeled dataset does not exist. For example, biomedical datasets that may enable disease diagnosis through machine learning do not currently exist for most diseases or even disease categories. As shown in Fig 3.2 (see Section 3.3), a huge fraction of ICD-10-CM disease categories is still unexplored in terms of available biomedical datasets. Collection of these datasets can yield enormous social benefits. For example, once an HDSS disease diagnosis module is generated from a disease dataset, it can be used by anyone anywhere anytime for pervasive disease diagnosis.

6.2 Protocol Standardization and Infrastructure Support

Standardized protocols are needed to facilitate a smooth transition between and synchronization of different smart healthcare resources. Smart healthcare systems typically consist of various treatment/monitoring protocols and heterogeneous sensors/devices for both personal and clinical utilization. They need to perform smart healthcare ***tasks*** and work collaboratively in the smart healthcare ***loop*** to improve the health condition of the user. For example, in the case of an emergency, personal smart healthcare systems should communicate efficiently with clinical healthcare systems. On the other hand, when the emergency subsides, clinical follow-up and treatment/rehabilitation should be compatible with personal smart healthcare systems. Therefore, smooth transition and synchronization based on standard protocols are necessary for proper functioning of personal and clinical smart healthcare resources.

In all these scenarios, handling of health data requires utmost care since the data unveil sensitive user information. Hence, it imposes a high bar on the infrastructure support system to facilitate secure health data collection and storage (to make smartness possible and hold its outcome), and data transfer/sharing (to allow the full exploitation of smartness) within/among health organizations, hospitals, research institutes, data centers, and edge devices. Infrastructural deficiencies can place a roadblock in the development and utilization of innovative

healthcare solutions.

6.3 Fog Computing as an Alternative to the Cloud

Cloud computing, with its adaptive computational power and scalable storage, has tremendously empowered healthcare applications. Despite its obvious benefits, its applicability is limited in many healthcare applications, in particular, when the application is mission-critical, data-dominant, or latency-sensitive. Even a short period of Cloud unavailability, which may be a result of a failure in Cloud servers or loss of Internet connectivity, may be life-threatening in some healthcare services. Furthermore, healthcare applications, e.g., seizure or arrhythmia detection, may need to capture and process a huge amount of data every day. For such data-dominant applications, sending the data to the Cloud is not cost-efficient, especially, if the application relies on cellular Internet connectivity [Mosenia et al., 2017a]. Moreover, the round-trip delay caused by sending the raw data to the Cloud, making inferences, and sending the inferences back to the user side are not tolerable in applications that require a fast response [Mosenia et al., 2017b].

To address the above-mentioned shortcomings of Cloud computing, new computing/networking paradigms, in particular, Fog computing, have recently emerged. They push scalable computational/storage power to the edge of the network. Fog computing has a distributed horizontal architecture that exploits computational, networking, and storage resources along the edge-to-Cloud continuum [Dastjerdi and Buyya, 2016]: it utilizes both edge-side and Cloud resources, along with other resources available in computational/networking nodes located between the edge and the Cloud commonly referred to as *Fog nodes* (Fig. 6.1).

In healthcare, Fog-based applications offer four main advantages over traditional Cloud-based services:

- Low latency: Fog-based applications take advantage of several close-to-the-user resources. This minimizes the round-trip delay overhead imposed by transmitting data to remote resources and enables the implementation of a variety of low-latency services with minimal reliance on remote on-Cloud resources.

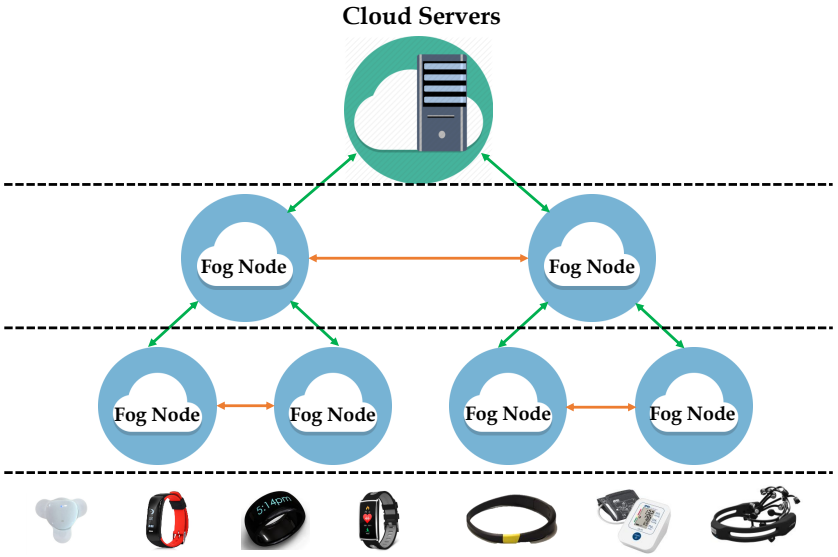


Figure 6.1: Fog computing exploits computational, networking, and storage resources along the edge-to-Cloud continuum. Each Fog node can talk to other nodes at the same layer and/or other layers. Commonly, Fog nodes located closer to the Cloud have more computational power and storage capacity.

- Cost efficiency: Fog computing minimizes the need of Cloud services, at the same time, significantly reduces the edge-to-Cloud data transmission overhead by exploiting close-to-the-user resources.

- Privacy: Close-to-the-user resources enable the processing of sensitive raw data before sharing them with third-party servers, thus significantly enhancing patient privacy. For example, noise can be added to raw data to hide private information or the privacy-sensitive portions of the data can be filtered [Mosenia et al., 2017a, Zao et al., 2017].

- Resilience to failures: In smart healthcare, availability is an essential consideration in the design and implementation of mission-critical applications. Relying on distributed resources along the edge-to-Cloud continuum enables fast recovery of services in the

event of failures. In the Fog computing paradigm, critical tasks can be reallocated to local resources upon the detection of a failure [Zao et al., 2017].

A few recent research studies have shed light on the above-mentioned advantages of Fog computing in real-world healthcare applications. For instance, Cao et al. [Cao et al., 2015] presented a Fog-enabled system to detect, predict, and prevent falls by stroke patients. It offers lower energy consumption and faster response time compared to its Cloud-based alternative. Stantchev et al. [Stantchev et al., 2015] described how low latency and privacy-sensitive healthcare applications can benefit from Fog computing. Such studies are paving the way for bringing Fog-based services to smart healthcare. These trends show that Fog has the potential to enable numerous healthcare applications, thus making Fog-enabled systems a promising research direction [Mosenia et al., 2017b].

7

Conclusion

In this article, we defined a standard framework for smart healthcare that can enable exploitation of the rapid clinical-to-daily healthcare expansion due to the proliferation of IoT and machine learning. We investigated five emerging smart healthcare systems: IBM Watson, Open mHealth, health decision support system, stress detection and alleviation system, and an energy-efficient health monitoring system. We discussed nine design considerations for both existing and future smart healthcare systems: efficiency, security, accuracy, cost, responsiveness, maintainability, scalability, reliability, and fault tolerance. We explained five innovative research trends that help address some of these design considerations. We first described an NN synthesis tool call NeST that may enable Cloud-based healthcare services to be implemented on a smartphone. We then described compressive sensing and compressed signal processing that enable inference to be energy- and storage-efficiently performed on wearable medical sensors. We then described MedMon, OpSecure, and SecureVibe that impart security to healthcare systems. Finally, we discussed several research directions that offer avenues for future innovations, including the need to address unsatisfactory datasets and machine learning algorithms,

standardization and infrastructure, and the promising role of Fog computing in smart healthcare.

Acknowledgments: This work was supported by NSF under Grant No. CNS-1617628.

References

M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.

A. O. Akmandor and N. K. Jha. Keep the stress away with SoDA: Stress detection and alleviation system. *IEEE Trans. Multi-Scale Computing Systems*, 3(4):269–282, Oct. 2017.

A. O. Akmandor and N. K. Jha. Smart health care: An edge-side computing perspective. *IEEE Consumer Electron. Mag.*, 7(1):29–37, Jan. 2018.

B. Alipanahi, A. Delong, M. T. Weirauch, and B. J. Frey. Predicting the sequence specificities of DNA-and RNA-binding proteins by deep learning. *Nature Biotechnology*, 33(8):831–838, 2015.

M. Arif and S. Basalamah. Similarity-dissimilarity plot for high dimensional data of different attribute types in biomedical datasets. *Int. J. Innovative Computing, Information and Control*, 8(2):1275–1297, 2012.

L. Atzori, A. Iera, and G. Morabito. The Internet of Things: A survey. *Computer Networks*, 54(15):2787–2805, 2010.

E. J. Candes and T. Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *IEEE Trans. Information Theory*, 52(12): 5406–5425, 2006.

X. H. Cao, I. Stojkovic, and Z. Obradovic. A robust data scaling algorithm to improve classification accuracies in biomedical data. *BMC Bioinformatics*, 17:1–10, 2016.

Y. Cao, S. Chen, P. Hou, and D. Brown. FAST: A Fog computing assisted distributed analytics system to monitor fall for stroke mitigation. In *Proc. IEEE Int. Conf. Networking, Architecture and Storage*, pages 2–11, 2015.

F. R. Cerqueira, T. G. Ferreira, A. de Paiva Oliveira, D. A. Augusto, E. Krempser, H. J. C. Barbosa, S. do Carmo Castro Franceschini, B. A. C. de Freitas, A. P. Gomes, and R. Siqueira-Batista. NICeSim: An open-source simulator based on machine learning techniques to support medical research on prenatal and perinatal care decision making. *Artificial Intelligence in Medicine*, 62(3):193–201, 2014.

R. Chandrasekar. Elementary? Question answering, IBM's Watson, and the Jeopardy! challenge. *Resonance*, 19(3):222–241, 2014.

C. Chen, D. Haddad, J. Selsky, J. E. Hoffman, R. L. Kravitz, D. Estrin, and I. Sim. Making sense of mobile health data: An open architecture to improve individual-and population-level health. *J. Medical Internet Research*, 14(4): e112, 2012.

Y. Chen, E. Argentinis, and G. Weber. IBM Watson: How cognitive computing can be applied to big data challenges in life sciences research. *Clinical Therapeutics*, 38(4):688 – 701, 2016.

J. Czerniak and H. Zarzycki. Application of rough sets in the presumptive diagnosis of urinary system diseases. *Artificial Intelligence and Security in Computing Systems*, pages 41–51, 2003.

X. Dai, H. Yin, and N. K. Jha. NeST: A neural network synthesis tool based on a grow-and-prune paradigm. *arXiv preprint arXiv:1711.02017*, 2017.

D. K. Das, M. Ghosh, M. Pal, A. K. Maiti, and C. Chakraborty. Machine learning approach for automated screening of malaria parasite using light microscopic images. *Micron*, 45:97–106, 2013.

A. V. Dastjerdi and R. Buyya. Fog computing: Helping the Internet of Things realize its potential. *IEEE Computer*, 49(8):112–116, 2016.

J. Deng, W. Dong, R. Socher, L. Li, K. Li, and F. Li. ImageNet: A large-scale hierarchical image database. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 248–255, 2009.

D. L. Donoho. Compressed sensing. *IEEE Trans. Information Theory*, 52(4): 1289–1306, 2006.

A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *Nature*, 542(7639):115–118, 2017.

D. Estrin and I. Sim. Open mHealth architecture: An engine for health care innovation. *Science*, 330(6005):759–760, 2010.

W. Gao, S. Emaminejad, H. Y. Y. Nyein, S. Challa, K. Chen, A. Peck, H. M. Fahad, H. Ota, H. Shiraki, and D. Kiriya. Fully integrated wearable sensor arrays for multiplexed in situ perspiration analysis. *Nature*, 529(7587): 509–514, 2016.

H. Ghayvat, J. Liu, S. C. Mukhopadhyay, and X. Gui. Wellness sensor networks: A proposal and implementation for smart home for assisted living. *IEEE Sensors J.*, 15(12):7341–7348, 2015.

U. Gupta, J. Park, H. Joshi, and U. Y. Ogras. Flexibility-aware system-on-polymer (SoP): Concept to prototype. *IEEE Trans. Multi-Scale Computing Systems*, 3(1):36–49, 2017.

T. Halevi and N. Saxena. Acoustic eavesdropping attacks on constrained wireless device pairing. *IEEE Trans. Information Forensics and Security*, 8 (3):563–577, 2013.

M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The WEKA data mining software: An update. *SIGKDD Explorations Newsletter*, 11(1):10–18, 2009.

D. Halperin, T. S. Heydt-Benjamin, B. Ransford, S. S. Clark, B. Defend, W. Morgan, K. Fu, T. Kohno, and W. H. Maisel. Pacemakers and implantable cardiac defibrillators: Software radio attacks and zero-power defenses. In *Proc. IEEE. Symp. Security and Privacy*, pages 129–142, 2008.

S. Han, J. Pool, J. Tran, and W. Dally. Learning both weights and connections for efficient neural network. In *Proc. Advances in Neural Information Processing Systems*, pages 1135–1143. 2015.

R. High. The era of cognitive systems: An inside look at IBM Watson and how it works. *IBM Corporation, Redbooks*, 2012.

C. W. Hoge, C. A. Castro, S. C. Messer, D. McGurk, D. I. Cotting, and R. L. Koffman. Combat duty in Iraq and Afghanistan, mental health problems, and barriers to care. *New England J. Medicine*, 351(1):13–22, 2004.

C. Hu, F. Zhang, X. Cheng, X. Liao, and D. Chen. Securing communications between external users and wireless body area networks. In *Proc. ACM Wkshp. Hot Topics on Wireless Network Security and Privacy*, pages 31–36, 2013.

T. J. Huang, J. Huang, and K. T. Cheng. Design, automation, and test for low-power and reliable flexible electronics. *Foundations and Trends® in Electronic Design Automation*, 9(2):99–210, 2015.

D. L. Hunt, R. B. Haynes, S. E. Hanna, and K. Smith. Effects of computer-based clinical decision support systems on physician performance and patient outcomes: A systematic review. *J. American Medical Association*, 280(15): 1339–1346, 1998.

M. Irie, S. Asami, S. Nagata, M. Miyata, and H. Kasai. Relationships between perceived workload, stress and oxidative DNA damage. *J. Int. Archives of Occupational and Environmental Health*, 74(2):153–157, 2001.

S. M. Jadhav, S. L. Nalbalwar, and A. A. Ghatol. Modular neural network based arrhythmia classification system using ECG signal data. *Int. J. Information Technology and Knowledge Management*, 4(1):205–209, 2011.

P. Jokic and M. Magno. Powering smart wearable systems with flexible solar energy harvesting. In *Proc. IEEE Int. Symp. Circuits and Systems*, pages 1–4, 2017.

A. H. Khandoker, M. Palaniswami, and C. K. Karmakar. Support vector machines for automated recognition of obstructive sleep apnea syndrome from ECG recordings. *IEEE Trans. Information Technology in Biomedicine*, 13(1):37–48, 2009.

Y. Kim, W. S. Lee, V. Raghunathan, N. K. Jha, and A. Raghunathan. Vibration-based secure side channel for medical devices. In *Proc. IEEE Design Automation Conf.*, pages 1–6, 2015.

J. Ko, C. Lu, M. B. Srivastava, J. A. Stankovic, A. Terzis, and M. Welsh. Wireless sensor networks for healthcare. *Proc. IEEE*, 98(11):1947–1960, 2010.

L. T. Kohn, J. M. Corrigan, and M. S. Donaldson. *To Err Is Human: Building A Safer Health System*, volume 6. National Academies Press, 2000.

K. Korotkov and R. Garcia. Computerized analysis of pigmented skin lesions: A review. *Artificial Intelligence in Medicine*, 56(2):69–90, 2012.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet classification with deep convolutional neural networks. In *Proc. Advances in Neural Information Processing Systems*, pages 1097–1105. 2012.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998.

M. K. K. Leung, A. Delong, B. Alipanahi, and B. J. Frey. Machine learning in genomic medicine: A review of computational problems and data sets. *Proc. IEEE*, 104(1):176–197, 2016.

C. Li, A. Raghunathan, and N. K. Jha. Hijacking an insulin pump: Security attacks and defenses for a diabetes therapy system. In *Proc. IEEE Int. Conf. e-Health Networking, Applications and Services*, pages 150–156, Jun. 2011.

M. Lichman. UCI machine learning repository, 2013. URL `http://archive.ics.uci.edu/ml`.

S. B. Localystics. An analysis of consumer health apps for Apple's iPhone, 2012. URL `http://www.mobihealthnews.com/research/an-analysis-of-consumer-health-apps-for-apples-iphone-2012`.

J. Lu, N. Verma, and N. K. Jha. Compressed signal processing on Nyquist-sampled signals. *IEEE Trans. Computers*, 65(11):3293–3303, 2016.

M. A. Makary and M. Daniel. Medical error - the third leading cause of death in the US. *British Medical J.*, 353:i2139, 2016.

B. S. McEwen. Protection and damage from acute and chronic stress: Allostasis and allostatic overload and relevance to the pathophysiology of psychiatric disorders. *Ann. New York Academy of Sciences*, 1032(1):1–7, 2004.

R. Miotto, L. Li, B. A. Kidd, and J. T. Dudley. Deep patient: An unsupervised representation to predict the future of patients from the electronic health records. *Scientific Reports*, 6:26094, 2016.

A. Mosenia and N. K. Jha. OpSecure: A secure unidirectional optical channel for implantable medical devices. *IEEE Trans. Multi-Scale Computing Systems*, 2017.

A. Mosenia, A. Bechara, T. Zhang, P. Mittal, and M. Chiang. ProCMotive: Bringing programability and connectivity into isolated vehicles. *arXiv preprint arXiv:1709.07450*, 2017a.

A. Mosenia, S. Sur-Kolay, A. Raghunathan, and N. K. Jha. Wearable medical sensor-based system design: A survey. *IEEE Trans. Multi-Scale Computing Systems*, 3(2):124–138, 2017b.

S. C. Mukhopadhyay. Wearable sensors for human activity monitoring: A review. *IEEE Sensors J.*, 15(3):1321–1330, 2015.

M. A. Musen, B. Middleton, and R. A. Greenes. Clinical decision-support systems. In *Biomedical Informatics*, pages 643–674. 2014.

A. M. Nia, M. Mozaffari-Kermani, S. Sur-Kolay, A. Raghunathan, and N. K. Jha. Energy-efficient long-term continuous personal health monitoring. *IEEE Trans. Multi-Scale Computing Systems*, 1(2):85–98, 2015.

A. M. Nia, S. Sur-Kolay, A. Raghunathan, and N. K. Jha. Physiological information leakage: A new frontier in health information security. *IEEE Trans. Emerging Topics in Computing*, 4(3):321–334, 2016.

R. Palaniappan, K. Sundaraj, and S. Sundaraj. A comparative study of the SVM and k-NN machine learning algorithms for the diagnosis of respiratory pathologies using pulmonary acoustic signals. *BMC Bioinformatics*, 15(1): 1–8, 2014.

A. Pantelopoulos and N. G. Bourbakis. A survey on wearable sensor-based systems for health monitoring and prognosis. *IEEE Trans. Systems, Man, and Cybernetics*, 40(1):1–12, 2010.

N. R. Potlapally, S. Ravi, A. Raghunathan, and N. K. Jha. A study of the energy consumption characteristics of cryptographic algorithms and security protocols. *IEEE Trans. Mobile Computing*, 5(2):128–143, Feb. 2006.

H. Quan, V. Sundararajan, P. Halfon, A. Fong, B. Burnand, J. Luthi, L. D. Saunders, C. A. Beck, T. E. Feasby, and W. A. Ghali. Coding algorithms for defining comorbidities in ICD-9-CM and ICD-10 administrative data. *Medical Care*, pages 1130–1139, 2005.

C. C. Quinn, M. D. Shardell, M. L. Terrin, E. A. Barr, S. H. Ballew, and A. L. Gruber-Baldini. Cluster-randomized trial of a mobile phone personalized behavioral intervention for blood glucose control. *Diabetes Care*, 34(9): 1934–1942, 2011.

P. Rajpurkar, J. Irvin, K. Zhu, B. Yang, H. Mehta, T. Duan, D. Ding, A. Bagul, C. Langlotz, K. Shpanskaya, M. P. Lungren, and A. Y. Ng. CheXNet: Radiologist-level pneumonia detection on chest X-rays with deep learning. *arXiv preprint arXiv:1711.05225*, 2017.

J. A. Salomon, H. Wang, M. K. Freeman, T. Vos, A. D. Flaxman, A. D. Lopez, and C. J. L. Murray. Healthy life expectancy for 187 countries, 1990–2010: A systematic analysis for the global burden disease study 2010. *The Lancet*, 380(9859):2144–2162, 2013.

C. Schubert, M. Lambertz, R. A. Nelesen, W. Bardwell, J. Choi, and J. E. Dimsdale. Effects of stress on heart rate complexity - A comparison between short-term and chronic stress. *J. Biological Psychology*, 80(3):325–332, 2009.

M. Shoaib, K. H. Lee, N. K. Jha, and N. Verma. A 0.6-106 μW energy scalable processor for seizure detection with compressively-sensed EEG. *IEEE Trans. Circuits and Systems-I*, 61-I(4):1105–1118, 2014.

M. Shoaib, N. K. Jha, and N. Verma. Signal processing with direct computations on compressively sensed data. *IEEE Trans. Very Large Scale Integr. Syst.*, 23(1):30–43, 2015.

V. Stantchev, A. Barnawi, S. Ghulam, J. Schubert, and G. Tamm. Smart items, Fog and Cloud computing as enablers of servitization in healthcare. *Sensors & Transducers*, 185(2):121, 2015.

C. Strydis, D. Zhu, and G. N. Gaydadjiev. Profiling of symmetric-encryption algorithms for a novel biomedical-implant architecture. In *Proc. ACM Conf. Computing Frontiers*, pages 231–240, 2008.

D. Suryakumar, A. H. Sung, and Q. Liu. Influence of machine learning vs. ranking algorithm on the critical dimension. *Int. J. Future Computer and Communication*, 2(3):215–220, 2013.

N. Tahir and H. H. Manap. Parkinson disease gait classification based on machine learning approach. *J. Applied Science*, 12(2):180–185, 2012.

A. T. Tzallas, M. G. Tsipouras, and D. I. Fotiadis. Epileptic seizure detection in EEGs using time-frequency analysis. *IEEE Trans. Information Technology in Biomedicine*, 13(5):703–710, Sept 2009.

S. Ullah, H. Higgins, B. Braem, B. Latre, C. Blondia, I. Moerman, S. Saleem, Z. Rahman, and K. S. Kwak. A comprehensive survey of wireless body area networks. *J. Medical Systems*, 36(3):1065–1094, 2012.

K. O. Wrzeszczynski, M. O. Frank, T. Koyama, K. Rhrissorrakrai, N. Robine, F. Utro, A. Emde, B. Chen, K. Arora, M. Shah, et al. Comparing sequencing assays and human-machine analyses in actionable genomics for glioblastoma. *Neurology Genetics*, 3(4):e164, 2017.

H. Yin and N. K. Jha. A health decision support system for disease diagnosis based on wearable medical sensors and machine learning ensembles. *IEEE Trans. Multi-Scale Computing Systems*, 3(4):228–241, Oct. 2017.

H. Yin, B. H. Gwee, Z. Lin, A. K., S. G. Razul, and C. M. S. See. Novel real-time system design for floating-point sub-Nyquist multi-coset signal blind reconstruction. In *Proc. IEEE Int. Symp. Circuits and Systems*, pages 954–957, May 2015.

H. Yin, Z. Wang, and N. K. Jha. A hierarchical inference model for Internet-of-Things. *IEEE Trans. Multi-Scale Computing Systems*, Submitted.

J. K. Zao, B. Martin, F. Michaud, D. Banks, A. Mosenia, R. Zolfonoon, S. Irwan, and S. Schrecker. OpenFog security requirements and approaches. In *Proc. Fog World Congress*, 2017.

M. Zhang, A. Raghunathan, and N. K. Jha. MedMon: Securing medical devices through wireless monitoring and anomaly detection. *IEEE Trans. Biomedical Circuits and Systems*, 7(6):871–881, Dec. 2013.

M. Zhang, A. Raghunathan, and N. K. Jha. Trustworthiness of medical devices and body area networks. *Proc. IEEE*, 102(8):1174–1188, 2014.