Modelling and Reasoning for Indirect Sensing over Discrete-time via Markov Logic Networks

Athanasios Tsitsipas[*] , **Lutz Schubert**
Ulm University, Germany
{firstname, surname}@uni-ulm.de

Abstract

With the always increasing availability of sensor devices, there is constant unseen monitoring of our environment. A physical activity has an impact on more sensor modalities than we could imagine. It is so vivid that distinctive patterns in the data look almost interpretable. Such knowledge, which is innate to humans, ought to be encoded and reason upon declaratively. We demonstrate the power of Markov Logic Networks for encoding uncertain knowledge to discover interesting situations from the observed evidence. We formally relate distinguishable patterns from the sensor data with knowledge about the environment and generate a rule basis for verifying and explaining occurred phenomena. We demonstrate an implementation on a real dataset and present our results.

1 Introduction

With the always-changing physical environments, uncertainty and incompleteness are innate in them. Context-aware pervasive systems have been the centre of research regarding approaches to modelling uncertain contextual information and reasoning upon it [Bettini *et al.*, 2010]; moving from low-level contextual data (i. e., sensors) to higher-level contextual information, where it is most commonly referred to as "situation" [Dey, 2001; Gellersen *et al.*, 2002]. Setting up systems to observe an environment includes deploying probes (e. g., sensors) tailored to specific situations. Today, such efforts fell under the terms "internet of things" and "smart homes". Many situations are worth identifying using sensors in a single room, ranging from "is someone present" to "water boiling". Considering an entire home, we may end up with hundreds of such situations. An office building could have thousands, increasing dedicated sensors to cover all the above situations, driving higher economic and maintenance costs.

A compelling method in such deployments is to use indirect sensing, which is employed when the property in need (e. g., a situation) is not attainable to direct sense, either due to sensor malfunctions, connectivity issues or energy loss. In the literature, indirect sensing is interwoven with remote sensing or sensing from afar [Zhang *et al.*, 2019]. In our study, we translate indirect sensing to a cooperative model of sensor fusion [Durrant-Whyte, 1990], where surrounding heterogeneous sensors capture different aspects of the same phenomenon (i. e., activity[1]). Activity is often described by a specific temporal organisation of low-level sensor data, or as we call it, a "dimensional footprint"(DF). The low-level sensor data in a DF are the primary source of information used as evidence to understand and recognise the observed situation. Such techniques following a bottom-up approach to recognising situations are well-established in the area of context-aware pervasive computing [Schmidt, 2003]. Dealing with a concept as the DF requires handling both *uncertainty* and the *relational organisation*. Existing approaches for an indirect sensing task typically fail to capture such aspects at the same time.

For the mechanics of an indirect sensing task, recent research targets data analysis techniques employing machine learning to train complex models labelling the property they want to infer from the data. For example, in [Laput *et al.*, 2017] the authors train Support Vector Machine (SVM) models, in an automatic learning mode à la "programming by demonstration" [Dey *et al.*, 2004; Hartmann *et al.*, 2007], with raw sensor data while performing the activity of interest. The major limitation of such systems is that they use representations that are not relatable to humans. In addition, they do not support explicit encoding of knowledge about the environment. Background knowledge (e. g., contextual, domain or commonsense) may describe situations absent in training data or challenging to grasp and annotate. In addition, apart from the definition of knowledge, the occurred observables (i. e., events) in sensor data may be uncertain, as much as the manifestations of knowledge are (i. e., rules) in an analytical reasoning process.

We address these limitations by choosing a probabilistic logic-based approach using an amalgam of Event Calculus (EC) [Kowalski and Sergot, 1989] and Markov Logic Network (MLN) [Richardson and Domingos, 2006] to model uncertain knowledge about the relational manifestations of different and heterogeneous sensors reasoning to infer interesting situations. EC drives the modelling task by a set of meta-

---

[*] Contact Author

[1] A situation, in that case, is the *state* of activity.

rules that encode the interaction between the sensor events and their effects over discrete time. One of the exciting properties of EC is that a situation of interest persists over time unless it gets interrupted by the occurrence of other events. On the other hand, MLN combines first-order logic and concepts from probability theory to tackle uncertainty, which has received considerable attention in recent years with applications in video activity analysis [Cheng *et al.*, 2014], maritime surveillance [Snidaro *et al.*, 2015], music analysis [Papadopoulos and Tzanetakis, 2016] and others. Our goal is to design a reasoning mode for indirect sensing that handles uncertainty and uses interpretable representations from data. To this end, we make the following contributions: (1) We model existing sensor data into interpretable symbolic representations as elements in a narrative on a running scenario (cf. Section 2.2), (2) design a knowledge base (KB) within MLN for supporting indirect sensing while emulating commonsense reasoning, (3) evaluate the realisation of the approach using an open-source implementation of MLN, (4) demonstrate how the probability of an occurred situation changes over time while using different combinations of sensors.

Section 2 provides the terminology used in this document, including the running example and background information on Event Calculus and Markov Logic Networks. This leads to Section 3 where we introduce the concept of DF and how to model it. In Section ,4 we elaborate on MLN definitions, while in Section 5, we present the results and experiments. Section 6 provides a brief related work around the topic of event modelling and recognition. In Section 7, we summarise the main contributions and discuss details, including future work.

2 Preliminaries

2.1 Terminology

The Oxford English Dictionary gives a general definition for an *event* as "*a thing that happens or takes place, especially one of importance*". In our context, a "*thing*" is represented by a (sensor) data pattern. "*Importance*" matches the (subjective) interest in finding an explanation for this pattern. Many researchers try to use the term event in their way, depending on the context and the investigated environment, even though the definition of the word event remains the same.

We assume that an (interesting) event occurred on identifying a visible change in the sensor data. The identification involves a pre-processing step using some pattern extraction techniques [Patel *et al.*, 2002; Lin *et al.*, 2003; Yeh *et al.*, 2016]. Therefore, the timestamps for the respective pattern represents the event's *temporality*. This work clarifies a *time point* and a *series of time points* (exhibiting the concept of *duration*) bounded by a predefined window value. For example, the increase in the temperature readings is an interesting event and reflects the development of sensing data (temperature) over time. Therefore, a representation should semantically annotate an event's time point.

Interpreting symbols as representations of objects is a proxy to describe something instead of the actual thing. For example, if something is an ambient "high" temperature[2], that temperature does not reside in our heads when we think of it. The "it" of the temperature is a representation of the actual natural environmental property. This representation of something is an entity that transmits to us the idea of the real something. Perhaps we think of our discomfort or imaging ourselves reacting to this phenomenon (e. g., sweating) to represent the high ambient temperature. Alternatively, we use the colour red accompanied by the temperature degree.

An event *representation* in our work is a lexical word embedded in a "sentence" among other additional contextual words, which we understand. Therefore, the development of sensing data over time (i. e., a time series) is wrapped in a word that best describes its nature (e. g., data pattern). The event representation has two lexical parts. The one part is the *trend* of the pattern, and the other one is the *type of the pattern*. The trend of a pattern is represented by the words *upward* or *downward*. The patterns we may derive in the sensor readings could resemble a shape currently named *shapeoid*. For the sake of presentation, the lexical *shapeoids* are the following:

ANGLE A gradual, continuous line with an increasing (upward) or a decreasing (downward) trend in the sensor readings.

HOP A stage shift in the sensor readings, where the data have an apparent difference between two consecutive recognition time points (e. g., binary sensor values).

HORN This pattern is a transient increase or decrease in the sensor readings curve.

FLAT A horizontal line in the data, with either unchangeable values in the pattern duration or minimal changes.

We extract the *shapeoids* using the Symbolic Aggregate Approximation (SAX) technique. Many time series representation alternatives exist, but most of them result in a downsampled real-valued representation. In contrast, SAX boils down to a symbolic discretised form of the time series, which is abstract enough to extract the *shapeoids* generally. The paper's focus is not to describe how to obtain the proposed patterns from the sensor data but to put forward a concept of using temporal organisations of such representations to reason in a robust and declarative way.

2.2 Running Example

In Figure 1, we illustrate the activity of opening and closing a window and its impact (i. e., their DF) on five surrounding sensor types that happen to be in the same room. Later in the paper (cf. Section 3.3), we will showcase the extracted *shapeoids* from the raw data, which put forward a sufficient abstraction, serving as an input for a reasoning task.

The data are from a real-world public dataset [Birnbach *et al.*, 2019], where the authors collected sensor data while performing different activities. The data timeline spawns over two minutes, sufficient for demonstrating the essence of our approach.

---

[2] We use a threshold-based term to describe the comfort level for a human to endure.

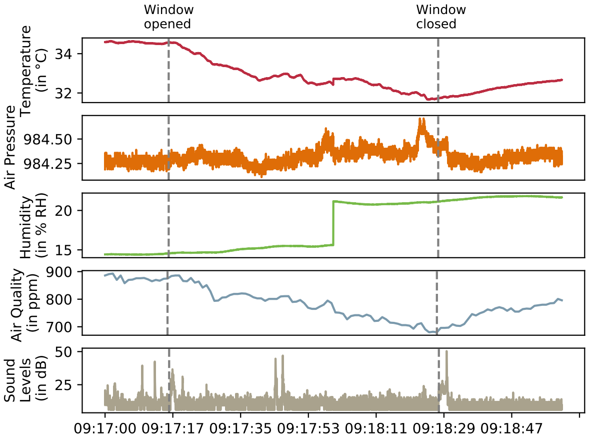


Figure 1: An example of how the activity of opening/closing a window affects the listed surrounding sensors.

2.3 Event Calculus

Representing and reasoning about actions and temporally-scoped relations has been a critical research topic in the area of Knowledge Representation and Reasoning (KRR) since the 60s [Shoham and McDermott, 1988]. Since then, various approaches have been proposed to overcome the Frame Problem in classical Artificial Intelligence (AI) [McCarthy and Hayes, 1981; Shanahan, 2006]; the challenge of representing the effects of actions. Among them, EC, which Kowalski and Sergot have initially proposed in 1986 [Kowalski and Sergot, 1989], is a system for reasoning about events (or actions) and their effects in the scope of Logic Programming. It comprises excellent expressiveness with intuitive and readable representations, making it feasible to extend reasoning. It is an adequate tool to fit domain knowledge representing how an entity progresses in time using events. It has found applications ranging from the scope of robotics [Russel *et al.*, 2013], game design [Nelson and Mateas, 2008] and commonsense reasoning [Shanahan, 2004; Mueller, 2014] to name a few.

From a technical point, the core ontology of the EC involves *events*, *fluents* and *time points*. The continuum of time is linear, and integers or real numbers represent the time points. A *fluent* can be whatever whose value is subject to change over time. At the occurrence of an *event*, it may change the value of a fluent. This could be a quantity, such as “the temperature in the room”, whose value varies in numbers, or a proposition, such as “the window is open”, whose truth state changes from time to time. In EC, the core axioms are domain-independent and define whether a fluent holds or not at a particular time point. In addition, these axioms can capture what is known as the common sense *law of inertia*; formal logic is a way of declaring that an event is assumed not to change a given property of a fluent *unless* there is evidence to the contrary [Shanahan and others, 1997].

We use a simplified version of EC (named MLN-EC), based on a discrete-time reworking of EC [Mueller, 2008], which was proven to work in a probabilistic setting [Skarlatidis *et al.*, 2015]. Other dialects may have additional restrictions (e. g., complex time quantification) that hinder the realisation of the approach. For more information, we point the reader to this paper [Mueller, 2004]. The basic predicates and the domain-independent axioms are presented in Table 1. One can read the upper line of two axioms from left to right: (1) a fluent f holds at time t if it was initiated at a previous time point, and (2) that the fluent f continues to hold, providing it was not previously terminated. The domain-dependent predicates $\mathsf{initiatedAt}/2$ and $\mathsf{terminatedAt}/2$ are expressed in an application-specific manner guiding the logic behind the occurrence of events and some contextual constraints. One example of a common rule for $\mathsf{initiatedAt}/2$ is:

Predicate	**Meaning**
$\mathsf{Happens}(e,\ t)$	Event e happens at time t
$\mathsf{HoldsAt}(f,\ t)$	Fluent f holds at time t
$\mathsf{InitiatedAt}(f,\ t)$	Fluent f is initiated at time t
$\mathsf{TerminatedAt}(f,\ t)$	Fluent f is terminated at time t
Axioms	
$\mathsf{HoldsAt}\,(f,\ t+1) \Leftarrow$ $\mathsf{InitiatedAt}\,(f,\ t)$	$\mathsf{HoldsAt}\,(f,\ t+1) \Leftarrow$ $\mathsf{HoldsAt}\,(f,\ t) \land$ $\neg\,\mathsf{TerminatedAt}\,(f,\ t)$
$\neg\,\mathsf{HoldsAt}\,(f,\ t+1) \Leftarrow$ $\mathsf{TerminatedAt}\,(f,\ t)$	$\neg\,\mathsf{HoldsAt}\,(f,\ t+1) \Leftarrow$ $\neg\,\mathsf{HoldsAt}\,(f,\ t) \land$ $\neg\,\mathsf{InitiatedAt}\,(f,\ t)$

Table 1: The core predicates and domain-independent axioms of the EC dialect, MLN-EC.

$$\begin{aligned}\mathsf{InitiatedAt}\,(f,\ t) \Leftarrow&\\ &\mathsf{Happens}\,(e,\ t)\ \land \\ &\mathit{Constraints}[t]\end{aligned} \tag{1}$$

The above definition states that a fluent f is initiated at time t if an event e happens, and some optional constraints depend on the domain. EC supports default reasoning via circumscription, representing that the fluent continues to persist unless other events happen. Therefore, in our definition of the event narrative, we assume these are the only events that occurred.

2.4 Markov Logic Networks

A Markov Logic Network (MLN) amalgam of a Markov Network (aka. Markov Random Field) and a first-order logic KB [Richardson and Domingos, 2006]. Specifically, it softens the constraints posed by the formulas with weights that support (positive weights) or penalise (negative weights) worlds in which they are satisfied. As opposed to classical logic, all the statements are hard constraints (i. e., preserving truthfulness).

The formulas, being first-order logic objects [Genesereth and Nilsson, 1987], are constructed using four symbols: *constants*, *variables*, *functions* and *predicates*. Predicates and constants start with an upper-case letter, whereas the functions and variables have lower-case letters. The variables are quantifiable over the given domain (e. g., type={Temperature, Humidity}). The *constants* are objects in the respective domain (e. g., sensor types: Temperature, Air Quality, Microphone etc.). *Variables* are ranges over the objects of the domain. The *functions* (e. g., downwardAngleTemp) represent actual mappings from a single object to a value or another object. Finally, the *predicate* symbols represent relations among objects associated with truth values(e. g., Happens(DownwardAngle_Temp,4)).

A KB in MLN consists of both hard- and soft-constrained formulas. Hard constraints (clauses with infinite weight) are interwoven with unequivocal knowledge. Therefore, an acceptable world fulfils all of the hard constraints. By contrast, the soft constraints are related to the imperfect knowledge of the domain, which can be falsified in the world's existence in discourse. This means that when a world violates a formula, it is less probable but not impossible.

Formally, a MLN is a set of pairs (F_i, w_i), where F_i is a first-order logic formula and w_i is a real numbered weight. The KB L, with the weighted formulas together with a finite set of constants $C = \{c_1, c_2, \ldots, c_{|C|}\}$, defines a ground Markov Network $M_{L,C}$ as follows [Richardson and Domingos, 2006]:

- $M_{L,C}$ has one binary node for each possible grounding of each predicate in L. The value of the node is 1 if the grounded atom is true and 0 otherwise.
- $M_{L,C}$ contains one feature for each possible grounding of each formula F_i in L. The value of this feature is 1 if the formula is true and 0 otherwise. The weight of the feature is the w_i associated with F_i in L.

An MLN is a template for constructing Markov networks: it will produce different networks given different constants. The *grounding* process is the replacement of variables with a constant in their domain. The nodes of $M_{L,C}$ correspond to all ground atoms that can be generated by grounding a formula F_i in L, with constants of C. Thus there is an edge between two nodes of $M_{L,C}$ iff the corresponding ground predicates are conditionally dependant on a grounding of a formula F_i in L. A possible world from the MLN must satisfy all of the hard-constrained formulas and be proportional to the exponential sum of the weights of the soft-constrained formulas satisfied in this world (cf. Equation 2). Hence, a MLN defines a log-linear probability distribution over Herbrand interpretations(i. e., possible worlds).

In an indirect sensing task context, we know *a priori* that we will have two kinds of predicates; the evidence variable X, containing the narrative of real-time input events, translated with the Happens predicates of EC, and the set of query HoldsAt predicates Y, as well as other groundings of "hidden" predicates (i. e., neither query nor evidence); in EC these are the InitiatedAt and TerminatedAt predicates. Finally, the conditional likelihood of Y given X is defined as follows [Singla and Domingos, 2005]:

$$P(y \mid x) = \frac{1}{Z_x} \exp\left(\sum_{i \in F_Y} w_i n_i(x, y)\right) \tag{2}$$

$x \in X$ and $y \in Y$ represent the possible assignment of the evidence set X and the query set Y, respectively. F_Y is the set of all MLN clauses produced from the KB L and the finite set of constants C. The $n_i(x, y)$ is the number of true groundings of the i-th clause involving the query atoms y given the evidence atoms x. Finally, Z_x is a partition function that normalises for all the possible assignments of x.

Equation 2 shows the probability distribution of the set of query variables conditioned over the set of observations. By modelling the conditional probability directly, the model remains agnostic about potential dependencies between the variables in X, and any factors that depend on X are eliminated. Instead, the model makes conditional independence assumptions among the Y and assumptions on its inherent structure with dependencies of Y on X. Therefore, in such a way, the number of the possible words is constrained [Singla and Domingos, 2005; Sutton and McCallum, 2006] and the inference is much more efficient. However, calculating exactly the formula might become intractable even for a small domain. Consequently, other approximate inference methods are preferred.

Originally, the authors in [Richardson and Domingos, 2006] propose to use Gibbs sampling to perform inference, but they found out that the sampling breaks down when the KB has deterministic dependencies[3] [Poon and Domingos, 2006; Domingos and Lowd, 2009]. The authors proposed another Markov Chain Monte Carlo method called MC-SAT [Poon and Domingos, 2006] based on satisfiability with slice-sampling. Another type of inference is the *Maximum A Posteriori* (MAP) which described the problem of finding the most probable state of the world given some evidence, which reduces to find the truth assignment that maximises the sum of weights of satisfied clauses (i. e., $\underset{y}{\text{argmax}} p(y \mid x)$).

The problem is generally NP-hard, but both exact and approximate satisfiability solvers exist [Domingos and Lowd, 2009]. In our experiments, we run approximate inference using the MC-SAT algorithm.

3 Modelling a DF

An activity affects various fundamental environmental properties, such as speed, pressure, temperature, luminosity, etc. Surrounding sensors may capture the various changes (forming the activity's DF), which depends on different contextual information, such as their proximity from the occurred phenomenon and their type (cf. Section 3.1). In addition, a sensor may observe ambient values (e. g., temperature) or require manual intervention to observe a change (e. g., separating the two magnetic elements of a contact sensor) (cf. Section 3.2).

This *"change"* (i. e., the forming pattern) is the "interesting event" we want to focus on. This observed change mostly stays unobserved. Thus, the emitted DF indicates its occurrence. In addition, its state is a continuous value in time, which is tracked under the definition of the *"fluent"*. With no sensor modality to identify the occurrence of an activity, due to its unavailability at the given time, or by simply stating that there does not exist any direct one, we account its DF as a *space* with equivalent options that "indirectly" account for the same activity.

Our work uses commonsense knowledge (CK) to characterise how activity affects its environment. From the running

---

[3] They are formed from hard-constrained formulas in the KB.

example in Section 2.2, some distinct data patterns exist, almost as recognisable to the human eye where one may exercise a hypothesis against the data. We consider that a data-processing step is viable to extract such patterns, but it is out of the scope of the current paper. The abstracted representations (cf. Section 2.1) from low-level sensor data reflect their organisations in shapes and trends (e. g., an increasing angle in the sensor data). Therefore, one with a naive knowledge of physics can make hypotheses about the occurrence of an activity using the abstractions from sensor data as evidence (cf. Section 3.3).

3.1 Contextual Constraints

Sensors are interfaces that serve as occurrence indicators for various monitored situations. The sensor numbers could increase accordingly as their numbers increase, making the instrumentation, deployment and maintenance cumbersome tasks. A sensor primarily measures an environmental change as accurate as possible, varying between the different manufacturers. Selecting a sensor to monitor a situation ought to obey some criteria, which formulate the sensing fidelity of its output. In this paper, we propose the following criteria:

Type There exist different vendors for various sensors. Nonetheless, the type of sensor is of key importance. There is no doubt that different manufacturers may offer a better sensor device, affecting accuracy. Semantically, the sensor type determines if the sensor participates in the verification process, not its model.

Location The location is another important aspect of determining the credibility of the sensor output. Either the physical location or the position of the sensor in the space should affect the decision of selecting any sensor of a given type in a location (e. g., a room).

As discussed later in the paper, the above criteria are minimal constraints for a sensor to participate in reasoning. However, the sensors have a fundamental high-level classification, making the *shapeoid* extraction from their data clearer and focused.

3.2 Sensor Classification

A sensor is an interface between the physical and the digital world. The raw sensor data rarely matches human semantics, but the representations of patterns in them are. The kind of sensor classification the paper foresees, bases on the nature of the resulting sensor data, is as follows:

Binary sensors restrict their result to two possible values. Usually, the values resemble the category itself (i. e., being binary); thus, *one* and *zero*. Furthermore, depending on the context[4] the result may take values from it. For example, the output of a physical switch is "on" or "off", the result of a motion sensor may be "present" or "not present", and so on. The suitable data patterns for the binary sensors are the HOP and FLAT representations.

Numerical sensors are almost every sensor with an arithmetic output in the set of real numbers $\mathbb{R}$. Some examples of quantifiable sensors are an accelerometer, humidity, temperature, pressure sensor etc.. Accordingly, the data patterns, which we found in the raw sensor data, are those of ANGLE, HORN and FLAT.

One could say that a binary sensor is a subset of numerical sensors. However, we make the distinction explicit, as the binary sensors are semantically a practical standalone class. In the running example, we use numerical sensors. The sensor data's available observations (i.e., shapeoids) are the simple events with their respective time point in the focused bounded time window. We represent them with the Happens predicate, where finally a collection of such predicates form the so-called "narrative" in EC.

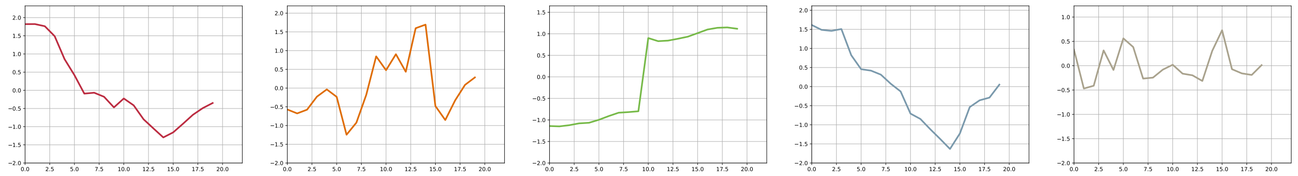
3.3 The Narrative of Events in EC

An event "just" *happens*, with an accompanied discrete time point to keep a reference in the timeline. The chosen representation of it, according to the dialect of EC, is the predicate $\mathsf{Happens}(e, t)$. Time t can be quantified over the spectrum of integers, exhibiting coherence among the occurred events. The events in the sensing timeline are the formed *shapeoids*, and by using lexical words for the symbolic representation, the intuition behind them is human-readable (e. g., *downward* ANGLE). For example, in Figure 1, the two activities of opening and closing the window produce an impact in the five surrounding sensors. We observe that around the time of opening the window, distinct patterns are forming. Figure 2 contains in separate graphs a more clear view of the data in Figure 1, after performing a dimensionality reduction step (e. g., Piecewise Aggregate Approximation (PAA) [Ding *et al.*, 2008]). The patterns were extracted empirically, resembling the proposed lexical *shapeoids* (cf. Section 2.1):

$$\begin{array}{l}
\dots \\
\mathsf{Happens(Flat\_Mic,3)} \\
\mathsf{Happens(Flat\_Hum,3)} \\
\mathsf{Happens(DownwardAngle\_Temp,4)} \\
\mathsf{Happens(DownwardAngle\_Aq,4)} \\
\mathsf{Happens(UpwardHorn\_Mic,4)} \\
\dots \\
\mathsf{Happens(UpwardHorn\_Temp,11)} \\
\mathsf{Happens(UpwardAngle\_Hum,11)} \\
\mathsf{Happens(Flat\_Mic,11)} \\
\dots \\
\mathsf{Happens(DownwardAngle\_Temp,14)} \\
\mathsf{Happens(DownwardAngle\_Pres,14)} \\
\mathsf{Happens(Flat\_Hum,15)} \\
\mathsf{Happens(UpwardHorn\_Mic,15)} \\
\mathsf{Happens(UpwardAngle\_Temp),15)} \\
\dots
\end{array} \tag{3}$$

---

[4] *Context is any information that one can use to characterise the situation of an entity. An entity is a person, place, or object considered relevant to the interaction between a user and an application, including the user and application themselves* [Dey, 2000]



(a) Temperature (b) Air Pressure (c) Humidity (d) Air Quality (e) Microphone

Figure 2: The z-normalised sensor data (in 20 data points) from Figure 1, after a dimensionality reduction step.

4 Probabilistic Indirect Sensing via MLN definitions

.

In the following, we elaborate on constructing the KB containing the representations of the sensor events, using contextual words in "sentences" that comply with the formalism of EC and are expressed in first-order logic.

4.1 Knowledge Base

For our purposes, the KB, or the so-called "theory", contains a few function definitions, predicate definitions, as well as the inertia laws axioms of EC[5] as seen in (2). We consider the observed patterns as a continuous narrative of Happens predicates (cf. (3)). InitiatedAt and TerminatedAt determine under which factors a fluent is initiated or terminated at a given time point, using the form in (1). Finally, the query predicate HoldsAt incorporate a possible quantification over the verification of a monitored situation (i. e., a fluent).

Table 2 shows a fragment of the KB and the associated weights. The formulas are converted to a clausal form during the grounding phase, also known as conjunctive normal form (CNF), a disjunction of literals. The next step is the replacement of the variables with the constants, which formulate grounded predicates. As such, the construction of the Markov Network consists of one binary node V for each possible grounding of each predicate. A world is an assignment of a truth value to each of these nodes.

The definition of the indirect sensing rules follow CK represented as a theory in MLN enacting it as part of commonsense reasoning (CR)[6]; the sort of reasoning people perform in daily life [Mueller, 2014], which is vague and uncertain. For example, the Table 2 contains two separate rules, which reflect an atomic instruction of the DF, using a temperature sensor and a microphone. For our purposes, we consider that the events in the narrative are the only one occurred.

A rise in the temperature readings, or a sudden spike in the sound pressure levels, could be anything in an open world, including the opening/closing of a door in a room. However, with the help of context, we may exercise the hypothesis that a temperature sensor *close* to the window could indicate its state. The hypothesis is asked in the form of a query, representing the probability for the situation of *an opened window* to be true for the given observations (i. e., ground truth). For example, if we require to encode an "opened door", we may include the same rule with a lower weight encoding our confidence for the result. Then, using the background knowledge that the sensors are closer to the window, we encode this with a higher weight value to the *opened window* rule. MLN has many learning algorithms [Richardson and Domingos, 2006] to determine the weight assignment; however, as we do not intend to select the absolute probabilities of a specific occurred situation, we opt for the most likely situation given the evidence.

4.2 Evidence

The evidence contains ground predicates (facts) (e. g., the narrative of events in Section 3.3) and optionally ground *function mappings*. A *function mapping* is a process of mapping a function to a unique identifier. For example, the first formula in Table 2 contains the function downwardAngleTemp(r). During the grounding phase, constants from the domain of the variable r substitute it[7]. Thus, a function mapping could be the following: DownwardAngle_Temp_LocA = downwardAngleTemp(LocationA). All the events of the grounded Happens predicates in Section 3.3 follow the same procedure for their function mappings.

5 Experiments and Results

In this section, we evaluate our approach in the domain of smart homes. As presented in Section 2.2, we use a publicly available dataset. The data timeline spawns over twelve consecutive full days. The dataset was in a zip format, which contains multiple comma-separated value (CSV) files with a total size of approximately 50 Gigabytes (GB)[8]. We selected one device close to the interest situation (i.e., close to the window). We extracted the relevant data points using the five sensors capturing the DF of opening/closing the room's window. We do not process the raw data points, but instead, we use the *shapeoids* from the data; their extraction was possible via our tool *Scotty*[9]. The total number of shapeoid events are 4393, where the ground truth events from the window contact sensor are 87.

---

[5] They should remain hard-constrained; otherwise, the recognition of the situation will converge to be uncertain up to the horizon of probability.

[6] CR is implemented as a valid (or approximately valid) inference [Davis, 2017] in MLN as part of the EC law of inertia.

[7] We assume a single room and its context is not reflected in the naming scheme of the function.

[8] The actual size of the raw data exceeds the 250 GB.

[9] This work is meant to be published in a forthcoming conference.

FOL formula	Weight
$\mathsf{InitiatedAt}\,(\mathsf{openedWindow}\,(r)\,,\;t) \Rightarrow \mathsf{Happens}\,(\mathsf{downwardAngleTemp}\,(r)\,,\;t) \wedge \mathsf{Happens}\,(\mathsf{flatTemp}\,(r)\,,\;t-1)$	2.1
$\mathsf{InitiatedAt}\,(\mathsf{openedWindow}\,(r)\,,\;t) \Rightarrow \mathsf{Happens}\,(\mathsf{upwardHornMic}\,(r)\,,\;t) \wedge \mathsf{Happens}\,(\mathsf{flatMic}\,(r)\,,\;t-1)$	0.2

Table 2: An excerpt of the first-order KB and the corresponding weights in the MLN.

Scenario	Description	Duration
S#1	Two sensors with weak and strong weights.	1 m 45 s
S#2	Three sensors with one weak and two strong weights.	1 m 9 s

Table 3: The described scenarios with their inference duration times.

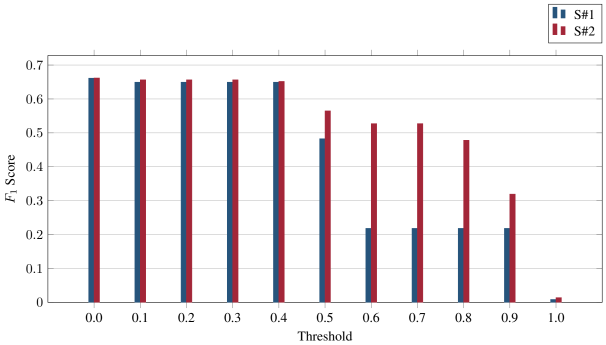


Figure 3: F_1 scores using various threshold values for the situation recognition of the opened window.

We put forward two scenarios (cf. Table 3), which contain rules for declaring the alternatives in recognising the situation of an opened/closed window. The purpose of the scenarios is to run the computation against the existing narrative with the discovered events but using different sensor compositions. Each recognition rule also contains a weight value, which was empirically assigned, as we consider them confidence values of the rule.

We implemented the KB and the narrative evidence file to demonstrate the approach's feasibility using an open-source implementation of Markov Logic Networks, named LoMRF [Skarlatidis and Michelioudakis, 2014]. Together with the domain-dependent rules for each scenario, the full KB and the evidence file are publicly available online[10], enabling the reproducible results. The KB, given the evidence, is transformed into a Markov Network of 26353 ground clauses and 13177 ground predicates. We run marginal inference from the developed MLN on vanilla runs without any interference from other processes. All the results are averaged over five runs with a corresponding standard deviation. The experiments are executed on a virtual machine(VM) running in a self-hosted data centre at the University of Ulm running on OpenStack under the series "Victoria". The VM runs with 8 cores (16 threads) and 16 GB of RAM.

Scenario	TP	TN	FP	FN	Precision	Recall	F_1
S#1	288	2039	174	1892	0.6234	0.1321	0.2180
S#2	1016	1554	659	1164	0.6066	0.4661	0.5271

Table 4: Performance results using the marginal inference and a threshold of 0.6.

In the experimental analysis, we present the results for the marginal inference in terms of F_1 score for a range of thresholds between 0.0 and 1.0. We consider the situation recognition task successful with a probability above the specified threshold. In Table 4, we present a snapshot of the performance using the threshold value 0.6 in terms of True Positives (TP), True Negatives (TN), False Positives (FP), False Negatives (FN), Precision, Recall and F1 score.

The scenarios have a certain flavour. The basic intuition from the experiments is to showcase that we may use sensors that have an obscure interpretation (e. g., a spike in the microphone can be anything, even being next to the window) and sensors that act as a more direct verification step (e. g., air quality, temperature). We assume that the *shapeoid* events are the only ones that happen in the environment in focus. More alternative sentences may be encoded accordingly, using *shapeoids* of the humidity or the air pressure sensor. Based on the inertia laws of EC, the fluent start to hold at the time point *t+1*, and therefore the assignment to the next time point from the used pattern event in the narrative (3). In Figure ,3 the F_1 score is higher for the marginal inference in S#2 due to the additional strong sensor. The S#1, similar to S#2, contains a shapeoid in the microphone data (increasing horn), which matches both the fluent's initiation and termination rules. Hence, during the inference process, the probability always strives towards 0.5, which is regulated by another sensor in the rules (air quality sensor) with a higher weight value.

We note here that in a real setting, the verification of situation (i. e., the fluent) depends on whether the required observation is made (e. g., the shapeoid event from the temperature sensor), which may be a delayed effect of the activity itself - in other words: it takes some time until the open window affects the temperature sufficiently. In the experimental analysis, we calculate the performance measures strictly based on the time range of an opened window. Therefore the ground truth is the single point of reference for calculating the performance. The delay between the activity and its observable DF should be accounted for a more accurate timing prediction. We observe a considerable amount of FP, which indicates a plausible calculation of an opened window but with a certain recognition delay. Thus, we consider the F_1 scores in the scenarios to be slightly higher.

---

[10] https://osf.io/n3ury/

6 Related Work

Research in context modelling, context reasoning, and their unified view via various middleware systems is tremendous; for a recent survey, we point the reader to [Perera *et al.*, 2013]. In the paper, we focus more on a bottom-up approach to the recognition of occurred situations. We employ a probabilistic rule-based approach, using occurred sensor events as evidence for the reasoning task.

In [Liu *et al.*, 2017], the authors create a bottom-up hierarchical model using the raw sensor data as evidence while creating inference rules encoded in an MLN to recognise complex events. In order to create abstractions from the raw data, they use various thresholds per sensor type. In our approach, we use generic template abstractions which base on the data shapes and trends. The core contribution of their paper is the dynamic assignment of weights learned from a training dataset; we do not assume that the user has a training dataset to learn the weights from because we use them as confidence values for the inference rules. Finally, in our paper, we foresee scalability issues that may arise from the free variables in the MLN rules, which may drive the computation times to higher levels.

Considering our choice for a rule-based reasoning technique has a broad spectrum of applications to many domains, making it a commonly used technique [Perera *et al.*, 2013]. Another interesting technique, which bases on previously acquired knowledge, is case-based reasoning (CBR) [Aamodt and Plaza, 1994; Biswas *et al.*, 2014]. It offers solving mechanisms by adopting solutions that have been suggested to similar issues in the past. The authors in [Kofod-Petersen and Aamodt, 2003] use CBR to understand an occurred situation based on available contextual information. A case-based solution is not favourable in our case because collecting and maintaining previous cases is a cumbersome task. Our work does not require any previous known input from sensor observations and domain-dependent knowledge during the rule specification.

In the paper, we focus on finding alternatives for recognising a situation. Similarly, Loke [Loke, 2006] advocates that the situation *in_meeting_now* has different recognition ways based on contextual cues. The author follows an abductive treatment of the subject as we also do. In the forthcoming years, the author developed a formalism to represent compositions of sensors that can act on an understanding of their situations [Loke, 2016].

Finally, although sensing data contain implicit information, explicit domain knowledge is required for situation recognition. Many research works employ logic-based models for situation recognition in smart homes, such as the Event Calculus (EC) [Chen *et al.*, 2008]. Other works have also employed EC in activity recognition from video streams [Artikis *et al.*, 2014] and health monitoring [Falcionelli *et al.*, 2019]. However, it is unclear how they move from the raw data to the tagged symbolic representations in these systems.

7 Conclusion & Discussion

In the paper, we employed Markov Logic Networks for the modelling and reasoning over uncertain alternatives for the method of indirect sensing. We use the temporal formalism of EC as a "linchpin" for driving the reasoning about the sensing objects and creating observations for the occurrence of certain situations (e. g., "is the window open"). The concept of the DF allows using different sensor setups to monitor the same situation(s). In other words, it is parallel to interpreting the given evidence (e. g., sensor data) for finding the most likely explanation, which created the DF. As such, we declare these logical "inference" sentences in a human-readable form of reasoning that incorporates commonsense logic.

Due to the nature of environmental situations, the interpretation (i. e., evaluation) of such sentences depends on the full context. For example, the same sentence in Table 2 might not apply if the weather outside is warmer than the sensor's environment. In this case, the temperature may not decrease but stay the same or even increase. Therefore, one will never evaluate the according to sentence to true. Instead, a fallback to another sensor is needed. Nevertheless, the approach defends the redundancy, or alternatives, in detecting the desired situation, considering that we usually use direct means for sensing (e. g., use a contact sensor to detect if the door is open).

The lack of sensors to capture the whole DF of activity leads to an incomplete "view of the world". The question thereby is, which physical effects are of specific relevance for interpreting an event and omitted. These conditions may vary enormously between different events, e. g., a person speaking or the sun rising both have other effects on the environment and thus (to a degree) require various sensors for interpretation, but also both could be observed using additional information: sound, visual, temperature, time etc.

Concerning the employed method of MLNs, there is an issue using predicates with free variables in the body of a rule; during the grounding phase, it creates a disjunction of the cartesian grounded conjunction of the formulas, translating these variables to existentially quantified leading to a possible combinatorial explosion. We consider any additional constraint in a domain-dependent rule should contain as variables only the time t and the location r. Any knowledge engineer should follow this and remove any existentially quantified variables, using the technique of skolemisation [Broeck *et al.*, 2013], overcoming this limitation for the solution's scalability.

Finally, the observed data patterns may also result from multiple overlapping activities challenging to separate, such as speaking in traffic, leading to uncertainty about the interpretation. As future work, we want to overcome the limitations of MLN concerning the free variables in the rules and concentrate on a dynamic ecosystem that realises the proposed work.

Acknowledgments

This work was partially funded by the Federal Ministry of Education and Research (BMBF) of Germany under Grant No. 01IS18072.

References

[Aamodt and Plaza, 1994] Agnar Aamodt and Enric Plaza. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI communications*, 7(1):39–59, 1994.

[Artikis *et al.*, 2014] Alexander Artikis, Marek Sergot, and Georgios Paliouras. An event calculus for event recognition. *IEEE Transactions on Knowledge and Data Engineering*, 27(4):895–908, 2014.

[Bettini *et al.*, 2010] Claudio Bettini, Oliver Brdiczka, Karen Henricksen, Jadwiga Indulska, Daniela Nicklas, Anand Ranganathan, and Daniele Riboni. A survey of context modelling and reasoning techniques. *Pervasive and mobile computing*, 6(2):161–180, 2010.

[Birnbach *et al.*, 2019] Simon Birnbach, Simon Eberz, and Ivan Martinovic. Peeves: Physical event verification in smart homes. In *Proceedings of the 2019 ACM Conference on Computer and Communications Security*. ACM, 2019.

[Biswas *et al.*, 2014] Saroj K Biswas, Nidul Sinha, and Biswajit Purkayastha. A review on fundamentals of case-based reasoning and its recent application in different domains. *International Journal of Advanced Intelligence Paradigms*, 6(3):235–254, 2014.

[Broeck *et al.*, 2013] Guy Van den Broeck, Wannes Meert, and Adnan Darwiche. Skolemization for weighted first-order model counting. *arXiv preprint arXiv:1312.5378*, 2013.

[Chen *et al.*, 2008] Liming Chen, Chris Nugent, Maurice Mulvenna, Dewar Finlay, Xin Hong, and Michael Poland. Using event calculus for behaviour reasoning and assistance in a smart home. In *International Conference on Smart Homes and Health Telematics*, pages 81–89. Springer, 2008.

[Cheng *et al.*, 2014] Guangchun Cheng, Yiwen Wan, Bill P Buckles, and Yan Huang. An introduction to markov logic networks and application in video activity analysis. In *Fifth International Conference on Computing, Communications and Networking Technologies (ICCCNT)*, pages 1–7. IEEE, 2014.

[Davis, 2017] Ernest Davis. Logical formalizations of commonsense reasoning: a survey. *Journal of Artificial Intelligence Research*, 59:651–723, 2017.

[Dey *et al.*, 2004] Anind K Dey, Raffay Hamid, Chris Beckmann, Ian Li, and Daniel Hsu. a cappella: programming by demonstration of context-aware applications. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 33–40, 2004.

[Dey, 2000] Anind Kumar Dey. *Providing Architectural Support for Building Context-Aware Applications*. PhD thesis, Georgia Institute of Technology, USA, 2000. AAI9994400.

[Dey, 2001] Anind K Dey. Understanding and using context. *Personal and ubiquitous computing*, 5(1):4–7, 2001.

[Ding *et al.*, 2008] Hui Ding, Goce Trajcevski, Peter Scheuermann, Xiaoyue Wang, and Eamonn Keogh. Querying and mining of time series data: experimental comparison of representations and distance measures. *Proceedings of the VLDB Endowment*, 1(2):1542–1552, 2008.

[Domingos and Lowd, 2009] Pedro Domingos and Daniel Lowd. Markov logic: An interface layer for artificial intelligence. *Synthesis lectures on artificial intelligence and machine learning*, 3(1):1–155, 2009.

[Durrant-Whyte, 1990] Hugh F Durrant-Whyte. Sensor models and multisensor integration. In *Autonomous robot vehicles*, pages 73–89. Springer, 1990.

[Falcionelli *et al.*, 2019] Nicola Falcionelli, Paolo Sernani, Albert Brugués, Dagmawi Neway Mekuria, Davide Calvaresi, Michael Schumacher, Aldo Franco Dragoni, and Stefano Bromuri. Indexing the event calculus: towards practical human-readable personal health systems. *Artificial intelligence in medicine*, 96:154–166, 2019.

[Gellersen *et al.*, 2002] Hans W Gellersen, Albrecht Schmidt, and Michael Beigl. Multi-sensor context-awareness in mobile devices and smart artifacts. *Mobile Networks and Applications*, 7(5):341–351, 2002.

[Genesereth and Nilsson, 1987] Michael R Genesereth and Nils J Nilsson. *Logical foundations of artificial intelligence*. Morgan Kaufmann, 1987.

[Hartmann *et al.*, 2007] Björn Hartmann, Leith Abdulla, Manas Mittal, and Scott R Klemmer. Authoring sensor-based interactions by demonstration with direct manipulation and pattern recognition. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 145–154, 2007.

[Kofod-Petersen and Aamodt, 2003] Anders Kofod-Petersen and Agnar Aamodt. Case-based situation assessment in a mobile context-aware system. In *Artificial Intelligence in Mobile Systems*, pages 41–49, 2003.

[Kowalski and Sergot, 1989] Robert Kowalski and Marek Sergot. A logic-based calculus of events. In *Foundations of knowledge base management*, pages 23–55. Springer, 1989.

[Laput *et al.*, 2017] Gierad Laput, Yang Zhang, and Chris Harrison. Synthetic sensors: Towards general-purpose sensing. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 3986–3999, 2017.

[Lin *et al.*, 2003] Jessica Lin, Eamonn Keogh, Stefano Lonardi, and Bill Chiu. A symbolic representation of time series, with implications for streaming algorithms. In *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, pages 2–11, 2003.

[Liu *et al.*, 2017] Fagui Liu, Dacheng Deng, and Ping Li. Dynamic context-aware event recognition based on markov logic networks. *Sensors*, 17(3):491, 2017.

[Loke, 2006] Seng Wai Loke. On representing situations for context-aware pervasive computing: six ways to tell if you are in a meeting. In *Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW'06)*, pages 5–pp. IEEE, 2006.

[Loke, 2016] Seng W Loke. Representing and reasoning with the internet of things: a modular rule-based model for ensembles of context-aware smart things. *EAI endorsed transactions on context-aware systems and applications*, 3(8), 2016.

[McCarthy and Hayes, 1981] John McCarthy and Patrick J Hayes. Some philosophical problems from the standpoint of artificial intelligence. In *Readings in artificial intelligence*, pages 431–450. Elsevier, 1981.

[Mueller, 2004] Erik T Mueller. Event calculus reasoning through satisfiability. *Journal of Logic and Computation*, 14(5):703–730, 2004.

[Mueller, 2008] Erik T Mueller. Event calculus. *Foundations of Artificial Intelligence*, 3:671–708, 2008.

[Mueller, 2014] Erik T Mueller. *Commonsense reasoning: an event calculus based approach*. Morgan Kaufmann, 2014.

[Nelson and Mateas, 2008] Mark J Nelson and Michael Mateas. Recombinable game mechanics for automated design support. In *AIIDE*, 2008.

[Papadopoulos and Tzanetakis, 2016] Helene Papadopoulos and George Tzanetakis. Models for music analysis from a markov logic networks perspective. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(1):19–34, 2016.

[Patel *et al.*, 2002] Pranav Patel, Eamonn Keogh, Jessica Lin, and Stefano Lonardi. Mining motifs in massive time series databases. In *2002 IEEE International Conference on Data Mining, 2002. Proceedings.*, pages 370–377. IEEE, 2002.

[Perera *et al.*, 2013] Charith Perera, Arkady Zaslavsky, Peter Christen, and Dimitrios Georgakopoulos. Context aware computing for the internet of things: A survey. *IEEE communications surveys & tutorials*, 16(1):414–454, 2013.

[Poon and Domingos, 2006] Hoifung Poon and Pedro Domingos. Sound and efficient inference with probabilistic and deterministic dependencies. In *AAAI*, volume 6, pages 458–463, 2006.

[Richardson and Domingos, 2006] Matthew Richardson and Pedro Domingos. Markov logic networks. *Machine learning*, 62(1-2):107–136, 2006.

[Russel *et al.*, 2013] Stuart Russel, Peter Norvig, et al. *Artificial intelligence: a modern approach*. Pearson Education Limited, 2013.

[Schmidt, 2003] Albrecht Schmidt. *Ubiquitous computing-computing in context*. Lancaster University (United Kingdom), 2003.

[Shanahan and others, 1997] Murray Shanahan et al. *Solving the frame problem: a mathematical investigation of the common sense law of inertia*. MIT press, 1997.

[Shanahan, 2004] Murray Shanahan. An attempt to formalise a non-trivial benchmark problem in common sense reasoning. *Artificial intelligence*, 153(1-2):141–165, 2004.

[Shanahan, 2006] Murray Shanahan. Frame problem, the. *Encyclopedia of cognitive science*, 2006.

[Shoham and McDermott, 1988] Yoav Shoham and Drew McDermott. Problems in formal temporal reasoning. *Artificial Intelligence*, 36(1):49–61, 1988.

[Singla and Domingos, 2005] Parag Singla and Pedro Domingos. Discriminative training of markov logic networks. In *AAAI*, volume 5, pages 868–873, 2005.

[Skarlatidis and Michelioudakis, 2014] Anastasios Skarlatidis and Evangelos Michelioudakis. Logical Markov Random Fields (LoMRF): an open-source implementation of Markov Logic Networks, 2014.

[Skarlatidis *et al.*, 2015] Anastasios Skarlatidis, Georgios Paliouras, Alexander Artikis, and George A Vouros. Probabilistic event calculus for event recognition. *ACM Transactions on Computational Logic (TOCL)*, 16(2):1–37, 2015.

[Snidaro *et al.*, 2015] Lauro Snidaro, Ingrid Visentini, and Karna Bryan. Fusing uncertain knowledge and evidence for maritime situational awareness via markov logic networks. *Information Fusion*, 21:159–172, 2015.

[Sutton and McCallum, 2006] Charles Sutton and Andrew McCallum. An introduction to conditional random fields for relational learning. *Introduction to statistical relational learning*, 2:93–128, 2006.

[Yeh *et al.*, 2016] Chin-Chia Michael Yeh, Yan Zhu, Liudmila Ulanova, Nurjahan Begum, Yifei Ding, Hoang Anh Dau, Diego Furtado Silva, Abdullah Mueen, and Eamonn Keogh. Matrix profile i: all pairs similarity joins for time series: a unifying view that includes motifs, discords and shapelets. In *2016 IEEE 16th international conference on data mining (ICDM)*, pages 1317–1322. Ieee, 2016.

[Zhang *et al.*, 2019] Pei Zhang, Shijia Pan, Mostafa Mirshekari, Jonathon Fagert, and Hae Young Noh. Structures as sensors: Indirect sensing for inferring users and environments. *Computer*, 52(10):84–88, 2019.