Journal of Theoretical and Applied Information Technology
10th March 2013. Vol. 49 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A NEW INTEGRATION METHOD OF INFORMATION FOR TABLE TENNIS TRAINING FROM A DATA WAREHOUSE

[1]JI PING, [2]DING ZHIYI

[1] Assoc. Prof., School of Physical Education, Ning Xia University
[2] Prof., School of Mathematics and Computer Science, Ning Xia University
E-mail: [1]jiping319@163.com

ABSTRACT

A continuing challenge for Integration Information of Table Tennis Training (TTT) is to develop efficient data unit component part. Many see data unit integration as a potential solution; however, the quality of data integration of Table Tennis Training tends to outweigh the potential benefits. The quality of TTT data unit integration include establishing and maintaining a data warehouse of Table Tennis Training data unit, searching for applicable data unit to be integrated in a design, as well as adapting data unit toward a proper structure. In this paper, a new Data Feature Model (DFM) method is suggested here for data classification and integration of TTT which consists of K-Shortest Path (KSP) algorithm and Data Feature Model Method. We found that this new method gives a higher accuracy and precision in data selection and integration process of TTT compared to the existing formal methods.

Keywords: *Table Tennis Training, Data Unit, Integration Of Information, Data Feature Model*

1. INTRODUCTION

Applications of ANN to power systems are a growing area of interest. Considerable efforts have been placed on the applications of ANNs to power systems. Several interesting applications of ANNs to power system problems [1]-[5], indicate that ANNs have great potential in power system on-line and off-line applications. The feature of an ANN is its capability to solve a complicated problem very efficiently because the knowledge about the problem is distributed in the neurons and the connection weights of links between neurons, and information are processed in parallel.

When we query a warehouse, it is said that we are retrieving information from it. This is taken for granted. But, how this happens is not always fully understood. As a result, when a user queries a warehouse [1], the query can only be answered through a "direct match" between the selection criteria within a query and data (up to aggregations of the data) [2]. In a case of querying a warehouse beyond this, the system is unlikely to answer the query. A conventional query is, in essence, concerned with only the propositional content of data [3]. We believe that data carries information [4–6]. A piece of data may carry information about another, and moreover it may carry information about a real world situation. Therefore, if we can define and formulate the notion of "the information con-tent of data", not only may we obtain insight about the essence of conventional queries, but also we may derive more information beyond "direct match".

However, it would appear that the notion of "information content of data" is elusive. It has been taken as the instance of a warehouse and the information capacity of a data schema as the collection of instances of the schema. Another view on the topic of the relationship between information and data is that if it is truthful, meaningful data is semantic information. We argue that such views miss two fundamental points. One is a convincing conception of "information content of data". To equate data with information overlooks the fact that data in a warehouse is merely raw material for bearing and conveying information. Information must be veridical [7], that is, it must relate to a contingent truth, while for data there is no such requirement. The other is a frame-work for approaching the information content of data whereby to reveal information.

That is to say, we define the following research question that we tackle in this paper: how the "information content" of data in a warehouse may be defined with mathematical rigor, and how this

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

notion after have been defined may help retrieve information through reasoning that cannot otherwise be possible through conventional queries.

To answer this research question, we purpose to look at the relationships between the information content of data, warehouse structure and domain knowledge, which may be captured as business rules. These include how tacit domain knowledge may be explicitly expressed and used.

In this paper, we present a novel framework for approaching the information content of data in a warehouse, which is centered on the notion of information content inclusion relation. It helps us understand how a warehouse does its job, i.e., providing information, and helps a warehouse system improve its capability of providing information through inference. The latter is achieved by introducing a variety of information sources such as domain knowledge. With the help of external information sources, queries that deal with a wider range of information than the propositional content of data within a warehouse may be answered. The underlying thought of the framework is based on a concept of information content of a signal. Dretske [4] firstly introduced the concept. Then Xu et al. extended Dretske's idea and gave a more detailed definition of the information content of a state of affairs. Our thoughts are based on the latter definition.

Many software organizations realized that developing the software using reusable data unit could dramatically reduce development effort, quality and accelerate delivery. But the non-existence of a standard searching technique for finding the suitable data and also the lack of appropriate tool in this field contributed towards in large-scale failures in their method. From the past studies on this field, it is found that researchers are tried with different methods to improve the adaptability of the data but very few studied had taken place in improving the efficiency of integration of information for Table Tennis Training (TTT). Fuzzy linguistic method is familiar in the information integration process [1]. In this paper we have used an algebraic model namely Data feature model in which text documents are represented as data features of identifiers, such as, index terms which is used in information filtering, in-formation integration, indexing and relevancy rankings along with K-Shortest Path (KSP) algorithm for classification of documents.

Data unit integration at its most basic level consists of making use of any existing information, data or product when designing and implementing a new system or product.

There are differing opinions as to which activities constitute genuine data unit integration. Replication of an entire software program does not count as data integration. Data integration of assets is dependent upon both similarities and differences between the applications in which the data is being used [2].

Many organizations already practice a limited form of data integration, for example, most developers have libraries of data unit that they have developed in previous projects, or they use standard libraries, which are available with many programming languages [1]. About 30% of the cases, it is a very ad-hoc method of data integration, and it will work very well on a small scale and it will not be suitable for entire organizations [3]. Instead, businesses need to implement a systematic data integration program in order to gain the full advantages of data integration.

The definition of a reusable data is "any data that is specifically developed to be used, and is actually used, in more than one context [3]. This does not just include code; other products from the system lifecycle can also be integrated, such as specifications and designs, and even requirements on occasion [4]. 'Data unit' in this case can be taken to include all potentially reusable products of the system lifecycle, including code, documentation, design, requirements etc.

There are various criteria that should be satisfied in order for an asset to be successfully reusable. These are grouped into General, Functional and Technical requirements [5]. General requirements focus on aspects such as compliance with relevant standards, completeness, modularity and simplicity. All data unit should conform to the General requirements. Functional requirements include such concerns as which business processes it will simulate or automate, and how well it does this. Functional requirements mainly concern Vertical or Domain-specific assets and tend to be very specific to each in-formation domain. Lastly, Technical requirements refer to criteria such as interoperability, portability, communication, security etc [2].

There are different levels of integration, which can be considered [3]. At the highest level, entire applications can be integrated on different platforms provided they are portable. Sub-systems can be integrated within different applications, possibly within different domains. Reusable assets can be

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

also being built in-house, retrieved from legacy systems or can be bought from an external source.

2. PRESENT METHOD IN THE DATA INTEGRATION PROCESSION

Existing methods to software integration of information for TTT, process cover a wide spectrum of data encoding methods and search or matching algorithms. The en-coding methods differ with respect to their soundness, completeness, and the extent to which they support an estimate of the effort it takes to modify a data. Text-based encoding and integration is neither sound nor complete. Its disadvantages have been thoroughly in the information integration literature [5, 6]. Lexical descriptor-based encoding method also suffers from a number of problems about developing and using classification vocabulary [7]. Software specific challenges include the fact that one-word or one-phrase abstractions are hard to come by in the software domain [8]. From the user's point of view, lack of familiarity with the vocabulary is also pointed out as draw back in using a integration of information for TTT, system effectively [9]. In this context Data feature model will be a promising solution for integration of information for TTT, process [10].

3. METHODS USED

It is an algebraic model in which documents and queries are represented as data features as follows:

dj = (w1, j, w2,j, ... ,wt, j)

q = (w1,q, w2, q, ... ,wt, q)

Each dimension corresponds to a separate term. If a term occurs in the document, its value in the data feature is non-zero. An indexed collection of documents is represented as a term table which has documents as fields and words as primary key for row. The (D)i (Word)j-th entry of this table records how many times the j-th search term appeared in the i-th document.

The first major data of a data feature space search model is the concept of a term space. A term space consists of every unique word that appears in a collection of documents. The second major data of a data feature space search model is term counts. Term counts are simply records of how many times each term occurs in an individual document. This is represented as a table. By using the term space as a coordinate space, and the term counts as coordinates within that space, we can create a data feature for each document. As the number of terms in-creases, the dimensionality of DFM also increases.

For these words documents and corresponding ranks will be stored in the rank table. Based on the ranking terms are compared as "ranked higher than", "ranked lower than" or "ranked equal to" the second, making it possible to evaluate complex information according to query criteria. Here search data feature space search model ranks the documents it finds according to the estimation of their relevance, making it possible for the user quickly to select the data unit according to their requirements [7].

Relevancy rankings of documents in a keyword search can be calculated, using the assumptions of document similarities theory, by comparing the deviation of angles between each document data feature and the original query data feature where the query is represented as same kind of data feature as the documents.

It is easier to calculate the cosine of the angle between the data features instead of the angle:

$$\cos\theta = \frac{d_2 \cdot q}{\|d_2\|\|q\|}$$

A cosine value of zero means that the query and document data feature are orthogonal and have no match (i.e. the query term does not exist in the document being considered).

The Document Classification Algorithm

KSP classifier is an instance-based learning algorithm that is based on a distance function for pairs of observations, such as the Euclidean distance or Cosine. The K-Shortest Path (KSP) classifier algorithm has been studied extensively for text categorization by Yang and Liu [6]. In this classification paradigm, k shortest paths of a training data are computed first. Then the similarities of one sample from testing data to the k shortest paths are aggregated according to the class of the paths, and the testing sample is assigned to the most similar class. The similarity in score of each path document to the test document is used as the weight of the categories of the path document [8]. If there are several training documents in the k shortest path, which share a category, the category gets a higher weight. In this work, we used the Cosine distance to calculate the similarity score for the document representation.

One of advantages of KSP is that it is well suited for multi-modal classes as its classification decision

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

is based on a small path of similar objects (i.e., the major class). So, even if the target class is multi-modal (i.e., consists of objects whose independent variables have different characteristics for different subsets), it can still lead to good accuracy. A major drawback of the similarity measure used in KSP is that it uses all features equally in computing similarities. This can lead to poor similarity measures and classification errors, when only a small subset of the features is useful for classification [5].

Steps for KSP Using Average Cosine:

Step 1: Select k nearest training documents, where the similarity is measured by the cosine between a given testing document and a training document.

Step 2: Using cosine values of k shortest paths and frequency of documents of each class i in k shortest paths, compute average cosine value for each class i, Avg_Cosine (i).

Step 3: Classify the testing document a class label which has largest average cosine.

In order to reduce the dimensionality of DFM and keep useful information, we first compute concept data features for given categories. Then, using the concept data features as projection matrix, projection of both training and testing data is done. Finally, we apply KSP algorithm on the projected DFM model that has reduced dimensionality.

Steps of Combined Method for Data feature Based Algorithm and K-Shortest Path Algorithm:

Step 1: Compute a concept data feature for each category using true label information of training documents and then construct concept data feature matrix C (w-by-c), where c is the number of categories.

Step 2: Do projection of DFM model A (w-by-d) using concept data feature matrix C (w-by-c) (i.e., C^T * A).

Step 3: Apply KSP with the projected DFM model (i.e., c-by-d matrix).

4. A SYSTEM FOR QUERYING A WAREHOUSE WITH DFM

With the idea of DFM and other associated notions just presented, we have created a system for reasoning about the information content of data whereby to help derive information in a warehouse by drawing on Wang and Feng [6] and Eessaar [8]. Intuitively, the system works like this.

Let us reiterate that to select a student from table Students is seen as a random event, and the term "particulars of a random event" is used to describe a single occurrence of a random event. For example, student John's record happens to be selected from table Students, and this particular occurrence of John's record being selected is a "particular" of the random event that the record happens to be John's. A random variable may be seen as an aggregation of random events. In a table, an attribute can be seen as a random variable because it normally contains many random events in it. For example, Student Name is a random variable, which contains Student Name being John and Student Name being Herman, among others. The DFM closure of Student ID being B001, for example, contains Student Name being "John", Student Major being "history" and Class Name being "BD445". If a user queries about the class name about John, the query can be answered by searching in this DFM closure of Student ID being B001. That is, once DFM closures are known, queries can be checked against these closures. This way some information that cannot be found by conventional queries may be discovered.

Figure 1 illustrates the information base for Warehouse queries. It consists of three main parts. The upper part is where users pose queries to the data. The middle part is the Data-log implementation of Formulation of Data. The lower part shows a Relations DFM Closures as the Information Base for warehouse Queries, namely domain knowledge and the syntactic and semantic properties of the warehouse that are inherent to it.

The form of the queries is the conventional SQL. Most programming efforts were made on computing the DFM closures. The core algorithm is based on the DFM rules. Original DFMs were then added into the unit. This is one of the most difficult tasks in the programming required for the construction of the system as when more original DFM were discovered more computation capability has to be added into the program such that the closures can continually increase accordingly. The output of the unit is simple however, which are DFM closures. User queries, then, are checked against these closures. Thus, more in-formation can be discovered through queries.

The process of discovering original DFMs could be hard. There is a variety of source out there that could potentially contain huge amount of original DFM [7]. The two main sources though are domain knowledge and the properties of the warehouse per se. The latter can be further divided into those of

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

semantic and syntactic levels respectively. Hereinto, the syntactic level includes plenty of constraints such as data dependencies, integrity rules and the cardinality ratio between tables.

We now wish to demonstrate how the experimental system was created using DFM. The previous version of the system was coded in Oracle's PL/SQL [8]. It is now coded in the deductive warehouse language, Data-log First, we show how our DFM inference rules may be implemented by using Data-log in order to make use of the deduction power of it.

5. CONCLUSIONS AND FUTURE WORK

In this paper, we have proposed a novel approach to the information content of data in a warehouse. We gave a set of basic concepts and described an experimental system that makes use of this notion. A number of examples were used to test our system. With information sources outside a warehouse imported into the system, the information content of a random event (data values) within the data-base expanded dramatically. Users could make the most of the information content of data by posing queries. Thus, more information can be discovered

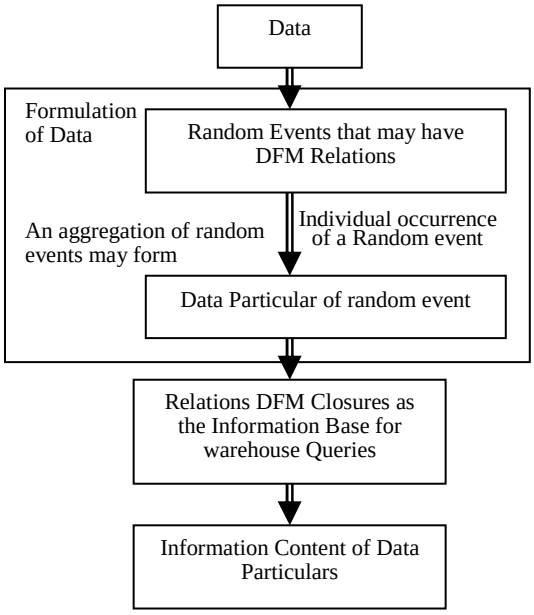

Figure 1. The Base For Warehouse Queries

than conventional queries. The increase of random events' closures is based on the boost in original DFM and the inference capability using DFM. Identification of original DFM rules could be hard due to wide range of sources outside warehouse. However, once original DFM have been identified and then integrated into the computing unit of our system, the system provides a powerful engine for users to query a warehouse. Our experiment shows that with the DFM inference capability hidden information within warehouse can be discovered with the increase of original DFM derived from warehouse itself and external sources.

With DFM rules, we discussed the relation of information content inclusion between random events. Such a relation at a higher level, i.e., that between random variables requires more work. How the relations on different levels are connected also deserves further investigation. The process of identifying original DFM was done manually, for which a semi-automated technique making use of meta-data to suit the need of a user is desirable and looks feasible. Moreover, how to approach and inference about the information content of data that are stored in independent and yet inter-operating warehouses should be investigated.

In summary, our work thus far seems to have shown that the information that a warehouse can potentially provide is definable by using the notion of Data Feature Model (DFM). Furthermore, the inference rules for formally reasoning about such a relation enables the development of a seemingly elegant way, by means of DFM closure, of identifying the information content of data in an warehouse, which serves as a basis for answering queries.

A novel KSP classification algorithm combining model and evidence theory is proposed in this paper. The new method not only overcomes the main shortage of lazy learning in traditional KSP, but also takes the distances between samples to be recognized and samples in k-paths into account. At the same time the method resolves the unrecognizable cases of unknown samples. Applying the classification algorithm into the document recognition, experimental results show its satisfied recognition rate and fast categorization speed.

There are also models based on and extending the data feature model such as generalized data feature model, Topic-based Data feature model and latent semantic indexing etc and also combination algorithms which consist of clustering, Singular Value Decomposition (SVD)-based Algorithm, Naive Bayesian Algorithm and variations of KSP algorithm. Future work can be aimed in these directions.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

AKNOWLEDGEMENT

This paper is supported by the national natural fund of China （Grant Number：61063003）

REFRENCES:

[1] M. C. Huebscher and J. A. McCann, "A Survey of Autonomic Computing—Degrees, Models, and Applications," ACM Computing Surveys, Vol. 40, No. 3, 2008, pp. 1-28. doi:10.1145/1380584.1380585

[2] J. O. Kephart, "Research Challenges of Autonomic Computing," Proceedings of 27th International Conference on Software Engineering, St. Louis, 15-21 May 2005, pp. 15-22.

[3] Purwoharjono, Muhammad Abdillah, "Optimal Placement of TCSC Using Linear Decreasing Inertla Weight Gravitational Search Algorithm," Journal of Theoretical and Applied Information Technology, Vol. 47. No. 2, 2013, pp 460-470.

[4] G.V.Nadiammal and M.Hemalatha, "an Enhanced Rule Approach for Network Intrusion Detection Using Efficient Data Adapted Decision Tree Algorithm," Journal of Theoretical and Applied Information Technology, Vol. 47. No. 2, 2013 , pp. 426 - 433 .

[5] Ruo hu, Channel Access Controlling in Wireless Sensor Network using Smart Grid System, Applied Mathematics & Information Sciences, No. 6-3S (Nov. 2012), PP:813-820.

[6] Ruo hu, Stability Analysis of Wireless Sensor Network Service via Data Stream Methods, Applied Mathematics & Information Sciences, No. 6-3S (Nov. 2012), PP:793-798.

[7] Ruo hu, New Network Access Control Method Using Intelligence Agent Technology, Applied Mathematics & Information Sciences, (Mar. 2013).

[8] M. Aldinucci, M. Danelutto and P. Kilpatrick, "Towards Hierarchical Management of Autonomic Components: A Case Study," 17th Euromicro International Conference on Parallel, Distributed and Network-Based Processing, Weimar, 18-20 February 2009, pp. 3-10. doi:10.1109/PDP.2009.48

[9] T Mukherjee, A Banerjee, G. Varsamopoulos and S. K. S. Gupta. "Model-Driven Coordinated Management of Data Centers," Computer Networks, Vol. 54, No. 16, 2010, pp. 2869-2886. doi:10.1016/j.comnet.2010.08.011

[10] J. O. Kephart, H. Chan, R. Das, D. W. Levine, G. Tesauro and F. R. A. C. Lefurgy, "Coordinating Multiple Autonomic Managers to Achieve Specified Power-Performance Tradeoffs," Proceeding of the 4th International Conference on Autonomic Computing, Dublin, 13-16 June 2006, pp. 145-154.