Data Protection in AI Services: A Survey

CHRISTIAN MEURISCH and MAX MÜHLHÄUSER, Technical University of Darmstadt, Germany

Advances in artificial intelligence (AI) have shaped today's user services, enabling enhanced personalization and better support. As such AI-based services inevitably require user data, the resulting privacy implications are de facto the unacceptable face of this technology. In this article, we categorize and survey the cutting-edge research on privacy and data protection in the context of personalized AI services. We further review the different protection approaches at three different levels, namely, the management, system, and AI levels—showing that (i) not all of them meet our identified requirements of evolving AI services and that (ii) many challenges are addressed separately or fragmentarily by different research communities. Finally, we highlight open research challenges and future directions in data protection research, especially that comprehensive protection requires more interdisciplinary research and a combination of approaches at different levels.

CCS Concepts: • **Human-centered computing** → Ubiquitous and mobile computing; • **Computing methodologies** → Artificial intelligence; • **Security and privacy** → Privacy protections;

Additional Key Words and Phrases: AI services, personalization, data protection, privacy, data decentralization

ACM Reference format:

Christian Meurisch and Max Mühlhäuser. 2021. Data Protection in AI Services: A Survey. *ACM Comput. Surv.* 54, 2, Article 40 (March 2021), 38 pages.

<https://doi.org/10.1145/3440754>

1 INTRODUCTION

Artificial intelligence (AI) is a key driver of the “fourth industrial revolution,” and the data required is the new currency.¹ In 2020, every human being is creating 1.7 megabytes of new data every second [51]; around 26B connected devices collect, process, and share data [152]. Such a “web of computation, data, and physical entities” enabled new types of user services [76]. Specifically, services can base their actions on AI models (termed *AI services*) to enable better support personalized to users, (R1) requiring an extensive and constant stream of personal data—i.e., digital data about individuals [141]. What is more, AI services can run on nearby Internet of Things (IoT) devices (termed *device-bound AI services*) to make human environments more responsive and supportive, (R2) requiring parts of the computation to run locally or “at the edge” [186].

¹<https://www.information-age.com/artificial-intelligence-fourth-industrial-revolution-123475170>.

This research work has been co-funded by the German Federal Ministry of Education and Research and the Hessian Ministry of Higher Education, Research, Science and the Arts within their joint support of the National Research Center for Applied Cybersecurity ATHENE, and by the Deutsche Forschungsgemeinschaft (DFG) - GRK 2050 Privacy & Trust/251805230. Authors' addresses: C. Meurisch and M. Mühlhäuser, Telecooperation Lab, Technical University of Darmstadt, Hochschulstraße 10, D-64289 Darmstadt, Germany; emails: {meurisch, max}@tk.tu-darmstadt.de.

This work is licensed under a Creative Commons Attribution International 4.0 License.

© 2021 Copyright held by the owner/author(s).

0360-0300/2021/03-ART40

<https://doi.org/10.1145/3440754>

Both requirements (R1+2) make data protection, and thus, the prevention of privacy violations and intellectual property (IP) infringements increasingly challenging. Referring to R1, providing more personal data to AI services usually leads to a higher personalization level of these services, and thus, increases their value for the user. At the same time, potentially sensitive data leaves the user's territory—the digital space that is fully controlled (and often owned) by the user—as such services typically run in the cloud of the respective provider; this requires users to sacrifice a certain level of privacy. Consequently, users always have to make a compromise between the personalization of AI services and their privacy (*personalization-privacy paradox*) [192]. Referring to R2, device-bound AI services running locally on nearby (possibly third-party) IoT devices allow their ad hoc personalization and enable low-latency (ambient) support [68]. At the same time, parts of the underlying proprietary AI algorithms/models must run outside the providers' territories, which can lead to a serious infringement of IP rights [109].

Approaches to protecting user privacy (and the providers' IP) are manifold; they have been published across different application domains and research communities. However, the data protection properties claimed by their authors—if given at all—are only tenable under certain trust assumptions, which are often insufficiently discussed and also differ considerably across the communities: Some assume trusted underlying (possibly third-party) systems or devices; others assume trusted (authorized) AI services or providers; others even assume both. It can be said that there is a lack of common understanding on the distinctive characteristics of data protection approaches, especially in AI services, making it difficult to compare them and understand their individual strengths. Many data protection approaches also take the interests of only one party (users or providers) into account and ignore the interests of the other party—which makes them a non-starter for business.

In light of these deficits, this article makes the following contributions:

- We review the various notions, definitions, and terminologies that are related to personalized AI Services. To enable the reader to understand existing works, we provide some technical background and some current examples for AI services—we systematically categorize the latter based on their initiative automation to support users and their application domains.
- We categorize the existing literature of (technical) data protection approaches across different research communities in a systematic and novel way (based on their underlying trust assumptions and key techniques used).
- We survey and assess these protection approaches based on our identified and derived category-specific requirements, opening up a new user and provider perspective. Both perspectives lead to interesting observations. For instance, each of the protection approaches—besides all their advantages—shows an individual compromise made in the personalization-privacy paradox, and not all of them meet the requirements (R1+2) of AI services. Most notably, the generally neglected challenge in device-bound AI services is the protection of proprietary AI algorithms/models locally or on nearby IoT/edge devices. All in all, many challenges are either studied separately, just optimizing certain aspects only, or fragmentarily by different communities, most of which are not yet interconnected.
- We identify open research gaps (based on our tabular summaries) and make recommendations for the future direction of data protection research in AI services.

All in all, this article opens up a new perspective on data protection in AI services; it is intended to help researchers to better understand the trust assumptions (or level) under which the claimed properties could be achieved. Finally, our classification might enable researchers to refer to their novel approach in a more concise way, fostering communication among researchers in the field.

Scope and Related Surveys. This article has a unique focus on *data protection in personalized AI services* that support individuals. We further point out that we do not focus on specific use cases and do not consider protection approaches for inter-user environments (e.g., social networks), as they are already covered by other surveys. Specifically, Humbert et al. survey works related to interdependent privacy, where the actions of some individuals affect the privacy of others [69]. Toch et al. analyze the privacy risks associated with the personalization step but do not survey approaches that deal with these risks [175]. Other privacy-related surveys focus on a specific aspect or use case only, e.g., on healthcare [10], mixed reality [39], genomic data [130], cloud services [173], privacy-preserving data mining [4], and publishing [50]. None of them discusses the works in the context of AI services and their resulting privacy implications to users. On a technical level, Wagner et al. survey privacy metrics to measure these privacy implications to users and the amount of protection offered by privacy-enhancing technologies [182]. Such technologies include confidential data stores [42] and the blockchain [199]; these surveys can be seen complementary to our article, as they partly cover data protection but have no link to AI services.

2 OVERVIEW OF PERSONALIZED AI SERVICES

In this section, we first introduce the notions and definitions that are relevant to the topic of AI services (Section 2.1). We further provide a categorization of AI services based on the degree of their proactivity (Section 2.2) and discuss their application domains (Section 2.3). As AI services inevitably require user data, we analyze users' privacy concerns and demands (Section 2.4).

2.1 Terminology

We now elaborate on the terminology around personalized AI services. Given the ambiguous use of the terms in the literature, we also clarify their use in this article.

2.1.1 Artificial Intelligence (AI). The term “artificial intelligence,” or “AI,” is used ambiguously today, without a clear definition [154]. Commonly, AI refers to an area of computer science concerned “with designing intelligent computer systems” [14] or “with the automation of intelligent behavior” [100]. However, the term “intelligent” is controversial and not yet well-defined.²

Historically, AI was coined in 1956 to refer to the process of enabling computer systems to simulate the human brain function and handle any general cognitive task in any setting (termed *human-imitative AI*). In literature, such AI-powered systems are usually classified into two groups, namely, *general* and *narrow AI*. The general (or strong) AI claims that computer systems can be conscious, being able to cope with any generalized task/problem in any domain, much like a human. The narrow (or weak) AI makes no such claim, being able to perform one specialized task/problem very well, sometimes better than humans [154].

The past two decades have seen major progress in two complementary disciplines, namely, *intelligence augmentation* (IA) and *intelligent infrastructures* (II). In the former, “computation and data are used to create services that augment human intelligence and creativity.” The latter describes “a web of computation, data, and physical entities that makes human environments more supportive” [76]. Here, physical entities often refer to mobile or IoT devices [152]. A related (sub-)discipline dealing with AI on such (edge) devices is represented by *edge AI* or *edge intelligence* (EI), with a focus mainly on aspects of resource management and communication [7, 201]. For further details on AI research, the reader is referred to Reference [98].

Machine Learning. Today, most of what is commonly labeled AI is actually *machine learning* (ML) [76]. ML, a subfield of AI, refers to statistical methods or “tools” that are used to empower

²<http://jmc.stanford.edu/artificial-intelligence/what-is-ai>.

Fig. 1. Schematic representation of a sample personalized AI service (blue) with three different kinds (A–C) of underlying AI algorithms (green); yellow marks target user data or its required involvement.

computer systems to learn from experiences/data without being programmed explicitly. Three popular types of ML are *supervised*, *unsupervised*, and *reinforcement learning*. In short, the former is the class of tasks that have access to “correct” input-output pairs (i.e., the input data is annotated with an expected target value, the so-called *class* or *label*). The latter two are classes of tasks that have no access to “correct” input-output pairs, but the latter has access to a method for quantifying its performance—the reader is referred to Reference [120] for the further introduction to ML.

Deep Learning. Still worth mentioning is *deep learning* (DL), a subfield of ML, which has become very popular in recent years. Deep learning “allows computational models that are composed of multiple processing layers [of artificial neurons] to learn representations of data with multiple levels of abstraction.” Using the backpropagation algorithm to indicate how such a neural network should change its internal parameters used to calculate the representation in each layer from those in the previous one, DL can discover intricate structures in large datasets. Compared to traditional ML algorithms, DL requires very little (feature) engineering by hand and thus little domain knowledge. Different variants of deep learning (including convolutional and recurrent neural networks) have led to breakthroughs in various areas such as image, video, or speech processing [90].

2.1.2 AI Service, Algorithm, and Model. In this article, we use the term “AI services” to subsume any AI-powered system that provides some kind of direct support to users (see IA). We further use the term “device-bound AI services” to refer to those AI services that require parts of them to run locally or “at the edge” (see II/EI). This may be the case, e.g., for privacy reasons or if AI services need low-latency access to sensors or actuators of user equipments (UE) or surrounding IoT devices.

Figure 1 shows a schematic representation of a sample AI service (blue). An AI service incorporates at least one *AI model* (circle), which processes input data to make a prediction (output) for a specific task. AI models can operate hierarchically, i.e., an output of one AI model can serve as an input of another. Technically, there are two phases: *training* and *inference*. The training phase refers to the process of creating an AI model, i.e., learning new capabilities from existing (training) data by using a specific *AI algorithm* (green). The inference phase refers to the process of using this trained AI model to make a prediction on new (unseen) data.

For the sake of completeness, an AI service may incorporate additional non-AI components, e.g., a *sensing* unit to collect data required for the underlying AI models or a *decision logic* to interpret the output of AI models and trigger appropriate actions to *actuators* [141].

2.1.3 Personalization. In the AI/ML context, *personalization* (broadly known as *customization*) is the process of tailoring an AI service, or more precisely, its underlying AI models, to an individual user to enhance the service experience/value specifically for this user. Technically speaking,

personalization aims at improving an AI model w.r.t. specific performance metrics (e.g., accuracy) or its *effectiveness* for individuals using their personal data—a detailed definition is given below.

In general, the need for *personalized AI models* (yellow circle) stems from the fact that a *general AI model* (grey circle) can work well for many users, but not for all [113]. It is important to note that the general model itself can already be biased towards a certain kind of individual (unfairness), e.g., because the training data (or the corresponding “ground-truth” data, often provided by humans) are highly biased [197]—a tutorial and further details about fairness in AI is given in Reference [13].

We now basically define the following three different types of model training that are widely practiced today (see Figure 1):

- (A) *general model training* relies on data from many users, but not from the target user;
- (B) *personal model training*, in contrast, only relies on data from the target user;
- (C) “*hybrid*” *model training* is a combination of both, considering data from other users, but also from the target user, e.g., in a subsequent *personalization step* using transfer learning, multi-task learning, or meta-learning, just to name the most important ones [138, 181].

The personalization degree of AI models depends on the personal data available to them. Thus, (A) cannot achieve personalization in the training phase. However, (B+C) come with a so-called inherent *cold start problem* for new users: AI models cannot draw any personalized inferences for users about whom they have not yet collected enough personal data [95].

2.1.4 Personal Data. The term “personal data” has several meanings. In this article, we broadly define personal data as “[digital] data relating to an identifiable person,” i.e., the digital record of “everything a person makes and does online and in the [physical] world” [64, p. 284]. According to Reference [162], personal data can be defined more precisely as “[digital] data (and metadata) created by and about people, encompassing” the following three groups of data:

- *Volunteered data* is created and explicitly shared by individuals (e.g., social media profiles) [115].
- *Observed data* is captured by recording the actions of individuals (e.g., location data).
- *Inferred data* is higher-level data about individuals based on analysis of the two previous groups of data (e.g., physical activities derived from observed accelerometer data).

We further identify the following—partly interdependent—*four factors of personal data* that affects the degree of personalization:

- The *type of data* allows AI algorithms to draw different personal inferences about users, ranging from low-level (e.g., location data) to high-level data (e.g., depressive state) [184, 190]—an initial list of categories (e.g., digital identity, communications data) is given in Reference [64, p. 285f].
- The *quantity and quality of data* (and especially of its labels) have a direct influence on the performance of the resulting AI models [163]. In general, more data leads to better results, up to a certain saturation point. However, bad data or erroneous signals (e.g., a small bias in a sensor) can lead to incorrect results of the AI models, regardless of the volume of data. Consequently, only enough high-quality data ensures meaningful and accurate insights.
- The *time frame of data* (including its temporal resolution) allows AI models to make different kinds of modeling and predictions, on which AI services base their support:
 - *present data* allows the interpretation of data only, such as in location-based services [86] or voice-controlled information retrieval [157];
 - *past data* allows an in-depth understanding of user behavior and user contexts, such as in fitness and health applications [149, 159];

- *anticipated future data* mostly relies on past data and allows an anticipatory behavior of AI services based on future states, such as in anticipatory mobile computing applications [141].

2.1.5 Privacy, Data Protection, and Ownership. All three terms have been defined and interpreted in several ways by researchers from different communities or by different governments—there exists no common, overarching concept [122, 169].

In 1890, Warren and Brandeis were the first who recognized the threats to privacy caused by technological and societal developments, formulating today’s *right to privacy* (formerly “the right to be let alone”): a general right that basically ensures the protection against the unwanted disclosure of private facts, thoughts, emotions, and so on [187]. Westin defines *privacy* in terms of information control, as the right “to control, edit, manage, and delete information about themselves and decide when, how, and to what extent information is communicated to others” [189]. This definition applies to any personal data that in any way reveals information about an individual, including situations in which the actions of some individuals affect the privacy of others (termed *interdependent privacy*). The latter can be quite challenging, as individuals do not always have full control over the sharing policies of such interdependent personal data (e.g., a photo with several—possibly unwilling or unsuspecting—individuals, taken and shared by only one individual) [69]. With the General Data Protection Regulation (GDPR)³ and the California Consumer Privacy Act (CCPA),⁴ first governments—the European Union (EU) and the U.S. state of California, respectively—have established appropriate, strong legal frameworks for the protection of their citizens’ privacy.

Whereas data privacy is all about authorized access, concerning who has it and who defines it, *data protection* (or data security) is essentially a technical issue of what data privacy dictates, concerning the securing of data against unauthorized access [179]—latest technical approaches to data protection are surveyed and further discussed in Sections 3 and 4. Closely related to data protection is data ownership [64]. Derived from English common law on ownership rights, Pentland defines *data ownership* as follows: individuals (1) have the right to *possess* their personal data, (2) must have full control over the *use* of their data, and (3) have the right to *dispose* or *distribute* their data [143].

In this context, inspired by Reference [19], we now introduce the concept of (*digital*) *territory*, which (“metaphorically”) defines “the digital space that is considered as belonging to or connected with a particular user or provider who completely controls and often even owns it.” For instance, the coined *user’s territory* is the digital space in which the user has “true” data ownership of the data contained therein; as soon as the data leaves it in raw form (e.g., unencrypted), data ownership—as defined above—can no longer be technically guaranteed (or enforced). Analogously, the coined *provider’s territory* is the digital space (often “the cloud”) in which the provider has full control over data/code (including AI services, algorithms, and models) contained therein; as soon as they leave it in raw form (e.g., unencrypted), the protection of business secrets and intellectual property can no longer be technically guaranteed. In this article, we use these terms to better assess the protection approaches considered and their (technically established) “digital boundaries.”

2.1.6 Personalization-privacy Paradox. The *privacy paradox* is a phenomenon concerning the online behavior of users, which describes the discrepancies between user attitude and their actual behavior: Although users claim to be very concerned about their privacy, they do very little to protect their personal data [73]. There are multiple theories explaining the privacy paradox; the

³<https://gdpr.eu>.

⁴https://leginfo.ca.gov/faces/billTextClient.xhtml?bill_id=201720180AB375.

most convincing of them is probably the *privacy calculus*: Users decide whether to disclose personal data on the basis of a risk-benefit calculation, in which the benefits are ultimately preferred to the risks [15]. The reasons for the latter include a lack of awareness and uncertainty, missing rationality, context dependency (e.g., social setting), and malleability of preferences [55].

A special subcategory of the privacy paradox is the *personalization-privacy paradox* [29, 192], which we now discuss in the context of AI services. Providing more, appropriate, and better personal data (see the four factors of personal data) to AI services usually leads to a higher personalization degree of them—especially in the training phase (see B+C)—and thus, increases their value for the user. At the same time, providing personal data out of the user’s territory (e.g., to the provider’s cloud) requires the user to sacrifice a certain level of privacy. Consequently, users always have to make a compromise between the personalization of AI services and their privacy, in which personalization tends to negate the downside of perceived risks to privacy (see Section 2.4). In particular, it is often difficult for users to assess the actual risk to their privacy—personal data can be of varying *sensitivity to privacy*, depending on the extent of the individual factors described in Section 2.1.4.

2.2 Categorization: Degree of Proactivity

AI services operate on the proactivity continuum ranging from zero to full initiative automation to support users [157]. We provide an adapted and refined categorization for AI services along this continuum, more precisely, based on the following three classes: *reactive-proactive-autonomous* [114].

2.2.1 Reactive Support. Reactive AI services only *respond to commands from users* to support them, currently representing the most widespread class of AI services. Commands can be entered through different input modalities (e.g., voice, gesture)—a more detailed overview of modalities is given in Reference [74]. Probably, the most prominent representatives of this class are conversational agents [60], which have made the transition from research prototypes to consumer products in the current decade: Apple Siri (2011), Google Assistant (formerly Google Now, 2012), Microsoft Cortana (2015), Amazon Alexa (2015), to name a few [99]. With such AI services, users can trigger information searches or simple task executions (e.g., finding the fastest route) [38].

2.2.2 Proactive Support. From this class onwards, AI services *proactively take actions without explicit user request*. We further divide this class into the following two levels:

- (1) AI services *inform users what to pay attention to*, e.g., using notifications or alerts. For instance, location-aware services in this class can send updates to weather and traffic conditions, among other things, that may be relevant for users in their current situation [157].
- (2) AI services *make personalized recommendations*, i.e., they additionally interpret the outcome of the underlying AI models w.r.t. user needs [171] and, if necessary, suggest concrete actions to meet these needs. For instance, fitness and health applications in this class automatically learn a user’s physical activity to strategically suggest changes to this behavior (e.g., “take a break” or “walk for 30 minutes”) for a healthier lifestyle [149] or meeting a specific goal [159].

In general, the actual actions must still be confirmed or taken by the users. To engage users in these actions, latest AI services also rely on predictive models to find opportune moments to notify [141] and intervene users [106], or incorporate mechanisms to persuade them [49, 86]. However, the key challenge in determining the type and timing of suggestions is still a hot research topic, as there are associated costs with them (e.g., if the action, relevance, or timing is wrong).

2.2.3 Autonomous Support. AI services in this class *act autonomously, making decisions and taking the actual actions on behalf of users without confirmation*. For instance, such AI services enter our lives in the form of digital agents [196], (social) robots [142], or (un-)manned vehicles (e.g., drones [45]). Even though preliminary work and first prototypes exist, this class still holds many open challenges to achieve its breakthrough. In the coming decades, this class will increasingly become the focus of research, shaping our lives the most—even having the potential to revolutionize them.

2.3 Application: Domains and Examples

We have further identified three relevant—somewhat overlapping—domains that (i) cover most of the users’ daily life, (ii) in which AI services provide direct support to users, and (iii) which are of current interest for research. If applicable, some domains are further divided into subareas [111]. For each domain, we name representative application examples of AI services that are either highly cited or widely known—Table 1 (p. 9) gives a more detailed overview of these examples.

2.3.1 Healthcare & Well-being. Well-being refers to “diverse and interconnected dimensions of physical, mental, and social well-being that extend beyond the traditional definition of health” [126]. Along this definition, we categorize this domain into the following three subareas:

Physical health covers all “visible” states and body conditions of an individual, “taking into consideration everything from the absence of disease to fitness level.”⁵ State-of-the-art AI services can detect and analyze a user’s state with respect to certain factors and intervene or make recommendations if either a worsening is imminent or an improvement is to be achieved—many of such works can be found in the related discipline of digital behavior change interventions [89]. For instance, MyBehavior automatically learns a user’s physical activity and dietary habits to strategically suggest changes to this behavior for a healthier lifestyle [149]. Schmidt et al. present a digital coach that automatically identifies the user’s strengths and weaknesses; on that basis, it creates appropriate training plans to motivate and help a user in achieving his fitness goals [159].

Mental health covers all “invisible” states of an individual, including “subjective well-being, perceived self-efficacy, autonomy, competence, inter-generational dependence, and self-actualization of one’s intellectual and emotional potential, among others”—according to the World Health Organization (WHO) [136]. This also includes coping with stress and productive work. In literature, there are several AI services/systems that support these factors. For instance, InterruptMe represents an AI service that automatically infers opportune moments for interruption [140]. Pielot et al. further identify opportune moments in which users are particularly open to engage with suggested content [145]. Other AI services can detect and monitor stress [97], depressive states [26], and emotions [151], just to name a few.

Social well-being covers all social dimensions, including social acceptance, integration, and the interaction with others [174]. For instance, SociableSense models the “sociability” of users based on their co-location and interaction patterns; it provides users with real-time feedback to foster and improve social interactions [150]. Similarly, BeWell calculates well-being scores to raise a user’s awareness, helping users to manage their overall well-being; it also includes social interactions detected by inferences from sensor data of users’ smartphones (e.g., microphone data) [85].

2.3.2 Activity Support. AI services can support users with activities, or even take them over to achieve the desired result. We categorize this kind of support into the following two subareas:

Support with physical activities covers all physical actions of an individual, e.g., movements, transportation, or cooking. For instance, Google Now (now: Google Assistant) supports users to

⁵<https://www.eupati.eu/glossary/physical-health>.

Table 1. Overview of Selected State-of-the-art Examples of User-supporting AI Services

Sample	Subarea(s)	Issue(s) addressed	Proactivity level	Personalization	PoP	Personal data (observed)	Personal data (inferred)
Health & Well-being							
MyBehavior [149]	physical	activity level, diet	☞	●	☞☞	GPS, accelerometer (acc), photo	phys. activities, places, food intake
Digital fitness coach [159]	physical	activity level	☞	●	☞	physical activities	workouts
InterruptMe [140]	mental	interruptibility	☞	●	☞	GPS, acc, emotions, BT/WiFi finger-prints, notifications	places, activities, co-located people, engagement
StressSense [97]	mental	stress	☞	●	☞	microphone (mic)	voice-based stress
Detecting & monitoring depressions [26]	mental	depression	☞	●	☞	GPS, phys. activities	depressive states
EmotionSense [151]	mental/social	social interactions	☞	●	☞	GPS, Bluetooth (BT) fingerprint, acc, mic	places, in/outdoor, co-located people, speaker, emotions
SociableSense [150]	social	sociability	🔔	●	☞☞	acc, mic, BT finger-prints	co-located people, social interactions
BeWell [85]	all	activity level, sleep, social interactions	>_	●	☞☞	GPS, acc, mic, phone recharging	social interactions, phys. activities, sleep patterns
Activity Support							
Google Assistant [141]†	physical	daily routine organization‡	☞	●	☞	location, calendar	places
Guiding experiments [160]	physical	experiment execution	>_	●	☞☞	mic, camera, acc.	voice commands, gestures, activities
Autonomous driving [104] (e.g., Tesla autopilot†)	physical	driving‡	⚙️	●	☞☞	GPS, (video, radar, car telemetry, ...)	mobility patterns, (driver/car states, ...)
Conversational agents [60] (e.g., Amazon Alexa†)	digital	info retrieval, task exec.‡	>_	●	☞	mic, task-specific data	voice commands (intents)
Google Duplex [92]†	digital	appointments	>_ / ⚙️	●	☞	—	—
Spam filtering [21]	digital	organization of emails	⚙️	●	☞	emails	contacts, categories
Ambient Intelligence							
Smart speakers [60] (e.g., Apple HomePod†)	smart home	home control‡	>_	●	☞	mic, task-specific data	voice commands (intents)
SHE vision [30]	smart home	energy management	⚙️	●	☞☞	electricity, GPS, activities, calendar	usage patterns, daily demands
Detecting drowsiness [33]	smart car	drowsiness	🔔	●	☞	camera	drowsiness level
SLaap vision [124] (e.g., safe guidance)	smart city	nightly risk areas‡	☞	●	☞☞	(depth) camera	location, activity, (risk assessment)

Note: We do not consider *volunteered data* for the “personal data” required by personalized AI services, as this data—mostly used for training the general model only—is usually of the same type as the data observed or inferred from the target users. *Encoding:* **proactivity levels** (see Section 2.2): >_—reactive, proactive (🔔—referring to “inform the user”; ☞—referring to “make personalized recommendations”), ⚙️—autonomous, ()—indicating a possible use of the system, which is, however, not realized in the paper; **personalization** (see Section 2.1.3): ●—general model, ●—personal model, ●—hybrid model (e.g., with a subsequent personalization step); **place of processing (PoP)**: ☞—local, ☞—remote, ☞☞—hybrid (local–remote), ☞☞—hybrid but confidential (personal data remains local); **misc.:** †—proprietary examples; ‡—only these selected issues are discussed.

organize their daily lives by suggesting when to leave places, finding the fastest routes and best transportation [141]. Other AI services can also support specific professional activities, such as conducting experiments in a laboratory [160] or the decision-making process of lawyers [178].

Support with digital activities covers all actions related to an individual's digital world. Probably the best known example of this area is email spam filtering—concerning the processing of emails to organize them according to specified criteria [21]. Other AI services help users by automatically rescheduling appointments or organizing task management [196].

2.3.3 Ambient Intelligence. Last but not least, ambient intelligence is the support of users with and from the physical environment [155, 188]. Research in this class can be roughly categorized into three main user-centered environments: *smart home*, *smart car* (/mobility), and *smart city*.⁶ In the former, AI services can, for instance, automatically regulate air and light conditions based on the users' preferences [41]. In smart car environments, several AI services support the driver in various situations—such as parking aid, collision avoidance, drowsiness detection—or through infotainment assistance [83, 118]. In the smart city context, recent AI-based approaches and visions rely, for instance, on so-called 4D models captured and processed at nearby smart street lamps to offer user services for augmented reality (AR), civil protection, or emergency assistance [124].

2.4 End-user: Privacy Concerns and Demands

As illustrated in Table 1, personalized AI services can address various user-related issues, offering great benefits to individuals. At the same time, these AI services require corresponding personal data of these individuals (e.g., location, microphone, or camera data) and can find out new—partly highly sensitive—data about them (e.g., depressive states, dietary habits). We now examine studies on current privacy concerns and demands of users towards such data-intensive AI services.

2.4.1 Privacy Concerns. We first provide a taxonomy of privacy concerns adapted from Wang et al. [183] to better name and classify user concerns:

- *Improper acquisition* concerns the gathering of personal data without notice or acknowledgment from users, e.g., through unauthorized access to it, additional data collection beyond the service, or unnoticed monitoring of users' physical and digital activities.
- *Improper use* concerns the *analysis* and *transfer of personal data*. In the former case, personal data is analyzed without proper notice to draw conclusions about user behavior. In the latter case, personal data is transferred to third parties without users' notice and acknowledgment.
- *Privacy invasion* mostly concerns unwanted solicitation (e.g., sending information to potential users) and unwanted executions of tasks without users' acknowledgment or permission.
- *Improper storage* is related to the concerns of information confidentiality and data integrity.

According to Reference [8], the main concern is *data transfer*. In particular, users are mostly concerned about the resulting disclosure of their behavior and the trading/selling of personal data to third parties—often leading to loss of data ownership without the awareness and consent of users.

A recent online study with 942 participants from 2019 investigates the *privacy risks perception* of users [56]. In particular, the study compares abstract (e.g., collecting data and usage patterns) and specific risk scenarios (e.g., stalking someone or planning burglaries). Users perceive both scenario types differently: Abstract risk scenarios are rated as likely but only moderately severe, while specific risk scenarios are rated as rather severe but only moderately likely. The study further shows that lay users only have a vague understanding of concrete consequences that can result

⁶ *Smart nature* and *smart rural areas* complement these environments but currently play only a subordinate role in research.

from data collection, or more specifically, from the subsequent analysis on this data. This lack of understanding of possible consequences causes users to make intuitive decisions. More precisely, users tend to base their data-sharing decisions on the perceived risk rather than the actual risk—the former is often assessed lower for the reasons mentioned above.

Several studies (e.g., References [11, 20, 81]) also investigated users' *sensitivity* when sharing different types of personal data with third parties. In Reference [20], Bilogrevic and Ortlieb conducted an online survey with 918 participants followed by interviews with 14 participants. The study reveals that the comfort level of users w.r.t. their privacy is influenced by three contextual factors, namely, the *type of data*, the *type of service*, and the *involvement of third parties*—complemented by a fourth factor, namely, the *context of use* [102]. For instance, users are less comfortable with the collection of data types such as biometrics (including fingerprints), images, contact information than environmental data (including room temperature and physical presence) [102, 127]. In particular, location values are perceived as being the most personally identifiable information among the various types of data—only the combination of (online) data surpasses this. Using “location” as an example [11], Barkhuus and Dey showed that location tracking services (relying on tracking by providers or third parties) raise more privacy concerns among users than position-aware services (relying only on local knowledge)—even though the perceived benefits of both services are the same.

2.4.2 Privacy Demands. Partly derived from the above privacy concerns, partly based on recent studies, we summarize the identified (non-technical) privacy demands of users in the following:

- *Flexible user consent & clarity of consequences.* Before AI services can be used, users must agree to the “terms of service”—the legal agreements between a service provider and the user of that service—and give their consent to the data acquisition. Three aspects are essential: (1) the terms of service should be presented by default, being short and concise, supported by standardized and quickly understandable images [170]; (2) possible concrete consequences (especially for abstract risk scenarios) should be presented to the user in an intelligible manner, in particular when sharing personal data [56]; (3) users should be able to flexibly opt-in/-out from data permissions while seeing its effect on the quality of service (QoS).
- *Contextual integrity.* Users expect contextual integrity (i.e., the right context, actors, types and flow of data) from AI services: (i) only such personal data will be collected that is appropriate for the intended context of the AI services used (*data minimization*) [67]; (ii) the collected data is only used in the intended context and not in any other [133].
- *Local storage & confidential use.* Users consider the Internet and its services intrinsically insecure and prefer local/decentralized data stores for sensitive data over cloud storage [117]. In particular, data confidentiality, integrity, and availability (called *CIA triad*) are the three desired characteristics in storing, sharing, and processing personal data [72].
- *Anonymity.* Users prefer data collections and transmissions in which they cannot be immediately identified—according to a vignette study with 1,007 participants [127]. Specifically, users want inferences to be drawn only from anonymous data, including their communication meta-/data (e.g., device-specific information, communication patterns) [139].
- *Non-data-commercial business models.* In general, users predominantly reject the commercialization of their data. As means of finance-free AI services, users rather prefer advertising to an improper use of personal data. However, users are willing to pay for AI services that incorporate additional security and privacy (S&P) mechanisms [72]. However, their willingness to pay a cost-covering price for their preferred version of an AI service (e.g., strong encryptions, decentralized data stores) is relatively low—resulting in a

Fig. 2. Schematic representation of an example AI service environment, categorized along three technical perspectives: ①—protection at management level, ②—protection at system level, ③—protection at AI level.

discrepancy between the actual cost of providing the desired S&P mechanisms and the service value to a user [117].

3 CLASSIFICATION OF DATA PROTECTION IN AI SERVICES

A generic pipeline in information systems represents *input–processing–output* (IPO). Since the input and output (I/O) protection in such systems are particularly associated with challenges in sensing [86] and human-computer interaction (HCI) [39], this survey focuses only on data (access) protection for the AI-relevant middle part (processing). The middle part of the simplified IPO model may also include *storing* and *networking* capabilities, especially in distributed computing systems.

Figure 2 presents an example of an AI service environment. Based on this environment, we explain below our categorization scheme using three different technical perspectives:

- (1) *From the Management Perspective: Data Access Rights & Storage Protection.* After sensing (*input*), user-related data enters the system/platform; it is either stored in the data storage and/or immediately processed by the AI services. This first category of data management (see Figure 2(a), ①) is concerned with protecting this data from *unauthorized access* when storing, retrieving, or acting on it. In particular, approaches in this category assume trusted underlying systems/devices. This means that threats to privacy arise from unauthorized, malicious AI services, or authorized AI services when they draw new conclusions about users from their data—even beyond their intended use [84]—and store them in separate data stores where users have little/no control over their own data.
- (2) *From the System Perspective: Data Protection against Threats/Attacks.* While in the former perspective we assumed trusted underlying systems, we now look at these systems when they are untrusted, e.g., when users use cloud-based or device-bound AI services hosted by third parties. Specifically, this category (②) is concerned with the protection of user data and the locally running proprietary code of the providers against *threats and attacks* from the system—e.g., from the hardware and privileged software (e.g., OS, hypervisor) [24]. Therefore, protection approaches in this category must support secure processing not only locally on personal devices (see Figure 2(a), ②) but also on third-party IoT/edge devices (see Figure 2(b), ②). It is also important to note that this article considers mechanisms for protecting both data “at rest” (see R1) and data “in motion” (i.e., when it is communicated across devices; see R2)—the latter is also particularly relevant in edge computing scenarios [158].
- (3) *From the AI Perspective: Data Leakages.* AI services need to access and process the data to provide personalized support (*output*). However, conventional AI algorithms can have

Fig. 3. Taxonomy of data (access) protection in AI services.

inherent data leakages, depending on their use and split between local sites (see Figure 2(a), ③) and the provider’s cloud (see Figure 2(b), ③). Accordingly, this category (③) is concerned with data protection challenges in data sharing with authorized but untrusted AI services, i.e., when data (or metadata) needs to leave the user’s territory. In addition, *confidentiality* and *integrity* can also be threatened by algorithm-specific attacks, such as membership inference [167] or comprehensive privacy analysis [129]. Like ①, this category also assumes trust in the underlying systems and devices.

In general, data protection approaches for AI services must take the inherent nature of AI algorithms/models into account to avoid performance issues w.r.t. efficiency or effectiveness. All categories proposed above are further non-exclusive, but rather complement and overlap each other. Indeed, comprehensive protection (without trust assumptions) can only be ensured if approaches in the various categories are reasonably combined and work together synergistically.

4 DATA PROTECTION APPROACHES

In this section, we discuss and review the relevant data protection approaches in the context of AI services. To this end, we categorize these approaches based on the scheme presented in Section 3. For each category, we first establish specific (technical) requirements, which are then used to assess the approaches under the given trust assumption. This means that we have interpreted the fulfillment of the requirements on the basis of the trust assumptions. Therefore, we also omitted requirements that are impossible to achieve in this category. After surveying and reviewing data protection approaches relevant to this category, we identify open challenges that should be addressed next. It is important to note that (i) some of the presented approaches have not been developed directly for AI services; (ii) others may address several aspects and thus fall into more than one category. In the former case, we discuss the approach from the perspective of its applicability in the context of AI services. In the latter case, we focus on the primary aspect of protection or the primary challenge addressed by the approach. Figure 3 gives a detailed overview of the taxonomy used below.

4.1 Protection at Management Level

This first category presents data protection approaches *at management level* (see Figure 2, ①), i.e., protection against *unauthorized access* when storing, retrieving, or acting on user data. Such protection is essential to meet user demands for “local storage & confidential use” (see Section 2.4.2) and especially relevant for proactive and autonomous AI services (see Section 2.2).

4.1.1 Specific Requirements. We first derive specific requirements that can be addressed by approaches under the given trust assumptions in this category. We divide the requirements into three

groups, namely, *data protection*, *runtime environment*, and *applicability*. The group concerning data protection, which has a direct impact on user privacy, contains the following requirements:

- *Authentication*: the system should be able to verify that an entity is who it claims to be, so that only legitimate users or AI services are allowed to access personal data or the service. After successful authentication, fine-grained authorization can follow.
- *Authorization*: the system should be able to determine whether an authenticated user or AI service has access to particular resources and to what extent.
- *Integrity (data)*: the system should be able to ensure the accuracy and consistency of data (validity) over its lifecycle, i.e., unauthorized or untrusted entities should not be able to modify or tamper with personal data.
- *Data confidentiality (observed/inferred/metadata)*: the system should protect personal data—both observed and inferred, as well as metadata—against disclosure, theft, and unintentional, unlawful, or unauthorized access. We argue that when personal data leaves the user’s territory without appropriate modifications (e.g., encryption), it cannot be reliably kept secret.

The group concerning the runtime environment, which has a direct impact on possible architectures and the resulting performance, contains the following requirements:

- *Privacy-preserving processing*: personal data should be processed in a privacy-preserving environment that is either fully under the control of the users or ensures that no data can be leaked through untrusted AI algorithms/models.
- *Data store (central/per-app)*: personal data can be stored centrally and/or per AI service. Depending on the design, the availability of personal data can be better ensured.
- *Low performance overhead/ performance optimizations*: additional or complementary data protection mechanisms should not affect performance or entail only a low overhead.

The group concerning the applicability, which has a direct impact on whether and how simply a provider can apply the approach to its AI services, contains the following requirements:

- *Support of any data type*: protection mechanisms should be designed to support all types of data, so that providers are not restricted in their AI services.
- *Support of any AI algorithm*: protection mechanisms should be designed to support all underlying AI algorithms, so that providers can easily deploy their unmodified AI services.
- *Proof-of-concept prototype (implemented/evaluated)*: the proposed data protection approach/concept should have been implemented and evaluated to show its feasibility.
- *Low platform-specific dependence*: protection mechanisms should be designed in such a way that AI services do not need to integrate platform-specific programming.
- *Cross-site deployment/use (local/nearby/cloud)*: depending on the use case (e.g., due to latency or resource demands), protection mechanisms should support and securely make use of multiple sites (e.g., local, nearby IoT/edge devices, or the cloud)—one of them should act as a “base station.”

4.1.2 Protection Approaches. According to the underlying mechanisms, we classify the different data protection approaches at management level into the following two sub-categories.

Access Control & Permission Management. Conventional access control (AC) mechanisms such as discretionary (DAC), role-based (RBAC), identity-based (IBAC), mandatory (MAC), attribute-based (ABAC), and task-based (TBAC) ACs are widely used in centralized databases [18]. However,

the administration of such AC mechanisms is complex and critical; it is therefore usually handled by security experts. Another—and probably the most critical—issue is that personal (raw) data needs to be stored in centralized databases to which the AI services get partial access. The adaptation of such AC mechanisms to ubiquitous or IoT environments is quite limited, as the access control policy is often part of the database schema definition [176]. In contrast, fully decentralized approaches such as Pretty Good Privacy (PGP) [52], Web of Trust (WOT) [204], and Friend of a Friend (FOAF) [59] are more flexible sharing models. However, such approaches require individuals to define each basic rule manually, which places a heavy burden on them and is more prone to error. Also, not all of such approaches (e.g., FOAF) are well suited for personalized (single-user) AI services.

We now survey relevant and selected approaches in this category that are more user-friendly and better applicable for ubiquitous/IoT environments and especially for AI services. Early approaches are still based on *policy enforcement*. For instance, pawS in 2002 is proposed as a privacy infrastructure, which relies on privacy beacons and proxies [88]: Nearby beacons announce the data collection of each service and their policies; the user’s mobile device delegates this request to the user’s cloud-based privacy proxy that contacts the corresponding service proxy; both negotiate the collection and use of data. However, the confidentiality of the data cannot be guaranteed as the data leaves the user’s territory; also, AI services in pawS must be based on a platform-specific infrastructure. Another approach, KP-ABE, uses attribute-based encryption to enforce fine-grained sharing of encrypted data [57]: While it allows selective sharing of data and granting access to it, this data is decrypted and used outside the user’s territory. The last approach in this category, SemaDroid, extends Android’s sensor management framework to enforce a more fine-grained AC [193]: So-called *SemaHooks* are placed within existing OS components, intercepting sensor access requests from apps to enforce new policies during data collection and to monitor the data usage of apps. However, SemaDroid is limited to Android OS and specific sensor data—a similar, more general concept constitutes *privacy mediators*, which is expected to run on edge devices [37]. Neither approach can fully ensure confidentiality and integrity: Raw data is directly passed to authorized apps in SemaDroid; in *privacy mediators*, the data can still be aggregated or obfuscated before it is ultimately released to the app or the associated cloud.

To address integrity and auditing issues, blockchain (BC)-based AC approaches were proposed—in-depth surveys on this subject can be found in References [199, 202]. Examples of such *decentralized* approaches include Enigma [205], ControlChain [147], and *BC-based audible store*. The former uses distributed hash tables (to store secret data), an external blockchain (to manage access control and serve as a tamper-proof event log), and secure multi-party computations (to process the data). In this way, Enigma “removes the need for a trusted third party, enabling autonomous control of personal data” [205]. ControlChain further distributes this management on four blockchains—namely, (1) *context BC* for storing contextual information, (2) *relationships BC* for storing public credentials and entity relationships, (3) *rules BC* for storing the authorization policies, and (4) *accountability BC* for logging and auditing data accesses [147]. Authorization decisions can then be made based on contextual information and the existing entity relationship. Similar to Enigma, the latter of the three BC-based approaches also decouple the control and data planes [164]. The main difference is that the *BC-based audible store* works with data streams: Permissions are per stream; each data stream is chunked at pre-defined lengths and end-to-end (i.e., users ↔ AI services) encrypted. All in all, however, the former two are neither implemented nor evaluated. No BC-based approach can ensure data confidentiality, as they act more or less as decentralized authorization services.

Besides authorization mechanisms, the following approaches additionally integrate *privacy-preserving processing* environments. SWYSWYK (*share what you see with who you know*) “allows each

user to physically visualize the net effects of sharing rules” on the personal data management [176]. With isolated and secure execution environments, SWYSWYK can enforce the sharing policies and ensure data confidentiality and integrity—but untrusted code can still leak data. Also, the system is only used locally. FlowFence requires providers to split their AI services into (i) a set of “quarantined modules” that operates on sensitive data and (ii) code that orchestrates the execution by chaining the former [48]. FlowFence achieves a high level of protection but introduces a performance overhead and high platform-specific dependencies. A completely new direction of data ecosystem is given by Data Cooperatives, which refer to “the voluntary collaborative pooling by individuals of their personal data for the benefit of the membership of the group or community” [63]. Specifically, individuals with their data stores can become members of a data cooperative—the legal entity in charge. External entities (called *queriers*, e.g., AI services) can then interact with such “cooperatives” and query data or execute algorithms on them. Despite the many advantages of this new data ecosystem, Data Cooperatives are neither implemented nor evaluated and focus on the common good rather than on providing AI services to individuals. The latest approach in this category, AssistantGraph, is one of the few that was designed specifically for AI services, providing design and runtime support for them [108, 112]. For the former, providers must implement parts of their AI services in AI modules that run locally in sandboxes to access personal data, which ensure data integrity and confidentiality. These modules—managed in a central repository—can be reused by multiple AI services, reducing redundancies. In the latter case, the AssistantGraph engine enables backward compatibility of versioned AI modules and manages permissions hierarchically along the data flow; it uses filters and so-called *demands* between two modules to recursively control the data flow (e.g., sampling rates). While AssistantGraph offers a high level of protection, AI services require the integration of platform-specific programming.

Personal Data Stores. In contrast to traditional solutions (e.g., Google Drive, Microsoft OneDrive), which are hosted in the provider’s cloud [184], and self-hosted solutions (e.g., own-Cloud), where users can keep their own data storage at home under their control [128], so-called *personal data stores* (PDS) choose the golden mean [180]: They introduce protected data stores and managed application execution environments, overcoming data confidentiality issues of standalone access control mechanisms. Various PDS implementations have been proposed with their individual pros and cons; we survey the most relevant and widely known among them in the following:

PDV (*personal data vault*) in 2010 is one of the first PDS implementations, only supporting specific data types (e.g., location) [125]. Nevertheless, PDV demonstrates the decoupling of collection and storage from the sharing of personal data: This data is selectively filtered by the user before being passed on to AI services—but the usability remains quite low. P3 enables two-party secure photo sharing by splitting a photo into two parts: a public part (containing the volume of the photo needed for server-side scaling and mass storing) and a secure private part (only containing the meaningful, privacy-sensitive information of the photo) that only the selected recipients can view [148]. However, P3 is limited to one specific data type only, i.e., JPEG-compliant photos. π Box is more general, supporting arbitrary data types and even any AI algorithm [91]. π Box addresses the issue of untrusted AI services by sandboxing both the local and the remote parts of them, shifting the data protection responsibilities to the trusted platform. Personal data is stored in per-sandbox, per-user stores in the cloud with a sharing channel between them. Besides all advantages, the AI services must use platform-specific programming. OMS (*open mustard seed*) presents a visionary framework that should give individuals “control over their own digital identity in a secure, transparent and accountable way” by introducing so-called *Trusted Compute Cells* (TCC) [62]. Each TCC is a network of virtual resources under control of a registry to

integrate trusted *group-based* functionalities—but OMS has limited support for AI services and is neither implemented nor evaluated.

Game changers in this category are openPDS [40, 194] and Databox [28, 35, 144]. The former newly introduces so-called *SafeAnswers*: Parts of the code run locally in a secure environment of the PDS and have full access to personal data; only the aggregated results are then returned. Although openPDS achieves greater control over user data, it has a high platform dependency and limited support for AI algorithms due to the necessity of pre-approved queries. Also, the confidentiality of the data cannot be guaranteed, as (aggregated) data leaves the user’s territory. The latter, similar to π Box, overcomes this data confidentiality issue by using trusted sandboxes to establish connected local and virtual environments for “divided” applications [123]. Further, a mix of a central and per-app data stores allows only authorized applications to access personal data. However, AI services in Databox must partly rely on platform-specific programming, which limits their flexibility; a proof-of-concept prototype has also not yet been evaluated.

Recently, Solid Pod [156] allows users to have multiple data stores (called *pods*) from different providers: Users can control access to their data and have the ability to switch between applications at any time. However, Solid Pod is designed for social and web apps; personal raw data needs to be shared to a certain extent. A proof-of-concept implementation has been initiated but not evaluated. The latest approach, PrivAI, newly *combines* and integrates two data protection mechanisms, one at system level and one at AI level, into PDS systems [110]: trusted processing with sandboxing (to protect both user data and providers’ AI algorithms) and a privacy-preserving AI layer that can (i) locally adapt general AI models to personalized models and (ii) share model updates to new users. PrivAI first shows that a combination of protection approaches on several levels overcomes the individual issues of standalone approaches and can achieve a new tradeoff between privacy and personalization of AI services.

4.1.3 Open Challenges. Table 2 provides a summary of data protection approaches *at management level*. Most of the data protection mechanisms discussed in this category are presented on generic systems and not specifically for AI services, which can limit their usefulness or applicability. Also, not all of the proposed concepts were implemented, and even fewer were evaluated with a proof-of-concept prototype. We can further see that none of the presented approaches can *fully* fulfill all of the category-specific requirements. In particular, access control and permission management approaches are not sufficient to ensure the confidentiality of data, as authorized entities ultimately gain access to (unmodified) personal data. However, personal data stores complement authentication mechanisms and introduce privacy-preserving processing environments—both local and virtual (e.g., Databox). Only, the latter enable the confidentiality of data in most approaches (e.g., FlowFence), as AI services only access personal data locally within a sandboxed environment. Combining both types of data protection mechanisms to exploit their respective advantages is very promising and would address the user’s privacy demand “local storage & confidential use” derived in Section 2.4.2—but there are still unresolved challenges: Approaches such as AssistantGraph [108] and PrivAI [110] that already combine both are still associated with performance overhead and/or platform-specific dependencies. Another open challenge is the (ad hoc) deployment of device-bound (possibly latency-demanding) AI services on nearby devices, which is not supported by almost all approaches with the same protection level.

In a world of numerous, heterogeneous AI services, it is also increasingly problematic to obtain *meaningful user consent* (which is necessary for most of these data protection approaches to work at all). In addition, proactive AI services (which are not fully autonomous) may frequently ask the user whether they are allowed to perform the proposed action (see Section 2.2)—this would lead to unwanted obtrusive behavior of them, especially in application domains such as digital

Table 2. Summary of Data Protection Approaches at Management Level
Discussed in the Context of AI Services

Approach	Year	Data Protection				Runtime Env.			Applicability					Page
		Authentication	Authorization	Integrity (data)	Data Confidentiality (observed/inferred/metadata)	Privacy-preserving Processing	Data Store (central/per-app)	Low Performance Overhead/Optim.	Support of Any Data Type	Support of Any AI Algorithm	Proof-of-Concept Prototype (implemented/evaluated)	Low Platform-specific Dependence	Cross-site Deployment/Use (local/nearby/cloud)	
Access Ctrl. & Perm. Mgmt. (10)		4	10	7	4/4/4	4	9/2	10	10	10	6/4	7	9/2/6	
pawS [88]	2002	○	●	○	○/○/○	○	●/○	●	●	●	●/○	○	●/○/★	15
Fine-gr. AC of enc. data [57]	2006	○	●	○	○/○/○	○	●/○	●	●	●	○/○	●	○/○/★	15
SemaDroid [193]	2015	○	●	○	○/○/○	○	○/○	●	●	●	●/○	○	★/○/○	15
Enigma [205]	2015	●	●	●	○/○/○	○	●/○	○	●	●	○/○	●	★/○/★	15
FlowFence [48]	2016	○	●	●	●/●/○	●	●/○	○	●	●	●/●	○	★/○/○	16
ControlChain [147]	2017	○	●	●	○/○/○	○	●/○	○	●	●	○/○	●	★/○/○	15
BC-based Audit. Store [164]	2017	●	●	●	○/○/○	○	●/○	○	●	●	○/○	●	★/○/○	15
SWYSWYK [176]	2017	○	●	●	●/●/○	●	●/○	●	●	●	○/○	○	★/○/○	15
Data Cooperatives [63]	2019	●	●	●	○/○/○	○	●/○	●	●	○	○/○	○	○/○/★	16
AssistantGraph [108]†	2019	○	●	●	●/●/○	●	●/○	●	●	●	●/○	○	★/○/○	16
Personal Data Stores (8)		7	7	6	7/5/5	7	7/6	8	6	8	7/4	4	8/2/8	
PDV [125]	2010	●	●	○	○/○/○	○	●/○	●	○	●	●/○	○	○/○/★	16
P3 [148]	2013	○	○	○	○/○/○	●	●/○	○	○	○	○/○	○	○/○/★	16
πBox [91]	2013	●	○	○	○/○/○	●	○/○	○	●	●	○/○	○	○/○/★	16
OMS [62]	2014	●	●	●	○/○/○	○	○/○	○	●	●	○/○	○	★/○/○	16
openPDS [40, 194]	2014	○	●	●	○/○/○	○	○/○	○	●	○	○/○	○	★/○/○	17
Databox [28, 123]	2015	○	●	●	○/○/○	○	○/○	○	●	●	○/○	○	★/○/○	17
Solid Pod [156]	2016	●	●	●	○/○/○	○	○/○	○	●	●	○/○	○	★/○/○	17
PrivAI [110, 112]†	2020a	●	○	○	○/○/○	○	○/○	○	●	●	○/○	○	★/○/○	17
Total Count (18)		11	17	13	11/9/9	11	16/8	18	16	18	13/8	11	17/4/14	

Note: There may be approaches that address several aspects and thus fall into more than one category or overlap with other categories (see Section 3). In such cases, we focus on the primary protection goal or challenged addressed by the approach. **Encoding: fulfillment of requirements** (see Section 4.1.1): ●—fulfilled, ○—partially fulfilled, ○—little or not fulfilled; **primary development context**: †—specially designed for AI algorithms; **deployment**: ★—base deployment. The numbers in parentheses indicate how many approaches are considered (per category or in total). The numbers per requirement column indicate how many of these approaches fulfill this requirement (per category or in total)—partial fulfillment is also counted.

activity support and ambient intelligence, where concrete actions can be taken by the AI services (see Section 2.3). A possible future direction, as in Reference [132], is to enable users to delegate consent decisions to third parties (including friends, experts, groups, and AI entities)—this would be a *first* step to (i) cope with the above numerous consent requests and to (ii) address the user’s privacy demand “flexible user consent & clarity of consequences” derived in Section 2.4.2.

However, the most critical point is that approaches in this category require the assumption that the underlying (possibly third-party) systems/devices are fully trusted. This assumption does not always hold in real-world settings (e.g., in open decentralized IoT environments). If this assumption does not hold true, this could have serious consequences for *confidentiality*, *integrity*, and *availability* (called *CIA triad* [32]) of user data and AI services.

4.2 Protection at System Level

This category considers approaches that reliably solve the potential trust issue of users and/or providers towards the underlying (possibly third-party) system (see Figure 2, ②). Such protection is essential to meet user demands for “local storage & confidential use” (see Section 2.4.2) and especially relevant for “ambient intelligence” use cases in which user data is required on “nearby” IoT/edge devices (see Section 2.3.3).

4.2.1 Specific Requirements. We first derive specific requirements that can be addressed by data protection approaches under the given trust assumptions in this category. Analogous to the previous category, we divide the requirements into the following three groups: *data/code protection*, *performance*, and *applicability*. The group concerning data and code protection, which have a direct impact on user privacy and providers’ IP rights, contains the following requirements:

- *Integrity (code/data)*: protection mechanisms should be able to ensure the accuracy and consistency of data and code—including the proprietary AI algorithms/models of providers—over their lifecycles, i.e., unauthorized or untrusted entities should not be able to modify or tamper with personal data and the underlying provider’s code of AI services.
- *Data confidentiality (code/data)*: protection mechanisms should protect personal data—both observed and inferred, as well as metadata—and code (including protected AI algorithms/models) against disclosure, theft, and unintentional, unlawful, or unauthorized access. We argue that when personal data leaves the user’s territory or code leaves the provider’s territory without appropriate modifications (e.g., encryption), it cannot be reliably kept secret.
- *Protection against untrusted code*: protection mechanisms should be resistant against untrusted code/AI services, i.e., code that processes user data should not be able to disclose this data.

The group concerning the system performance, which has a direct impact on the user experience, contains the following requirements:

- *Low performance overhead*: additional or complementary data protection mechanisms should not affect the performance of AI services or entail only a low overhead.
- *Initialization optimizations*: protection mechanisms should optimize their initialization overhead (if any), so that the user experience of AI services is not affected.
- *Runtime optimizations*: protection mechanisms should optimize their runtime overhead (if any), so that the user experience of AI services is not affected.

The group concerning the applicability, which has a direct impact on whether and how simply a provider can apply the approach to its AI services, contains the following requirements:

- *Data streaming (local/1-/n-hop)*: data protection mechanisms should be designed to support continuous and confidential data streaming on local devices (e.g., mobile device) and to 1-/n-hop nearby IoT/edge devices (e.g., public displays⁷ [119, 124]).

⁷<https://petras-iot.org/project/displays-and-sensors-on-smart-campus-dissc>.

- *Support of any AI algorithm*: protection mechanisms should be designed to support all underlying AI algorithms, so that providers can easily deploy their unmodified AI services.
- *Support of any use case*: protection mechanisms should be designed to support arbitrary use cases, so that providers are not restricted in their AI services.
- *Support for ad hoc deployment*: protection mechanisms should support ad hoc personalization of AI services on nearby devices, i.e., AI services should be able to be deployed there with the same level of protection, and personal data should be transferred there confidentially.
- *Required HW/SW technologies*: protection mechanisms should depend as little as possible on special/non-widely used hardware (HW) and software (SW) components, so that providers are not restricted in the dissemination of their AI services.

We have not considered *availability*, as untrusted system/device owners can easily terminate all processes at any time—we further discuss this issue in Section 4.2.3.

4.2.2 Protection Approaches. According to the underlying mechanisms, we classify the different data protection approaches at system level into the following two sub-categories.

Software-based Virtualization. To achieve trust *at system level*, early approaches rely on *privilege software enhancements*. For instance, InkTag extends “a standard hypervisor [(HV)] to monitor the untrusted OS using paravirtualized device drivers and virtualization hardware” [66]. In this way, trusted AI services can be run in high-assurance processes (termed HAP), which are isolated from the untrusted OS, to protect code, data, and control flow. While InkTag provides a high level of protection against the OS, such protection is not sufficient against side-channel attacks; it also requires customized hypervisors. Virtual Ghost integrates an HW abstraction layer (HAL) between the kernel and the HW to provide a set of trusted services to both the kernel (for manipulating the HW) and the applications (e.g., memory encryption, key management). By this means, Virtual Ghost outperforms InkTag in several benchmarks [36]—but it requires recompiling the entire OS (including its kernel). CQSTR [198] allows users to release their data in so-called data buckets and send a reference to the service provider; services are then released to a group of tenant *virtual machine* (VM) instances with restricted networking capabilities but access to the respective data bucket; the output is returned to the users. However, CQSTR requires the cloud provider as the root of trust—similar to proprietary cloud-based solutions like *Microsoft Azure Confidential*.⁸ Another approach in this category, MiniBox, combines two mechanisms, proposing a two-way sandbox (SB): (1) a hypervisor-based memory isolation mechanism to protect AI services from the OS using TrustVisor; (2) a one-way sandbox to protect the OS and user data from untrusted AI services using Google Native Client (NaCl) [94]. However, MiniBox cannot provide protection against physical attacks. Only complex *hardware enhancements* can provide such protection and achieve “secure” memory isolations defining a trusted boundary, but they required specialized HW in the early days—a survey of early memory encryption approaches can be found in Reference [65].

Hardware-based Virtualization. A game-changer in this category was the release of SGX (*Software Guard Extensions*) in 2015 with the sixth-generation Intel Core microprocessors [34]; first prototypes and emulations of SGX were already available since 2014. With SGX, required *hardware enhancements* for trusted computing got widely available, even on commodity processors. Specifically, SGX uses hardware-protected memory regions (so-called *enclaves*) to execute code isolated from any other code in the system including privileged software (e.g., OS, BIOS, hypervisors)—i.e., only code running in an enclave can access data in this enclave [34]. Using *remote attestation*

⁸<https://azure.microsoft.com/en-us/solutions/confidential-compute>.

(RA), the desired code can be proven to actually run securely and unmodified within the enclave of a particular device [3]. The first approaches relying on SGX are Haven [16] and VC3 [161]. The former (Haven) provides a two-way protection: (1) a trusted execution environment (TEE) through an enclave to run and protect unmodified apps from the host; (2) conversely, a secure isolation container with no access to traditional OS services to protect the host from untrusted AI services. The latter (VC3) uses the enclave concept for protecting user data in distributed *MapReduce* computations in the cloud. Both can truly ensure the integrity and confidentiality of code and data, even if the underlying system is not trusted. However, VC3 does not support arbitrary AI algorithms and protect against untrusted code; both introduce a performance overhead and are not suitable for ad hoc deployments.

The following approaches are for the most part only a “variant” of these two approaches, improving on certain aspects—therefore, we will only briefly list the new features. For instance, SCONE combines container-based virtualization (e.g., LXC, Docker) with enclaves to increase the integrity and confidentiality of the former and its corresponding use cases [9]—but it can only be used locally, as RA is not supported. Graphene-SGX combines *Graphene*—a library OS, similar to unikernels—with enclaves, providing integrity support for dynamically loaded libraries and secure multi-processes [177]. Ryoan enables distributed trusted sandboxes—relying on *NaCl* that confines data-processing modules—over trusted hardware using enclaves [71]—but Ryoan is not designed for stream processing, only supporting request-oriented data models. Opaque enables distributed enclave-protected data analytics on the cluster-computing framework *Spark* from Apache [200].

Only some approaches in this category specifically consider the nature of AI services and their underlying algorithms. For instance, Chiron distinguishes two types of enclaves: (1) several concurrent *training enclaves* and (2) a *parameter enclave* for exchanging AI model parameters between the former. In each training enclave, a service provider can run its AI models with a standard ML toolchain and access user data within an extended Ryoan sandbox [70]—both user data and AI models are kept confidential from other parties, but at the expense of performance. To reduce the inherent performance overhead while still providing *secure data analysis*, Chandra et al. incorporate data randomization mechanisms [27]. While this mechanism shows higher computational efficiency and is more resistant to side-channel attacks, it adds noise to the data—resulting in a lower effectiveness of the AI algorithms. Ohrimenko et al. propose a *multi-party ML* algorithm that is *data-oblivious*, i.e., attackers interacting with it and observing these interactions learn nothing except possibly the public parameters [134]. Specifically, multiple users not trusting each other send their encrypted data to an enclave for a joint ML task; so-called *data-oblivious primitives* are used to carefully eliminate data-dependent accesses. The latter makes the algorithm more resistant to side-channel attacks, but at the expense of performance. Also, the latter two approaches are limited to specific—simple and modified—AI algorithms.

There are also some approaches in this category that are specifically designed and optimized for specific data types. For instance, VoiceGuard exploits the secure channel established for remote attestation to subsequently load AI algorithms/models into the enclave [25]—but VoiceGuard is only evaluated with voice data, introduces a high performance overhead, and does not provide runtime protection against untrusted code. SafeKeeper, however, is very specialized in server-side password protection, which limits its applicability for AI services considerably [82].

There is one approach (namely, Ekiden) in this category that is quite different from the others: It particularly addresses the *availability* and *fault tolerance* issues by combining TEEs with decentralized blockchains [31]. In particular, it decouples consensus from data processing, enabling confidentiality-preserving and stateless TEEs that can be set up on several distributed nodes on demand (or in case of failure), while each persistent state is stored in the blockchain.

The latest approaches in this category newly address issues for mobile and ad hoc use. For instance, EdgeBox was the first approach that optimizes the initialization process to reduce the waiting time in ad hoc use cases (which would otherwise have a negative impact on the user experience). To this end, EdgeBox introduces four different modes. Depending on the mode, different initialization steps are already prepared (e.g., preloading of data and code in an encrypted way) without losing confidentiality and integrity [109]. PDSProxy extends EdgeBox with the capability of multi-hop deployment and data streaming using enclave-based proxies on untrusted nodes [107, 110]. Due to the use of SGX, there is still an inherent runtime performance overhead for both approaches. OfflineModelGuard (OMG) goes for lighter *ARM TrustZone* to make mobile devices with ARM architecture accessible for the required data protection mechanisms [17]. Among other things, OMG also optimizes the initialization steps by caching encrypted AI models locally—but it is only evaluated with speech processing.

4.2.3 Open Challenges. Table 3 provides a summary of data protection approaches *at system level*. These approaches can actually ensure *confidentiality* and *integrity* of code and data (two of the three properties of the CIA triad) by addressing the trust issue of (third-party) systems. However, we can see that none of the presented approaches can fulfill *all* of the category-specific requirements. In particular, most approaches rely on SGX hardware support to address the trust issue, which has an inherent performance overhead at setup and runtime.

First approaches—such as EdgeBox [109]—have already started to optimize the setup phase by securely preloading AI services/models; this also enables *ad hoc deployment* and initialization of (personalized, data-protected) AI services on nearby devices, especially when users are on the move and have short contact times with the device. Examples for this are especially “ambient intelligence” use cases (see Section 2.3), such as the personalization of display-saturated environments [119]: Current data protection approaches (e.g., PDSProxy [107]) would enable the confidential streaming of user data (even over several hops or public displays) and, at the same time, technically prevent the disclosure of data during processing [165]. However, to better support or enable latency-demanding and real-time use cases, *runtime optimizations* are as necessary as a fast and zero-configuration ad hoc deployment of AI services, which should be further investigated. Alternatively, another possible future direction is the *proactive deployment* of AI services, e.g., based on user mobility: Accurate short-term prediction models can identify relevant public displays that a user intends to use; multi-hop streaming of user data over interconnected public displays can enable a just-in-time initialization of AI services before the user “arrives” at the display while saving valuable resources.

Overall, addressing these challenges should not be to the disadvantage of supporting different AI algorithms. Indeed, protection approaches should be *specifically designed for AI services* to reduce the attack area and increase the potential for optimization. Moreover, more approaches should also consider and work for the ARM architecture so that mobile devices can also be supported—OfflineModelGuard demonstrates a possible way.

Untrusted code or AI services inside enclaves can endanger data confidentiality. Only a few approaches (e.g., Ryoan [71], PDSProxy [110]) include protection mechanisms against such data leaks, with additional sandboxing being the most promising. Data leakage detection mechanisms that verify the contextual integrity of data (e.g., VACCINE [168]) can further complement them.

Another open challenge is the remaining property of the CIA triad, *availability*, which cannot be guaranteed in most of the discussed approaches, since a malicious host can terminate an enclave at any time. Only two approaches—namely, Ekiden [31] and PDSProxy++ [116], an extension of PDSProxy—address this availability issue by maintaining redundant and decentralized stateless processing but in different ways: The former relies on blockchain; the latter is based on an

Table 3. Summary of Data Protection Approaches at System Level Discussed in the Context of AI Services

Approach	Year	Data/Code Protection			Performance			Applicability				Page	
		Integrity (code/data)	Confidentiality (code/data)	Protection Against Untrusted Code	Low Performance Overhead	Initialization Optimization	Runtime Optimization	Data Streaming (local/1-/n-hop)	Support of Any AI Algorithm	Support of Any Use Case	Support for Ad-hoc Deployment		Required HW/SW Technologies
Software-based Virtualization (5)		3/3	2/2	3	5	1	3	4/2/1	5	5	1		
Container-based Virtualiz.	BL	○/○	○/○	●	●	●	●	●/●/●	●	●	●	(LXC)	20
InkTag [66]	2013	○/○	○/○	○	●	○	○	●/○/○	●	●	○	HV	20
Virtual Ghost [36]	2014	●/●	●/●	●	●	○	●	●/○/○	●	●	○	HAL^	20
MiniBox [94]	2014	●/●	●/●	●	●	○	●	●/○/○	●	●	○	HV/SB	20
CQSTR [198]	2016	●/○*	○/○*	○	●	○	○	○/●/○	●	●	○	VM	20
Hardware-based Virtualization (15)		15/15	15/15	6	15	3	5	10/12/1	13	14	2		
Haven [16]	2014	●/●	●/●	●	●	○	○	●/●/○	●	●	○	SGX	21
VC3 [161]	2015	●/●	●/●	○	●	○	○	●/●/○	○	●	○	SGX	21
SCONE [9]	2016	●/●°	●/●	●	●	○	○	●/●/○	●	●	○	SGX	21
Ryoan [71]	2016	●/●	●/●	●	●	○	○	●/●/○	●	●	○	SGX	21
Multi-party ML [134]†	2016	●/●	●/●	○	●	○	●	○/●/○	●	●	○	SGX	21
Securing Data Analyt. [27]†	2017	●/●	●/●	○	●	○	●	○/●/○	●	●	○	SGX	21
Graphene-SGX [177]	2017	●/●	●/●	○	●	○	●	●/○/○	●	●	○	SGX	21
Opaque [200]	2017	●/●	●/●	○	●	○	●	○/●/○	●	●	○	SGX	21
Chiron [70]†	2018	●/●	●/●	●	●	○	○	●/●/○	●	●	○	SGX	21
VoiceGuard [25]†	2018	●/●	●/●	○	●	○	○	●/○/○	●	●	○	SGX	21
SafeKeeper [82]	2018	●/●	●/●	○	●	○	○	○/●/○	○	○	○	SGX	21
Ekiden [31]	2019	●/●	●/●	○	●	○	●	○/●/○	●	●	○	SGX	21
EdgeBox [109]†	2019	●/●	●/●	●	●	●	○	●/●/○	●	●	●	SGX	22
OfflineModelGuard [17]†	2020	●/●	●/●	○	●	●	○	●/○/○	●	●	○	TZ	22
PDSProxy [110]†	2020	●/●	●/●	●	●	●	○	●/●/●	●	●	●	SGX	22
Total Count (20)		18/18	17/17	9	20	4	8	14/14/2	18	19	3		

Note: There may be approaches that address several aspects and thus fall into more than one category or overlap with other categories (see Section 3). In such cases, we focus on the primary protection goal or challenged addressed by the approach. **Encoding:** fulfillment of requirements (see Section 4.2.1): ●–fulfilled, ○–partially fulfilled, ○–little or not fulfilled (see BL–baseline); **primary development context:** †–specially designed for AI algorithms; **trust anchor:** *–third-party system/platform; **remote attestation:** °–not supported; **compiling:** ^–recompiling of the OS (including kernel) required; **software-based technologies:** HAL–hardware abstraction layer, HV–hypervisor, SB–sandbox, VM–virtual machine. The numbers in parentheses indicate how many approaches are considered (per category or in total). The numbers per requirement column indicate how many of these approaches fulfill this requirement (per category or in total)—partial fulfillment is also counted.

interconnected (densely deployed) edge infrastructure [131]. However, there is still a conflict between our goals: low performance overhead and high availability.

Last but not least, a general challenge with such trusted computing approaches is the coping with side-channel attacks [24, 135]—for further hardening of such approaches, which is outside the scope of this article, we refer to the active research in the security community.

4.3 Protection at AI Level

As approaches from the previous two categories ensure the protection of data from unauthorized entities (see Section 4.1) and the system itself (see Section 4.2), we now look at approaches that enable the protection of data against the (possibly untrusted) providers themselves *at AI level* (see Figure 2, ③). Such protection is essential to meet user demands for “contextual integrity” (see Section 2.4.2) and especially relevant for “healthcare & well-being” use cases in which AI services require access to highly sensitive user data (see Section 2.3.1).

4.3.1 Specific Requirements. We first derive specific requirements that can be addressed by data protection approaches under the given trust assumptions in this category. Analogous to the previous categories, we divide the requirements into the following three groups: *data protection*, *personalization*, and *applicability*. The group concerning data protection, which has a direct impact on user privacy, contains the following requirements:

- *Integrity (data)*: protection mechanisms should be able to ensure the accuracy and consistency of (shared) data over its lifecycle, i.e., unauthorized or untrusted entities should not be able to modify or tamper with (shared) personal data used in AI services.
- *Data confidentiality (observed/inferred/metadata)*: protection mechanisms should protect personal data—both observed and inferred, as well as metadata—against disclosure, theft, and unintentional, unlawful, or unauthorized access. We argue that when personal data leaves the user’s territory without appropriate modifications (e.g., encryption, anonymization), it cannot be reliably kept secret. It is important to note that we do not consider communication metadata explicitly; the anonymization of such data is more relevant in online social networks.

The group concerning the personalization of AI services, which has a direct impact on the user experience and the resulting benefit to users, contains the following requirements:

- *Performance (w.r.t. accuracy)*: protection mechanisms should not negatively affect the resulting performance of AI services, e.g., in terms of accuracy.
- *Personalization capabilities*: protection mechanisms should continue to provide AI services with access to sufficient and accurate personal data to enable them to adapt to the user.
- *Low personal data involvement*: data-protecting AI services should require less personal data, thus minimizing (i) the risk of leaking sensitive data and (ii) the inherent cold start problem.
- *Low labeling effort*: data-protecting AI services should require the user to label less personal data, thus reducing the burden on the user and improving the user experience.
- *Low local resource usage*: data-protecting AI services should save local resources as far as possible, thus reducing the burden on the personal devices and improving the user experience.

The former two refer to the *effectiveness* of personalization; the latter three refer to the *efficiency* of personalization. It is important to note that the main challenge is to find the best tradeoff between data protection (*privacy*), effectiveness, and efficiency (both *personalization*) (see

Section 2.1.6). In addition, the approach should still be applicable to AI services. For this group relating to applicability, we have identified the following requirements:

- *Support of any data type*: protection mechanisms should be designed to support all types of data, so that providers are not restricted in their AI services.
- *Support of any AI algorithm*: protection mechanisms should be designed to support all underlying AI algorithms, so that providers can easily deploy their unmodified AI services.
- *Low algorithm-specific dependence*: protection mechanisms should be designed in such a way that AI services do not need to integrate specific algorithms.
- *Low complexity of applicability*: protection mechanisms should be easy to deploy for providers, i.e., the architectural and infrastructural complexity should be low.
- *GM learning/improvement capabilities*: protection mechanisms should be designed to support learning and improving the general model, thereby alleviating the cold start problem.

4.3.2 Protection Approaches. We have classified the different data protection approaches *at AI level* according to the following four specific data handling techniques that enhance user privacy.

Data-modifying Approaches. Approaches in this category modify or sanitize user data so that it cannot be linked to specific individuals—leading to an inherent conflict between both goals: *privacy* and *effectiveness*. An early key concept is *k-anonymity*, which addresses the risk of re-identification of individuals within a dataset, e.g., by removing or hiding personally identifiable information [172]. *k-anonymity* can also be used as a quality measure for privacy guarantees: Data for an individual contained in the dataset cannot be distinguished from at least $k-1$ individuals whose data also appears in the dataset. Using *k-anonymity*, for instance, Gedik et al. propose an approach in 2007 for *protecting location privacy* only, allowing users to specify k according to their individual privacy preferences [53]. However, it has been shown that such anonymization techniques such as *k-anonymity* and its variants *l-diversity* [101] and *t-closeness* [93] are vulnerable to composition attacks [77]—surveys related to such early anonymization approaches can be found in References [4, 5].

We now briefly survey approaches in this category relevant over the past decade. In particular, obfuscation- or perturbation-based approaches such as *differential privacy* were prevailing [75]. In short, for instance, *differential privacy* [44] mathematically guarantees that the output of a query is insensitive to the presence or absence of data on an individual in a dataset [191]. The *privacy loss* at a differential change in dataset can be measured by the privacy parameter ϵ (also called *privacy budget*)—the smaller the value, the better the privacy protection but the larger the perturbation noise. RAPPOR from Google is an example that enables differential privacy in a practical (real-world) setting, allowing to collect statistics from end-users (clients) with strong privacy protection using randomized responses, eliminating the need for a trusted third party [46]. The SuLQ (*sub-linear queries*) framework further modifies the privacy analysis to real-valued functions and arbitrary data types, achieving *sub-linear* computational performance with (slightly) noisy replies [22]. To further reduce the perturbation errors introduced by differential privacy mechanisms, Fan et al. propose two sampling-based methods—namely, *simple random algorithm* (SRA) and *hand-picked algorithm* (HPA)—to obtain a subset of individual data for *privacy-preserving data analytics* [47]. Xu et al. and Abadi et al. enable the use of differential privacy in *logistic regression* [191] and *deep learning* (DL) [1], respectively—but both are limited to their specific AI algorithms.

Other approaches in this category with less restrictive assumptions than those for differential privacy are Incognito, Obfuscation At-source, and Pickle. As the former two are either focused on obfuscating web data [103] or location data only [77], we concentrate more on the latter: Pickle works in two phases—namely, *regression phase* and *training phase*—for privacy-preserving

collaborative learning [96]. In the regression phase, each user u_i randomly perturbs a set of public feature vectors by a private transformation matrix M_{p_i} ; these vectors are used to learn approximation functions f_μ to calculate the desired mathematical relationships between the feature vectors using regression methods. In the training phase, users can then send their M_{p_i} -perturbed feature vectors with labels to the cloud, which can learn a collaborative model under consideration of f_μ .

All these data-modifying approaches are suitable for learning a general model in a more or less privacy-friendly way. However, these approaches show poor performance in terms of effectiveness when learning personalized AI models (as they require the modification of the data of individuals, e.g., by adding noise).

Data-encrypting Approaches. This category comprises protection approaches that work with encrypted user data, ensuring *integrity* and *confidentiality* when sharing data. In particular, two complementary encryption techniques shape this category, namely, homomorphic encryption (HE) [54] and secure multi-party computation (MPC) [146]. The former makes it possible to analyze or manipulate encrypted data without revealing the data—but low computational efficiency and limited operations limit its applicability. The latter is a cryptographic protocol that enables secure and private computations on distributed data without ever revealing or moving them outside the territory of the parties involved—but MPC entails a high communication and computational overhead.

We now briefly survey relevant approaches for AI services, many of which are based on the above two encryption techniques. For instance, Barni et al. propose a hybrid protocol based on a combination of HE and garbled circuits to classify encrypted electrocardiogram (ECG) signals from users [12]. Another approach, CryptoImg, relying on HE, allows processing (e.g., image adjustment, spatial filtering, edge sharpening) on encrypted images [203]. However, both approaches are limited to specific data types and AI algorithms. ML Confidential and CryptoNets are more general, working with different data types but only with specific AI algorithms. The former proposes an HE-based confidential protocol for AI tasks and develops appropriate confidential AI algorithms for binary classification based on their polynomial approximations [58]. The latter further demonstrates the use of HE with trained neural networks [43]—but efficiency is still challenging for both.

Other approaches in this category aim at collaborative learning of a general model. For instance, SecureML [121] proposes protocols for specific AI algorithms using MPC and the stochastic gradient descent method—often used to train AI models, including neural networks. Specifically, SecureML distributes user data among two non-colluding servers, which then train various AI models on the joint data using MPC. Also using MPC but with one trusted server only, Bonawitz et al. propose a *practical secure aggregation* mechanism that allows to compute sums of model parameter updates from local AI models of individuals (see Federated Learning, p. 27) in a secure and more efficient way than before [23]. Last but not least, DeepSecure uses the garbled circuits protocol to securely perform scalable DL execution over distributed data from individuals, assuming an honest-but-curious adversary model [153]—it still maintains the efficiency. However, the latter three approaches are associated with a high complexity of applicability, and effectiveness is still challenging.

Data-minimizing Approaches. Approaches in this category aim to increase efficiency by minimizing the amount of personal data required. Depending on the setup, the current practice of *general model (GM) training* does not require personal data of individuals during training—if any are required, then only during the inference phase. While this practice achieves high efficiency and has a low complexity of applicability, as it usually relies on voluntary data and is performed

in the cloud, the resulting general model may have low effectiveness, as it works well for many users but not for all—we use this practice as the *baseline* (BL) for this category.

To address efficiency issues (w.r.t. local resource use), the first type of approach in this category proposes the *partitioning of AI algorithms*. For instance, Neurosurgeon is an approach that splits the training of neural networks between the cloud and the users with layer granularity; it further identifies the optimal points for this split, taking into account the latency and energy consumption of personal devices [78]. Similarly, Osia et al. propose *hybrid deep learning*, in which the first layers of a layer-separated, pre-trained Siamese neural network are trained locally and the output (intermediate layer) is sent to the cloud share to complement the remaining layers [137]. However, both approaches still require labeled data, which results in a cold start problem for new users.

The next type of approach relates to *collaborative learning* of a shared model. For instance, Shokri et al. propose a practical system for *privacy-preserving DL* [166]: It can jointly learn a neural network among multiple users by running the stochastic gradient descent method with global parameters on local data and selectively updating them back. Probably the best known approach in this category is Federated Learning (FL) from Google [6, 61, 195]: Each user learns a personal model or adapts a general model locally; the provider then learns or improves this general model by aggregating locally computed model updates (e.g., weights) using a newly introduced *averaging* method [105]. However, the effectiveness of FL remains similar to that of the BL, but does not require raw volunteered data. An extended, more practical version of FL represents When Edge Meets Learning [185], allowing such distributed FL algorithms to be adapted for resource-limited devices [79].

To overcome the low effectiveness of a general model and the cold start problem of personalized models for new users, *community-based approaches* form an appropriate tradeoff. For instance, CSN [2, 87] incorporates three kinds of inter-person similarity measurements—namely, physical, lifestyle, and sensor data similarities—into the cloud-based training process. In particular, CSN builds a personalized model for a user by also integrating labeled data from other “similar” users—achieving high effectiveness and efficiency. However, CSN cannot guarantee integrity and confidentiality, as personal data leaves the user’s territory and is siloed in the cloud. {P}Net [113], however, works in two steps—combining local personalization of a general model using Patching (p. 29) with community-based personalization. In particular, {P}Net builds a personal model for a new user based on anonymously contributed model modifications (so-called *patches*) learned in the first step from other “similar” users. To determine the inter-person similarity only based on sensor data, {P}Net uses comparisons between LSH-hashed histograms of users. While {P}Net achieves a new tradeoff between privacy and personalization, the complexity of its applicability is still high.

Data-confining Approaches. This category comprises AI approaches that do not require sharing personal data outside the user’s territory. In this way, these approaches ensure data *integrity* and *confidentiality*; they are also most *effective* in personalization due to the full access of personal data locally. On the downside, approaches in this category have low *efficiency* due to the high burden to users and their personal devices; nor do they contribute to the improvement of a general model, nor can personal data be used for commercial purposes (e.g., advertising), thereby reducing the benefits for providers. Roughly speaking, standard AI algorithms (e.g., Random Forest) that can run locally, manage with little personal data, and need to re-train a personal model (PM) when new data becomes available have these properties more or less in common—we subsume approaches based on such algorithms under the term *PM training (re-trained)* representing the baseline (BL).

Other approaches in this category are mainly aimed at increasing efficiency with the same or similar effectiveness. In particular, *transfer* and *incremental learning* algorithms are a promising

Table 4. Summary of Data Protection Approaches *at AI Level* Discussed in the Context of AI Services

Approach	Year	Privacy		Personalization					Applicability					Page
		Data Protection		Effectiv.	Efficiency									
		Integrity (data)	Data Confidentiality (observed/inferred/metadata)	Performance (w.r.t. accuracy)	Personalization Capabilities	Low Personal Data Involvement	Low Labeling Effort	Low Local Resource Usage	Support of Any Data Type	Support of Any AI Algorithm	Low Algorithm-spec. Dependence	Low Complexity of Applicability	GM Learn./Improv. Capabilities	
Data-modifying Approaches (7)		6	7/7/6	0	6	0	0	7	5	6	7	7	4	
Protecting location privacy [53]	2007	●	●/●/●	○	●	○	○	●	○	●	●	●	●	
Pickle [96]	2012	○	●/●/○	○	○	○	○	●	●	●	●	●	●	
RAPPOR [46]	2014	●	●/●/●	○	●	○	○	●	●	●	●	●	●	
Priv.-prsvn. data analytics [47]	2015	●	●/●/●	○	●	○	○	●	●	●	●	●	●	
DL with differential privacy [1]	2016	●	●/●/●	○	●	○	○	●	●	○	●	●	○	
Incognito [103]	2018	●	●/●/●	○	●	○	○	●	●	●	●	●	○	
Obfuscation At-source [77]	2018	●	●/●/●	○	●	○	○	●	○	●	●	●	○	
Data-encrypting Approaches (7)		7	7/7/7	5	5	7	4	7	5	3	3	4	3	
Privacy-prsvn. ECG classif. [12]	2011	●	●/●/●	●	●	●	○	●	○	○	●	●	○	
ML Confidential [58]	2012	●	●/●/●	●	●	●	○	●	●	○	○	●	○	
CryptoNets [43]	2016	●	●/●/●	●	●	●	○	●	●	○	○	●	○	
CryptoImg [203]	2016	●	●/●/●	●	●	●	●	●	○	○	○	●	○	
SecureML [121]	2017	●	●/●/●	○	○	●	○	●	●	●	○	○	●	
Secure aggregation for FL [23]	2017	●	●/●/●	○	○	●	●	●	●	●	●	○	●	
DeepSecure [153]	2018	●	●/●/●	●	●	●	●	●	●	○	○	○	●	
Data-minimizing Approaches (8)		7	7/7/3	4-7	4	5	6	7	8	4	6	4	7	
GM training (cloud-based)	BL	●	●/●/●	○	○	●	●	●	●	●	●	●	●	
CSN [2, 87]	2011	○	○/○/○	●	●	●	●	●	●	●	●	●	●	
Privacy-preserving DL [166]	2015	●	●/●/●	○-●	○	○	●	●	●	○	●	○	●	
Neurosurgeon [78]	2017	●	●/●/○	●	●	○	○	●	●	○	○	●	○	
Hybrid Deep Learning [137]	2017	●	●/●/○	●	●	○	○	●	○	○	○	●	●	
Federated Learning [61, 105]	2017	●	●/●/○	○-●	○	○	●	○	○	○	○	○	●	
When Edge Meets Learning [185]	2018	●	●/●/○	○-●	○	●	●	●	●	●	○	○	●	
{P}Net [113]	2019	●	●/●/●	●-●	●	●	●	●	●	●	●	○	●	
Data-confining Approaches (3)		3	3/3/3	3	3	3	2	1	3	3	2	2	0	
PM training (re-trained)	BL	●	●/●/●	●	●	○	○	○	●	●	●	●	○	
PM training (incremental) [163]	2018	●	●/●/●	●	●	●	●	○	●	○	○	●	○	
Patching [80]	2018	●	●/●/●	●	●	●	●	●	●	●	●	●	○	
Total Count (25)		23	24/24/19	12-15	18	15	12	22	21	16	18	17	14	

Note: There may be approaches that address several aspects and thus fall into more than one category or overlap with other categories (see Section 3). In such cases, we focus on the primary protection goal or challenged addressed by the approach. *Encoding: fulfillment of requirements* (see Section 4.3.1): ●–fulfilled, ○–partially fulfilled, ○–little or not fulfilled; *performance scale* (see 2.1.3): ○ $\cong$ accuracy of a general model, ● $\cong$ accuracy of personal models (see *baselines* [BL]). The numbers in parentheses indicate how many approaches are considered (per category or in total). The numbers per requirement column indicate how many of these approaches fulfill this requirement (per category or in total) – partial fulfillment is also counted.

direction. The former refers to the ability to use knowledge acquired in the past to learn new tasks (with less personal data)—reducing the burden on users (e.g., lower labeling effort). The latter refers to the ability to train an existing AI model incrementally only on the basis of newly available data—reducing the burden on personal devices (e.g., lower resource use). Both can also be combined, as demonstrated by the following two exemplary approaches: Servia et al. propose a neural network architecture that is trained in the cloud and *incrementally* adapted to a user in a subsequent local personalization step by readjusting the model parameters and weights [163]. Although this approach enables deep neural networks with less personal data, it is limited to only these AI algorithms. In contrast, the idea of Patching [80] is more general, and thus, widely applicable: A general (cloud-based) “black-box” model—that can be immutable and inscrutable—is adapted to new user data (locally) by observantly inferring and fixing error regions of this new instance space, in which the model is error-prone. In this way, Patching (a kind of meta-algorithm) requires less personal data and works for arbitrary AI models—even for neural networks.

4.3.3 Open Challenges. Table 4 provides a summary of data protection approaches *at AI level*. We can see that none of the presented approaches can fulfill all of the category-specific requirements. In particular, while data-modifying approaches have a low overhead, they have an inherent conflict between effectiveness and privacy that remains unsolved. Data-encryption approaches are perfectly suited to ensure confidentiality and integrity of data, but they are limited in their applicability, as they only support a limited set of operations on encrypted data and thus AI algorithms. As the open challenge of efficiently training complex AI algorithms with encrypted data is a revolutionary step and will not be achieved in the foreseeable future, a combination with data-minimizing approaches is currently the more promising direction. For instance, shared model parameters/weights of locally trained models to improve the cloud-based general model can be securely aggregated [23]. However, to achieve high personalization accuracy while preserving user privacy, local approaches are best suited, as they have full access on personal data that never leaves the user’s territory—but low efficiency (i.e., labeling effort, local resource use) remains an open challenge.

All in all, a promising future way to achieve a new tradeoff between privacy and personalization (w.r.t. efficiency and effectiveness) is the combination of multiple steps and different approaches. For instance, Google’s Federated Learning with secure aggregation (a *data-encrypting approach*) should be used to train and improve a general model; transfer learning algorithms such as Patching (a *data-confining approach*) should then be used for a subsequent local adaptation of the general model to a user. Finally, to overcome the inherent cold start problem for new users, *data-minimizing approaches* such as {P}Net should be used to share small model updates via communities of “similar” users. Another future direction could be the integration of such communities in Federated Learning to learn and further improve a shared model confidentially and more effectively.

The research community should now continue to work on (i) reducing the amount of personal data required in AI algorithms to achieve equal or higher personalization and (ii) reducing the complexity of these protection mechanisms so that it becomes easier for providers to apply them.

5 CONCLUSION

In this article, we surveyed data protection approaches in AI services at different levels with their respective trust assumptions—Tables 2–4 summarize all these approaches per level, showing an overall comparison of them based on the identified data protection requirements.

We can say that there is *no one-size-fits-all solution* that completely fulfills all requirements for AI services. Indeed, many challenges are either studied separately, just optimizing certain aspects

only, or fragmentarily by different communities, most of which are not yet interconnected. In other words, only the combination of different approaches at different levels can achieve comprehensive protection (without any trust assumptions). This, in turn, requires *more interdisciplinary research* among the communities involved. Further, today's data protection approaches are mostly either very limited to one specific data type or AI algorithm (see Section 4.3.2) or too generic, which in turn leads to performance issues (see Section 4.2.2). Either way, future approaches to data protection require both a *stronger specialization in AI services*, taking their nature into account to cope with the inherent performance overhead, while still supporting a wide range of data types and state-of-the-art AI algorithms.

All in all, *data decentralization* has turned out to be a promising future direction to retain “true” data ownership at all—it shifts the personalization-privacy paradox to a pure personalization challenge (when data is confined) aimed at achieving the best tradeoff between effectiveness and efficiency. Decentralized computing can further address the neglected system property “availability” for data-protecting AI services. Last but not least, providers need *appropriate incentives* to apply such protection approaches despite the higher complexity than centralized architectures, which must then be profitable, too. Incentives for providers can be, for instance, higher personalized advertising (as the local code can have liberal access to personal data) or lower cloud resource consumption (as at least parts of the AI services are executed locally), which allows cost-efficient scaling. Either way, overcoming the generally neglected challenge of protecting proprietary AI algorithms/models locally is a must to get the providers on board.

Overall, this article opens up a new perspective on data protection in personalized AI services, highlights identified open challenges, and provides a suitable starting point for future research.

REFERENCES

- [1] Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *CCS'16*. ACM, 308–318. DOI: <https://doi.org/10.1145/2976749.2978318>
- [2] Saeed Abdullah, Nicholas D. Lane, and Tanzeem Choudhury. 2012. Towards population scale activity recognition: A framework for handling data diversity. In *AAAI'12*. AAAI Press, 851–857.
- [3] Tigist Abera, N. Asokan, Lucas Davi, Farinaz Koushanfar, Andrew Paverd, Ahmad-Reza Sadeghi, and Gene Tsudik. 2016. Things, trouble, trust: On building trust in IoT systems. In *DAC'16*. ACM, 121–126. DOI: <https://doi.org/10.1145/2897937.2905020>
- [4] Charu C. Aggarwal and Philip S. Yu. 2008. A general survey of privacy-preserving data mining models and algorithms. In *Privacy-preserving Data Mining*. Springer, 11–52. DOI: https://doi.org/10.1007/978-0-387-70992-5_2
- [5] Rakesh Agrawal and Ramakrishnan Srikant. 2000. Privacy-preserving data mining. In *SIGMOD'00*. ACM, 439–450. DOI: <https://doi.org/10.1145/342009.335438>
- [6] Google AI. 2019. Federated Learning. Retrieved from <https://federated.withgoogle.com>.
- [7] JongGwan An, Wenbin Li, Franck Le Gall, Ernoe Kovac, Jaeho Kim, Tarik Taleb, and JaeSeung Song. 2019. EiF: Toward an elastic IoT fog framework for AI services. *IEEE Commun. Mag.* 57, 5 (2019), 28–33. DOI: <https://doi.org/10.1109/MCOM.2019.1800215>
- [8] Annie I. Antón, Julia B. Earp, and Jessica D. Young. 2010. How internet users' privacy concerns have evolved since 2002. *IEEE Secur. Priv.* 8, 1 (2010), 21–27. DOI: <https://doi.org/10.1109/MSP.2010.38>
- [9] Sergei Arnautov, Bohdan Trach, Franz Gregor, Thomas Knauth, Andre Martin, Christian Priebe, Joshua Lind, Divya Muthukumaran et al. 2016. SCONe: Secure Linux containers with Intel SGX. In *USENIX OSDI'16*. 689–703.
- [10] Sasikanth Avancha, Amit Baxi, and David Kotz. 2012. Privacy in mobile technology for personal healthcare. *Comput. Surv.* 45, 1 (2012), 3:1–3:54. DOI: <https://doi.org/10.1145/2379776.2379779>
- [11] Louise Barkhuus and Anind K. Dey. 2003. Location-based services for mobile telephony: A study of users' privacy concerns. In *INTERACT'03*, Vol. 3. Citeseer, 702–712.
- [12] Mauro Barni, Pierluigi Failla, Riccardo Lazzeretti, Ahmad-Reza Sadeghi, and Thomas Schneider. 2011. Privacy-preserving ECG classification with branching programs and neural networks. *IEEE Trans. Inf. Forens. Secur.* 6, 2 (2011), 452–468. DOI: <https://doi.org/10.1109/TIFS.2011.2108650>
- [13] Solon Barocas, Moritz Hardt, and Arvind Narayanan. 2017. Fairness in machine learning. *NIPS Tutor*. 1 (2017).
- [14] Avron Barr and Edward A. Feigenbaum. 1981. *The Handbook of Artificial Intelligence*. Butterworth-Heinemann. DOI: <https://doi.org/10.1016/C2013-0-07690-6>

- [15] Susanne Barth and Menno D. T. De Jong. 2017. The privacy paradox—Investigating discrepancies between expressed privacy concerns and actual online behavior—A systematic literature review. *Elsevier Telemat. Inf.* 34, 7 (2017), 1038–1058. DOI: <https://doi.org/10.1016/j.tele.2017.04.013>
- [16] Andrew Baumann, Marcus Peinado, and Galen Hunt. 2014. Shielding applications from an untrusted cloud with Haven. In *USENIX OSDI'14*. 267–283.
- [17] Sebastian P. Bayerl, Tommaso Frassetto, Patrick Jauernig, Korbinian Riedhammer, Ahmad-Reza Sadeghi, Thomas Schneider, et al. 2020. Offline model guard: Secure and private ML on mobile devices. In *DATE'20*. IEEE, 460–465. DOI: <https://doi.org/10.23919/DATE48585.2020.9116560>
- [18] Elisa Bertino, Gabriel Ghinita, and Ashish Kamra. 2011. Access control for databases: Concepts and systems. *Found. Trends® Datab.* 3, 1–2 (2011), 1–148. DOI: <https://doi.org/10.1561/19000000014>
- [19] Laurent Beslay and Hannu Hakala. 2007. Digital territory: Bubbles. In *European Visions for the Knowledge Age: A Quest for New Horizons in the Information Society*. Cheshire Henbury, 69–78.
- [20] Igor Bilogrevic and Martin Ortlieb. 2016. “If you put all the pieces together...”: Attitudes towards data combination and sharing across services and companies. In *CHI'16*. ACM, 5215–5227. DOI: <https://doi.org/10.1145/2858036.2858432>
- [21] Enrico Blanzieri and Anton Bryl. 2008. A survey of learning-based techniques of email spam filtering. *Artif. Intell. Rev.* 29, 1 (2008), 63–92. DOI: <https://doi.org/10.1007/s10462-009-9109-6>
- [22] Avrim Blum, Cynthia Dwork, Frank McSherry, and Kobbi Nissim. 2005. Practical privacy: The SuLQ framework. In *PODS'05*. ACM, 128–138. DOI: <https://doi.org/10.1145/1065167.1065184>
- [23] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, et al. 2017. Practical secure aggregation for privacy-preserving machine learning. In *CCS'17*. ACM, 1175–1191. DOI: <https://doi.org/10.1145/3133956.3133982>
- [24] Ferdinand Brasser, Srdjan Capkun, Alexandra Dmitrienko, Tommaso Frassetto, Kari Kostinen, and Ahmad-Reza Sadeghi. 2019. DR.SGX: Automated and adjustable side-channel protection for SGX using data location randomization. In *ACSAC'19*. ACM, 788–800. DOI: <https://doi.org/10.1145/3359789.3359809>
- [25] Ferdinand Brasser, Tommaso Frassetto, Korbinian Riedhammer, Ahmad-Reza Sadeghi, Thomas Schneider, and Christian Weinert. 2018. VoiceGuard: Secure and private speech processing. In *Interspeech'18*. ISCA, 1303–1307.
- [26] Luca Canzian and Mirco Musolesi. 2015. Trajectories of depression: Unobtrusive monitoring of depressive states by means of smartphone mobility traces analysis. In *UbiComp'15*. ACM, 1293–1304. DOI: <https://doi.org/10.1145/2750858.2805845>
- [27] Swarup Chandra, Vishal Karande, Zhiqiang Lin, Latifur Khan, Murat Kantarcioglu, and Bhavani Thuraisingham. 2017. Securing data analytics on SGX with randomization. In *ESORICS'17*. Springer, 352–369. DOI: https://doi.org/10.1007/978-3-319-66402-6_21
- [28] Amir Chaudhry, Jon Crowcroft, Heidi Howard, Anil Madhavapeddy, Richard Mortier, Hamed Haddadi, and Derek McAuley. 2015. Personal data: Thinking inside the box. In *AA'15*. ACM, 29–32. DOI: <https://doi.org/10.7146/aahcc.v1i1.21312>
- [29] Ramnath K. Chellappa and Raymond G. Sin. 2005. Personalization versus privacy: An empirical examination of the online consumer's Dilemma. *Inf. Technol. Manag.* 6, 2-3 (2005), 181–202. DOI: <https://doi.org/10.1007/s10799-005-5879-y>
- [30] Siyun Chen, Ting Liu, Feng Gao, Jianting Ji, Zhanbo Xu, Buyue Qian, Hongyu Wu, et al. 2017. Butler, not servant: A human-centric smart home energy management system. *IEEE Commun. Mag.* 55, 2 (2017), 27–33. DOI: <https://doi.org/10.1109/MCOM.2017.1600699CM>
- [31] Raymond Cheng, Fan Zhang, Jernej Kos, Warren He, Nicholas Hynes, Noah Johnson, Ari Juels, Andrew Miller, and Dawn Song. 2019. Ekiden: A platform for confidentiality-preserving, trustworthy, and performant smart contract execution. In *EuroS&P*. IEEE, 185–200. DOI: <https://doi.org/10.1109/EuroSP.2019.00023>
- [32] Terry Chia. 2012. Confidentiality, integrity, availability: The three components of the CIA triad. *IT Security Community Blog* (2012). <https://security.blogoverflow.com/2012/08/confidentiality-integrity-availability-the-three-components-of-the-cia-triad/>.
- [33] Minhho Choi and Sang Woo Kim. 2017. Online SVM-based personalizing method for the drowsiness detection of drivers. In *EMBC'17*. IEEE, 4195–4198. DOI: <https://doi.org/10.1109/EMBC.2017.8037781>
- [34] Victor Costan and Srinivas Devadas. 2016. Intel SGX explained. *IACR Cryptol. Arch.* 2016, 086 (2016), 1–118.
- [35] Andy Crabtree, Tom Lodge, James Colley, Chris Greenhalgh, et al. 2016. Enabling the new economic actor: Data protection, the digital economy, and the databox. *Person. Ubiq. Comput.* 20, 6 (2016), 947–957. DOI: <https://doi.org/10.1007/s00779-016-0939-3>
- [36] John Criswell, Nathan Dautenhahn, and Vikram Adve. 2014. Virtual ghost: Protecting applications from hostile operating systems. In *ASPLOS'14*. ACM, 81–96. DOI: <https://doi.org/10.1145/2541940.2541986>

- [37] Nigel Davies, Nina Taft, Mahadev Satyanarayanan, Sarah Clinch, and Brandon Amos. 2016. Privacy mediators: Helping IoT cross the chasm. In *HotMobile'16*. 39–44. DOI : <https://doi.org/10.1145/2873587.2873600>
- [38] Allan de Barcelos Silva, Marcio Miguel Gomes, Cristiano André da Costa, Rodrigo da Rosa Righi, Jorge Luis Victoria Barbosa, Gustavo Pessin, Geert De Doncker, and Gustavo Federizzi. 2020. Intelligent personal assistants: A systematic literature review. *Exp. Syst. Applic.* 147 (2020), 113193. DOI : <https://doi.org/10.1016/j.eswa.2020.113193>
- [39] Jaybie A. De Guzman, Kanchana Thilakarathna, and Aruna Seneviratne. 2019. Security and privacy approaches in mixed reality: A literature survey. *Comput. Surv.* 52, 6 (2019). DOI : <https://doi.org/10.1145/3359626>
- [40] Yves-Alexandre De Montjoye, Erez Shmueli, Samuel S. Wang, and Alex Sandy Pentland. 2014. openPDS: Protecting the privacy of metadata through SafeAnswers. *PLoS ONE* 9, 7 (2014), e98790. DOI : <https://doi.org/10.1371/journal.pone.0098790>
- [41] George Demiris and Brian K. Hensel. 2008. Technologies for an aging society: A systematic review of smart home applications. *Yearb. Med. Inf.* 17, 01 (2008), 33–40. DOI : <https://doi.org/10.1055/s-0038-1638580>
- [42] Sarah M. Diesburg and An-I Andy Wang. 2010. A survey of confidential data storage and deletion methods. *Comput. Surveys* 43, 1 (2010). DOI : <https://doi.org/10.1145/1824795.1824797>
- [43] Nathan Dowlin, Ran Gilad-Bachrach, Kim Laine, Kristin Lauter, Michael Naehrig, and John Wernsing. 2016. CryptoNets: Applying neural networks to encrypted data with high throughput and accuracy. In *ICML'16*. JMLR.org, 201–210.
- [44] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 2006. Calibrating noise to sensitivity in private data analysis. In *TCC'06*. Springer, 265–284. DOI : https://doi.org/10.1007/11681878_1
- [45] Milan Erdelj, Enrico Natalizio, Kaushik R. Chowdhury, and Ian F. Akyildiz. 2017. Help from the sky: Leveraging UAVs for disaster management. *IEEE Pervas. Comput.* 16, 1 (2017), 24–32. DOI : <https://doi.org/10.1109/MPRV.2017.11>
- [46] Úlfar Erlingsson, Vasily Pihur, and Aleksandra Korolova. 2014. RAPPOR: Randomized aggregatable privacy-preserving ordinal response. In *CCS'14*. ACM, 1054–1067. DOI : <https://doi.org/10.1145/2660267.2660348>
- [47] Liyue Fan et al. 2015. A practical framework for privacy-preserving data analytics. In *WWW'15*. ACM, 311–321. DOI : <https://doi.org/10.1145/2736277.2741122>
- [48] Earleane Fernandes, Justin Paupore, Amir Rahmati, Daniel Simionato, Mauro Conti, and Atul Prakash. 2016. FlowFence: Practical data protection for emerging IoT application frameworks. In *USENIX Security'16*. 531–548.
- [49] Brian J. Fogg. 1999. Persuasive technologies. *Commun. ACM* 42, 5 (1999), 26–29. DOI : <https://doi.org/10.1145/301353.301396>
- [50] Benjamin C. M. Fung, Ke Wang, Rui Chen, and Philip S. Yu. 2010. Privacy-preserving data publishing: A survey of recent developments. *Comput. Surv.* 42, 4 (2010). DOI : <https://doi.org/10.1145/1749603.1749605>
- [51] John Gantz and David Reinsel. 2012. The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east. *IDC iView: IDC Anal. Fut.* 2007, 2012 (2012), 1–16.
- [52] Simson Garfinkel. 1995. *PGP: Pretty Good Privacy*. O'Reilly Media, Inc.
- [53] Bugra Gedik and Ling Liu. 2007. Protecting location privacy with personalized k-anonymity: Architecture and algorithms. *IEEE Trans. Mob. Comput.* 7, 1 (2007), 1–18. DOI : <https://doi.org/10.1109/TMC.2007.1062>
- [54] Craig Gentry. 2009. Fully homomorphic encryption using ideal lattices. In *STOC'09*. ACM, 169–178. DOI : <https://doi.org/10.1145/1536414.1536440>
- [55] Nina Gerber, Paul Gerber, and Melanie Volkamer. 2018. Explaining the privacy paradox: A systematic review of literature investigating privacy attitude and behavior. *Comput. Secur.* 77 (2018), 226–261. DOI : <https://doi.org/10.1016/j.cose.2018.04.002>
- [56] Nina Gerber, Benjamin Reinheimer, and Melanie Volkamer. 2019. Investigating people's privacy risk perception. In *PETS'19*. Scienco, 267–288. DOI : <https://doi.org/10.2478/popets-2019-0047>
- [57] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters. 2006. Attribute-based encryption for fine-grained access control of encrypted data. In *CCS'06*. ACM, 89–98. DOI : <https://doi.org/10.1145/1180405.1180418>
- [58] Thore Graepel, Kristin Lauter, and Michael Naehrig. 2012. ML confidential: Machine learning on encrypted data. In *ICISC'12*. Springer, 1–21. DOI : https://doi.org/10.1007/978-3-642-37682-5_1
- [59] Mike Graves, Adam Constabaris, and Dan Brickley. 2007. FOAF: Connecting people on the semantic web. *Catalog. Classif. Quart.* 43, 3–4 (2007), 191–202. DOI : https://doi.org/10.1300/J104v43n03_10
- [60] Jonathan Grudin and Richard Jacques. 2019. Chatbots, humbots, and the quest for artificial general intelligence. In *CHI'19*. ACM, 209:1–209:11. DOI : <https://doi.org/10.1145/3290605.3300439>
- [61] Andrew Hard, Kanishka Rao, Rajiv Mathews, Swaroop Ramaswamy, Françoise Beaufays, Sean Augenstein, Hubert Eichner, Chloé Kiddon, and Daniel Ramage. 2018. Federated Learning for Mobile Keyboard Prediction. arxiv:1811.03604.
- [62] Thomas Hardjono, Patrick Deegan, and John Henry Clippinger. 2014. Social use cases for the ID3 open mustard seed platform. *IEEE Technol. Soc. Mag.* 33, 3 (2014), 48–54. DOI : <https://doi.org/10.1109/MTS.2014.2345197>

- [63] Thomas Hardjono and Alex Pentland. 2019. Data cooperatives: Towards a foundation for decentralized personal data management. arxiv:1905.08819.
- [64] Thomas Hardjono, David L. Shrier, and Alex Pentland. 2019. *Trusted Data: A New Framework for Identity and Data Sharing*. MIT Connection Science & Engineering.
- [65] Michael Henson and Stephen Taylor. 2014. Memory encryption: A survey of existing techniques. *Comput. Surv.* 46, 4 (2014). DOI : <https://doi.org/10.1145/2566673>
- [66] Owen S. Hofmann, Sangman Kim, Alan M. Dunn, Michael Z. Lee, and Emmett Witchel. 2013. InkTag: Secure applications on an untrusted operating system. In *ASPLOS'13*, Vol. 48. ACM, 265–278. DOI : <https://doi.org/10.1145/2451116.2451146>
- [67] White House. 2012. Consumer data privacy in a networked world: A framework for protecting privacy and promoting innovation in the global digital economy. *White House, Washington, DC* (2012), 1–62. Retrieved from <https://www.hsdll.org/?view&did=700959>.
- [68] Yun Chao Hu, Milan Patel, Dario Sabella, Nurit Sprecher, and Valerie Young. 2015. Mobile edge computing—A key technology towards 5G. *ETSI White Paper* 11, 11 (2015), 1–16.
- [69] Mathias Humbert, Benjamin Trubert, and Kévin Huguenin. 2019. A survey on interdependent privacy. *Comput. Surv.* 52, 6 (2019). DOI : <https://doi.org/10.1145/3360498>
- [70] Tyler Hunt, Congzheng Song, Reza Shokri, Vitaly Shmatikov, and Emmett Witchel. 2018. Chiron: Privacy-preserving machine learning as a service. arxiv:1803.05961.
- [71] Tyler Hunt, Zhiting Zhu, Yuanzhong Xu, Simon Peter, and Emmett Witchel. 2016. Ryoan: A distributed sandbox for untrusted computation on secret data. In *USENIX OSDI'16*. 533–549.
- [72] Iulia Ion, Niharika Sachdeva, Ponnurangam Kumaraguru, and Srdjan Čapkun. 2011. Home is safer than the cloud! Privacy concerns for consumer cloud storage. In *SOUPS'11*. ACM, 13. DOI : <https://doi.org/10.1145/2078827.2078845>
- [73] Corey Brian Jackson and Yang Wang. 2018. Addressing the privacy paradox through personalized privacy notifications. *Proc. ACM Interact. Mob. Wear. Ubiqu. Technol.* 2, 2 (2018), 68. DOI : <https://doi.org/10.1145/3214271>
- [74] Alejandro Jaimes and Nicu Sebe. 2007. Multimodal human–computer interaction: A survey. *Comput. Vis. Image Underst.* 108, 1–2 (2007), 116–134. DOI : <https://doi.org/10.1016/j.cviu.2006.10.019>
- [75] Zhanglong Ji, Zachary C. Lipton, and Charles Elkan. 2014. Differential Privacy and Machine Learning: A Survey and Review. arxiv:1412.7584.
- [76] Michael I. Jordan. 2019. Artificial intelligence—The revolution hasn't happened yet. *Harvard Data Sci. Rev.* DOI : <https://doi.org/10.1162/99608f92.f06c6e61>
- [77] Thivya Kandappu, Archan Misra, Shih-Fen Cheng, Randy Tandriansyah, et al. 2018. Obfuscation at-source: Privacy in context-aware mobile crowd-sourcing. *Proc. ACM Interact. Mob. Wear. Ubiqu. Technol.* 2, 1 (2018), 16. DOI : <https://doi.org/10.1145/3191748>
- [78] Yiping Kang, Johann Hauswald, Cao Gao, Austin Rovinski, Trevor Mudge, et al. 2017. Neurosurgeon: Collaborative intelligence between the cloud and mobile edge. *ACM SIGARCH Comput. Archit. News* 45, 1 (2017), 615–629. DOI : <https://doi.org/10.1145/3093337.3037698>
- [79] Fabian Kaup, Stefan Hacker, Eike Mentzendorff, Christian Meurisch, and David Hausheer. 2018. *The Progress of the Energy-efficiency of Single-board Computers*. Technical Report. NetSys-TR-2018-01.
- [80] Sebastian Kauschke and Johannes Fürnkranz. 2018. Batchwise patching of classifiers. In *AAAI'18*. 3374–3381.
- [81] Predrag Klasnja, Sunny Consolvo, Tanzeem Choudhury, Richard Beckwith, and Jeffrey Hightower. 2009. Exploring privacy concerns about personal sensing. In *Pervasive'09*. Springer, 176–183. DOI : https://doi.org/10.1007/978-3-642-01516-8_13
- [82] Klaudia Krawiecka, Arseny Kurnikov, Andrew Paverd, Mohammad Mannan, and N. Asokan. 2018. SafeKeeper: Protecting web passwords using trusted execution environments. In *WWW'18*. ACM, 349–358. DOI : <https://doi.org/10.1145/3178876.3186101>
- [83] Andrew L. Kun, Susanne Boll, and Albrecht Schmidt. 2016. Shifting gears: User interfaces in the age of autonomous driving. *IEEE Pervas. Comput.* 15, 1 (2016), 32–38. DOI : <https://doi.org/10.1109/MPRV.2016.14>
- [84] Carl Landwehr. 2019. 2018: A big year for privacy. *Commun. ACM* 62, 2 (2019), 20–22. DOI : <https://doi.org/10.1145/3300224>
- [85] Nicholas D. Lane, Mu Lin, Mashfiqui Mohammad, Xiaochao Yang, Hong Lu, Giuseppe Cardone, Shahid Ali, Afsaneh Doryab, Ethan Berke, Andrew T. Campbell, et al. 2014. BeWell: Sensing sleep, physical activities and social interactions to promote wellbeing. *Mob. Netw. Applic.* 19, 3 (2014), 345–359. DOI : <https://doi.org/10.1007/s11036-013-0484-5>
- [86] Nicholas D. Lane, Emiliano Miluzzo, Hong Lu, Daniel Peebles, Tanzeem Choudhury, and Andrew T. Campbell. 2010. A survey of mobile phone sensing. *IEEE Commun. Mag.* 48, 9 (2010), 140–150. DOI : <https://doi.org/10.1109/MCOM.2010.5560598>
- [87] Nicholas D. Lane, Ye Xu, Hong Lu, Shaohan Hu, Tanzeem Choudhury, Andrew T. Campbell, and Feng Zhao. 2011. Enabling large-scale human activity inference on smartphones using community similarity networks (CSN). In *UbiComp'11*. ACM, 355–364. DOI : <https://doi.org/10.1145/2030112.2030160>

- [88] Marc Langheinrich. 2002. A privacy awareness system for ubiquitous computing environments. In *UbiComp'02*. Springer, 237–245. DOI : https://doi.org/10.1007/3-540-45809-3_19
- [89] Neal Lathia, Veljko Pejovic, Kiran K. Rachuri, Cecilia Mascolo, Mirco Musolesi, and Peter J. Rentfrow. 2013. Smartphones for large-scale behavior change interventions. *IEEE Pervas. Comput.* 12, 3 (2013), 66–73. DOI : <https://doi.org/10.1109/MPRV.2013.56>
- [90] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature* 521, 7553 (2015), 436–444. DOI : <https://doi.org/10.1038/nature14539>
- [91] Sangmin Lee, Edmund L. Wong, Deepak Goel, Mike Dahlin, et al. 2013. π Box: A platform for privacy-preserving apps. In *USENIX NSDI'13*. 501–514.
- [92] Yaniv Leviathan and Yossi Matias. 2018. Google duplex: An AI system for accomplishing real-world tasks over the phone. *Google AI Blog* 8 (2018). Retrieved from <https://ai.googleblog.com/2018/05/duplex-ai-system-for-natural-conversation.html>.
- [93] Ninghui Li, Tiancheng Li, and S. Venkatasubramanian. 2007. t-closeness: Privacy beyond k-anonymity and l-diversity. In *ICDE'07*. IEEE, 106–115. DOI : <https://doi.org/10.1109/ICDE.2007.367856>
- [94] Yanlin Li, Jonathan McCune, James Newsome, Adrian Perrig, Brandon Baker, and Will Drewry. 2014. MiniBox: A two-way sandbox for x86 native code. In *USENIX ATC'14*. 409–420.
- [95] Blerina Lika, Kostas Kolomvatsos, and Stathes Hadjiefthymiades. 2014. Facing the cold start problem in recommender systems. *Exp. Syst. Applic.* 41, 4 (2014), 2065–2073. DOI : <https://doi.org/10.1016/j.eswa.2013.09.005>
- [96] Bin Liu, Yurong Jiang, Fei Sha, and Ramesh Govindan. 2012. Cloud-enabled privacy-preserving collaborative learning for mobile sensing. In *SenSys'12*. ACM, 57–70. DOI : <https://doi.org/10.1145/2426656.2426663>
- [97] Hong Lu, Denise Fraundorfer, Mashfiqui Rabbi, Marianne Schmid Mast, Gokul T. Chittaranjan, Andrew T. Campbell, Daniel Gatica-Perez, and Tanzeem Choudhury. 2012. StressSense: Detecting stress in unconstrained acoustic environments using smartphones. In *UbiComp'12*. ACM, 351–360. DOI : <https://doi.org/10.1145/2370216.2370270>
- [98] Yang Lu. 2019. Artificial intelligence: A survey on evolution, models, applications and future trends. *J. Manag. Anal.* 6, 1 (2019), 1–29. DOI : <https://doi.org/10.1080/23270012.2019.1570365>
- [99] Ewa Luger and Abigail Sellen. 2016. Like having a really bad PA: The gulf between user expectation and experience of conversational agents. In *CHI'16*. ACM, 5286–5297. DOI : <https://doi.org/10.1145/2858036.2858288>
- [100] George F. Luger and William A. Stubblefield. 1993. *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*. Benjamin/Cummings Pub. Co.
- [101] Ashwin Machanavajjhala, Daniel Kifer, Johannes Gehrke, and Muthuramakrishnan Venkatasubramanian. 2007. L-diversity: Privacy beyond k-anonymity. *ACM Trans. Knowl. Discov. Data* 1, 1 (2007), 3. DOI : <https://doi.org/10.1145/1217299.1217302>
- [102] Kirsten Martin and Katie Shilton. 2016. Putting mobile application privacy in context: An empirical study of user privacy expectations for mobile devices. *Inf. Soc.* 32, 3 (2016), 200–216. DOI : <https://doi.org/10.1080/01972243.2016.1153012>
- [103] Rahat Masood, Dinusha Vatsalan, Muhammad Ikram, et al. 2018. Incognito: A method for obfuscating web data. In *WWW'18*. ACM, 267–276. DOI : <https://doi.org/10.1145/3178876.3186093>
- [104] Markus Maurer, J. Christian Gerdes, Barbara Lenz, and Hermann Winner. 2016. *Autonomous Driving*. Springer Berlin. DOI : <https://doi.org/10.1007/978-3-662-48847-8>
- [105] H. Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. 2017. Communication-efficient learning of deep networks from decentralized data. In *AISTATS'17*. JMLR: W&CP, 1273–1282.
- [106] Abhinav Mehrotra and Mirco Musolesi. 2017. Intelligent Notification Systems: A Survey of the State of the Art and Research Challenges. arxiv:1711.10171
- [107] Christian Meurisch, Bekir Bayrak, Florian Giger, and Max Mühlhäuser. 2020. PDSProxy: Trusted IoT proxies for confidential ad-hoc personalization of AI services. In *ICCCN'20*. IEEE, 1–2. DOI : <https://doi.org/10.1109/ICCCN49398.2020.9209655>
- [108] Christian Meurisch, Bekir Bayrak, and Max Mühlhäuser. 2019. AssistantGraph: An approach for reusable and composable data-driven assistant components. In *COMPSAC'19*. IEEE, 513–522. DOI : <https://doi.org/10.1109/COMPSAC.2019.00079>
- [109] Christian Meurisch, Bekir Bayrak, and Max Mühlhäuser. 2019. EdgeBox: Confidential ad-hoc personalization of nearby IoT applications. In *GLOBECOM'19*. IEEE, 1–6. DOI : <https://doi.org/10.1109/GLOBECOM38437.2019.9013520>
- [110] Christian Meurisch, Bekir Bayrak, and Max Mühlhäuser. 2020. Privacy-preserving AI services through data decentralization. In *WWW'20*. ACM, 190–200. DOI : <https://doi.org/10.1145/3366423.3380106>
- [111] Christian Meurisch, Maria-Dorina Ionescu, Benedikt Schmidt, and Max Mühlhäuser. 2017. Reference model of next-generation digital personal assistant: Integrating proactive behavior. In *Proceedings of the ACM International Joint*

- Conference on Pervasive and Ubiquitous Computing and Proceedings of the ACM International Symposium on Wearable Computers*. ACM, 149–152. DOI : <https://doi.org/10.1145/3123024.3123145>
- [112] Christian Meurisch, Bennet Jeutter, Wladimir Schmidt, Nickolas Gündling, Benedikt Schmidt, Fabian Herrlich, and Max Mühlhäuser. 2017. An extensible pervasive platform for large-scale anticipatory mobile computing. In *COMP-SAC'17*. IEEE, 459–464. DOI : <https://doi.org/10.1109/COMPSAC.2017.54>
- [113] Christian Meurisch, Sebastian Kauschke, Tim Grube, Bekir Bayrak, and Max Mühlhäuser. 2019. {P}Net: Privacy-preserving personalization of AI-based models by anonymous inter-person similarity networks. In *MobiQuitous'19*. ACM, 60–69. DOI : <https://doi.org/10.1145/3360774.3360819>
- [114] Christian Meurisch, Cristina Mihale-Wilson, Adrian Hawlitschek, Florian Giger, Florian Müller, Oliver Hinz, and Max Mühlhäuser. 2020. Exploring user expectations of proactive AI systems. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 4 (12 2020), 1–22. DOI : <https://doi.org/10.1145/3432193>
- [115] Christian Meurisch, Benedikt Schmidt, Michael Scholz, Immanuel Schweizer, and Max Mühlhäuser. 2015. Labels: Quantified self app for human activity sensing. In *UbiComp/ISWC'15*. ACM, 1413–1422. DOI : <https://doi.org/10.1145/2800835.2801612>
- [116] Christian Meurisch, Dennis Werner, Bekir Bayrak, Florian Giger, and Max Mühlhäuser. 2020. PDSProxy++: Proactive proxy deployment for confidential personalization of AI services. In *ICCCN'20*. IEEE, 1–9. DOI : <https://doi.org/10.1109/ICCCN49398.2020.9209747>
- [117] Cristina Mihale-Wilson, Jan Zibuschka, and Oliver Hinz. 2017. About user preferences and willingness to pay for a secure and privacy protective ubiquitous personal assistant. In *ECIS'17*. AIS Electronic Library, 32–47.
- [118] Cristina Mihale-Wilson, Jan Zibuschka, and Oliver Hinz. 2019. User preferences and willingness to pay for in-vehicle assistance. *Electron. Mark.* 29, 1 (2019), 37–53. DOI : <https://doi.org/10.1007/s12525-019-00330-5>
- [119] Mateusz Mikusz, Kenny Tsu Wei Choo, Rajesh Krishna Balan, Nigel Davies, and Youngki Lee. 2019. New challenges in display-saturated environments. *IEEE Pervas. Comput.* 18, 2 (2019), 67–75. DOI : <https://doi.org/10.1109/MPRV.2019.2906992>
- [120] Thomas M. Mitchell. 1997. *Machine Learning* (1st ed.). McGraw-Hill, Inc., New York, NY.
- [121] Payman Mohassel and Yupeng Zhang. 2017. SecureML: A system for scalable privacy-preserving machine learning. In *SP'17*. IEEE, 19–38. DOI : <https://doi.org/10.1109/SP.2017.12>
- [122] Adam D. Moore. 2003. Privacy: Its meaning and value. *Amer. Philos. Quart.* 40, 3 (2003), 215–227.
- [123] Richard Mortier, Jianxin Zhao, Jon Crowcroft, Liang Wang, Qi Li, Hamed Haddadi, Yousef Amar, Andy Crabtree, et al. 2016. Personal data management with the databox: What's inside the box? In *CAN'16*. ACM, 49–54. DOI : <https://doi.org/10.1145/3010079.3010082>
- [124] Max Mühlhäuser, Christian Meurisch, Michael Stein, Jörg Daubert, Julius von Willich, Jan Riemann, and Lin Wang. 2020. Street lamps as a platform. *Commun. ACM* 63, 6 (2020), 75–83. DOI : <https://doi.org/10.1145/3376900>
- [125] Min Mun, Shuai Hao, Nilesh Mishra, Katie Shilton, Jeff Burke, Deborah Estrin, Mark Hansen, et al. 2010. Personal data vaults: A locus of control for personal data streams. In *CoNEXT'10*. ACM. DOI : <https://doi.org/10.1145/1921168.1921191>
- [126] Huseyin Naci and John P. A. Ioannidis. 2015. Evaluation of wellness determinants and interventions by citizen scientists. *Jama* 314, 2 (2015), 121–122. DOI : <https://doi.org/10.1001/jama.2015.6160>
- [127] Pardis Emami Naeini, Sruti Bhagavatula, Hana Habib, Martin Degeling, Lujio Bauer, Lorrie Faith Cranor, and Norman Sadeh. 2017. Privacy expectations and preferences in an IoT world. In *USENIX SOUPS'17*. 399–412.
- [128] Arvind Narayanan, Vincent Toubiana, Solon Barocas, Helen Nissenbaum, and Dan Boneh. 2012. A Critical Look at Decentralized Personal Data Architectures. arxiv:1202.4503.
- [129] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2018. Comprehensive Privacy Analysis of Deep Learning: Stand-alone and Federated Learning under Passive and Active White-box Inference Attacks. arxiv:1812.00910.
- [130] Muhammad Naveed, Erman Ayday, Ellen W. Clayton, Jacques Fellay, Carl A. Gunter, Jean-Pierre Hubaux, Bradley A. Malin, and Xiaofeng Wang. 2015. Privacy in the genomic era. *Comput. Surv.* 48, 1 (2015). DOI : <https://doi.org/10.1145/2767007>
- [131] The An Binh Nguyen, Christian Meurisch, Stefan Niemczyk, Doreen Böhnstedt, Kurt Geihs, Max Mühlhäuser, and Ralf Steinmetz. 2017. Adaptive task-oriented message template for in-network processing. In *NetSys'17*. IEEE, 1–8. DOI : <https://doi.org/10.1109/NetSys.2017.7903952>
- [132] Bettina Nissen, Victoria Neumann, Mateusz Mikusz, Rory Gianni, Sarah Clinch, Chris Speed, and Nigel Davies. 2019. Should I agree? Delegating consent decisions beyond the individual. In *CHI'19*. 1–13. DOI : <https://doi.org/10.1145/3290605.3300745>
- [133] Helen Nissenbaum. 2004. Privacy as contextual integrity. *Wash. L. Rev.* 79 (2004), 119.
- [134] Olga Ohrimenko, Felix Schuster, Cédric Fournet, Aastha Mehta, Sebastian Nowozin, Kapil Vaswani, and Manuel Costa. 2016. Oblivious multi-party machine learning on trusted processors. In *USENIX SEC'16*. 619–636.

- [135] Oleksii Oleksenko, Bohdan Trach, Robert Krahn, Mark Silberstein, and Christof Fetzter. 2018. Varys: Protecting SGX enclaves from practical side-channel attacks. In *USENIX ATC'18*. 227–240.
- [136] World Health Organization. 2001. *The World Health Report 2001: Mental Health: New Understanding, New Hope*. World Health Organization. <https://www.who.int/whr/2001/en/>.
- [137] Seyed Ali Osia, Ali Shahin Shamsabadi, Sina Sajadmanesh, Ali Taheri, Kleomenis Katevas, Hamid R. Rabiee, Nicholas D. Lane, et al. 2017. A Hybrid Deep Learning Architecture for Privacy-Preserving Mobile Analytics. arxiv:1703.02952.
- [138] Sinno Jialin Pan and Qiang Yang. 2009. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* 22, 10 (2009), 1345–1359. DOI: <https://doi.org/10.1109/TKDE.2009.191>
- [139] Elias P. Papadopoulos, Michalis Diamantaris, Panagiotis Papadopoulos, Thanasis Petsas, Sotiris Ioannidis, et al. 2017. The long-standing privacy debate: Mobile websites vs mobile apps. In *WWW'17*. ACM, 153–162. DOI: <https://doi.org/10.1145/3038912.3052691>
- [140] Veljko Pejovic and Mirco Musolesi. 2014. InterruptMe: Designing intelligent prompting mechanisms for pervasive applications. In *UbiComp'14*. ACM, 897–908. DOI: <https://doi.org/10.1145/2632048.2632062>
- [141] Veljko Pejovic and Mirco Musolesi. 2015. Anticipatory mobile computing: A survey of the state of the art and research challenges. *Comput. Surv.* 47, 3 (2015). DOI: <https://doi.org/10.1145/2693843>
- [142] Zhenhui Peng, Yunhwan Kwon, Jiaan Lu, Ziming Wu, and Xiaojuan Ma. 2019. Design and evaluation of service robot's proactivity in decision-making support process. In *CHI'19*. ACM, 98:1–98:13. DOI: <https://doi.org/10.1145/3290605.3300328>
- [143] Alex Pentland. 2009. Reality mining of mobile communications: Toward a new deal on data. In *Social Computing and Behavioral Modeling*. Springer-Verlag.
- [144] Charith Perera, Susan Y. L. Wakenshaw, Tim Baarslag, Hamed Haddadi, et al. 2017. Valorising the IoT databox: Creating value for everyone. *Trans. Emerg. Telecomm. Technol.* 28, 1 (2017), e3125. DOI: <https://doi.org/10.1002/ett.3125>
- [145] Martin Pielot, Bruno Cardoso, Kleomenis Katevas, Joan Serrà, Aleksandar Matic, and Nuria Oliver. 2017. Beyond interruptibility: Predicting opportune moments to engage mobile phone users. *Proc. ACM Interact. Mob. Wear. Ubiqu. Technol.* 1, 3 (2017), 91. DOI: <https://doi.org/10.1145/3130956>
- [146] Benny Pinkas. 2002. Cryptographic techniques for privacy-preserving data mining. *ACM SigKDD Explor. Newslett.* 4, 2 (2002), 12–19. DOI: <https://doi.org/10.1145/772862.772865>
- [147] Otto Julio Ahlert Pinno, Andre Ricardo Abed Gregio, and Luis C. E. De Bona. 2017. ControlChain: Blockchain as a central enabler for access control authorizations in the IoT. In *GLOBECOM'17*. IEEE, 1–6. DOI: <https://doi.org/10.1109/GLOCOM.2017.8254521>
- [148] Moo-Ryong Ra, Ramesh Govindan, and Antonio Ortega. 2013. P3: Toward privacy-preserving photo sharing. In *USENIX NSDI'13*. 515–528.
- [149] Mashfiqui Rabbi, Min Hane Aung, Mi Zhang, and Tanzeem Choudhury. 2015. MyBehavior: Automatic personalized health feedback from user behaviors and preferences using smartphones. In *UbiComp'15*. ACM, 707–718. DOI: <https://doi.org/10.1145/2750858.2805840>
- [150] Kiran K. Rachuri, Cecilia Mascolo, Mirco Musolesi, and Peter J. Rentfrow. 2011. SociableSense: Exploring the trade-offs of adaptive sampling and computation offloading for social sensing. In *MobiCom'11*. ACM, 73–84. DOI: <https://doi.org/10.1145/2030613.2030623>
- [151] Kiran K. Rachuri, Mirco Musolesi, Cecilia Mascolo, Peter J. Rentfrow, et al. 2010. EmotionSense: A mobile phones based adaptive platform for experimental social psychology research. In *UbiComp'10*. ACM, 281–290. DOI: <https://doi.org/10.1145/1864349.1864393>
- [152] Dave Raggett. 2015. The web of things: Challenges and opportunities. *Computer* 48, 5 (2015), 26–32. DOI: <https://doi.org/10.1109/MC.2015.149>
- [153] Bitu Darvish Rouhani, M. Sadegh Riazi, and Farinaz Koushanfar. 2018. DeepSecure: Scalable provably-secure deep learning. In *DAC'18*. ACM. DOI: <https://doi.org/10.1145/3195970.3196023>
- [154] Stuart J. Russell and Peter Norvig. 2016. *Artificial Intelligence: A Modern Approach*. Pearson Education Limited.
- [155] Fariba Sadri. 2011. Ambient intelligence: A survey. *Comput. Surv.* 43, 4 (2011). DOI: <https://doi.org/10.1145/1978802.1978815>
- [156] Andrei Vlad Sambra, Essam Mansour, Sandro Hawke, Maged Zereba, Nicola Greco, Abdurrahman Ghanem, Dmitri Zagidulin, Ashraf Aboulnaga, and Tim Berners-Lee. 2016. *Solid: A Platform for Decentralized Social Applications Based on Linked Data*. Technical Report. MIT CSAIL & Qatar Computing Research Institute.
- [157] Ruhi Sarikaya. 2017. The technology behind personal digital assistants: An overview of the system architecture and key components. *IEEE Sig. Proc. Mag.* 34, 1 (2017), 67–81. DOI: <https://doi.org/10.1109/MSP.2016.2617341>
- [158] Mahadev Satyanarayanan. 2017. The emergence of edge computing. *Computer* 50, 1 (2017), 30–39. DOI: <https://doi.org/10.1109/MC.2017.9>

- [159] Benedikt Schmidt, Sebastian Benchea, Rüdiger Eichin, and Christian Meurisch. 2015. Fitness tracker or digital personal coach: How to personalize training. In *Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers*. ACM, 1063–1067. DOI : <https://doi.org/10.1145/2800835.2800961>
- [160] Philipp M. Scholl, Matthias Wille, and Kristof Van Laerhoven. 2015. Wearables in the wet lab: A laboratory system for capturing and guiding experiments. In *UbiComp'15*. ACM, 589–599. DOI : <https://doi.org/10.1145/2750858.2807547>
- [161] Felix Schuster, Manuel Costa, Cédric Fournet, Christos Gkantsidis, Marcus Peinado, Gloria Mainar-Ruiz, and Mark Russinovich. 2015. VC3: Trustworthy data analytics in the cloud using SGX. In *SP'15*. IEEE, 38–54. DOI : <https://doi.org/10.1109/SP.2015.10>
- [162] Klaus Schwab, Alan Marcus, J. O. Oyola, William Hoffman, and M. Luzi. 2011. Personal data: The emergence of a new asset class. Retrieved from http://www3.weforum.org/docs/WEF_ITTC_PersonalDataNewAsset_Report_2011.pdf.
- [163] Sandra Servia-Rodriguez, Liang Wang, Jianxin R. Zhao, Richard Mortier, and Hamed Haddadi. 2018. Privacy-preserving personal model training. In *IoT'18*. IEEE, 153–164. DOI : <https://doi.org/10.1109/IOTDI.2018.00024>
- [164] Hossein Shafagh, Lukas Burkhalter, Anwar Hithnawi, and Simon Duquennoy. 2017. Towards blockchain-based auditable storage and sharing of IoT data. In *CCSW'17*. ACM, 45–50. DOI : <https://doi.org/10.1145/3140649.3140656>
- [165] Peter Shaw, Mateusz Mikusz, Ludwig Trotter, Mike Harding, and Nigel Davies. 2019. Towards an understanding of emerging cyber security threats in mapping the IoT. In *IoT'19*. IET Digital Library. DOI : <https://doi.org/10.1049/cp.2019.0158>
- [166] Reza Shokri and Vitaly Shmatikov. 2015. Privacy-preserving deep learning. In *CCS'15*. ACM, 1310–1321. DOI : <https://doi.org/10.1145/2810103.2813687>
- [167] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *SP'17*. IEEE, 3–18. DOI : <https://doi.org/10.1109/SP.2017.41>
- [168] Yan Shvartzshnaider, Zvonimir Pavlinovic, Ananth Balashankar, Thomas Wies, Lakshminarayanan Subramanian, et al. 2019. VACCINE: Using contextual integrity for data leakage detection. In *WWW'19*. ACM, 1702–1712. DOI : <https://doi.org/10.1145/3308558.3313655>
- [169] Daniel J. Solove. 2005. A taxonomy of privacy. *U. Pa. L. Rev.* 154 (2005), 477.
- [170] Nili Steinfeld. 2016. "I agree to the terms and conditions": (How) do users read privacy policies online? An eye-tracking experiment. *Comput. Hum. Behav.* 55 (2016), 992–1000. DOI : <https://doi.org/10.1016/j.chb.2015.09.038>
- [171] Yu Sun, Nicholas Jing Yuan, Yingzi Wang, Xing Xie, Kieran McDonald, and Rui Zhang. 2016. Contextual intent tracking for personal assistants. In *KDD'16*. ACM, 273–282. DOI : <https://doi.org/10.1145/2939672.2939676>
- [172] Latanya Sweeney. 2002. k-anonymity: A model for protecting privacy. *Int. J. Uncert. Fuzz. Knowl.-based Syst.* 10, 5 (2002), 557–570. DOI : <https://doi.org/10.1142/S0218488502001648>
- [173] Jun Tang, Yong Cui, Qi Li, Kui Ren, Jiangchuan Liu, and Rajkumar Buyya. 2016. Ensuring security and privacy preservation for cloud data services. *Comput. Surv.* 49, 1 (2016). DOI : <https://doi.org/10.1145/2906153>
- [174] Daniel Teghe and Kathryn Rendell. 2005. *Social Well-being: A Literature Review*. School of Social Work and Welfare Studies, CQU, Rockhampton. DOI : <https://doi.org/10.13140/RG.2.2.28891.26406>
- [175] Eran Toch, Yang Wang, and Lorrie Faith Cranor. 2012. Personalization and privacy: A survey of privacy risks and remedies in personalization-based systems. *User Model. User-adapt. Interact.* 22, 1–2 (2012), 203–220. DOI : <https://doi.org/10.1007/s11257-011-9110-z>
- [176] Paul Tran-Van, Nicolas Anciaux, and Philippe Pucheral. 2017. SWYSWYK: A privacy-by-design paradigm for personal information management systems. In *ISD'17*. AIS.
- [177] Chia-Che Tsai, Donald E. Porter, and Mona Vij. 2017. Graphene-SGX: A practical library OS for unmodified applications on SGX. In *USENIX ATC'17*. 645–658.
- [178] Kenneth Tung. 2019. AI, the Internet of legal things, and lawyers. *J. Manag. Anal.* 6, 4 (2019), 390–403. DOI : <https://doi.org/10.1080/23270012.2019.1671242>
- [179] Robert van den Hoven van Genderen. 2017. Privacy and data protection in the age of pervasive technologies in AI and robotics. *Eur. Data Prot. L. Rev.* 3 (2017), 338. DOI : <https://doi.org/10.21552/edpl/2017/3/8>
- [180] Max Van Kleek and Kieron O'Hara. 2014. The future of social is personal: The potential of the personal data store. In *Social Collective Intelligence*. Springer, 125–158. DOI : https://doi.org/10.1007/978-3-319-08681-1_7
- [181] Ricardo Vilalta and Youssef Drissi. 2002. A perspective view and survey of meta-learning. *Artif. Intell. Rev.* 18, 2 (2002), 77–95. DOI : <https://doi.org/10.1023/A:1019956318069>
- [182] Isabel Wagner and David Eckhoff. 2018. Technical privacy metrics: A systematic survey. *Comput. Surv.* 51, 3 (2018). DOI : <https://doi.org/10.1145/3168389>
- [183] Huaqing Wang, Matthew K. O. Lee, and Chen Wang. 1998. Consumer privacy concerns about Internet marketing. *Commun. ACM* 41, 3 (1998), 63–70. DOI : <https://doi.org/10.1145/272287.272299>
- [184] Jiaqiu Wang and Zhongjie Wang. 2014. A survey on personal data cloud. *Sci. World J.* 2014 (2014). DOI : <https://doi.org/10.1155/2014/969150>

- [185] Shiqiang Wang, Tiffany Tuor, Theodoros Salonidis, Kin K. Leung, Christian Makaya, et al. 2018. When edge meets learning: Adaptive control for resource-constrained distributed machine learning. In *INFOCOM'18*. IEEE, 63–71. DOI: <https://doi.org/10.1109/INFOCOM.2018.8486403>
- [186] Xiaofei Wang, Yiwen Han, Chenyang Wang, Qiyang Zhao, Xu Chen, and Min Chen. 2019. In-edge AI: Intelligentizing mobile edge computing, caching and communication by federated learning. *IEEE Netw.* 33, 5 (2019), 156–165. DOI: <https://doi.org/10.1109/MNET.2019.1800286>
- [187] Samuel D. Warren and Louis D. Brandeis. 1890. Right to privacy. *Harv. L. Rev.* 4 (1890), 193.
- [188] Mark Weiser. 1991. The computer for the 21st century. *Sci. Amer.* 265, 3 (1991), 94–105.
- [189] Alan F. Westin. 1968. Privacy and freedom. *Wash. Lee Law Rev.* 25, 1 (1968), 166. DOI: <https://doi.org/10.1093/sw/13.4.114-a>
- [190] Jason Wiese, Sauvik Das, Jason I. Hong, and John Zimmerman. 2017. Evolving the ecosystem of personal behavioral data. *Hum.-comput. Interact.* 32, 5–6 (2017), 447–510. DOI: <https://doi.org/10.1080/07370024.2017.1295857>
- [191] Depeng Xu, Shuhan Yuan, and Xintao Wu. 2019. Achieving differential privacy and fairness in logistic regression. In *WWW'19 Comp.* ACM, 594–599. DOI: <https://doi.org/10.1145/3308560.3317584>
- [192] Heng Xu, Xin Robert Luo, John Carroll, and Mary Rosson. 2011. The personalization privacy paradox: An exploratory study of decision making process for location-aware marketing. *Decis. Supp. Syst.* 51, 1 (2011), 42–52. DOI: <https://doi.org/10.1016/j.dss.2010.11.017>
- [193] Zhi Xu and Sencun Zhu. 2015. SemaDroid: A privacy-aware sensor management framework for smartphones. In *CODASPY'15*. ACM, 61–72. DOI: <https://doi.org/10.1145/2699026.2699114>
- [194] Zhu Yan, Guhua Gan, and Khaled Riad. 2017. BC-PDS: Protecting privacy and self-sovereignty through blockchains for OpenPDS. In *SOSE'17*. IEEE, 138–144. DOI: <https://doi.org/10.1109/SOSE.2017.30>
- [195] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. 2019. Federated machine learning: Concept and applications. *ACM Trans. Intell. Syst. Technol.* 10, 2 (2019), 12:1–12:19. DOI: <https://doi.org/10.1145/3298981>
- [196] Neil Yorke-Smith, Shahin Saadati, Karen L. Myers, and David N. Morley. 2012. The design of a proactive personal agent for task management. *Int. J. Artif. Intell. Tools* 21, 01 (2012), 1250004. DOI: <https://doi.org/10.1142/S0218213012500042>
- [197] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P. Gummadi. 2017. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *WWW'17*. 1171–1180. DOI: <https://doi.org/10.1145/3038912.3052660>
- [198] Yan Zhai, Lichao Yin, Jeffrey Chase, Thomas Ristenpart, and Michael Swift. 2016. CQSTR: Securing cross-tenant applications with cloud containers. In *SoCC'16*. ACM, 223–236. DOI: <https://doi.org/10.1145/2987550.2987558>
- [199] Rui Zhang, Rui Xue, and Ling Liu. 2019. Security and privacy on blockchain. *Comput. Surv.* 52, 3 (2019). DOI: <https://doi.org/10.1145/3316481>
- [200] Wenting Zheng, Ankur Dave, Jethro G. Beekman, Raluca Ada Popa, Joseph E. Gonzalez, and Ion Stoica. 2017. Opaque: An oblivious and encrypted distributed analytics platform. In *USENIX NSDI'17*. 283–298.
- [201] Zhi Zhou, Xu Chen, En Li, Liekang Zeng, Ke Luo, and Junshan Zhang. 2019. Edge intelligence: Paving the last mile of artificial intelligence with edge computing. *Proc. IEEE* 107, 8 (2019), 1738–1762. DOI: <https://doi.org/10.1109/JPROC.2019.2918951>
- [202] Qingyi Zhu, Seng W. Loke, Rolando Trujillo-Rasua, Frank Jiang, and Yong Xiang. 2019. Applications of distributed ledger technologies to the Internet of Things: A survey. *Comput. Surv.* 52, 6 (2019). DOI: <https://doi.org/10.1145/3359982>
- [203] M. Tarek Ibn Ziad, Amr Alanwar, Moustafa Alzantot, and Mani Srivastava. 2016. CryptoImg: Privacy preserving processing over encrypted images. In *CNS'16*. IEEE, 570–575. DOI: <https://doi.org/10.1109/CNS.2016.7860550>
- [204] Philip R. Zimmermann. 1995. *The Official PGP User's Guide*. Vol. 5. The MIT Press, Cambridge, MA.
- [205] Guy Zyskind, Oz Nathan, and Alex Pentland. 2015. Enigma: Decentralized Computation Platform with Guaranteed Privacy. arxiv:1506.03471

Received March 2020; revised October 2020; accepted December 2020