Neurocomputing 74 (2011) 1497–1501

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Letters

A neural network approach for data masking

Vishal Anjaiah Gujjary, Ashutosh Saxena *

Security and Privacy Lab, SETLabs, Infosys Technologies Ltd., Survey No. 210, Gachibowli, Hyderabad - 500 032, India

ARTICLE INFO

Article history:
Received 7 July 2009
Received in revised form
2 September 2010
Accepted 29 January 2011
Communicated by G. Phillips-Wren
Available online 24 February 2011

Keywords:
Data privacy
Adaptive data masking (ADM)
Neural network

ABSTRACT

In this letter we present a neural network based data masking solution, in which the database information remains internally consistent yet is not inadvertently exposed in an interpretable state. The system differs from the classic data masking in the sense that it can understand the semantics of the original data and mask it using a neural network which is a priori trained by some rules. Our adaptive data masking (ADM) concentrates on data masking techniques such as shuffling, substitution, masking and number variance in an intelligent fashion with the help of adaptive neural network. The very nature of being adaptive makes data masking easier and content agnostic, and thus finds place in various vertical domains and systems.

1. Introduction

Outsourcing of information technology is not an isolated trend but part of a bigger shift towards the globalization of business processes. As more and more organizations are joining the bandwagon of outsourcing, data privacy is gaining a lot of corporate and media attention. Countries around the world are developing regulations designed to support privacy. The ease at which data can be collected automatically, stored in databases and queried efficiently over the Internet (or otherwise) has paradoxically worsened the privacy situation and has raised numerous ethical and legal concerns. Legislations are being passed insisting that the companies which are outsourcing consumer data to foreign countries must assume responsibility for the data. The recent legislative actions, as well as frequent media reports of security breaches, have made the prevalence of such threats clear [17]. Problems arising from private data falling into malicious hands include identity theft, stalking on the web, spam, etc. [1].

A worldwide movement towards data privacy legislation has put pressure on organizations to improve their information privacy and security standards. Data privacy research indicates that more than 70% of all security incidents come from internal threats. Moreover, data breaches coming from the inside and the associated costs of such internal breaches are more than 50 times as costly when compared with external breaches [18]. Test and development teams are restricted from viewing any security sensitive information. However, this information could be structurally correct functional copies of original databases. The actual content of databases is irrelevant for test and development teams, as long as the data looks real [11]. Names, addresses, phone numbers, e-mail addresses, credit card details are good examples of this kind of data. The requirements for data masking are greatly dependent on country, environment, etc. Many countries have their own legal regulations of some form or other.

This work provides a comprehensive overview of the issues that are involved in data masking and present a method to adaptively mask the data in the databases while retaining the same look-and-feel of the original data. The letter is organized as follows. In Section 2, the fundamentals of the data masking along with various data sanitization techniques are discussed. Section 3 provides a list of limitations of traditional data masking techniques. Section 4 presents the proposed data masking technique. Some conclusions are drawn in Section 5.

2. Fundamentals of data masking

Data masking is a process whereby the information in a database is masked or 'de-identified'. It enables the creation of realistic data in non-production environments without the risk of exposing sensitive information to unauthorized users. We now discuss various data masking techniques [11,2] with their individual benefits and drawbacks.

NULL'ing out is a technique where a column is simply deleted or is replaced by NULL values. Even though this is the most effective way of data masking, this is least preferred from test database point of view, as it leaves the test team with no data to work on. The other technique *Masking* means replacing certain

* Corresponding author. Tel.: +91 40 664 20000.
E-mail addresses: Vishal_Gujjary@infosys.com (V. Anjaiah Gujjary), Ashutosh_Saxena01@infosys.com (A. Saxena).

0925-2312/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.neucom.2011.01.002

fields with mask character (such as *). It disguises the data, retaining the format. For example, the date of birth of an employee in an organization could be 07/06/1986. And after masking the information would look like 07/06/19** (disguising the year of birth).

Substitution is a technique wherein the content of the column data is randomly replaced with data that is similar but is completely unrelated. Another technique *Shuffling* is similar to substitution except that the data is derived from the column itself. The method of *Number Variance* is operated on numeric data. It means modifying each value in a column with a random percentage of the number within predefined limits. *Encryption/Decryption* is a technique which offers the option of leaving the data in place and visible to those who have the key and hiding it from those who do not have the key.

Major of the business environments today depend on some or other form of data masking and among the variety of techniques available, several of them are required based on size and structure of the databases. Each of the techniques discussed above have some limitations which are addressed in the following section for effective and safe masking.

3. Limitations of data masking techniques

Among the available variety of software tools, solutions and systems implementing data masking techniques, most of them have some major drawbacks [12] which include:

Relevant data: Resemble the look-and-feel of the original information.

Row-internal data synchronization: In many cases the contents of one column in a row are related in some way to the contents of the other columns in the same row. For example, if the gender field in the row is then the FIRST_NAME column should really have a female first name.

Table-internal data synchronization: Often due to masking operation rows in the table are massively de-normalized and contain information that must be identical among many rows. For example all the rows of a column in a table pertaining to a specific geographical location must contain the same currency for salary and not different currencies.

Table–table data synchronization: Quite often in test systems, important information which must be obfuscated is used as a join key to a column in one (or more) other tables. For example, changes to the EMPLOYEE_NUMBER column in one table must trigger identical changes in other tables.

Field overflow: Care must be taken when replacing the real data with false test data to avoid overflowing previously allocated storage capacity.

Consistent masking: A common requirement for a sanitization process is to ensure that the output is consistent across multiple runs. In practice this means that if the name of employee Joe Smith gets changed to Bill Jones during masking and later when the database is again cloned and sanitized, the same entity, i.e. Joe Smith should again be masked as Bill Jones, hence maintaining consistency while masking.

Isolated case phenomena: For example, could the record for the organizational CEO be determined by finding the largest salary in the table? Sometimes each record is its own special case. An unmasked birth date could readily be attributable to a specific individual. Whether this issue is important depends largely on how the remainder of the information is masked.

Meta information: Sometimes even if information is not attributable to a specific individual, the collection of information might well be sensitive. For example, salary figures may be anonymous because the associated employee names have been masked, but does it matter if someone can add up the salary figures for a department?

Some other issues include the existence of pre-defined dictionaries for masking data types such as names, currency, etc. These dictionaries are limited and cannot be generalized for all databases types. Above all there is a possibility and threat of insider attack on the database, metadata and procedures for masking. To address these problems, ADM is proposed. It is different from previously presented data masking techniques as it is based on neural network which is so far not explored to our knowledge. Moreover, this approach is not dependent on external dictionaries as it is self-learning from the data provided by the databases and hence using the databases as the dictionaries. The architecture and the method is discussed in the following sections.

4. Data masking using neural network

A neural network is a massively parallel distributed processor made up of simple processing units that has a natural propensity for storing experiential knowledge and making it available for use [21,8,22,7]. In general, neural networks can have any number of layers, and any number of nodes per layer. Most applications use the three layer (input, hidden and output) structure with a few hundred input nodes. The hidden layer is usually about 10% the size of the input layer. The number of nodes in the output layer is framed based on the kind of result we expect.

In general, neural networks are organized based on categories of learning (supervised and unsupervised) [13] and the categories of structures (feedforward and feedback) [6,19]. The kind of neural network architecture used depends on pattern recognition task at disposal. For instance, the associative memory function based neural network greatly fits for operations such as learning, storing and recall. Thus, based on the pattern recognition task, neural networks can be broadly classified into five different classes of architecture namely associative memories, pattern mapping, pattern classification, temporal patterns and pattern variability.

We adopt associative memory [9] based approach in this work which falls into the pattern storage classification. Such memory function helps in storage of pattern information (not data) for future recall. Any neural network in general acts like an associative memory, in which a pattern is coupled with another pattern. The pattern knowledge is stored in the weight matrix of a feedback neural network. Based on the partial input, the stored pattern can be recalled.

Before the neural network is passed to the training phase, the weights of the nodes are either initialized to zero or to any random value. After training, the weights of the different nodes change accordingly adjusting to the dataset with which it is trained. Moreover, the weights characterize different data masking rules, using which the training dataset is formed.

4.1. Formation of training database and training

The first phase is related to training of neural network as depicted in Fig. 1. During training all the exhaustive database set is considered. The database set consists of different databases with different properties and with different data types. The way neural network is trained, is governed by the rules of data masking. The block consists of exhaustive list of rules (rule 1, rule 2 …) that could be applied as part of masking. Based on the set of rules checked, the rule set forms a metadata from the given dataset. This metadata is a sparse way of representing training data and is presented to database extractor E. The process