DOI:
10.1145/3514094.3534129

Document Version
Peer reviewed version

A Model for Governing Information Sharing in Smart Assistants

Xiao Zhan
King's College London
London, United Kingdom
xiao.zhan@kcl.ac.uk

Stefan Sarkadi
King's College London
London, United Kingdom
stefan.sarkadi@kcl.ac.uk

Natalia Criado
Universitat Politècnica de València
Valencia, Spain
ncriado@upv.es

Jose Such
King's College London
London, United Kingdom
jose.such@kcl.ac.uk

ABSTRACT

Smart Personal Assistants (SPAs), such as Amazon Alexa, Google Assistant and Apple Siri, leverage different AI techniques to provide convenient help and assistance to users. However, inappropriate information sharing decisions can lead SPAs to incorrectly disclose user information to undesired parties, or mistakenly block their reasonable access in specific scenarios to desired parties. In fact, reports about privacy violations in SPAs and associated user concerns are well known and understood in the related literature. It is difficult for SPAs to automatically decide how data should be shared with respect to the privacy preferences of the users. We argue norms, which are regarded as shared standards of acceptable behaviour of groups and/or individuals, can be used to govern and reason about the best course of action of SPAs with regards to information sharing, and our work is the first to propose a practical model to address the above issues and govern SPAs based on normative systems and the contextual integrity theory of privacy. We evaluated the performance of the model using a real dataset of user preferences for privacy in SPAs and the results showed a very marked and significant improvement in understanding user preferences and making the right decisions with respect to data sharing.

CCS CONCEPTS

• **Computing methodologies** → **Artificial intelligence**; **Machine learning**; • **Security and privacy** → **Usability in security and privacy**; **Privacy protections**.

KEYWORDS

privacy, smart personal assistants, voice assistants, personal data, data protection

ACM Reference Format:
Xiao Zhan, Stefan Sarkadi, Natalia Criado, and Jose Such. 2022. A Model for Governing Information Sharing in Smart Assistants. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society (AIES'22), August 1–3, 2022, Oxford, United Kingdom.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Conference'17, July 2017, Washington, DC, USA
2022. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Recent advances in AI, such as in Natural Language Processing among others, have finally made the promise of intelligent personal assistants a reality and are helping unravel their potential. In particular, voice-based Smart Personal Assistants (SPAs), such as Amazon Alexa, Apple Siri, Microsoft Cortana, Google Assistant, and the like, are utilized by millions of users around the world, with 147 million SPA sold in 2019 [48], and their use is only predicted to increase in the next few years. SPAs now perform many tasks on behalf of their users, including buying groceries, playing music, controlling smart home devices, etc. [29, 55]. In fact, the list of SPAs capabilities, called skills in Amazon Alexa, keeps growing, having already surpassed 100,000 skills in Alexa [54].

Not having regulatory mechanisms for these AI systems would let them wreak havok on the infosphere, causing a Tragedy of The Digital Commons [23] by polluting the information shared, deleting valuable information, spreading false information [45], discriminating users [21], or, as it is in the focus of this paper, violating the right to individual's privacy[1]. A right that is now part of regulations around the globe[2] and a key aspect of ethical principles for AI such as the Asilomar principles[3]. This is all the more important, given the concerns users have about SPAs invading their privacy [1, 25, 27, 30], exacerbated by media reports and studies demonstrating privacy issues in SPAs [17–20, 24, 46].

Previously proposed approaches for privacy-respecting smart devices, such as access control mechanisms [59], do not effectively address these issues, as it is unrealistic to ask users to educate SPAs on how to respond to all scenarios, users just expect smart devices to learn the appropriate privacy norms without putting considerable human effort [59]. Therefore, there is a need for SPAs to be improved to automatically learn privacy norms and regulate their behaviours with respect to the expectations of users. In addition, and importantly, to ensure SPAs are truly aligned with humans values from a normative perspective, then we must be able to check what norms they learn from us and explain how they reason about these norms in different privacy contexts. To do this, some form of explicit privacy norm representation is required, something difficult to achieve with subsymbolic AI approaches such as traditional

[1] Article 12 of the UN Universal Declaration of Human Rights https://www.un.org/en/about-us/universal-declaration-of-human-rights.
[2] Including the EU with the GDPR, California with CCPA, and regulations in other countries like Canada and the UK.
[3] https://futureoflife.org/ai-principles/

machine learning approaches. So, how do we represent these privacy norms in different contexts? More specifically, what are the contextual aspects that we must consider to represent norms in terms of privacy? And, how do we learn and reason about these norms so SPAs can better align with users' privacy expectations?

In this paper, we first present the formalization of privacy norms to govern the behavior of SPAs based on normative systems and Contextual Integrity [41, 42]. Based on this formalization, we also present a model to learn and represent privacy norms and to reason about their relevance and applicability in different contexts, allowing SPAs to make decisions about information flows. Finally, we show experimentally the performance of the model with a real dataset of user-provided acceptability of different information flows in SPAs in different contexts.

The structure of the paper is as follows. We first present, in Section 2, background on contextual integrity and normative systems, and then present the norm definitions used by our model. We illustrate the formalization of our normative model and reasoning method in Section 3 respectively. In Section 4, we describe the model evaluation combined within a specific case study. Finally, we discuss and conclude the paper in Section 6.

2 PRIVACY NORMS

Norms, such as prohibitions and permissions, are used in the existing literature to monitor and control the behaviour of agents [11]. Previous studies [5, 26, 28] expressed those norms using logic statements. In particular, norms usually define patterns of behaviour by means of deontic logic and its modalities: Obligation, which defines the action that the agent should perform; Permission, which defines the behavior that the SPA is allowed to perform; and Prohibition, which defines the action that should not be performed by the SPA. This is has been extensively used in multi-agent domains, where norms are the socially agreed-upon codes of behaviour that most members of a society can understand. Norms can also represent the most acceptable behavior, and thus play a significant role in decision making. Another advantage of using norms is that they can be used for logical reasoning. Hence, when faced with an unspecified situation, we can also reason about the appropriate action based on what we know. The question is how to conceptualize privacy in order to define privacy norms for AI agents such as SPAs.

Separately, and independently, the field of privacy has been evolving over the past years under different conceptualizations. The most modern of those conceptualizations is the theory of Contextual Integrity (CI) [41, 42], which defines privacy as the appropriateness of information flows based on social and cultural norms in a specific given context. That is, the same information sharing action may lead to a privacy violation or not depending on the context. For instance, one may not give their health details to a passer by but may do so to a doctor so they can treat them. CI describes the information flows using five parameters: (1) the sender of the information, (2) the attribute or type of the information, (3) the subject of the information that is being transferred, (4) the recipient of the information, and (5) the transmission principles or conditions imposed on the transfer of the information from the sender to the recipient.

CI has been known as an appropriate framework to elicit expected privacy norms in different contexts [3, 4, 31, 47]. This is usually done through surveys varying the contextual parameters in Contextual Integrity, asking users for the acceptability of the information flows described in a vignette using the contextual parameters. That is, how acceptable it is that one specific sender sends information of a distinct type and specific subject to a specific recipient using (not using) a set of transmission principles (e.g., for a given purpose). However, CI lacks a formal definition, and the types of norms usually elicited cannot be properly represented or reasoned about.

In this work, we bridge the gap between normative systems and contextual integrity to define privacy norms for SPAs. For this, we first introduce some preliminary definitions in Section 2.1 and then define the privacy norms themselves in Section 2.2. After this, we use these definitions in Section 3 to present our model to learn and reason about privacy norms for SPAs.

2.1 Preliminary Definitions

We begin with defining the terms and notation used throughout the rest of the paper. We assume that the SPA has a representation of *the state of the word* in which it operates. To that aim, we make use of a finite set of predicate and constant symbols, that characterise the properties of the world relevant to the SPA. In this paper we write predicate and constant symbols starting with a lower case letter and variables starting with a capital letter. We also make use of the true (vs. false) predicate $\top$ (vs. $\bot$). By a state of the SPA (denoted by s), we mean the properties of the world that are true at a particular moment; i.e., a state is built on a "closed world assumption" and defined by a set of properties (i.e., a set of grounded predicates) that hold at a given moment.

We define that a state s satisfies a property represented as an atomic grounded predicate l, denoted by $s \vdash l$, iff $l \in s$. Similarly, s satisfies the negation of a property $\neg l$, denoted by $s \nvdash l$ iff $l \notin s$. We extend this definition to sets of properties (note the empty set is always satisfied). We also give the common semantics to the true (vs. false) predicates; i.e., for any state s, $s \vdash \top$ (vs. $s \nvdash \bot$).

2.1.1 Predicates. For the purpose of regulating SPAs actions, we define a set of predicates[4]. These predicates are used to represent the state of SPAs and they are added to and removed from the state representation in accordance with the SPA perceptions. Perceptions are formed by observations from the physical environment — e.g., identification of the user interacting with the SPA; observations from the software environment — e.g., notifications and logging information from the third-party skills; and the effects of the actions executed by the SPA. For the purpose of this paper, let's assume the following two predicates:

- $achieved(Purpose)$, which is a unary predicate representing that a given $Purpose$ (e.g., "to improve the functionality") has been achieved.
- $anonymised(Data)$ is also a unary predicate indicating the data element represented by $Data$ has been anonymised.

[4]In our paper, we take into consideration the "tenses" of predicates and assign them various meanings.

Note, we focus on the information being transmitted and processed by a SPA, which is the data. We consider the data is not only a collection of information such as observations, measurements and facts but also contains metadata which provides granular information such as file type, format, origin, date, etc. In this case, it is obviously that metadata could be used to discover the personal identity of the owner (subject) of the data. Information is normally transmitted in the form of complete data but metadata will be separated only in special circumstances, for example, when an anonymous transmission is required.

2.1.2 Actions. In our paper, we mainly concentrate on all the actions of the SPA in the execution process. Since it is often the case that the actions have been taken in a given contextual environment, we embed contextual-related parameters (such as the sender of the action, the purpose of performing the action, etc.) into the definition.

*Definition 2.1 (**Action**).* Formally, we represent an action as an atomic sentence that contains a combination of both name and parameters: $name(parameter_1, parameter_2, ..., parameter_n)$, where:

- *name* identifies the action;
- each $parameter_i$ is a constant value representing the specific context in which the action takes place. For example, the actor of the action, the recipient of the action, and so on.

The primary action involved in this paper is always that a SPA sends various data on behalf of the users. In addition, the SPA can also implement auxiliary actions such as deleting data in accordance with the wishes of users. In order to fully reflect the entire world of the SPA, we propose the definition of the following actions.

Action (Send). The action of sending is defined as: $Send(Actor, Target, Attribute, Data, Subject, Purpose)$, represents that the $Data$ corresponding to the $Attribute$ of the $Subject$ is sent by the $Actor$ to the $Target$ with the a particular $Purpose$. Where $Actor$ is the principle who send the information[5], $Target$ illustrates the recipient. $Attribute$ is the type of information i.e. playlist, $Data$ is the specific data being processed which corresponding to the $Attribute$, $Subject$ is the user about whom information is sent. The last parameter in this definition is $Purpose$, representing the specific purpose for sending the data, or constant value *none* if there is none.

Example 1. *The following example is used to present a scenario that neighbours can access to the playlists without providing a specific reason.*

$$send(spa, neighbour, music, playlists, bob, none)$$

Action (Delete). The action $delete$ is defined as $delete(Actor, Attribute, Data, Subject)$, where $Subject$ is the person who currently owns this data and $Actor$ is responsible for performing this delete action. $Attribute$ and $Data$ are defined as before.

Action (Review). We denote the action that data can be reviewed as $review(Actor, Attribute, Data)$ where $Actor$ is the subject who owns the data and has the authorization to view the data. $Attribute$ and $Data$ are defined as before.

[5]In our paper, the $Actor$ is always the SPA that performs tasks on behalf of users.

Action (Notify). This action was defined as $notify(Actor, Subject, Action)$, indicating that the $Actor$ notifies the $Subject$ that the $Action$ will occur.

Finally, and for each of the actions defined above, there is an associated predicate representing that the action has been executed. For instance, when the action $review(Actor, Data)$ is executed then the predicate $reviewed(Actor, Data)$ is inserted into the knowledge base — which is defined in Section 3.

2.2 Norm and Norm Instance Definitions

We now define privacy norms for SPAs based on normative systems and contextual integrity. We start with the definition of a norm, and then with the definition of when a norm becomes instantiated given a state of the world, and hence becomes relevant to the state of the world.

*Definition 2.2 (**Norm**).* We define a norm n as a tuple $n = \langle Deontic, Condition, Action, Effect, \delta \rangle$, where:

- $Deontic$ represents deontic modality, namely *Obligations (O), Permissions (P) and Prohibitions(F).*
- $Condition$ is a set of literals (i.e., a predicate and its negation) that can contain variables. The condition represents the situations in which the norm is applicable (i.e., has effect).
- The $Action$ expresses the action regulated by the norm, i.e. what is being permitted, etc.
- $Effect$ is a set of norms representing the SPA duties after performing the $Action$.
- $\delta \in [0, 1]$ is a real number representing the importance of the norm. Norm with a higher δ value also means that it has a higher priority in model decision making.

Example 2. *A norm regulates that it is an obligation to send the door locker logs to law enforcement agencies can be represented as:*

$$\langle O, \emptyset, send(spa, law\_enforcement\_agencies, smart\_door, door\_locker\_logs, primary\_user, none), \emptyset, \delta \rangle$$

Note that norms can have variables in their definition. When their condition holds, then they are applicable and norm instances (or instances for short) are created, according to the possible groundings of the activation condition.

*Definition 2.3 (**Norm Instance**).* Given a norm $n = \langle Deontic, Condition, Action, Effect, \delta \rangle$ and a state of the world s, we say that the norm n is *instantiated* in s if it exists a substitution σ of variables in $Condition$ such that $s \vdash \sigma(Condition)$. Then, we define $i = \langle Deontic, \sigma(Condition), \sigma(Action), \sigma(Effect), \delta \rangle$ as a norm instance.

Note that when the condition of the norm is undefined — i.e, when $Condition$ is $\emptyset$, the norm is instantiated by default. Also note that norm instances may also contain variables.

2.3 Transmission Principles

In Contextual Integrity, transmission principles are a special type of parameter that impose conditions on the information shared or to be shared. Some of these principles are descriptions of whether the recipients have a reasonable purpose for getting the data, while others include the confidentiality that prohibits the recipient from

sharing the data with others in the future, the awareness and consent of the information subject, etc. In this paper, we mainly address the five principles that are representative of previous research on contextual integrity [3, 7, 42]: *(1) if the subject of the information is notified about an information flow; (2) if the data shared is anonymous; (3) if the data is kept confidential, i.e., not shared further with others; (4) if the data is only stored as long as necessary for the purpose it is shared; (5) if the user can later review or delete the data shared.* Nevertheless, our method could be extended to be compatible with the presentation further kinds of principles of transmission. We propose five templates to express the various transmission principles. Note that the templates are not actual norms but only used for creating the norms.

Norm Template (Transmission Principle 1). As for the transmission principle '*If you are notified*', we define the following formula to represent it. We emphasise that the owner of the information is notified that this action will take place before the data is sent.

$$\begin{aligned}\langle Deontic, \{notified(Actor, Subject, Action)\},\\ send(Actor, Target, Attribute, Data,\\ Subject, Purpose), \emptyset, \delta\rangle\end{aligned} \tag{1}$$

Example 3. *Here is the example if the user agrees to share their sugar reading results with partner, but only if the user receives a notification prior to the action:*

$$\begin{aligned}\langle P, \{notified(spa, primary\_user, send(spa, partner,\\ healthcare, sugar\_reading, primary\_user, none))\},\\ send(spa, partner, healthcare, sugar\_reading,\\ primary\_user, none), \emptyset, \delta\rangle\end{aligned}$$

Norm Template (Transmission Principle 2). As for the transmission principle '*If the data is anonymous*', we assume that the data to be transmitted must be anonymised, and that precondition must first be satisfied.

$$\begin{aligned}\langle Deontic, \{anonimised(Data)\}, send(Actor, Target,\\ Attribute, Data, Subject, Purpose), \emptyset, \delta\rangle\end{aligned} \tag{2}$$

Example 4. *For example, if Bob, who is the primary user, he would like to share his shopping data to the advertisement agencies as long as the data is anonimised.*

$$\begin{aligned}\langle P, \{anonimised(shopping\_list)\}, send(spa, ads\_agency,\\ shopping, shopping\_list, bob, none), \emptyset, \delta\rangle\end{aligned}$$

Norm Template (Transmission Principle 3). As for the transmission principle '*If the data is kept confidential*', we assume that when the recipient receives the data, he/she cannot forward the data to other parties. In other words, other parties are not able to access this data.

$$\begin{aligned}\langle Deontic, \emptyset, send(Actor, Target, Attribute, Data,\\ Subject, Purpose), \langle F, \emptyset, send(Actor, Target,\\ Attribute, Data, Subject, Purpose), \emptyset, \delta'\rangle, \delta\rangle\end{aligned} \tag{3}$$

Here, the value of $Actor$ is equivalent to the value of $Target$, see the example below:

Example 5. *When skill receives the healthcare data, it cannot be forwarded to other parties, hence makes sure that data can only be shared with the authorized recipient (skill).*

$$\begin{aligned}\langle P, \emptyset, send(spa, skill, healthcare, healthcare\_data,\\ primary\_user, research), \langle F, \emptyset, send(skill, \_, healthcare,\\ healthcare\_data, primary\_user, none), \emptyset, \delta'\rangle, \delta\rangle\end{aligned}$$

Norm Template (Transmission Principle 4). We use the following formula to represent the fourth transmission principle *'If the data is stored as long as necessary for the purpose'*:

$$\begin{aligned}\langle Deontic, \emptyset, send(Actor, Target, Attribute, Data,\\ Subject, Purpose), \langle O, \{achieved(Purpose)\},\\ delete(Actor, Attribute, Data, Subject), \emptyset, \delta'\rangle, \delta\rangle\end{aligned} \tag{4}$$

Example 6. *Alice agrees to share her bank information to the assistant to help develop system, as long as the assistant deletes her information after the purpose is achieved.*

$$\begin{aligned}\langle P, \emptyset, send(spa, assistant\_provider, finance,\\ bank\_account, Alice, system\_updating), \langle O,\\ \{achieved(system\_updating)\}, delete(assistant\_provider,\\ finance, bank\_account, Alice), \emptyset, \delta'\rangle, \delta\rangle.\end{aligned}$$

Norm Template (Transmission Principle 5). We use the following formula to represent the fifth transmission principle *'If you can review/or delete the data'*[6]:

$$\begin{aligned}\langle Deontic, \emptyset, send(Actor, Target, Attribute,\\ Data, Subject, Purpose), \langle P, \emptyset,\\ review(Actor, Attribute, Data), \emptyset, \delta'\rangle, \delta\rangle\end{aligned} \tag{5}$$

Example 7. *We assume that Bob would like to share his fitness data if he can review it.*

$$\begin{aligned}\langle P, \emptyset, send(spa, friends, fitness, exercise\_routine,\\ Bob, none), \emptyset), \langle P, \emptyset, review(Bob, exercise, exercise\_routine),\\ \emptyset, \delta'\rangle, \delta\rangle\end{aligned}$$

3 NORMATIVE SUPERVISION MODEL

In this paper, we propose a normative supervision model that can help SPAs reason about information sharing decisions. To this aim, it needs to: i) manage the knowledge base of norms, and ii) use to make decisions about action performance based on the norms. Managing the knowledge base and ensuring that it is constantly well-updated and operational is the foundation for providing the SPA with the knowledge it requires to make decisions. The two features of the model operate independently of each other, i.e. the knowledge base is updated and runs *independently* of the decision-making process. When the SPA needs to make a decision, it then considers the knowledge it has on the knowledge base about the applicable norms. We detail the management of the knowledge base in Section 3.1, and present the process for making decisions considering the knowledge base in Section 3.2.

[6]For delete, just change action $review$ to $delete$.

3.1 Knowledge Base

Our supervision model assumes a Knowledge Base that contains an explicit representation of the relevant norms and instances as well as the feedback the SPA may gather directly from the user. We formally define the knowledge base as follows.

*Definition 3.1 (**Knowledge Base**).* The Knowledge Base is a tuple $KB = \langle N, I, E, L \rangle$, where N is a set of norms, I is a set of norm instances, E is a set of norms representing user feedback, and L is a hybrid learning and reasoning function about user feedback.

Figure 1 illustrates the structure of the knowledge base and the basic relationship between its components. We now describe its of the elements of the knowledge base and the relationship between them.

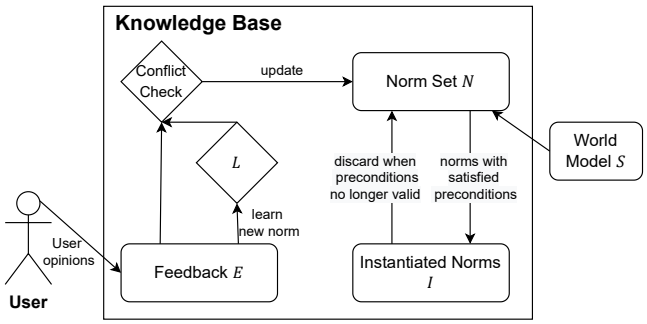

Figure 1: Knowledge Base and its management.

3.1.1 Norm Creation.

In this paper, we use N to express the norm set, which contains the norms used to govern the behaviour of the SPA. To avoid the well-known *cold start* problem in the absence of user feedback at the beginning, we assume that N starts with a default set of norms. These can be established in a variety of methods, product manufacturers, for example, could set certain norms manually, and users could customise them manually before using the SPA. In order to reduce the amount of effort needed to set the default norms in N, one could base on user studies to elicit norms that are widely accepted by users. For instance, there have been many previous studies that, using contextual integrity, define a large set of situations and conduct large-scale studies of the information flows users may find acceptable or not [2–4, 31, 47]. Given that low-level information, previous work has shown one could easily abstract it into more general if-then rules using unsupervised machine learning methods, such as association rule mining, to distil general norms accepted by the vast majority of users [2]. In this paper, as detailed in the experimental section, we use such an approach to extract default general norms to populate the set N at the beginning, when user feedback is yet to be had. Note, however, that this initial set of default norms is updated as feedback from the user is available, as detailed later in this section.

3.1.2 Norm Instantiation.

In each time step, the SPA updates the representation it has of the current state s according to its perceptions. Hence, properties are removed and added as needed. Similarly, new properties corresponding to the actions performed are added. Then, the SPA checks the norm set to determine which of the norms apply to the current situation; i.e., which norms are active according to the current situation. In each state, new instances are added to I, corresponding to the norms whose condition is meet; and existing instances are removed from I, corresponding to those instance whose condition is no longer satisfied. The whole process is described in Alg. 1.

Algorithm 1 Norm instantiation

Require: Knowledge Base KB
Require: World model S
Ensure: Updated KB
1: **while** true **do**
2: $P := perceive()$
3: $s := update(s, P)$
4: $I := \emptyset$
5: **for** each $\langle Deontic, Condition, Action, Effect, \delta \rangle \in N$ **do**
6: **if** $\exists \sigma : s \vdash \sigma(Condition)$ **then**
7: $I := I \cup \{\langle Deontic, \sigma(Condition), \sigma(Action), \sigma(Effect), \delta \rangle\}$

Example 8. *A norm regulates that visitors do not have access to ANY information flow ($\langle F, \emptyset, send(spa, visitor, \_, \_, \_), \emptyset, \delta \rangle$) is instantiated by default, and the corresponding instance is always stored in set I.*

3.1.3 Norm Update.

The norm set N is updated as feedback from the user becomes available in E. The SPA can gather feedback from the user in a number of ways, the simplest being to ask the user directly for their opinion on a scenario or to confirm an information flow. Since user feedback is an opinion on a very specific context, it can be formalised as a fully specified norm. At the beginning, the feedback set is empty ($E = \emptyset$). After receiving the feedback, it can be represented as $E = \{n_1, n_2, ..., n_m\}$. To distinguish norms created by collecting user feedback from those already stored in the norm set N, we refer to norms stored in the feedback set E as *feedback norms*.

The update of the norm set N consists of two steps. First, the model checks regularly for new user feedback. Second, the any new norms arising from user feedback are checked for conflicts with norms already in N, and any such conflicts are resolved to adequately update N.

Step 1. Creating new norms from user feedback. In the first step, all the new norms in E gathered from user feedback are considered. In addition, any other norms that can be inferred from those in E will also be considered using the following learning and reasoning function:

*Definition 3.2 (**Function** L).* The function L is a function that, given a set of feedback norms $E = \{n_1, n_2, ..., n_m\}$, reasons about E and generates a set of new norms $L(E)$.

The main objective of function L is to find patterns in the user feedback. For example, if a user allowed the SPA to share his/her location with their partner, parents and children, this may imply that the user generally does not mind that people living in the

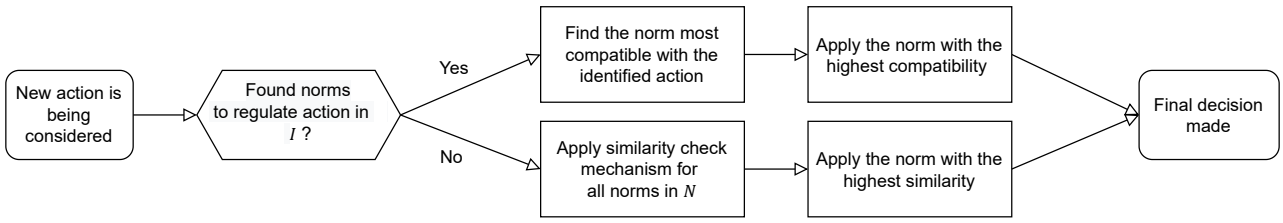

Figure 2: Decision mechanism

Algorithm 2 Norm Set Update

Require: A set of norms N
Require: A set of feedback norms E
Ensure: Updated Norm set
1: **while** true **do**
2: $E' := updateFeedback()$
3: $N' := E'$
4: **if** $\theta \vdash \top$ **then**
5: $N' := N' \cup \{L(E)\}$
6: **for** each $n' \in N'$ **do**
7: **if** $\exists n_j \in N : conflict(n', n_j)$ **then**
8: $N := (N \setminus \{n_j\}) \cup \{n'\}$
9: **else**
10: $N := N \cup \{n'\}$
11: **for** each $n' \in N'$ **do**
12: **if** $\exists n_j \in E : conflict(n', n_j)$ **then**
13: $E := (E \setminus \{n_j\}) \cup \{n'\}$
14: **else**
15: $E := E \cup \{n'\}$

same house know his/her location. Therefore, we infer that this user's location can also be shared to his/her cousin who also lives in the same house. In principle, this can be achieved through a variety of methods. For instance, machine learning approaches such as argument mining and Kalman filters can be embedded in the model to predict over time users' opinions on information sharing flows that have not yet been regulated. In Section 4 we showcase how this can be done using the templates together with examples. Finally, L can be executed regularly when a given condition θ. That condition can represent that a given time interval has elapsed, a certain amount of feedback already received, etc.

Step 2. Detecting and Resolving Conflicts. The second step is to check whether the new norms to be added conflict with a norm that already exists in the norm set. Conflict can be defined in many ways, depending on the context and role of the SPA holds. Accordingly, the rules for conflict detection can be strict or broad. In this paper, we define that a conflict arises when an action is simultaneously prohibited and permitted/obliged, and its variables have overlapping values.

*Definition 3.3 (**Norm conflict**).* Norms n_i, n_j are in conflict, denoted as $conflict(n_i, n_j) \iff$:

$$n_i = \langle Deontic, Condition, Action, Effect, \delta \rangle$$
$$n_j = \langle Deontic', Condition', Action', Effect', \delta \rangle$$
$$opposite(Deontic, Deontic')$$
$$equal(Condition, Condition')$$
$$equal(Action, Action')$$
$$equal(Effect, Effect')$$

That is, a conflict occurs between a prohibition and either an obligation or a permission if the values of the other norm elements (except δ) are equal. Besides, two norms that are not in conflict with each other can be represented as $\neg conflict(n_i, n_j)$.

Taking the practical meaning of adding new feedback norms to the norm set into consideration, they represent the latest opinions of users on a particular flow of information, i.e, stop sharing playlist to friends. If there is a norm stored in the norm set N, that specifies totally opposite meaning, i.e, friends can access to the playlist. We will use the newest feedback norm to replace the one stored in the N that is conflict with it. The whole process of checking for conflicts and resolving them is shown in Alg. 2 line 7 - 10.

In addition to updating the norm set, when collecting opinions from users, the model should update the feedback set by checking if there is any conflict between the new feedback norm e' and each e stored in the feedback set. If conflict exists, the model will replace e with e', otherwise, the model add the new feedback norm e' directly to the feedback set E, thus completing the update. The whole process of checking for conflicts and resolving them is shown in Alg. 2 line 12 - 18.

3.2 Decision Making Mechanism

Once put into use, the model makes decisions, that is, *permit* or *prohibit* the actions under consideration by the SPA based on the user expectations/preferences. To accomplish this, the model considers the norms that may apply given a particular state of the world. The whole process of decision mechanism (deliberation) is shown in Figure 2. The decision protocol is the following:

(1) Best case scenario - SPA uses a fully instantiated norm which is relevant to the action (Algorithm 3 line 10 - 21).
(2) Worst-case scenario - SPA applies similarity checking mechanism to find the most similar norm available for the action (Algorithm 3 line 22 - 39).

By instantiated norms which is relevant to the action, we mean the action defined in the norm express the same as the action detected. See details in Alg. 3 line 1-9. The best scenario is that the model can find instantiated norms that are relevant and can be

applied to regulate the observed action, otherwise, the model will invoke the similarity checking mechanism.

Example 9. *The action considered by the SPA is* $send(spa, friends, fitness, exercise\_routine, Bob, none)$. *Three instantiated norms are found:* $\langle P, \emptyset, send(spa, friends, fitness, exercise\_routine, Bob, none), \langle P, \emptyset, review(Bob, fitness, exercise\_routine), \emptyset, \delta' \rangle, \delta \rangle$; $\langle F, \emptyset, send(spa, visitor, \_, \_, \_), \emptyset, \delta \rangle$; *and* $\langle F, \emptyset, send(spa, friends, \_, \_, \_), \emptyset, \delta \rangle$. *Only the first and last norm found are relevant to the action.*

3.2.1 Best case scenario. This happens when there is at least one instantiated norm in the set I that is relevant to the action being considered. If more than one such norm exists, the one that is most compatible is chosen, i.e., the norm that is more specific to the action at hand (*lex specialis*). If there are more than one norm that are the most compatible then the most important, that is, the one with the highest δ, prevails.

Example 10. *From Example 9, two instantiated norms are relevant to the action detected, then the model will select the norm which is more compatible with the action (contains more similar parameters to the action). If there are two norms with the same compatibility, the model will apply the one with the higher importance value (*δ*). In this case, the model will select the first norm to make the decision.*

3.2.2 Worst case scenario. When the model is not able to find any instantiated norms that is relevant to the action, it will invoke the similarity checking mechanism using the norm set N. The objective of this mechanism is to find a norm that captures the most similar context to the action. In particular, we assume that there is a similarity function $sim : Action \times Action \rightarrow [0, 1]$ that, given two actions, returns their similarity. In this way, given action a under consideration, the model would pick the norm $n_1 \in N$ that has action a_1 so that there is no other norm $n_2 \in N$ with action a_2 and $sim(a, a_2) > sim(a, a_1)$. The whole process is described in Alg. 3 line 22 - 39. Our model is agnostic to the similarity function used for that regard. For instance, it could be based on semantic similarity and/or the particular domain. In the evaluation section, we described the similarity function used for the particular experiment setup.

Algorithm 3 Decision Making

Require: A set of instances I
Require: A set of norms N
Require: An action $Action$ being considered
Ensure: Final decision on this action
1: $mostCompatibles := \emptyset$
2: $compatibility := \infty$
3: **for** $\langle Deontic', Condition', Action', Effect' \rangle \in I$ **do**
4: **if** $\exists \sigma : \sigma(Action') = Action$ **then**
5: **if** $|\sigma| < compatibility$ **then** //If less substitutions are required
6: $mostCompatibles := \{\langle Deontic', Condition', Action', Effect', \delta \rangle\}$
7: $compatibility = |\sigma|$
8: **if** $|\sigma| = compatibility$ **then**
9: $mostCompatibles := mostCompatibles \cup \{\langle Deontic', Condition', Action', Effect', \delta \rangle\}$
10: **if** $mostCompatibles <> \emptyset$ **then** //Best Case Scenario
11: $decision := null$
12: $importance := -\infty$
13: **for** $\langle Deontic', Condition', Action', Effect', \delta \rangle \in mostCompatibles$ **do**
14: **if** $\delta \geq importance$ **then**
15: $decision := Deontic'$
16: $importance = \delta$
17: **if** $decision = P$ **then**
18: $perform(Action)$
19: **else**
20: $abort(Action)$
21: **else** // Worst Case Scenario
22: $mostSimilars := \emptyset$
23: $similarity := -\infty$
24: **for** $\langle Deontic', Condition', Action', Effect', \delta \rangle \in N$ **do**
25: **if** $sim(Action, Action') > similarity$ **then**
26: $mostSimilars := \{\langle Deontic', Condition', Action', Effect', \delta \rangle\}$
27: $similarity = sim(Action, Action')$
28: **if** $sim(Action, Action') = similarity$ **then**
29: $mostSimilars := mostSimilars \cup \{\langle Deontic', Condition', Action', Effect', \delta \rangle\}$
30: $decision := null$
31: $importance := -\infty$
32: **for** $\langle Deontic', Condition', Action', Effect', \delta \rangle \in mostSimilars$ **do**
33: **if** $\delta \geq importance$ **then**
34: $decision := Deontic'$
35: $importance = \delta$
36: **if** $decision = P$ **then**
37: $perform(Action)$
38: **else**
39: $abort(Action)$

4 EVALUATION

In this section, we describe the procedure for evaluating the performance of our model.

4.1 Experiment Setup

4.1.1 DataSet. We considered a real dataset[7] of 1,739 users' rated information flows in SPA regarding their acceptability [2]. Different values of parameters, such as 15 data types, 15 recipient types, 5 transmission principles, and corresponding purposes, are used to form these information flows using contextual integrity via vignettes. Each user evaluated an average of 220 information flows in different contexts and identified an *'Acceptable (i.e., to permit)'* or *'Unacceptable (i.e., to forbid)'* decision for each case.

[7]The dataset collected by [2] can be accessed via the link https://osf.io/63wsm/.

4.1.2 Default Norm Set. The training set was used to extract default general norms to populate the set of N when no feedback was collected from users at the beginning. We follow the unsupervised machine learning approach applied in [2], which used the well-known Apriori algorithm to mine association rules. We use this approach for two main reasons: 1) this large-scale user study has demonstrated the feasibility of employing this technique, as the norms mined are considered to be practically meaningful and are able to represent users' perceptions of how their information is shared in the SPA ecosystem; 2) the *Confidence* value in this approach is theoretically used to measure the frequency of a particular rule across the dataset. Therefore, we used it as the value δ for each norm created.

4.1.3 L Function. Recall that in introducing our model, we proposed a relatively broad and general solution of the hybrid learning and reasoning function. In this experiment, we fleshed out the process of the model executing the learning mechanism after receiving a batch of feedback. Since the results of the user study in [2] show that data types can be classified into three groups based on their

sensitivity level[8], and that recipients can be classified into three different types: internal recipients, external recipients and third-party recipients[9], we specify that the model will automatically count the number of feedback received and compose the new norms for that group based on the available feedback. In particular, if a group contains n data types and the model has received feedback on $n-1$ data, the model will reason about the appropriate norm for the n_{th} data type using the feedback. Besides, when deciding on the modality to be used in a newly formed norm, we define it using the modality that appears most frequently in that group of norms. For example, if a user gives feedback that they do not want to share their information with their partner, parents and roommates, the model will automatically create a norm that prohibits sharing information with their children.

4.1.4 Similarity Function. Regarding the similarity checking mechanism, we simulate data types of the same sensitivity level to be similar. Moreover, recipients are also divided into different groups based on their *impact level*[10] on the information sharing flow, and we set that recipients in the same group are more similar. As for the rest of the parameters, we consider that their similarity depends on how many of them are the same. This method of calculating similarity can be summarized by the following equation.

$$Similarity(A, A') = \sum_{i=1}^{n} w_i \times sim(A_i, A'_i)$$

where $sim(A_i, A'_i) =$

$$\begin{cases} 1 - \frac{|impact(A_i) - impact(A'_i)|}{range\_impact}, & A_i, A'_i \in Recipient \\ 1 - \frac{|sen(A_i) - sen(A'_i)|}{range\_sen}, & A_i, A'_i \in Datatype \\ \frac{equal\_para}{sum\_para}, & otherA_i, A'_i \end{cases}$$

In this function, A represents the action detected, and A' represents the action described in a norm n. sim represents the similarity between two parameters, and the whole similarity between two actions depends on the similarity of the parameter pairs between them. To be more specific, *impact* represents the impact level if the parameter is recipient, while the *sen* express the sensitivity level of the parameter if it is a data type. For the remaining parameters we calculate the proportion of equal parameters to the total number of parameters. Moreover, as different pairs of parameters have varying degrees of influence on the decision, weights will be multiplied by them when calculating the total similarity. After computing the similarity between observed action with norms retrieved from norm base, the model will select the one with the highest $Similarity$ value and make the final decision.

4.1.5 Baselines and Evaluation Metrics. The performance of the model is compared to that of standard SPAs with no supervisory mechanism. We consider three cases: the first baseline model assumes that it will make random decisions for all cases, the second model assumes that all detected behaviours are prohibited, and the third model assumes that all detected behaviours are allowed. To evaluate and compare the performance of different models, we perform a 10-fold validation in a random 80/20 split. For NS model we use the training dataset for default norm creation (association rule mining). We use the *accuracy* as the metric.

[8]Sensitivity level:1: weather, playlist, sleeping hours, thermostat; 2: shopping, to do list, location; 3: call assistant, voice recording, video call, banking; 4: email, door locker, healthcare, smart camera. The higher the number, the higher the sensitivity.

[9]Internal recipients: partner, parents, children, roommate. External recipients: visitors, house keeper, friends. Third parties: law enforcement agencies, advertisement agencies, service providers.

[10]For instance, visitors and house keepers are in the same impact level to user perceptions of who can access their data.

4.2 Selecting Parameters

4.2.1 Min-support and confidence pairs. We studied the impact of using different minimal support and confidence value pairs for the association rule mining to create the initial norm set on model performance. The results for some values are not shown in this graph, such as the case for min-support=0.04 and min-confidence=0.85 because the model could not mine any rules at that moment, causing the issue that no values will be assigned to the norm set at the beginning. Fig. 3 illustrates how accuracy changes with different pairs. We selected min-support 0.01 and min-confidence 0.8 in the main experiment because it offers the highest accuracy.

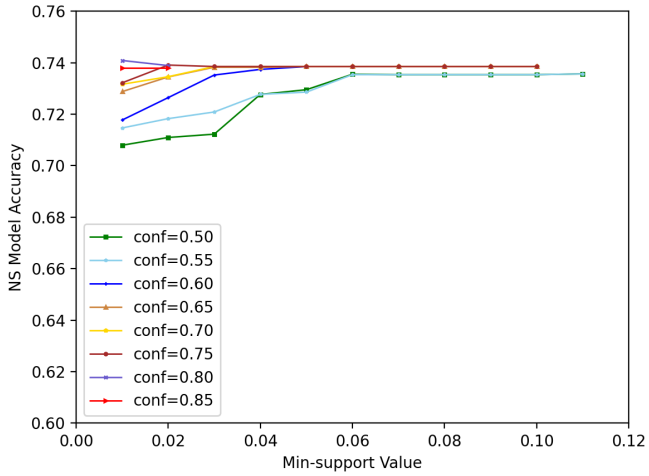

Figure 3: Accuracy with different minimal Support and Confidence value

4.2.2 Feedback rate. We tested how feedback rate affects the accuracy of the model. We started by adding 10% of the cases from test set as user feedback and repeated the experiment until 99% of the cases were taken. For example, as each user answered around 220 cases, having 10% (22 cases) as the feedback means 90% cases (198 cases) per user are judged by the model. The feedback was drawn at random from the testing set, so we ran ten sets of experiments for each different feedback percent value before averaging the accuracy rates. As expected, Figure 4 shows that as more cases are learned from users, the model can provide more accurate decisions. For the main experiment, we only provided details of the model's performance when approximately 25% of the feedback has been collected. This is because, firstly, as we can see in Figure 4, the accuracy rate increases sharply at the beginning, but then turns more stable when around 25% of user feedback is received. After that point, the accuracy rate increases, but not so significantly. Also, considering that, in practice, the less interactions the more convenient for users.

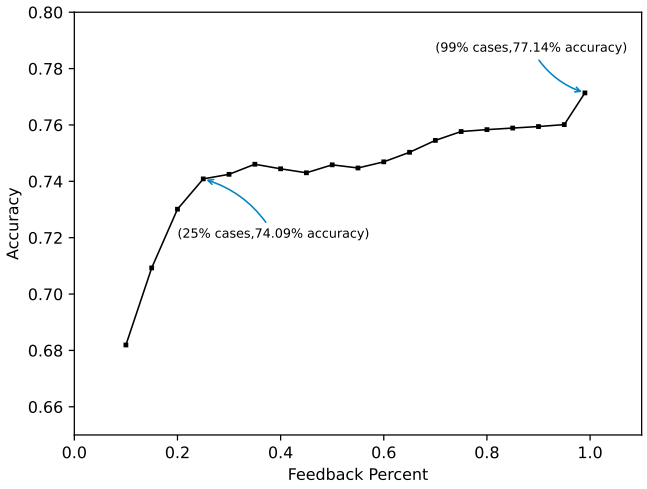

Figure 4: Accuracy changes with increasing feedback rate

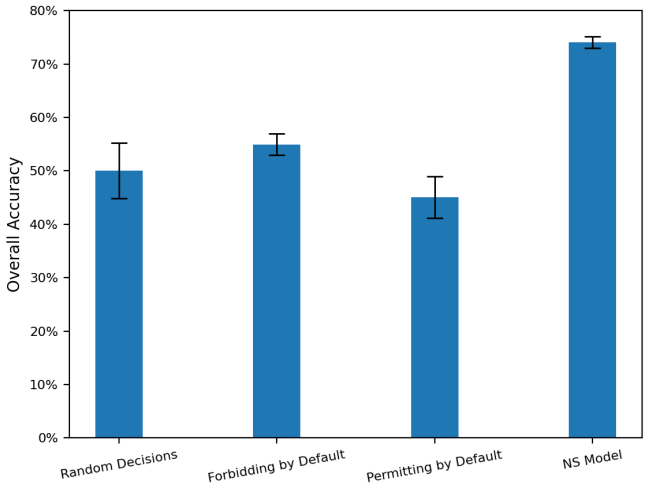

Figure 5: Model performance compared with baselines

Finally, interactions with users is an ongoing process, it may be difficult to predict when the model will be able to accumulate the user feedback.

4.3 Results

Figure 5 shows the accuracy of all the baseline methods together with the accuracy of the NS model with the parameters selected in the previous section. For each approach, we perform a 10-fold validation in a random 80/20 split. The overall accuracy of the NS model reaches is approximately 20% better than the baseline with the best results. We also performed a dependency t-test (paired sample t-test) to compare the differences between the NS model and each of the baseline models. First, for NS model with the baseline model which make random decisions, $t = -8.241$ and $p = 0.001 < 0.05$; Second, for NS model with the baseline model which prohibit all the detected actions, $t = -19.891$ and $p < 0.01$; Third, for NS model with the baseline model which permit all the detected actions, $t = -14.6471$ and $p < 0.01$. We can conclude that there was statistically significant improvements after deploying our method.

Table 1: Performance of NS model by norm origin and decision making mechanism

	Use	Accuracy
Feedback Norms	35.30%	77.33%
Default Norms	31.03%	73.69%
Similarity	33.67%	71.05%
Overall		74.09%

The detailed view of the NS model performance and the contribution made by its components are shown in Table 1. It can be seen that 35.30% cases were judged by instantiated norms that were created by using the hybrid learning and reasoning mechanism on the user's feedback, and the accuracy is 77.33%. Furthermore, 31.03% cases were also judged by using the relevant instantiated norms that were originally stored in the norm base N (default norms) at the beginning rather than learnt from the feedback set E. The remaining cases (33.67%) were cases in which no instantiated norms were relevant to the particular situation and were therefore judged by the similarity checking mechanism, and the accuracy obtained is 71.05%.

5 RELATED WORK

Privacy, data protection, and autonomous decision-making all fall within the scope of ethical issues raised by the development of AI, robotics, and autonomous systems [15, 16, 58], there is particular concern that when performing tasks, automated decisions made by the system may violate the privacy of the user (both single- and multi-user environments), enhancing the motivation to find ways to practically build machines that are ethically sound, and can also reason about actions as well as decisions that meet privacy needs [10].

Previous research [49] has identified autonomous systems as one of the potential solutions to privacy challenges, with the benefits far outweighing the privacy risks they may pose. These types of privacy-enhancing systems are designed to respect privacy in the first place, and have the ability to develop strategies for situations where data is shared by an individual [36, 37] or a group of users [38–40, 50, 51]. For instance, [22, 50, 51, 53] propose mechanisms to resolve the multi-party privacy management conflicts that arise in social media. More recently, [38–40] define and evaluate a value-aligned and explainable agent for managing multi-user privacy conflicts.

In terms of technologies that have been designed within the autonomous systems field, norm-based and normative systems have received considerable attention in recent years. These systems have been especially used as a method to regulate agents and autonomous systems to behave in ethically correct ways, preventing users' privacy from being violated as a result of the negative behaviours [11, 12, 49, 52]. In this case, privacy norms are not only

special forms of information that can be communicated and observed among agents, but they also represent an integration of knowledge by which agents can adjust their behavior to reasonably coordinate their actions with each other. Moreover, compared to frameworks and approaches from distributed systems that provide means of protection through a number of machine-readable access control policies, such as Ponder [14], SecPAL [6], and EPAL [32], agent-oriented privacy norms are able to represent and handle sophisticated relationships between resources. In turn, by capturing these relationships, privacy norms are able to prevent the triggering of a large number of policies about required resources that are redundant or that might overlap.

Extracting privacy norms, as we have done in our experiment, can also be done in other ways. In [56], the authors extracted the norms from multi-agent social simulations and showed how these norms can be used to make privacy decisions. In [57] they have extended the approach to show that access control can also be managed by norms. Agents in their approach can infer contextual information from image tags and compute privacy norms on their own, greatly reducing the burden of user participation in group decision-making mechanisms. A different, earlier, approaches were taken in [8], where the authors define a model based on inductive logic programming and social identity maps for learning privacy norms about group-sharing behaviours, and in [13], where acceptable information sharing norms were learned based on frequencies of topic communication.

The closest approach to ours, from an architectural perspective is the one described in [43], where the authors adopt a neuro-symbolic hybrid architectures for learning and reasoning about norms. More specifically, the authors first represent the rules with the help of I/O logic and transmit them to the neural network, which is responsible for learning the new norms. Similarly, our model follows this procedure of providing formal representation of norms before learning and processing. However, we use the deontic modality, as well as contextual parameters that can elaborately capture the context in which the SPA works, to express the privacy norms.

6 DISCUSSION AND CONCLUSION

SPAs lack the ability to self-monitor and manage the access rights held by different users in various scenarios, thus frequently make decisions that frustrate users and raise privacy concerns [9, 17, 33, 34]. Furthermore, users are unwilling to adopt access control mechanisms even if they work well, just because they require too much effort [59]. This motivated us to develop a model that can automatically reason about privacy norms to decide the best course of action according to the user's expectations. In addition, our model provides an explicit representation of privacy norms, considering information flows based on the contextual integrity theory of privacy, which makes the model easy to scrutinise to understand under which conditions and contexts the SPA will take a decision to share or not information. As future work, we would like to explore the best mechanisms to *explain* the privacy norms to users. One way to do this would be to enable our model to reason about the Theory-of-Mind of both individuals and groups, in different contexts [44], such that explanations can be tailored to the users' knowledge about privacy norms and communicated through dialogue. While our model is interpretable and allows scrutiny, the social process by which explanations would be made requires the design and validation of the explanations themselves and the best method to visualize and/or convey them [35, 40].

ACKNOWLEDGMENTS

We would like to thank the anonymous AIES reviewers for their helpful feedback. This research was partially supported by UKRI through REPHRAIN (EP/V011189/1), the UK's Research centre on Privacy, Harm Reduction and Adversarial Influence online, as part of its PRAISE inaugural project, and Xiao Zhan is funded by King's PGR International Scholarship.

REFERENCES

[1] Noura Abdi, Kopo Ramokapane, and Jose Such. 2019. More than smart speakers: security and privacy perceptions of smart home personal assistants. In Fifteenth USENIX Symposium on Usable Privacy and Security ({SOUPS} 2019).
[2] Noura Abdi, Xiao Zhan, Kopo M Ramokapane, and Jose Such. 2021. Privacy Norms for Smart Home Personal Assistants. In Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. 1–14.
[3] Noah Apthorpe, Yan Shvartzshnaider, Arunesh Mathur, Dillon Reisman, and Nick Feamster. 2018. Discovering smart home internet of things privacy norms using contextual integrity. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies 2, 2 (2018), 59.
[4] Noah Apthorpe, Sarah Varghese, and Nick Feamster. 2019. Evaluating the Contextual Integrity of Privacy Regulation: Parents' IoT Toy Privacy Norms Versus {COPPA}. In 28th {USENIX} Security Symposium ({USENIX} Security 19). 123–140.
[5] Tina Balke, Célia da Costa Pereira, Frank Dignum, Emiliano Lorini, Antonino Rotolo, Wamberto Vasconcelos, and Serena Villata. 2013. Norms in MAS: definitions and related concepts. In Dagstuhl Follow-Ups, Vol. 4. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.
[6] Moritz Y Becker, Cédric Fournet, and Andrew D Gordon. 2010. SecPAL: Design and semantics of a decentralized authorization language. Journal of Computer Security 18, 4 (2010), 619–665.
[7] Sebastian Benthall, Seda Gürses, Helen Nissenbaum, et al. 2017. Contextual integrity through the lens of computer science. Now Publishers.
[8] Gul Calikli, Mark Law, Arosha K Bandara, Alessandra Russo, Luke Dickens, Blaine A Price, Avelie Stuart, Mark Levine, and Bashar Nuseibeh. 2016. Privacy dynamics: Learning privacy norms for social software. In 2016 IEEE/ACM 11th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS). IEEE, 47–56.
[9] Yuxin Chen, Huiying Li, Shan-Yuan Teng, Steven Nagels, Zhijing Li, Pedro Lopes, Ben Y Zhao, and Haitao Zheng. 2020. Wearable Microphone Jamming. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. 1–12.
[10] Stefania Costantini. 2020. Ensuring trustworthy and ethical behaviour in intelligent logical agents. Journal of Logic and Computation (2020).
[11] Natalia Criado, Estefania Argente, and V Botti. 2011. Open issues for normative multi-agent systems. AI communications 24, 3 (2011), 233–264.
[12] Natalia Criado, Xavier Ferrer, and Jose Such. 2021. Attesting Digital Discrimination Using Norms. International Journal of Interactive Multimedia and Artificial Intelligence (IJIMAI) 6, 5 (2021), 16–23.
[13] Natalia Criado and Jose Such. 2015. Implicit contextual integrity in online social networks. Information Sciences 325 (2015), 48–69.
[14] Nicodemos Damianou, Naranker Dulay, Emil Lupu, and Morris Sloman. 2001. The ponder policy specification language. In International Workshop on Policies for Distributed Systems and Networks. Springer, 18–38.
[15] Virginia Dignum. 2019. Responsible artificial intelligence: how to develop and use AI in a responsible way. Springer Nature.
[16] Virginia Dignum, Matteo Baldoni, Cristina Baroglio, Maurizio Caon, Raja Chatila, Louise Dennis, Gonzalo Génova, Galit Haim, Malte S Kließ, Maite Lopez-Sanchez, et al. 2018. Ethics by design: Necessity or curse?. In Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society. 60–66.
[17] Daniel J Dubois, Roman Kolcun, Anna Maria Mandalari, Muhammad Talha Paracha, David Choffnes, and Hamed Haddadi. 2020. When Speakers Are All Ears: Characterizing Misactivations of IoT Smart Speakers. Proceedings on Privacy Enhancing Technologies 2020, 4 (2020), 255–276.

[18] Jide Edu, Xavier Ferrer-Aran, Jose Such, and Guillermo Suarez-Tangil. 2021. SkillVet: Automated Traceability Analysis of Amazon Alexa Skills. IEEE Transactions on Dependable and Secure Computing (TDSC) (2021).

[19] Jide Edu, Xavier Ferrer-Aran, Jose Such, and Guillermo Suarez-Tangil. 2022. Measuring Alexa Skill Privacy Practices across Three Years. In Proceedings of the Web Conference (WWW).

[20] Jide Edu, Jose Such, and Guillermo Suarez-Tangil. 2020. Smart home personal assistants: a security and privacy review. ACM Computing Surveys (CSUR) 53, 6 (2020), 1–36.

[21] Xavier Ferrer, Tom van Nuenen, Jose Such, Mark Cote, and Natalia Criado. 2021. Bias and Discrimination in AI: a cross-disciplinary perspective. IEEE Technology and Society 20, 2 (2021), 72–80.

[22] Ricard L. Fogués, Pradeep K. Murukannaiah, Jose Such, and Munindar P. Singh. 2017. SoSharP: Recommending Sharing Policies in Multiuser Privacy Scenarios. IEEE Internet Computing 21, 6 (2017), 28–36.

[23] Gian Maria Greco and Luciano Floridi. 2004. The tragedy of the digital commons. Ethics and Information Technology 6, 2 (2004), 73–81.

[24] Zhixiu Guo, Zijin Lin, Pan Li, and Kai Chen. 2020. SkillExplorer: Understanding the Behavior of Skills in Large Scale. In 29th USENIX Security Symposium (USENIX Security 20). 2649–2666.

[25] Yue Huang, Borke Obada-Obieh, and Konstantin Beznosov. 2020. Amazon vs. My Brother: How Users of Shared Smart Speakers Perceive and Cope with Privacy Risks. In Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. 1–13.

[26] Ilir Kola, Catholijn M Jonker, and M Birna van Riemsdijk. 2018. Modelling the Social Environment: Towards Socially Adaptive Electronic Partners.. In MRC@ IJCAI. 30–34.

[27] Josephine Lau, Benjamin Zimmerman, and Florian Schaub. 2018. Alexa, Are You Listening?: Privacy Perceptions, Concerns and Privacy-seeking Behaviors with Smart Speakers. Proceedings of the ACM on Human-Computer Interaction 2, CSCW (2018), 102.

[28] Emmanuel Letier and William Heaven. 2013. Requirements modelling by synthesis of deontic input-output automata. In 2013 35th International Conference on Software Engineering (ICSE). IEEE, 592–601.

[29] Alexander Maedche, Christine Legner, Alexander Benlian, Benedikt Berger, Henner Gimpel, Thomas Hess, Oliver Hinz, Stefan Morana, and Matthias Söllner. 2019. AI-based digital assistants. Business & Information Systems Engineering 61, 4 (2019), 535–544.

[30] Nathan Malkin, Joe Deatrick, Allen Tong, Primal Wijesekera, Serge Egelman, and David Wagner. 2019. Privacy Attitudes of Smart Speaker Users. Proceedings on Privacy Enhancing Technologies 2019, 4 (2019), 250–271.

[31] Kirsten Martin and Helen Nissenbaum. 2016. Measuring privacy: an empirical test using context to expose confounding variables. Colum. Sci. & Tech. L. Rev. 18 (2016), 176.

[32] Karsten Martiny, Daniel Elenius, and Grit Denker. 2018. Protecting privacy with a declarative policy framework. In 2018 IEEE 12th International Conference on Semantic Computing (ICSC). IEEE, 227–234.

[33] Donald McMillan, Barry Brown, Ikkaku Kawaguchi, Razan Jaber, Jordi Solsona Belenguer, and Hideaki Kuzuoka. 2019. Designing with Gaze: Tama–a Gaze Activated Smart-Speaker. Proceedings of the ACM on Human-Computer Interaction 3, CSCW (2019), 1–26.

[34] Abraham Mhaidli, Manikandan Kandadai Venkatesh, Yixin Zou, and Florian Schaub. 2020. Listen Only When Spoken To: Interpersonal Communication Cues as Smart Speaker Privacy Controls. Proceedings on Privacy Enhancing Technologies 2020, 2 (2020), 251–270.

[35] T. Miller. 2018. Explanation in artificial intelligence: Insights from the social sciences. Artificial Intelligence (2018).

[36] Gaurav Misra and Jose Such. 2017. PACMAN: Personal Agent for Access Control in Social Media. IEEE Internet Computing 21, 6 (2017), 18–26.

[37] Gaurav Misra and Jose Such. 2017. React: Recommending access control decisions to social media users. In Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2017. 421–426.

[38] Francesca Mosca, Ştefan Sarkadi, Jose Such, and Peter McBurney. 2020. Agent EXPRI: Licence to explain. In International workshop on explainable, transparent autonomous agents and multi-agent systems. Springer, 21–38.

[39] Francesca Mosca and Jose Such. 2021. ELVIRA: an Explainable Agent for Value and Utility-driven Multiuser Privacy. In International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS). 916–924.

[40] Francesca Mosca and Jose Such. 2022. An explainable assistant for multiuser privacy. Autonomous Agents and Multi-Agent Systems 36, 1 (2022), 1–45.

[41] Helen Nissenbaum. 2004. Privacy as contextual integrity. Wash. L. Rev. 79 (2004), 119.

[42] Helen Nissenbaum. 2009. Privacy in context: Technology, policy, and the integrity of social life. Stanford University Press.

[43] Alan Perotti, Guido Boella, Silvano Colombo Tosatto, Artur S d'Avila Garcez, Valerio Genovese, and Leon van der Torre. 2012. Learning and reasoning about norms using neural-symbolic systems. In International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2012, Valencia, Spain, June 4-8, 2012. 1023–1030.

[44] Ştefan Sarkadi, Alison R Panisson, Rafael H Bordini, Peter McBurney, and Simon Parsons. 2018. Towards an approach for modelling uncertain theory of mind in multi-agent systems. In International Conference on Agreement Technologies. Springer, 3–17.

[45] Ştefan Sarkadi, Alex Rutherford, Peter McBurney, Simon Parsons, and Iyad Rahwan. 2021. The evolution of deception. Royal Society open science 8, 9 (2021), 201032.

[46] Faysal Hossain Shezan, Hang Hu, Gang Wang, and Yuan Tian. 2020. Verhealth: Vetting medical voice applications through policy enforcement. Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies 4, 4 (2020), 1–21.

[47] Yan Shvartzshnaider, Schrasing Tong, Thomas Wies, Paula Kift, Helen Nissenbaum, Lakshminarayanan Subramanian, and Prateek Mittal. 2016. Learning privacy expectations by crowdsourcing contextual informational norms. In Fourth AAAI Conference on Human Computation and Crowdsourcing.

[48] Strategy Analytics. 2020. Global Smart Speaker Vendor & OS Shipment and Installed Base Market Share by Region: Q4 2019.

[49] Jose Such. 2017. Privacy and autonomous systems. In Proceedings of the 26th International Joint Conference on Artificial Intelligence. 4761–4767.

[50] Jose Such and Natalia Criado. 2016. Resolving multi-party privacy conflicts in social media. IEEE Transactions on Knowledge and Data Engineering 28, 7 (2016), 1851–1863.

[51] Jose Such and Natalia Criado. 2018. Multiparty privacy in social media. Commun. ACM 61, 8 (2018), 74–81.

[52] Jose Such, Agustin Espinosa, and Ana Garcia-Fornes. 2014. A survey of privacy in multi-agent systems. The Knowledge Engineering Review 29, 03 (2014), 314–344.

[53] Jose Such and Michael Rovatsos. 2016. Privacy policy negotiation in social media. ACM Transactions on Autonomous and Adaptive Systems (TAAS) 11, 1 (2016).

[54] TechRepublic. 2020. Alexa Skills: Cheat Sheet. https://www.techrepublic.com/article/alexa-skills-cheat-sheet/

[55] George Terzopoulos and Maya Satratzemi. 2019. Voice Assistants and Artificial Intelligence in Education. In Proceedings of the 9th Balkan Conference on Informatics. 1–6.

[56] Onuralp Ulusoy and Pınar Yolum. 2019. Emergent privacy norms for collaborative systems. In International Conference on Principles and Practice of Multi-Agent Systems. Springer, 514–522.

[57] Onuralp Ulusoy and Pinar Yolum. 2020. Norm-based access control. In Proceedings of the 25th ACM symposium on access control models and technologies. 35–46.

[58] Alan F Winfield, Katina Michael, Jeremy Pitt, and Vanessa Evers. 2019. Machine ethics: the design and governance of ethical AI and autonomous systems [scanning the issue]. Proc. IEEE 107, 3 (2019), 509–517.

[59] Eric Zeng and Franziska Roesner. 2019. Understanding and improving security and privacy in multi-user smart homes: a design exploration and in-home user study. In 28th {USENIX} Security Symposium ({USENIX} Security 19). 159–176.