Business Intelligence for Enterprise Systems: A Survey

Lian Duan and Li Da Xu, *Senior Member, IEEE*

***Abstract*—Business intelligence (BI) is the process of transforming raw data into useful information for more effective strategic, operational insights, and decision-making purposes so that it yields real business benefits. This new emerging technique can not only improve applications in enterprise systems and industrial informatics, respectively, but also play a very important role to bridge the connection between enterprise systems and industrial informatics. This paper was intended as a short introduction to BI with the emphasis on the fundamental algorithms and recent progress. In addition, we point out the challenges and opportunities to smoothly connect industrial informatics to enterprise systems for BI research.**

***Index Terms*—Business intelligence (BI), industrial informatics, enterprise systems, data mining.**

I. Introduction

THE TRADITIONAL industrial informatics focus on how to provide more efficient and productive operations. However, nowadays they cannot stay competitive any longer just by providing more efficient and productive operations. They are facing the challenges of processing huge amounts of data and turning it into smart and timely decisions to deliver better products and services. In order to deliver the useful information for decision-making, business intelligence (BI) is the key technologies [65] for users to efficiently extract useful information from oceans of data. The concept of BI was firstly introduced by Garter Group [38], and incipiently referred to the tools and technologies including data warehouses, reporting query and analysis. Now, BI is regarded as a key approach to increasing the value of the enterprise. The BI software market constantly grows and experiences a compound annual growth rate of 9.7% to reach 10.7 billion dollars through 2011 [21].

Besides abundant applications in enterprise systems [98]–[101], BI has been used in industrial informatics. For example, Kusiak and Song [55], [82] used it to improve combustion efficiency, and Wang [91] detected outliers to make on-spec product. Generally speaking, BI is used either to make a prediction for the dynamic environment or extract useful patterns by outlier detection, process mining, and clustering. We can see the significant boosting BI research in industry informatics in the recent three years according to Fig. 1, which shows the number of related papers published in the Transactions on Industrial Informatics (TII) each year. We classified all the 30 related papers in TII according to the taxonomy we used in Section II and in Table I. Currently, most research related to BI in industrial informatics is for quality control. Since there is a trend of the automatic process for enterprise systems, most of the related applications will be integrated with enterprise systems for decision making and reduce manual intervention in the future.

As we can see, BI is increasingly important for companies to stay competitive and provides many research opportunities. It not only has many applications in industrial informatics and enterprise systems, but also plays an important role to connect industrial informatics and enterprise systems. In this paper, we present a comprehensive survey of the state-of-the-art in BI in

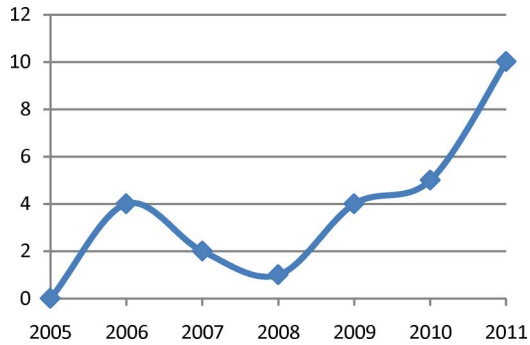

Fig. 1. The number of related papers in TII each year.

TABLE I
Papers in Each Category

	Supervised Learning	Unsupervised Learning
Papers	Kusiak et al., 2006; Acciani et al., 2006; Song et al., 2007; Gong et al., 2008; Lo et al., 2009; Zhou et al., 2009; Picon et al., 2009; Ning et al., 2009; Wang et al., 2010; Ghasemzadeh et al., 2011; Gomperts et al., 2011; Chan et al., 2011; Pan et al., 2011; Efe et al., 2011; Silva et al., 2011; Orlowska et al., 2011; Acampora et al., 2011; Kafai et al., 2012; Si et al., 2012; Tan et al., 2012	Ali et al., 2006; Quan et al., 2006; Kusiak et al., 2006; Song et al., 2007; Lin et al., 2007; Gong et al., 2008; Lo et al., 2009; Zhang et al., 2010; Ma et al., 2010; Shin et al., 2010; Filev et al., 2010; Tsai et al., 2011; Wang, 2011; Tsai et al., 2012

Manuscript received December 26, 2011; revised February 04, 2012; accepted February 15, 2012. Date of publication February 23, 2012; date of current version July 23, 2012. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 71132008, in part by the Changjiang Scholar Program of the Ministry of Education of China, and in part by the U.S. National Science Foundation under Grant 1044845. Paper no. TII-11-1061.

L. Duan is with the Department of Information Systems, New Jersey Institute of Technology, Newark, NJ 07102 USA (e-mail: duanlian.cn@gmail.com).

L. D. Xu is with Old Dominion University, Norfolk, VA 23529 USA, and also with the Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China (e-mail: lxu@odu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TII.2012.2188804

1551-3203/$31.00 © 2012 IEEE

Section II. Then, we identify various limitation of the current generation of BI methods and future research opportunities in this area in Section III, and Section IV summarizes this paper.

II. Business Intelligence

It is widely accepted that the technology categories of BI mainly encompass data warehouse or data mart, On-Line Analytical Processing (OLAP), and data mining [92]. More specifically, data warehouse or data mart are the fundament infrastructure to save historical data, and data mining is its core component to detect trends, identify patterns and analyzes data, while OLAP is the set of front-end analyzing tools. With respect to data warehouse and OLAP, we can treat them as components for next-generation database systems. Comparing to developing software packages, researchers in industrial informatics and enterprise systems tend to be more interested in applying BI in the industrial environment. Since the purpose of this paper is to provide the research insights of using BI, we focus on the survey on data mining, which detects interesting patterns and provides research opportunities in industrial informatics.

Here, we categorize all the data mining techniques into two big groups: supervised learning methods and unsupervised learning methods. Supervised learning methods build a model to predict a specified unknown attribute according to the observed attributes, while unsupervised learning methods extract patterns, such as clusters, process graphs, and correlated items from data. With respect to supervised learning, more accurate prediction is the golden goal. Although there are lots of ongoing researches for different supervised learning methods, the gain of selecting a proper method is usually much bigger than the gain of improving a given method. Since our goal is to improve the industrial informatics and enterprise systems applications rather than doing research on data mining area, we focus on introducing benchmark algorithms and providing guidelines for selecting the right type of supervised learning algorithms. Since better unsupervised learning algorithms can find better patterns, we focus on introducing the state-of-art unsupervised learning algorithms. Supervised learning includes Decision Tree, Bayesian Statistics, Neural Network, Support Vector Machine, Regression, and Nearest Neighbors, while unsupervised learning includes Itemset Mining, and Clustering.

A. Supervised Learning

All the supervised learning methods can be used for nominal prediction, while only neural network, regression model, and the nearest neighbor method can be used for numeric prediction.

1) Decision Tree: Decision Tree algorithms recursively select an attribute to partition records into smaller subsets to each branch in a top-down manner. The most important step is how to select an attribute for splitting, and three benchmark measures, Information Gain [75], Gain Ratio [76], and Gini Index [12], are used. Information Gain calculates the difference between the original entropy and the new entropy after the partition on a selected attribute, which tells us to what degree the attribute can improve the purity of partitions. However, information gain favors attributes with unique values for each record because such attributes partition the whole dataset into subsets containing only one record. Although each subset is pure, this model is overfitting and cannot be used for prediction. Gain Ratio uses the split entropy to regulate Information Gain, and Gini Index use another way to measure how fuzzy a given set is. Generally speaking, Gain Ratio favors unbalanced partition, and Gini Index is not good for large number of classes.

Although Decision Tree can be applied to numeric attributes, it adds a layer of complexity of how to discretize the value into different ranges and is not recommended for a data set with numeric values. Decision Tree can also be used for numerical prediction, but it has to integrate regression model to achieve that goal. The integration will be introduced in the Regression section. If all the attributes in the data set are nominal and it is for nominal prediction, the performance of Decision Tree is usually good if not the best, and it is easy to interpret. In addition, Decision Tree by itself involves the feature selection procedure; therefore, we don't need to preprocess data by feature selection.

2) Bayesian Statistics: Bayesian classifiers make use of the Bayes' theorem for prediction. For data sets with many attributes, it is extremely expensive to compute the conditional probability. Therefore, the naïve Bayesian assumes attributes are independent from each other. Since the assumption of independence is usually violated in practical applications, Bayesian network is proposed to solve this problem. It allows dependencies between attributes and constructs a directed acyclic network to calculate the joint conditional probability. The most challenge part is how to construct the network. It can be specified by human or learned from data [72].

Bayesian classifiers are supported by statistic theories and the best if the assumptions are satisfied. Therefore, we must make sure attributes are independent from each other before using naïve Bayesian. For Bayesian network, the learned topology from data is usually awkward and not working well. Currently, if the topology is not too complicated, the best way of constructing it is by human. If the network can be correctly constructed by human, its performance [51] can surpass the state-of-art classifiers like Neural Network and SVM. Bayesian classifiers can handle both nominal and numeric attributes well for nominal prediction. Generally speaking, Bayesian classifiers are not recommended unless users are experts on them. It involves lots of human efforts to satisfy the assumption of Bayesian theorem to get the best result.

3) Neural Network: Neural Network is composed of connected artificial neurons with a weight associated with each edge. The basic structure has three layers: input, hidden, and output. Although there are many different neural network algorithms, the most popular one is backpropagation [17]. It iteratively adjusts the weight on each edge by comparing the prediction with the actual value. Since the modification is made in the backward direction, it is called backpropagation. Basically, neural network generates the value for numeric prediction. However, by using the same transforming function as logistic regression, neural network can also predict binary value. If the hidden layer is removed and the structure only contains input and output layers, the Neural Network is equivalent to the linear regression model. The Neural Network is equivalent to the polynomial regression model if the hidden layer only has one layer, and it can make any nonlinear separation if the hidden layer has two layers. Therefore, two hidden layers is enough for any type

TABLE II
SUPERVISED METHOD SUMMARY (O INDICATES PARTIAL SUPPORT; X INDICATES FULL SUPPORT)

Methods	Attribute		Output		Important Issue
	Numeric	Nominal	Numeric	Nominal	
Decision Tree	O	X	O	X	Overfitting
Bayesian Statistics	X	X		X	Network Construction
Neural Network	X	X	X	X	Overfitting
Support Vector Machine	X	X		X	None
Regression	X	X	X	X	Regression Type Selection
Nearest Neighbor	X	X	X	X	Feature Selection

of prediction and we don't have any gain by adding more hidden layers than two.

Comparing to decision tree and Bayesian classifiers, it is hard to explain how results are produced. However, Neural Network usually provides better results than decision tree and Bayesian classifiers. If users are only interested in the prediction accuracy and don't care how the decision is made, Neural Network is a good choice [53]. While the training of Neural Network is extremely time-consuming, the advantage is that it usually only needs thousands of records to train a good model. Therefore, it is very useful if the user only has a small dataset. In addition, Neural Network is well-suited for both numeric and nominal attributes while decision tree handles nominal attributes better than numeric attributes.

4) Support Vector Machine: Support Vector Machine searches for the hyperplane that separates two classes most. It was originally proposed for a linear separation [90]. Later on, Boser *et al.* [11] used a non-linear kernel function to map the original data into a higher linear separable dimension space. Three commonly used kernel functions are polynomial function, Gaussian radial basis function, and Sigmoid function. Lin *et al.* [58] found the relationship between non-linear SVM and two layer neural network. There is no guarantee that non-linear SVM predicts better than linear SVM because it depends on the characteristics of the data set. Sometimes, users need to try different types of SVM to find the best model for their data.

Both SVM and neural network are considered as the best classifiers with respect to the accuracy. The SVM can only be applied for nominal prediction but less prone to overfitting than other methods. If users want to get high accurate nominal prediction and get round with the overfitting issues, SVM is a good choice.

5) Regression: The regression models have the long history in the statistical area and are widely applied to any research area dealing with data. They estimate the conditional expected change of the predicted value given the observed attribute. The linear regression model is the basic model. The general linear model can handle the correlation between attributes. Logistic regression model is specialized for binary prediction, and multivariate adaptive regression splines [35] deals with non-linearity and interactions. The regression models can be applied to any applications because they are able to do any types of prediction and handle any type of data. However, the regression models are not specially designed for prediction purposes. The accuracy performance is generally not as good as Neural Network and SVM.

6) Nearest Neighbor: The nearest neighbor classifier learns the target value by comparing a record with the historical record similar to it. There are two important issues related to the performance of nearest neighbor: distance function and feature selection. If the absolute value is important such as temperature in boilers, Euclidean distance is recommended. If the absolute value is not important but the angle is such as the word frequency in a document, Cosine distance is recommended. Any distance-based function intrinsically assigns equal weights to each attribute. In other words, irrelevant attributes degrade the accuracy heavily and feature selection can improve the accuracy of nearest neighbor a lot. The nearest neighbor classifier doesn't need to build a model beforehand, but the time spending on prediction is expensive.

7) Advanced Issues: We summarize different supervised learning methods in Table II. In addition to the difference among different methods, there are two important general issues for supervised learning methods: overfitting and ensemble. A supervised learning is overfitting when it is more accurate in the training data but less accurate in new data. A lot of supervised learning methods adapt themselves to outliers if the learning was performed too long. In order to avoid overfitting, the most commonly used technique is cross-validation, where a portion of data is used only for evaluation and not used for training. Other techniques to prevent overfitting include regularization, early stopping, and pruning [45], [84], which are specially designed for different supervised learning methods. Ensemble uses several different models for better predictions. The ensemble can be considered as a committee where each model is a committee member. The majority vote usually makes better decision than any single model. Two commonly used techniques are bagging [13] and boosting [36]. Bagging uses a randomly-drawn subset to build different models, while boosting trains each new model with emphasis on the mis-classified data from previous models. Bagging is less likely to be overfitting than boosting because bagging involves the randomness while boosting focuses more on the mis-classified data which might be an outlier. If the data are noisy, bagging can generate better results; otherwise, boosting is better.

B. Unsupervised Learning

1) Itemset Mining: The first research on itemset mining is frequent itemset mining which searches sets of items appearing together frequently in a data set by the Apriori algorithm [6]. The Apriori algorithm makes use of an important downward-closed property to prune the exponential superset search space.

A property is downward-closed if for every set with this property, all its subsets also have this property. Without making use of this property, it is even impossible to search all the frequent itemsets for a dataset only with 100 items. Later on, a lot of different frequent itemset search algorithms were proposed, such as FP-growth [46] and ECLAT [95]. A recent progress on this issue is to formulate this problem as constraint programming problems, and then apply an existing solver to constraint programming for speeding up the search [44]. Those algorithms will speed up the search procedure; however, a more important issue related to frequent itemsets is its effectiveness. Therefore, the concept of maximal frequent itemsets [18] was proposed to reduce redundant information. The shortcoming of maximal frequent itemsets will not help the generation of association rules. Therefore, Pasquier *et al.* introduced the closed itemsets [71]. An itemset is closed if none of its supersets has the same support which is the occurrence.

No matter how the above itemset types are defined, they are simply related to the support of the itemset. As a metric, the support facilitates fast search, but it has drawbacks [15]. For example, if A and B both occur in 90% of the records, the finding of A and B occurring together 81% of the records is not interesting. Most statistical tests over A and B reflect the lack of dependence between them, even though A and B together have very high support. Therefore, the concept of correlated itemsets attracts more attention recently and there are several challenges on this topic. First, there are numerous measures [39], [85] available for evaluating correlation, but many of them provide conflicting information. Second, we are generally interested in correlated itemsets with arbitrary size, but most of the published work with regard to correlation is related to finding correlated pairs. Related work with association rules is a special case of correlated pairs since each rule has a left- and right-hand side. Given an association rule $X \Longrightarrow Y$, $\text{Support} = P(X \cap Y)$ and $\text{Confidence} = P(X \cap Y)/P(X)$ are often used to represent its significance. However, these can produce misleading results because of the lack of comparison to the expected probability under the assumption of independence. In order to overcome the shortcoming of support, Lift [15], Conviction [16], and Leverage [77] are proposed. $\text{Lift} = P(X \cap Y)/(P(X)P(Y))$ measures the ratio of X and Y's actual co-occurrence to the expected value under the independence assumption. $\text{Conviction} = (P(X)P(\bar{Y}))/P(X \cap \bar{Y})$ compares the probability that X appears without Y if they were independent with the actual frequency of the appearance of X without Y. $\text{Leverage} = P(X \cap Y) - P(X)P(Y)$ measures the difference of X and Y appearing together in the data set and what would be expected if X and Y were statistically independent. Dunning introduced a more statistically reliable measure, Likelihood Ratio [31], which outperforms other correlation measures in text mining. Jermaine [48] extended Dunning's work and examined the computational issue of Probability Ratio and Likelihood Ratio. Bate [10] proposed a correlation measure called Bayesian Confidence Propogation Neural Network (BCPNN) which is good at searching for the correlated patterns occurring rarely in the whole dataset.

The above correlation measures are intuitive; however, different correlation measures provide drastically different results. Given an itemset $S = \{I_1, I_2, \ldots, I_m\}$, Shapiro [77] proposed the three mandatory correlation properties for a correlation measure M to satisfy. The three mandatory correlation properties help rule out some bad measures like support, all-confidence [68], bond [68], IS (Tan, 2000). However, other bad measures like the simplified chi-squre, probability ratio still satisfy all the three mandatory correlation properties. Tan [86] also proposed several correlation properties, and the purpose of the proposed properties by Tan is to categorize the correlation measures which generate the similar results. In addition, the proposed properties by Tan can only be used for the pair-only correlation measures. Recently, Duan *et al.* [23], [28] studied the upper and lower bound of different correlation measure and proposed three desired correlation properties to rule out the remaining bad correlation measures.

Among 19 different correlation measures examined by Duan, only four measures, leverage, likelihood ratio, the simplified chi-square with continuity correction, and BCPNN, satisfy all the mandatory and desired correlation properties [23]. Experimental results also show that the above four measures usually search better patterns than other measures which violate mandatory or desired correlation properties.

Much previous research focuses on finding correlated pairs instead of correlated itemsets in which all items are correlated with each other. However, there are some applications in which we are specifically interested in correlated itemsets rather than correlated pairs. For example, we are interested in finding sets of correlated stocks in a market, or sets of correlated gene sequences in a microarray experiment. But finding correlated itemsets is much harder than finding correlated pairs because of two major problems. First, computing correlation for each possible itemset is an NP-complete problem [48]. Second, if there are some highly correlated items within an itemset and the rest are totally independent items, all the correlation measures still indicate that the itemset is correlated. No existing measure provides information to identify the itemsets with independent items. Comparing to correlation measures, support is not effective but the search is fast for high dimension because of its natural downward-closed property. An alternate is to find the top-k correlated itemset. The correlation measures can search for more meaningful correlated patterns than support, but do not have the downward-closed property to reduce the computational expense. To sum up, support is efficient but not effective, while correlation measures are effective but not efficient. In order to solve the problem, other trade-off measures like all-confidence [68] were proposed. All-confidence is as fast as support; however, the correlation is still measured in a sub-optimal way. Requiring the correlation measure to be both effective and efficient is too demanding. Instead of proposing another efficient correlation measure, Duan *et al.* [24], [28] proposed a fully-correlated itemsets framework, in which any two subsets are correlated. This framework can not only decouple the correlation measure from the need for efficient search, but also rules out the itemsets with irrelevant items. With it, we only need to focus on effectiveness when selecting correlation measures.

Another computational issue related to the correlated itemset search is the performance of the correlated pair search. Unlike finding frequent itemsets which can start pruning from 1-itemsets, searching fully-correlated itemsets must start pruning

from pairs. As the number of items and records in a dataset increases, calculating the correlation values for all the possible pairs is computationally expensive. The most significant progress in speeding up correlated pair search was made by Xiong *et al.* [93], [94]. He made use of the upper bound of the Pearson correlation coefficient. The computation of this upper bound is much cheaper than the computation of the exact correlation because this upper bound is a function of single item supports. In addition, the upper bound has special 1-dimensional and 2-dimensional properties that prune many pairs from the search space without the need to computer their upper bounds. However, this work is only related to the Pearson correlation coefficient. Duan *et al.* [24] extended this work to any correlation measure that satisfies the three mandatory properties, and proposed a Token-ring algorithm to speed up the search. By combining the fully-correlated itemset framework, Token-ring algorithm, and the selected correlation measure likelihood ratio, we are able to find the more useful correlated patterns on a large dataset.

2) Clustering: Clustering assigns records into groups. A lot of clustering algorithms have been proposed. No matter how the algorithms are designed, their goal can be summarized into two objectives: (1) The objects in the same group are similar to each other. (2) The objects in different groups are dissimilar to each other. Different types of clustering algorithms strike a different balance between the two objectives. The widely-used clustering algorithms have three categories: partitioning, hierarchical, and density-based clustering.

The partitioning clustering uses an iterative relocation procedure to move objects from one group to another. The most famous clustering algorithm K-means [62] is a partitioning clustering algorithm. It randomly selects k objects as centers and assigns objects to its closest center. When all the partitions are made, it recalculates the center of each partition and reassigns objects to the new centers. The algorithm stops when the k centers will not move any more. However, K-means tends to partition the whole space to equal size in order to minimize the global variance and the value of k is usually hard to determine. In order to solve this problem, a more robust Expectation-Maximization clustering algorithm [22] was proposed. The goal is to estimate the parameters of possible mixing Gaussian distributions. It alternatively estimates the expected probability for each object as a member of each group, and alters the parameters of each group to maximize those probabilities. The EM clustering algorithm has many successful applications in computer vision and natural language processing; however, it has the assumption that clusters are spherical-shaped and this assumption holds for any partitioning clustering algorithm. Therefore, partitioning clustering algorithms have trouble to detect arbitrary shape clusters.

A hierarchical clustering method groups objects in a way that can be represented in a tree structure. The hierarchical tree is constructed either in a bottom-up (merging) or top-down (splitting) fashion. AGNES [49] merges closest group through a single-linkage approach in which the similarity is measured by the closest pair between two groups. In general, the complexity of hierarchical clustering is not less than $O(n^3)$ and it is too expensive for large datasets. Only for some special cases, some efficient algorithms can be $O(n^2)$, such as SLINK [80] for the single-linkage. Another problem of hierarchical clustering is that the wrong decision made in the previous merge or split step affects the following steps. However, hierarchical clustering is capable of finding clusters of arbitrary shapes which cannot be discovered by a partitioning clustering method.

Although hierarchical clustering can find arbitrary shape clusters, it only considers cluster proximity yet ignores cluster interconnectivity, and an outlier is still assigned to the closest cluster. To discover clusters with arbitrary shape and outliers, density-based clustering methods have been developed. These typically regard clusters as dense regions of objects in the data space separated by regions of low. DBSCAN [33] was the first density-based algorithm which grows clusters according to a density-based connectivity analysis. Another density-based algorithm DENCLUE [47] clusters objects based on a set of density distribution functions. Both DBSCAN and DENCLUE use a global density parameter to find clusters. However, a common property of many practical data sets is that their intrinsic cluster structures cannot be characterized by global density parameters. As a result, very different local densities may be needed to reveal clusters in different regions of the data space. Therefore, OPTICS [9] was proposed to extend DBSCAN to produce a cluster ordering obtained from a wide range of parameter settings. OPTICS only considers the absolute distance from its closest neighbor without considering the neighborhood density. Sometimes, an outlier closer to dense region will be considered as a member of the cluster [29]. LDBSCAN [30] combines the concepts of DBSCAN and LOF [14] to discover clusters and outliers. LOF uses a more meaningful way to assign to each object a degree of being an outlier than to consider being an outlier as a binary property. There are two potential benefits of combining clustering and outlier detection: increasing precision and facilitating data understanding. LDBSCAN is able to detect not only arbitrary shape and different density clusters, but also clusters that reside in other clusters.

Same as nearest neighbor for prediction, the distance function is very important to clustering; however, the more important issue is feature selection, especially for high dimensional clustering. For prediction problem, feature selection is relatively easy because we only need to search the feature related to the class. For clustering, different groups of features need to be found for a different partition. Take oil, water, nature gas, and vapor for example, both oil and water are liquid, while nature gas and vapor are gas if we use their physical attributes. However, from the energy perspective, oil and nature gas are energy source while water and vapor are not. Different clustering algorithms are proposed to handle this problem, such as axis-parallel subspace clustering, pattern-based clustering, and arbitrarily-oriented subspace clustering [54]. Different types use different strategies to search for combinations of features. Since there are no generally accepted criteria to guide the search, the high dimensional clustering is still an open research issue.

III. Future Research Opportunities

A lot of research progresses have been made recently in BI. However, there are more emerging issues to challenge the traditional methods. In this section, we identify the related opportu-

nities in industrial informatics and enterprise systems to apply and improve the existing BI methods.

A. Outlier Detection

Outlier detection refers to finding patterns that do not conform to what are expected. People are interested in outlier detection because outliers can be very useful actionable information in many applications. It has been extensively studied in industry informatics. One third papers [34], [42], [57], [60], [63], [78], [88], [89], [91], [96] are related to BI in TII focus on this topic.

Outlier detection can be achieved by both supervised and unsupervised methods. The easiest way of identifying outliers by supervised methods is to build a model to a specified attribute and find the record whose actual value contradicts most to the predicted value. Any supervised method can be used this way. If we have the dataset which has labeled records for normal and abnormal classes, we can build a predictive model for normal vs. abnormal classes. There are several related issues for the above way of using supervised methods for outlier detection. First, it is not easy to get the labeled data for abnormal class. Second, both normal and abnormal behaviors keep evolving and the existing predictive model might not be representative enough in the future. Second, the number of normal and abnormal records is usually unbalanced which causes a problem for most predictive models [20]. The predictive model will be adjusted according to different costs of different wrong predictions. For example, the cost of omitting a real attack is higher than that of preparing for a wrong one. However, in real life, the unlabeled data are much easier to get and behaviors keep evolving; therefore, unsupervised methods are more widely used. Such unsupervised methods have three widely used types: statistical distribution-based, distance-based, and density-based. The statistical distribution based methods try to use a distribution to model the data. However, they have trouble in modeling the high dimensional data. The distance-based methods [52] assume that outliers are far away from the most records or its nearest neighbors in the whole dataset. Sometimes, in a normal low density region, normal points have larger absolute distance from its neighbors and that of an outlier deviating from a density region. Therefore, the density-based methods are proposed to handle this type of problems. The most successful density-based outlier detection method is local outlier factor (LOF) [14]. It calculates the ratio of deviation to its neighborhood density to indicate the abnormal degree. LOF can only be used for single point outliers. Duan *et al.* [29] combined LOF and DBSCAN for cluster outlier detection. The advantage of combining LOF and DBSCAN can successfully detect small clusters with objects less than ten, which is hard to be achieved by traditional clustering methods and make this algorithm especially suitable for cluster outlier detection. Same as high dimensional issue for clustering, the feature selection is very important for high dimensional outlier detection and it is an increasingly hot area of outlier detection. A related concept for high dimensional outlier detection is contextual outliers. Traditional techniques focus on single point outliers and treat attributes independently. However, a record could be abnormal due to a specific context [37], [83]. For example, the temperature of 0° is normal in the winter but abnormal in the summer. Such outliers can only be detected by specifying some attributes as contextual attributes to divide the whole dataset into smaller partitions.

B. Graph Data

The traditional data mining methods treat objects independently in a data set. However, in many applications objects are usually related to each other and can be more naturally represented in a graph model, where each object is a node and there is an edge between two nodes if they interact with each other. For example, Devices communicate with each other to performance a certain function. The industry companies collaborate with each other to produce a product. The modern science of graphs has significantly helped us understand complex systems. This type of graph data gains an increasingly attention from BI research community and pose a lot of challenging questions that cannot be solved directly by traditional methods. The linkage analysis, community detection, and graph summarization are extremely hot research topics.

Linkage analysis helps us to find the potential partners in an industry social network or establish a link between two procedures to improve the productivity in an industrial procedure network. The linkage prediction can be achieved both in supervised [56] or unsupervised [23] framework. The supervised learning has to deal with the imbalance and sparse connections, while unsupervised learning searches for an effective measure according to the characteristics of different networks.

Since nodes in the graph are not isolated from each other, detecting nodes that work together help use to understand the whole system. Two commonly-used methods are spectral-based [61] and modularity-based [66]. Spectral-based methods use the eigenvectors of the adjacency matrix for community detection, and the change of the representation induced by eigenvectors makes the community structure more obvious. Modularity-based methods make use of the modularity function to search better partition. Duan *et al.* [26] found the connection between the modularity function and the correlation measure Leverage. By using other correlation measures like likelihood ratio to change the objectives, the new method can detect better small communities.

One typical graph summarization is process mining [1], [7], [43], which distill a structured process description from system logs. It answers many important industrial or business process questions. For example, what is the real workflow here? Do we do what we agree on? Where are the system bottlenecks? How to redesign the process to improve the efficiency? In addition, Aalst *et al.* [2] constructed an annotated transition system from the historical data in the ProM framework [3] and extend its function to predict how long a given process will take.

C. Environmental Issues

The special industrial environment-related issues, such as distributed and continuous data, also challenge the tradition BI methods. For example, sensor networks are widely used in industry today. Each sensor has very limited computing power, and widely distributed in different locations. We require the light data mining algorithm for those units to gather useful information, and a component to coordinate the distributed data mining.

When monitoring industrial process, we receive data continuously. The pattern search conducted on the entire data cannot provide the timely result. Instead, we need the algorithm which can incrementally update the existing model rather than the one that starts from scratch.

IV. Summary

Since BI can be applied to most of the applications that generate data, it is impossible to list all the possible applications in the area of industrial informatics and enterprise systems. Instead, we introduce the fundamental methods to readers, and let them choose the proper method for their applications. In addition, BI plays an important role to connect industrial informatics and enterprise systems. Ning *et al.* [67] published their research in TII to predict retailer demand and adjust their inventory replenishment correspondingly. This research significantly reduces operation costs and improves retailer satisfaction. However, it is not the only application to connect industrial informatics and enterprise systems. Take product design for example, companies can gather the demographic information of the customer who purchased the product from enterprise systems. Itemset mining on the above data can search the correlation between product features and customer demographic attributes. The extracted pattern helps the product design for a target customer group. In addition, companies can also conduct the opinion mining [59] from the online customer reviews for their products. Opinion mining can detect the market reaction and search the product features that customers like and dislike from the unstructured text reviews. These information help companies promptly alternate the product design and the produced amount. In all, BI has a bright future and deserves the research attention from industrial informatics and enterprise systems.

References

[1] W. Aalst, *Process Mining: Discovery, Conformance and Enhancement of Business Processes*, 1st ed. New York: Springer, 2011.

[2] W. Aalst, M. H. Schonenberg, and M. Song, "Time prediction based on process mining," *Inform. Syst.*, vol. 36, no. 2, pp. 450–475, 2011.

[3] W. Aalst, B. F. van Dongen, C. W. Günther, R. S. Mans, A. K. Alves de Medeiros, A. Rozinat, V. Rubin, M. Song, H. M. W. Verbeek, and A. J. M. M. Weijters, "ProM 4.0: Comprehensive support for real process analysis," in *Lecture Notes in Computer Science*. Berlin, Germany: Springer-Verlag, 2007, vol. 4546, pp. 484–494.

[4] G. Acampora, J. M. Cadenas, V. Loia, and E. M. Ballester, "Achieving memetic adaptability by means of agent-based machine learning," *IEEE Trans. Ind. Informat.*, vol. 7, no. 4, pp. 557–569, Nov. 2011.

[5] G. Acciani, G. Brunetti, and G. Fornarelli, "Application of neural networks in optical inspection and classification of solder joints in surface mount technology," *IEEE Trans. Ind. Informat.*, vol. 2, no. 3, pp. 200–209, Aug. 2006.

[6] R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases," in *Proc. ACM SIGMOD Int. Conf. Manage. Data, SIGMOD'93*, New York, 1993, pp. 207–216.

[7] L. Aldin and S. Cesare, "A literature review on business process modelling: New frontiers of reusability," *Enterprise Inform. Syst.*, vol. 5, no. 3, pp. 359–383, Aug. 2011.

[8] S. Ali, B. Soh, and T. Torabi, "A novel approach toward integration of rules into business processes using an agent-oriented framework," *IEEE Trans. Ind. Informat.*, vol. 2, no. 3, pp. 145–154, Aug. 2006.

[9] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "OPTICS: Ordering points to identify the clustering structure," in *Proc. ACM SIGMOD Int. Conf. Manage. Data, SIGMOD'99*, New York, pp. 49–60, ACM.

[10] A. Bate, M. Lindquist, I. R. Edwards, S. Olsson, R. Orre, A. Lansner, and R. M. De Freitas, "A Bayesian neural network method for adverse drug reaction signal generation," *Eur. J. Clinical Pharmacology*, vol. 54, no. 4, pp. 315–321, 1998.

[11] B. E. Boser, I. M. Guyon, and V. N. Vapnik, D. Haussler, Ed., "A training algorithm for optimal margin classifiers," in *Proc. 5th Annu. ACM Workshop on COLT*, Pittsburgh, PA, 1992, pp. 144–152.

[12] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. London, U.K.: Chapman & Hall, 1984.

[13] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, 1996.

[14] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying density-based local outliers," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Dallas, TX, May 15–18, 2000, pp. 93–104.

[15] S. Brin, R. Motwani, and C. Silverstein, "Beyond market baskets: Generalizing association rules to correlations," in *Proc. ACM SIGMOD Int. Conf. Manage. Data, SIGMOD'97*, New York, 1997, pp. 265–276, ACM.

[16] S. Brin, R. Motwani, J. D. Ullman, and S. Tsur, "Dynamic itemset counting and implication rules for market basket data," in *Proc. ACM SIGMOD Int. Conf. Manage. Data, SIGMOD'97*, New York, 1997, pp. 255–264, ACM.

[17] A. E. Bryson and Y.-C. Ho, *Applied Optimal Control: Optimization, Estimation, and Control*. London, U.K.: Blaisdell, 1969, p. 481.

[18] D. Burdick, M. Calimlim, J. Flannick, J. Gehrke, and T. Yiu, "MAFIA: A maximal frequent itemset algorithm," *IEEE Trans. Knowl. Data Eng.*, pp. 1490–1504, 2005.

[19] K. Y. Chan, T. S. Dillon, and C. K. Kwong, "Modeling of a liquid epoxy molding process using a particle swarm optimization-based fuzzy regression approach," *IEEE Trans. Ind. Informat.*, vol. 7, no. 1, pp. 148–158, Feb. 2011.

[20] D. A. Cieslak and N. V. Chawla, IEEE Computer Society, Washington, DC, USA, "Start globally, optimize locally, predict globally: Improving performance on imbalanced data," in *Proc. 8th IEEE Int. Conf. Data Mining, ICDM'08*, 2008, pp. 143–152.

[21] N. Davey, KnowledgeBoard, "Productivity concerns driving business intelligence growth—Gartner," Feb. 2011. [Online]. Available: http://www.knowledgeboard.com/item/3119/23/5/3, Online

[22] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Royal Statist. Soc.*, ser. B, vol. 39, Methodological, no. 1, pp. 1–38, 1977.

[23] L. Duan and W. N. Street, "Selecting the right correlation measure for binary data," Working Paper, Univ. Iowa, Iowa City, IA, 2011.

[24] L. Duan and W. N. Street, "Speeding up correlation search for binary data," Working Paper, Univ. Iowa, Iowa City, IA, 2011.

[25] L. Duan, M. Khoshneshin, and W. N. Street, "Adverse drug effect detection," Working Paper, Univ. Iowa, Iowa City, IA, 2011.

[26] L. Duan and W. N. Street, "Community detection through correlation," in *Proc. INFORMS Annu. Meeting*, 2011.

[27] L. Duan, W. N. Street, and E. Xu, "Healthcare information systems: Data mining methods in the creation of a clinical recommender system," *Enterprise Inform. Syst.*, vol. 5, no. 2, pp. 169–181, Apr. 2011.

[28] L. Duan and W. N. Street, "Finding maximal fully-correlated itemsets in large databases," in *Proc. IEEE Int. Conf. Data Mining*, Nov. 2009, pp. 770–775.

[29] L. Duan, L. Xu, Y. Liu, and J. Lee, "Cluster-based outlier detection," *Ann. Oper. Res.*, vol. 168, no. 1, pp. 151–168, Apr. 2009.

[30] L. Duan, L. Xu, F. Guo, J. Lee, and B. Yan, "A local-density based spatial clustering algorithm with noise," *Inform. Syst.*, vol. 32, no. 7, pp. 978–986, Nov. 2007.

[31] T. Dunning, "Accurate methods for the statistics of surprise and coincidence," *Comput. Linguistics*, vol. 19, no. 1, pp. 61–74, 1993.

[32] M. O. Efe, "Neural network assisted computationally simple PID control of a quadrotor UAV," *IEEE Trans. Ind. Informat.*, vol. 7, no. 2, pp. 354–361, May 2011.

[33] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, E. Simoudis, J. Han, and U. M. Fayyad, Eds., "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowl. Discovery and Data Mining, AAAI*, Menlo Park, CA, 1996, pp. 226–231.

[34] D. P. Filev, R. B. Chinnam, F. Tseng, and P. Baruah, "An industrial strength novelty detection framework for autonomous equipment monitoring and diagnostics," *IEEE Trans. Ind. Informat.*, vol. 6, no. 4, pp. 767–779, Nov. 2010.

[35] J. H. Friedman, "Multivariate adaptive regression splines," *Ann. Statist.*, vol. 19, no. 1, pp. 1–67, 1991.

[36] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," in *Proc. 2nd Eur. Conf. Comput. Learning Theory (EuroCOLT'95)*, London, U.K., 1995, pp. 23–37.

[37] J. Gao, F. Liang, W. Fan, C. Wang, Y. Sun, and J. Han, "On community outliers and their efficient detection in information networks," in *Proc. 16th ACM SIGKDD Int. Conf. Knowl. Discovery and Data Mining*, Washington, D.C., Jul. 25–28, 2010.

[38] Gartner Group, "How secure is your business intelligence environment," pp. 1–6, 2002. [Online]. Available: http://www.gartner.com/id=367264

[39] L. Geng and H. J. Hamilton, "Interestingness measures for data mining: A survey," *ACM Comput. Surveys*, vol. 38, no. 3, p. 9, 2006.

[40] H. Ghasemzadeh and R. Jafari, "Physical movement monitoring using body sensor networks: A phonological approach to construct spatial decision trees," *IEEE Trans. Ind. Informat.*, vol. 7, no. 1, pp. 66–77, Feb. 2011.

[41] A. Gomperts, A. Ukil, and F. Zurfluh, "Development and implementation of parameterized FPGA-based general purpose neural networks for online applications," *IEEE Trans. Ind. Informat.*, vol. 7, no. 1, pp. 78–89, Feb. 2011.

[42] R. Gong, S. H. Huang, and T. Chen, "Robust and efficient rule extraction through data summarization and its application in welding fault diagnosis," *IEEE Trans. Ind. Informat.*, vol. 4, no. 3, pp. 198–206, Aug. 2008.

[43] D. Grigori, F. Casati, M. Castellanos, U. Dayal, M. Sayal, and M. Shan, "Business process intelligence," *Comput. Ind.*, vol. 53, no. 3, pp. 321–343, Apr. 2004.

[44] T. Guns, S. Nijssen, and L. De Raedt, "Itemset mining: A constraint programming perspective," *Artif. Intell.*, vol. 175, pp. 1951–1983, Aug. 2011.

[45] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. San Mateo, CA: Morgan Kaufmann, 2011.

[46] J. Han, J. Pei, Y. Yin, and R. Mao, "Mining frequent patterns without candidate generation," *Data Mining and Knowl. Discovery*, vol. 8, pp. 53–87, 2004.

[47] A. Hinneburg and D. A. Keim, "An efficient approach to clustering in large multimedia databases with noise," in *Proc. 4th Int. Conf. Knowle. Discovery and Data Mining*, New York, 1998, pp. 58–65.

[48] C. Jermaine, "Finding the most interesting correlations in a database: How hard can it be?," *Inform. Syst.*, vol. 30, no. 1, pp. 21–46, 2005.

[49] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*. New York: Wiley, 1990.

[50] M. Kafai and B. Bhanu, "Dynamic Bayesian networks for vehicle classification in video," *IEEE Trans. Ind. Informat.*, vol. 8, no. 1, pp. 100–109, Feb. 2012.

[51] M. Khoshneshin and W. N. Street, "A latent factor graph for cancer diagnosis using microarray gene expression data," in *Proc. 6th INFORMS Workshop on Data Mining and Health Informatics, DM-HI'11*, 2011, pp. 1–6.

[52] E. M. Knorr and R. T. Ng, "Algorithms for mining distance-based outliers in large datasets," in *Proc. VLDB*, 1998, pp. 392–403.

[53] T. O. Kowalska and M. Kaminski, "FPGA implementation of the multilayer neural network for the speed estimation of the two-mass drive system," *IEEE Trans. Ind. Informat.*, vol. 7, no. 3, pp. 436–445, Aug. 2011.

[54] H. Kriegel, P. Kröger, and A. Zimek, "Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering," *ACM Trans. Knowl. Discov.*, vol. 3, no. 1, pp. 1–58, Mar. 2009, Article 1.

[55] A. Kusiak and Z. Song, "Combustion efficiency optimization and virtual testing: A data-mining approach," *IEEE Trans. Ind. Informat.*, vol. 2, no. 3, pp. 176–184, Aug. 2006.

[56] R. Lichtenwalter, J. T. Lussier, and N. V. Chawla, "New perspectives and methods in link prediction," in *Proc. KDD*, 2010, pp. 243–252.

[57] C. Lin and C. Liu, "Failure detection and adaptive compensation for fault tolerable flight control systems," *IEEE Trans. Ind. Informat.*, vol. 3, no. 4, pp. 322–331, Nov. 2007.

[58] H. T. Lin and C. J. Lin, Dep. Comput. Sci., National Taiwan Univ., Taiwan, China, "A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods" Tech. Rep., 2003. [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/papers/tanh.pdf

[59] B. Liu, "Sentiment analysis: A multifaceted problem," *IEEE Intell. Syst.*, vol. 25, no. 3, pp. 76–80, 2010.

[60] C. H. Lo, E. H. K. Fung, and Y. K. Wong, "Intelligent automatic fault detection for actuator failures in aircraft," *IEEE Trans. Ind. Informat.*, vol. 5, no. 1, pp. 50–55, Feb. 2009.

[61] U. Luxburg, "A tutorial on spectral clustering," *Statist. Comput.*, vol. 17, no. 4, pp. 395–416, Dec. 2007.

[62] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th Berkeley Symp. Math. Statist. Probability*, 1967, pp. 281–297.

[63] M.-D. Ma, D. S.-H. Wong, S.-S. Jang, and S.-T. Tseng, "Fault detection based on statistical multivariate analysis and microarray visualization," *IEEE Trans. Ind. Informat.*, vol. 6, no. 1, pp. 18–24, Feb. 2010.

[64] W. McCulloch and W. Pitts, "A logical calculus of ideas immanent in nervous activity," *Bull. Math. Biophys.*, vol. 5, pp. 115–133, 1943.

[65] L. T. Moss and S. Atre, *Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications*. Reading, MA: Addison-Wesley, 2003.

[66] M. E. J. Newman, "Detecting community structure in networks," *Eur. Phys. J. B - Condensed Matter and Complex Systems*, vol. 38, no. 2, pp. 321–330, Mar. 2004.

[67] A. Ning, H. Lau, Y. Zhao, and T. T. Wong, "Fulfillment of retailer demand by using the MDL-optimal neural network prediction and decision policy," *IEEE Trans. Ind. Informat.*, vol. 5, no. 4, pp. 495–506, Nov. 2009.

[68] E. R. Omiecinski, "Alternative interest measures for mining associations in databases," *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, no. 1, pp. 57–69, 2003.

[69] T. Orlowska-Kowalska and M. Kaminski, "FPGA implementation of the multilayer neural network for the speed estimation of the two-mass drive system," *IEEE Trans. Ind. Informat.*, vol. 7, no. 3, pp. 436–445, Aug. 2011.

[70] T.-H. Pan, B.-Q. Sheng, D. S.-H. Wong, and S.-S. Jang, "A virtual metrology system for predicting end-of-line electrical properties using a MANCOVA model with tools clustering," *IEEE Trans. Ind. Informat.*, vol. 7, no. 2, pp. 187–195, May 2011.

[71] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal, "Discovering frequent closed item sets for association rules," in *Proc. ICDT*, 1999, pp. 398–416.

[72] J. Pearl, *Causality: Models, Reasoning, and Inference*. : Cambridge University Press, 2000.

[73] A. Picon, O. Ghita, P. F. Whelan, and P. M. Iriondo, "Fuzzy spectral and spatial feature integration for classification of nonferrous materials in hyperspectral data," *IEEE Trans. Ind. Informat.*, vol. 5, no. 4, pp. 483–494, Nov. 2009.

[74] T. T. Quan, S. C. Hui, and A. C. M. Fong, "Automatic fuzzy ontology generation for semantic help-desk support," *IEEE Trans. Ind. Informat.*, vol. 2, no. 3, pp. 155–164, Aug. 2006.

[75] J. R. Quinlan, "Induction of decision trees," *Mach. Learn*, vol. 1, no. 1, pp. 81–106, 1986, Mar. 1986.

[76] J. R. Quinlan, *C4.5: Programs for Mach. Learn*. : Morgan Kaufmann Publishers, 1993.

[77] G. P. Shapiro, *Discovery, Analysis, and Presentation of Strong Rules*. Cambridge, MA: MIT Press, 1991.

[78] S. Shin, T. Kwon, G.-Y. Jo, Y. Park, and H. Rhy, "An experimental study of hierarchical intrusion detection for wireless industrial sensor networks," *IEEE Trans. Ind. Informat.*, vol. 6, no. 4, pp. 744–757, Nov. 2010.

[79] Y. Si, J. Mei, and H. Gao, "Novel approaches to improve robustness, accuracy and rapidity of iris recognition systems," *IEEE Trans. Ind. Informat.*, vol. 8, no. 1, pp. 110–117, Feb. 2012.

[80] R. Sibson, "SLINK: An optimally efficient algorithm for the single-link cluster method," *Comput. J. (British Comput. Soc.)*, vol. 16, no. 1, pp. 30–34, 1973.

[81] D. De Silva, X. Yu, D. Alahakoon, and G. Holmes, "A data mining framework for electricity consumption analysis from meter data," *IEEE Trans. Ind. Informat.*, vol. 7, no. 3, pp. 399–407, Aug. 2011.

[82] Z. Song and A. Kusiak, "Constraint-based control of boiler efficiency: A data-mining approach," *IEEE Trans. Ind. Informat.*, vol. 3, no. 1, pp. 73–83, Feb. 2007.

[83] X. Song, M. Wu, C. Jermaine, and S. Ranka, "Conditional anomaly detection," *IEEE Trans. Knowl. Data Eng.*, vol. 19, no. 5, pp. 631–645, 2007.

[84] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*. Reading, MA: Addison-Wesley, 2005.

[85] P.-N. Tan, V. Kumar, and J. Srivastava, "Selecting the right objective measure for association analysis," *Inform. Syst.*, vol. 29, no. 4, pp. 293–313, 2004.

[86] P. N. Tan and V. Kumar, "Interestingness measures for association patterns: A perspective," in *Proc. Workshop on Postprocessing in Mach. Learn. Data Mining, KDD'00*, 2000.

[87] W. R. Tan, C. S. Chan, P. Yogarajah, and J. Condell, "A fusion approach for efficient human skin detection," *IEEE Trans. Ind. Informat.*, vol. 8, no. 1, pp. 138–147, Feb. 2012.
[88] D.-M. Tsai and J.-Y. Luo, "Mean shift-based defect detection in multicrystalline solar wafer surfaces," *IEEE Trans. Ind. Informat.*, vol. 7, no. 1, pp. 125–135, Feb. 2011.
[89] D.-M. Tsai, I.-Y. Chiang, and Y.-H. Tsai, "A shift-tolerant dissimilarity measure for surface defect detection," *IEEE Trans. Ind. Informat.*, vol. 8, no. 1, pp. 128–137, Feb. 2012.
[90] V. Vapnik and A. Lerner, "Pattern recognition using generalized portrait method," *Autom. Remote Control*, vol. 24, pp. 774–780, 1963.
[91] D. Wang, "Robust data-driven modeling approach for real-time final product quality prediction in batch process operation," *IEEE Trans. Ind. Informat.*, vol. 7, no. 2, pp. 371–377, May 2011.
[92] H. J. Watson and B. Wixom, "The current state of business intelligence," *IEEE Computer*, pp. 96–99, Sep. 2007.
[93] H. Xiong, W. J. Zhou, M. Brodie, and S. Ma, "Top-kϕ correlation computation," *INFORMS J. Comput.*, vol. 20, no. 4, pp. 539–552, 2008.
[94] H. Xiong, S. Shekhar, P.-N. Tan, and V. Kumar, "TAPER: A two-step approach for all-strong-pairs correlation query in large databases," *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 4, pp. 493–-508, 2006.
[95] M. J. Zaki, "Scalable algorithms for association mining," *IEEE Trans. Knowl. Data Eng.*, vol. 12, no. 3, pp. 372–390, May/Jun. 2000.
[96] Y. Zhang, H. Zhou, S. J. Qin, and T. Chai, "Decentralized fault diagnosis of large-scale processes using multiblock kernel partial least squares," *IEEE Trans. Ind. Informat.*, vol. 6, no. 1, pp. 3–10, Feb. 2010.
[97] J. H. Zhou, C. K. Pang, F. L. Lewis, and Z. W. Zhong, "Intelligent diagnosis and prognosis of tool wear using dominant feature identification," *IEEE Trans. Ind. Informat.*, vol. 5, no. 4, pp. 454–464, Nov. 2009.
[98] R. Perez-Castillo, B. Weber, J. Pinggera, S. Zugal, I. G.-R. de Guzman, and M. Piattini, "Generating event logs from non-process-aware systems enabling business process mining," *Enterprise Inform. Syst.*, vol. 5, no. 3, pp. 301–335, Aug. 2011.
[99] Z. Gong, M. Muyeba, and J. Guo, "Business information query expansion through semantic network," *Enterprise Inform. Syst.*, vol. 4, no. 1, pp. 1–22, Feb. 2010.
[100] D. M.-H. Chiang, C.-P. Lin, and M.-C. Chen, "The adaptive approach for storage assignment by mining data of warehouse management system for distribution centres," *Enterprise Inform. Syst.*, vol. 5, no. 2, pp. 219–234, May 2011.
[101] B. Wetzstein, P. Leitner, F. Rosenberg, S. Dustdar, and F. Leymann, "Identifying influential factors of business process performance using dependency analysis," *Enterprise Inform. Syst.*, vol. 5, no. 1, pp. 79–98, Feb. 2011.

Lian Duan received the Ph.D. degree in information systems with emphasis on business intelligence from the University of Iowa, in 2012, and the Ph.D. degree in computer science from the Chinese Academy of Sciences, Beijing, in 2007.

His main research interests include correlation analysis, community detection, and clustering.

Li Da Xu (M'86–SM'11) received the M.S. degree in information science from the University of Science and Technology of China, Hefei, in 1981, and the Ph.D. degree in systems sciences with emphasis on information systems from Portland State University, Portland, OR, in 1986.

Dr. Xu serves as the Founding Chair of IFIP TC8 WG8.9 and the Founding Chair of the IEEE SMC Society Technical Committee on Enterprise Information Systems.