DB CyberTech

Technology Brief

Data Classification for Structured Data Stores

Introduction

Data classification is a pervasive challenge in security and compliance for any company collecting or managing sensitive data. With examples from the European Union's General Data Protection Regulation (GDPR) to California's Consumer Privacy Act A.B. 375 derivative, governments are putting increasing emphasis on the privacy and security of sensitive data. Companies are answerable for what data they have, who has access, and how it is handled.

For unstructured data, a great wealth of document classification approaches are available. From older document clustering techniques and traditional classification algorithms such as support vector machines and neural networks to forward looking transfer learning, academia and industry have provided powerful technology and solutions. However, when it comes to the *largest* collections of sensitive data, universally stored in *databases*, there is a striking deficiency in good tools and technologies. **Why is it hard to answer simple questions like "Where is there personally identifiable information in my database infrastructure?" or "Who is accessing my regulated data?"**

The simple answer is complexity, scale, and context. Database environments are not a collection of static documents. They are characterized by a dense and evolving set of connections supporting multiple software applications, business interests, and a myriad of users. This complexity is exacerbated by the sheer scale of each dimension. There can be hundreds or thousands of users for a single database, and a comparable number of tables and processes running on each database server. Together, the complexity and scale of structured database environments exceed what traditional document classification can handle. Perhaps more fundamentally, information in structured databases is quite different than traditional documents. There is generally a rigorous separation between *data* and the *metadata* which gives it context, rendering document-centric classification ineffective.

Common Practice

So, what to do? The most common approach is to rely on human assistance to discover the database objects of potential interest and manually select and transform this data for traditional *content*-oriented scanning. Given the scale of enterprise data infrastructure, this is extremely labor-intensive and expensive. Worse, it is inevitably incomplete and out of date, leaving vast silos of dark structured data, all too often sensitive, and subject to regulation.

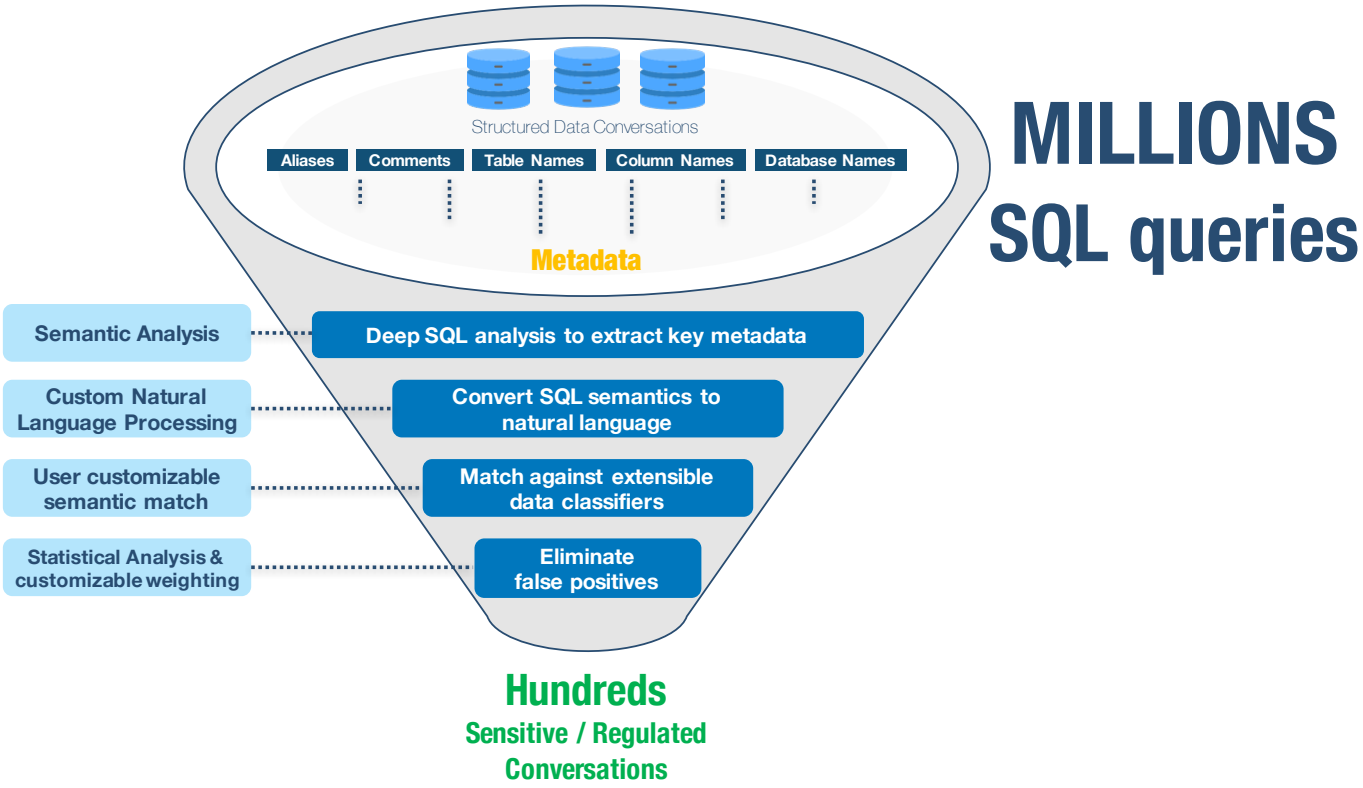
DB CyberTech Approach

DB CyberTech's data classification uses an innovative approach to solve these problems. There are three key ingredients:

We examine a *continuous* feed of database *conversations* decoded from the SQL protocols that describe data-in-motion. We do this at enterprise scale, and in real time, and consider the full breadth of all database activity.

We automatically extract the critical **metadata** describing these conversations from database queries and other statements. There is an enormous wealth of information here, encoded by *human* programmers, according to the rules of data-access language, principally, *Structured Query Language* (SQL). We recognize these human artifacts via *Semantic Analysis* which uncovers syntax-constrained information such as database, table, and column names, query aliases, often conveying the meaning of complex expressions to human programmers, and of course the unconstrained comments which frequently convey the *meaning* of the conversations. Much of this information exists nowhere else – it's not in the metadata or data within databases, but only in the SQL statements accessing them!

Lastly, we exploit this recovered SQL information to classify data meaningfully for our customers. We use bespoke *Natural Language Processing* (NLP) technology to lift the abstraction level of typically compressed and idiomatic artifacts recovered from SQL statements up to human language, telling us what each query is about. Users can optionally tailor how abbreviations are expanded to standard words in a *semantic dictionary*, or even add custom entries. *Semantic networks* transparently represent a set of *data classifiers* describing the categories and human-language terms of the data we analyze. Classifiers can be customized from fine details up to their structure. Finally, we minimize false positives, quantifying classification *evidence* based on both user-defined weights and statistical analysis of how *meaningful* matched terms are within their context.



As a simple example of this abstract-sounding technology, consider the SQL query

"select lstNm from acmCust"

recovered from our data feed. Our semantic analysis tells us that "lstNm" is a column read by this query of a table called "acmCust". Our NLP expands "lstNm" to "last" and "name", for which we find an entry "last name" within our standard dictionary (by this we actually *understand* what a last name *is*). Similarly, "acmCust" expands to "acm" and "customer", because there is an enterprise-wide abbreviation, "acm" is known to mean "Acme Corporation", and we understand the table pertains to Acme's customer data. Our standard *Personally Identifiable Information* (PII) classifier ultimately *infers* that this specific table, in its containing database has PII data, sub-categorized as *individual identification* pertaining to Acme's customers.

In summary, from purely passive copies of database network traffic, our data classification technology can *autonomously* discover where sensitive data resides, how its accessed, and by whom, across the enterprise as a whole, in real time, and updated continuously. This lets users quickly and accurately focus their attention, security technology, and human resources on *just* the data that matters.