SYNTHETIC DATA GENERATION CAPABILTIES FOR TESTING DATA MINING TOOLS

Daniel R. Jeske
Pengyue J. Lin
Carlos Rendón
Rui Xiao
University of California, Riverside
djeske@ucr.edu

Behrokh Samadi
Lucent Technologies
samadi@lucent.com

ABSTRACT

Recently, due to commercial success of data mining tools, there has been much attention to extracting hidden information from large databases to predict security problems and terrorist threats. The security applications are somewhat more complicated than commercial applications due to (i) lack of sufficient specific knowledge on what to look for, (ii) R&D labs developing these tools are not able to easily obtain sensitive information due to security, privacy or cost issues. Tools developed for security applications require substantially more testing and revisions in order to prevent costly errors. This paper describes a platform for the generation of realistic synthetic data that can facilitate the development and testing of data mining tools. The original applications for this platform were people information and credit card transaction data sets. In this paper, we introduce a new shipping container application that can support the testing of data mining tools developed for port security.

KEYWORDS

Knowledge Discovery and Data Mining, Synthetic Data Generation, Semantic Graphs, Shipping Container.

INTRODUCTION

Knowledge discovery and data mining (KDD) includes the technology of extracting unknown and possibly useful information from data. This process has been compared to finding a needle in a haystack. In spite of apparent complexity, KDD technology has shown to be successful in some commercial applications such as fraud prevention and medical diagnosis. KDD is a powerful technology with great potential to help focus attention on the most important information in data warehouses. Data mining tools predict future trends and behaviors, allowing businesses to make proactive, knowledge-driven decisions. Recently, due to this commercial success, there has been much attention paid to extraction of hidden information from large databases to predict security problems and terrorist threats. The security applications are somewhat more complicated than the commercial applications due to (1) lack of sufficient specific knowledge on what to look for, and (2) R&D labs developing these tools are not able to easily obtain sensitive information due to security, privacy or cost issues. KDD tools developed for security applications require substantially more testing and revisions of rules in order to minimize the false positives and false negatives that could be very costly.

The combination of (1) and (2) motivated us to develop a platform for the generation of realistic synthetic data that can facilitate the development and testing of KDD tools. Realistic synthetic data can serve as background data sets into which hypothetical future scenarios can be overlaid. KDD tools can then be measured in terms of their false positive and false negative error rates. In addition, the availability of synthetic data sets provides necessary traction for new data mining ideas and approaches, and thereby facilitates the development and feasibility assessment of techniques that might otherwise die on vine.

To be adequate substitutes for real data, the quality of synthetic data sets needs to be reasonable. Pitfalls

associated with unrepresentative data include improper training of data mining tools and masking of true benefits and virtues associated with specific techniques. Some synthetic data generation tools are available commercially [1]. To varying degrees, each tool comes with a pre-defined set of attributes whose values are available from built-in lists. For example, lists could include names, addresses, occupations, etc. None of the tools we are aware of preserve complex between-attribute relationships, but instead simply generate attributes as though they are independent. Some of the commercial tools allow integration of user-customized data sets. However, the practical issue of obtaining the data sets to be integrated with these tools remains an obstacle. To support various KDD tools, the system architecture should be portable to different platforms, scaleable for simple to complex applications, flexible to integrate new applications, industrially compatible in data output formats, and highly usable for *off-the-shelf* and *power users*.

In previous work [2, 3] we developed a prototype of an IDAS Dataset Generator (IDSG). The initial prototype was supported by a Technical Support Working Group (TSGW) that identifies, prioritizes and coordinates R&D requirements for combating terrorism [4]. The IDSG platform uses semantic graphs to represent relationships among data attributes and to define data generation rules for specific data set applications. Some attributes are generated by statistical algorithms, while others follow rules-based algorithms. The platform allows the development of customized applications to serve specific user needs. The applications developed for the prototype included *people information* and *credit card transactions*.

The rest of this paper is organized as follows. In the next Section we review the applications, features, and algorithms associated with the prototype IDSG by walking through a typical user interaction with the tool. Though not all features will be illustrated, the main functions of IDSG will become evident. Following that, we introduce a new *shipping container* application. The shipping container application poses a different challenge since unlike *people information* and *credit card transactions*, its data attributes have not been the subject of numerous studies and relatively little a-priori information is available to persons not directly working in the shipping container domain. For example, names, addresses, social security numbers and even relationships such as the association between income and education level can be found in public sources. Similarly rules for valid generation of credit card numbers can be found. However, the nature of the contents of shipping containers, the relationships between carriers, originating countries, arriving ports and commodity descriptions is not common knowledge and is difficult to ascertain through web surfing.

TOUR OF IDSG

The prototype IDSG is based on the client-server computing model. Whether the client and the server execute on the same machine or not is transparent to the system. The role of the client is to allow the user to select an application type and specify requirements for the data sets they need. Specifically, the user constructs a schema for how the outputted data sets should be organized by specifying how many files of data they need and what attributes should be attached to each file. Figure 1 is a screen shot of the opening screen where the user selects the application they wish to generate synthetic data for, and Figure 2 is the screen that allows them to construct files for the output and to select the attributes (using pull-down menus) that are to appear in each.

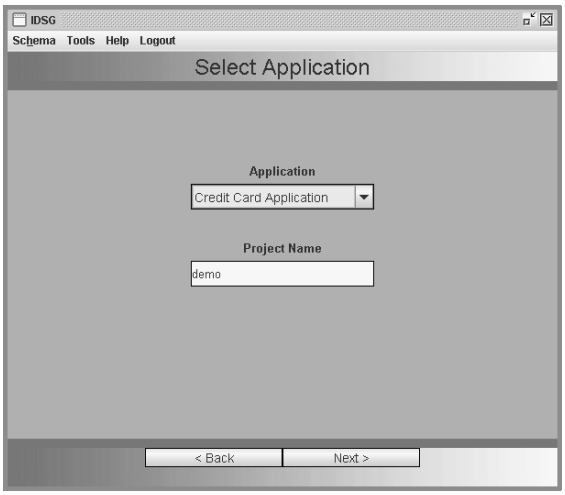


Figure 1. IDSG – Select Application

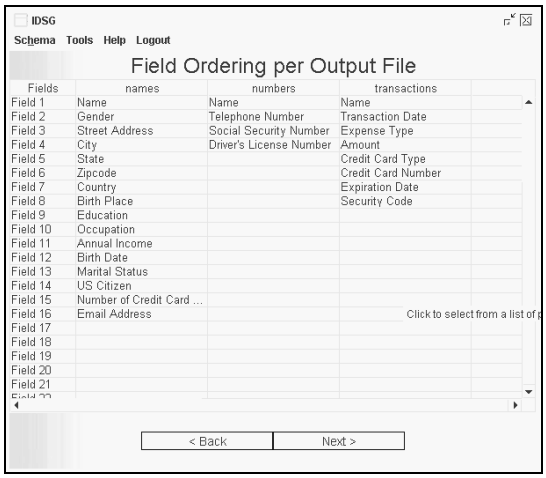


Figure 2. IDSG – Organizing Output Files

In this example, the user selected the *credit card transaction* application, and created three output files (names, numbers and transactions) with the attributes as shown. In total, the user has over 30 different attributes within this application that they can optionally select for inclusion in the output files.

The primary function of IDSG is to generate background data with the intent of looking like real data. However, the user is offered an opportunity to overlay pre-defined scenarios that can be randomly inserted and mixed with the background data. With this feature, the resulting data sets can be seeded with events and competing KDD tools can be tested for their ability to find the signal amongst noise. For example, a user might insert an unusually large number of credit card transactions for chemical purchases. A KDD tool that looks for anomalous credit card purchases could then be tested by presenting it with the transactions file that is outputted from IDSG. Figure 3 shows how a user can mix in pre-defined scenarios with the synthetic data that is generated by IDSG. The user simply enters file names that have the same structure as the IDSG output files (names, numbers, and transactions). The user also selects the insertion mode that determines whether to add the anomalous records at the beginning, at the end, or randomly throughout the IDSG output files.

At this point the user has completed the requirements specification on the data sets and the data generation responsibility is passed to the server. The user can check on the status of the job by viewing the progress monitor that is shown in Figure 4. The status of the job 'demo' in this example is that it has run 16 seconds and is 46% complete.

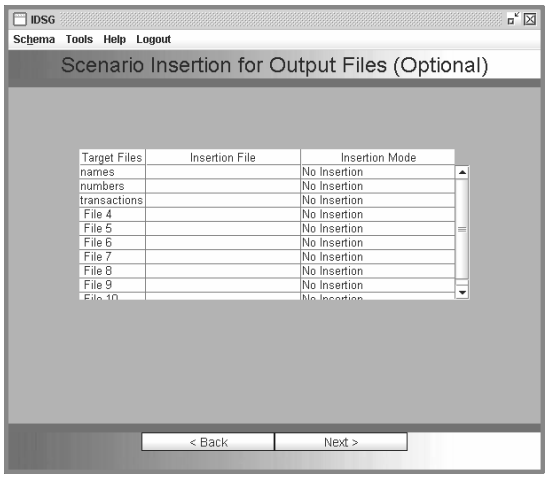


Figure 3. IDSG – Insertion of Anomalous Records

When the job completes, it will move from the lower table to the upper table in Figure 4. At that point, all of the requested files are available to the user and are presented individually as CSV files. Early in the interaction between the user and IDSG, the user would have specified a minimum number of cases to be generated in each file. The server packages all the output CSV files into a zipped file that the user can download from the server to their client by using the Destination Directory and Browse features shown in Figure 4.

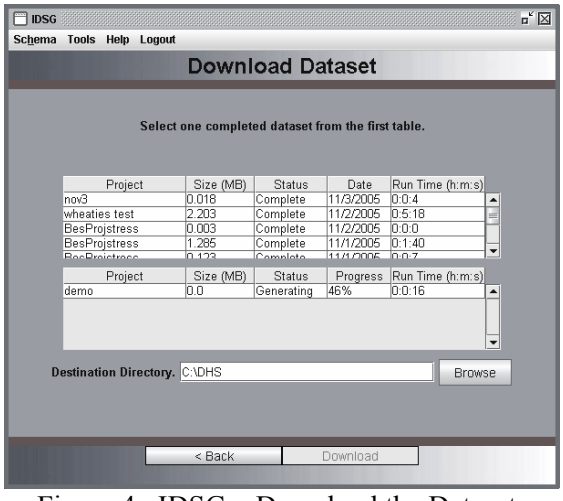
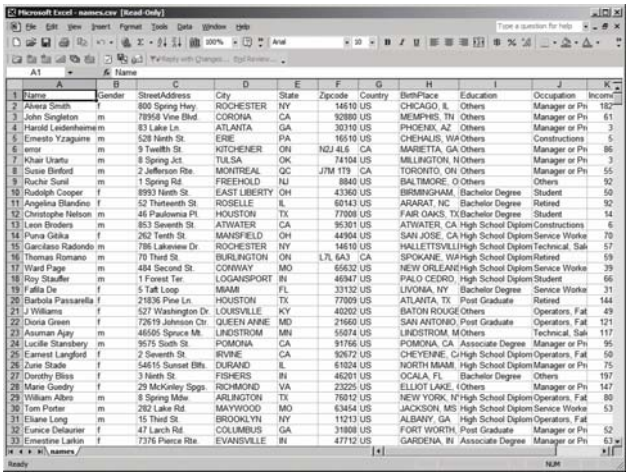


Figure 4. IDSG – Download the Datasets

Figure 5 illustrates the data that was created for the names file. The numbers file would similarly show all the user requested numbers (telephone, social security and driver's license) for each person in the names file. Likewise, the transactions file would show credit card purchases that were generated for each person including the date of purchase, type of credit card used, expense type and transaction amount.



	A	B	C	D	E	F	G	H	I	J	K
1	Name	Gender	StreetAddress	City	State	Zipcode	Country	BirthPlace	Education	Occupation	Income
2	Alvera Smith	f	800 Spring Hwy	ROCHESTER	NY	14610	US	CHICAGO, IL	Others	Manager or Pr	182
3	John Singleton	m	78958 Vine Blvd	CORONA	CA	92880	US	MEMPHIS, TN	Others	Manager or Pr	61
4	Harold Leidenheime	m	83 Lake Ln	ATLANTA	GA	30310	US	PHOENIX, AZ	Others	Manager or Pr	3
5	Ernesto Yzaguirre	m	528 Ninth St	ERIE	PA	16510	US	CHEHALIS, WA	Others	Constructions	5
6	error	m	9 Twelfth St	KITCHENER	ON	N2J 4L6	CA	MARIETTA, GA	Others	Manager or Pr	86
7	Khair Urartu	m	8 Spring Jct	TULSA	OK	74104	US	MILLINGTON, N	Others	Manager or Pr	3
8	Susie Binford	m	2 Jefferson Rte	MONTREAL	QC	J7M 1T9	CA	TORONTO, ON	Others	Manager or Pr	55
9	Ruchir Sunil	m	1 Spring Rd	FREEHOLD	NJ	8840	US	BALTIMORE, O	Others	Others	92
10	Rudolph Cooper	f	8993 Ninth St	EAST LIBERTY	OH	43360	US	BIRMINGHAM,	Bachelor Degree	Student	50
11	Angelina Blandino	f	52 Thirteenth St	ROSELLE	IL	60143	US	ARARAT, NC	Bachelor Degree	Retired	92
12	Christophe Nelson	m	46 Paulownia Pl	HOUSTON	TX	77008	US	FAIR OAKS, TX	Bachelor Degree	Student	14
13	Leon Broders	m	853 Seventh St	ATWATER	CA	95301	US	ATWATER, CA	High School Diplom	Constructions	6
14	Puna Gitika	f	262 Tenth St	MANSFIELD	OH	44904	US	SAN JOSE, CA	High School Diplom	Service Worke	70
15	Garcilaso Radondo	m	786 Lakeview Dr	ROCHESTER	NY	14610	US	HALLETTSVILL	High School Diplom	Technical, Sale	57
16	Thomas Romano	m	78 Third St	BURLINGTON	ON	L7L 6A3	CA	SPOKANE, WA	High School Diplom	Retired	59
17	Ward Page	m	484 Second St	CONWAY	MO	65632	US	NEW ORLEANS	High School Diplom	Service Worke	39
18	Roy Stauffer	m	1 Forest Ter	LOGANSPORT	IN	46947	US	PALO CEDRO,	High School Diplom	Student	66
19	Fatila De	f	5 Taft Loop	MIAMI	FL	33132	US	LIVONIA, NY	Bachelor Degree	Service Worke	31
20	Barbola Passarella	f	21836 Pine Ln	HOUSTON	TX	77009	US	ATLANTA, TX	Post Graduate	Retired	144
21	J Williams	f	527 Washington Dr	LOUISVILLE	KY	40202	US	BATON ROUGE	Others	Operators, Fat	49
22	Dioria Green	f	72619 Johnson Ctr	QUEEN ANNE	MD	21660	US	SAN ANTONIO,	Post Graduate	Operators, Fat	121
23	Asuman Ajay	m	46505 Spruce Mt	LINDSTROM	MN	55074	US	LINDSTROM, M	Others	Technical, Sale	117
24	Lucille Stansbery	m	9575 Sixth St	POMONA	CA	91766	US	POMONA, CA	Associate Degree	Manager or Pr	95
25	Earnest Langford	f	2 Seventh St	IRVINE	CA	92672	US	CHEYENNE, C	High School Diplom	Operators, Fat	58
26	Zune Stade	f	54615 Sunset Blfs	DURAND	IL	61024	US	NORTH MIAMI,	High School Diplom	Manager or Pr	75
27	Dorothy Bliss	f	3 Ninth St	FISHERS	IN	46201	US	OCALA, FL	Bachelor Degree	Others	197
28	Marie Guedry	f	29 McKinley Spgs	RICHMOND	VA	23225	US	ELLIOT LAKE,	Others	Manager or Pr	147
29	William Albro	m	8 Spring Mdw	ARLINGTON	TX	76012	US	NEW YORK, NY	High School Diplom	Operators, Fat	80
30	Tom Porter	m	282 Lake Rd	MAYWOOD	MO	63454	US	JACKSON, MS	High School Diplom	Service Worke	53
31	Eliane Long	m	15 Third St	BROOKLYN	NY	11213	US	ALBANY, GA	High School Diplom	Operators, Fat	
32	Eunice Delaurier	f	47 Larch Rd	COLUMBUS	GA	31808	US	FORT WORTH,	Post Graduate	Manager or Pr	52
33	Ernestine Larkin	f	7376 Pierce Rte	EVANSVILLE	IN	47712	US	GARDENA, IN	Associate Degree	Manager or Pr	63

Figure 5. IDSG – Output File 'Names'

The data generation algorithms used for the *people* and *credit card transaction* applications in IDSG have been described in [2,3,5]. In summary, there are three types of algorithms that get employed: statistical, rule-based, and resampling. Attributes such as gender, age, income, occupation, education level, number of credit cards, credit card use frequency, transaction type and transaction amount are statistically associated. For example, a person with higher education generally has higher income and therefore would generally have a higher number of credit card purchases and have higher transaction amounts. IDSG generates these attributes by constructing a 9-dimensional joint distribution for these attributes from knowledge about lower dimensional associations. In particular, a survey of 5000 adults in the U.S. provided sufficient data to identify 23 pairs of these attributes that had significant statistical associations. These associations are depicted as undirected links between the attributes in the semantic graph shown in Figure 6.

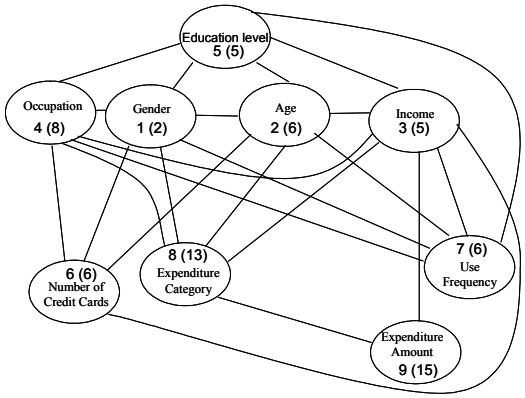


Figure 6. Semantic Graph Depicting Pair wise Associations between Attributes

The 9-dimensional joint distribution was then built by imposing that its corresponding two-way marginal distributions match these observed distributions. The 9-dimensional distribution was fit to the data using the iterative proportional fitting algorithm (IPF) [6]. This approach has two desirable features. First, it reflects exactly the information that is available concerning associations between these attributes – nothing more and nothing less. Second, the number of inputted lower dimensional marginals is a lever that is proportional to the quality of the synthetic data that gets generated. As additional lower dimensional information is input, the synthetic data becomes more realistic. Furthermore, IPF algorithm can be easily adapted to include additional information about attribute associations as it becomes available.

Rule-based algorithms are used for attributes that have specified types of random patterns. A classic example is how IDSG generates credit card numbers. American Express card numbers are 15 digits long, while Visa, Diners Club, Discovery and Master Card numbers are 16 digits long. Moreover, there are rules associated with the leading digits (e.g., Visa always begins with 4 and AMEX always begins with either 34 or 37) and the Luhn algorithm [6] is used to certify the entire string of digits is a valid card number. IDSG creates credit card numbers that adhere to all of these rules. Similarly, driver's licenses and plate numbers respect individual state conventions.

Finally, the third type of data generation algorithm is resampling which refers to generating values for an

attribute by resampling form a large population of real values. The classic example in IDSG is the social security number. The Social Security Death Master Index (SSDI) is a publicly available data base [8] that lists the names, last known addresses and the social security number of over 70 million deceased persons. Resampling from among these social security numbers ensures the fidelity of the social security numbers generated by IDSG. The SSDI is also resampled to randomly generate a surname, and then a first name is randomly selected from lists of common gender-based first names. As another example, the U.S. Postal Service maintains a database that connects a zip code to a city, state and NPA-NXX for nearly 500,000 cities in the U.S. IDSG resamples records from this database to ensure the fidelity between these four attributes.

SHIPPING CONTAINER APPLICATION

Approximately 140 different countries and regions deliver merchandise to the U.S. through approximately 170 different ports of call. Our new application is concerned with generating bills of lading for this imported merchandise. Interest in this application was motivated by our awareness of public concern over the security of U.S. ports and the corresponding anticipation that the development of KDD tools in this area are likely underway. As shown in Figure 6, a user can now generate synthetic data sets for this application.

For the initial development of this application, resampling was used for the data generation algorithm. The source data set that was resampled was purchased from the Port Import Export Reporting Service (PIERS) [9]. PIERS maintains a database of bills of lading that covers all vessels making calls between U.S. ports and foreign countries. For this particular application, we are interested only in the import data, which is the data pertaining to the vessels and cargos coming into the U.S. from foreign countries. A single shipping container can contain more than one order and therefore be mapped to more than one bill of lading. Similarly, a single merchandise order can span multiple shipping containers.

Figure 6. IDSG – Shipping Container Application

The data set we purchased from PIERS covers the shipping containers on all incoming vessels during the months of June, July and August of 2004. The number of bills of lading during these months is 2,071,371. Each bill of lading has as many as 64 different attributes. Table 1 shows a selected set of these attributes to provide a feel for the type of data on a bill of lading.

Selected PIERS Data Attributes	
Container Number	Name of Vessel
Commodity Description	Carrier Name
Origin Country	Carrier Code
Consignee Name	U.S. Port Code
Consignee Address	U.S. Port Name
Arrival Date	Cargo Volume (cubic ft)
Bill of Lading Number	Quantity of Cargo
Notify Party's Name	Weight of Cargo (Kg)
Notify Party's Address	Value ($)

Table 1. Selected PIERS Attributes

Since little a-priori information is available concerning attribute associations, a straight forward resampling data generation algorithm has been implemented for the first version of this application. As a result, the data sets generated will have very high fidelity. Similar to the other applications, the user can select how they wish to organize the information in the bills of lading into different files. Figure 7 shows the same type of file definition and attribute selection features that was provided for the other applications.

Just as with the other applications, a user can insert anomalous bills of lading using the IDSG scenario insertion feature. Once the file structure is defined for the output files, the user is offered the screen shown in Figure 3 to optionally seed the generated data sets with unusual records that KDD tools developers might hope to discover during testing.

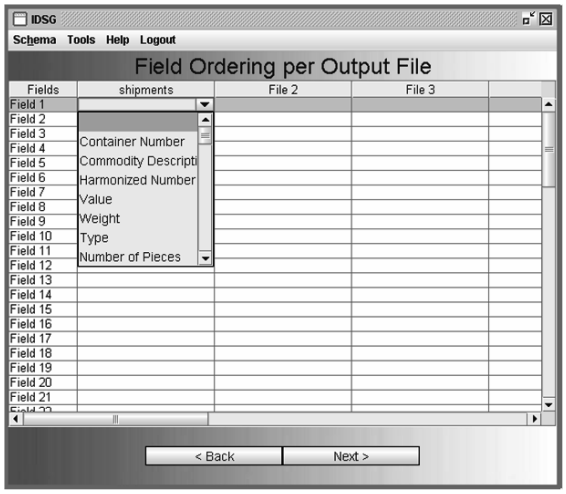


Figure 7. IDSG – Shipping Container Attribute File and Attribute Definitions

In future work on this application, learning algorithms will be developed to extract the most meaningful attribute characteristics and relationships from the data, so that new bills of lading can be formed by appropriately mixing information from different PIERS records.

SUMMARY

We have described a design and application for our test data generation tool IDSG that can facilitate building test cases for data mining tools by enabling the data mining developer to overcome time, cost, organizational and legal issues associated with gathering real data to build test cases. Our semantic graph with data dependency and scenario insertion approach differentiates IDSG from other commercial software. The shipping container data generation application demonstrated that the software design and architecture allow for the unlimited development of new applications without a change to the system infrastructure. Our approach and software architecture not only provide for *off-the-shelf* application users who want data sets of the type IDSG already generates, but also give the computing infrastructure to develop a *wizard* for *power users* who want to design customized application data sets themselves by specifying their own semantic graph.

REFERENCES

[1] Turbo Data (turbodata.com), GS Data Generator (GSApps.com), DTM Data Generator (sqledit.com) and RowGen (iri.com)

[2] Jeske, D. R., Behrokh Samadi, B., Lin, P., Ye, L., Cox, S., Xiao, R., Younglove, T., Ly, M., Holt, D., Rich, R. (2005), Generation of Synthetic Data Sets for Evaluating the Accuracy of Knowledge Discovery Systems. *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 756-763, August 21-24, 2005, Chicago, USA

[3] Lin, P., Behrokh Samadi, Jeske, D. R., Cox, S., Rendón, C., Holt, D., Cipolone, A., Xiao, R. (2006), Development of a Synthetic Data Set Generator for Building and Testing Information Discovery Systems, *Proceedings of The Third International Conference on Information Technology : New Generations,* IEEE Computer Society, pp. 707-712, April 10-12, 2006, Las Vegas, USA

[4] TSWG TASK IP-IA-2126

[5] Jeske, D. R., Gokhale, D. V. and Ye, L. (2006) Generating Synthetic Data from Marginal Fitting for Testing the Efficacy of Data Mining Tools, *International Journal of Production Research* (*Special Issue on Data Mining*) Vol. 44, No. 14, pp. 2711-2730.

[6] Deming, W. E. and Stefan, F. F. (1940), On a Least Squares Adjustment of a Sampled Frequency Table When the Expected Marginal Totals are Known, *Annals of Mathematical Statistics*, Vol. 11, pp. 27-44.

[7] http://en.wikipedia.org/wiki/Luhn_formula

[8] SSDI - http://ssdi.genealogy.rootsweb.com

[9] PIERS - http://www.piers.com/