An Agent Framework for Dynamic Health Data Aggregation for Research Purposes

Alevtina Dubovitskaya
AISLab, HES-SO,
LSIR, EPFL, Switzerland
Alevtina.Dubovitskaya
@hevs.ch

Visara Urovi
AISLab, HES-SO, Switzerland
Visara.Urovi@hevs.ch

Karl Aberer
LSIR, EPFL, Switzerland
karl.aberer@epfl.ch

Michael I. Schumacher
AISLab, HES-SO, Switzerland
Michael.Schumacher
@hevs.ch

ABSTRACT

This paper presents a model of a MAS framework for dynamic aggregation of population health data for research purposes. The contribution of the paper is twofold: First, it describes a MAS architecture that allows one to built on the fly anonymized databases from the distributed sources of data. Second, it shows how to improve the utility of the data with the growth of the database.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Keywords

Dynamic Data Aggregation, Cooperative Agents, Privacy, Anonymization, TuCSoN

1. INTRODUCTION

The treatment of certain diseases, such as cancer, HIV, or other serious medical conditions, relies on a regular administration of critical drugs that are necessary to keep those life-threatening diseases under control. Those drugs (e.g. *Efavirenzum, Gentamicin, Imatinib, Tacrolimus, Tobramycin*) have a narrow therapeutic range and a poorly predictable relationship between the dose and the drug concentration in the blood, which may greatly vary among individuals.

Therapeutic Drug Monitoring (TDM) aims to tackle this problem by monitoring drug levels in the blood and adjust the dosage individually. TDM employs mathematical models that are based on the analysis of the population healthcare data. These models are developed by pharmacologists and allow one to compute characteristics (pharmacokinetic parameters) of the drug based on the patient' covariates and, therefore, to make a personalized recommendations of the drug dosage for the optimal treatment. In order to create, evaluate, and enhance the models the population health data are needed.

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.*

Both the structure and the representation of the health data that need to be aggregated for the research purposes depend on the requirements for a particular study. Therefore, it is not possible to specify a unique static schema of the research database (*RSDB*) for different types of clinical research. Only in TDM data requirements significantly vary for different drugs. For instance, in order to study the pharmacokinetics of *Gentamicin* in neonates, gestational age and postnatal age of the patients has to be specified in weeks and days correspondingly [8]. However, conducting research in adult population does not require such detailed data. Hence, the same data representation will be not only inconvenient but also will complicate the modeling. Therefore, it is not feasible to build one centralized *RSDB* that will fit the requirements of different studies that involve medical data.

Datasets containing health related information about an individual are increasingly becoming "open". However, if one aims at building a publicly available centralized database that contains patients' data and tries to keep them as detailed as possible, patients' privacy may be violated. For example, insurance companies may infer that a person is suffering from a chronic disease and may be willing to refuse an application or reject the renewal of the policy. An employer may try to infer healthcare data of potential employees and based on the information (e.g., a serious health condition or a chronic disease susceptibility) and may discriminate a candidate based on his or her sensitive data. Therefore, in case of medical data aggregation the data have to be de-identified such that the re-identification of an individual is impossible.

Building a dynamic and privacy-preserving mechanism for healthcare data aggregation in a distributed environment is of a high importance. Mainly because it would allow one to collect statistically significant amount of data in a shorter period, since several sources of data are available. However, one has to take into account that aggregation of the data even from the locally anonymized databases can reveal sensitive information about the patient (e.g., in case of a composition attack first described in [2]).

Several models in the area of distributed privacy-preserving data publishing have already been proposed (i.e., pseudonymization [7, 20], secure multi-party computations (SMC) [4], microaggregation [17], cloning [2]). However, those models significantly affect the utility of the data, since they do not take into account certain requirements regarding the use of *RSDB*: such as data structure and data representation.

The novelty of our work stands on using a MAS system for defining a dynamic mechanism for privacy-preserving data aggregation

from multiple sources with a possibility to improve data utility with the database growth. We employ multi-agent coordination and the publish/subscribe paradigm to find potential contributors and to achieve an agreement on the characteristics of *RSDB*. This is done taking into account the data requirements specified by the study, the data that are currently available, and the privacy guarantees. With the growth of *RSDB* we are able to maximize the utility by performing de-generalization in the distributed environment. For this we employ the properties of the functional encryption scheme proposed in [1] that allows to decrypt only a result of a function applied to the encrypted data.

The advantages of using our approach are the following:

- Participation in data collection is based on the similarities of the data that the agents possess. Therefore, the organization of peer-to-peer (P2P) network of the potential contributors is efficient.
- The characteristics of the database can be defined with respect to a particular research question and current states of the contributors - local databases, *LDB*s. Therefore, it maximizes the utility of the data.
- More fine-grained (less anonymized) data can be collected in a shorter period.

The rest of the paper is organized as follows. In Section 2, we describe the existing models and schemes that we employ in our framework and compare our approach with the models proposed in the related work. Section 3 presents the architecture of the MAS, data structure, P2P network organization and coordination between agents. In Section 4, we define the problem of de-generalization of the data stored in *RSDB* and present a solution for the one-step de-generalization problem. We conclude and list the directions of our future work in Section 5.

2. BACKGROUND KNOWLEDGE

In this section, we provide the background knowledge about existing models and algorithms that we use for building our MAS framework. We describe the publish/subscribe paradigm and TuCSoN [14] - the models that are used for the coordination in the distributed environment. We also describe privacy model that we apply for aggregating the data in *RSDB*s, as well as the properties of the cryptographic algorithms that we use to improve the utility of *RSDB*s in a privacy preserving way. In the Subsection 2.4 we present an overview of the related work and specify how the existing models differ from our approach.

2.1 Coordination Models

In this paper, we assume two types of coordination: the first is publish / subscribe – to search for and publish information about the data held in various clinical systems and to organize a P2P network, the second one is TuCSoN [14] – a coordination model for negotiation and data exchange. The publish/subscribe paradigm allows one to deliver the data from their producers (publishers) to their consumers (subscribers) in the distributed environment in the decoupled fashion [11]. This means that publishers can introduce the data into the system (publish/subscribe broker) being unaware of the subscribers. Subscribers can register their interests by subscriptions which act as queries or filters and are used to deliver relevant events to the subscribers. The context publish/subscribe broker enables publication of context information by publishers, so that the information becomes available to subscribers which are interested in processing the information provided by publishers.

Interaction between agents within TuCSoN coordination model is happening through the tuple spaces that can be seen as a shared system such as blackboard system [9]. Using the tuple center an agent can insert (`out` operation), read (`rd` operation) and consume (`in` operation) the tuples. The templates of the tuples need to be specified with respect to their structure, or the ontology model needs to be employed to interpret the information transferred by the tuples. In order to establish interaction between agents the coordination rules can be set up. ReSpecT [5] – the first-order logic language – allows one to define the behavior of the tuple centers. The reaction rules syntax is defined as follows:
`reaction (action, conditions, react)` ,
where `action` is an operation that was performed at the tuple center, `conditions` – are the conditions that need to be verified before the execution of `react`, that describes the events caused by the `action` if the `conditions` were satisfied.

2.2 Anonymity of Medical Data

A variety of privacy models, (e.g., $k-anonymity$, $(k^m)-anonymity$, $l-diversity$, ε–differential privacy, etc. [10]) can be used for privacy preserving data publishing. However, Poulis et al. show that all these methods are not appropriate for the anonymization of the datasets containing both relational (i.e., single-valued) and transaction (i.e., set-valued) attributes, such as medical datasets that contain patient demographics and diagnosis information together [15].

$(k,k^m)-anonymity$ proposed in [15] ensures that for any record r in the dataset and any set of m or less items in transaction attribute of r, there should be at least $(k-1)$ records that are indistinguishable from record r. However, $k-anonymity$ for relational attributes (i.e., existence of at least $(k-1)$ records that are indistinguishable from record r with respect to relational attributes of the record r) and $(k^m)-anonymity$ for transaction attribute do not imply $(k,k^m)-anonymity$. Poulis et al. developed two frameworks that produce $(k,k^m)-anonymous$ datasets with bounded information loss in one attribute type (relational or transaction) and minimal information loss in the other (transaction or relational).

In this paper we focus on the anonymity of single valued attributes, $k-anonymity$, to simplify the description of our MAS system. However, our system supports $(k,k^m)-anonymity$ by design.

2.3 Functional Encryption and Secret Sharing Scheme

Functional encryption is a new paradigm in public-key cryptography. It gives users more control of the amount of information that is revealed by a ciphertext to a given receiver [1]. It allows the receiver to decrypt only a result of the function (e.g., `sum`) applied to the encrypted data without having access to the parameters of the function (e.g., the terms of the resulting summation). The functionality of this approach is based on the homomorphic properties of the ElGamal scheme [1] that allows to sum up encrypted values by multiplying corresponding ciphertexts.

Shamir secret sharing scheme allows to divide data D into n pieces in such a way that D is easily reconstructible from any k pieces, but even complete knowledge of $(k-1)$ pieces reveals absolutely no information about D [16]. The scheme is based on polynomial' interpolation and provides the following guarantees: (1) knowledge of any k or more pieces makes D easily computable; (2) knowledge of any $k-1$ or fewer pieces leaves D completely undetermined. This is called (k,n) threshold scheme. In our MAS we will use the scheme where $k=n$, meaning that if the secret D is divided into n pieces, all the pieces (n) are required to reconstruct

D.

2.4 Related Work

Urovi et al. in [19, 18] proposed a secure mechanism for *EHR* exchange over a P2P agent based coordination framework. In this approach the encrypted heterogeneous data are exposed over a P2P network. The authors provide the algorithms for searching and for publishing the *EHR*s in the untrusted P2P network without compromising the privacy, integrity and the authenticity of the shared data. This work, however, does not cover the aggregation of the data for the research purposes, as we propose here.

MOSAIC [3] is a protocol for clinical data exchange with multilateral agreement. It provides a way of exchanging clinical records for the purpose of conducting medical research. MOSAIC allows clinicians to search for specific cases in the other sources and compare with the other cases that they already have. However, it does not allow one to build a shared research database with respect to the particular requirements since this approach does not consider the data structure and data representation. Moreover, the patients' privacy is not taken into account in the design of the protocol.

Elger et al. [7] present strategies for medical data exchange. The authors give an overview of technical, practical, legal, and ethical aspects of secondary data use and discusses their implementation in the multi-institutional @neurIST research project. The pseudonymization approach and a hight-level hybrid access-control system that partially address privacy issues are presented in this paper. The authors also list security vulnerabilities, including the possibility of cracking the proposed pseudonym generation mechanism, dependence on a trusted third party and the possibility of establishing an indirect identification. However, they do not provide any solutions to these problems.

An approach for continuous privacy preserving publishing of data stream is presented in [21]. The authors use R-trees, and allow publishing the data into the research database only after performing microaggregation locally. Similar to another approach based on two-phase microaggregation proposed in [17] the authors do not present any algorithm that allow the sources of data (medical institutions) to negotiate and find the agreement on the characteristics of the research database (including anonymity parameters).

3. COORDINATION BETWEEN AGENTS

In this section we present our MAS architecture and the functionalities of its components. We describe the structure of the data that are stored in the databases: *LDB* that is the local knowledge and the resources of an agent, and *RSDB* – one of the multiple anonymized databases constructed for the research purposes. We also present the structure of the metadata that are used during the processes of P2P network organization and agents negotiation.

3.1 MAS Architecture

The architecture shown on the Figure 1 presents a network with two types of nodes. The first type is an agent that is either interested in data aggregation or is able to share the data for the research purposes. The second type - is a node that stores the metadata of the existing *RSDB*. The nodes are connected in the P2P network based on a publish/subscribe paradigm. Subscription rules indicate what type of data the node is interested in or will be willing to provide. When the construction of *RSDB* starts the metadata of *RSDB* is published so that the nodes (including the ones that join the network later on) with corresponding subscription rules would be able to access and populate the *RSDB* based on its metadata.

The knowledge of an agent is represented by a local database, *LDB* (database located in a medical institution), and its metadata.

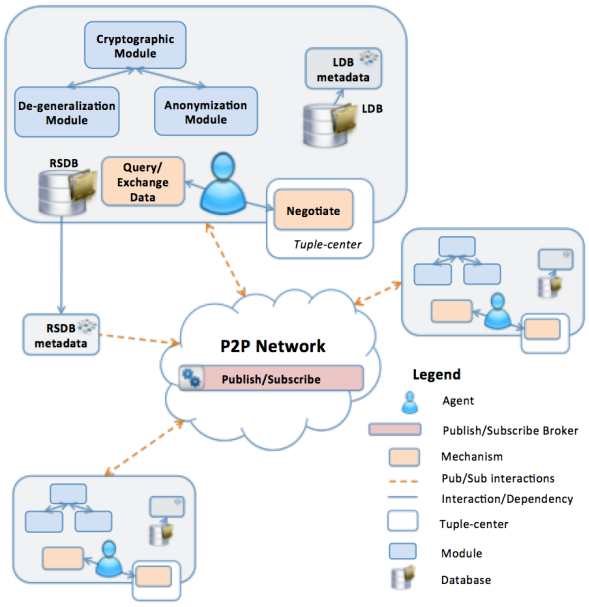

Figure 1: Architecture of the multi-agent system

An agent can use the functionalities of the following modules: Cryptographic Module, Anonymization Module and De-generalization module.

The functionalities of the Cryptographic Module are the following: first, to create pseudonyms with which the data about the patient will be uploaded to *RSDB*; second, to perform functional encryption and to take part in secret sharing scheme; and, third, to generate the signature before data transfer in order to ensure the authenticity and integrity of the data. Anonymization Module is a realization of the medical data anony-mization that includes the generalizaion step. The functionalities of the modules are described in details in our previous work[6]. De - generalization Module is used to improve the utility of the data with the growth of *RSDB*. Its description is presented further in the paper, in Section 4.

The following mechanisms are available to the agents: Query / Exchange Data and Negotiate. Query mechanism is used to query the subscription rules as well as the metadata of *RSDB*. Exchange Data mechanism is used to transfer the data to the research database and for the metadata exchange. Negotiation is based on the TuCSoN coordination model [14] and aims at adjusting the characteristics of *RSDB* (e.g., the scheme of the database, statistically significant number of records to be collected, anonymity parameters). In addition, semantic agreement between schema of different databases and different representations of the data that are stored in *LDB* needs to be established. Ontologies and existing schema matching solutions [12, 13] can be employed during the negotiation phase.

3.2 Data Structure

Local Database, *LDB*, stores healthcare data about the patients that receive treatment from the particular caregiver or in a certain medical institution. In particular, it contains *Pseudonym(s)*, QID^N, and *Healthcare data*.

Pseudonym(s) – a set of uniquely identifiable patient data, ID_P, (such as combination of date of birth, place of birth and the name) that are stored in an encrypted form. Pseudonyms are used in order to recontact the patient if needed. It is only possible for the doctor

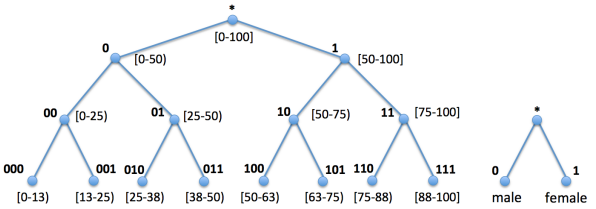

Figure 2: Example of a representation of single valued QID: age and gender using binary trees

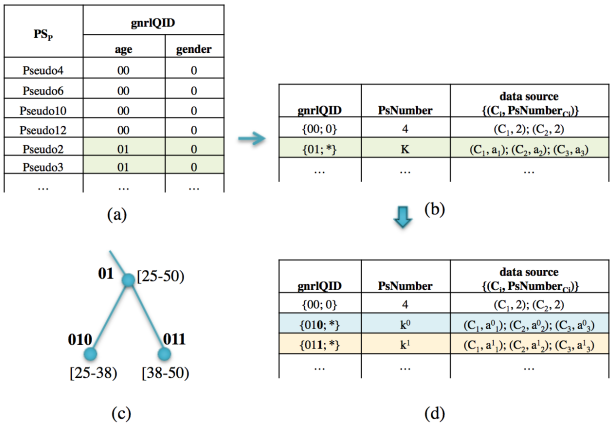

Figure 3: An example of one-step de-generalizarion: part of *RSDB* (a), database that represents the state of the *RSDB* before (b) and after (d) one-step de-generalization, selected nodes from the generalization tree (c).

that uploaded the data, or according to the access control policy specified by the patient.

QID^N – quasi-identifiers – a set of the attributes ($\{qid_n\}$) that in combination can uniquely identify the person (e.g., single-valued qid_n, such as age, gender, address (i.e., ZIP code) and set-valued qid_n, such as diagnosis codes). Therefore, QID^N can not be sent to *RSDB* as they are, they need to be generalized (i.e., the value needs to be substituted by a range, the set – by a set with higher cardinality). $gnrlQID^N$ – a combination of generalized qid_n (in a form of a binary string), with which the data about patient *P* are to be uploaded to the *RSDB*.

In order to represent the way of generalization the binary trees are used. An example of such trees is shown on Figure 2. Each node corresponds to the range (if the attribute is numerical) or to the value generalized up to a certain degree (if the attribute is categorical). Every node is coded as a binary string depending on the depth of the node. The structure of a thee is defined according to the particular study an requires a knowledge of an expert from the corresponding field. The description of a tree could be stored in an XML-file, or as comma-separated values, or using JSON[1] - lightweight easily readable data-interchange format.

Healthcare data – drug intakes (time, dosage, drug name), characteristics of the drug (absorption, clearance), co-medications, covariates (weight, age), concentration measurements (time, measurement) – multiple attributes, that can be set-, or single-valued).

The metadata of *LDB* consists of the following information: *KW* – a set of the keywords that describe the content of the database (e.g., can contain the name of the drug or the diseases which the data from the database are related to), $RANGE^N$ – is a set of intervals, where each interval, $range_{qid_n}$, refers to the range of the values corresponding to a certain attribute, qid_n. N_{rec} – is a number of records stored in the database. N_{rec} can be dynamically updated with respect to *KW*. The metadata of *LDB* plays an important role in the functionality of the whole framework, first, as the metadata are used to make a decision what data will be shared and aggregated. Second, as the metadata are used in negotiation phase to adjust the characteristics of the *RSDB*.

RSDB can be seen as a combination of the data aggregated from different *LDB*s and anonymized on the fly. As *LDB*, *RSDB* also contains *Pseudonyms* and *Healthcare data*. However, instead of QID^N, *RSDB* stores their generalized versions, $gnrlQID^N$, because publishing QID^N as they are may result in re-identification of the patient, thus violation of patient' privacy.

StRSDB–is a database that stores the metadata of *RSDB*, and are constantly updating to reflect the current state of *RSDB* during the process of *RSDB* construction. For each combination of $gnrlQID_r^N, \forall r \in \overline{1,S}$[2] that are presented in *RSDB*, *StRSDB* stores the following information: *PsNumber* — a number of different *pseudonyms* from *RSDB* associated with the same $gnrlQID_r^N$ and an array that shows the source of data, in particular: C_i, identity of the caregiver that uploaded the data, and $PsNumber^i$, a number that shows how many *pseudonyms* associated with this $gnrlQID_j^N$ have been uploaded by C_i. One has to notice that as *RSDB* is $k-$*anonymous*, $PsNumber \geq k$ and $\sum_{i \in \overline{1,T}} PsNumber^i = PsNumber$. Figure 3(b) presents an example of *StRSDB*.

3.3 P2P Network Organization

We assume that medical institutions will be willing to collaborate and share the data based on the following reasons. First, they could be interested in aggregation of data similar to the data they possess in order to obtain statistically significant number of cases for their own research purposes. Second, we can also assume that they would provide the data without looking for a certain profit but aiming at patient care improvement in general by providing the data for the research.

Publish/subscribe approach is used in our framework in order to create a P2P network of nodes that are interested to create, to populate or to use particular *RSDB* with certain requirements. We consider the following actors: Data Publisher, Subscriber, and Metadata Publisher. Data Publisher – is an agent that possesses certain data and is interested in their sharing and, potentially, using *RSDB*. Subscriber – is an agent that is interested in obtaining a database for the particular research. If there is no suitable database already built, the Subscriber can initiate the process of the data aggregation. When the process starts the Subscriber creates another node - Metadata Publisher. It will host the research database (*RSDB*) and its metadata providing the information about *RSDB* to the network even if the Subscriber that initiated data aggregation leaves the network. Every subscription contains a part of metadata of *LDB* such as keywords (*KW*), and, possibly, the ranges of certain attributes that represent the data stored in the *LDB*.

Publisher initiates the process by sending the initial description of the *RSDB* such that the mediator – context publish-subscribe broker – can simply match the requirements with the data provided by publishers (Data Publishers and Metadata Publishers). If there is no suitable *RSDB* found the data aggregation process starts from organizing the network of potential contributors of the data based on the data provided by Data Publishers. Detailed definition of the publish-subscribe rules and their implementation are the focus of

[1] http://json.org

[2] *S*-is a number of different combinations of $gnrlQID^N$

our future work.

3.4 Negotiation between Agents

We assume that P2P network is organized when the nodes from Data Publishers are selected and they start communicating with the Subscriber that initiated the process of data aggregation. Different types of interactions that occur between agents while building *RSDB* can be split into the following groups depending of the aim of the interaction: establishing sources of the data, adjusting parameters of *RSDB* (*RSDB* schema, anonymity parameters, generalization trees), and updating *RSDB* (by populating it with the new records or updating with the de-generalized ones).

First, we assume that the agent (further, initiator) that is interested in data aggregation starts interaction with the potential contributors trough the tuple centers by specifying more concrete requirement for *RSDB*. In order to establish the sources of the data it will start communication with the agents identified at the previous step by writing the tuple t_1:

$t_1 = \{KW(RSDB), \{range_{qid_n}(RSDB)\}, N_{rec}(RSDB)\}$.

The agent receiving the information will act according to the following rule:

`reaction (out(`t_1`), condition_1, out_r(`t_2`));`

condition_1: $KW(RSDB) \subset KW(LDB), range_{qid_n}(RSDB) \subset \subset range_{qid_n}(LDB), n \in \overline{1,N}, N_{rec}(RSDB) \subset N_{rec}(LDB)$[3].

$t_2 = \{KW(LDB), \{range_{qid_n}(LDB)\}, N_{rec}(LDB)\}$.

where `condition_1` verifies whether the agent, that received the message, possesses the data required and is ready to contribute to *RSDB*. If the verification is successful, the agent (further, contributor) shares its *LDB*' metadata.

Hereafter we show the description of the rules for the initiator and contributor, as well as the structure of the tuples and the conditions.

$t_1 = \{KW(RSDB), \{range_{qid_n}(RSDB)\}, N_{rec}(RSDB)\}$,
$t_2 = \{KW(LDB), \{range_{qid_n}(LDB)\}, N_{rec}(LDB)\}$,
$t_3 = \{\{qid_n(RSDB), TreeURL_{qid_n}\}, k\}$,
$t_4 = \{qid_n(RSDB), node, range_{node}, k\}$,
$t_5 = \{k, StRSDB_{URL}\}, \{qid_n(RSDB), TreeURL_{qid_n}\}\}$,
$t_6 = \{PS, gnrlQID^N, Healthcare\,data\}$.

condition_1: $KW(RSDB) \subset KW(LDB), range_{qid_n}(RSDB) \subset \subset range_{qid_n}(LDB), n \in \overline{1,N}, N_{rec}(RSDB) \subset N_{rec}(LDB)$.

% initiator:

`reaction (out(`t_2`), in(`t_2`), out(`t_3`));`
`reaction (out(`t_4`),((in(`t_4`))`$\wedge$`(agreement is archived)), out(`t_5`));`
`reaction (out(`t_6`), in(`t_6`), ((update RSDB)`$\wedge$`(update StRSDB)).`

% contributor:

`reaction (out(`t_1`), condition_1, out(`t_2`));`
`reaction (out(`t_3`), ((in(`t_3`))`$\wedge$`(agreement is not achieved)), out(`t_4`));`
`reaction (out(`t_5`), ((in(`t_5`))`$\wedge$`(data are anonymized according to `t_5`)),out(`t_6`)).`

After writing the initial requirements the initiator will read and aggregate the information carried out by tuples of a type t_2, that have been received from different contributors. Based on the expert knowledge and data provided by other agents initiator will construct the initial version of the generalization binary trees for every qid_n.

[3]the conditions can be relaxed by verifying the intersection between the requirements for *RSDB* and *LDB* metadata and comparing with the threshold (e.g., if the threshold for the intersection between the keywords is set up to be tr, than the condition for the keywords can be modified as follows: $KW(RSDB) \cap KW(LDB) \geq tr$)

The next step is to achieve an agreement on the description of the generalization binary trees that correspond to the set of QID^N. The initiator of data collection creates an additional node in the network to store the metadata of *RSDB*, anonymity parameter k, and the files with the specification of the generalization trees. Next, it shares these data in the network by writing the tuple t_3. The other agents in the P2P network will be able to access this node and to suggest modifications (if any). It can be expressed by tuple t_4 by providing an alternative suggestion for the specification of the particular nodes or the anonymity parameter k.

When the agreement of the specification of the binary trees is reached, the initiator provides to contributors the updated generalization trees, the anonymity parameter k, and the link to the metadata of the initialized *RSDB*. (For the simplicity we assume that *RSDB* is initialized as $k-anonymous$ version of the *LDB*, which has the maximum number of records ($N_{rec}(LDB)$) that satisfy the requirements sent from initiator. $k-anonymous$ version is constructed by applying one of the microaggregation algorithms [10] locally.) After the initialization of *RSDB* all the contributors can update the database with any number of records using the anonymization algorithm presented in our previous work[6]. Insertion of the new record in *RSDB* will automatically cause the update of *StRSDB*.

4. IMPROVING UTILITY OF THE DATA

In this section we focus of the de-generalization problem: first, we define one-step de-generalization and we present a coordination mechanism to perform one-step de-generalization using cooperative agents. Second, we define the de-generalization problem for the general case.

4.1 One-step De-generalization

We define one-step de-generalization as a process of splitting the records with the same attributes into two groups such that the level of *generalization* of one of the attributes will be *decreased* by one, or, in other words, the level of *de-generalization* will be *increased* by one. (The level of de-generalization is shown bz the depth of the node to which the initial value of the attribute had been generalized, while the level of generalization = (height of the tree - level of de-generalization)). For instance, according to the binary trees shown on Figure 2 the records with the attribute age from 25 to 50 years old coded as $\{01;*\}$ can be de-generalized by splitting them into the groups with more fine-grained description of the attributes. For the example shown on Figure 3 with respect to the age we can consider forming two groups [25-38) and [38-50) coded as "010" and as "011" respectively, while the gender stays generalized the same way:

$$\{01;*\} \Rightarrow \{010;*\} \vee \{011;*\} \quad (1)$$

However, the following constrains need to be taken into account:

$$\begin{cases} k^p \geq k, \forall p \in \{0,1\}, \\ \sum_{p=0}^{1} k^p = K, \\ (\sum_{p=0}^{1} a_t^p = a_t). \end{cases} \quad (2)$$

The set of inequalities (2) requires the following. First, the numbers of records in both groups after splitting (k^0, k^1) should be greater then the parameter k required for $k-anonymity$. Second, the sum of the numbers of records in both groups after splitting should be equal to the number of records in the group before splitting (K). This can be guaranteed only by taking into account the numbers of records that came from different sources ($a_t, t \in \overline{1,T}$, where T - is a number of sources that contributed the records to the spitted group).

The solution is straightforward if there exists a node a_t, such that $(a_t^0 \geq k) \wedge (a_t^1 \geq k)$. In other words, if there exists a contributor (one of the *LDB*s) that provided to *RSDB* a certain amount of records ($\geq 2k$) such that the de-generalization of this group of records will result in having at least k records in both groups after splitting (first inequality of (2) is satisfied locally). Therefore, de-generalization of the records (with the same initial set of $gnrlQID_r^N$) provided by the other agents will not violate k-*anonymity* property of the *RSDB*.

For the case when one-step de-generalization is not possible locally, we propose the following coordination mechanism for collaborative privacy preserving one-step de-generalization.

1. Each agent sends $Enc(a_t^0)$, $Enc(a_t^1)$,
2. The evaluation of:
$$\left(Dec\left(\sum_{t=0}^{T} Enc(a_t^0)\right) \geq k\right) \bigwedge \left(Dec\left(\sum_{t=0}^{T} Enc(a_t^1)\right) \geq k\right) \quad (3)$$
is performed using de-generalization module (that implements the properties of the functional encryption scheme [1]) without knowing a_t^0 and a_t^1.
3. If (3) is `true`, it is possible to improve the utility of the data by one-step de-generalizing the records. To transmit the updated values the Shamir [16] secret sharing scheme is used. The agents will have to send one-step de-generalized values ($a_t^p, p \in \{0,1\}$) that are encrypted with key D, an access to which is "blocked" until all the agents provide their part of the secret ($\{D_t\}$). Those parts will be used to reconstruct the key D, to decrypt de-generalized values and update *RSDB* improving the utility of the data while maintaining anonymity property of the *RSDB*. Therefore, preserving privacy of the patients.

The tuples and coordination rules can be specified as follows.

$t_7 = \{gnrlQID^N, gnrlqid_n, range_{gnrlqid_n}\}$;
$t_8 = \{gnrlqid_n, Enc(a^0), Enc(a^1)\}$;
$t_9 = \{\{PS, gnrlqid+_n\}, D_t\}$.
% initiator:
`reaction (out(`t_8`), (3) is true, out(true));`
`reaction (out(`t_9`), ((in(`t_9`))`$\wedge$`(D is true)),`
`((update RSDB)`$\wedge$`(update StRSDB))).`
% contributor:
`reaction (out(`t_7`), in(`t_7`), out(`t_8`));`
`reaction (out({true}), in({true}), out(`t_9`)).`

Hereafter we analyze the possibility of the agent's misbehavior. The first case is the following: if not all agents sent the data at the Step 1 (encrypted one-step de-generalized values), it would only decrease a possibility to successfully evaluate (3) at the Step 2. However, it would not affect the privacy, as the data (a_t^0, a_t^1) are encrypted. The second possibility is if not all nodes sent the de-generalized data, then based on the properties of Shamir secret sharing scheme it would not be possible to decrypt the de-generalized data. Then as in the previous case, it is only not possible to de-generalize, but the privacy is preserved.

4.2 De-generalization Problem Definition

We want to maximize the utility of the data, which implies their de-generalization as much as possible. This means the maximization of two parameters. First, $L^N = \{L_n\}$, $n \in \overline{1,N}$, or, more precisely, every component of the vector that consists of the depths of the nodes that show the least generalized representation of the data in *RSDB*. Second, the number of possible splitting between the nodes whose depth is smaller then L_n, $n \in \overline{1,N}$. This will bring us to the maximization of $S^+ = \| \ gnrlQID+^N \ \|$-a set of different $gnrlQID+^N$, that represent the de-generalized versions of the quasi-identifiers, $gnrlQID^N$ with which the data were upload initially. Therefore, $gnrlQID+^N$ can only be chosen such that $gnrlqid_n$ is a prefix of $gnrlqid+_n$ for every $n \in \overline{1,N}$. In addition, we also have to take into account $(k,k^m) - anonymity$ requirements.

We can define the requirements and constraints described above in the following way:

$$\begin{cases} S^+ \text{ is max}, \\ k_{gnrlQID+^N} \geq k, \forall gnrlQID+^N \in S^+ \\ \sum_{j=1}^{S} k_{gnrlQID+_j^N} = k_{gnrlQID^N}. \end{cases} \quad (4)$$

In practice, quasi-identifiers that will be de-generalized can be chosen based on their importance that depends on the initial *RSDB* requirements specification.

5. CONCLUSION AND FUTURE WORK

In this paper, we showed how to build research databases taking into account certain requirements with respect to the purposes of a particular research question. In addition, we provide a possibility to constantly improve data utility with the databases growth in a privacy preserving way. To achieve this, we developed an architecture that is based on the coordination between cooperative agents organized in a peer-to-peer network. We also presented the advantages of using publish / subscribe paradigm and TuCSoN coordination model for the aggregation of population health data for the research purposes.

We analyzed the possible threats in the case of malicious behavior of the agents and showed that our approach preserves the patients privacy while improving the utility of the data aggregated for the research. We studied the problem of de-generalization and presented a solution for one-step de-generalization and defined the problem for a general case.

In the future work, we will continue working on the implementation of the MAS proposed in the paper and will evaluate our solution with a synthetic dataset and the patient data in the framework of the NanoTera project[4]. We also plan to extend the proposed model to solve the de-generalization problem in general case.

Acknowledgements

This work was supported by the Nano-Tera initiative, in the framework of an RTD project ISyPeM2: developing therapeutic drug monitoring by designing a point-of-care system to measure drug concentration in blood samples and adjust dosage accordingly.

REFERENCES

[1] M. Abdalla, F. Bourse, A. D. Caro, and D. Pointcheval. Simple functional encryption schemes for inner products. *IACR Cryptology ePrint Archive*, 2015:17, 2015.

[2] M. M. Baig, J. Li, J. Liu, and H. Wang. Cloning for privacy protection in multiple independent data publications. *Proceedings of the 20th ACM international conference on Information and knowledge management - CIKM '11*, page 885, 2011.

[3] A. Brugues De La Torre, M. Lluch-Ariet, and J. Pegueroles-Valles. Security analysis of a protocol based on multiagents systems for clinical data exchange. In *Complex, Intelligent, and Software Intensive Systems (CISIS), 2013*

[4] http://www.nano-tera.ch/projects/368.php

Seventh International Conference on, pages 305–311, July 2013.

[4] C. Clifton and W. Jiang. CERIAS Tech Report 2005-134 Information Assurance and Security Privacy-Preserving Distributed k -Anonymity. 2005.

[5] E. Denti, A. Natali, and A. Omicini. On the expressive power of a language for programming coordination media. In *Proceedings of the 1998 ACM symposium on Applied Computing*, pages 169–177. ACM, 1998.

[6] A. Dubovitskaya, V. Urovi, M. Vasirani, K. Aberer, and M. I. Schumacher. A cloud-based ehealth architecture for privacy preserving data integration. In *IFIP Advances in Information and Communication Technology*, SEC 2015. Springer Science and Business Media, 2015.

[7] B. S. Elger, J. Iavindrasana, L. Lo Iacono, H. Müller, N. Roduit, P. Summers, and J. Wright. Strategies for health data exchange for secondary, cross-institutional clinical research. *Computer methods and programs in biomedicine*, 99:230–251, 2010.

[8] A. Fuchs, M. Guidi, E. Giannoni, D. Werner, T. Buclin, N. Widmer, and C. Csajka. Population pharmacokinetic study of gentamicin in a large cohort of premature and term neonates. *British Journal of Clinical Pharmacology*, 78(5):1090–1101, 2014.

[9] D. Gelernter. Generative communication in linda. *ACM Trans. Program. Lang. Syst.*, 7(1):80–112, Jan. 1985.

[10] A. Gkoulalas-Divanis, G. Loukides, and J. Sun. Publishing data from electronic health records while preserving privacy: A survey of algorithms. *Journal of Biomedical Informatics*, 50:4–19, 2014.

[11] A. Gupta, O. Sahin, D. Agrawal, and A. El Abbadi. Meghdoot: Content-based publish/subscribe over p2p networks. In H.-A. Jacobsen, editor, *Middleware 2004*, volume 3231 of *Lecture Notes in Computer Science*, pages 254–273. Springer Berlin Heidelberg, 2004.

[12] N. Q. V. Hung, X. H. Luong, Z. Miklós, T. T. Quan, and K. Aberer. An MAS negotiation support tool for schema matching. In *International conference on Autonomous Agents and Multi-Agent Systems, AAMAS '13, Saint Paul, MN, USA, May 6-10, 2013*, pages 1391–1392, 2013.

[13] N. Q. V. Hung, D. S. Thanh, N. T. Tam, and K. Aberer. Privacy-preserving schema reuse. In *Database Systems for Advanced Applications - 19th International Conference, DASFAA 2014, Bali, Indonesia, April 21-24, 2014. Proceedings, Part II*, pages 234–250, 2014.

[14] E. Nardini, M. Viroli, and E. Panzavolta. Coordination in open and dynamic environments with tucson semantic tuple centres. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, SAC '10, pages 2037–2044, New York, NY, USA, 2010. ACM.

[15] G. Poulis, G. Loukides, A. Gkoulalas-Divanis, and S. Skiadopoulos. Anonymizing data with relational and transaction attributes. In H. Blockeel, K. Kersting, S. Nijssen, and F. Železný, editors, *Machine Learning and Knowledge Discovery in Databases*, volume 8190 of *Lecture Notes in Computer Science*, pages 353–369. Springer Berlin Heidelberg, 2013.

[16] A. Shamir. How to share a secret. *Commun. ACM*, 22(11):612–613, Nov. 1979.

[17] A. Solanas, A. Martinez-Balleste, and J. Mateo-Sanz. Distributed architecture with double-phase microaggregation for the private sharing of biomedical data in mobile health. *Information Forensics and Security, IEEE Transactions on*, 8(6):901–910, June 2013.

[18] V. Urovi, A. C. Olivieri, S. Bromuri, N. Fornara, and M. I. Schumacher. A peer to peer agent coordination framework for IHE based cross-community health record exchange. In *Proceedings of the 28th Annual ACM Symposium on Applied Computing, SAC '13, Coimbra, Portugal, March 18-22, 2013*, pages 1355–1362, 2013.

[19] V. Urovi, A. C. Olivieri, A. B. de la Torre, S. Bromuri, N. Fornara, and M. Schumacher. Secure p2p cross-community health record exchange in ihe compatible systems. *International Journal on Artificial Intelligence Tools*, 23(01):1440006, 2014.

[20] L. Xu and A. B. Cremers. A Decentralized Pseudonym Scheme for Cloud-based eHealth Systems. *HEALTHINF*, 2014.

[21] B. Zhou, Y. Han, J. Pei, B. Jiang, Y. Tao, and Y. Jia. Continuous privacy preserving publishing of data streams. In *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*, EDBT '09, pages 648–659, New York, NY, USA, 2009. ACM.