

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800

Open access books available

122,000

International authors and editors

135M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.

For more information visit www.intechopen.com



Applying a Hybrid Data Mining Approach in Machining Operation for Surface Quality Assurance

Tzu-Liang (Bill) Tseng, Yongjin Kwon and Ryan B. Wicker

1. Introduction

Conventionally, the quality of a machined product has been measured based on the specifications, once the machining process is complete. However, this post-process inspection has several shortcomings: (1) it is difficult to isolate the causes of the defect; (2) the manufacturing cost has already been incurred when a non-conformance is detected; (3) rework of any scope increases the manufacturing cost and can be very difficult to accomplish; and (4) there could be a significant time lag between the detection of the defects and subsequent corrective actions. Today, efforts of manufacturers are shifting from the post-process inspection to improved monitoring of the manufacturing processes, utilizing sensors and other measurement devices, to effectively control the process. Improvements in machining precision can only be accomplished by the development of manufacturing systems that are capable of monitoring processes. Process monitoring reduces scrap, rework, lead-time, and conventional non value-added inspection activities, thereby, increases the system's productivity. The monitoring has to be based on sound, reliable process control algorithms. Computer numerical control (CNC) of machine tools do help to produce consistent part quality. However, in most cases, CNC machines don't utilize sensor data to compensate for anomalies generated by the cutting processes (e.g., tool wear, chatter, incorrect machine setup, etc.). If sensors such as cutting force, vibration and spindle motor current were integrated into CNC machine tools, the control functions should be able to interpret and respond to sensory data as the process continues. However, when many process variables need to be considered, it becomes rather difficult to predict quality attributes in machining (i.e., surface roughness).

To solve the aforementioned prediction problems, especially with the consideration of negative information and data to improve prediction accuracy, two data mining approaches have been developed. Here, negative information or

data represent the set of data points that do not conform to the conventional modeling techniques but it can be used to facilitate quality prediction. The approaches involve both individual and population based paradigms, such as a rough set theory (RST) and SVMs with the negative information & data training. To validate the proposed approach, one case of the perdition problem related to the surface roughness is applied. Literature review suggests that the hybrid approach of combined individual and population based paradigms has not been widely applied, thus making this research novel. By using the hybrid approach, the following objectives can be attained: (1) search the minimal number of features, or rules, for decision making in prediction; (2) aggregate the weight of the feature and frequency of the object to search the optimal rules; (3) simultaneously identify the outcomes and significant features in prediction; and (4) achieve a high prediction accuracy through the application of negative information & data. In this context, this study uses a hybrid data mining approach to identify variables affecting the quality characteristic of CNC machining operations. Instead of predicting exact surface roughness values, the focus is on the prediction of quality acceptance in machined parts. The hybrid approach is an effective tool for multi-attribute classification problems. This can be instrumental in constructing intelligent control systems, especially when a clear delineation within variables as to how they affect the surface roughness is difficult to achieve.

2. Theoretical Background of Data Mining and Hybrid Approach

Data mining is a process of extracting and refining knowledge from large databases (Berry & Linoff, 1997; Dhar & Stein, 1997; Cheung *et al.*, 1996). It is a process that uses a variety of data analysis tools to discover the patterns and relationships in the data. The extracted information can be used to predict, classify, model, and summarize the data being mined. Data mining, a major step in knowledge discovery from databases, involves the application of specific algorithms for identifying interesting structures in data, where the structure designates patterns, statistical, or predictive models from the data as well as the relationships among parts of the data (Fayyad & Uthurusamy, 2002). Data mining is also an emerging area of computational intelligence that offers new theories, techniques, and tools for processing large volumes of data. The growing volume of data available in digital format has accelerated this interest.

Basically, data mining approaches can be categorized into two different cases. One is called "individual based" while the other is called "population based" paradigm (Kusiak, 2001(a); Kusiak & Kurasek, 2001). There are fundamental differences between the two approaches. The individual based approach generates a number of models (usually in the form of decision rules) capturing relationships between the input features and the decision. In other words, the individual based approach identifies unique features of an object and finds whether they are shared with other objects. The population based approach creates a model based on a training data set. The model normally uses a predetermined set of features (Kusiak, 2001(a); Kusiak & Kurasek, 2001). For example, the rule induction approach follows a data object paradigm, while neural networks and SVMs can be viewed as a single model that is formed for the entire population (training data set). The models (rules) created by the rule induction are explicit. The "population based" tools determine features that are common to a population (training data set) (Kusiak, 2001(b); Kusiak & Kurasek, 2001). The deficiency of the individual based approach for prediction is that low accuracy decision rules cannot be used, and the quality rules with a high accuracy do not guarantee to be used since the condition part of the rule should match with the input domain of the testing data sets. Consequently, the limitations of the rule based prediction can be easily observed, and the population based data mining approaches are able to counteract this deficiency.

In general, the material for learning is given in a positive form. This type of information will help organize the core of the target knowledge. Instead of this type of information, negative information will help sharpen the edge or extent of the target knowledge. Hence, it is expected that the negative information will have an effect of minimizing the chance of making errors and thus making the learning faster (Kurosu & Ookawa, 2002). In the data mining domain, negative information/data, which is defined as information/data, misclassified the outcomes from the testing data set and is normally discarded (Chen et al. 2004 (b)). However, the information/data is possible to be re-used for the training purpose and contains a positive impact on the prediction accuracy (Chen *et al.*, 2004 (b)). To date, there is little literature related to using data mining approaches to predict surface roughness with the consideration of utilizing negative information/data. To conduct the individual and population based data mining approaches that are to be applied in the preferred supplier prediction, the two classification approaches: RST and SVMs are reviewed.

2.1 Rough Set Theory

RST has been applied to address a variety of problems (Ziarko, 1993), including (1) representation of uncertain or imprecise knowledge; (2) empirical learning and knowledge acquisition from experience; (3) knowledge analysis; (4) analysis of conflicting data; (5) quality evaluation of the available information with respect to its consistency and presence or absence of repetitive data patterns; (6) identification and evaluation of data dependencies; and (7) approximation of pattern classification. In RST, data is expressed in a decision table, in which each row represents an object and each column represents an attribute. Formally, the decision table is represented by an information function (Pawlak, 1991) in the form of:

$$S = \langle U, Q, V, f \rangle \quad (1)$$

where U = a finite set of objects, Q = finite set of attributes, $V = \bigcup_{q \in Q} V_q$ and $V_q =$ domain of the attribute q , and $f: U \times Q \rightarrow V$ = the total decision function such that $f(x, q) \in V_q$ for every $q \in Q, x \in U$.

The main theme of RST is concerned with measuring what may be described as the “ambiguity” inherent in the data. The essential distinction is made between objects that may definitely be classified into a certain category, and those that may possibly be classified. Considering all decision classifications yields to what is referred to as the “quality of approximation” that measures the proportion of all objects for which definite classification may be achieved. A rough set can be described as a collection of objects that in general cannot be precisely characterized in terms of their values of their sets of attributes, but can be characterized in terms of lower or upper approximations. The upper approximation includes all objects that possibly belong to the concept, while the lower approximation contains all objects that definitely belong to the concept. As each object is characterized with attributes, discovering dependencies between attributes and detecting main attributes is of primary importance in RST. Attribute reduction is one unique aspect of the rough set approach. A reduct is a minimal sufficient subset of attributes, which provides the same quality of discriminating concepts as the original set of attributes. For example, let’s consider the five objects in Table 1, each with four input features (F1-F4) and an output feature.

| Object No. | Features | | | | Output |
|---|----------|----|----|----|--------|
| | F1 | F2 | F3 | F4 | |
| 1 | 1 | 0 | 1 | 2 | 2 |
| 2 | 1 | 1 | 0 | 3 | 1 |
| 3 | 1 | 0 | 0 | 0 | 0 |
| 4 | 0 | 2 | 2 | 1 | 0 |
| 5 | 1 | 1 | 1 | 0 | 2 |
| 0: Not Applicable, 1: Low, 2: Medium, 3: High | | | | | |

Table 1. Example Data Set

To derive reducts, consider the first feature F_1 . The set of objects corresponding to the feature value $F_1 = 1$ is $\{1, 2, 3, 5\}$. This set $\{1, 2, 3, 5\}$ cannot be further classified solely using the relation $F_1 = 1$. It is discernible over the constraint $F_1 = 1$, which is expressed as $[x][F_1 = 1] = \{1, 2, 3, 5\}$. For the objects in the set $\{1, 5\}$, the output feature is $O = 2$, for the object 3, the output feature is $O = 0$ and for the object 2, the output feature is $O = 1$. Therefore, additional features are needed to differentiate between $O = 0, 1$, or 2. Applying this concept, the classification power of each feature can be evaluated. For instance, the feature value $F_1 = 0$ is specific to $O = 0$. This discernible relation can be extended to multiple features, e.g., $[x][F_1 = 1] \wedge [F_2 = 0] = \{1, 3\}$ and $[x][F_1 = 1] \vee [F_2 = 0] = \{1, 2, 3, 5\}$, where \wedge and \vee refers to "or" and "and", respectively.

Reduct Generation

Most rough set based approaches generate more than one reduct for an object. This paper adapts the reduct generation procedure proposed by Pawlak (1991) and presents it in the form of "reduct generation procedure," as illustrated in Figure 1. The reduct generation procedure enumerates all possible reducts with one, two and three features that are presented in Table 2.

- Step 1: Set object number $i = 1$.*
- Step 2: Select object i and find a set of reducts with 1 to $(m - 1)$ features.*
- Step 3: Set $i = i + 1$. If all objects have been considered, go to Step 3; otherwise go to Step 1.*
- Step 4: Terminate the algorithm and output the result.*

Figure 1. Reduct Generation Procedure

| Object No. | F ₁ | F ₂ | F ₃ | F ₄ | O | Reduc No. | U | F ₁ | F ₂ | F ₃ | F ₄ | O |
|------------|----------------|----------------|----------------|----------------|---|-----------|---|----------------|----------------|----------------|----------------|---|
| 1 | 1 | 0 | 1 | 2 | 2 | 1 | 1 | x | x | 1 | x | 2 |
| | | | | | | 2 | | x | x | x | 2 | 2 |
| | | | | | | 3 | 2 | 1 | x | 1 | x | 2 |
| | | | | | | 4 | | 1 | x | x | 2 | 2 |
| | | | | | | 5 | | x | 0 | 1 | x | 2 |
| | | | | | | 6 | | x | 0 | x | 2 | 2 |
| | | | | | | 7 | | x | x | 1 | 2 | 2 |
| | | | | | | 8 | 3 | 1 | 0 | 1 | x | 2 |
| | | | | | | 9 | | 1 | 0 | x | 2 | 2 |
| | | | | | | 10 | | x | 0 | 1 | 2 | 2 |

Table 2. Partial Reducts for Data in Table 1

2.2 Support Vector Machines

SVMs based on the statistical theory have been developed as the tools for classification, regression, and density estimation in noisy data (Vapnik, 1998). There are three significant features in SVMs. The first is the generalization theory, which leads to a structure risk minimization (SRM) model. The generalization error is from either the model or hypothesis space. The SRM model improves the generalization ability through extending the margins in the feature space. The second is the kernel functions, which maps non-linear system into a linear feature space without explicitly requiring an exact map function. SVMs computational problem is connected with the size of the feature space. This makes SVMs perform efficiently over neural networks. The last feature is that the parameters are found by solving a quadratic programming problem with linear equality and inequality constraints, which return the global optimal solution. By doing so, the estimation errors can be minimized.

SVMs are designed for a binary classification. Generally, there are two types of approaches for a multi-class classification. One is that multi-class problems have been tackled by indirectly by combining a series of binary problems. Another is considering all data in one optimization formulation. Several methods based on the combining approach are one-versus-rest, one-versus-one, and DAG (Directed Acyclic Graph) SVMs methods (Platt et al., 2000). Using the SVMs in the one-versus-rest fashion is very common, but it has potential drawbacks when classes overlap considerably. It constructs k SVMs, where k is

the number of classes. The i^{th} SVM is trained by the i^{th} class associated with a positive label and all other examples with negative labels. The predication of an example x is determined by the maximum margin of k SVMs.

One-versus-one method is combining the binary SVMs for all pairs of classes. The DAG SVMs algorithm is a pair wise approach that exhibits large variability since each binary classifier is estimated from a small subset of the training data. It allows only a simple cost structure when different misclassification costs are concerned. As a generic approach to multi-class problems, treating all the classes simultaneously is considered. Although several extensions to the multi-class cases have been proposed (Vapnik, 1998; Bredensteiner & Bennett, 1999), its optimal extension was not obvious in relation to the theoretically best classification rule. The DAG SVMs and one-versus-one have good practical performance than the other methods (Hsu & Lin, 2002). In the DAG SVMs and one-versus-one, the training phase of k -classes classification problem is completed by $k(k-1)/2$ binary SVMs. In the testing phase, DAG SVMs uses a root binary acyclic graph with k leaves, where each node is a binary SVM. To test an example x , testing begins with a root node along with the DAG to reach a leaf node. The testing of one-versus-one is using a voting approach. The result of predicating is the largest vote number. The advantage of DAG SVM is to cut down the testing time as compared to the one-versus-one method.

From the review, RST application and SVMs appear to be both robust and efficient in automatic classification. Furthermore, the methods that automatically generate diagnostic rules have shown to have a significant aim in decision making of prediction. In this paper, the concept of feature extraction, cluster analysis, and SVMs model are used to develop a methodology for aiding the preferred supplier selection. Motivation for conducting combination of RST and SVMs is the hybrid approach capable of performing significant feature identification (dimension reduction), noise elimination (object reduction), and learning from negative information/data to improve prediction accuracy. Moreover, the hybrid approach is the combination of the individual and population based data mining approaches that are able to overcome low accuracy of prediction and other limitation. The methodology development is introduced next.

3. Methodology Development

This section illustrates the development procedure of methodology. The overall concept has been represented in Figure 2.

3.1 Rule Identification Algorithm

The rule identification algorithm is incorporated with the weights. Measurement of the weight is based on domain experts' judgment and external assessment. Basically, each feature has been considered and the "ratio estimates" method assigns the weight of each feature without any bias. Moreover, adjustment of the weights assigned through pair wise comparisons is also required. Frequency of each object is derived from the original database during a certain period. All of the weights, which include feature and frequency domains, are subjected to normalization.

In Table 3, the weight value of the feature is taken into consideration and subject to a positive normalized value, which is between 0 and 1. It can be obtained from domain experts' judgment. With each column of incidence matrix $A = [a_{ij}]_{m \times n}$, frequency f_i for reduct i , $i = 1, \dots, m$ and weight w_j for input feature j , $j = 1, \dots, n$ are associated. It is possible to assign frequency f_i , $i = 1, \dots, m$ as a different kind of weight since the frequency of each object can be derived from the original database. Furthermore, it is also possible to assign weights w_j , $j = 1, \dots, n$ since the weight of each feature can be determined from domain experts. Table 3 is with a column indicating the number of objects and a row containing weights associated with the features. Using the weight coefficients w_j and f_i , an auxiliary matrix $[e_{ij}]$ will be generated from the original reduct – input feature matrix.

The weight coefficient assigned to each feature is denoted as w_j and each object (reduct) as f_i . Using the weight coefficients w_j and f_i , an auxiliary matrix $[e_{ij}]$ will be generated from the original reduct – input feature matrix. The entries of the transformed matrix are defined as follows:

$$e_{ij} = f_i \times (w_j \times v_j) \quad (2)$$

where e_{ij} = entry of the transformed reduct-input feature matrix, f_i = weight of reduct I , w_j = weight of feature j , and $v_j = 1$ if feature $j \neq "x"$; 0 otherwise.

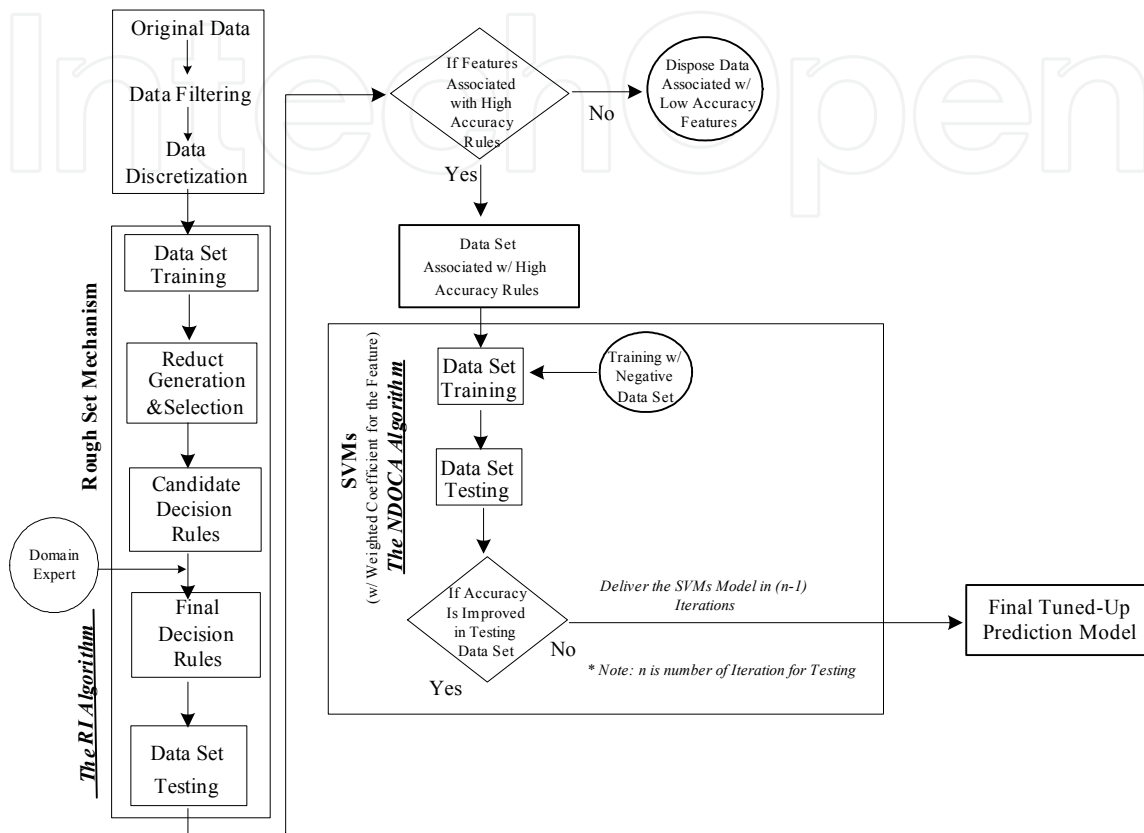


Figure 2. Conceptual framework of the hybrid data mining approach to prediction problem

| Object No. | F1 | F2 | F3 | F4 | F5 | F6 | F7 | O | Weight f_i |
|--------------|-----|------|-----|-----|-----|-----|-----|---|--------------|
| 1 | 3 | 1 | 0 | 1 | 2 | 0 | 2 | 2 | 100% |
| 2 | 3 | 0 | 1 | 2 | 1 | 0 | 3 | 0 | 100% |
| 3 | 0 | 1 | 2 | 2 | 1 | 0 | 1 | 2 | 62% |
| 4 | 0 | 1 | 1 | 1 | 2 | 0 | 1 | 1 | 38% |
| 5 | 1 | 2 | 2 | 0 | 2 | 1 | 0 | 1 | 92% |
| 6 | 2 | 2 | 0 | 0 | 2 | 1 | 1 | 1 | 54% |
| 7 | 1 | 0 | 0 | 1 | 3 | 0 | 1 | 2 | 54% |
| 8 | 3 | 2 | 1 | 1 | 2 | 1 | 1 | 1 | 69% |
| Weight w_j | 80% | 100% | 90% | 60% | 70% | 60% | 90% | | |

Table 3. Data set with un-equal weight for object and feature

The RI Algorithm

Step 0.

- (i) List the auxiliary matrix.
- (ii) Compare the reducts (rows of matrix $[a_{ij}]$). Select the features used in only single feature reducts of the object(s).
- (iii) List the number of known value for each column in $[a_{ij}]$. Select the potential features used, base on the higher number of known value (refer the results from (ii)).
- (iv) Set iteration number $k = 1$.

Step 1. Compare those reducts (rows of matrix $[a_{ij}]^{(k)}$) for one specific case at a time. Select the reducts from the potential features used and based on the auxiliary matrix. If more than one solution for the reduct selection, then select the reduct which can be merged by most of objects; otherwise, select the reducts which are most frequently selected from previous iterations. Draw a horizontal line h_i through each row of matrix $[a_{ij}]^{(k)}$ corresponding to these reducts.

Step 2. For each column in $[a_{ij}]^{(k)}$ corresponding to an entry of feature, which is not "x", single crossed by any of the horizontal lines h_i , draw a vertical line v_j .

Step 3. Repeat steps 1 and 2 until one reduct has been selected for each object in the current outcome. All double-crossed entries of features of the matrix form the rules.

Step 4. If all objects have been concerned in the current outcome, transform the incidence matrix $[a_{ij}]^{(k)}$ into $[a_{ij}]^{(k+1)}$ by removing all the rows and corresponding to an entry of feature, which is not "x", included in the current outcome.

Step 5. If matrix $[a_{ij}]^{(k+1)} = " "$ (where " " denotes a matrix with all elements equal to blank, stop and output the results; otherwise set $k = k + 1$ and go to step 1.

Note that the difference between the equal and un-equal cases for the use of the RI algorithm is "Step 0 (i) is not required by equal weight case." Consider the data set in Table 3. Determine the desired reducts (rules) in Table 4 using the RI algorithm. Repeating Steps 1-5, the final results are shown in Table 4, indicating four features 2, 3, 5, and 7 have been selected. The proposed RS based approach aims to incorporate a weight factor into each feature, process qualitative data, generate decision rules, and identify significant features. This entails that the feature (dimension) domain can be reduced tremendously.

Note that the key contribution of weight in the reduct induction is that the assigned weights help determine the preferred reducts whenever the alternative reducts are produced.

| Object No. | F1 | F2 | F3 | F4 | F5 | F6 | F7 | O |
|------------|----|----------|----------|----|----------|----|----------|---|
| 1 | x | x | x | x | x | x | 2 | 2 |
| 4 | x | 1 | 1 | x | x | x | x | 1 |
| 5, 6 and 8 | x | 2 | x | x | x | x | x | 1 |
| 2 | x | x | x | x | x | x | 3 | 0 |
| 7 | x | x | x | x | 3 | x | x | 2 |
| 3 | x | 1 | 2 | x | x | x | x | 2 |

Table 4. The desired reducts for Table 3

At this point, it is discerned that the weight assignment approach supports to generate the preference-based rule. Furthermore, the preferred decision rules (normally with a high accuracy) derived from the RST based approach (an individual based data mining approach) are not capable of predicting upcoming testing data sets, except when the condition part from test sets matches the preferred decision rules. Therefore, a population based data mining approach (e.g., SVMs based approach) with the consideration of negative data sub-set is introduced next.

3.2. Learning Problem Description through SVMs

The training data set is partitioned into three disjointed subsets: misclassified, not well-separated, and well-separated examples. The misclassified and not well-separated examples together are in the negative data subset whereas the well-separated examples are called in the positive data subset. For example, in the surface roughness prediction, misclassified, non-conformation part is an example of the negative data sub-set. To illustrate the structure of the data set, there is an *instance* vector \mathbf{x} from an input space \mathbf{X} , a response or *label* \mathbf{y} from an output space \mathbf{Y} and a hypothesis h that forms a hypotheses space H for a learner L . For example, \mathbf{X} represents all input features (F1 - F7) in Table 2, while \mathbf{Y} represents one output feature (O). Assume we have

$$\mathbf{x} = (x^{(1)}, \dots, x^{(n)})', \mathbf{X} \in \mathbb{R}^n, \mathbf{x} \in \mathbf{X}, x^{(i)} \in \mathbb{R} \quad (3)$$

where $R =$ a set of real numbers, integer $n > 0 =$ the size of vector x , for *multi-category classification*, $Y = \{1, 2, \dots, m\}$. A *training set* or *training data* S is a collection of *training examples* or *observations* given by $z_i = (x_i, y_i)$. It is denoted by

$$S = (z_1, \dots, z_\ell) = ((x_1, y_1), (x_2, y_2), \dots, (x_\ell, y_\ell)) \subseteq Z^\ell. z_i \in Z = (X, Y), i = 1..l \quad (4)$$

where $\ell = |S|$ is the size of the training set. There exists a true functional relationship or underlying function $f: X \rightsquigarrow R^n \rightsquigarrow Y$, which is often based on the knowledge of the essential mechanism. These types of model are called *mechanistic models*. A hypothesis h is an approximation to the underlying functional relationship f between variables of interest. The problem for the learner L is to learn an unknown target function $h: X \rightsquigarrow Y$ drawn from H and output a maximum likelihood hypothesis.

3.3 Negative Data Oriented Compensation Algorithm

It is not likely to select a perfect model for a practical problem without approximation errors in a learning algorithm. To select a perfect model, imagining that underlying function $f(x)$ is a fluctuant terrain, it is hard to fit the terrain by using a huge size of carpet $h(x)$. The reason is that only the training set and limited prior knowledge is available. The main idea of reducing the approximation error is to compensate the parts of an oversized carpet by a sequence of small sized carpets $h^{(i)}(x)$ which is driven by the negative data subset of training data. The procedure of the Negative Data Oriented Compensation Algorithm (NDOCA) has three parameters, S_0 is the training data set; T_0 is the testing data set; and δ is a degree of vector similarity. For example, δ is difference between two suppliers (objects) in the preferred supplier selection. The return value of the algorithm is the predictive labels of the testing data set. Six subroutines are invoked,

1. $h^{(i)}(x) = \text{LEARN}(S_i)$
2. $P_i = \text{PREDICT}(T_i, h^{(i)}(x))$
3. $S_{i+1} \cup \mathcal{S}_{i+1} = \text{DIVIDER}(S_i, h^{(i)}(x))$
4. $T_i = \text{VS}(S_i, T_{i-1}, \delta)$
5. $P_{i+1} = \text{OV}(P_{i-1}, P_i)$
6. $\text{TC}(k, S)$

LEARN is for training to get the model or hypothesis; PREDICT is to predict the labels of given data set and model. These two procedures are from classical learning algorithms such as SVMs and artificial neural networks. DIVIDER is to divide training data set into positive and negative data subsets by given the hypothesis and the function partitioner $d(h,x,y)$. DIVIDER will call PREDICT routine. In each pass, the function VS and DIVIDER could be different. The following is an algorithm described as pseudo-code (Figure 3).

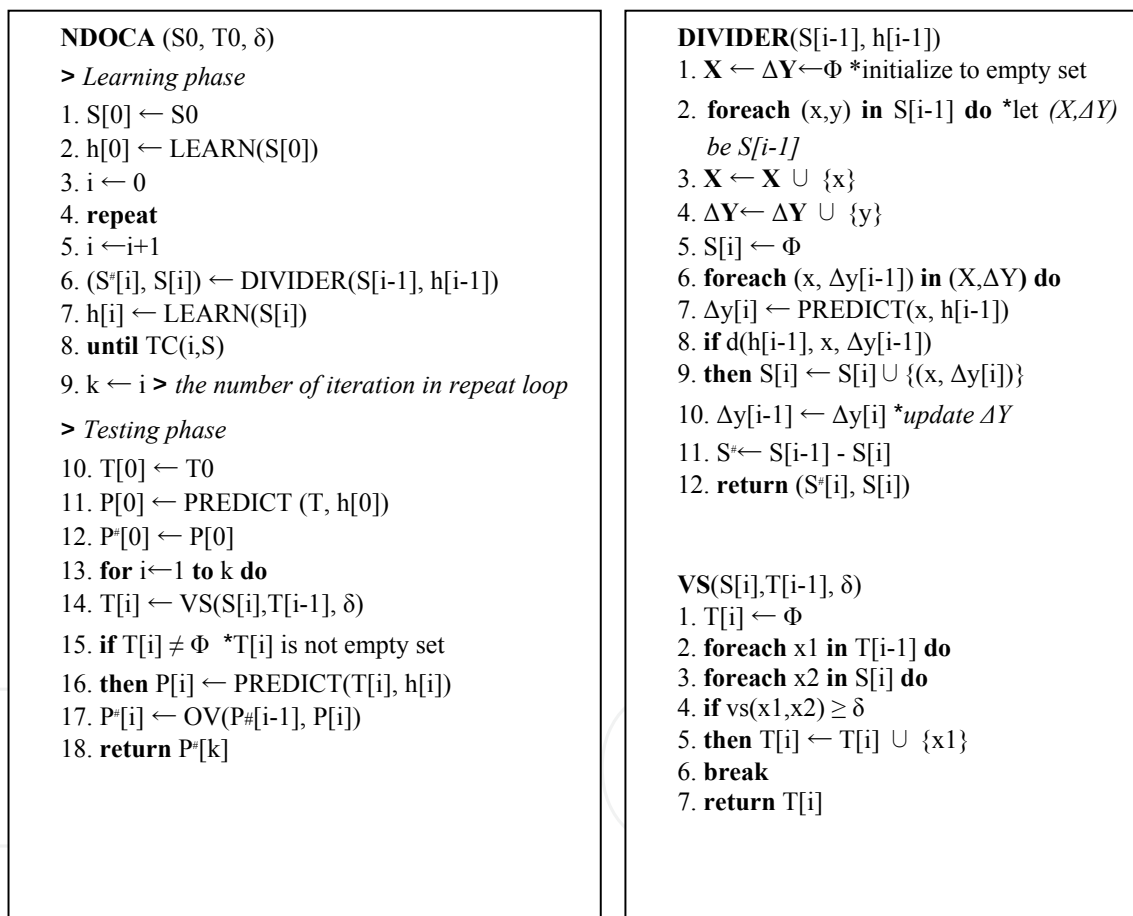


Figure 3. Pseudo-code of the NDOCA

To prepare for the NDOCA learning algorithm, partitioner function $d(h,x,y)$, terminate criteria function $TC(k,S)$, and vector similarity $vs(x1,x2)$ need to be provided. The performance of NDOCA very depends on the selecting of partitioner and vector-similarity function, which needs priori knowledge of learning problems. Note that the NDOCA algorithm is taken as weighted data based on weight coefficients, given by the domain experts.

4. An Empirical Study

4.1 Problem Structure and Data Set Description

Over the years, A-Metal Inc. (a pseudonym for the company) has collected over 1,000 records (objects) of machining data and wishes to investigate the machining features which have a significant impact on the quality of surface finish. Figure 4 illustrates the intelligent CNC control scheme that A-Metal is planning to implement, as opposed to the conventional CNC control that has no response capability as machining process changes.

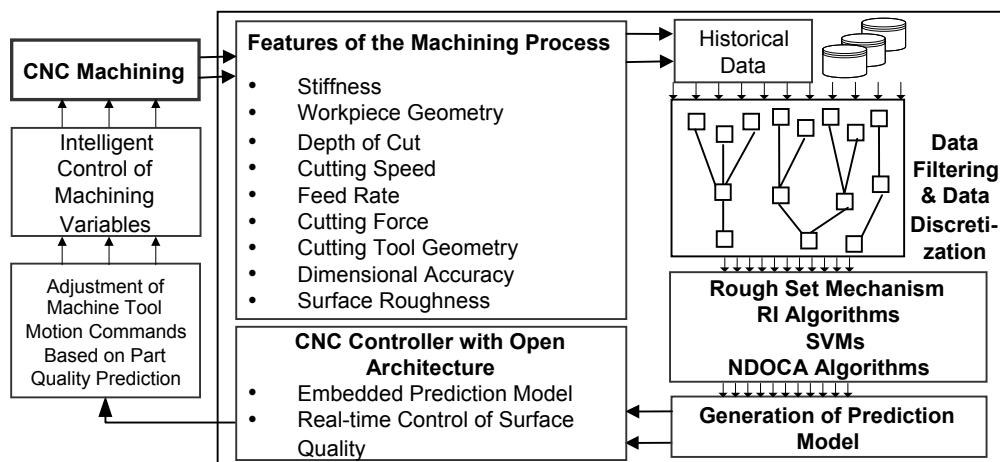


Figure 4. Structure of the closed loop machining operation process

In order to derive the rules and algorithm, conditions of the variables, which could meet the required surface roughness, were identified. Those specific variables will be used to develop the intelligent control system, and in addition can be used by industry to optimize the surface roughness of machined metal (e.g., aluminum, steel) components. Each information object was described with the eight features, F1 through F8, and one outcome, O (Table 5). The work-piece materials include three different types, including 6061-T6 aluminum, 7075-T6 aluminum, and 4140 medium carbon steel (Figure 5).

The surface roughness of the machined bores was measured along a machine Z-axis (parallel to the height of the bore). The machining has been performed on the Cincinnati Hawk CNC Turning Center. The effects of cutting speed, depth of cut, machine set up-modal stiffness, feed rate, cutting tool, tool nose

radius and resultant cutting force on the performance of surface roughness estimation were studied. After roughing and semi-finishing operations, the surface roughness was measured by means of a Taylor Hobson® surface profilometer.



Figure 5. A snapshot of CNC machining (a) and a mixed array of parts consisted of 6061 Al (top), 7075 Al (middle), and 4140 medium carbon steel (the bottom two rows) (b).

| | Factor | Weight |
|----------------|--------------------------------|--------|
| F1 | Types of work piece material | .9 |
| F2 | Cutting speed | .8 |
| F3 | Depth of cut | .8 |
| F4 | Machine set up-modal stiffness | .8 |
| F5 | Feed rate | .7 |
| F6 | Cutting tool | .9 |
| F7 | Tool nose radius | .85 |
| F8 | Resultant cutting force | .75 |
| Outcome | Surface roughness (Ra) | |

Table 5. Feature set of the machining operation process

The contents of the outcome are recorded in a binary format. “ONE” means surface roughness is acceptable, while “ZERO” means unacceptable. The significant variables, which have impact on the quality of surface roughness, were determined through the rule identification algorithms. The decision pro-

duced by the algorithm became decision rules stored in the process control system.

4.2 Computational Results

To show the superiority of the proposed approach, the computational results from the RST part and the hybrid approach part are illustrated. Section 4.2.1 describes the final decision rules with significant features derived from RST. The summary of accuracy results from the test set is presented to show performance of the proposed RI algorithm. Section 4.2.2 represents solutions through the hybrid approach. Comparison among RST, SVMs, and the hybrid approach is also depicted to demonstrate accuracy of each approach in this section.

4.2.1 Rough Set Theory Part

The “Rough Set Based Decision Support System” software (Figure 6) was developed by the authors and implemented in the Advanced Manufacturing Laboratory at the University of Texas at El Paso. It was installed using an Apache 1.3 web server to enable the remote use. The system was developed with C++ language and the Common Gateway Interface (CGI) is used as a communication protocol between the server and client ends. The historical data were split into two data sets. One is the training data set to derive the decision rules; the other is the testing data set to verify the decision rules. Kusiak (2001) suggested the split of the data set using the bootstrapping method according to the following ratio: 0.632 for the training set and 0.368 for the test set. In this study, training data set was collected for 667 parts and testing data set was collected for 333 parts. 41 out of 667 parts in the training set were unacceptable for surface roughness, while 19 out of 333 parts in the testing set were rejected.

All decision rules derived by the RI algorithm were expressed in the form of IF-THEN rules, as illustrated in Table 6. Number of support (see the 3rd column) was recorded from the training set. The selection criteria were based on the threshold value, indicating the ratio of the number of objects supported by that individual rule to the number of total objects. In this case study, a 15% threshold value is selected based on the quality engineer’s expertise. All selected decision rules should be equal or greater than this selected threshold value. For example, the first rule in Category I shows 102 acceptable parts

based on a surface roughness leading to 16% non-defective population. Category I describes the relationship between the features and the acceptable parts. The third rule in Category I is strongly supported because it represents 20% of the acceptable population. In Category II, 17% and 20% of the unacceptable parts are identified by the two rules. Overall, more simple rules (less features as conditional features) are shown in Table 6. The simple rule is treated as the desirable rule because if only two conditions are matched then the rule is fired. Based on the 15% threshold value, significant features F1, F2, F3, F5, and F8 are identified. One can observe that all rules include Feature 1 (types of work piece materials). Therefore, Feature 1 is significant in this set of rule induction. F2, F3, F5, and F8 are significant as well since they are included in the final decision rules. It can be seen that the type of work piece materials, cutting speed, depth of cut, feed rate, and resultant cutting force are important factors for the quality characteristic.

The screenshot displays the 'Rough Set Based Decision Support System' interface. The main window shows a table of objects and their attributes, with the title 'Reducts Merged'. The table has columns for 'Pick', 'Objects', 'A1', 'A2', 'A3', 'A4', 'A5', 'A6', 'A7', 'R1', 'Freq', 'Level', 'SW', and 'SI'. Below the table, there is a 'Weights' section with input fields for A1 through A7 and an 'Update Weights' button. At the bottom, there is a 'Decision Rules' section with an 'Update' button and a 'Verify' button.

| Pick | Objects | A1 | A2 | A3 | A4 | A5 | A6 | A7 | R1 | Freq | Level | SW | SI |
|--------------------------|-------------|----|----|----|----|----|----|----|----|------|-------|----|----|
| <input type="checkbox"/> | 1,2,4,11,14 | X | X | X | 1 | X | X | X | 0 | 5 | 1 | 1 | 5 |
| <input type="checkbox"/> | 1 | X | X | X | X | 1 | X | X | 0 | 1 | 1 | 1 | 1 |
| <input type="checkbox"/> | 2 | X | X | X | X | 0 | X | X | 0 | 1 | 1 | 1 | 1 |
| <input type="checkbox"/> | 3,4 | X | 4 | X | X | X | X | X | 0 | 2 | 1 | 1 | 2 |
| <input type="checkbox"/> | 3 | X | X | X | X | 7 | X | X | 0 | 1 | 1 | 1 | 1 |
| <input type="checkbox"/> | 4 | X | X | X | X | 0 | X | X | 0 | 1 | 1 | 1 | 1 |
| <input type="checkbox"/> | 4 | X | X | X | X | X | 4 | X | 0 | 1 | 1 | 1 | 1 |
| <input type="checkbox"/> | 5,7,8,10 | X | X | X | X | 2 | X | X | 1 | 4 | 1 | 1 | 4 |
| <input type="checkbox"/> | 6 | X | X | X | X | 6 | X | X | 1 | 1 | 1 | 1 | 1 |
| <input type="checkbox"/> | 9 | X | X | X | X | 1 | X | X | 0 | 1 | 1 | 1 | 1 |
| <input type="checkbox"/> | 11 | X | X | X | X | 4 | X | X | 0 | 1 | 1 | 1 | 1 |
| <input type="checkbox"/> | 11 | X | 0 | X | X | X | X | X | 0 | 1 | 1 | 1 | 1 |
| <input type="checkbox"/> | 12 | X | 2 | X | X | X | X | X | 1 | 1 | 1 | 1 | 1 |
| <input type="checkbox"/> | 12 | X | X | X | X | 2 | X | X | 1 | 1 | 1 | 1 | 1 |
| <input type="checkbox"/> | 13 | X | X | X | X | 6 | X | X | 1 | 1 | 1 | 1 | 1 |
| <input type="checkbox"/> | 14 | X | X | X | X | 4 | X | X | 0 | 1 | 1 | 1 | 1 |
| <input type="checkbox"/> | 1,4,14 | 0 | X | X | 1 | X | X | X | 0 | 3 | 2 | 2 | 6 |
| <input type="checkbox"/> | 1 | 0 | X | X | X | 1 | X | X | 0 | 1 | 2 | 2 | 2 |
| <input type="checkbox"/> | 1 | X | 2 | X | X | X | X | X | 0 | 1 | 2 | 2 | 2 |

Figure 6. Screen shot of rough set application software

| Rule No. | Rule expression | No. of support | % of the part population by the rule (from training set) |
|----------|--|----------------|--|
| 1 | IF (F1 = Al 6061) AND (F3 = .2) THEN (D = 1). | 102 | 16 |
| 2 | IF (F1 = Al 6061) AND (F5 = .017) THEN (D = 1). | 91 | 15 |
| 3 | IF (F1 = Al 7075) AND (F8 = 600) THEN (D = 1). | 125 | 20 |
| 4 | IF (F1 = Al 7075) AND (F5 = .005) THEN (D = 1). | 75 | 12 |
| 5 | IF (F1 = Steel 4140) AND (F2 = 1200) AND (F8 = 300) THEN (D = 0). | 7 | 17 |
| 6 | IF (F1 = Al 6061) AND (F8 = 600) THEN (D = 0). | 8 | 20 |

Table 6. Examples of decision rules. Note: (1) F3: depth of cut, F5: feed rate, F8: resultant cutting force, F2: cutting speed, (2) Category I includes Rule 1– 4 and Category II includes Rule 5–6.

Testing on the validity of the rules, which extracted from a data set, was carried out by the rule-validation procedure, which includes a comparison between each decision rule and each new object from the test set. One set of 314 parts with 19 defectives is used as the test set. The accuracy of results for 314 test set parts is shown in Table 7. As Pawlak (1991) explains, the “classification quality” of a feature set is the percentage of all objects in the training data set that can be unambiguously associated with the decision values based on the features in this set. At the same time, the “Diagnostic Accuracy” or so called “Classification Accuracy” for a rule set is the number of correctly classified objects from the test set to all objects in the test set. These results are animate since all of selected rules with a 15% threshold value denote close to 90% accuracy except the third rule in the first category. Four out of six rules (the 1st and 2nd rules in category I, the 1st and 2nd rules in category II) are shown over 90% accuracy. However, the good quality of rule depends on its diagnostic accuracy (Kusiak, 2001). In Table 7, the significant features are identified as F1, F2, F3, F5 and F8. Since the significant features in this case study are fathom, the dimension of interest can be reduced from 8 features to 5 features.

| Test Set | Category | I | | | | II | |
|---|--|---------------|-------------------|-----------|-----------|-----------------------|---------------|
| | | 1 | 2 | 3 | 4 | 1 | 2 |
| 333 Parts (314 acceptable vs. 19 unaccept- able) | Rule ID | | | | | | |
| | Feature Set | F1, F3 | F1, F5 | F1, F8 | F1, F5 | F1, F2, F8 | F1, F8 |
| | % of the part popu- lation by the rule (from training set) | 16% | 15% | 20% | 12% | 17% | 20% |
| | Classification qual- ity | 94.9% | 100% | 70.1 % | 88.9 % | 94.6% | 100% |
| | Diagnostic accuracy | 95.4% | 100% | 71% | 89.3 % | 95.3% | 100% |

Table 7. Summary of accuracy results from the test set. Note: Bold text represents the threshold values of 15% case with acceptable Classification quality and Diagnostic accuracy

4.2.2 Hybrid Approach Part

The NDOCA algorithm is implemented by Perl and uses a modified SVMlight (Joachims, 2002; Joachims, 1999) as a base learning algorithm, including learning and classifying modules. Before the case is studied, the three functions-partitioner function $d(h,x,y)$, terminate criteria function $TC(k,S)$, and vector similarity $vs(x1,x2)$ -need to be defined. To simplify the complexity of computation, the partitioner is defined on the feature space by $d(h, x, y) = \mathbf{iff}(h(x) < \varepsilon, \mathbf{true}, \mathbf{false})$, $\varepsilon \in [0,0.5]$. And $TC(k,S)$ is defined by $TC(i, S[i]) = \mathbf{iff}(|S[i]| \leq |x|, \mathbf{true}, \mathbf{false})$. Basically, the Vector Similarity Euclidean method is used for training and testing data sets. The vector-similarity is a metric to describe the similar degree of two vertices. The vector-similarity plays an extremely important role in the NDOCA learning algorithm. By applying for repairing hypersurface, the first thing is to find which vertices in the testing data set need to be compensated. The vector-similarity is used to find the relationship of vertices in the negative data subset S_i and testing data subset T_{i-1} . Only those vertices in T_{i-1} with high similarity to the ones in S_i need to be compensated.

Since A-Metal Inc. would like to observe the impact of weights and negative data training, the performance measurement includes the following four different cases: 1) equal weight without non-negative data training, 2) un-equal weight without non-negative data training, 3) equal weight with non-negative data training, and 4) un-equal weight with non-negative data training. The n -cross validation is performed in each case. The average result of n -fold is the final accuracy while the minimum and maximum values of accuracy are given

as shown in Table 8. Here, the training data set (with 667 objects) used for rule induction from the previous stage is used for five-fold cross validation. Note that the data set only contains significant features (e.g., F1, F2, F3, F5 and F8). In Table 8, one can observe that the case of equal weight without negative data training contains the lowest diagnostic accuracy with 94.8%. The case of unequal weight with negative data training comprises the highest diagnostic accuracy with 97.3%. The case of un-equal weight without training is not exactly prevailing over the case of equal weight since the accuracy of some individual groups (e.g., group 3) in the equal weight with training case are pretty high (e.g., 97%). Therefore, it is difficult to conclude that the weight effect is predominating over the negative data training effect. In this case study, comparison of accuracy of RST, SVMs, and the hybrid approach is also investigated in order to demonstrate the advantages of applying RS rules and SVMs to prediction. The original 667 objects are applied in this case. The results are shown in Table 9. Note that the accuracy of RST is based on objects that meet the condition of the decision rules. In conclusion, most of the hybrid approaches performed better than the others.

| No. | T ⁺ | T ⁻ | F ⁺ | F ⁻ | C | M | DA% | T ⁺ | T ⁻ | F ⁺ | F ⁻ | C | M | DA% |
|------|----------------|----------------|----------------|----------------|-------|-----|-------|----------------|----------------|----------------|----------------|-------|-----|-------|
| 1 | 120 | 6 | 5 | 2 | 126 | 7 | 94.7% | 122 | 6 | 3 | 3 | 128 | 6 | 95.5% |
| 2 | 121 | 7 | 5 | 1 | 128 | 6 | 95.5% | 122 | 6 | 2 | 3 | 128 | 5 | 96.2% |
| 3 | 120 | 6 | 5 | 2 | 126 | 7 | 94.7% | 120 | 7 | 6 | 0 | 127 | 6 | 95.5% |
| 4 | 119 | 6 | 6 | 3 | 125 | 9 | 93.3% | 119 | 7 | 8 | 0 | 126 | 8 | 94.0% |
| 5 | 121 | 6 | 4 | 2 | 127 | 6 | 95.5% | 121 | 6 | 3 | 3 | 127 | 6 | 95.5% |
| Avg. | 120.2 | 6.2 | 5 | 2 | 126.4 | 7 | 94.8% | 120.8 | 6.4 | 4.4 | 1.8 | 127.2 | 6.2 | 95.4% |
| 1 | 118 | 7 | 9 | 0 | 125 | 9 | 93.3% | 122 | 8 | 3 | 0 | 130 | 3 | 97.7% |
| 2 | 121 | 6 | 4 | 3 | 127 | 7 | 94.8% | 121 | 8 | 4 | 0 | 129 | 4 | 97.0% |
| 3 | 122 | 7 | 2 | 2 | 129 | 4 | 97.0% | 122 | 9 | 2 | 0 | 131 | 2 | 98.5% |
| 4 | 120 | 7 | 5 | 1 | 127 | 6 | 95.5% | 122 | 8 | 4 | 0 | 130 | 4 | 97.0% |
| 5 | 120 | 7 | 5 | 1 | 127 | 6 | 95.5% | 121 | 8 | 5 | 0 | 129 | 5 | 96.3% |
| Avg. | 120.2 | 6.8 | 5 | 1.4 | 127 | 6.4 | 95.2% | 121.6 | 8.2 | 3.6 | 0 | 129.8 | 3.6 | 97.3% |

Table 8. Comparison of four different cases (5-fold cross validation). Note: (1) T⁺ = true positive (good part), T⁻ = true negative (defective), F⁺ = false positive, F⁻ = false negative, C = correct classified and M = misclassified = F⁺ + F⁻ and DA% = diagnostic accuracy = C/(C+M) * 100%; (2) Upper left: equal weight w/o negative data training; upper right: un-equal weight w/o training; lower left: equal weight with training; and lower right: un-equal weight with training.

| Approach | No. of features used/No of data used | Weight included | Negative data training | Correct (%) | Incorrect (%) |
|--------------|--------------------------------------|-----------------|------------------------|-------------|---------------|
| [1] RST | 8/677 | Yes | No | 95.04 | 4.96 |
| [2] SVMs-1 | 8/677 | Yes | Yes | 87.90 | 12.10 |
| [3] SVMs-2 | 8/677 | No | Yes | 89.30 | 10.70 |
| [4] Hybrid-1 | 5/677 | Yes | Yes | 94.80 | 5.20 |
| [5] Hybrid-2 | 5/677 | No | Yes | 95.20 | 4.80 |
| [6] Hybrid-3 | 5/677 | Yes | No | 95.40 | 4.60 |
| [7] Hybrid-4 | 5/677 | No | No | 97.30 | 2.70 |

Table 9. Comparison of proposed hybrid approach with RST and SVMs approaches. Note: (1) The accuracy of RST is based on objects meet the condition of the decision rules; (2) 5-fold cross validation is applied in all cases.

5. Conclusions

Based on the historical data, this study employed a hybrid method that connects with the causal relationships between the features of the machining process and acceptance of surface roughness. This methodology is applied to the case of surface roughness prediction. Several features that significantly impact surface roughness were identified and considered in the case study. Several experiments with the RST, SVMs, and hybrid approach (included equal and unequal weights and with or without negative data training, and different data sets) were also conducted and the results are compared. Two main algorithms are proposed in this study. One is called the RI algorithm, while the other is named the NDOCA algorithm. The RI is used to derive high accuracy decision rules and identify significant features. The NDOCA is used to improve the learning algorithm performance through compensating the base hypothesis by using the negative data set. According to the hybrid approach, combination of RI and NDOCA provides a high accuracy prediction tool for investigating features that contribute to surface roughness. The hybrid approach provides important information for acceptance of surface roughness in the machining operations. The results showed practical viability of this approach for quality control. Future research can focus on the derived rules constitute the basis for developing a rule-based intelligent control system for surface roughness in the machining operation process.

6. References

- Berry, M. & Linoff, G. (1997). *Data Mining Techniques: For Marketing, Sales, and Customer Support*, John Wiley & Sons, New York, NY, USA
- Bredensteiner, E.J. & K.P. Bennett (1999). Multicategory classification by support vector machines. *Computational Optimizations and Applications*, Vol. 12, pp. 53-79
- Chen, C.M.; Lee, H.M. & Kao, M.T., 2004 (b). (2004). Multi-class SVMs with negative data selection for Web page classification. *Proceedings of 2004 IEEE International Joint Conference on Neural Networks*, Vol. 3, pp. 2047-2052
- Cheung, D.W.; Ng, V.T.; Fu, A.W. & Fu, Y. (1996). Efficient Mining of Association Rules in Distributed Databases. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 8, No. 6, pp. 911-922
- Dhar, V. & Stein, R. (1997). *Seven Methods for Transforming Corporate Data into Business Intelligence*, Upper Saddle River, Prentice-Hall, New Jersey, USA
- Fayyad, U.M. & Uthurusamy, R. (2002). Evolving data mining into solutions for insights. *Communications of the ACM*, Vol. 45, No. 8, pp.28-31
- Hsu, C.-W. & Lin, C.J. (2002). A comparison of methods for multi-class support vector machines. *IEEE Transactions on Neural Networks*, Vol. 13, No. 2, pp.415-425
- Joachims, T. (1999). *Making large-Scale SVM Learning Practical*, MIT Press, MA, USA
- Joachims, T. (2002). *Learning to Classify Text Using Support Vector Machines*, Kluwer Academic Publishers, Dordrecht, The Netherlands
- Kusiak A., 2001(a). Feature transformation methods in data mining. *IEEE Transaction on Electronics Packaging Manufacturing*, Vol. 24, No. 3, pp.214-221
- Kusiak, A., 2001(b). Rough Set Theory: A Data Mining Tool for Semiconductor Manufacturing. *IEEE Transactions on Electronics Packaging Manufacturing*, Vol. 24, No. 1, pp. 44-50
- Kusiak, A. & Kurasek, C. (2001) Data mining of Printed Circuit Board Defects. *IEEE Transactions on Robotics and Automation*, Vol. 17, No. 2, pp. 191-196
- Kurosu, M. & Ookawa, Y. (2002) Effects of negative information on acquiring procedural knowledge. *Proceedings of International Conference on Computers in Education*, 3-6 December, Vol.2, pp.1371-1372
- Pawlak, Z. (1991) *Rough Sets: Theoretical Aspects of Reasoning About Data*, Boston: Kluwer Academic Publishers, MA, USA
- Platt, J.; N. Cristianini & J. Shawe-Taylor. (2000) Large margin DAGs for multiclass classification. *Advances in Neural Information Processing Systems*, Vol. 12, pp.547-553, MIT Press, MA, USA
- Vapnik, V. N. (1998). *Statistical Learning Theory*, John Wiley & Sons, New York, NY, USA
- Ziarko, W.P. (1994) *Rough Sets, Fuzzy Sets and Knowledge Discovery*, Springer-Verlag, New York, NY, USA



Manufacturing the Future

Edited by Vedran Kordic, Aleksandar Lazinica and Munir Merdan

ISBN 3-86611-198-3

Hard cover, 908 pages

Publisher Pro Literatur Verlag, Germany / ARS, Austria

Published online 01, July, 2006

Published in print edition July, 2006

The primary goal of this book is to cover the state-of-the-art development and future directions in modern manufacturing systems. This interdisciplinary and comprehensive volume, consisting of 30 chapters, covers a survey of trends in distributed manufacturing, modern manufacturing equipment, product design process, rapid prototyping, quality assurance, from technological and organisational point of view and aspects of supply chain management.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Tzu-Liang (Bill) Tseng, Yongjin Kwon and Ryan B. Wicker (2006). Applying a Hybrid Data Mining Approach in Machining Operation for Surface Quality Assurance, Manufacturing the Future, Vedran Kordic, Aleksandar Lazinica and Munir Merdan (Ed.), ISBN: 3-86611-198-3, InTech, Available from:
http://www.intechopen.com/books/manufacturing_the_future/applying_a_hybrid_data_mining_approach_in_machining_operation_for_surface_quality_assurance

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2006 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen