Couch potato or gym addict? Semantic lifestyle profiling with wearables and fuzzy knowledge graphs

Natalia Díaz-Rodríguez*
U2IS, ENSTA ParisTech, France
natalia.diaz@ensta-paristech.fr

Aki Härmä and Rim Helaoui
Philips Research, Eindhoven, Netherlands
{aki.harma, rim.helaoui}@philips.com

Ignacio Huitzil and Fernando Bobillo
I3A, University of Zaragoza, Spain
{ihuitzil, fbobillo}@unizar.es

Umberto Straccia
ISTI-CNR, Pisa, Italy
umberto.straccia@isti.cnr.it

Abstract

Automatic lifestyle profiling to categorize users according to their daily routine-based lifestyles is an unexplored area. Despite the current trends on having wearable devices that generate large amounts of heterogeneous data, figuring out the lifestyle patterns of people is not a trivial task. We present *Lifestyles-KG*, a knowledge graph (fuzzy ontology) for semantic reasoning from wearable sensors. It can serve as a pre-processing taxonomical step that can be integrated into further prediction techniques for intuitively categorizing fuzzy *lifestyle* concepts, treats or profiles. The ultimate aim is to help tasks such as long-term human behavior classification and consequently, improve virtual coaching or customize lifestyle recommendation and intervention programs from free form non-labelled sensor data.

1 Introduction

Over the last decade a number of technologies have been developed that support individuals in keeping themselves active. This can be done via e-coaching mechanisms and by installing more advanced technologies in their homes.

This paper presents an ontology that allows to describe the lifestyle of a user given its digital footprints such as wrist-born activity trackers, GPS and mobile phone applications such as Moves[2]. The ontology includes concepts such as height, weight, locations, cholesterol, sleep, activity levels, activity energy expenditure, heart rate, or stress levels, among many other aggregated features. Its purpose is serving application development in Ambient Intelligence scenarios ranging from activity monitoring in smart homes to active healthy aging or lifestyle profiling.

The proposed ontology is a fuzzy version and augmentation of the wearables ontology in [17]. In this occasion, the fuzzy datatypes (ranges of the data properties) are learned from real data using clustering. This step can be seen as an extension to include in the fuzzy DL-Learner[3] tool. In particular, we consider fuzzy $\mathcal{EL}(\mathbf{D})$[4] concept descriptions (*lifestyles*) that explain sufficient conditions for being an

*Work done during internship at Philips Research

[2] https://moves-app.com/

[3] http://www.umbertostraccia.it/cs/software/FuzzyDL-Learner/index.html

[4] The basic elements of DLs are concept descriptions (in First-Order Logic terminology, unary predicates), inheritance relationships among them and instances of them. The standard language for ontology representation OWL 2 has three sublanguages with a polynomial time reasoning complexity (http://www.w3.org/TR/owl2-profiles). OWL 2 $\mathcal{EL}$ is a fragment that is particularly useful in applications employing ontologies that contain very large numbers of properties and/or classes; OWL 2 $\mathcal{QL}$ is designed to enable easier access

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

instance of such class according to the values of the datatype properties. Given a set of day (segments) records from digital traces such as sleep and activity sensors, the ontology serves to provide intuitive lifestyle treats categorizations. The definition of such categorizations is another contribution of the present work. The rest of this paper is structured as follows. Section 2 summarizes related approaches to ontological lifestyle profiling and description logics (DL) learning tools, Section 3 explains the learning methodology used, Section 4 describes the ontology proposed and Section 5 concludes the paper and discusses future work.

2 Related Work

2.1 Lifestyle profiling from digital traces and wearable sensors

Lifestyle can be defined as a collection of routines and behaviors shaped by the social, economic, and environmental structure around a person. In a computational application the behaviors are represented by measurements from wearable sensors. The lifestyle of an individual can then be modeled by the statistics of the measurements conditioned by the elements of the surrounding structure. A model of a lifestyle can be discovered blindly from the data using clustering methods or it may be based on matching a pre-defined semantic template to the data. [18] proposed a blind method to model routines as linear combinations of *eigenbehaviors.* The use of blind modeling is difficult because the discovered routines lack semantics that is needed to provide understandable feedback to the user. The lifestyle model proposed in [39] was based on matching a semantic template of workday and weekend routines to the wearable data. Semantic template models have also been used for mining personalized insights from wearable sensor data [24] to attach semantics to recurring ambulatory patterns [22].

Semantically meaningful and interpretable models to better understand the underlying statistics of individual lifestyle patterns of people is not a trivial task because of the variability of the individuals. Even if technology allows for a broad spectrum of sensors, it is not straight-forward to choose the most appropriate data acquisition, data imputation and data fusion techniques [29]. Attention should also be put into semantic definitions in order to achieve matching of lifestyle coaching programs, to target compatible profiles (i.e., accounting for the user's devices -and their metric units-, their diseases, time schedules and hobbies). Common-sense representation can enhance data-driven processes and improve accuracy and precision of recognition in human activities [16, 15, 13]. Likewise, knowledge-driven human activity models can be upgraded through data-driven learning techniques [3, 2] improving accuracy and clustering techniques.

2.2 Knowledge Graph unsupervised learning with fuzzy Description Logics

FOIL(First-order Inductive Learning) is a rule-based learning algorithm [30]. Related FOIL-like algorithms for the fuzzy case are reported in the literature [14, 33, 34] but they are not conceived for DL ontologies (see [36] for a discussion).

In the context of DL ontologies, DL-FOIL adapts FOIL to learn crisp OWL DL equivalence axioms under Open World Assumption (OWA)[5] [19]. Fuzzy DL-Learner supports the same learning problem as in DL-FOIL but implements algorithms that are not based on FOIL [23]. [10] combine a refinement operator for $\mathcal{EL}^{++}$ with a reinforcement learning algorithm and [25] can learn fuzzy OWL DL equivalence axioms from FuzzyOWL 2 ontologies [6] by interfacing the *fuzzyDL* reasoner [7].

[27] is the closest tool to DL-learner and presents FOIL-$\mathcal{DL}$, a FOIL variant for fuzzy and crisp DLs. However, the fuzzy probabilistic ensemble of fuzzy concept description candidates of DL-Learner proves more effective and furthermore, it does not face problems of other learners such as providing a mapping from fuzzy DLs to logic programming that is incomplete or its entailment can become undecidable [36].

---

and query to data stored in databases; and OWL 2 $\mathcal{RL}$ is a rule subset of OWL 2. The logic $\mathcal{EL}(\mathbf{D})$ is closely related to OWL 2 $\mathcal{EL}$.

[5]Under OWA (*Open World Assumption*), $\mathcal{E}^-$ consists of all those individuals which can be proved to be instance of $\neg T$, while under CWA (*Closed World Assumption*) $\mathcal{E}^-$ consists of those individuals which cannot be proved to be instance of T.

[6]`http://www.straccia.info/software/FuzzyOWL`

3 Methods: Fuzzy Description Logics, OWL and pFOIL Fuzzy DL-Learner

Fuzzy Description Logics (DLs) is a well-known family of logics for knowledge representation. They are the main formalism used to represent fuzzy ontologies; for example, the logical counterpart of the language Fuzzy OWL 2 is a fuzzy DL [6]. The fact of being fuzzy allows them to handle incomplete, imprecise or vague information. Fuzzy DLs have been used in a unlimited set of scenarios [35], principally, in Semantic Web applications or in distributed information retrieval to help in the integration of deep web resources [8], making heterogeneous information not only machine readable but also meaningful. The advantage of using DLs is the interpretability and readability of the concept definitions.

Fuzzy DL-Learner is an automatic learning system for OWL 2 ontologies. Given an OWL 2 ontology, it allows to learn graded OWL 2 descriptions of a target class in terms of specific inclusion axioms expressed in OWL $\mathcal{EL}$, in which fuzzy concepts and fuzzy datatypes may occur, such as *High Sleep Quality* or *Very low number of steps/calories* [36]. Fuzzy DL-Learner extends FOILalgorithm in [28]; it takes as input a crisp OWL ontology and can compute among others, fuzzy DLs datatypes via c-means[7] using various parameters, e.g., number of clusters, etc. Fuzzy DL-Learner allows to improve the readability of characterizations such as *Mediterranean-napper, Bar-Lover, Gym-Addict, Night-Owl* or *Sportive-Commuter*. It also permits to define the so-called *fuzzy concept descriptions* [36] such as a *Dutch-luncher* as a "worker that often takes sandwiches for lunch while walking along the lake", or a Mediterranean-Luncher as a person that "takes lunch at *Mediterranean-lunch-time* with *long-lunch-duration*". Here, the concept *Long-Lunch-Duration* is a so-called fuzzy concept [41], *i.e.* a concept for which the belonging of an individual to the class is not necessarily a binary yes/no question, but rather a matter of degree.

Fuzzy DL-Learner uses pFOIL-$\mathcal{DL}$, a method for the automated induction of fuzzy $\mathcal{EL}(\mathbf{D})$ concept descriptions by adapting the popular rule induction method FOIL [30]. But, unlike FOIL, in which the goodness of an induced rule is evaluated independently of previously learned rules, fuzzy DL-learner evaluates the goodness of a candidate in terms of the *ensemble* of so far learned rules, in a similar way as it happens in nFOIL [26]. However, when there are not enough examples of instances of a class, this learning technique is not appropriate. This happens for example in our case, where we have a lot of data property assertions (values of attributes for different people) but not a lot of class assertions (categorization of people as members of a concept). To overcome this situation, we implement clustering algorithms to learn the fuzzy datatypes.

4 Lifestyles-KG: a fuzzy ontology for lifestyle profiling

The percentage of day time spent doing certain activity can provide a pretty good idea of the person's type of lifestyle. However, the locations where a person spends his time, and the amount of times or frequency with which a person performs an activity or visits a place (supermarket, gym, doctor, library or theatre) can as well show great insights about a lifestyle. Lifestyle profiles can be formalized by using 4 different types of features or dimensions: time based features (in a [0,1] ratio per day-segment unit), clock features (i.e., start timestamp and duration), frequency (e.g., amount of steps or average nr. of key places visited in a commute), and location (lat. and longitude) features.

Table 1: Excerpt of sensor input data modeling and handling

Day Segment Feature	**Description /Unit**
Id	Day segment
StartT, EndT	start, end time [min]
Steps	count
Calories	count
Max, Mean HR	heart rate
Walking, Cycling, Running, Transport R	time ratio [0,1]
Cycling, Running, Transport and Walking distance	in m
Lunch walk steps	count in [11,13] PM

[7] in the yet unpublished latest version

The basic unit in which a *day footprint* is divided into is a *day segment*. Fixed day segments by clock 0-6, 6-12, 12-18, 18-24h exists, although regular ones such as *morning* (the time between waking up and going to work) are based on behaviour sensing. They are provided with start and end time -in minutes-, each of them divided into:

- *Home*, H – All places user calls 'Home' or where the user is last thing in the evening and first thing in the morning.
- *Work*, W – All places the user calls 'Work', 'Philips', 'HTC' or where the user regularly commutes to and from or is in the middle of a weekday.
- *Key* points, K – Frequented places in W <–> H transfers where the user regularly stops at (kindergarten, supermarket, fitness centre). K has no loops.
- *Other* places, O - All other places that users visit either from H or W. These will often be more sporadically visited places, such as restaurants, theaters, museums or other cities.

A Knowledge Graph (KG) in form of ontology contains concepts, individuals (concept instances) and relations in form of data and object properties. General lifestyle profile concepts in Lifestyles-KG include *Sedentary, Active, POI-Frequent-Visitor, Always-Late*, while more fine-grained lifestyles contained are *Shopaholic, Kindergarden visitor*, *Bar Lover*, etc. Each of these concepts' typical numerical values might be learned using fuzzy DL-Learner. Because the data is not provided with lifestyle labels, we apply clustering over the 40 records of volunteers of middle age living in the Eindhoven area (The Netherlands).

4.1 Machine learning workflow for fuzzy datatype learning

The long term goal is learning the rules, e.g. as general concept inclusions (GCIs); however, we start learning what the fuzzy DLs state of the art permits at the moment, i.e., fuzzy datatypes. The procedure followed for designing and learning the ontology datatypes is the following:

1. Manual design (e.g., using *NeON* methodology [37]) of a crisp ontology by the domain data scientists (together with diet and specialists that monitor cardiac disease patients), accounting for features of interest to perform queries on.
2. Use a learning algorithm to obtain from real data a fuzzy ontology that is consistent, non-redundant, sound, and complete [36]. The input is a crisp OWL ontology (.owl). The output is a fuzzy ontology which can be exported into Fuzzy OWL 2 [6] or into the *fuzzyDL* syntax (.fdl) for *fuzzyDL* reasoner[7][8] to be able to perform reasoning, e.g., with queries such as max-satisfiability degrees of individuals belonging to a given concept or satisfaction degree of axioms in a KB among others. See Table 2 for an excerpt of the *fuzzyDL* ontology and examples of queries.
   (a) We start by selecting a set of target atomic (in our case, lifestyle) concepts T for which one would like to learn classification rules.
   (b) If we have enough examples of instances of T, we can use fuzzy DL-Learner to obtain the rules. In particular, it provides fuzzy $\mathcal{EL}(\mathbf{D})$ GCIs axioms of the form "$C_1 \sqcap \ldots \sqcap C_n \sqsubseteq T$", where each C_i is either an atom A, or an existential restriction *(objectProperty some C)*, *(datatypeProperty some d)*, where d is a fuzzy membership function learned from data. Given the target concept name T, the *training set* $\mathcal{E}$ consists of crisp concept assertions of the form a:T where a is an individual occurring in $\mathcal{K}$.
   (c) If we do not have enough examples of instances of T, we use clustering algorithms such as k-means, fuzzy c-means or mean shift in order to learn fuzzy membership function values for each datatype that appears as a possible range of the attributes (data properties) for the instances of T. This is actually our case, thus in the following, we focus on learning the fuzzy datatypes (future work will add learned CGIs to the ontology).

[8] `http://www.umbertostraccia.it/cs/software/fuzzyDL/fuzzyDL.html`

```
% Concrete concepts (concrete features are generally a subset of all properties/relations)
(define-primitive-concept DaySegment *top* )
(define-primitive-concept NightAndMorning DaySegment)
(define-primitive-concept CommuteToWork WeekdaySegment)
(define-primitive-concept AtWork WeekdaySegment)
(define-primitive-concept CommuteSegment (g-or CommuteToWork CommuteToHome))
(define-primitive-concept OutHomeSegment (g-or AtWork CommuteToHome CommuteToWork DayTime ))
(define-primitive-concept NotHomeNorWorkSegment (g-or CommuteToHome CommuteToWork DayTime ))

% Axioms expressing concrete features or datatypes' range
(range hasStartT *integer* 0 1440) # in min
(range hasLunchStartT *integer* 0 1440 )
(range walkedNStepsAtLunch *integer* 0 999999)
(range spentNCalories *integer* 0 999999 )
(range hasMaxHR *integer* 0 1000)

% Fuzzy datatypes (define-fuzzy-concept HighHR right-shoulder (30, 300, 85, 100))
(define-fuzzy-concept LowHR left-shoulder (30, 300, 50, 62))
(define-fuzzy-concept GymSessionDuration trapezoidal (0, 1440, 15, 20, 45, 90))
(define-fuzzy-concept MediterraneanLunchStartT trapezoidal (0, 1440, 690, 720, 780, 830)) # 11.30, 12, 13, 13.30 am
(define-fuzzy-concept MediterraneanJobStartT trapezoidal (0, 1440, 360, 420, 480, 540)) # 6, 7, 8, 9 am

% Definition of lifestyle fuzzy concepts (actual definitions corresponding to rules)
(define-concept DutchLuncher (g-and AtWork (some walkedNStepsAtLunch HighNSteps)(some hasLunchStartT Dutch-
LunchStartT)(some hasLunchDuration DutchLunchDuration) (some spentNCalories MediumCalorieCountMan)))
(define-concept MediterraneanLuncher (g-and AtWork (some walkedNStepsAtLunch LowNSteps)(some hasLunch-
StartT MediterraneanLunchStartT)(some hasLunchDuration MediterraneanLunchDuration) (some spentNCalories
LowCalorieCountMan)))
(define-concept DutchWorker (g-and AtWork (some hasEndT DutchJobEndT)(some hasStartT DutchJobStartT )))
(define-concept MediterraneanWorker (g-and AtWork (some hasEndT MediterraneanJobEndT)(some hasStartT Mediter-
raneanJobStartT)))
(define-concept EarlyBird (g-and NightAndMorning (some hasEndT VeryEarlyLeaveHomeTime)(some hasStartT
EarlyArriveHomeTime)))
(define-concept Sedentary (g-and DaySegment (some hasMaxHR LowHR ) (some hasAvgHR LowHR) (some spent-
NCalories LowCalorieCountMan) (some walkedNSteps LowNSteps) (some hasWalkingDistance ShortWalkingDistance)
(some hasWalkingR LowWalkingR) (some hasCyclingDistance ShortCyclingDistance) (some hasCyclingR LowCy-
clingR) (some hasTransportDistance LongTransportDistance) (some hasTransportR HighWalkingR)))

% Input axioms to the Knowledge Base: Location-based Day Segment for e.g., tm (this morning)
(define-fuzzy-concept startT_tm crisp(0,1440,0,0))
(define-fuzzy-concept nCalories_tm crisp(0,1440,0,0))
(define-fuzzy-concept maxHR_tm crisp(0,1440,0,0))
(define-fuzzy-concept transportDistance_tm crisp(0,1440,0,0))
(define-fuzzy-concept cyclingR_tm crisp(0,1440,0,0))


% Input instances (individuals) (define-concept day1segment1 (g-and NightAndMorning (= hasStartT 0) (= hasEndT
519) (= walkedNSteps nSteps_tm) (= walkedNStepsAtLunch 0) (= spentNCalories 0)

% Queries: max. satisfiability of matching footprint day1segment1 with each lifestyle profile (LP)
(max-sat? (and GymAddict day1segment1))
(max-sat? (and DutchLuncher day1segment1))
```

Table 2: Lifestyles-KG ontology, a profile library (excerpt of ontology and rules in *fuzzyDL*).

4.2 Learning fuzzy datatypes

We use unsupervised learning algorithms to cluster real data values of the data properties. The result of our application is an array of k centroids (one per cluster) from which we compute the fuzzy membership functions of the fuzzy datatypes. We implemented k-means, fuzzy c-means, and mean shift clustering algorithms. Some advantages of our approach are its independence of the application domain and not requiring any manual intervention. In the case of mean-shift no parameters are required; the other algorithms require to fix a priori the number of clusters k.

K-means groups a set of n data x_j into k clusters described by means of their centroids c_i [5]. The algorithm starts by computing randomly the k initial centroids c_i. Then, it repeats two steps: first, each point x_i is assigned to its nearest centroid c_i according to the Euclidean distance. Second, the centroids c_i are updated by minimizing a square error function. The algorithm finishes when a stopping criteria is met (total iteration number or no changes in centroids).

Fuzzy c-means [4] is an extension where every point can belong to several clusters with different degrees of membership. This algorithm is more robust to the random initialization of the centroids.

Mean-shift [9, 11] is widely used in clustering but also in image segmentation. It seeks modes or local maxima of density in a feature space by computing a mean-shift vector. The algorithm defines a window around each point, computes the mean of the data points in the window and then shifts its center to the mean. It uses a Gaussian Kernel K_g to keep track of the nearest neighbors of each x_i according to a bandwidth or window size h. To compute the bandwidth we use the rule of thumb proposed in [38]. This rule can be used to compute a quick estimation of h for a given K_g, and allows to define a local seeking distance $l = \frac{h}{2}$. At the end of the process, the mean-shift vector converges into a set of centroids after removing data points at a too close distance.

5 Conclusions and Future work

We proposed *Lifestyles-KG*, a Knowledge Graph that can learn his original data properties in terms of descriptive fuzzy $\mathcal{EL}(\mathbf{D})$ sub-concepts. The fuzzy datatypes in our fuzzy ontology were automatically learnt from data records using clustering algorithms. Our learning strategy is complementary to the existing implementation of the fuzzy DL-Learner and could be used to extend the tool. *Lifestyles-KG* models heterogeneous sensor data from a large variety of wearable (sleep, HR, GSR) sensors, phone apps and scales, etc. The ontology, both in crisp and *fuzzyDL* syntax is available online[9].

This work focuses on learning fuzzy datatypes; however, future work should aim at automatizing a larger part of the KB design procedure, e.g., learning concepts and sub-concepts, as well as relations and their cardinality, functionality, etc. Reproducibility, i.e., producing another KB instance for a new domain (e.g., classifying usage patterns of new types of public transportation) reusing existing ontologies is the most challenging but interesting area of future research.

Future promising directions can be applying deep networks for learning relational reasoning [32] and other properties from richer (image) data [31] and interactions [40]. Another interesting idea is using a deep reinforcement or active learning setting [10]. This would serve both finer grained activity recognition, but also, more generally, learning state representations for real-time control and advice. We plan to improve the ontology avoiding design pitfalls [10] and applying use-case adapted tools towards coverage evaluation. Since scalability and suitability of an ontology can be only evaluated depending on a given set of requirements, using CQOA (Competency Question-driven Ontology Authoring) tests to assess whether a given *competency question* is able to be answered by the ontology [12] seems promising. Other ideas are enhancing missing data with crowd-sourced approaches such as in social LSTM [1], as well as using unsupervised online discovery of periodic and stationary patterns with temporal behaviour assumptions [21] and mobile phone only datasets [20][11].

Acknowledgments

We acknowledge AAPELE.eu EU COST Action IC1303 and EU Erasmus+ Funding. I. Huitzil was partially funded by Universidad de Zaragoza – Santander Universidades (Ayudas de Movilidad para Latinoamericanos - Estudios de Doctorado).

[9] `https://github.com/NataliaDiaz/Ontologies`

[10] `http://oops.linkeddata.es/`

[11] Sussex-Huawei Locomotion (SHL) dataset `www.shl-dataset.org`

References

[1] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 961–971, 2016.

[2] L. Atallah and G.-Z. Yang. The use of pervasive sensing for behaviour profiling — a survey. *Pervasive and Mobile Computing*, 5(5):447 – 464, 2009.

[3] G. Azkune, A. Almeida, D. L. de Ipiña, and L. Chen. Extending knowledge-driven activity models through data-driven learning techniques. *Expert Systems with Applications*, 42(6):3115 – 3128, 2015.

[4] J. C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Advanced Applications in Pattern Recognition. Plenum Press, 2nd edition, 1987.

[5] J. C. Bezdek, R. Ehrlich, and W. Full. Fcm: The fuzzy c–means clustering algorithm. *Computers & Geosciences*, 10(2):191–203, 1984.

[6] F. Bobillo and U. Straccia. Fuzzy ontology representation using OWL 2. *International Journal of Approximate Reasoning*, 52(7):1073–1094, 2011.

[7] F. Bobillo and U. Straccia. The fuzzy ontology reasoner fuzzyDL. *Knowledge-Based Systems*, 95:12–34, 2016.

[8] A. Calì and U. Straccia. Integration of deep web sources: A distributed information retrieval approach. In *Proceedings of the 7th International Conference on Web Intelligence, Mining and Semantics (WIMS-17)*, pages 33:1–33:4, New York, NY, USA, 2017. ACM.

[9] Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17(8):790–799, Aug. 1995.

[10] M. Chitsaz, K. Wang, M. Blumenstein, and G. Qi. Concept learning for $\mathcal{EL}^{++}$; by refinement and reinforcement. In *Proceedings of the 12th Pacific Rim international conference on Trends in Artificial Intelligence*, PRICAI'12, pages 15–26, Berlin, Heidelberg, 2012. Springer-Verlag.

[11] D. Comaniciu, P. Meer, and S. Member. Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:603–619, 2002.

[12] M. Dennis, K. van Deemter, D. Dell'Aglio, and J. Z. Pan. *Computing Authoring Tests from Competency Questions: Experimental Validation*, pages 243–259. Springer International Publishing, Cham, 2017.

[13] N. Díaz-Rodríguez, M. P. Cuéllar, J. Lilius, and M. D. Calvo-Flores. A survey on ontologies for human behavior recognition. *ACM Computing Surveys (CSUR)*, 46(4):43, 2014.

[14] M. Drobics, U. Bodenhofer, and E.-P. Klement. FS-FOIL: an inductive learning method for extracting interpretable fuzzy descriptions. *Int. J. Approximate Reasoning*, 32(2-3):131–152, 2003.

[15] N. Díaz-Rodríguez, O. L. Cadahía, M. P. Cuéllar, J. Lilius, and M. D. Calvo-Flores. Handling real-world context awareness, uncertainty and vagueness in real-time human activity tracking and recognition with a fuzzy ontology-based hybrid method. *Sensors*, 14(10):18131–18171, 2014.

[16] N. Díaz Rodríguez, M. P. Cuéllar, J. Lilius, and M. Delgado Calvo-Flores. A fuzzy ontology for semantic modelling and recognition of human behaviour. *Knowledge-Based Systems*, 66(0):46 – 60, 2014.

[17] N. Díaz Rodríguez, S. Grönroos, F. Wickström, J. Lilius, H. Eertink, A. Braun, P. Dillen, J. Crowley, and J. Alexandersson. An ontology for wearables data interoperability and ambient assisted living application development. In V. Kreinovich, M. Z. Reformat, and R. R. Yager, editors, *World Conference on Soft Computing*, page 1–5. Springer, 2016.

[18] N. Eagle and A. S. Pentland. Eigenbehaviors: Identifying structure in routine. *Behavioral Ecology and Sociobiology*, 63(7):1057–1066, 2009.

[19] N. Fanizzi, C. d'Amato, and F. Esposito. DL-FOIL concept learning in description logics. In F. Zelezný and N. Lavrač, editors, *Inductive Logic Programming*, volume 5194 of *Lecture Notes in Computer Science*, pages 107–121. Springer, 2008.

[20] H. Gjoreski, M. Ciliberto, F. J. Ordoñez Morales, D. Roggen, S. Mekki, and S. Valentin. A versatile annotated dataset for multimodal locomotion analytics with mobile devices. 2017.

[21] H. Gjoreski and D. Roggen. Unsupervised online activity discovery using temporal behaviour assumption. In *Proceedings of the 2017 ACM International Symposium on Wearable Computers*, ISWC '17, pages 42–49, New York, NY, USA, 2017. ACM.

[22] A. Härmä and K. de Groot. Automatic characterization of ambulatory patterns of utilitarian and leisure trips. *PeerJ Preprints*, 5:e2751v1, 2017.

[23] S. Hellmann, J. Lehmann, and S. Auer. Learning of OWL class descriptions on very large knowledge bases. *Int. J. Semantic Web Inf. Syst.*, 5(2):25–48, 2009.

[24] A. Härmä and R. Helaoui. Probabilistic scoring of validated insights for personal health services. In *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–6, Dec 2016.

[25] J. Iglesias and J. Lehmann. Towards integrating fuzzy logic capabilities into an ontology-based inductive logic programming framework. In *Proceedings of the 11th International Conference on Intelligent Systems Design and Applications (ISDA 2011)*, pages 1323–1328, 2011.

[26] N. Landwehr, K. Kersting, and L. D. Raedt. Integrating naïve bayes and FOIL. *Journal of Machine Learning Research*, 8:481–507, 2007.

[27] F. A. Lisi and U. Straccia. A system for learning GCI axioms in fuzzy description logics. In *Proceedings of the 26th International Workshop on Description Logics (DL-13)*, volume 1014 of *CEUR Workshop Proceedings*, pages 760–778. CEUR-WS.org, 2013.

[28] F. A. Lisi and U. Straccia. A FOIL-like method for learning under incompleteness and vagueness. In *23rd International Conference on Inductive Logic Programming*, volume 8812 of *Lecture Notes in Artificial Intelligence*, pages 123–139, Berlin, 2014. Springer Verlag. Revised Selected Papers.

[29] I. M. Pires, N. M. Garcia, N. Pombo, and F. Flórez-Revuelta. From data acquisition to data fusion: A comprehensive review and a roadmap for the identification of activities of daily living using mobile devices. *Sensors*, 16(2):184, 2016.

[30] J. R. Quinlan. Learning logical definitions from relations. *Machine learning*, 5(3):239–266, 1990.

[31] D. Raposo, A. Santoro, D. Barrett, R. Pascanu, T. Lillicrap, and P. Battaglia. Discovering objects and their relations from entangled scene representations. *arXiv preprint arXiv:1702.05068*, 2017.

[32] A. Santoro, D. Raposo, D. G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, and T. Lillicrap. A simple neural network module for relational reasoning. *arXiv preprint arXiv:1706.01427*, 2017.

[33] M. Serrurier and H. Prade. Improving expressivity of inductive logic programming by learning different kinds of fuzzy rules. *Soft Computing*, 11(5):459–466, 2007.

[34] D. Shibata, N. Inuzuka, S. Kato, T. Matsui, and H. Itoh. An induction algorithm based on fuzzy logic programming. In N. Zhong and L. Zhou, editors, *Methodologies for Knowledge Discovery and Data Mining, Third Pacific-Asia Conference, PAKDD-99, Beijing, China, April 26-28, 1999, Proceedings*, volume 1574 of *Lecture Notes in Computer Science*, pages 268–273. Springer, 1999.

[35] U. Straccia. All about fuzzy description logics and applications. In *Reasoning Web, 11th International Summer School, Tutorial Lectures*, volume 9203 of *Lecture Notes in Computer Science*, pages 1–31. Springer Verlag, 2015.

[36] U. Straccia and M. Mucci. pFOIL-DL: Learning (fuzzy) $\mathcal{EL}$ concept descriptions from crisp OWL data using a probabilistic ensemble estimation. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing (SAC-15)*, pages 345–352, Salamanca, Spain, 2015. ACM.

[37] M. C. Suárez-Figueroa, A. Gómez-Pérez, and M. Fernández-López. The neon methodology for ontology engineering. In *Ontology engineering in a networked world*, pages 9–34. Springer, 2012.

[38] B. A. Turlach. Bandwidth selection in kernel density estimation: A review. In *CORE and Institut de Statistique*.

[39] A. van Wissen, A. Härmä, R. Helaoui, I. C. Gyllensten, and D. Lowet. Optimization of automated health programs by simulating user behaviors and program effects. In *Proceedings of the Conference on Measuring Behaviour*, 2016.

[40] N. Watters, A. Tacchetti, T. Weber, R. Pascanu, P. Battaglia, and D. Zoran. Visual interaction networks. *CoRR*, abs/1706.01433, 2017.

[41] L. A. Zadeh. Fuzzy sets. *Information and control*, 8(3):338–353, 1965.