Ubiquitous Data

Andreas Hotho[1], Rasmus Ulslev Pedersen[2], and Michael Wurst[3,⋆]

[1] University Kassel
Depart. of Electrical Engineering/Computer Science
Knowledge and Data Engineering Group
[2] Copenhagen Business School,
Dept. of Informatics, Embedded Software Lab,
Copenhagen, Denmark
[3] Technical University Dortmund, Computer Science LS8
44221 Dortmund, Germany
`hotho@cs.uni-kassel.de, rup.inf@cbs.dk, michael.wurst@uni-dortmund.de`

Ubiquitous knowledge discovery systems must be captured from many different perspectives. In earlier chapters, aspects like machine learning, underlying network technologies etc. were described. An essential component, which we shall discuss now, is still missing: *Ubiquitous Data.* While data themselves are a central part of the knowledge discovery process, in a ubiquitous setting new challenges arise. In this context, the emergence of data itself plays a large role, therefore we label this part of *KDubiq* systems *ubiquitous data.* It clarifies the KDubiq challenges related to the multitude of available data and what we must do before we can tap into this rich information source.

First, we discuss key characteristics of ubiquitous data. Then we provide selected application cases which may seem distant at first, but after further analysis display a set of clear commonalities. The first example comes from Web 2.0 and includes network mining and social networks. Later, we look at sensor networks and wireless sensor networks in particular. These examples provide a broad view of the types of ubiquitous data that exist. They also emphasize the difficult nature of ubiquitous data from an analysis/knowledge discovery point of view, such as overlapping or contradicting data. Finally, we provide a vision how to cope with current and future challenges of ubiquitous data in KDubiq.

1 Characteristics of Ubiquitous Data

Data mining can be applied to many different types of data (cf. [1]). These data types range from simple atomic types to more complex constructs such as time series, relational data, graphs, images and many others. Over the years, many methods were developed to transform and incorporate these different data types in the overall data mining process.

One aspect that has been studied only briefly so far is ubiquity. In many current applications, data emerges in an ubiquitous way rather than in cooperated data bases. Orthogonal to the actual type of data, this adds new facets to the

[⋆] With contributions from N. Korfiatis, C. Körner, C. Lauth, C. Schmitz.

M. May and L. Saitta (Eds.): Ubiquitous Knowledge Discovery, LNAI 6202, pp. 61–74, 2010.

data that is collected. We will refer to such data as *ubiquitous data* and to the corresponding augmented data types as *ubiquitous data types.*

Definition of Ubiquitous Data. We can define ubiquitous data as such data that emerges in an asynchronous, decentralized way from many different, loosely-coupled, partially overlapping, possibly contradicting sources.

A first characteristic of such ubiquitous scenarios is that data is produced asynchronously in a highly decentralized way. Collecting such data in a central data warehouse can be prohibitively expensive and even impossible in some applications (if no fast and reliable network connection is available). This is even more challenging if data emerges rather quickly. Often there is no data repository, growing over time, but rather a stream of data (see Chapter 3). Therefore, ubiquitous data measurements are usually annotated with a timestamp, as information about time is essential in the subsequent data mining process. Furthermore, most ubiquitous data has an element of location associated. This enables the so-called space-time trajectories, which have recently become a topic of significant research, e.g. in the GeoPKDD project [2].

Another aspect of ubiquitous scenarios is that usually many different data types are involved that make up a whole. For a user of the popular MySpace platform, for instance, it is very natural to deal with a combination of website, pictures, audio tracks, forums and other information. In sensor networks, different sensors may measure image data, sound, video, temperature, light, acceleration, etc. to obtain a complete picture of a situation. From a data mining point of view, this situation leads to highly complex preprocessing and data integration issues (involving, e.g., sensor fusion or ontology mapping). Furthermore, many of the data types involved in such scenarios have a complex inner structure, e.g. blog networks or cascaded sensors.

A third characteristic feature of ubiquitous data mining scenarios is the fact that data in almost all cases emerges from a very high number of partially overlapping, loosely connected sources. An example are social bookmarking systems. Users are allowed to annotate resources with arbitrary textual expressions, called tags. Users here represent independent data sources that may be in different relation to each other. Tags assigned by different users can complement each other, they can be contradictory, they can be redundant, etc. Processing data from many different sources is significantly more challenging than processing data from a single source or a small number of sources, because contradictions and overlaps cannot be resolved manually, like in data warehouse systems. A common characteristic of ubiquitous data types is thus, that each data point or measurement is additionally annotated with a data source and that this annotation plays a central role in the data mining process.

We will exemplify the implications of ubiquitous data types on two different areas: data mining in Web 2.0 environments and in distributed sensor networks.

Web 2.0 denotes a shift in the way the Internet is used. Instead of a small number of users that produces all content, now a very large number of users actively contributes to extend the internet. These contributions range from blogs, social bookmarking, videos, pictures, articles in discussion, ratings, reviews, and many others. Applying data mining to this data is, on the one hand, very appealing, as it helps to consolidate this information and to make it maximally useful. On the other hand it is very challenging as different data types are involved, data is produced by many different users distributed all over the world and is often noisy and contradicting, partially overlapping etc. While this kind of data can be stored centrally, it emerges in a distributed way. Because of its huge amount of data, consolidating and cleaning this data manually is not a reasonable option. It thus falls under the discussed ubiquitous paradigm.

Belonging to a seemingly completely different area are sensor networks. In sensor networks, a large number of distributed, partially connected sensors are used to measure some properties in a given domain. Measurements of the different sensors may overlap (neighboring sensors) or contradict (if a sensor does not work properly). Sensors can also be connected in networks of cascading sensors. Finally, sensors can capture many different data types, such as images, video, etc. A good illustration is the popular sensor network systems is TinyOS. On the website (`www.tinyos.net`) there are numerous examples of real applications: mobile agents, magnetic measurements, and robotic grids just to name a few.

On a second sight, sensor networks and Web 2.0 data is very similar, as both expose the typical characteristics of ubiquitous data. In both cases, data points are annotated with a source (the user or the sensor) that plays an important role in the data mining process or with some kind of timestamp to describe the history of the data. In both areas we face different data types that represent the same entities from different perspectives (e.g. MySpace profiles and different sensors in a mobile robot). Also, data emerges asynchronously in a distributed way in both areas and there is usually no way of collecting this data in a single data base.

Research in both areas has developed almost completely independently, despite these strong connections. In the following, we therefore present first challenges of ubiquitous user-centered data types with focus on Web 2.0 mining, and then research on data mining in distributed sensor networks. Finally, we discuss the common vision of both, by analyzing which challenges they have in common and which methods would be mutually applicable.

2 Web 2.0

2.1 Emerging Data from Distributed Sources

Ubiquitous data can be processed in different ways in the context of Web 2.0 applications. On the one hand, data and annotations provided by users may emerge in a distributed way and be also stored locally. Famous examples are P2P networks [3,4] or distributed agent based systems [5]. On the other hand, there are many applications in which data emerges in a distributed way, but is then collected and stored at a central server like in social bookmarking systems [6,7,8] Providing content in a Web 2.0 like way is applicable to a wide range of resources and data types, such as web pages, images, multimedia, etc. This popular principle is currently applied to many domains.

The data with which we have to deal in these applications is quite diverse. On the one hand, we have textual data, which might be partially structured (as in Wikipedia entries) or which may consist of short snippets only, such as in social bookmarking systems. Other applications, as for instance audio, image or video sharing platforms, are concerned with multimedia data.

All these applications have in common that data emerges in a ubiquitous way from many independent sources (users). This leads to similar challenges in all these applications connected to the basic characteristics of ubiquitous data, such as contradictions, overlaps and heterogeneity. Tags in social bookmarking systems may be contradicting, for instance. Also, there are many duplicate images and videos on popular multi-media sharing platforms.

The current hype around Web 2.0 applications contributes to several important challenges for future data and web mining methods. Such challenges include the analysis of loosely-coupled snippets of information, such as overlapping tag structures, homonym or synonym tags, blog networks, multimedia content, different data types etc. Other challenges arise from scalability issues or new forms of fraud and spam. Often, a more structured representation, which allows for more interaction, is needed. Mining could help here to bridge the gap between the weak knowledge representation of the Web 2.0 and the semantic web by extracting hidden patterns from the data.

In the following, we will show how the described Web 2.0 data relates with more structured data and how data mining can be applied to these data types. We will finish this section with applications showing the emergence of the ubiquitous data.

2.2 Semantic Web and Web 2.0

Web 2.0 applications often use very weak meta data representation mechanisms that reflect the distributed, loosely-coupled, nature of the underlying processes. The most basic meta data annotation is a tag, thus an arbitrary textual description that users can assign to any resource. Tags may denote any kind of information and are, by default, not structured in any way. There are several approaches to allow users to express slightly more information while tagging. Such approaches allow them to create hierarchies of tags or groups of tags [8] (sometimes referred to as aspects [10]). Tags are limited to express only very simple information. However, the same principle allowing users to create their own conceptualization in a decentralized way, is also applied to more complex data. Structured tag representations, such as XML microformats, created by users for various special applications, are a good example for this kind of knowledge representation.

All of these efforts can be denoted as local. They represent the requirements and views of individual users or of user groups. An extreme case of locality would be a user that applies tags that are private in the sense that they are not shared by any other user. Local approaches complement global approaches, such as standardized XML formats or the Semantic Web, in a way that is better applicable in highly distributed scenarios.

The Semantic Web is based on a vision of Tim Berners-Lee, the inventor of the WWW. The great success of the current WWW leads to a new challenge: a huge amount of data is interpretable by humans only; machine support is limited. Berners-Lee suggests to enrich the Web by machine-processable information which supports the user in his tasks. For instance, today's search engines are already quite powerful, but still too often return excessively large or inadequate lists of hits. Machine-processable information can point the search engine to the relevant pages and can thus improve both, precision and recall. In such global approaches, all users must obey a common scheme. For instance, authors of scientific publications usually have to classify their work according to a static classification scheme of topics provided by the publisher.

A key approach is to use data mining techniques that mediate between the different levels of meta data [10] and to use semantic web technology to store and transport the knowledge. This can be achieved, for instance, by identifying patterns such as combinations of tags that co-occur often [11], by analyzing, identifying and using the emergence structure and inherent semantics of web 2.0 systems [12,13,14] or by extracting information from natural language texts by information extraction. Then a strong knowledge representation will represent the data appropriately, can help to transfer distributed learned knowledge and act in this way as a mediator between any kind of learning system.

2.3 Peer-to-Peer Based Web 2.0 Applications

Despite of its conceptual and architectural flaws, many current Web 2.0 applications store their data centrally. This is not necessarily so. There are several approaches to combine Web 2.0 tagging or Semantic Web (SW) with P2P technology. General characteristics of P2P systems and their relevance for ubiquitous knowledge discovery have been discussed in Chapter 2, Section 1. In this section we focus on the relation between P2P, Web 2.0 and the Semantic Web.

In [3] P2P is defined in the following way:

> "Peer-to-peer is a class of applications that takes advantage of resources—storage, cycles, content, human presence—available on the edge of the Internet."

This definition combines two aspects. First, data is created by users, which act as a resource by annotating data, locations, etc. On the other hand, these users are often located at the edge of the internet, communicating only with mobile phones or handheld devices. This fully distributed nature of not only data emergence but also data storage has a strong influence on the way data is represented and processed.

The combination of Semantic Web and P2P technology, for instance, opens up a feasible way of combining rich knowledge representation formalisms, on the one hand, with low overhead but with immediate benefit, on the other hand. Another possible application of semantic P2P networks, that has been researched in projects such as Nepomuk[1], is the *Social Semantic Desktop.* Every user of desktop applications is faced with the problem that the office documents, emails, bookmarks, contacts, and many other pieces of information on their computers are being processed by isolated applications which do not maintain the semantic connections and metadata that apply to the resources. Semantic desktop applications offer approaches to deal with this problem of distributed storage based on semantic integration.

Connecting semantic desktops by a P2P network is an idea put forward by [15]. The result is called *Networked Semantic Desktop* or *Social Semantic Desktop* (the latter name being the one most commonly used today). This idea has received a lot of attention lately and has spawned a successful series of workshops [16,17]. Data mining is a very natural extension to make semantic desktop systems more flexible an adaptive. The corresponding methods must, just as for text mining, respect the local context in which the information emerged. They require, therefore, ubiquitous data mining methods almost by definition.

An example of an approach that combines social bookmarking systems and P2P technology is the Nemoz system [10]. Nemoz (see Chapter 1, Section 6.2) is a distributed media organization framework that focuses on the application of data mining in P2P networks.

Media collections are usually inherently distributed. Multimedia data is stored and managed on a large variety of different devices with very different capabilities concerning network connection and computational power. Such devices range from high performance work stations to cell phones. This demands for sophisticated methods to distribute the work load dynamically including the preparation of the data.

The key point in processing multi media data is an adequate data representation. Finding and extracting adequate features is computationally very demanding and therefore not simply applicable in an interactive end user system. The key to solve this problem is twofold. First, by distributing the work load among nodes with different computational capabilities, a much higher efficiency can be achieved, which allows the application of data mining even on devices with only little resources. Second, it is possible to exploit the fact that although all local tag structures created by the users may differ in any possible way, it can be assumed that at least some of them resemble each other to some extent, e.g. many users arrange their music according to genre. Thus by collaborating in the Data Mining process, each learner can profit from the work done by other nodes [18].

Network Mining. Mining networks is of particular interest to ubiquitous data mining, as most data addressed in this book forms some kind of graph or network. One research area, which has addressed the analysis of graphs comparable to the one discussed in this chapter, is the area of “Social Network Analysis”. Therefore, methods developed in that area are a good starting point to develop and analyze

[1] `http://nepomuk.semanticdesktop.org/`

other algorithms for mining ubiquitous data. We will shortly recall the ideas behind (social) network mining.

A (social) network (G) is a graph that consists of a set of nodes (V) and a set of edges (E) such as: $G := (V, E)$ where V is the set of network actors (e.g. people or institutions) and E is the set of relations between them. Most Web 2.0 systems can be represented as such a graph. Social networks can be explicit, such as in popular applications as LinkIn. They often, however, emerge rather implicitly by users communicating or collaborating via mail, instant messaging or mobile phones. This distributed emergence of social networks through communication patterns is an important challenge for ubiquitous data mining methods. Methods developed in this area are especially important as they provide valuable insights into the relationship among Web 2.0 participants.

Social network mining tackles the extraction of social network data from sources that contain recordings of interactions between social entities (e.g. emails). Social network data describes structural relations between social entities, such as blog authors or public profiles of users. Social network data consists of two types of variables: structural and compositional. *Compositional variables* represent the attributes of the entities that form the social network. That kind of data can be, for instance, the topic of the webpage (e.g. portal, homepage) or things related to the profile of the social entity they describe (e.g. age, address, profession, etc). *Structural variables* define the ties between the network entities (E) and the mode under which the network is formed. The term mode refers to the number of entities which the structural variables address in the network.

Current social network mining methods process data usually in a centralized fashion. This will soon no longer be possible. With the emergence of massive amounts of implicit social networking data, such as phone data or email data, the analysis of this data can not be performed in a centralized way anymore. Rather, it will be important to find highly distributed approaches such that communities or other strongly connected components in the social graph are identified locally, bottom-up, instead of doing this centrally, top-down. Research on such methods will certainly play an essential role for the scalability of future social networking systems.

3 Sensor Networks

Today, only few people can participate in everyday life without leaving traces of their actions. The mass usage of mobile phones, GPS traveling assistance and the spreading application of radio frequency identification (RFID) technology to pursue the lifetime of goods are only the most prominent examples of how collections of data appear in everyday life. In addition, the costs of sensor nodes are constantly decreasing, which makes their future application very attractive and provides for many large-scale sources of ubiquitous data. However, the knowledge extraction from ubiquitous data sources and careful handling of private data are only in their infancy (for privacy, see the discussion in Chapter 5).

Ubiquitous systems come in different forms as well. Some ubiquitous systems are web-based, others are rooted in the physical surroundings. Sensor networks belong to the latter category. In the following, we will introduce sensor networks and discuss selected aspects related to the ubiquitous knowledge discovery process. At all times these systems are subject to severe resource constraints (see Chapter 3, Section 2. Our discussion will focus on data aspects. Distributed computing aspects of wireless sensor networks have been discussed in Chapter 2, Section 3.

3.1 Sensors as Data Sources

Just as humans possess senses, there are many different kinds of sensors to form a sensor network. Basic sensors record temperature, light or sound and form fundamental parts in today's industrial applications. New types of sensing emerged in the areas of health services and environmental monitoring. Think, for example, of nearly invisible devices to monitor heart beats or to measure diverse chemical substances, acid levels or carbon dioxide in natural surroundings. The data sources record phenomenons that spread in space. It is also possible to record characteristics of the landscape itself as in satellite recordings and to monitor the spatial location of objects. The latter has become very popular with the emergence of mobile phones, GPS and RFID.

Sensors. There has been significant work by IEEE to provide information regarding the output of sensors. This is a pre-requisite for getting access and using the information in an ubiquitous knowledge discovery system. Sensors with these capabilities can be referred to as smart transducers. This is also described in the IEEE 1451 (proposed) standards.

Sensors may provide information according to the specifications in the IEEE 1451 data sheets. The electrical connections of this system are defined for 2, 3, or 4 wires. It works by a template that provides semantic information regarding the data from the sensor.

3.2 Sensor Network Modeling Languages

In wireless sensor networks (see Chapter 2, Section 3) there has been some work on a sensor model language called sensorML. It uses XML to encode and describe sensor models. The whole process ranging from input to output and associated parameters can be described using this language. It is part of the standards proposed by the Open Geospatial Consortium.

SensorML. SensorML is a XML schema-based markup language to describe sensor systems. This language can describe both stationary and dynamic sensors such as robotic sensor systems. Furthermore, it can describe sensors placed inside the monitored object as well as sensors placed remote with respect to the object. Processes describe which input they measure, which output they produce. In addition the method used to produce the output can be described. One example could be a 2-dimensional input sampled at 1 Hz to produce one classification output with a binary support vector machine. In wireless sensor networks, data types are often real values, representing specific physical phenomena. This can include GPS sensors, mobile heart monitors, web cams, satellite-borne earth images. It is already clear from this wide range of wireless sensor networks data that any ubiquitous knowledge discovery agent needs meta information regarding the sensor data. It can be challenging to provide this information in situ. Therefore, an external description of a sensor and the services it can provide is necessary. SensorML is one such approach.

3.3 Wireless Sensor Network Applications

TinyOS is a popular sensor network platform. Other sensor network platforms are starting to emerge such as the Java-based Sun SPOT from Sun Microsystems. The list of example applications for wireless sensor networks is long. However, data management in these distributed loosely coupled systems are difficult. Later in this section we will discuss TinyDB [19] as example abstraction of this problem. Furthermore, a company named Sentilla has taken an interesting approach to Java object serialization which we shall also touch upon. Next we look at some general examples which are all related to ubiquitous data in some way.

We find examples like mobile agents, magnetic measurements, robotic grids, pre- and in-hospital monitoring, volcanic eruption detection, hog monitoring, distributed location determination, structural health monitoring, rescue equipment, neural wireless interface, oil/gas exploitation, and elite athlete monitoring. There are some performance challenges described in Chapter 7, Section 7, which add a systems perspective to some of these applications.

Recently, a new experimental platform for sensor networks using LEGO MINDSTORMS NXT has been introduced by Pedersen [20]. It runs the small wireless operating system TinyOS, while providing access to many different sensors on this popular educational platform.

The traditional message abstraction in TinyOS is a message structure, which is similar to a `struct` in the C programming language. This approach is fine for the application domain because the individual sensors know the context of the data. Now we can look at two more general approaches.

In the context of TinyOS, which was just introduced, there is an data abstraction called TinyDB. In TinyDB, the data in the distributed sensor network are stored within the network nodes (called motes in TinyOS terms). What is interesting is that the data can be retrieved using SQL language which is familiar to many people. That is one example from the TinyOS area. Java-based sensor networks are also emerging and a company like Sentilla presented their approach to data management at the JavaOne 2008 conference. They acknowledge that the packet payload in a 802.15.4 network is so small that traditional Java object serialization is not feasible. Instead they send raw packets of data and let a gateway mote server translated the raw data to Java objects again. In that way the data can be used like traditional Java objects in the client application. To summarize, we have shown TinyDB as one SQL-based abstraction of ubiquitous data and we have shown another abstraction of ubiquitous data in terms of Java objects. Common to both solutions are that they abstract the access to raw ubiquitous data.

3.4 Object Monitoring in Space and Time

Trajectory data will become a highly important data type in the near future. An example for the monitoring in space and time using GPS trajectories has been given in Chapter 1, Section 6.1, in the activity recognition example. A trajectory is the path of a moving object within time and space. It can be viewed as a function mapping each moment in time to the location of a given object. In practice, trajectories are represented as (finite) number of pairs $(time, location)$ where both time and space have been discretized [21]. Depending on the measurement technology, recordings occur at regular or irregular time intervals. The location can either be expressed within a coordinate system, e.g. latitude / longitude, or symbolic coordinates, as for example cell identifiers of mobile phone data.

GPS Data. GPS technology records the x,y-position of an object every second and thus provides a quite accurate picture about the object's position and motion. If GPS signals are used without other assisting technologies, a standard error of 10 - 25 m or 5 - 10 m for newer devices is made during location approximation. However, GPS signals are blocked by buildings, and measurements indoors, inside of tunnels or in alleys between tall buildings may be lost.

Mobile Phone Data. In cellular networks, mobile phones perform a location update procedure in regular or irregular intervals to communicate their current location area. In addition, cell identifiers are recorded during the time of a call. It is therefore possible to derive movement information or call densities from logging data of mobile phone networks. The data, however, possesses a coarse level of granularity. Cell sizes of GSM networks range between 0.1 and 1 km in the city centre but can extend as far as 30 km in rural areas. An additional uncertainty is introduced by overlapping cells. Depending on the signal strength and workload of a base station antenna, a handover to another cell may occur although the user does not change his or her position.

Both sources for trajectory data have their strengths and weaknesses. While GPS trajectories are very accurate in location, they may contain gaps and the number of available trajectories is usually small. In comparison, mobile phone data have a low resolution but are produced in great masses. One challenge lies in the combination of both data sources and mutual exploitation of advantages.

4 Common Characteristics of Data from Web 2.0 and Sensor Networks

In both areas, Web 2.0 environments and sensor networks, we face ubiquitous data and ubiquitous data types. Ubiquitous data types augment traditional data types like texts, photos, videos, bookmarks (in the case of Web 2.0) or data types like temperature, sound and light (in the case of sensor networks) with several novel aspects: a source identifier (user id or sensor id), a timestamp (either logical or real time) possibly a location or more specialized features depending on the scenario. On the one hand, this information is very valuable and provides new possibilities for the application of data mining. On the other hand, it leads to a set of challenges in both areas. The massive amount of different sources and the necessity to preprocess data locally,"in context" goes beyond the capabilities of current data mining systems. In the following, we will address the common characteristics of both application areas and involved data types.

A first important common challenge is the large number of heterogeneous sources in both application areas. In Web 2.0 applications, we have a large user base, providing different kinds of meta data, that partially overlap, contradict or complement each other. In sensor networks, we have a number of sensors (sometimes referred to as smart dust) that measure partially overlapping entities. Sensor data maybe contradictory as well, e.g., if a sensor is faulty. In both cases, there is a need to aggregate measurements and annotations into a smaller, concise set of information. For Web 2.0 applications, this can be approached by applying advanced clustering methods. In sensor networks, the data are combined using sensor fusion. There is an additional challenge as the data streams cannot be cached on the sensor nodes, so the sensor fusion has to be done in real time. Existing methods for both, tag aggregation and sensor fusion, are not fully satisfying. For instance, many graph mining methods are not directly applicable to the hypergraph structure of folksonomies [14] that emerge from the augmentation of annotations with a user id or a timestamp. The same holds for the multi-mode structure of blog networks and wiki page editing networks. Common to both scenarios is that the distributed computers/sensors try to collaborate. It could have been the other way around such that each computing unit needed to be aware of malicious information. In this sense we have a *cooperative* scenario.

Another problem common to data that emerges in a loosely coupled, distributed way is that object references are hard to define globally. One solution for the web is to use the URI as identifier. For sensor networks, RFID provides a possible solution. Both solutions are not able to cope with all challenges that arise from this, such as duplication detection problems, missing and wrong links and information. This topic will require substantial research in the next years.

In both areas, sources are often structured. In blog networks, individual blogs can be mash-ups of other blogs. In sensor networks, sensor nodes are often explicitly associated. This association is a sort of meta tagging of the data. Often the nodes can be localized with respect to one another. Using many sensors results in higher reliability, but also in redundancy. There are currently few methods that take the interrelation of data sources into account.

In both areas, problems can only be tackled by applying highly distributed processing and intelligence. In Web 2.0 applications, these are intelligent clients performing all kinds of support for the user (mostly Ajax based). In sensor networks, distributed units performs some kind of autonomous computing to deal with the large number of measurements under restricted resources. The resource constraints can be addressed with local computing that can save on the radio use. This can extend the deployment of the sensor network.

Data and meta data is of essential importance for knowledge discovery. A study of ubiquitous data in the two area of interest reveals many common characteristics and challenges. In our opinion, several methods of distributed processing and intelligence are equally applicable in both areas. In particular, with an increasing amount of data that emerges in social applications, there is an increasing need to process this data locally. Several methods, as highly distributed sensor fusion or aggregation, could be applicable in this domain. This would enable future Web 2.0 applications to process data in place and in context, increasing their scalability and the soundness of results.

To achieve the benefits of such synergy, researchers in both areas have to collaborate closely to make ubiquitous knowledge discovery an everyday experience.

5 Emerging Challenges and Future Issues

There are a lot of open issues which need to be solved to make different kinds of ubiquitous data useful by applying knowledge discovery algorithms to it. In this chapter we have discussed two example domains: data from Web 2.0 applications and sensor networks. There are several issues still open. First, there is the need for a coherent implementation-independent description of ubiquitous data including more semantics. Second, there is a need to describe the dynamic nature of ubiquitous data and its refinement over time. Third, there is a need for patterns and best practices of how to implement ubiquitous data descriptors in resource constrained environments.

In terms of data description, we find that there are several existing approaches to these problems in both Web 2.0 and in sensor networks. These are the two dominant examples we have used in this chapter on ubiquitous data. To make the data available for various data mining and knowledge discovery tasks we need to first describe the data. One step to get an independent description is to make use of semantic web technology and combine this with the advantages of Web 2.0. This is often referred to as Web 3.0. Semantic web approaches provide also nice solutions to describe sensor data in an abstract way. A more abstract and machine readable description of data allows for a better use of data. But to make this vision reality in all ubiquitous environments, one has to solve problems resulting from limited resources, limited bandwidth or missing and contradicting data.

The second dimension of ubiquitous data description is the dynamic nature of the data and the need to describe updates to the data. Today, most of the data descriptions (like XML for example) is targeted toward static data. However, data changes rapidly in ubiquitous environments, and furthermore we may use techniques like machine learning on streaming data to refine the information content of the data on the fly. This dynamic nature of the system is not reflected in current approaches.

The third part of the vision is to provide practical implementations. We discussed earlier in this chapter two different solutions in sensor networks for an SQL abstraction and a Java abstraction which were mapped to the constraints of the sensor network. As of today, there is not any standard solution to this problem. Semantic Web technology provides a principle solution to store such data in a reusable way but does not provide implementations respecting the resources constraints of ubiquitous devices.

We need a systematic way to design our data representation in terms of constraints such as system related constraints (CPU, memory, battery limitations etc.), data mining related constraints (privacy, dynamic data, etc.), and other constraints (domain specific such as data quality constraints in social network mining).

With these three main dimensions of ubiquitous data representation, we can look at the next steps in the development and application of knowledge discovery algorithm to ubiquitous data. These steps are centered around the combination of Web 2.0, semantic web, and the physical world of sensors including all restrictions of small devices which can be seen as the next step of the evolution of Web. As this will be, however, a long way to go, we may label this vision *Web 4.0*.

References

1. Hand, D., Mannila, H., Smyth, P.: Principles of Data Mining. MIT Press, Cambridge (2001)
2. Giannotti, F., Pedreschi, D. (eds.): Mobility, privacy, and geography: a knowledge discovery perspective. Springer, Heidelberg (2008)
3. Shirky, C.: Listening to Napster. In: [4], pp. 21–37
4. Oram, A. (ed.): Peer-to-Peer. O'Reilly, Sebastopol (2001)
5. Weiß, G. (ed.): Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. MIT Press, Cambridge (1999)
6. Golder, S., Huberman, B.A.: The structure of collaborative tagging systems (2005)
7. Hammond, T., Hannay, T., Lund, B., Scott, J.: Social Bookmarking Tools (I). D-Lib Magazine (2005)
8. Jäschke, R., Hotho, A., Schmitz, C., Stumme, G.: Analysis of the publication sharing behaviour in BibSonomy. In: Priss, U., Polovina, S., Hill, R. (eds.) ICCS 2007. LNCS (LNAI), vol. 4604, pp. 283–295. Springer, Heidelberg (2007)
9. Flasch, O., Kaspari, A., Morik, K., Wurst, M.: Aspect-based tagging for collaborative media organisation. In: Proceedings of the ECML/PKDD Workshop on Ubiquitous Knowledge Discovery for Users (2006)
10. Stumme, G., Hotho, A., Berendt, B.: Semantic web mining - state of the art and future directions. Journal of Web Semantics 4(2), 124–143 (2006)
11. Schmitz, C., Hotho, A., Jschke, R., Stumme, G.: Mining association rules in folksonomies. In: Batagelj, V., Bock, H.H., Ferligoj, A., Ziberna, A. (eds.) Data Science and Classification. Proceedings of the 10th IFCS Conf. Studies in Classification, Data Analysis and Knowledge Organization, pp. 261–270. Springer, Heidelberg (2006)
12. Mika, P.: Ontologies are us: A unified model of social networks and semantics. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) ISWC 2005. LNCS, vol. 3729, pp. 522–536. Springer, Heidelberg (2005)

13. Hotho, A., Jäschke, R., Schmitz, C., Stumme, G.: Information retrieval in folksonomies: Search and ranking. In: Sure, Y., Domingue, J. (eds.) ESWC 2006. LNCS, vol. 4011, pp. 411–426. Springer, Heidelberg (2006)
14. Cattuto, C., Schmitz, C., Baldassarri, A., Servedio, V.D.P., Loreto, V., Hotho, A., Grahl, M., Stumme, G.: Network properties of folksonomies. AI Communications 20(4), 245–262 (2007)
15. Decker, S., Frank, M.R.: The Networked Semantic Desktop. In: Proc. WWW Workshop on Application Design, Development and Implementation Issues in the Semantic Web, New York (2004)
16. Decker, S., Park, J., Quan, D., Sauermann, L. (eds.): The Semantic Desktop - Next Generation Information Management & Collaboration Infrastructure. Proc. of Semantic Desktop Workshop at the ISWC 2005, CEUR Workshop Proceedings, vol. 175 (2005), ISSN: 1613–0073
17. Decker, S., Park, J., Sauermann, L., Auer, S., Handschuh, S. (eds.): Proceedings of the Semantic Desktop and Social Semantic Collaboration Workshop (SemDesk 2006) at the ISWC 2006, Proceedings of the Semantic Desktop and Social Semantic Collaboration Workshop (SemDesk 2006) at the ISWC 2006. CEUR-WS, vol. 202 (2006)
18. Wurst, M., Morik, K.: Distributed feature extraction in a p2p setting - a case study. Future Generation Computer Systems, Special Issue on Data Mining (2006)
19. Madden, S.R., Franklin, M.J., Hellerstein, J.M., Hong, W.: Tinydb: an acquisitional query processing system for sensor networks. ACM Trans. Database Syst. 30(1), 122–173 (2005)
20. Pedersen, R.U.: Tinyos education with lego mindstorms nxt. In: Gama, J., Gaber, M.M. (eds.) Learning from Data Streams. Processing Techniques in Sensor Networks, pp. 231–241. Springer, Heidelberg (2007)
21. Andrienko, N., Andrienko, A., Pelekis, N., Spaccapietra, S.: Basic concepts of movement data. In: Mobility, Privacy and Geography: a Knowledge Discovery Perspective. Springer, Heidelberg (2008)