Miminum Exposure in Classification Scenarios

Nicolas Anciaux*,**
*INRIA Paris-Rocquencourt
Domaine de Voluceau
Bat. 9, SMIS team
78153 Le Chesnay, France
+33 1 39 63 56 35
Nicolas.Anciaux@inria.fr

Benjamin Nguyen*,**
**Université de Versailles St-Quentin- En-Yvelines (UVSQ)
45, Avenue des Etats-Unis
78035 Versailles, France
+33 1 39 25 40 48
benjamin.nguyen@uvsq.fr

Michalis Vazirgiannis***
***Athens U. of Economics & Business
Patision Street, Athens, Greece
and LIX, Ecole Polytechnique,
91128, Palaiseau, France
+30 210 8203 519
mvazirg@aueb.gr

ABSTRACT

Personal information about applicants is often requested by service providers to be used as an input of a classification process establishing the specific situation of each applicant. This is a prerequisite for the service provider to make an appropriate offer to the applicant. For example, the rate and duration of personal loans are usually adapted depending on the risk based on the income, the assets or past lines of credits of the borrower. In the eyes of privacy laws and directives, the set of exposed documents collected to achieve a service must be restricted to the minimum necessary. This *Limited Data Collection* principle reduces the impact of data breaches both in the interest of service providers and customers. In this article, we show that in practice, the data collected traditionally is excessive. We propose a new approach that we call *Minimum Exposure*, where the minimum set of documents required can be computed on the user's side. We formalize the underlying problem and show it is NP hard. We propose algorithms to compute a solution and validate them with experiments. The *Minimum Exposure* approach leads to a very significant reduction of the quantity of personal information exposed, therefore leading to important privacy gains for the applicant and large scale savings for service providers in the event of data breaches.

1. INTRODUCTION

A massive digitalization of personal information is currently underway. Individuals are receiving an ever increasing amount of important documents in digital form (financial, professional, medical, relative to insurance, administrative, linked to daily consumption, etc.), issued by their employers, banks, insurances companies, civil authorities, hospitals, schools, ISP, telcos, etc.

In parallel, secured online personal stores are emerging, like Adminium[1] or Securibox[2]. The domain of the personal cloud is flourishing, and a recent report forecasts a $12 billion market[3]. Alternative offers propose storage facilities on the user's side with extended privacy controls, like for example Plug servers[4], Personal Data Servers [6], Nori[5], or Personal Data Ecosystem[6]. This thriving market

[1] See http://www.adminium.fr/
[2] See http://www.securibox.fr/
[3] The Personal Cloud: Transforming Personal Computing, Mobile, And Web Markets, Frank Gillett, a Forrester report, June 2011.
[4] See http://freedomboxfoundation.org/
[5] See http://www.projectnori.org/
[6] See http://personaldataecosystem.org/

attests a reality: official documents are continuously accumulated and treasured by their owner. The reason is simple: legal obligations require them to be kept (e.g., 1 year for bank statements) and these documents are used as evidence when performing subsequent administrative tasks (e.g., paying taxes) or applying to services (e.g., applying for a loan).

Indeed, many services are calibrated to adapt to the particular situation of each applicant. For example, the characteristics of a personal loan (rate, duration, insurance fee…) are defined according to proofs of income, employment, title deeds, personal references, forms of collateral, medical records, past lines of credits, etc. To cite other examples, contracting an insurance (health, car, job protection, etc.), social assistance or tax refund, also require providing evidence of one's specific situation.

Although privacy intrusive, the necessity of evaluating the particular situation of an applicant is unquestionable and is in the interest of both the service provider and the customer. However, the requested set of documents must be restricted to the minimum required to take the correct decision. First, the reason is to protect the privacy of the applicant. Privacy legislations and directives worldwide enacted the *Limited Data Collection* (*LDC*) principle to this end: this principle states that organizations should only collect the personal data strictly required to achieve a goal the user consents to [17], [31], [36]. The second reason is to limit the cost of information leakage. Indeed, all too often personal data ends up being disclosed by negligence or hack. Since the beginning of 2011, the Open Security Foundation[7] has already reported 322 data loss incidents affecting 126 millions records. The Privacy Rights Clearinghouse[8] has tracked 275 data breaches with a total of more than 20 million records involved. This is not only a serious privacy incident, but also a potential financial disaster for the companies in charge of the data. A recent study [33] estimates the cost of data breaches for US companies at an average $7.2million per incident and it has kept increasing since 2006. In addition, The New York Times reports that 90% of companies have experienced at least one data breach last year[9]. Moreover, data breach laws have been adopted in many countries[10]. They compel companies to notify data owners in the event of data breaches, assist the victims in minimizing the impact of the data leak (e.g., canceling their credit card if the number has been disclosed) and often incur financial compensations. Security companies have created online breach cost calculators[11] to draw attention to this phenomenon: the more data exposed, the greater the cost in the event of a data breach.

The target of this paper is precisely to restrict the set of documents to expose to third parties to a minimum subset, in accordance with the *LDC* principle. Existing works have already transposed this legal principle to data management systems. Hippocratic databases maintain the set of attributes that are required for each purpose [3]. However, this solution assumes that the data useful and useless to reach a given purpose can be distinguished *a priori* (at collection time). This assumption holds for simple cases, e.g., when ordering online, the address of the customer is mandatory to deliver the purchased items. However, it does not hold in many cases. What data is useful to come

---

[7] See http://www.datalossdb.org/

[8] See http://www.privacyrights.org/

[9] The New York Times BITS. June 22, 2011. By Riva Richmond. Security Professionals Say Network Breaches Are Rampant.

[10] In EU (European Parliament legislative resolution, 6 May 2009), in 46 states in US (http://www.ncsl.org/issues-research/telecommunications-information-technology/security-breach-notification-laws.aspx), etc.

[11] See http://databreachcalculator.com.sapin.arvixe.com/

to the decision of lowering the rate of the loan proposed to a user? Not only does the information harvesting depend on the purpose, it also depends on the data itself. Consider a reduction of rate based on either the salary or the assets of an individual. Revealing her income of *$30.000* if her age is below *25* may be enough. But an income of *$50.000* would suffice, regardless of age. Maybe *both* the income and age values are useless if sufficient assets (e.g., greater than *$100.000*) can be justified. For a user with values u_1=[*income=$35.000*, *age=21*, *assets=$10.000*] the minimum data set would be [*income*, *age*]. For a user with u_2=[*income=$40.000*, *age=35*, *assets=$250.000*] it would be [*assets*]. Hence, the bank cannot specify a minimum set of attributes needed to make its decision since this decision depends on looking at the entire attributes available. Therefore, fixing the data to be collected *a priori* inevitably leads to over-estimating the data to be collected, so as to cover all the information which *may turn out to be of use* at some point in the decision process.

This illustrates what we call the *limited data collection paradox* expressed as follows: *third parties require users to reveal data in order to determine whether this data is required to achieve the expected purpose*. To the best of our knowledge, all the existing techniques implementing *LDC* do not escape falling into this paradox. We further position our work with respect to previous studies in Section 6.

In this paper, we propose a strategy to strictly comply with the *LDC* principle, while alleviating the paradox and solving the above problems. Our solution is based on a novel concept, that we call *Minimum Exposure* which is a reverse implementation of the traditional *LDC* strategy, where individuals are given enough knowledge to determine on their side the minimum set of data to expose to achieve the expected service with maximum benefit.

Our contribution is threefold. First, we formalize the *Minimum Exposure* optimization problem. Second, we study the complexity of the problem, prove it is NP-Hard and discuss approximation algorithms. Third, we validate our proposal with experiments illustrating important reductions of the information to be exposed.

The paper is organized as follows. Section 2 gives the general scenario, and presents a running example used along the paper. In Section 3, we state the *Minimum Exposure* optimization problem, prove it is NP-Hard and study its complexity. Several algorithms are introduced in Section 4, and validated in Section 5. Section 6 discusses the related work and Section 7 concludes the paper.

2. SCENARIO FOR MINIMUM EXPOSURE

2.1 General Scenario

We consider the general scenario depicted in Figure 1 which involves three main parties: Data Producers, Users, and Service Providers. **Data Producers** act as data sources. They include for example banks, employers, hospitals, or administrations. The information they deliver to users has an official value and is signed to prove integrity and origin (e.g., salary forms, bank records history, tax receipts, etc.). **Users** store the documents they receive in their personal digital spaces. We make here no hypothesis on users' personal space, which could be their own PC, cloud storage, secure personal devices, etc. **Service Providers** may include banks or insurances companies, but also public welfares or administrations. They propose customized services like bank loans, health insurance, social benefits, etc.

We call *Minimum Exposure* (*ME*) the process which identifies the minimum subset of documents produced by Data Producers to be exposed by a User to a Service Provider to trigger the desired service with the (set of) advantage(s) she can (and wants) to obtain. *ME* requires confronting (1) the set of documents owned by the user, with (2) the advantages, associated with *collection rules* describing the information requested by the service provider.

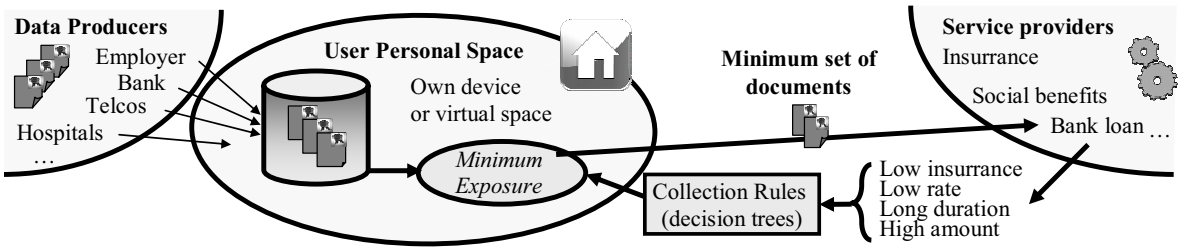

Figure 1. General architecture enabling Minimum Exposure.

The execution of *ME* must take place on the user's side or on any third party trusted by the user, to escape the *LDC* paradox: indeed, the system in charge of running *ME* –including the service provider itself– would need to collect more documents than the minimum subset computed by *ME*.

The general scenario is thus as follows. When a user wants to obtain a service, she (1) downloads collection rules published by the service provider, (2) computes locally the advantages she can obtain based on the documents she owns, (3) selects among the advantages the ones she desires to obtain, (4) uses *ME* to compute the minimum set of documents to expose to obtain the service with the selected advantages, (5) exposes these documents to the service provider, where their integrity and origin are checked.

2.2 Setting

2.2.1 Collection Rules

The collection rules describe the information required by the service provider and the advantages associated with it. Collection rules must (i) be expressive enough to successfully reflect the decision making process of the service provider. Indeed, decision making rules are complex in practice, e.g., loans are granted based on decision trees, SVM or neural networks [16]. Collection rules must also (ii) be comprehensible by humans (end users have to check their appropriateness). Note that comprehensibility and justifiability are often required by law, for e.g., credit scoring and computer-aided medical diagnosis [23]. Rules extraction tools have been designed to obtain comprehensive rules (e.g., modeled as decision trees) from "black box" decision making models like SVM or neural networks [10].

In this paper, we consider that collection rules are modeled using disjunctions of conjunctions of constraints on attribute-value pairs. This is a comprehensible rule-based model, which is sufficiently expressive since it covers the widely used decision tree model [26].

For example, an organization offering loans, e.g., grant *Non Interest Loans* (NILo) to families and young students, could articulate this with the following rule:

$$NILo: (married=true \wedge children>0) \vee (age<30 \wedge Edu='Univ')$$

We assume that no-one can force users to transmit documents. The only penalty is to prevent them from obtaining advantages. Therefore, rules must be *positive*, in the sense that it is beneficial for a user to trigger them. This is not a limitation of the model since rules leading to constraints that prevent the grant of services (called *negative* rules) can be constructed by integrating the negation of the rule into the collection rule set. For example, if the *NILo* mentioned above is *not* granted to people with a police record ($police\_record='YES' \Rightarrow \neg NILo$), the rule can be written:

$$NILo: (married=true \wedge children>0 \wedge police\_record='NO') \vee (age<30 \wedge Edu='Univ' \wedge police\_record='NO')$$

2.2.2 Users Documents

The granularity at which data is produced and signed by the data producers has a strong impact on the quality of the result of the *ME* process. In general, each official document is transmitted and signed as a *whole document*. The *ME* algorithm will then view each document as a set of inseparable (attribute, value) pairs, that would all be exposed –or not– to a service provider.

If documents are signed at a finer granularity by data producers, the *ME* algorithm can process (attribute, value) pairs separately, and expose the value of a given attribute, e.g., the annual income, without revealing the whole official document, e.g., the income tax receipt, containing this value. Although users' documents are currently signed as a whole (in today's applications there is no need for finer granularity), there is no technical difficulty to sign (attribute, value) pairs separately, avoiding artificial complexity. In this paper, we focus on signed (attribute, value) pairs.

To go further, data producers could even sign expressions of the form *attribute* θ *value*, θ being the comparator $<$, $\leq$, $=$, $\neq$, $\geq$, or $>$. The *ME* algorithm could thus manipulate *attribute* θ *value* triples and avoid exposing the precise values of each attribute. The approach we propose when considering signed (attribute, value) pairs can be extended to such a context (see Appendix A).

2.2.3 Metrics to Evaluate the Degree of Exposure

The minimization of the set of documents resulting from the application of the *ME* algorithm can be appreciated in terms of reduction of the data exposed, harmful to both user (in terms of privacy) and service provider (in terms of financial cost).

We consider on the one hand that the privacy harm associated with a dataset is proportional to the usefulness of that dataset, and on the other hand that the cost of a data breach for service providers is directly proportional to its exposure. The financial cost for service providers is determined by two dominant factors [33]. First, the *ex-post response* represents 20% of the cost. It includes the actions taken by the company to provide assistance to the victims in the necessary procedures conducted to minimize the harm: the greater the exposure, the greater the harm. Second, *lost business* (50% of data breach cost) is the direct consequence of the negative publicity associated with the data breach incident headings: the greater the exposure, the worse the publicity. These components of the breach costs are tightly linked with information loss, and as such are captured by the metric we have chosen.

The techniques that we propose can accommodate any metric that associates an exposure value to each dataset item independently. This covers many traditional information loss metrics based on data entropy, e.g., *minimal distortion* [34], [35] or *ILoss* [38].

2.3 Running Example

We introduce here the loan scenario, used as a running example (see Table 1) throughout the paper. The example has deliberately been simplified, real consumer loan applications often requesting hundreds of personal data items[12].

An institution proposes, to any applicant, personal loans of *$5.000* at *10%* rate with *1* year duration and a *$50* per month insurance cost for job loss protection. But, a higher loan of *$10.000* can be offered to wealthy customers fulfilling the following requirement:

(*income>$30.000 and assets>$100.000*)
or (*collateral>$50.000 and life_insurance='yes'*)

Table 1. Collection rules and documents for the Loan Scenario

Collection rules:		
	r_1: $(p_1 \wedge p_2) \vee (p_3 \wedge p_4)$	$\Rightarrow c_1$
	r_2: $(p_5 \wedge p_6 \wedge p_7) \vee (p_4 \wedge p_8 \wedge p_9)$	$\Rightarrow c_2$
	r_3: $(p_1 \wedge p_6 \wedge p_7) \vee (p_2 \wedge p_4 \wedge p_{10})$	$\Rightarrow c_3$
	r_4: $(p_2 \wedge p_5 \wedge p_6 \wedge p_7) \vee (p_1 \wedge p_4 \wedge p_8 \wedge p_9)$	$\Rightarrow c_4$
with	p_1: *year_income>$30.000,*	p_2: *assets>$100.000,*
	p_3: *collateral>$50.000,*	p_4: *life_insurance='yes',*
	p_5: *tax_rate>10%,*	p_6: *married=true,*
	p_7: *children>0,*	p_8: *edu='university',*
	p_9: *age<30,*	p_{10}: *insurance_claims<$5.000.*
and	c_1=*high_loan,*	c_2=*5%_rate,*
	c_3=*long_loan,*	c_4=*low_insurance.*
User's documents:		
	doc_1: *year_income=$35.000,*	doc_2: *assets=$150.000,*
	doc_3: *collateral=$75.000,*	doc_4: *life_insurance='yes',*
	doc_5: *tax_rate=11.5%,*	doc_6: *married=true,*
	doc_7: *children=1,*	doc_8: *edu='univ',*
	doc_9: *age=25,*	doc_{10}: *insurance_claims=$250.*

This leads to the first collection rule r_1 given in Table 1. Collection rule r_2 enables the obtaining of a loan granted at only *5%* rate for families and low risk factor young people; collection rule r_3 expresses that loans can be granted for an extended duration of *2* years to high revenues families and to low risk people; and rule r_4 states that the insurance cost for job loss protection can be proposed with a *30%* discount to rich families and promising young workers. Each collection rule is made of a disjunction of conjunction of predicates p_i of the form *attribute* θ *value*, with θ the comparator $<$, $\leq$, $=$, $\neq$, $\geq$, or $>$. We consider a user owning a set of *attribute* = *value* documents

[12] HSBC loans applications are good representatives:
https://applymort.us.hsbc.com/secure/application/mortgage_equity.hus

termed doc_1 to doc_{10} such that[13] $doc_i \Rightarrow p_i$. This user could then activate the complete set of advantages c_1 to c_4. The *ME* algorithm has to identify the minimum set of documents allowing this.

3. THE *MINIMUM EXPOSURE* PROBLEM

This section first states the *Minimum Exposure* problem more formally and studies its complexity.

3.1 Problem Statement

We denote by $|S|$ the cardinality of a set S. We introduce below the other required definitions, and then state the problem. We illustrate the notions using our running example (see in Table 1).

3.1.1 Definitions

Attributes. Let $A = \{a_i\}$ represent a finite set of attributes. Each attribute a_i has an associated domain $dom(a_i)$.

Example: A={*year_income, assets, collateral, life_insurance, tax_rate, married, children, edu, age, insurance_claims*} and *dom*(*year_income*)=[0;∞], *dom*(*married*)=Boolean, etc.

Classes. Let $C = \{c_j\}$ represent a finite set of Boolean variables, interpreted as *positive* classes to which users can belong. If c_j=*true* for a given user, this means she can obtain the advantage associated with c_j.

Example: C={*high_loan, 5%_rate, long_loan, low_insurance*}. If c_1=*true* for a given user, then this means the user can benefit from receiving a *high_loan.*

Predicates. We call *predicate over A* any expression of the form $a\theta v$ where $a \in A$, $v \in dom(a)$ and $\theta \in \{=, <, >, \leq, \geq, \neq\}$.

Example: p_1*: year_income>$30.000* is a predicate. doc_6*: married=true* is also a predicate.

Signed documents. Let doc_i represent a signed document containing a single equality predicate *attribute = value* over A. We denote by $data_u=\{doc_i\}$ the set of signed documents a given user u can expose (i.e., the user already possesses those signed documents or can request them from a data producer). We say that a signed document doc_i *proves* a predicate p if $doc_i \Rightarrow p$.

Example: doc_1*: year_income=$35.000* is the signed document which proves predicate p_1.

Atomic Rules. An atomic rule leading to class c_j, denoted by $atom_j$ is a conjunction of predicates such that $atom_j$=*true* $\Rightarrow c_j$=*true.* Since there are usually several atomic rules leading to a class c_j we write $atom_{j,k}$ using k to distinguish them.

Example: $atom_{1,1}$: (*year_income>$30.000* ∧ *assets>$100.000*) and $atom_{1,2}$: (*collateral>$50.000* ∧ *life_insurance='yes'*) are two atomic rules leading to class c_1.

---

[13] We use implication rather than entailment because (although unlikely) the service provider may decide to give a user a benefit without proof.

We say that a set of signed documents $data_u=\{doc_i\}$ proves an atomic rule $atom_{j,k} = \wedge_m \ q_{j,k,m}$ where $q_{j,k,m}$ is a predicate over A, if and only if $\forall j,k,m \ \exists \ i : doc_i \Rightarrow q_{j,k,m}$ and *uniquely proves* $atom_{j,k}$ if and only if $\forall j,k,m \ \exists! \ i : doc_i \Rightarrow q_{j,k,m}$.

Example: $data_u=\{doc_1, doc_2, doc_3, doc_3\}$ uniquely proves atomic rules $atom_{1,1}$ and $atom_{1,2}$.

Collection Rules. A collection rule r_j is a disjunction of atomic rules leading to class c_j. More formally: $r_j: \vee_k atom_{j,k}$. If a signed set of documents $data_u$ proves an atomic rule $atom_{j,k}$ then we say that $data_u$ *proves* r_j, which means that user u can benefit from the advantage associated with c_j (obviously, $r_j=true \Rightarrow c_j=true$).

Example: r_1: (*year_income>$30.000* $\wedge$ *assets>$100.000*) $\vee$ (*collateral>$50.000* $\wedge$ *life_insurance='yes'*) is a collection rule leading to class *high_loan*.

In what follows, we write $r_j = \vee_k (\wedge_m \ q_{j,k,m})$ where $q_{j,k,m}$ is a predicate over A. Considering r_1 in the previous example, we have $q_{1,1,1}$: *year_income>$30.000*, $q_{1,1,2}$: *assets>$100.000*, $q_{1,2,1}$: *collateral>$50.000* and $q_{1,2,2}$: *life_insurance='yes'*.

Rule Set. Let $R = \{r_j\}$ represent a set of $|C|$ *collection rules*, one for each class c_j. If $data_u$ (uniquely) proves all the rules of R then we say that $data_u$ (uniquely) proves R.

Example: The four collection rules of Table 1 form a rule set.

Rule Set Boolean Formula. Since only one document uniquely proves a given predicate used in the rules, deciding whether $data_u$ proves the rule set R is *equivalent* to testing the truth-value of a Boolean formula E_R called Rule Set Boolean Formula associated to R constructed as follows:

$$E_R = \wedge_j (\vee_k (\wedge_m b_{f(j,k,m)}))$$

where $f(j,k,m)$ is a function of domain $[1;|A|]$ defined by $[f(j,k,m)=i$ such that $doc_i \Rightarrow q_{j,k,m}]$ and $b_{f(j,k,m)}$ is a Boolean variable which is *true* if $doc_{f(j,k,m)}$ is exposed and *false* otherwise. Note that if we consider the truth assignment that sets all the values $b_{f(j,k,m)}$ to *true*, then $E_R=true \Leftrightarrow data_u$ *proves* R.

Example: Table 2(a) illustrates a Rule Set Boolean Formula based on R defined in Table 1.

Exposure metric. Let $B=\{b_i\}$ represent a set of Boolean variables. Let T_B represent a truth assignment of these variables such that $b_i=true \Leftrightarrow$ the signed document doc_i is published. We note $\mathbf{EX}(T_B)$ a function representing the exposure of the associated published document set. Exposure is proportional to financial cost for service providers, and privacy harm for the user.

We use $\mathbf{EX_{card}}(T_B) = |\{ \ b_i \in B: T_B(b_i)=true \ \}|$, a simple function that counts the number of documents exposed, as exposure metric for the rest of the article. Note that information loss metrics can be assumed proportional to $\mathbf{EX_{card}}$.

Table 2(a). Rule Set Boolean Formula for the loan scenario

$B = \{b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8, b_9, b_{10}\}$ such that:
$\forall i \in [1;10]$, $b_i = true \Leftrightarrow$ document doc_i *is exposed.*
The Rule Set Boolean Formula E_R is as follows:

$E_R = (b_1 \wedge b_2) \vee (b_3 \wedge b_4)) \wedge ((b_5 \wedge b_6 \wedge b_7) \vee (b_4 \wedge b_8 \wedge b_9)) \wedge ((b_1 \wedge b_6 \wedge b_7) \vee (b_2 \wedge b_4 \wedge b_{10})) \wedge ((b_2 \wedge b_5 \wedge b_6 \wedge b_7) \vee (b_1 \wedge b_4 \wedge b_8 \wedge b_9))$

Suppose that the user owns in fact documents 1-9 only we prune out all the classes and atomic rules that can not be proven:

$E_R = (b_1 \wedge b_2) \vee (b_3 \wedge b_4)) \wedge ((b_5 \wedge b_6 \wedge b_7) \vee (b_4 \wedge b_8 \wedge b_9)) \wedge (b_1 \wedge b_6 \wedge b_7) \wedge ((b_2 \wedge b_5 \wedge b_6 \wedge b_7) \vee (b_1 \wedge b_4 \wedge b_8 \wedge b_9))$

Table 2(b). Algorithm notations using the loan scenario

D=10; C=4;
B[] is an array of Booleans of size D such that:
$\forall i \in [1;10]$, *B[i]=true* $\Leftrightarrow$ document d_i is *exposed*
R[] is an array of C collection rules;
R[j].atom[] for $j \in [1;4]$ are arrays of *2* atomic rules;
R[j].atom[k].b[] with $j \in [1;4]$, $k \in [1;2]$ are arrays of references to *B[i]* elements. We denote by **B[i]* a reference to *B[i]*. *R[j].atom[k].b[m]*are set as follows:
R[1].atom[1].b[1]←**B[1]*;
R[1].atom[1].b[2]←**B[2]*;
(…) *R[2].atom[2].b[2]*←**B[8]*;
R[2].atom[2].b[3]←**B[9]*; (…)
R[4].atom[2].b[3]←**B[8]*;
R[4].atom[2].b[4]←**B[9]*;

3.1.2 Minimum Exposure Problem

Using these definitions and notations, we can now define the *Minimum Exposure* decision problem of a set of documents $data_u$ with regards to a rule set R and an exposure metric **EX**. Note that with no loss of generality, we suppose that $data_u$ proves R. Should this not be the case, we would simply use R' the subset of rules of R proven by $data_u$. Our goal is to find a truth assignment T_B of the Boolean variables associated to the publication of the documents minimizing their exposure computed using **EX**.

PROBLEM 1. *Boolean Minimum Exposure (ME) decision problem*:

Given a rule set R, $data_u=\{doc_i\}$ a set of q signed documents that **uniquely proves** R, B a set of Boolean variables $B=\{b_1,..,b_q\}$ such that $b_i=true \Leftrightarrow doc_i$ is exposed, $E_R=\wedge_j (\vee_k (\wedge_m b_{j,k,m}))$ where $\forall j,k,m\ b_{j,k,m} \in B$ the rule set formula associated to R, and the exposure function **EX,** $data_u$ is *n-exposable* with regards to R if and only if there exists a truth assignment T_B of B such that $\mathbf{EX}(T_B) \leq n$ and E_R is *true*.

Example: Considering the Rule Set R and the set of signed documents $data_u$ defined in Table 1, we see that $data_u$ proves R. Moreover, $data_u$ is *5*-exposable with regards to R and exposure metric $\mathbf{EX_{card}}$ since the truth assignment $T_B=\{b_1$=T, b_2=T, b_3=F, b_4=F, b_5=T, b_6=T, b_7=T, b_8=F, b_9=F, b_{10}=F$\}$ satisfies E_R and $\mathbf{EX_{card}}(T_B)$=5.

We study the *ME* optimization problem which consists in finding the smallest n for which a set of signed documents is n-exposable with regards to given rule set R and exposure function $\mathbf{EX_{card}}$.

3.2 Complexity Results

3.2.1 Complexity Analysis

We analyze here the complexity and approximability of the *ME* problem. Our results concern both polynomial time approximation algorithms [32] and differential approximation [19][14].

THEOREM 1.

All Positive Min Weighted SAT decision (resp. optimization) problem is reducible to ME decision (resp. optimization) problem.

PROOF. The *Min Weighted Sat* decision (resp. optimization) problem is defined in [5] as follows:

"Given an integer n, an instance $\{P_{j,k}\}$ of P Boolean variables, a CNF formula $F=\wedge_j (\vee_k P_{j,k}))$ over $\{P_{j,k}\}$ and a (positive) weight function w:$\{P_{j,k}\}\rightarrow$R+, find a truth assignment T for $\{P_{j,k}\}$ that satisfies F such that w(T)= $\Sigma_{j,k}$ w($P_{j,k}$)×T($P_{j,k}$) is $<n$ (resp. is minimum)."

When the formula contains no negative variables, the problem is called *All Positive Min Weighted Sat* (*APMWS*). The *ME* decision (resp. optimization) problem considers a formula $E=\wedge_j (\vee_k (\wedge_m b_{j,k,m}))$. An *APMWS* instance can be mapped to a *ME* decision (resp. optimization) instance by choosing $\forall j,k$: m=1 and $b_{j,k,1}=P_{j,k}$ (i.e., all the atomic rules contain only one predicate). Any solution to the *ME* decision (resp. optimization) problem (i.e., find a minimum truth assignment to $b_{j,k,1}$ such that E_R is *true*) will be a solution for *APMWS* by choosing $P_{j,k}=b_{j,k,1}$■

COROLLARY 1.

ME decision problem is NP-Complete.

PROOF. Given THEOREM 1, since *APMWS* is NP-Complete, the result is immediate.

COROLLARY 2.

ME optimization problem is NP-Hard, is not in APX[15]*, and has a differential approximation ratio of 0-DAPX*[16].

PROOF. Complexity results are a direct consequence of the fact that the *APMWS* optimization problem is reducible to the *ME* optimization problem, and has negative complexity results. In [5], Theorem 6 states that the *APMWS* problem is not in APX, which leads to the fact that *ME* is not in APX either. [19] studies the *APMWS* problem, but from a differential approximation angle, and shows that the problem is of class 0-DAPX as defined in [11]. Therefore the *ME* optimization problem is in 0-DAPX. ■

3.2.2 Solving the ME Problem

---

[14] Given an instance I of an optimization problem, and a feasible solution S of I, we denote $m(I,S)$ the value of solution S, $opt(I)$ the value of an optimal solution of I and $W(I)$ the value of a worst solution of I. The differential approximation ratio of S is defined by $DR(I,S)=abs((m(I,S)-W(S))/(opt(I)-W(I))$. The traditional approximation ratio for a minimization problem is simply defined by $m(I,S)/opt(I)$.

[15] APX is the class of NP optimization problems that allow polynomial-time approximation algorithms with an approximation ratio bounded by a constant.

[16] 0-DAPX is the class of NP optimisation problems for which all polynomial approximation algorithms have a differential approximation ratio of 0.

COROLLARY 2 is a negative complexity result in the sense that it shows that the problem is difficult and that polynomial approximation algorithms will provide bad approximation guarantees in the worst case. In Section 4, we examine the problem by (experimentally) exploring the domain where it is possible to provide an exact resolution using a state of the art solver. When such a resolution is not possible (too long to compute), we rely on polynomial approximation algorithms.

3.3 Problem Generalization

We have considered in Section 3.1 the case where data producers can provide the user with documents made of a single *attribute = value* predicate (signed at this granularity). In a more general setting the problem can be expressed as follows, using any exposure metric $\mathbf{EX_{DOC}}$ defined on a set of documents:

PROBLEM 2. *Generalized Minimum Exposure decision problem*

Given a rule set R, $data_u=\{doc_i\}$ a set of signed documents of the form $doc_i=a\theta v$ where $a\in A$, $v\in dom(a)$ and $\theta\in\{=, <, >, \leq, \geq, \neq\}$ that **proves** a rule set R, and an exposure function $\mathbf{EX_{DOC}}$. We say that $data_u$ is *n-exposable* with regards to R if we can find $d_{min}\subset data_u$ such that d_{min} proves R and $\mathbf{EX_{DOC}}(d_{min})\leq n$.

In such a setting, each predicate in the rule set can potentially be proven by several documents (e.g., a user might have a document claiming *salary>$1.000* and another claiming *salary>$3.000*). We say that we have a *mismatch* between collection predicates and documents. This general problem and some ideas towards its resolution are presented in Appendix A.

4. SOLUTIONS OF THE *ME* PROBLEM

In this section, we provide exact and approximation algorithms to compute a solution of the *ME* problem. For the exact resolution, we use a *Binary Integer Programming* (BIP) state of the art solver. For the approximate resolution, we propose a naïve random algorithm, a simulated annealing based meta-heuristics algorithm, and a *ME* specific heuristic algorithm.

In all algorithms, we consider a Boolean formula E_R constructed as explained in Section 3 using a rule set R composed of a set of C collection rules associated with classes (or benefits) that the user can (and wants) to claim, and where each atomic rule can be proven using her documents (atomic rules that cannot be proven are pruned before constructing R). The size of $data_u$, the user document set related to the rule set, is denoted by D.

T_B is a truth assignment function to $data_u$ that we implement as an array of Booleans with the semantics $B[i]=true \Leftrightarrow doc_i$ *is exposed*. The rule set is represented as an array *R[]* of C collection rules, each collection rule *R[i]* being an array *atom[]* of atomic rules, each atomic rule *R[i].atom[j]* being an array *b[]* of references to the elements of B (see example in Table 2(b)). Note that E_R is *true* when each collection rule *R[i]* has at least one atomic rule where all referenced Boolean elements are *true*.

4.1 Exact Resolution (BIP model)

We propose to use a state of the art BIP solver, generally termed as *Mixed Integer Non-Linear Program* (MINLP) solver, to produce an exact result. We have chosen the popular and open source *COUENNE* solver [12] to this respect.

In order to use a MINLP solver, an instance of the problem must be written as a MINLP program. This is a direct transformation where each document corresponds to a Boolean variable, where the objective function is simply the sum of all the variables, and in which we express one *non-linear* constraint per collection rule r_j:

$$r_j: \Sigma_k \Pi_m \, doc_{j,k,m} \geq 1$$

The running example presented in Section 2.3 can be expressed by the following program, written in *AMPL* [21]:

```
var b1 binary; ... var b10 binary;
minimize EXCARD:
b1+b2+b3+b4+b5+b6+b7+b8+b9+b10;
subject to
r1: b1*b2 + b3*b4 >= 1;
r2: b5*b6*b7 + b4*b8*b9 >= 1;
r3: b1*b6*b7 + b2*b4*b10 >= 1;
r4: b2*b5*b6*b7 + b1*b4*b8*b9 >= 1;
```

The program is then fed to the BIP solver. As shown in Section 5, the range of parameters for which the BIP solver computes the solution in an acceptable time (under *10* minutes) is small.

4.2 Approximate Solutions (Polynomial Time)

We need to revert to a polynomial time approximation in order to compute results for the instances of the problem that cannot be tackled within reasonable time by the solver. We propose three algorithms: a naïve fully random algorithm called *RAND**, a simulated annealing meta-heuristics based algorithm called *SA**, and an algorithm called *HME* using a heuristic specially designed for the *ME* problem. These algorithms are non deterministic, therefore they can be run many times and the best solution is kept. However, they produce their first result in linear or polynomial time, depending on the algorithm. We discuss the complexity of the algorithms on a single run, to compare their speed. To compare their quality, we run the longest algorithm (*HME*) once, and we execute the other algorithms (*RAND** and *SA**) as many times as necessary, until they run out of processor time.

4.2.1 Fully Random Algorithm

We first introduce the *RAND** algorithm (see Algorithm 1), based on a random choice of rules. *RAND** serves as a baseline to be compared with the (smarter) algorithms presented next.

*RAND** randomly chooses one atomic rule for each collection rule and sets to *true* the value of each Boolean in *B* that this atomic rules refers to. Since each class is covered, the corresponding set of documents determined by the truth assignment *B* is a solution to the *ME* problem instance. The result is the solution found within the allocated time limit for which $\mathbf{EX_{card}}$ is minimum (best

result). The algorithm complexity is O($C \times \text{MAX}_{\text{atom}}(atom.length) + D$) where $\text{MAX}_{\text{atom}}(atom.length)$ is the length of the longest atomic rule involved. This algorithm therefore provides an approximate solution in polynomial (linear) time. Since this algorithm is straightforward, we do not provide its code.

4.2.2 Simulated Annealing Algorithm

Meta-heuristics are used in optimization problems in order to guide the algorithm towards better solutions, instead of simply randomly selecting them. We consider here simulated annealing [24] and introduce the *SA** algorithm to serve as a representative for meta-heuristic guided algorithms.

A first parameter function of the meta-heuristic is *neighbour*(*B*) which randomly chooses one class and randomly chooses a different atomic rule for this class[17]. For the second parameter function, as proposed in [24], we define $temp(i)=0.9^{i} \times 10$ and $P(E,E_{new},T)=1$ if $E_{new} < E$, and $exp((E-E_{new}) / T)$ otherwise. We use $\mathbf{EX_{card}}$ to have an energy function in the same range as the one proposed for the Traveling Salesman Problem.

SA* **(Simulated Annealing) algorithm**
Input: *R* a Rule set
N a number of runs
M a number of cooling iterations
Ouput: B_{best} a truth assignment of the documents that proves *R*

```
1.    for i = 1 to D do
2.       B_best[i] ← true
3.    endfor
4.    for i=1 to N do
5.          B ← SA(M, RAND*(R, 1))
6.          if EX_card(B) < EX_card(B_best) then
7.                B_best←B
8.    endfor
9.    return S_best
```

As in the case of *RAND** algorithm, *SA** algorithm provides a solution in polynomial (linear) time of complexity O(*M*+*D*), disregarding the initial initialization phase that uses *RAND**.

[17] This means maintaining for a truth assignment *B* the atomic rules that compose it, in order to easily perform the switch.

Function SA
Input: M a number of cooling iterations
B_0 an initial truth assignment
Ouput: B_{best} a truth assignment of the documents that proves R

```
1.   B ← B0
2.   Bbest ← B0
3.   E ← EXcard(B)*D
4.   Ebest ← E
5.   for i= 1 to M do
6.        Bnew ← neighbour(B)
7.        Enew ← EXcard(Bnew)*D
8.        if P(E, Enew, temp(i)) > random() then
9.             B ← Bnew
10.            E ← Enew
11.            if Enew < Ebest then
12.                 Bbest←Bnew
13.                 Ebest ← Enew
14.            endif
15.       endif
16.  endfor
17.  return Bbest
```

4.2.3 The HME Algorithm

The Heuristic for Minimum Exposure (*HME*) algorithm that we propose uses a specific heuristic for the *ME* problem. We show how it works on an example, and we discuss its complexity.

The heuristic lies in the computation of *score[i]* the score of the i^{th} Boolean entry in B, using the function *fix*(B). This function computes a lower bound of the value of $\mathbf{EX_{card}}$, by computing the number of predicates that can no longer be set to *false* for the given B. For instance, suppose that *B[i]=false* (i.e., doc_i is not published). All the atomic rules referring to *B[i]* cannot be proven anymore. This leads to the fact that $\mathbf{EX_{card}}$ will be greater (or equal) to the value of the cardinality of the set of predicates in the atomic rules that are the only ones left to prove a given class. Using the running example (see Section 2.3), we illustrate the computation of *fix(B)* in Table 3 for each Boolean entry at each step of the algorithm. Let us briefly see how *score[1]* and *score[3]* are computed for the first step. If *B[1]=false*, then we have to prove collection rules *R[1]*, *R[3]*, *R[4]* using respectively atomic rules *R[1].atom[2]*, *R[3].atom[2]*, *R[4].atom[1]* (i.e., which means setting to *true* the *7* Booleans *B[2]*, *B[3]*, *B[4]*, *B[5]*, *B[6]*, *B[7], B[10]*), leading to *score[1]=7*. If *B[3]=false*, this means proving *R[1]* using *R[1].atom[1]* (i.e., set to *true* the *2* Booleans *B[1]*, *B[2]*), therefore *score[3]=2*. We show in grey the lowest score, which means a truth assignment set to *false* in next steps, indicated by the symbol –. Documents for which the score is denoted by ∞ are those for which the final truth assignment is set to *true*. The final result is here *B=[B[1]=true*,

B[2]=true, *B[3]=false*, *B[4]=false*, *B[5]=true*, *B[6]=true*, *B[7]=true*, *B[8]=false*, *B[9]=false*, *B[10]=false]* which happens to be the minimal value of $\mathbf{EX}_{\mathbf{card}}$ on this instance of the problem.

***HME* algorithm**
Input: E_R a Rule Set Boolean Formula
Ouput: *B* a truth assignment of the documents that proves *R*

```
1.for i = 1 to D do
2.  B[i] ← true
3.endfor
4.while (          exists i such that: B[i]=true and
          if B[i] is set to false then E_R (B) remains true) do
5.  for i = 1 to D do
6.    score[i] ← ∞
7.  endfor
8.  forall i such that B[i] = true do
9.    B[i] ← false
10.       if E_R (B)=true then // E_R (B) is true iff B proves R
11.             score[i] ← fix(B)
12.       endif
13.       B[i] ← true
14.     endforall
15.     m ← i such that score[i] is minimum
16.     B[m] ← false
17.   endwhile
18.   return B
```

Table 3. Execution of the *HME* algorithm

Steps	*B[1]*	*B[2]*	*B[3]*	*B[4]*	*B[5]*	*B[6]*	*B[7]*	*B[8]*	*B[9]*	*B[10]*
1: score[i]	7	7	**2**	5	4	6	6	4	4	3
2: score [i]	∞	∞	–	5	5	6	6	5	5	**4**
3: score [i]	∞	∞	–	**5**	7	∞	∞	5	5	–
4: score [i]	∞	∞	–	–	∞	∞	∞	**5**	5	–
5: score [i]	∞	∞	–	–	∞	∞	∞	–	**5**	–
Final B[i]	*true*	*true*	*false*	*false*	*true*	*true*	*true*	*false*	*false*	*false*

We see that the cost of *HME* algorithm is proportional to $O(\mathrm{COST_{FIX}} \times \mathrm{D}^2)$, where $\mathrm{COST_{FIX}}$ is the cost of computing the *fix* function. More precisely, in our implementation, $COST_{FIX} = O(C \times d_R \times d_{QD})$,

where d_R is the number of atomic rules per collection rule and d_{QD} is the number of predicates per atomic rule (see Section 5 for more precisions concerning these notations).

The intuition behind the heuristic is to successively get rid of the documents which require keeping the least number of other documents (such that all benefits are preserved) among the remaining ones. This heuristic is particularly relevant when the number of atoms per collection rule is small. Our performance evaluation confirms this scope. Note that if this number increases then *HME* tends towards *RAND*.

We show in Section 5 that the *HME* algorithm provides very good results in terms of quality of approximation, while maintaining reasonable computational complexity. We show in Appendix A that the *HME* algorithm can be easily extended to documents containing predicates of the form *attribute* θ *value*, with $\theta \in \{<, \leq, =, \neq, \geq, >\}$.

5. EXPERIMENTS

In this section, we present an experimental validation of our approach. Its objective is twofold: (1) to show that the gain on the exposure metric is significant when using exact algorithms or approximation algorithms; (2) to show that the approach is scalable in the sense that approximation algorithms still achieve good gains when exact algorithms become too costly. We first present the parameters of the experiments. We then provide the results of our experiments and draw the main conclusions. Algorithms, data and BIP model generator code are available[18].

5.1 Experimental Setup

5.1.1 Platform, Metrics and Algorithms

Experiments were conducted on a HP workstation with 3.1MHz Intel CPU and 8GB RAM running Java1.6 (x64). The *COUENNE* solver was run on the same machine. We consider the algorithms given in Table 4 (details in Section 4).

Table 4. Algorithms used in the experiments

COUENNE	Computes the exact solution using the COUENNE solver, within a time limit (2 hours).
*RAND**	Selects the best random solution found within a time limit (same execution time as *HME*).
*SA**	Finds a solution using simulated annealing within a time limit (same execution time as *HME*).
HME	Selects the best solution using an ad-hoc heuristic.

The exposure metric that we use is $\mathbf{EX_{card}}$ as defined in Section 3.1.1 In the figures that follow, we measure the exposure reduction obtained using that metric, representing the percentile reduction of the number of documents exposed compared with classical server-side limited collection techniques

[18] See http://www-smis.inria.fr/~anciaux/MinExp

(i.e., where *all* the documents involved in the collection rules are exposed). Exposure reduction is denoted ***ER*** and is computed by:

$$\boldsymbol{ER}(T_B)=1-\mathbf{EX}_{\mathbf{card}}(T_B)/|B|$$

5.1.2 Synthetic Problem Generator

Privacy related data sets are inherently scarce to obtain. Thus we chose to build a tunable generator to produce many possible problems (user data and rule sets). To fix the parameters around realistic values, we used the characteristics of real decision trees obtained from [10] in the case of credit risk assessment. These decision trees were extracted from neural networks built on real credit risk datasets. To model the data generated used as input of the algorithms, we introduce a bipartite graph representation. We discuss below its parameters.

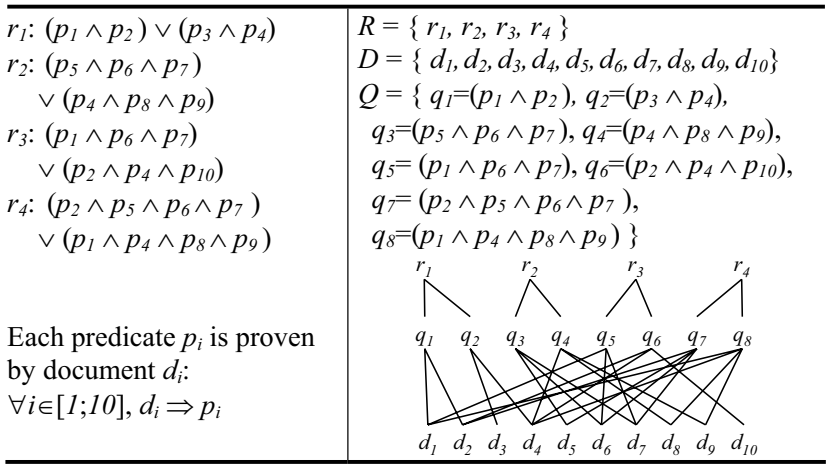

Figure 2. Loan scenario (left) and Bipartite graph (right)

We introduce three sets of nodes: $R=\{r_i\}$ the set of (provable) collection rules where r_i implies class c_i, $Q=\{q_i\}$ the set of (provable) atomic rules involved in the collection rules and $D=\{d_i\}$ the set of documents held by the user which can prove all the predicates involved in the rules. We also introduce two sets of edges. E_{QR} is the set of edges between Q and R. Each vertex in E_{QR} is interpreted as the fact that if the atomic rule is *true* (i.e., proven), the collection rule to which it belongs is also *true*. E_{QD} is the set of edges between Q and D. Each edge in E_{QD} is interpreted as the fact that if the document is *false* (i.e., not exposed), the atomic rule to which it belongs is also *false*. We have $D\cap Q=\varnothing$, $R\cap Q=\varnothing$ and $R\cap D=\varnothing$, and we note $G_{D,Q,R}=(D\cup R, Q, E_{QD}\cup E_{QR})$ which forms a bipartite graph. Our running example is illustrated using the bipartite graph representation in Figure 2.

To define a graph, we need to fix the average out degrees of nodes in D and R, noted d_D (the average number of atomic rules a document belongs to) and d_R (the average number of atomic rules in a collection rule), and the average out degrees of nodes in Q towards each subset of partition $D\cup R$ noted d_{QD} (the average number of document predicates in an atomic rule) and d_{QR} (the average

number of collection rules an atomic rule leads to). In our representation, we fix $d_{QR}=1$, i.e., an atomic rule leads to only one class. Note that this still captures the case in which two "identical" atomic rules lead to two different classes: those two atomic rules will simply be represented as two different nodes in Q, having the same set of document predicates in D and leading to different classes in R.

Considering our notations, the following relations hold:

(1) $|E_{QD}| = |D| \times d_D = |Q| \times d_{QD}$

(2) $|E_{QR}| = |Q| = |R| \times d_R$

We see that by fixing the quadruplet ($|D|$, d_D, $|R|$, d_R) we can uniquely determine $|Q|$ and d_{QD}. We call such a quadruplet the generator of a dataset. While it would be possible to choose a different quadruplet, we chose these attributes because we feel their semantics are easier to understand. For instance the dataset shown in Figure 2 consists of a set of $|R|=4$ collection rules, built over $|Q|=8$ atomic rules with a total of $|D|=10$ different documents, with an average of $d_D=2.4$ atomic rules per document and $d_R=2$ atomic rules per class. The generator of this dataset is therefore (10, 2.4, 4, 2). Both $|Q|=8$ and $d_{QD}=3$ can be computed using equations (1) and (2).

Based on these considerations we can build a dataset generator algorithm. This algorithm takes as input a quadruplet ($|D|$, d_D, $|R|$, d_R). Using these parameters, it constructs D and R, then deduces Q since $|Q|=|R| \times d_R$. It constructs E_{QD} and E_{QR} by generating the out degrees of nodes based on a given distribution (Gaussian) and randomly picks a value in this distribution. It then insures that all nodes are connected, by switching edges if necessary. The result is a dataset conforming to the generator with a given distribution of out degrees.

5.2 Measurements

We run three sets of experiments. Based on the real decision trees built in [10], we consider the following topology of the problem in all our experiments: we fix d_R =4 (a collection rule is composed of 4 atomic rules in the average) and d_D =4 (each document is involved on average in 4 atomic rules) and vary $|D|$ and/or $|R|$. Each measure is the average of several repeated experiments to reduce statistical bias (each measure is repeated *100* times except those lasting longer than *1* minute, where we ran them *10* times). The experiments are:

Experiment 1: increasing documents only. We vary the number of documents necessary to prove the predicates involved in the rule set and we fix all other parameters. We consider *10* collection rules. Regarding the input parameters of the data generator, we vary $|D|$ and fix $|R|=10$, $d_R=4$ and $d_D=4$. *COUENNE* is given a time limit of *2* hours. *RAND** and *SA** are given processor time equivalent to one execution of *HME* on the same instance. Note that execution time of approximation algorithms is always less than *10* minutes here. Results are presented in Figure 3 (shows the exposure reduction *ER*) and 4 (shows the execution time).

Experiment 2: increasing collection rules only. We vary the number of collection rules and we consider *1000* documents, i.e., the fixed parameters are $|D|=1000$, $d_R=4$ and $d_D=4$. In this case, in order to be able to scale $|R|$ we need to choose a sufficiently large value for $|D|$. Again, *RAND** and *SA** are given the same processor time as one execution of *HME* (always less than *10* minutes). Within *2* hours, *COUENNE* was not able to produce any result. Results are pictured in Figure 5.

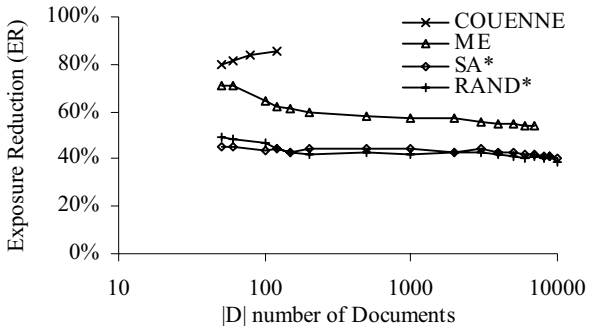

Figure 3. ER varying the number of documents

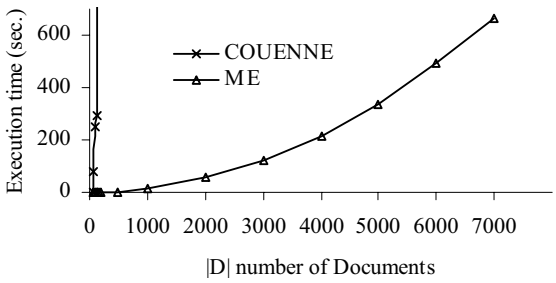

Figure 4. Execution time varying number of documents

Experiment 3: increasing documents and collection rules. We vary the numbers of collection rules and documents, keeping a constant ratio of $|D|/|R|=4$. We fix $d_R=4$ and $d_D=4$. This measure shows the behavior of the algorithms when the dataset increases, with a stable problem topology. Figure 6 plots the results.

The main conclusions that we draw from the experiments are:

The exposure reduction ER is (almost always) important. The algorithms, depending on the parameters of the problem, enable a reduction of the number of documents to be exposed ranging from *30%* to *80%* compared with traditional implementations of the limited collection principle. In the area of applicability of exact solutions, the reduction is of nearly *70%* on average. Of course, the expected reduction may vary depending on the input dataset, but to give a rough idea of the results that can be expected, the average exposure reduction in all the experiments (considering for each measure the result of the best algorithm) is around *50%*.

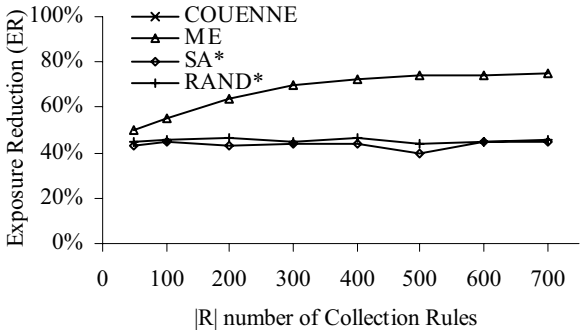

Figure 5. ER varying number of collection rules

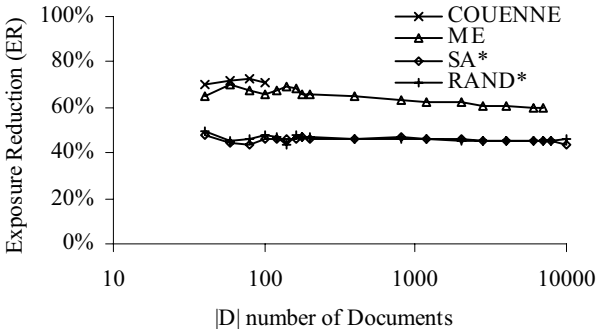

Figure 6. ER varying number of documents & collection rules

The scope of exact solution is limited. As expected, *COUENNE* which computes an exact solution consumes far too much time when the size of the problem increases. Within the time limit that was chosen, the result cannot be computed with more than *120* documents in *Experiment 1*, *1* collection rule in *Experiment 2*, and *15* collection rules in *Experiment 3*. Note that giving a time limit bigger

than *2* hours would not change much since the execution time grows exponentially with the size of the problem. With *200* documents in *Experiment 1*, and with *2* collection rules in *Experiment 2*, *COUENNE* needs nearly *10* hours. The use of approximation algorithms is therefore unavoidable in order to obtain results for a broad spectrum of parameters.

HME is the best approximation algorithm. The exposure reduction obtained using *HME* is higher than those obtained with *RAND** or *SA**. Results presented in Figures 3, 5 and 6 show that the *HME* algorithm outperforms the *RAND** and *SA** algorithms in this range of parameters, with a significant 10% relative gain.

To conclude, implementing the limited data collection principle using *Minimum Exposure* provides very significant gains in terms of reduction of the exposed documents and is scalable.

6. RELATED WORK

The transposition of legal privacy principles into privacy aware computing systems has fostered many studies during the last decade. Emblematic examples include the P3P Platform for Privacy Preferences [15], the emergence of privacy policy languages like EPAL [9] and Hippocratic databases [3]. P3P transposes the well known *need-to-know* and *consent* principles. Web sites can describe their practices in a machine readable format, which are automatically compared by the web browser to the consent given by the user. While P3P highlights conflicting policies, it offers no means to calibrate the data exposed by a user to a given service. In the last years, many other policy languages have been proposed for different application scenarios, including EPAL [9], XACML [28] and WSPL [7]. For example, WSPL (Web Services Policy Language) aims at describing and controlling various features of web services. To the best of our knowledge, no language has been introduced with limited data collection in mind. But studying possible support of minimum exposure with such languages is part of our future work.

Hippocratic databases [3] are another emblematic example of privacy aware systems. They are inspired by the axiom that databases should be responsible for the privacy preservation of the data they manage. The architecture of a Hippocratic database is based on ten guiding principles derived from privacy laws including limited data collection. However, limited data collection is implemented on server side and by nature falls into the paradox presented in the introduction. Interestingly, open problems associated with *LDC* are mentioned in [3] but left unsolved. The authors point out for example the problem of implementing the limited data collection principle whether attributes are needed (or not) depending on the values of other attributes. Our proposal helps resolving those problems.

Recent privacy studies enhance access control policies to bridge the gap with privacy policies. In particular, obligations are introduced [29], as well as actions to perform after the data has been obtained like notification or removal [8] and purpose binding features [14], [25]. Recently, the widely used RBAC model has been extended to support privacy policies [30]. A natural distinction holds between *ME* and access control. Indeed, in the context of *ME*, potential third parties (a bank or a social organization) are not known in advance, they can potentially be many, and the minimum subset of personal data to expose must be defined according to the claimed benefits and the personal data at disposal (the more documents at disposal, the more alternatives to obtain the desired benefits). This goes far beyond the scope of traditional access control techniques. Existing works closer to our study can be found in the area of credential based access control, where access

decisions are based on the confrontation of an access control policy with a set of credentials. In this area, both the policy and credentials are private, and thus most contributions use secure multiparty computing techniques to reach the decision, and therefore do not scale. A few number of works including [39, 40, 41] in this area can however be considered as following a minimum exposure approach. All those works minimize the privacy leak of a set of personal data items (credentials) while enabling a given decision to be made (the grant or deny access decision). Those contributions can be seen as vanguards in the strict application of the LDC principle for decision making systems. However, the problem and solutions are different for two founding reasons. First, the decision making processes that we consider are more complex than access control. The collection rules in ME can model sets of decision trees classifiers: several dimensions can be considered (e.g., lower credit rate, longer duration, lower cost of insurance, larger portion of 0% loan, etc.) each one potentially impacting the final offer made to the applicant. Second, in our context, the decision making process requires by nature a huge amounts of personal data (e.g., to obtain a loan offer customers are asked to fill in forms with hundreds to thousands fields), while in access control only a few credentials are considered (e.g., up to 35 in [39]). The results of these works can therefore not be used in our context, because they fall short on both the expressivity of the problem, and its scalability requirement.

Works dealing with Privacy Preserving Data Mining (PPDM) also take a different direction than *ME*. Recent PPDM surveys [2], [20] refer neither to *ME* type problems nor to their legal foundation (i.e., the *LDC* principle). Unlike *ME*, techniques protecting individual records with regards to the input of a data mining algorithm [4], [26], turn original data into encrypted or randomly perturbed data, which becomes unverifiable. On the contrary, *ME* preserves the original data and its ability to be verified by a third party (a signature guarantees its integrity and origin). Another aspect of PPDM techniques is that they try to protect sensitive rules (i.e., the output of a data mining algorithm) by removing raw data [1], [37]. However, these techniques *maximize* the information retained in the output data set, so long as the private results remain secret, whereas the goal of *ME* is to *minimize* it. Note that this approach is nevertheless compatible with *ME*. Indeed, the former (PPDM) would remove sensitive data upstream and the latter (*ME*) could minimize the remaining information, thereby guaranteeing maximal privacy.

Privacy Preserving Data Publishing (PPDP) [22] are also closely related to the *ME* problem. Indeed, PPDP focuses on publishing original raw data rather than data mining results or statistics. However, subsequent treatments are not known at the time of data publishing. In *ME*, the knowledge of these treatments is a prerequisite to identify the minimum subset of data to be exposed. Furthermore, PPDP tries to balance privacy gain and data utility, sometimes with difficulties [13], while *ME* preserves the full utility of the data (complete set of due benefits are obtained). Some PPDP techniques like [18], closer to statistical databases can be assimilated to (advanced) access control, where statistical data is exposed without revealing individual values.

7. CONCLUSION AND FURTURE WORKS

In this article, we have introduced the *Minimum Exposure* approach and the related *ME* problem. We have shown how it can be expressed in the form of a Boolean minimum weighted satisfiability problem. We have studied the scope of applicability of general operational research solutions, using a state of the art MINLP solver. For cases where an exact resolution was not applicable, we have proposed several algorithms to compute an approximation of the solution. In all cases, we have shown that the exposure reduction that can be achieved compared with traditional implementations of limited data collection is around 50% in the average. These benefits are not only interesting for the user, whose privacy is less exposed, but also for the service providers who can limit their losses in the event of a data breach.

Our hope is to open a new avenue for interesting applications of the minimal exposure principle introduced in this paper. For example, collective treatments, e.g., data mining tasks or distributed queries, could be addressed to achieve minimal exposure constraints. The result could be computed progressively, each participant would dynamically determine a minimum set of local documents remaining to be delivered in order to compute the final result, given some feedback on intermediate results. The approach proposed in this paper could be used as a first step towards achieving "minimally exposed" global computation.

8. REFERENCES

[1] Aggarwal, C.C., Pei, J., and Zhang, B. On Privacy Preservation against Adversarial Data Mining. In *Proceedings of ACM SIGKDD*, 2006.

[2] Aggarwal, C.C., and Yu, P.S. A General Survey of Privacy-Preserving Data Mining Models and Algorithms. *Advances in Database Systems*, *34*, 2008.

[3] Agrawal, R., Kiernan, J., Srikant, R., and Xu, Y. Hippocratic databases. In *Proceedings of VLDB*, 2002.

[4] Agrawal, R., and Srikant, R. Privacy-Preserving Data Mining. *ACM Sigmod Record, 29*(2), 2000.

[5] Alimonti, P., Ausiello, G., Giovaniello, L., and Protasi, M. *On the Complexity of approximating weighted satisfiability problems*. Technical Report, Università di Roma, 1998.

[6] Allard, T., Anciaux, N., Bouganim, L., Guo, Y., Le Folgoc, L., Nguyen, B., Pucheral, P., Ray, I., Ray, I., and Yin, S. Secure Personal Data Servers: a Vision Paper. In *Proceedings of the VLDB Endowment*, *3*(1)*,* 2010.

[7] Anderson, A.H. An Introduction to the Web Services Policy Language (WSPL). In *Proceedings of the POLICY Workshop*, 2004.

[8] Ardagna, C.A., Cremonini, M., De Capitani di Vimercati, S., and Samarati, P. A privacy-aware access control system. *Journal of Computer Security, 16*(4), 2008.

[9] Ashley, P., Hada, S., Karjoth, G., Powers, C., and Schunter, M. Enterprise privacy authorization language 1.2 (EPAL 1.2). W3C Member Submission, 2003.

[10] Baesens, B., Setiono, R., Mues, C. and Vanthienen, J. Using neural network rule extraction and decision tables for credit-risk evaluation. *Management Science*, *49*(3), 2003.

[11] Bazgan, C., and Paschos, V. Th. Differential approximation for optimal satisfiability and related problems. *European Journal of Operational research, 147*(2), 2003.

[12] Belotti, P., Lee, J., Liberti, L., Margot, F., and Wachter, A. Branching and bounds tightening techniques for non-convex MINLP. *Optimization Methods and Software, 24*(4-5), 2009.

[13] Brickell, J., and Shmatikov, V. The cost of privacy: destruction of data-mining utility in anonymized data publishing. In *Proceeding of ACM SIGKDD*, 2008.

[14] Byun, J.-W., and Li, N. Purpose based access control for privacy protection in relational database systems. *Very Large Data Bases Jounal*, *17*(4), 2008.

[15] Cranor, L., Langheinrich, M., Marchiori, M., Presler-Marshall, M., and Reagle, J. The Platform for Privacy Preferences 1.0 (P3P1.0) Specification. W3C Recommendation, 2002.

[16] Crook, J.N., Edelman, D.B., and Thomas, L.C. Recent developments in consumer credit risk assessment. *European Journal of Operational Research*, *183*(3), 2007.

[17] Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data. *Official Journal of the EC*, *23*, 1995.

[18] Dwork, C., and Lei, J. Differential privacy and robust statistics. In *Proceedings of ACM symposium on Theory of computing*, 2009.

[19] Escoffier, B., and Paschos, V.Th. Differential approximation of min sat, max sat and related problems. *European Journal of Operational research*, *181*(2), 2007.

[20] Evfimievski, A., and Grandison, T. *Privacy Perserving Data Mining*. Handbook of Research on Innovations in Database Technologies and Applications: Current and Future Trends, Chapter LVI, 2009.

[21] Fourer, R., Gay, D.M., and Kernighan, B.W. A Modeling Language for Mathematical Programming. *Management Science*, *36*, 1990.

[22] Fung, B.C.M., Wang, K., Chen, R., Yu, P. Privacy-Preserving Data Publishing: A Survey on Recent Developments. *ACM Computing Surveys*, *42*(4), 2010.

[23] Huysmans, J., Baesens, B., Vanthienen, J. Using rule extraction to improve the comprehensibility of predictive models. Open Access publications from Katholieke Universiteit Leuven, 2007.

[24] Kirkpatrick, S., Gelatt, C.D., and Vecchi, M.P. Optimization by Simulated Annealing. *Science*, *220*(4598), 1983.

[25] LeFevre, K., Agrawal, R., Ercegovac, V., Ramakrishnan, R., Xu, Y., and DeWitt, D. Disclosure in hippocratic databases. In *Proceedings of VLDB*, 2004.

[26] Lindell, Y., and Pinkas, B. Privacy Preserving Data Mining. In *Proceedings of Advances in Cryptology*, 2000.

[27] Mitchell, T. *Machine Learning*. McGraw-Hill, 1997.

[28] Moses, T. Extensible access control markup language (xacml) version 2.0. Oasis Standard, 2005.

[29] Ni, Q., Bertino, E., and Lobo, J. An obligation model bridging access control policies and privacy policies. In *Proceedings of ACM SACMAT*, 2008.

[30] Ni, Q., Bertino, E., Lobo, J., Brodie, C., Karat, C.-M. , Karat, J., and Trombetta, A. Privacy-aware role-based access control. *ACM TISSEC*, *13* (3), 2010.

[31] OECD Guidelines on the Protection of Privacy and Transborder Flows of Personal Data, 23rd Sept. 1980.

[32] Papadimitriou, C., and Yannakakis, M.. Optimization, approximation and complexity classes. In *Journal of Computer and System Sciences*, *43*, 1991.

[33] Ponemon Institute, LLC. 2010 Annual Study: U.S. Cost of a Data Breach. 2011.

[34] Samarati, P. Protecting respondents' identities in microdata release. *IEEE TKDE*, *13*(6), 2001.

[35] Sweeney, L. k-Anonymity: a model for protecting privacy. *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, *10*, 2002.

[36] US Department of Homeland Security. Privacy Policy Guidance Memorandum, the Fair Information Practice Principles: Framework for Privacy Policy at the Department of Homeland Security. Memorandum N° 2008-01, 2008.

[37] Verykios, V.S., Elmagarmid, A.K., Bertino, E., Saygin, Y., and Dasseni, E. Association Rule Hiding. *IEEE TKDE*, *16* (4), 2004.

[38] Xiao, X., Tao, Y. Personalized privacy preservation. In *Proceedings of ACM SIGMOD*, 2006.

[39] Ardagna, C.A., De Capitani di Vimercati, S., Foresti, S., Paraboschi, S., and Samarati, P. Minimising Disclosure of Client Information in Credential-Based Interactions. Int. Journal of Information Privacy, Security and Integrity, 1(2/3), to appear in 2012.

[40] Chen, W., Clarke, L., Kurose, J., and Towsley, D.. Optimizing cost-sensitive trust-negotiation protocols. IEEE Computer and Communications Societies (INFOCOM), 2005.

[41] Yao, D., Frikken, K.B., Atallah, M.J., and Tamassia, R. Private information: To reveal or not to reveal. In ACM TISSEC, 12(1), 2008.

APPENDIX A: PROBLEM EXTENSION

The results presented in this article can be extended to the cases where users official documents are composed of sets of predicates $a\theta v$ where $\theta \in \{=, <, >, \leq, \geq, \neq\}$. This means the same attribute may potentially be contained in distinct documents (e.g., doc_a: *salary>$1.000* and doc_b: *salary>$2.000*). This means that there is no longer just one document that can prove a given predicate, and therefore there is no longer and equivalence between computing the solution to *ME* and the truth value of the Rule Set Boolean Formula as described in Section 3.1. Instead, a more complex formula E'_R must be computed, where each rule is replaced by the *disjunction* of the Boolean variables representing the multiple documents proving it:

PROBLEM 3. *Multi-proof Boolean Minimum Exposure (MPME)*

Given a rule set R, $data_u=\{doc_i\}$ a set of q signed documents that proves R, B a set of Boolean variables $B=\{b_1,..,b_q\}$ such that $b_i=true \Leftrightarrow doc_i$ is exposed, $E'_R=\wedge_j(\vee_k (\wedge_m(\vee_q b_{j,k,m,t})))$ where $\forall j,k,m,t$ $b_{j,k,m,t} \in B$, and the exposure function **EX'**, $data_u$ is *n-exposable* with regards to R if and only if there exists a truth assignment T_B of B such that **EX'**$(T_B) \leq n$ and E'_R is *true*.

In what follows, we consider the related *MPME* optimization problem whose goal is to minimize n. It is straightforward that *ME* is reducible to *MPME*, since it is a special case of *MPME* where $\forall j,k,m$ $t=1$. Therefore, the hardness results of COROLLARY 1 and 2 also hold for PROBLEM 3.

Let us stress that if the user documents *uniquely prove* the rule set, then PROBLEM 1 and PROBLEM 3 are equivalent (t can only take value 1). In general, this will not be the case and PROBLEM 3 will have to be solved, which is at least as hard as PROBLEM 1. We show next that both problems can be solved in a similar way.

Roughly speaking, this simply requires adapting the $\mathbf{EX_{card}}$ function to take into account the fact that, e.g., *salary>$50.000* is less costly if a breach occurs than *salary=$61.000*, and choosing for each predicate the document with lowest cost. To this end, we introduce $\mathbf{EX_{PRED}}$, a function computing the cost of a given predicate $p=a\theta v$, where $dom(a)$ is finite:

if $\theta \in \{=\}$: $\mathbf{EX_{PRED}}(p) = 1$
And if $\theta \in \{<, >, \leq, \geq, \neq\}$:
$\mathbf{EX_{PRED}}(a\theta v) = 1 - |\{x \in dom(a): x\theta v\}| / |dom(a)|$

and $\mathbf{EX'_{card}}$ a function computing the exposure of a set of predicates on the *published attributes* of A by summing the maximum value of their exposure:

$$\mathbf{EX'_{card}}(B) = \Sigma_{a \in \text{published attributes}} \text{MAX}_{p = a\theta v}(\mathbf{EX_{PRED}}(p))$$

By using $\mathbf{EX'_{card}}$ as objective function and by choosing for each p_i the single document $doc_{j,k,m}$ minimizing $\mathbf{EX'_{card}}$ that proves it (obviously we disregard two identical documents), each predicate in the rule set can be replaced by the *unique* document predicate minimizing $\mathbf{EX'_{card}}$ and therefore a solution to the *MPME* problem can be computed by solving PROBLEM 1. We show next how the *HME* algorithm runs with $\mathbf{EX'_{card}}$ instead of $\mathbf{EX_{card}}$.

The idea is to change the *fix( )* function to use $\mathbf{EX_{PRED}}$ and $\mathbf{EX'_{card}}$: the function now computes a minimum bound on the value of $\mathbf{EX'_{card}}$ if a document is not sent. In this example, we suppose that the problem involves three different predicates on salary. The new predicates are p_{11}: *salary >$10.000*, p_{12}: *salary>$20.000*, p_{13}: *salary>$30.000*. The rule set is as follows:

r_1: $(p_{11} \wedge p_2) \vee (p_3 \wedge p_4) \Rightarrow c_1$
r_2: $(p_5 \wedge p_6 \wedge p_7) \vee (p_4 \wedge p_8 \wedge p_9) \Rightarrow c_2$
r_3: $(p_{12} \wedge p_6 \wedge p_7) \vee (p_2 \wedge p_4 \wedge p_{10}) \Rightarrow c_3$
r_4: $(p_2 \wedge p_5 \wedge p_6 \wedge p_7) \vee (p_{13} \wedge p_4 \wedge p_8 \wedge p_9) \Rightarrow c_4$

In order to compute $\mathbf{EX_{PRED}}$, we suppose for instance that a salary is in the domain [*0*; *100.000*]. Therefore, $\mathbf{EX_{PRED}}(p_{11}) = 0.3$, $\mathbf{EX_{PRED}}(p_{12}) = 0.2$ and $\mathbf{EX_{PRED}}(p_{13}) = 0.1$. For all other predicates p, $\mathbf{EX_{PRED}}(p)=1$. Also note that since $p_{13} \Rightarrow p_{12}$ and $p_{13} \Rightarrow p_{11}$ setting p_{13} to *false* means that setting p_{11} to *false* while leaving p_{13} to *true* will provide no breach cost improvement, since this information can be inferred. However, p_{13} can be set to *false* while p_{11} and p_{12} could remain *true*, which would provide the exposure gain $\mathbf{EX_{PRED}}(p_{11})$-$\mathbf{EX_{PRED}}(p_{12})=0.1$. Concerning the *fix( )* function note that removing p_{13} only fixes predicates p_2, p_5, p_6, and p_7.

In Table 5, we show an execution of the *HME* algorithm computing *fix( )* for each step. The result is T_B=[p_{11}:*true*, p_{12}:*true*, p_{13}:*false*, p_2:*true*, p_3:*false*, p_4:*false*, p_5:*true*, p_6:*true*, p_7:*true*, p_8:*false*, p_9:*false*, p_{10}:*false*], and the value of $\mathbf{EX'_{card}}(T_B)=4.2$. Compared to the example presented in Table 2, the same truth assignment was found for predicates p_2 through p_{10} and the algorithm was also able to set p_{13} to *false*, while keeping p_{12} and p_{11} *true*, thus slightly reducing data exposure: instead of exposing *salary=$35.000* the user will expose *salary>$20.000*.

Table 5. Execution of the *HME* Algorithm (signed predicates).

Steps	p_{11}	p_{12}	p_{13}	p_2	p_3	p_4	p_5	p_6	p_7	p_8	p_9	p_{10}
1:score	7	6	4	4.2	**1.1**	4.1	3.3	5.3	5.3	4	4	0
2:score	∞	6.1	4.1	∞	–	4.2	4.3	5.3	5.3	4.1	4.1	**3.2**
3:score	∞	∞	**4.2**	∞	–	4.2	4.3	∞	∞	4.2	4.2	–
4:score	∞	∞	–	∞	–	**4.2**	∞	∞	∞	4.2	4.2	–
5:score	∞	∞	–	∞	–	–	∞	∞	∞	**4.2**	4.2	–
6:score	∞	∞	–	∞	–	**–**	∞	∞	∞	–	**4.2**	–
7:B[*i*]	*true*	*true*	*false*	*true*	*false*	*false*	*true*	*true*	*true*	*false*	*false*	*false*