A Semantic Approach for Semi-Automatic Detection of Sensitive Data

J. Akoka, I. Comyn-Wattiau, H. Fadili, N. Lammari, E. Métais, C. du Mouza

and Samira Si-Saïd Cherfi - Lab. CEDRIC, CNAM Paris, France

Context and Problem

Context

- to test and validate new applications developers need realistic data
- final tests generally performed on excerpts from the on-going production databases
- recent phenomenon of the externalization of any development and test

Problem

- information in many databases is proprietary and must be protected
- existing proposals lack an automatic detection of the sensitive data

Motivating examples

Hospital database

- [data] all personal and medical information about patients
- [risk] any person developing an application on the medical data not to be able to extract any personal information about a patient

Clients database

- [data] all information and coordinates of the different clients of a large company
- [risk] a leak of information can cause considerable business damage if transmitted to a competitor

Our Proposal

Main features of our approach

- Automatic detection of the values to be scrambled
- Automatic propagation to other semantically linked values

Techniques used

- a rule based approach implemented under an Expert System architecture
- a semantic graph to ensure the propagation of the confidentiality and the consistency with the other relations

Sensitive data

Confidential attributes

The confidential attributes set, denoted $\mathcal{S}_c \subseteq \mathcal{S}$ is the set of attributes whose content is confidential, whatever the number of occurrences they have.

Identifying attributes

The identity attributes set, denoted $\mathcal{S}_i \subseteq \mathcal{S}$ is the set of attributes such that for any $x \in \mathcal{S}_i$ it exists a subset $s_i \subseteq \mathcal{S}_i$ within a single table $\mathcal{T}$ and with $x \in s_i$, such that:
(i) each instance of s_i occurs less than k times in the records from $\mathcal{T}$
(ii) there is an attribute $y \in \mathcal{S}_c$ in $\mathcal{T}$.

Sensitive attributes

The sensitive attributes set, denoted $\mathcal{S}_s$, is the set of identifying and confidential attributes, *i.e.*, $\mathcal{S}_s = \mathcal{S}_i \cup \mathcal{S}_c$.

Sensitive data

Confidential attributes

The confidential attributes set, denoted $\mathcal{S}_c \subseteq \mathcal{S}$ is the set of attributes whose content is confidential, whatever the number of occurrences they have.

Identifying attributes

The identity attributes set, denoted $\mathcal{S}_i \subseteq \mathcal{S}$ is the set of attributes such that for any $x \in \mathcal{S}_i$ it exists a subset $s_i \subseteq \mathcal{S}_i$ within a single table $\mathcal{T}$ and with $x \in s_i$, such that:
(i) each instance of s_i occurs less than k times in the records from $\mathcal{T}$
(ii) there is an attribute $y \in \mathcal{S}_c$ in $\mathcal{T}$.

Sensitive attributes

The sensitive attributes set, denoted $\mathcal{S}_s$, is the set of identifying and confidential attributes, *i.e.*, $\mathcal{S}_s = \mathcal{S}_i \cup \mathcal{S}_c$.

Sensitive data

Confidential attributes

The confidential attributes set, denoted $\mathcal{S}_c \subseteq \mathcal{S}$ is the set of attributes whose content is confidential, whatever the number of occurrences they have.

Identifying attributes

The identity attributes set, denoted $\mathcal{S}_i \subseteq \mathcal{S}$ is the set of attributes such that for any $x \in \mathcal{S}_i$ it exists a subset $s_i \subseteq \mathcal{S}_i$ within a single table $\mathcal{T}$ and with $x \in s_i$, such that:
(i) each instance of s_i occurs less than k times in the records from $\mathcal{T}$
(ii) there is an attribute $y \in \mathcal{S}_c$ in $\mathcal{T}$.

Sensitive attributes

The sensitive attributes set, denoted $\mathcal{S}_s$, is the set of identifying and confidential attributes, *i.e.*, $\mathcal{S}_s = \mathcal{S}_i \cup \mathcal{S}_c$.

Why considering all sensitive attributes?

We observe that:

- The scrambling of the identity attributes preserves anonymity while confidential attributes keep their initial distribution.
- The scrambling of the confidential attributes aims at protecting individual privacy by modifying the value of confidential attributes while information that identifies persons remains unchanged.

Example

A HRD database storing information concerning employees: employee's id, name, city, department, name of the superior, wage, etc.

- the first two properties permit to identify an employee ($\mathcal{S}_i = \{id, name\}$), and thus to access all his data
- one may avoid to reveal the highest salary or the average salary of a given department $\Rightarrow$ considered as sensitive ($\mathcal{S}_c = \{wage\}$).
- in smaller companies, the couple (city,department) is sufficient to identify a small subset of employees $\Rightarrow$ must be added to $\mathcal{S}_i$. For larger companies this information is not identifying enough.
- finally for our large company we have to scramble $\mathcal{S}_s = \{id, name, wage\}$.

The rule-based approach

Let Δ be the set of all possible domains of application, Θ the set of all possible table names, Φ the set of all possible attribute names and Ψ the set of all possible attribute values.

Rule condition

A *rule condition* $\chi = \chi_1 \boxplus \chi_2$ is a condition with $\chi_1 \in \{\mathit{domainName}, \mathit{tableName}, \mathit{attributeName}, \mathit{attributeValue}\}$, $\chi_2 \in \Delta \cup \Theta \cup \Phi \cup \Psi$, and $\boxplus$ is an operator in $\{=, !=, <, >, \leq, \geq, \mathit{contains}, !\mathit{contains}\}$.

Rule

A *rule* is composed by disjunctions and conjonctions of rule conditions along a rule sensitivity score $\sigma \in [0, 1]$, where σ permits to evaluate how sensitive is an attribute that satisfies the rule.

Rule example

Assume we consider that a column whose name contains “salar” if the domain is HRD and there are values greater than 15,000 or lower than 5,000 is highly sensitive (score=0.9). The corresponding rule is expressed by the following expression:

$$
\begin{aligned}
&((\mathit{domainName} =' \mathit{HRD}') \\
&\wedge(\mathit{attributeName}\ \mathit{contains}\ '\mathit{salar}') \\
&\wedge(\mathit{attributeValue} > 15000 \\
&\vee \mathit{attributeValue} < 5000)) \ , \ 0.9
\end{aligned}
$$

Other detection techniques

The statistical computation

Some candidates for $\mathcal{S}_i$ can be found thanks to:

- metabase (primary key and unique integrity constraints)
- statistics, generally stored in the metabase for query optimization purpose
- but determining all the subsets of attributes that are quasi-identifiers is a *NP*-hard problem (but heuristics)

Necessity of Natural Language Processing

- attributes may not have been named with exactly the same word that the one used in the rules
- matching using NLP techniques (currently only a semantic matching based on Wordnet)

Propagation graph

Integrity and referential links

- $\rightarrow$ foreign key attribute references a primary or secondary key attribute $\Rightarrow$ any modification of the former must impact the latter
- $\rightarrow$ same problem with attribute in a table with same semantics than another one in another table

We build for any set $P \subseteq \mathcal{S}$, the result set of links

$$\Gamma(P) = \bigcup_{x \in 2^{|P|}} \gamma(x)$$

where $\gamma : 2^{|\mathcal{S}|} \rightarrow 2^{|\mathcal{S}|}$ is defined as

$$\forall x \in 2^{|\mathcal{S}|}, \gamma(x) = \begin{cases} \{y \mid y \in 2^{|\mathcal{S}|},\ y \textit{ referring or semantically linked to } x\} \\ \emptyset \quad \textit{otherwise} \end{cases}$$

Propagation algorithm

We use the referential and semantical links between attributes to extend the set of attributes $\mathcal{S}_s^{init}$ identified for scrambling:

Propagation algorithm

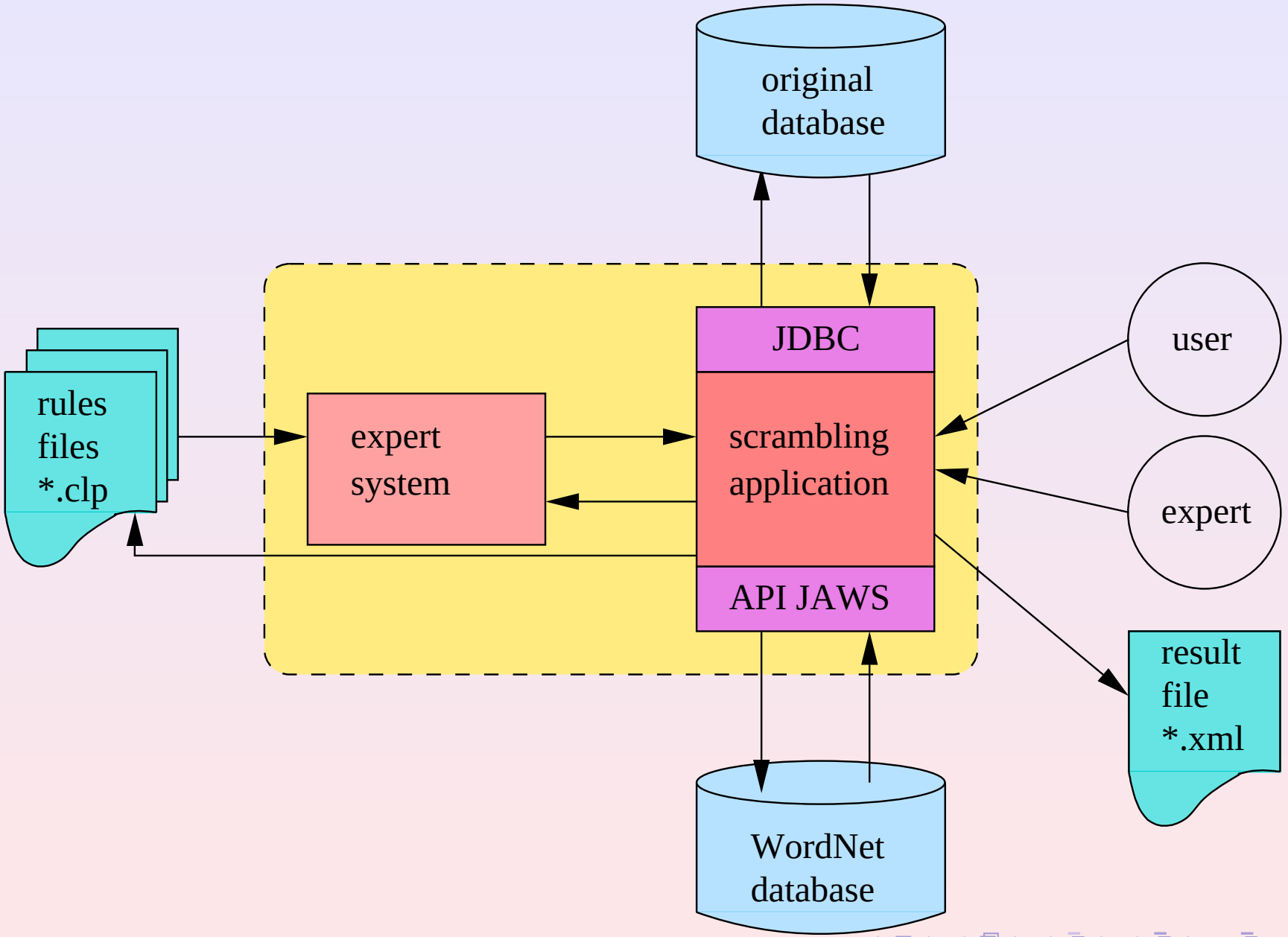
(*i*) $\mathcal{S}_s^{(0)} = \mathcal{S}_s^{init}$

(*ii*) $\mathcal{S}_s^{(k+1)} = \mathcal{S}_s^{(k)} \cup \Gamma(\mathcal{S}^{(k)})$

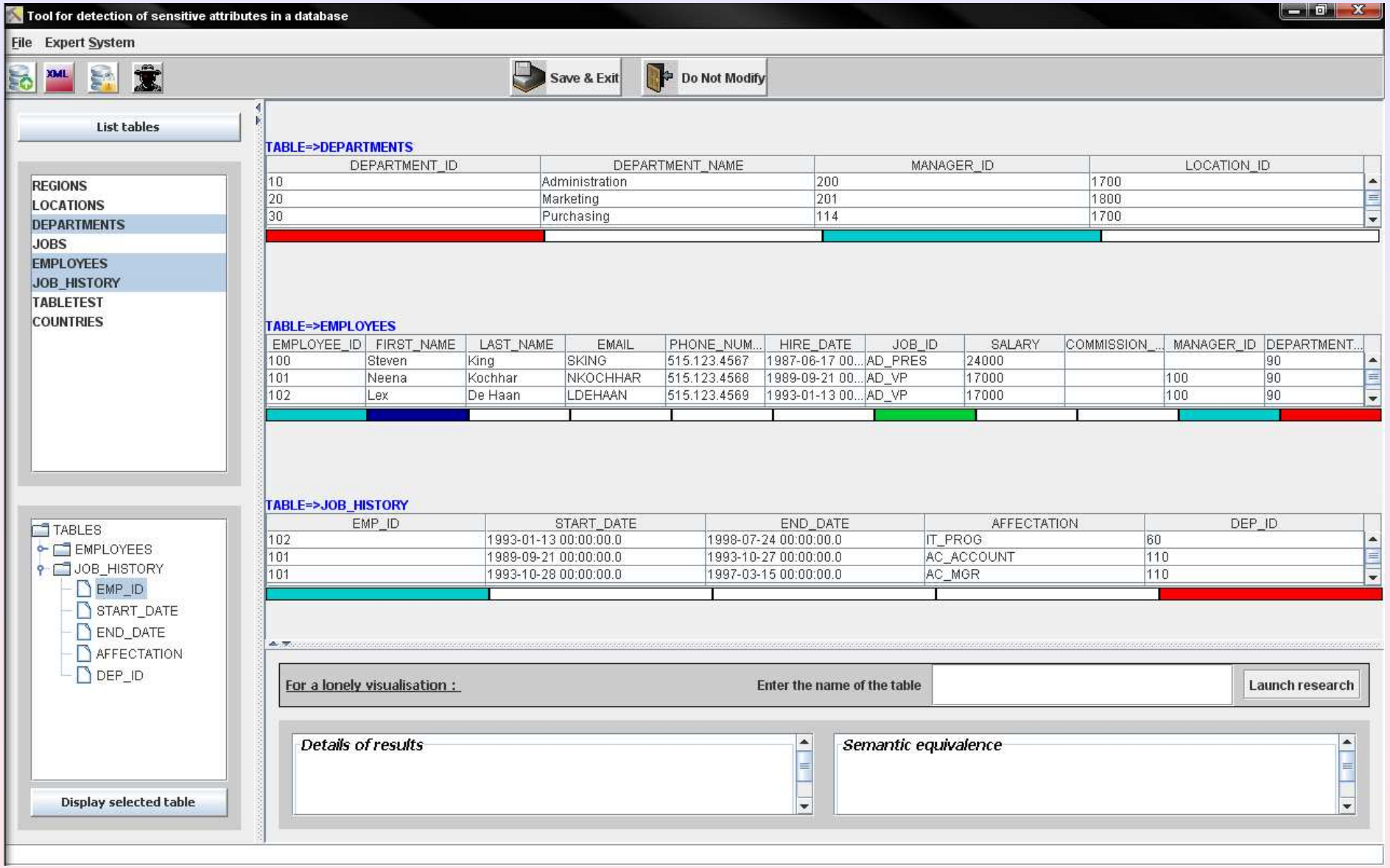
Lemma (convergence)

The algorithm converges to $\mathcal{S}_s$ with at most $|\mathcal{S}|$ iterations.

Prototype architecture



Prototype interface



Conclusion

Our proposal

- a rule-based approach for determining the attribute's sensitivity level
- integrity referential constraints and semantic links are used for the propagation of the sensitivity

Future work

- development of the NLP techniques
- automatically determining of the scrambling algorithms to use on sensitive data
- validation on real databases thanks to experts

Thanks for your attention