A Domain Ontology and Software Platform for Collaborative Personal Data Analytics *

Lauri Tuovinen[1,2][0000−0002−7916−0255] and
Alan F. Smeaton[1][0000−0003−1028−8389]

[1] Insight Centre for Data Analytics, Dublin City University, Dublin, Ireland
alan.smeaton@dcu.ie
[2] Biomimetics and Intelligent Systems Group, University of Oulu, Oulu, Finland
lauri.tuovinen@oulu.fi

Abstract. Collaborative knowledge discovery is a promising approach by which people with no data analytics expertise could benefit from an analysis of their own personal data by experts. To facilitate effective collaboration between data owners and knowledge discovery experts, we have developed a software platform that uses a domain ontology to represent knowledge relevant to the execution of the collaborative knowledge discovery process. The ontology provides classes representing the main elements of collaborations: collaborators and datasets. Furthermore, the ontology enables the specification of privacy constraints that determine the precise extent to which a given dataset of personal data is shared with a given collaborator. We have developed a client-server software platform that enables users to initiate collaborations, invite experts to join them, create datasets and share them with experts, and create visualisations of data. The collaborations are mediated through the creation, modification and deletion of individuals in the underlying ontology and the propagation of ontology changes to each client connected to the server.

Keywords: Knowledge discovery · Data analytics · Collaborative systems · Domain ontologies · Personal data.

1 Introduction

Collection of self-tracking data using various consumer-oriented wearable devices such as fitness trackers and sleep trackers is becoming increasingly commonplace. These devices and the software applications bundled with them provide the user with basic information such as steps taken, calories expended and hours slept. However by applying knowledge discovery from data (KDD), it would be possible for a user to extract additional knowledge from their own data, but most

* The work of Lauri Tuovinen is funded by the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement number 746837. The Insight Centre is funded by Science Foundation Ireland under grant number SFI/12/RC/2289, and is co-funded under the European Regional Development Fund.

people lack the knowledge and data analytics skills needed in order to carry out such analyses. To get around this problem, the data owner could collaborate remotely with a KDD expert who has those required skills using a software platform designed to support this type of collaboration. There are such platforms available, but none are specifically intended for use by non-expert individuals to extract knowledge from their personal data with the help of expert collaborators.

Supporting collaborative KDD in this special case introduces requirements that need to be taken into account in the design of the collaboration platform. Our work particularly focuses on supporting the data owner in the task of negotiating the terms of the collaboration with the KDD expert. Initially, we examined the process of establishing the boundaries of how much data the data owner will share with the expert; these are determined by a number of factors, including the data owner's personal privacy preferences, the objectives of the collaboration and the level of trust existing between the data owner and the expert. The software platform should help the data owner set boundaries that represent an acceptable trade-off between potentially conflicting requirements.

The solution we developed is based on a domain ontology representing the core concepts of collaborative KDD. The software platform uses the ontology as a knowledge base where all known information about past and ongoing collaborations is recorded. A user of the platform can create new collaborations, invite other users to join them, create new datasets composed of their own personal data and share them with collaborators including experts. When a request is made to share a dataset, the owner of the dataset can attach privacy constraints allowing the requester to access only some of the data included. Expert collaborators can use the software platform to share their analysis results and attach visualisations making it easier for the expert to explain their significance.

The main contributions of the paper are:

- A domain ontology representing collaborations, collaborators, datasets, privacy constraints and visualisations;
- A software implementation that uses the ontology to facilitate collaboration via data sharing.

The results reported in this paper refine and give a more concrete form to ideas originally discussed in [17], where the ontology was presented as a standalone artifact and tested using a reasoner to show that certain inferences concerning e.g. the scope of privacy constraints are made correctly. In the current paper, the ontology is put to practice by using it as a key component in a collaborative KDD software platform, where it provides the shared data structure in which the state of a collaboration is stored and which the participants of the collaboration can synchronously edit. Additionally, this version of the ontology includes classes and properties that were not present in [17], having to do with the representation of datasets, expertise, collaboration invitations, data requests and visualisations.

The remainder of the paper is organised as follows: Section 2 discusses the motivation for our work and reviews related research. Section 3 presents the classes of the ontology and the relationships among them. Section 4 describes the implementation of the software platform. Section 5 presents a critical discussion of the results and identifies key issues to be addressed by future work. Section 6 concludes the paper.

2 Background and Motivation

It is widely recognised that personal data constitutes a valuable resource for companies that control large quantities of it, and because of this, companies are willing to provide services free of charge in exchange for being allowed to collect and use it. Trading personal data for access to services is one example of how individuals can benefit from their own personal data, but it is not the only option. With the rising popularity of self-tracking products, an increasing number of people have access to a steadily accumulating database of physiological data about themselves, and this data is potentially valuable to them, although there are problems arising from uncertainties concerning the accuracy of the data and limitations concerning the ability of the people to control it [18].

The application or cloud service by which the user of a self-tracking device has access to their own data may already have some data analysis capabilities, but for many requirements the only option is to export the data and use another application to analyze it. For example, some users may wish to combine data from a sleep tracking application and a food intake logger, or to combine a tracker for their exercise level or step counter with a digital weighing machine. Others may wish to see long-term trends in their time spent online, or seasonal variations in their time commuting to work. Some services make it convenient to export personal data in a portable format such as CSV, but this does not help unless the user has the skills required to do the desired analysis. For most users this means that in order to extract knowledge hidden in their own data, they need to collaborate with someone who does have those skills.

The kind of collaboration we support involves an online software platform capable of bringing together people who have data with people who have data analysis skills regardless of where they are physically located. Using the platform, a person who has collected self-tracking data can find a KDD expert with the required skill set, negotiate with the expert to agree on the terms of the collaboration, collaboratively analyze the data and evaluate the results. Collaborative KDD platforms such as KDDVM [5] and LabBook [9] already exist, but to the best of our knowledge there are none that specifically target the sub-domain of personal analytics.

What makes personal analytics an interesting special case of collaborative KDD are the unique requirements that arise from the sensitive nature of the data and from the key role of non-expert participants in the collaboration. The rationale for proposing an ontology-based platform for facilitating such collaborations is that a comprehensive domain ontology would address many of these requirements. One major requirement, namely providing the ability to find expert collaborators, is already addressed by existing systems; the KDDVM platform uses an ontology named TeamOnto [4] to represent expertise, whereas LabBook has an underlying metadata graph that is functionally similar to an ontology.

Another purpose for which KDD ontologies can be used is to provide intelligent assistance in the composition of KDD workflows, which would be particularly important when dealing with a user who is not well versed in the application of KDD tools. KDDesigner [3], the workflow design tool of the KDDVM platform, uses the KDDONTO ontology [2] to support this, although it is unclear whether it is intended that even a complete novice should be able to use the tool to create workflows. Other recently proposed KDD ontologies include the KD ontology of [19], the data mining workflow ontology DMWF [11], the data mining optimisation ontology DMOP [10], the OntoDM family of ontologies [14–16] and the unnamed big data ontology of [12].

A part of the collaborative KDD process that is notably less supported by existing solutions is negotiation of the terms of the collaboration. Supporting this part would be important in personal analytics because it is arguably here that the data owner will have the most substantial impact on the outcome of the collaboration, and also because the data owner's expected low level of expertise may make it difficult to understand all the implications of what is being agreed. We hypothesise that a domain ontology could be used for this purpose as well, both as a knowledge base for intelligent assistance and as a way of representing and enforcing the results of the negotiation.

Our main concern at this point and in this paper is the negotiation of privacy constraints, where participants establish the boundaries of data sharing in the collaboration. A negotiation is necessary because the boundaries are not simply a matter of the data owners specifying their personal privacy preferences: there are multiple points of view to take into account and between these there may be conflicts that need to be resolved. Ontology-based approaches to privacy protection have been proposed in e.g. [1, 6–8], but the idea of creating privacy policies through a process of negotiation is mostly absent in these related works, as is that of data owners working in cooperation with data analysts. Our work can thus be viewed as filling a gap at the intersection of the domains of knowledge discovery, collaboration and privacy.

3 Ontology Classes and Properties

In the discussion below, the names of ontology classes are written with initial capitals. Additionally, when mentioned for the first time, the names are written in boldface. The names of object properties, when mentioned for the first time, are written in italics. The Protégé ontology development environment [13] was used to develop the ontology. The seven core classes of the ontology as used by the software platform are:

- **Collaboration**, representing a collaborative KDD project;
- **Collaborator**, representing a person participating in a Collaboration;
- **Dataset**, representing a collection of personal data used in Collaboration;

- **DataItem**, representing an individual item of data, such as a single numeric value;
- **DataRequest**, representing a request to share a given dataset;
- **PrivacyConstraint**, representing a constraint specified when a DataRequest is granted;
- **Visualisation**, representing a graphical representation of some data, usually the result of some data analysis by an expert.

A Collaboration may have any number of Collaborators participating in it, signified by the *hasParticipant* property. The Collaborator who starts a Collaboration is designated the leader, signified by the *isLeaderOf* property, and has the ability to add new participants. Similarly, a Collaboration may have any number of Datasets involved in it, signified by the *hasDataset* property. The Collaborator who creates a Dataset is designated the owner, signified by the *controls* property, and has the ability to determine how the Dataset is shared. The data owner role is represented by the Collaborator subclass **DataOwner**.

A Collaborator who is an expert in some areas has one or more **Expertise** individuals attached to it via the *hasExpertise* property. Once a suitable expert has been identified, an **Invitation** can be created by the leader of the Collaboration to request that the expert join it. The Invitation is linked to the Collaboration, the sending Collaborator and the receiving Collaborator by the *isForCollaboration*, *wasCreatedBy* and *isForCollaborator* properties, respectively. The expert role is represented by the Collaborator subclass **Expert**.

The Dataset class has three subclasses, **DataColumn**, **DataMatrix** and **DataCollection**. These are organised in a hierarchy where a DataCollection may have any number of DataMatrices as subsets, and a DataMatrix (essentially a table) in turn may have any number of DataColumns, which are typed (represented by the **DataType** class and the *hasDataType* property) and ordered one-dimensional collections of DataItems. The inclusion of a DataItem in a Dataset is signified by the *contains* property and the inclusion of a Dataset in another Dataset by the transitive *hasSubset* property. A Dataset is inferred via a property chain to contain all DataItems contained by its subsets.

A DataRequest targets a specific Dataset in a specific Collaboration, signified by the *isForDataset* and isForCollaboration properties, respectively. The Collaborator who created the request is connected to it via the wasCreatedBy property. The owner of the requested Dataset can grant the request, deny it or grant it with constraints. If the data owner specifies constraints, the PrivacyConstraint individuals are attached to the DataRequest via the *respondsTo* property. Each PrivacyConstraint has a scope, i.e. a specific set of DataItems that it applies to. The *appliesTo* property connects a PrivacyConstraint to each DataItem in its scope; if a given DataItem is found to be in the scope of a PrivacyConstraint, then it is not shared with the requesting Collaborator.

In practice, the scope is generally specified in terms of a **DataGrouping**, which is either a Dataset or a **DataCategory**. DataCategories can be used to group together DataItems that are not members of the same Dataset but are otherwise related; for example, it might be desirable to specify a PrivacyConstraint applying to all DataItems representing location data regardless of which Dataset they are included in. The *hasInstance* property connects a DataCategory to each DataItem belonging to it, and the *hasSubcategory* property signifies that a given DataCategory subsumes another DataCategory under it. The *hasScope* property is used to specify the DataGrouping that a PrivacyConstraint applies to; the individual DataItems in the scope of the PrivacyConstraint can then be inferred via property chains.

When a Collaborator creates a Visualisation of a Dataset, the ontology individual is linked to the Collaborator, the Dataset and the Collaboration by the wasCreatedBy, isForDataset and isForCollaboration properties, respectively. The details of the Visualisation are controlled by parameters attached to the individual as data properties; the parameters are specific to the type of the Visualisation and consist of all the information required to reproduce its appearance as designed by its creator. The ontology classes described above and the relationships among them are illustrated in Figure 1, where arrows with a solid line denote subclass-superclass relationships, while those with a dashed line denote object properties. For the sake of clarity, the ontology is presented in two parts and some classes and properties are omitted.

The ontology additionally specifies certain classes and properties that are not yet used by the software platform but have been tested in Protégé. The classes **AnalysisTask** and **AnalysisMethod** are intended to enable expert collaborators to specify the KDD operations to be carried out in terms of their inputs and outputs; the output Datasets are linked to the inputs via the transitive *isDerivativeOf* property, which enables the owners of the original input Datasets to exert control over the derivatives as well. For the data owners, the classes **AccessRestriction** and **ProtectionMethod** enable the specification of PrivacyConstraints that do not simply block access to data but transform it in some way to render it less sensitive. The **UtilityReduction** class represents the negative effect that a given ProtectionMethod has on the utility of a given AnalysisMethod. Further details on the ontology and these five classes in particular can be found in [17], but the reader should note that the current paper extends the ontology from the version described there; most notably, the classes Expertise, Invitation, DataRequest and Visualisation as well as the subclasses of Dataset only exist in the current version.

4 Software Architecture and Implementation

The software platform is a client-server system implemented in Java using the Apache MINA network application framework. The server component maintains a user database and a master copy of the ontology; when a client logs in to the server, the client downloads the ontology from the server and sets up a local copy. The OWL API and the JFact reasoner are used to process the ontology. Any modifications made by the client, such as creation of a new collaboration or dataset, are first applied to the local copy, then sent to the server, applied to the master copy and sent to other clients.

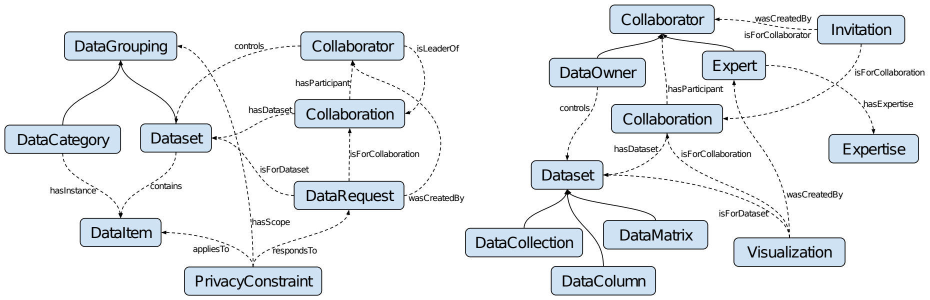


Fig. 1: The classes and object properties of the collaborative KDD ontology.

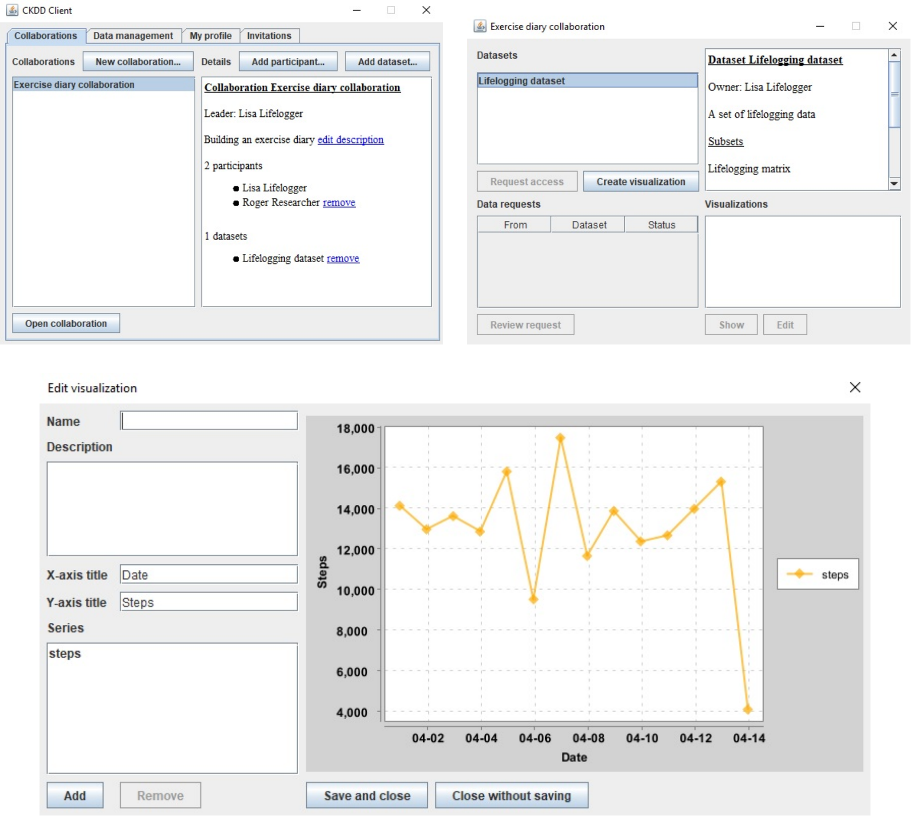
When a data owner initiates a collaboration, a corresponding Collaboration individual is created and added to the ontology. When the owner creates a new dataset, initially a DataCollection individual is created. The owner can then import data into the dataset from a CSV file and the software will generate DataMatrix, DataColumn and DataItem individuals according to the contents of the imported file. The data itself will be kept locally on the client host and not uploaded to the server until it is shared with another collaborator. Apache Commons CSV is used to process CSV files.

To invite an expert to join the collaboration, the data owner first performs an expert search by specifying some keywords characterising the desired expertise. The software then uses a simple algorithm to generate a list of experts ranked according to how closely their expertise matches the query. Once the data owner has selected an expert to invite, an Invitation individual is added to the ontology, triggering a notification in the client used by the expert in question. If the invitation is accepted, the expert will then be added as a participant.

Data sharing begins with the expert requesting it, which causes a DataRequest individual to be added to the ontology and triggers a notification in the client used by the data owner. The owner can then review the request and specify how much of the data will be shared. If the owner chooses to grant the request but deny access to some subsets, PrivacyConstraint individuals corresponding to these subsets will be created and added to the ontology, causing the data items in these subsets to be unavailable to the expert.

Once a data request has been granted, the data will be uploaded to the server and the expert's client notified. After the expert's client has downloaded the data, it will have a copy of the dataset that is identical to the one held by the data owner, with the exception of data items blocked by privacy constraints. The expert can then export the dataset and use a KDD tool of his/her choice such as Tableau or Python to analyze it because that is the environment the expert will be familiar with. The results of the analysis can be shared with the data owner via the same mechanism used by the owner to share the input data.

To illustrate some analysis results, the expert has the option of attaching visualisations and descriptive analysis to the result dataset. Visualisations are created by launching a dialog that allows the expert to specify the parameters of the visualisation. Once a visualisation has been created and saved, it becomes visible to other users participating in the collaboration and can be viewed by those who have access to the underlying data. When a user views a visualisation, the client reproduces it based on the parameters attached to the corresponding Visualisation individual as data properties. The visualisations are rendered using the XChart library. Some screenshots of the software can be seen in Figure 2, showing a collaboration between a user and an expert who is analysing long-term step count data looking for regular and repeating patterns.



Collaboration Exercise diary collaboration

Leader: Lisa Lifelogger

Building an exercise diary edit description

2 participants

- Lisa Lifelogger
- Roger Researcher remove

1 datasets

- Lifelogging dataset remove

Dataset Lifelogging dataset

Owner: Lisa Lifelogger

A set of lifelogging data

Subsets

Lifelogging matrix

Name:

Description:

X-axis title: Date

Y-axis title: Steps

Series: steps

Fig. 2: Some screenshots of the collaborative KDD client. Top left: main window, top right: collaboration window, bottom: visualisation window.

5 Discussion and Future Work

The ontology was originally designed with the idea that the underlying knowledge base of the collaboration platform should enable the resolution of privacy conflicts by suggesting transformations that achieve an acceptable trade-off between the information content of the data and the privacy preferences of the data owner. This goal is reflected in the classes and properties described briefly at the end of Section 3. The details of this, such as how to represent and quantify utility reductions, are a topic for future research, as are connections between our ontology and related ones such as those cited in Section 2.

The software platform demonstrates the feasibility of the concept of using an ontology to mediate collaboration for knowledge discovery from personal data. The ontology functions as a shared view of the state of the collaboration that each collaborator can observe and modify, and the propagation of modifications through the server component ensures that the collaborators remain synchronised with one another. As we continue to refine and expand the ontology, new functionality will be added to the software accordingly.

The scenario we implicitly assume when discussing the software, involving one data owner, one expert and one dataset, is simplistic and not necessarily representative of the kind of real-world collaborations the ontology and software can support. It could even be argued that under these assumptions the software makes it unnecessarily complicated to execute the scenario by requiring data requests to be submitted before any data can be shared. However, to avoid imposing unwanted restrictions on the number of datasets and collaborators, it is important to provide data owners with the means to manage multiple datasets and to deal with the data requirements of multiple expert partners. Multiple data owners in the same collaboration is also a possibility we take into consideration.

6 Conclusion

In this paper we presented an ontology-based software platform for collaborative knowledge discovery from personal data. The software enables individuals with no data analytics expertise to collaborate with analytics experts for the purpose of extracting useful knowledge from their own personal data, such as what can be collected using wearable activity and sleep trackers. The underlying ontology represents knowledge about domain concepts such as collaborations, collaborators and datasets, and the collaborations are mediated by using the ontology as a shared view that each collaborator can modify. The software supports data sharing, privacy constraints and visualisations, serving as a proof of concept for the ontology-based approach to collaboration. Future work includes conducting user tests and expanding the data analytics and privacy preservation functionality of the software.

References

1. Barhamgi, M., Perera, C., Ghedira, C., Benslimane, D.: User-centric privacy engineering for the Internet of Things. IEEE Cloud Computing **5**(5), 47–57 (2018)
2. Diamantini, C., Potena, D., Storti, E.: KDDONTO: An ontology for discovery and composition of KDD algorithms. In: Third Generation Data Mining: Towards Service-Oriented Knowledge Discovery (SoKD'09). pp. 13–24 (2009)

3. Diamantini, C., Potena, D., Storti, E.: A semantic-aided designer for knowledge discovery. In: 2011 International Conference on Collaboration Technologies and Systems (CTS). pp. 86–93 (2011)
4. Diamantini, C., Potena, D., Storti, E.: Semantically-supported team building in a KDD virtual environment. In: 2012 International Conference on Collaboration Technologies and Systems (CTS). pp. 45–52 (2012)
5. Diamantini, C., Potena, D., Storti, E.: Collaborative management of a repository of KDD processes. International Journal of Metadata, Semantics and Ontologies **9**(4), 299–311 (2014)
6. Gharib, M., Giorgini, P., Mylopoulos, J.: Towards an ontology for privacy requirements via a systematic literaturel review. In: International Conference on Conceptual Modeling. pp. 193–208 (2017)
7. Ghorbel, A., Ghorbel, M., Jmaiel, M.: PRIARMOR: An IaaS solution for low-level privacy enforcement in the cloud. In: 2017 IEEE 26th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE). pp. 119–124 (2017)
8. Hartmann, S., Ma, H., Vechsamutvaree, P.: Providing ontology-based privacy-aware data access through web services. In: International Conference on Conceptual Modeling. pp. 74–85 (2015)
9. Kandogan, E., Roth, M., Schwarz, P., Hui, J., Terrizzano, I., Christodoulakis, C., Miller, R.J.: LabBook: Metadata-driven social collaborative data analysis. In: 2015 IEEE International Conference on Big Data (Big Data). pp. 431–440 (2015)
10. Keet, C.M., Ławrynowicz, A., d'Amato, C., Kalousis, A., Nguyen, P., Palma, R., Stevens, R., Hilario, M.: The Data Mining OPtimization Ontology. Web Semantics: Science, Services and Agents on the World Wide Web **32**, 43–53 (2015)
11. Kietz, J.U., Serban, F., Fischer, S., Bernstein, A.: "Semantics inside !" But lets not tell the data miners: intelligent support for data mining. In: European Semantic Web Conference (ESWC). pp. 706–720 (2014)
12. Kumara, B.T.G.S., Paik, I., Zhang, J., Siriweera, T.H.A.S., Koswatte, K.R.C.: Ontology-based workflow generation for intelligent big data analytics. In: 2015 IEEE International Conference on Web Services. pp. 495–502 (2015)
13. Musen, M.A.: The Protégé project: A look back and a look forward. AI Matters **1**(4), 4–12 (2015)
14. Panov, P., Soldatova, L., Džeroski, S.: OntoDM-KDD: Ontology for representing the knowledge discovery process. In: Discovery Science. pp. 126–140 (2013)
15. Panov, P., Soldatova, L., Džeroski, S.: Ontology of core data mining entities. Data Mining and Knowledge Discovery **28**(5), 1222–1265 (2014)
16. Panov, P., Soldatova, L.N., Džeroski, S.: Generic ontology of datatypes. Information Sciences **329**, 900–920 (2016)
17. Tuovinen, L., Smeaton, A.F.: Ontology-based negotiation and enforcement of privacy constraints in collaborative knowledge discovery. Presentation at the 2nd International Workshop on Personal Analytics and Privacy (PAP 2018), http://kdd.di.unito.it/pap2018/papers/PAP_2018_paper_2.pdf, accessed 13 May, 2019
18. Tuovinen, L., Smeaton, A.F.: Unlocking the black box of wearable intelligence: Ethical considerations and social impact. In: 2019 IEEE Congress on Evolutionary Computation (2019)
19. Žáková, M., Křemen, P., Železný, F., Lavrač, N.: Automating knowledge discovery workflow composition through ontology-based planning. IEEE Transactions on Automation Science and Engineering **8**(2), 253–264 (2011)