Context Management in Health Care Apps

Albert Pla*
Universitat de Girona
Girona, Spain
albert.pla@udg.edu

Beatriz López
Universitat de Girona
Girona, Spain
beatriz.lopez@udg.edu

Natalia Mordvaniuk
Universitat de Girona
Girona, Spain
natalia.mordvaniuk@udg.edu

Cristina Armero
Institut d'Investigacio
Biomèdica de Girona
Girona, Spain
carmero@idibgi.org

Abel López-Bermejo
Institut d'Investigacio
Biomèdica de Girona
Girona, Spain
carmero@idibgi.org

ABSTRACT

Context information has been proved to enhance user's experience in mobile apps. In this paper we analyze the management of such information in health care apps, paying special attention to health recommenders and health monitoring applications. The paper first describes the kind of context which can be included in healthcare apps (geographical and temporal, environmental, and source related). Then, it discusses how this information can be handled in order to improve the outputs of the applications and their reasoning modules; to illustrate that we describe how context can be integrated into a well-known reasoning methodology such as knowledge-based reasoning. As a result, we describe how context is handled in an app for remote premature-baby monitoring.

Keywords

Context-reasoning, Health care apps, Decision support systems, Remote assistance

1. INTRODUCTION

Context-awareness claims that a particular set of information can have a different meaning or significance depending on the context where it is placed or gathered. The medical domain is no exception to such contention; for instance, patients can obtain different blood pressure measures depending on whether the pressure test has been taken by a health professional or not (white coat effect [21]). Similarly, having a high heart beat frequency when laying at home has different implications than having it when performing intensive physical activity. Therefore, keeping track of contextual data arises as a useful instrument for medical applications.

Mobile devices provide a new way of gathering context data to improve health application outcomes. Particularly, they enable to automatically gather information regarding the place and the time the patient is located when using the application. This allows identifying useful information like if the patient is using an application at home, at the hospital or in a different environment. In addition, the wide range of sensors that smart-phones are equipped with, combined to their access to the cloud allow to obtain other relevant contextual information such as environmental conditions, weather information, etc. Furthermore, smartphones can support external devices (e.g. glucometers or pulse oximeters) that provide health measurements but, in turn, introduce new contextual information that can affect such measurements (e.g. some sensor devices might be more reliable than others or might have different calibration parameters).

The understanding and proper management of context is directly related to the personalization of the apps to the individual patient needs. To this end, in this work we review the context data required in health apps (including the new information that should be incorporated due to the use of mobile devices) and we analyze how it can be integrated into commonly used reasoning techniques, illustrating such integration by means of knowledge-based reasoning (CBR). The review proposed in this paper is then used to improve the outputs of a health app that helps physicians and caregivers to remotely monitor the development of premature-born babies.

2. CONTEXT MANAGEMENT IN HEALTHCARE APPS

In the health care domain, mobile apps can be used to inform physicians regarding the patient status whilst providing recommendations to patients in order to improve their recovery process and to improve the patient's well-being. Such recommendations are based on smart components (decision support systems) that might take advantage of the mobile device to improve their outcome thanks to the management of contextual information. In such scenario, different kind of contextual data can be considered [4], and could influence in different ways the reasoning process of the smart component to make a decision. Context management is the whole process of context modeling, context gathering and context reasoning.

2.1 Context Modeling

Context modelling is the process of identifying the exist-

*Corresponding author

Appears in: *Proceedings of the 14th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015), Bordini, Elkind, Weiss, Yolum (eds.), May, 4–8, 2015, Istanbul, Turkey.*

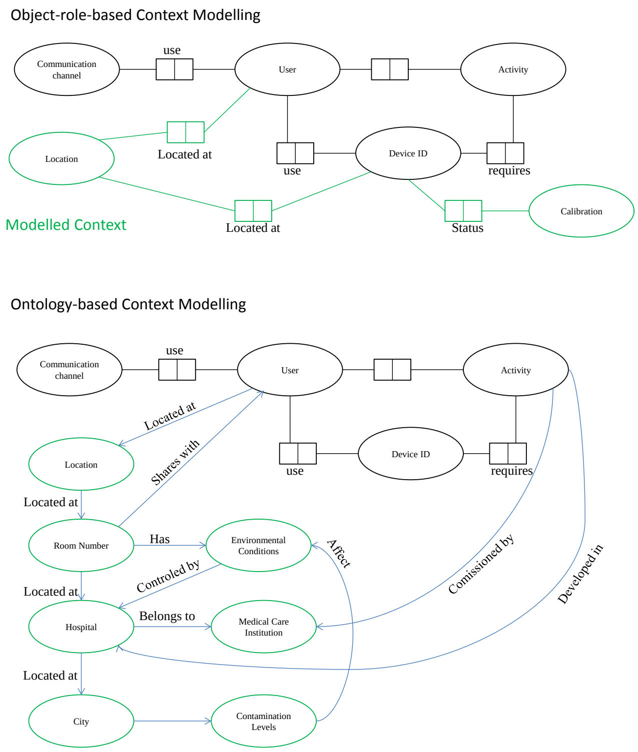

Figure 1: Context modelling techniques.

ing context within an existing domain and to include such context into the whole information model representation. Initially context management and modelling consisted in simple key-value pairs to define lists of attributes and their values describing context information used by context-aware applications. Currently context management has evolved in order to allow capturing a variety of context types; managing relationships, dependencies, timeliness, and quality of context information and supporting reasoning on context. Particularly, such approaches can be classified between object-role based and Ontology-based models [4].

The Object-Role Based Context Modeling (ORCM) is based in the Object-Role Modeling (ORM) languages [19]. ORM is used by software engineers to model software relations into relational databases. ORCM intends to enrich ORM software modeling tools with contextual information in order to allow the designing of context-aware applications. This modeling approach offers the advantage to ease the translation of the modeled context to SQL relational databases, allowing to work with context using SQL-like queries '*select * from users where location is X*'. However, it also has the weakness of accepting only flat context information structures, not allowing the modeling of certain more complex context situations.

When context modeling requires handling hierarchical and more complex information structures it can be represented using Ontology-based models [14]. Such approaches offer the chance to represent complex relations between contextual and non-contextual information whilst allowing to represent hierarchical context information structures. However, it is important to take into account that there is always a trade-off between expressiveness and complexity of modeling.

2.2 Types of Context

The modeling of context is tightly related with the kind of context that is being represented. In this section we propose a classification of the types of context that can be relevant for medical applications. Three main types of context data should be considered: source-related, geographical and temporal, and environmental.

First, healthcare applications that aim to monitor and evaluate the state of the patients need information about their vital signs, which are often measured by sensors. Measurements should always be provided by the same sensors, in order to ensure its proper analysis and comparability between data. Unfortunately, this is not always possible due to contextual characteristics related with the sensor and the retrieval procedure. In addition, it is important to highlight that the mobile device could be considered a sensor itself (or a concentrator of sensors) and its features are also a key issue (i.e. mobile model, operating system, etc.). We consider this kind of contextual information as ***source-related context*** as it concerns the origin of the information. For a measure, source-related context can reflect information like which sensor model has been used, as precision and accuracy can vary from a model to another one; what was the date of calibration of the sensor; who has taken, read and introduced the measures (e.g. if the data is automatically gathered, if the data is introduced by a care given or by the patient itself); when the measurement is related to subjective data, who has interpreted the measurement (e.g. impressions about the aspect of a patient or about the way the patient is feeling), etc. This kind of context allows to assess the reliability of the measurement; for instance, it can be considered that a measurement that is automatically taken by an app using a blue-tooth device is more reliable than a measurement that needs to be manually introduce by the patient, as the patient might misunderstand the device output or commit an error when introducing the measurement to the app. In turn, a measure obtained from an objective source (e.g. a medical device) can be considered more concise than a measure subject to interpretation such as measures evaluating the behavior of a kid.

Geographical and temporal information also plays an important role in healthcare applications, and should be included when modelling context. The place where a patient is or the moment of day when he perform an activity is crucial in assessing his state. Let us suppose an application that controls post-depression rehabilitation: if a patient lives in a big city, he will be subject to a higher level of stress than if he lives in a small town. Therefore, his recovery rate might be different. Furthermore, an application that monitors physical performance will get different results depending on whether it's early in the morning and the user just woke up, it is midday, or it is at night, when the user drags more fatigue. Consequently, geospatial and temporal context information can help to treat the patient data in a more accurate way. This kind of data is one of the most used contextual information in context-aware applications of different fields. For instance, Bose et al. [7] make usage of the geospatial context in their work, in the field of context sensitive services (CSS), for discovering the usage of mobile services that customers performed while being in different locations of a city.

Finally, ***environmental context*** information regarding the location where the patient is staying can also be a relevant aspect. This type of context is used to define the users surroundings, such as ambient conditions (e.g. light, noise levels, humidity or temperature) or social interactions (e.g. the patient shares its environment with smokers). These facts can be considered in order to assess the stress the patient may experiment or its exposure to toxic agents such as smoke. In previous studies, Alonso et al. [2] endowed a multi-agent system with environment and user context in order to enhance assistance and healthcare for Alzheimer patients living in geriatric residences.

2.3 Context Reasoning

First generation of mobile health care apps was designed as sensor concentrators to send the patient information to a server in which took place any reasoning or recommendation feedback for the user. Second generation of health apps are endowed with one or more intelligent modules (e.g. rule-based systems, knowledge-based reasoning [6, 5]) which both, locally (the mobile device itself) and remotely (taking advantage of a server located in a clinical facility), reason about medical and statistical data regarding the patient. The reasoning modules compare the patient data with its own clinical history and with data regarding other patients and then provide a recommendation for the physician and/or the patient. In this scenario, contextual information can enrich the app outputs by means of different procedures.

Contextual information can be used to determine the relevance of each patient measure during the reasoning process [18]. For instance, context information regarding sensors like accuracy and precision can be used to determine if a measure is trustable and, therefore, weight the influence of such measure in the reasoning process. In the same way, the source of the measure (e.g. a sensor, a physician or manually entered by the patient itself) might also condition the reliability of such measure. We call this approach context weighting.

In scenarios where the reasoning modules compare the patient data with other patients' clinic histories, contextual information can be used to narrow the subgroup of patients who will be considered during the reasoning process [10] in what is called context filgering [16]. For instance, patients can be selected according to geospatial and temporal criteria in order to improve the diagnosis of environmental diseases.

Finally, in cases where there is more than one reasoning module, contextual information can be used to decide which module can provide the best recommendations given the user situation. Similarly, the reasoning modules outputs can be merged taking into account the context particularities in what is known as context-boosting [12].

2.4 Knowledge-based Reasoning with Context

Following, methods described in Section 2.3 are illustrated using a well-known reasoning technique as knowledge-based Reasoning (CBR) [1]. Roughly speaking , CBR is the methodology of solving new problems or cases by reusing the information and solutions obtained from past similar experiences and problems (which are stored in a knowledge-base).

2.4.1 *Context weighting*

Context can be useful to, depending on the situation, determine which attributes play the most important role and which are not relevant or not trustworthy (Figure 2). In this approach, after identifying the existing contexts within the knowledge-base, the CBR must be trained in order to learn which weight corresponds to each attribute depending on the context. Hence, each context is associated with a different set of weights, and the weights used during in the CBR depend on the context of the query case. The steps which need to be carried out in order to include context weighting into CBR are the following:

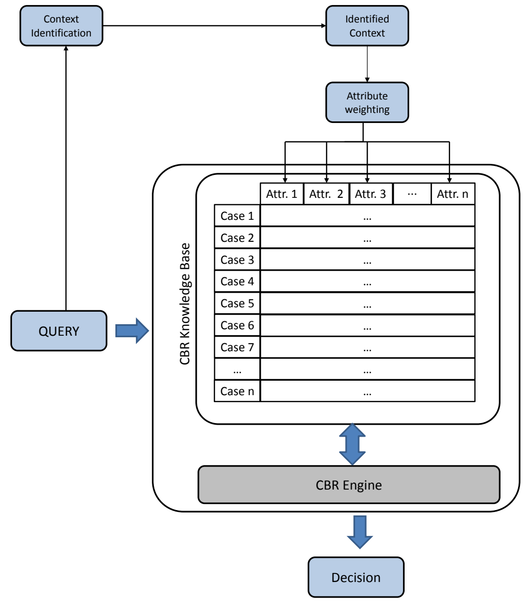

Figure 2: Contextual attribute weighting schema.

- Context representation: Using only the contextual information of the cases stored in the knolwedge-base, the different types of available contexts are identified. The process of identifying the existing contexts can be performed by an expert or by context classification algorithms (e.g. *ConText* [9], data fusion techniques [20] or subgroup discovery [22, 13]). The different types of detected contexts act as labels l which identify the different cases of the knowledge-base.

$$\begin{array}{ll} \text{context:} & \langle c_1, \dots, c_n \rangle \rightarrow \quad \text{label:} \quad l \\ \text{case } i\text{:} & \left(\langle at_1^i, \dots, at_n^i, \rangle, l_{i_j} \right) \end{array}$$

  where c_n corresponds to context attributes and at_n^i to case attributes.

- Attribute weighting: Once the existing contexts have been identified and the knowledge-based labeled, the attribute weights for each type of context must be determined. To this end, each case of the knowledge-base is labeled with its context information. Then, for each of the existing types of context, the weights of the attributes are learned considering only cases labeled with the specific context. The process of learning the attributes can be developed following different approaches (e.g. genetic algorithms [17], swarm intelligence [8] or decision trees [11]) as long as only cases labeled with the same context are used in the process. In this way, for every context label l_m a set of weights is learned:

$$W^{l_m} = \left\langle w_1^{l_m}, ..., w_n^{l_m} \right\rangle \tag{1}$$

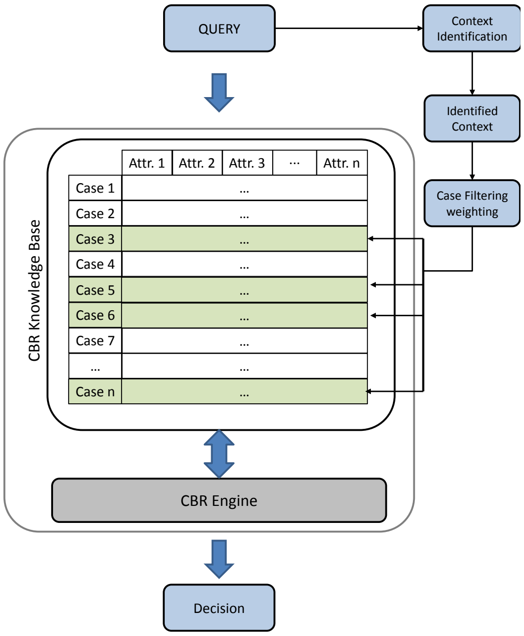

Figure 3: knowledge-base contextual filtering schema.

where W^{l_m} is the set of weights for the context label l_m and $w_n^{l_m}$ is the weight for the attribute n given the context label l_m.

- Query case representation: When a new query case c is introduced into the CBR, the context of the query case needs to be identified (l_c) and, then, the set of weights corresponding to the context of the case is selected.

- Context CBR: After identifying the context of the case and retrieving its corresponding set of weights, the CBR system is executed with the desired parameters but with the context specific set of weighs W^{l_c}. Therefore, the similarity function used in the retrieval stage is conditioned by the attribute weights. For example, the similarity function can be defined as follows:

$$sim(c, i_j) = \sum_{m}^{|AT|} w_m^{l_c} * f_n(at_m^c, at_m^{i_j}) \qquad (2)$$

where $AT = \langle at_1, .., at_n \rangle$ is the set of attributes of the knowledge-base, $w_m^{l_c} \in W^{l_m}$ and $f(at^a, at^b)$ is the metric used to evaluate the similarity between two attributes.

This approach can be useful in cases where the context conditions the validity of certain data. For instance, in a situation where the knowledge-base contains information about cases gathered with sensors with different precisions source-related context can be used to give more importance to the data gathered with the most precise sensors. For the attribute weighting, source-related context is used to give more weight to the measurements which have been obtained under objective and more error-robust methods. On the other hand, environmental and temporal context are used to filter the patients knowledge-base in order to give more importance to those patients who were evaluated under similar circumstances.

2.4.2 Context filtering

The second approach uses context to filter which parts of the knowledge-base are used by the CBR. Context is used to narrow the knowledge-base of the CBR, ensuring that the CBR only compares cases which have happened under similar or the same context. For that purpose, the different instances of the database and the query cases are labeled with their context(s). Context labels define against who a new case is compared to (Figure 3):

- Context representation: Similarly to the previous approach, first of all the existing contexts of the knowledge-base must be identified and the cases labeled.

- Query case representation: Every time a new case is introduced to the CBR its contextual information is analyzed in order to determine which their context labels are. This analysis needs to be performed following the same methodology used for identifying the contexts of the knowledge-base.

- Context CBR: The CBR system is executed with the desired retrieve, reuse, retain and revise methods as it would have been done without the contextual information. However, during the retrieve stage, the CBR considers only instances which share context labels with the query case:

$$sim(c, i_j) = \begin{cases} f(c, i_j) & \text{if } l_c = l_{i_j} \\ 0 & \text{otherwise} \end{cases} \qquad (3)$$

where $sim(c, i_j)$ is the similarity between the query case c and the case i_j of the knowledge-base; $f(c, i_j)$ the metric which defines its similarity; and l_c and l_{i_j} their context labels.

This approach can be useful for cases where the context differentiates the interpretation of the cases' information. For instance, in a case where the knowledge-base contains information regarding patients recovering at a hospital and patients recovering at their home the geographical context information of the patient can condition the interpretation of the data. Whilst geographical and temporal context, and environmental context seem appropriate for this reasoning techniques, context-filtering might not be suitable for dealing with source-related context as the knowledge-base for "non-reliable contexts" (e.g. those cases containing data coming from subjective opinions or non-reliable source) would only be compared with other non-reliable measures.

Note that this approach could be extended by assigning different weights to the different contexts. Giving more importance to the cases which share the same context but without discarding cases with different context.

2.4.3 Context boosting

Finally, context boosting is intended for boosting CBR approaches [17]. In such methodologies different CBR systems cooperate in order to deliver a final solution. Boosting means that the results of different CBR systems are aggregated according to certain rules. For instance, the different solutions can be combined into a single one using a weighted

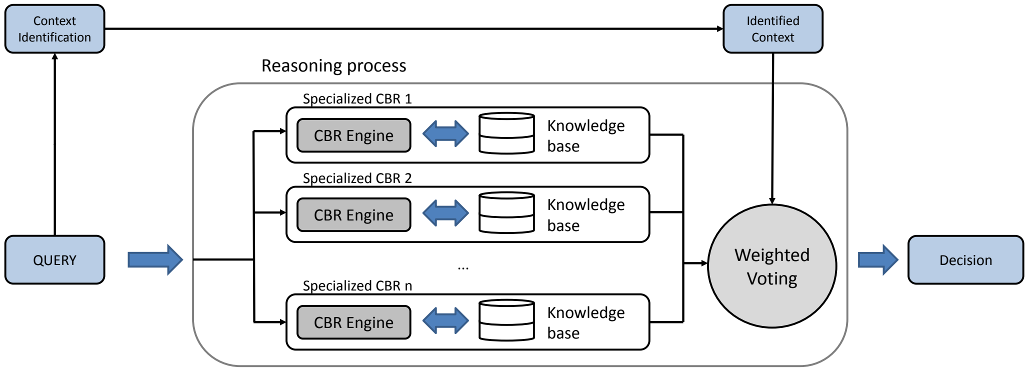

Figure 4: Contextual CBR boosting schema.

multi-criteria decision method (e.g. a weighted sum or a rated ranking [15]). In context boosting, contextual information is used to decide the set of weights to be used during the aggregation process (see Figure 2.4.3).

As in previous approaches, the available contexts within the database should be identified. When this is done, the CBR system needs to be trained in order to learn the set of weights of the aggregation functions which correspond to each of the contexts learned. Then, depending on the context of the query case, one or another set of weights will be used:

- Context representation: First of all, the available types of contexts within the knowledge-base need to be identified (manually or automatically). Conversely to the previous approaches, in boosting the knowledge-base can be distributed among different CBRs systems (which may have different cases among them), therefore the context identification process needs to be done for each existing knowledge-base. Thus, each of the cases is labeled with its context(s).

- Boosting CBR weighting: For each of the identified contexts, the attributes used to weight the different CBRs systems need to be learned. To this end, each set of weights are learned using only cases of a particular context. Therefore, there are as many sets of weights as different contexts, and each set is composed by the same number of weights than the number of CBR systems:

$$WS^{l_m} = \left\langle ws_1^{l_m}, ..., ws_n^{l_m} \right\rangle \quad (4)$$

  where WS^{l_m} is the set of weights for the context label l_m when using a boosting approach and $ws_n^{l_m}$ is the weight for the output of the CBR n inside the boosting schema.

- Query case representation: the context l_c of the new query case c needs to be identified.

- Context Boosting: After the context of the case has been identified, the set of weights corresponding to the case context is selected. Then, the query case query is submitted to each of the boosting CBRs. Every CBR component within the boosting schema studies the received case and proposes a solution according to their knowledge, next, they submit their different solutions. The different solutions are then combined using the desired aggregation function but taking into account the set of weights corresponding to the case of the query case c. In this way, the solution of the CBR is the following way:

$$r^c = mcdm(S, WS^{l_m c}) \quad (5)$$

  where r^c is the result of the query case c, WS^{l_c} is the set of weights for the context label l_c, $S = \langle sol_1^c \dots sol_k^c \rangle$ the solutions to the case c provided by each of the boosting CBRs and $mcdm(S, WS^{l_c})$ the multi-criteria decision method used to combine the provided solutions.

This approach is useful to aggregate the outputs of different CBRs that can be specialized in different types of context. For instance, a boosting CBR where the CBRs' knowledge-bases contain information of cases from different geographical areas (in this case context might give more weight to the knowledge-bases containing information from the same geographical area than the query case); note that in this case, the learning process will weight more the CBRs having a relevant relation with the specified context.

3. CASE OF STUDY

The methodology described in the previous section has been applied in a smart e-Health app for remote patient monitoring. Particularly, the application is intended for assisting parents in charge of premature born babies through a smartphone application and a set of sensors. The application uses data gathered by the sensors together with information provided by the baby's progenitors in order to assess the newborn development and, if necessary, warn parents and doctors regarding abnormal situations. In such scenario, the application uses context information in order to improve the system outputs and to evaluate the quality of the data gathered.

3.1 Premature neonate's motorization use case

Usually, preterm infants are discharged when they achieved a certain weight, typically around 2200g. However, there are some babies that only need supervision rather than medical treatment while they achieve this weight. Then, when a baby is suitable for minimal medical care, parents can measure the vitals of their infant comfortably at home and, using a smartphone, submit the results to be efficiently reviewed by the doctor.

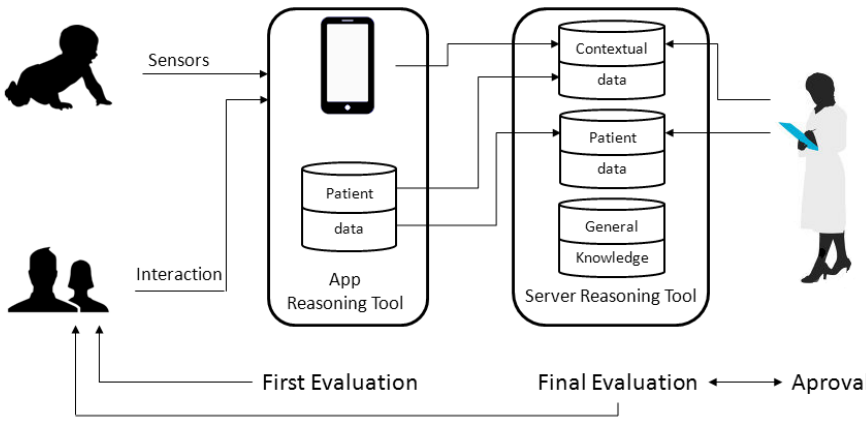

Figure 5: Simplified schema of the app and its architecture

The baby is monitored following a protocol given by the hospital, using non-invasive sensors such as scales, pulse-oxymeters or skin thermometers. Sensors information is gathered by a smartphone app. The app also requires progenitors to answer a set of questions regarding other parameters (e.g. daily stools, sleeping periods, etc.). Using a reasoning module (particularly, a combination of a rule-based system and a case based reasoning tool [17]), the app performs an analysis of the data and offers a first evaluation of the neonate current status (normal recovery, abnormal recovery, emergency actuation) which is shown to parents; this reasoning is performed using only information of the baby itself and general knowledge provided by doctors. The app, then, sends all the collected information together with its evaluation to a server located at the hospital facilities (see Figure 5). There, a more powerful reasoning module analyzes the data and enriches the evaluation by means of information regarding the patient history, information regarding other patients and contextual information gathered by the app, the hospital information service and the medical staff. Finally, the evaluation is checked by a doctor.

3.2 Handling Context Information

The contextual information used in the presented application concerns the three different aspects described in the methodology. First, our application monitors the quality of the sensed measures by means of source-related contextual information which can include its calibration data, its lifetime, its accuracy and its precision. Moreover, the app also keeps track of who or what provided the measure (a sensor, the patient, doctors or caregivers). This allows us to, based on the measure's quality, decide the influence of the measure during the reasoning process.

Second, regarding geographical and temporal data, the date and time of the measure is a key issue, since having all the measures gathered at the same moment of the day influences in their value. However, the localization is not important because the baby should not be moved from home. And third, regarding environmental data, in our app we include information regarding the family of the newborn (smoking habits, number of children in the home, etc.) and the home ambient conditions (light, temperature and humidity).

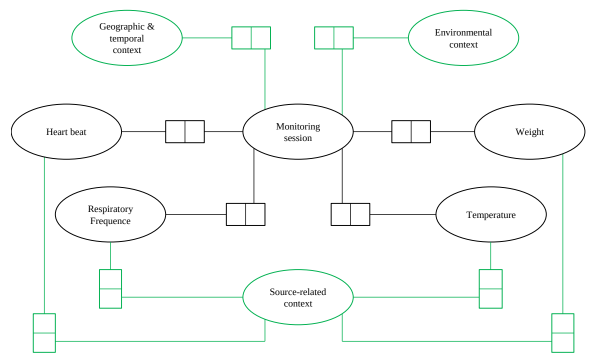

Figure 6: Simplified schema of the context modeling in the app.

Contextual information is gathered at two stages (Figure 6): for each monitoring session environmental, geographical and temporal context information is gathered and associated to the monitoring session (See Table 1). In addition, for each measure taken during the monitoring session source-related context regarding this particular measure is also stored (Table 2).

Regarding reasoning, we rely on knowledge-based reasoning. The knowledge-based reasoning module on the mobile device, provides outcomes based only on the baby historical data; this module is enriched with a rule-based system which prevents cold-start problems and provides a security mechanism by automatically warning about critical measurements. At the backend server, a more powerful knowledge-based reasoning module perform recommendations based on information about similar patients at a similar context (thus considering the own patient history but also information about other similar patients). As new measures are performed, the baby's historical data is improved and the reasoning module at the devie becomes more trustable. When this occurs, the reasoning module placed in the back-end becomes less important as their evaluations barely improve the ones of the mobile app. Regarding the use of context, the back-end cbr module combines attribute weighting (Section 2.4.1) and knowledge-based filtering (Section 2.4.1).

Parameter	Description	Values
Date & time	Records the instant where monitoring is done	Date
Light conditions	Describes if the habitat has good or bad light conditions	0 to 5
Humidity conditions	Describes if the habitat has good or bad light conditions	0 to 100 %
Smokers	Presence of smokers in the patient's habitat	Yes / No
Infants	Presence of other infants in the patient's habitat	Yes / No

Table 1: Context gathered in each monitoring session

Parameter	Description	Values
Source	Describes the origin of the data	Sensor, Impression
Introduction method	Describes how the data is introduced to the app	Wirless, Wired, Manual
Sensor Model	Defines the sensor model used to take the measure	String
Sensor Reference	Defines the particular sensor used	String

Table 2: Source-realted context gathered in each measure

3.3 Results

The app and the platform have been tested using in-silico data representing babies with different basal situations and with different contexts. Table 4 illustrates some of those tests: the first column provides the identifier of the monitoring session (chronologically ordered), the second column corresponds to the outputs of the mobile device reasoning module (which uses information from a single baby and ignores context), the third column corresponds to the outputs of the back end reasoning module (which uses context and information from multiple babies), the last column corresponds to the feedback of a pediatrician that evaluated whether the warning of the system was or was not necessary. Table **??** evaluates the results in terms of precision and recall, showing that the system is highly reliable in terms of recall (1.0) but has a low precision (0.34 and 0.61 depending on the reasoner).

The results show that the local reasoning module can be reliable, as all the situations that a pediatrician considers problematic are detected by the combination of the rule-based system and the CBR. Nevertheless, this reasoning module provides a high number of false positives, which can provoke a high anxiety in parents. In this sense, the back end CBR provided the same results in terms of false negatives, but with much less false positives. This fact is produced by the higher predictive power of the back end CBR which has a bigger knowledge-base and, in turn it can also deal with contextual information. The level of precision of the system could probably be improved by relaxing the strictness of the rule-based system, nevetheless, this would also reduce the recall performance of the app, which according to doctors was much more important than precision for this specific problem.

It important to note that this results are still preliminary as they have been obtained from in-silico data. The low volume of available information records regarding the vital signs of premature born babies hampers the offline testing and validation of the presented platform. To tackle this lack of data, a proof of concept will be carried in short.

	TP	FP	TN	FN	Precision	Recall
App reasoner	43	80	64	0	0.34	1.00
Hospital reasoner	43	28	117	0	0.61	1.00

Table 3: Results of the tests (TP stands for true prositives, FP false positives, TN true negatives, FN false negatives)

4. CONCLUSIONS

This paper analyzed context management for health applications. To this end, we have classified the different contexts that can be involved in health-apps in three different categories: source related context, geographical and temporal context, and environmental context. Such context can be used to improve the outputs of health apps in what is known as context reasoning. Particularly we have distinguished between attribute weighting, which weights the attributes relevance depending on context; knowledge-filtering, which uses context to narrow the knowledge-base; and context boosting, which uses context to merge the outputs of different reasoners or classifiers. These techniques have been illustrated by showing how they can be applied to knowledge-based reasoning.

This work has also presented a case of study where context is used in a eHealth application, an app for remote patient monitoring. The paper described how the context is integrated into the app, how it is modeled and which context variables are used. Preliminar tests performed with in-silico data have shown that context can be useful to enhance the reasoning power of the app, and a further proof of concept will be carried on.

As a future work, it would be interesting to study how context can be represented by existing medical terminologies (e.g. SNOMED-CT [3]) and to analyze if the semantic information of those terminologies can be used to empower context reasoning.

Acknowledgements

The work described in this paper was carried out as part of the MoSHCA project. The project has been funded by the Ministerio de Economía y Competitividad of the Spanish Government (Ref. EUREKA ITEA 2 n^{o} 11027 - IPT-2012-0943-300000) according to article 31 in General Grants Law 38/2003, November 17th, approved by Real Decreto 887/2006, July 21st. The project is also cofounded by European Union through the European Regional Development Fund (ERDF).

REFERENCES

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

[2] R. S. Alonso, D. I. Tapia, Ó. García, D. Sancho, and M. Sánchez. Improving context-awareness in a

Baby # case	Smartphone-app Reasoning module evaluation:	Hospital reasoning module evaluation:	Physicians Evalution
1.1	Normal	Normal	Normal
1.2	**Normal**	Normal	Normal
1.3	**Warning**	Normal	Normal
1.4	**Warning**	**Warning**	**Warning**
1.5	**Warning**	**Warning**	**Warning**
2.1	Normal	Normal	Normal
2.2	**Warning**	**Warning**	**Warning**
2.3	**Warning**	Normal	Normal
2.4	**Warning**	Normal	Normal
2.5	**Warning**	Normal	Normal
2.6	**Warning**	Normal	Normal
2.7	**Warning**	**Warning**	Normal
3.1	Normal	Normal	Normal
3.2	**Warning**	**Warning**	**Warning**
3.3	Normal	Normal	Normal
3.4	**Warning**	**Warning**	Normal
3.5	Normal	**Warning**	Normal
3.6	**Warning**	Normal	Normal

Table 4: Test results examples

healthcare multi-agent system. In *Trends in Practical Applications of Agents and Multiagent Systems*, pages 1–8. Springer, 2011.

[3] T. Benson. *Principles of health interoperability HL7 and SNOMED*. Springer Science & Business Media, 2012.

[4] C. Bettini, O. Brdiczka, K. Henricksen, J. Indulska, D. Nicklas, A. Ranganathan, and D. Riboni. A survey of context modelling and reasoning techniques. *Pervasive and Mobile Computing*, 6(2):161–180, 2010.

[5] I. Bichindaritz and C. Marling. Case-based reasoning in the health sciences: What's next? *Artificial intelligence in medicine*, 36(2):127–135, 2006.

[6] I. Bichindaritz, S. Montani, and L. Portinale. Special issue on case-based reasoning in the health sciences. *Applied Intelligence*, 28(3):207–209, 2008.

[7] I. Bose and X. Chen. A framework for context sensitive services: A knowledge discovery based approach. *Dec. Support Systems*, 48(1):158–168, 2009.

[8] P.-C. Chang, J.-J. Lin, and C.-H. Liu. An attribute weight assignment and particle swarm optimization algorithm for medical database classifications. *Computer methods and programs in biomedicine*, 107(3):382–392, 2012.

[9] W. W. Chapman, D. Chu, and J. N. Dowling. Context: An algorithm for identifying contextual features from clinical text. In *BioNLP 2007: Biological, Translational, and Clinical Language Processing*, pages 81–88, 2007.

[10] A. Chen. Context-aware collaborative filtering system: Predicting the users preference in the ubiquitous computing environment. In *Location-and Context-Awareness*, pages 244–253. Springer, 2005.

[11] S. Z. Doğan, D. Arditi, and H. Murat Günaydin. Using decision trees for determining attribute weights in a case-based model of early cost prediction. *Journal of Construction Engineering and Management*, 134(2):146–152, 2008.

[12] C. Hayes and P. Cunningham. Context boosting collaborative recommendations. *Knowledge-Based Systems*, 17(2):131–138, 2004.

[13] F. Herrera, C. J. Carmona, P. González, and M. J. del Jesus. An overview on subgroup discovery: foundations and applications. *Knowledge and information systems*, 29(3):495–525, 2011.

[14] J. Kim and K.-Y. Chung. Ontology-based healthcare context information model to implement ubiquitous environment. *Multimedia Tools and Applications*, 71(2):873–888, 2014.

[15] B. López, C. Pous, P. Gay, and A. Pla. Multi criteria decision methods for coordinating case-based agents. In *Multiagent System Technologies*, pages 54–65. Springer, 2009.

[16] S. Mehta and H. Banati. Context aware filtering using social behavior of frogs. *Swarm and Evolutionary Computation*, 17:25–36, 2014.

[17] A. Pla, B. López, P. Gay, and C. Pous. exit* cbr. v2: Distributed case-based reasoning tool for medical prognosis. *Dec. Sup. Systems*, 54(3):1499–1510, 2013.

[18] W. J. Scheirer, N. Kumar, P. N. Belhumeur, and T. E. Boult. Multi-attribute spaces: Calibration for attribute fusion and similarity search. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2933–2940. IEEE, 2012.

[19] L. Snidaro, J. García, and J. Llinas. Context-based information fusion: a survey and discussion. *Information Fusion*, 2015.

[20] A. N. Steinberg and C. L. Bowman. Adaptive context discovery and exploitation. In *Information Fusion (FUSION), 2013 16th International Conference on*, pages 2004–2011. IEEE, 2013.

[21] G. S. Stergiou, A. S. Zourbaki, I. I. Skeva, and T. D. Mountokalakis. White coat effect detected using self-monitoring of blood pressure at home comparison with ambulatory blood pressure. *American journal of hypertension*, 11(7):820–827, 1998.

[22] A. Vavpetič and N. Lavrač. Semantic subgroup discovery systems and workflows in the sdm-toolkit. *The Computer Journal*, 56(3):304–320, 2013.