Beyond Condition-Monitoring: Comparing Diagnostic Events with Word Sequence Kernel for Train Delay Prediction

Wan-Jui Lee[1], David M.J. Tax[2], and Robert P.W. Duin[3]

[1] *Maintenance Development, Fleet Services, NedTrain, Utrecht, 3500GD, the Netherlands*
Wan-Jui.Lee@ns.nl

[1,2,3] *Pattern Recognition and Bioinformatics Group, Delft University of Technology, Delft, 2628CD, the Netherlands*
W.J.Lee@tudelft.nl
D.M.J.Tax@tudelft.nl
rduin@xs4all.nl

ABSTRACT

In the modern trains operated by the Dutch Railways (Nederlandse Spoorwegen) in the Netherlands, there are on-board train management systems continuously monitoring the conditions of various train modules such as traction, climate, brake electronics and so forth. When an abnormal or particular situation occurs, the system will generate and store an event log on the local disk or on a remote disk using wireless data communications. These diagnostic events might give an indication of the train condition, and currently critical events are selected by business rules to give alarms on failure or malfunction to the control room. To give a better prediction on the trains status based on the condition monitoring data, sequences of diagnostic events instead of individual critical events are analyzed in this work. Moreover, train delays instead of train failures are used as targets for providing more insight on the degeneration behavior of trains. We have adopted the word sequence kernel for learning the similarity between all sequence pairs, where each diagnostic event is considered as a word. To include multi-length word interpretations, we propose to combine the word sequence kernels of various lengths, where length=1 means one word is matched, length=2 means two words are matched, and so on. A kernel machine or similarity-based model can be learned directly on this combined word sequence kernel. The experimental results demonstrate that combining word sequence kernels of different lengths can bring a richer description to similarity measurements and gives better prediction performance.

1. INTRODUCTION

NedTrain is the part of the Dutch Railways that is responsible for the cleaning, maintenance and service, and overhaul of rolling stock. To ensure all trains are reliable and safe to operate at the lowest cost, NedTrain is continuously optimizing the maintenance schedule to plan when and what to maintain (de Vos et al., 2015 & Jiang et al., 2012). Between 2009 and 2012, 131 new SLT (Sprinter Light Train) trains of the Electric Multiple Unit (EMU) train type (in configurations of four and six coaches) were introduced for intensive regional rail services on the Dutch rail network. These modern trains are equipped with mechatronics and digital train management systems, and therefore much information and data on the technical state of the components is available for monitoring the train condition. Condition monitoring is more than detecting train failure or malfunctions. Continuous gathering of data allows for trend analysis over the entire fleet and allows for data-driven performance improvements for instance actual state-dependent maintenance (Poot-Geertman et al, 2015). With the wealth of condition information, static and reactive preventive and corrective maintenance scheduling will be replaced by dynamic and proactive predictive maintenance in the future (Eker et al., 2014).

The condition-based monitoring system sends out diagnostic events of various severity degrees. However, it is in general difficult to define what is the actual health condition of trains based on these events. Also, there are more than 3000 individual events and these diagnostic events can hardly be interpreted as condition measures by even the most experienced technicians and engineers. To establish an automatic system predicting the actual performance and state of individual trains, the fleet services department of NedTrain decided to research on building predictive models of train delays based on the diagnostic events.

Usually there is more than one diagnostic event occurring within one single day on one of the monitored trains, and therefore methods that can handle sequences are required for building a predictive model based on such data. Machine learning methods for sequence data are most commonly developed in the application fields of document categorization and bioinformatics (Lodhi et al., 2002 & Cancedda et al., 2003). In the early works, documents were represented using the standard bag-of-word model to use word frequencies as data vectors. Such data vectors however fail to encode the sequence structure. The string kernel, one of the first significant departures from the vector space model was proposed by Lodhi et al. (2002), thanks to the fast development of kernel machines in early 2000. Instead of using word frequencies, string kernels compute the similarity between two sequences by comparing (in principle) all possible subsequences which can be considered as a dot product in an implicit high dimensional space. Because the number of all subsequences can be very high, for computational reasons only the subsequences that actually appear in the strings are considered. In 2003, Cancedda et al. (2003) proposed word sequence kernels to extend the idea of string kernels to process documents as sequences of words. Matching sequences of words are expected to be more linguistically meaningful compared to matching with sequences of characters. Also, the sequence length for computing sequence matching is greatly reduced by replacing characters with words and therefore the computing efficiency is significantly improved.

However, the string kernel and word sequence kernel work on a pre-defined length of *n* characters or words. It is not difficult to imagine that a lower *n* means comparing sequences in a more general level and a higher *n* means comparing sequences in a more specific level because the length of subsequences is respectively shorter and longer. In many applications, it is difficult to determine the optimal value of *n* and often the sequences need to be compared in different levels. To integrate the comparison of sequences in different levels, we adopt the kernel combination method (Lanckriet, 2004) to combine the word sequence kernels of various lengths of subsequences. A support vector machine (SVM) can directly be learnt on the combined kernel to construct a predictive model of train delays.

2. PREDICTIVE MODEL OF TRAIN DELAYS

In this paper, we concatenate a series of diagnostic events occurring within one day into a sequence and then compute the similarity between all pairs of sequences using the combined word sequence kernel. The sequences of a monitored train that will get a delay within two days are labeled as DELAY. All other sequences are labeled as NO-DELAY. The derived combined kernel matrix is directly used as the input data for support vector machine to build a predictive model of train delays. Support Vector Machine, String Kernel and Combined Word Sequence Kernel will be introduced briefly in the following sections.

2.1. Support Vector Machine

Support vector machine (SVM), motivated by the results of statistical learning theory, is one of the most popular kernel machines (Vapnik, 1995). Most of the kernel combination research is based on it. In an SVM, the decision boundary that is separating classes is represented by a small subset of training examples, called the support vectors. Unlike the traditional methods that minimize the empirical training errors, support vector machines implement the structural risk minimization principle. By adopting this principle, SVM can find the optimal discriminant hyperplane minimizing the risk of an erroneous classification of unseen test samples. In the following, we introduce the support vector classifier for 2-class problem with class label +1 and -1, and x_i and y_i represent i th input datum (a vector) and its corresponding class label. Extension to multi-class problems can be achieved by training multiple support vector machines.

To control both training error and model complexity, the optimization problem for SVM is formalized as follows:

$$\begin{aligned} &\text{Minimize } \tfrac{1}{2} < w, w > + C \textstyle\sum_{i=1}^{n} \varepsilon_i, \\ &\text{subject to } < w, x_i > + b \geq +1 - \varepsilon_i, \text{ for } \; y_i = +1, \\ &\qquad\qquad < w, x_i > + b \leq -1 + \varepsilon_i, \text{ for } \; y_i = -1, \\ &\qquad\qquad\qquad \varepsilon_i \geq 0, \forall i. \end{aligned} \tag{1}$$

By using Lagrange multiplier techniques, Eq. (1) could lead to the following dual optimization problem:

$$\begin{aligned} &\text{Maximize } \sum_{i=1}^{n} \alpha_i - \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \, \alpha_j y_i \, y_j < x_i, x_j >, \\ &\text{subject to } \textstyle\sum_{i=1}^{n} \alpha_i y_i = 0 \, , \\ &\qquad\qquad \alpha_i \in [0, C]. \end{aligned} \tag{2}$$

Using Lagrange multipliers, the optimal desired weight vector of the discriminant hyperplane is $w = \sum_{i=1}^{n} \alpha_i y_i x_i$. Therefore, the best discriminant hyperplane can be derived as

$$\begin{aligned} f(x) &= < \textstyle\sum_{i=1}^{n} \alpha_i y_i x_i \, , x > + \mathrm{b} \\ &= \textstyle\sum_{i=1}^{n} \alpha_i y_i < x_i \, , x > + \mathrm{b}, \end{aligned} \tag{3}$$

where b is the bias of the discriminant hyperplane.

When input data cannot be linearly separated in the original space, they can be mapped into a high dimensional feature space $\varphi(\cdot)$, where a linear decision surface separating the training data can be designed. The computation does not need to be performed in the feature space since SVM depends on the direct application of the kernel function over the input data. A kernel function is a function that calculates the inner product between mapped data objects x_i and x_j in the feature space, that is for any mapping $\varphi(\cdot)$, $K(x_i, x_j)=<\varphi(x_i),\varphi(x_j)>$. Therefore, the kernel function is a key component of SVM for solving nonlinear problems, and the performance of SVM classifiers largely depends on the choice of the kernels.

2.2. String Kernel

A string kernel compares pairs of sequences of diagnostic events by the subsequences they contain, and more subsequences they share, more similar they are. In order to deal with non-contiguous subsequences, a decay factor $\lambda \in (0,1)$ was introduced to weigh the presence of a certain event in a sequence.

Let Σ be a finite set of diagnostic events, a string is a finite sequence of events from Σ, including the empty sequence. The feature mapping $\varphi(s)$ for a string s measures the number of occurrences of subsequences in the string s weighting them according to their lengths. Therefore, for each dimension u of $\varphi_u(s)$, which is a subsequence in the string s with length n,

$$\varphi_u(s) = \sum_{i:u=s[i]} \lambda^{l(i)}, \tag{4}$$

where l(i) is the length of $s[i]$ and n is length of subsequence.

Hence, the inner product of the feature vectors for two strings s and t give a sum over all common subsequences weighted according to their frequency of occurrence and lengths as

$$\begin{aligned} K_n(s,t) &= \sum_{u\in\Sigma^n} <\varphi_u(s), \varphi_u(t)> \\ &= \sum_{u\in\Sigma^n} \sum_{i:u=s[i]} \lambda^{l(i)} \sum_{j:u=t[j]} \lambda^{l(j)} \\ &= \sum_{u\in\Sigma^n} \sum_{i:u=s[i]} \sum_{j:u=t[j]} \lambda^{l(i)+l(j)}. \end{aligned} \tag{5}$$

The string kernel in Eq. (5) needs to be further normalized with the following equation

$$\tilde{K}_n(s,t) = \frac{K_n(s,t)}{\sqrt{K_n(s,s)K_n(t,t)}}. \tag{6}$$

For instance, given three diagnostic sequences S_1=[ATB01, ATB02, TRC02], S_2=[ATB01, TRC02] and, S_3=[ATB02, ATB01, TRC02], where ATB01, ATB02, TRC02 are diagnostic events describing abnormal behaviors of automatic braking systems and traction systems. If we consider only subsequences of length two, we can obtain a 4 dimensional feature space, where the sequences are mapped as given in Table 1:

Table 1. Feature representation of sequences using mapping function $\boldsymbol{\varphi}(\cdot)$.

	ATB01-ATB02	ATB01-TRC02	ATB02-TRC02	ATB02-ATB01
$\boldsymbol{\varphi(S_1)}$	$\boldsymbol{\lambda^2}$	$\boldsymbol{\lambda^3}$	$\boldsymbol{\lambda^2}$	0
$\boldsymbol{\varphi(S_2)}$	0	$\boldsymbol{\lambda^2}$	0	0
$\boldsymbol{\varphi(S_3)}$	0	$\boldsymbol{\lambda^2}$	$\boldsymbol{\lambda^3}$	$\boldsymbol{\lambda^2}$

Hence, the unnormalised kernel between S_1 and S_2 is $\boldsymbol{K_2(S_1,S_2) = \lambda^5}$ and the unnormalised kernel between S_1 and S_3 is $\boldsymbol{K_2(S_1,S_3) = 2\lambda^5}$. Hence the normalized kernel between S_1 and S_3 is $\boldsymbol{\tilde{K}_2(S_1,S_3) = 2\lambda^5/(2\lambda^4+\lambda^6)}$, given $\boldsymbol{K_2(S_1,S_1) = K_2(S_3,S_3) = 2\lambda^4+\lambda^6}$.

In the application of train condition monitoring, some modules might be more critical in safety than the other modules, for instance the automatic braking system and the traction system which are more related to safety will have a higher impact on the train condition compared to the passenger information system. Typically, the diagnosis events are also divided into different categories of severity. Therefore, in this kind of application, the decay factor λ can be assigned with various values for different modules or different severity categories. However, for the ease of simplicity in this study the decay factor λ of all events are assigned with the same value.

2.3. Combined Word Sequence Kernel

Kernel combination (Lanckriet, 2004) is meant to improve the performance of single kernels and avoid the difficulty of kernel selection. Most kernel combination methods average the kernel matrices in one way or another. Suppose p original kernels are given as K_1, K_2, ..., and K_p and the empirical feature functions of these kernels are $\varphi_1(s)$, $\varphi_2(s)$..., and $\varphi_p(s)$, combining kernels by summing up all the kernels is equivalent to taking the Cartesian product of their respective empirical feature spaces as shown in Eq. (7)

$$\sum_{i=1}^{p} K_i(s,t) = \sum_{i=1}^{p} <\varphi_i(s), \varphi_i(t)>. \tag{7}$$

The summed kernel needs to be further normalized using Eq. (6).

3. EXPERIMENTAL RESULTS

From October 2013 to March 2015, a total of 27,365 data sequences from 131 SLT trains are collected, in which 5,133 sequences are labeled as DELAY due to their close proximity to train delays in time and the other 22,232 are labeled as NO-DELAY. In the experiments, 3 lengths, i.e., 1, 2 and 3, of subsequences are considered and combined. The computation of word sequence kernels grows exponentially with the increase in the length of subsequences, and therefore length ≥ 4 is not discussed in this work. Please note that when the length=1, the empirical feature space of word sequence kernel is the same as the bag-of-word representation. Also, all sequence data were collected during daily operations and therefore there were various human influence and interactions to the data. For instance, all trains are inspected daily in either a service or maintenance depot and therefore sometimes trains having critical diagnostic events might already be repaired before a delay occurs. Moreover, besides technical issues, train delays can also be caused by infrastructure, logistic or other external causes, and the registration on the cause of a delay is not always available. Due to the complexity and noisiness of this real-world application, none of the previous studies had delivered satisfactory results.

In the following experimental results, 80% of the SLT sequence dataset was randomly chosen as the training data, and the rest 20% was used as the testing data. The process is repeated 50 times, and the experimental results presented are averaged over 50 runs.

3.1. Performance of Individual and Combined Kernels

Figure 1 shows the ROC (Receiver Operating Characteristic) curves of SVM models built on different kernels. K1 is word sequence kernel with length=1, K2 is word sequence kernel with length=2, K3 is word sequence kernel with length=3, K1+K2 is the normalized combination of K1 and K2 as given in Eq. (6) and Eq. (7), K2+K3 is the normalized combination of K2 and K3, and K1+K2+K3 is the normalized combination of K1, K2 and K3. The decay factor λ is assigned to 0.2 for computing all string kernels. Type I error, aka False Positive, is a negative test object misclassified as positive, whereas Type II error, aka False Negative, is a positive test object misclassified as negative. From the ROC curves, it is clear that K1 gives significantly the worst performance at all times and this indicates the utilization of subsequences is necessary for building a predictive model in this application. Combined word sequence kernels K1+K2 and K1+K2+K3 often outperforms individual kernels K1, K2 and K3. However, the difference between the two combined kernels are not significant. One of the explanations could be that K2 often performs better than K3 and therefore adding K3 to K1+K2 might not give much advantage. Moreover, the performance of the combined kernel K2+K3 is very similar to the individual kernel K2 which suggests that K1 actually carries essential information and is necessary to be included in the combination.

Please notice that an SVM built on K1 can be considered as an SVM using linear kernel in which the input vectors are bag-of-word representation.

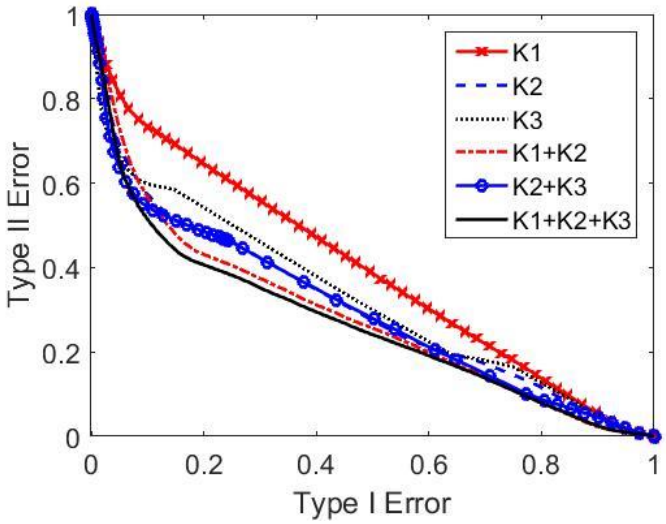


Figure 1. ROC curves of SVM models built on different kernels.

When the Type 1 Error (False Positive) is 6%, 45% of the delays can be correctly predicted by a SVM built on the combined kernel K1+K1+K3. In general, trains that are predicted to encounter a delay within 2 days can be fully inspected in the service or maintenance depot to fix any technical issue to avoid possible train failure or train delays. However, the aim of such a prediction system is to help the control center and maintenance technicians better detect possible defects and then prepare for a solution. In our prediction system, when a sequence is predicted as DELAY, the diagnostic events of the sequence will be ranked by their severity degree and then the most severe ones will be shown together with short descriptions to allow the engineers and technicians further understand the problem and decide whether there is a need for action.

3.2. Lower Bounds on Bayes Error of Overlapping Sequences

The diagnostic sequences collected in this study were events happening during daily operations and there was external interference such as maintenance activities and driving behavior of train drivers. The delay records used as labels are manually registered and it is not always clear whether a delay is caused by train defects or operational issues. All these factors result in a high percentage of overlapping sequences which means identical sequences have different labels. For instance two identical sequences S_1=[ATB01, ATB02,

TRC02] and S_2=[ATB01, ATB02, TRC02] might be labeled as DELAY and NO-DELAY separately.

To evaluate the nature of our problem, we have computed the Bayes error of our dataset for further investigation. The Bayes error is the irreducible probability of misclassification caused by the inherent overlap between the two classes. The computation of Bayes error consists of 2 steps. The first step is to find out all groups of overlapping sequences, and each group is the collection of the same sequence occurring on different days and/or on different trains. In the second step, the inherent error of each group is computed. For each group, the maximum risk happens at when all NO-DELAY sequences are classified as DELAY and vice versa. This maximum risk is used as the inherent error of a group and in this case the Type I Error and the Type II Error, respectively. The Bayes error is therefore the sum of all errors generated by all groups.

Figure 2 shows the ROC curve of the K1+K2+K3 kernel and a lower bound of the Bayes error based on overlapping sequences. The Bayes error suggests the problem itself is highly overlapped and difficult to solve, and the K1+K2+K3 kernel performs reasonably well considering the nature of the problem.

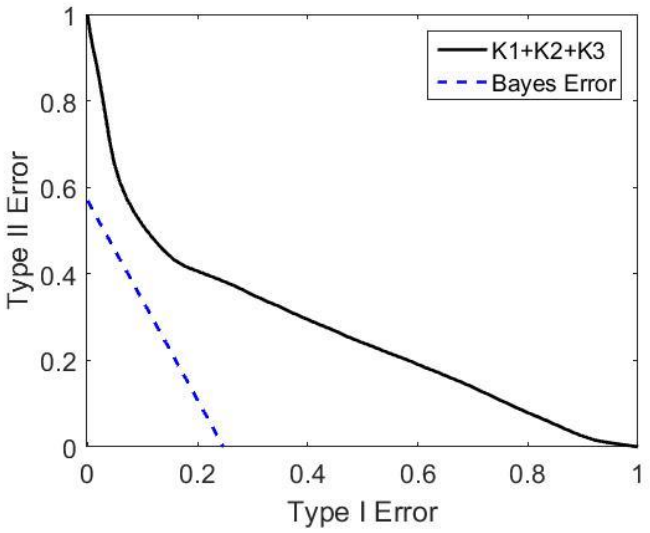


Figure 2. ROC curves of the SVM model built on K1+K2+K3 kernel and the Bayes error of overlapping sequences.

4. CONCLUSIONS

In this paper, we have studied the application of (combined) word sequence kernel on prediction of train delays based on sequences of diagnostic events. The experimental results have demonstrated that the combination of different lengths of subsequence kernels can enrich the SVM model but the contribution of individual diagnostic events are relatively significant. The Bayes error on the overlapping sequences gives a lower bound on our classification task, and gives a comparative reference to the performance of combined word sequence kernel.

ACKNOWLEDGEMENT

The authors would like to thank Mattijs Suurland and Pauline Poot-Geertman from NedTrain for their contributions in data preparation and process understanding.

REFERENCES

de Vos, J. I. A., & van Dongen, L. A. M. (2015). Performance Centered Maintenance as a Core Policy in Strategic Maintenance Control. *Procedia CIRP*, vol. 38, pp. 255–258, 2015.

Eker, O. F., Camci, F., & Jennions, I. K. (2014). A Similarity-based Prognostics Approach for Remaining Useful Life Prediction. *Proceedings of the Second European Conference of the Prognostics and Health Management Society 2014*.

Jiang, J., Huisman, B., & Dignum, V. (2012). Agent-based multi-organizational interaction design: A case study of the dutch railway system. *IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, vol. 2, pp. 196–203, 2012.

Poot-Geertman, P., Huisman, B., & van Rijn, C. F. H. (2015). Application of a Maintenance Engineering Decisin Method for Railway Operation: Managing Fleet Performance, Cost, and Risk. *Safety and Reliability of Complex Engineered Systems: ESREL 2015*, pp. 1863–1870, 2015.

Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., & Watkins, C. (2002). Text classification using string kernels. *Journal of Machine Learning Research*, vol. 2, pp. 419–444.

Cancedda, N., Gaussier, E., Goutte, C., & Renders, J. (2003). Word-Sequence Kernels. *Journal of Machine Learning Research*, pp. 1059-1082.

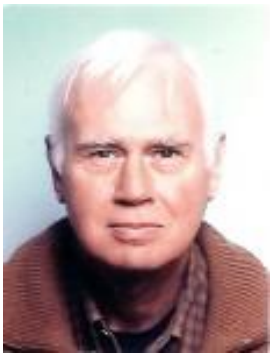
Vapnik, V., (1995). *The Nature of Statistical Learning Theory*. In Louis Redding & Rajkumar Roy (Eds.), New York: Springer.

Lanckriet, G. R. G., Cristianini, N., Bartlett, P., Ghaoui, L. E., & Jordan, M. I. (2004). Learning the Kernel Matrix with Semidefinite Programming. *Journal of Machine Learning Research*, vol. 5, pp. 27-72.

BIOGRAPHIES

Wan-Jui Lee received her Ph.D. degree from the Department of Electrical Engineering, National Sun Yat-Sen University, Taiwan, in 2008. From 2007 to 2011, she was a postdoctoral researcher in the Pattern Recognition Laboratory, Delft University of Technology. Currently she is a researcher of Dutch Railways focusing on predictive maintenance of train fleets. Her main research interests include predictive maintenance, structural pattern recognition, (dis)similarity-based representation and machine learning. In 2016, she won the SmartRail Europe Innovation Awards - Runner up prize.

David M.J. Tax studied Physics at the University of Nijmegen, the Netherlands, in 1996, and received his Master degree with the thesis "Learning of Structure by Many-takeall Neural Networks". After that he received his Ph.D. from the Delft University of Technology, the Netherlands, in the Pattern Recognition group, under the supervision of Dr. Robert P.W. Duin. In 2001 he was promoted with the thesis "One-class Classification". After working for two years as a MarieCurie Fellow in the Intelligent Data Analysis group in Berlin, he is currently an Assistant Professor in the Pattern Recognition Laboratory at the Delft University of Technology. His main research interest is in the learning and development of detection algorithms and (one-class) classifiers that optimize alternative performance criteria like ordering criteria using the Area under the ROC curve or a Precision-Recall graph. Furthermore, the problems concerning the representation of data, multiple instance learning, simple and elegant classifiers and the fair evaluation of methods have focus.



Robert P.W. Duin received in 1978 the Ph.D. degree in applied physics from Delft University of Technology, Delft, The Netherlands, for a thesis on statistical pattern recognition. He is currently an Associate Professor in the Faculty of Electrical Engineering, Mathematics and Computer Science of the same university. During 1980–1990, he studied and developed hardware architectures and software configurations for interactive image analysis. After that he became involved with pattern recognition by neural networks. His current research interests are in the design, evaluation, and application of algorithms that learn from examples, which includes neural network classifiers, support vector machines, classifier combining strategies, and one-class classifiers. Especially complexity issues and the learning behavior of trainable systems receive much interest. From 2000 he started to investigate alternative object representations for classification and he thereby became interested in dissimilarity-based pattern recognition, trainable similarities, and the handling of non-Euclidean data. Dr. Duin is an associated editor of Pattern Recognition Letters and a past-associate editor of the IEEE Transactions on Pattern Analysis and Machine Intelligence. He is a Fellow of the International Association for Pattern Recognition (IAPR). In August 2006 he was the recipient of the Pierre Devijver Award for his contributions to statistical pattern recognition.