Data-intensive analytics for predictive modeling

C. V. Apte
S. J. Hong
R. Natarajan
E. P. D. Pednault
F. A. Tipu
S. M. Weiss

The Data Abstraction Research Group was formed in the early 1990s, to bring focus to the work of the Mathematical Sciences Department in the emerging area of knowledge discovery and data mining (KD & DM). Most activities in this group have been performed in the technical area of predictive modeling, roughly at the intersection of machine learning, statistical modeling, and database technology. There has been a major emphasis on using business and industrial problems to motivate the research agenda. Major accomplishments include advances in methods for feature analysis, rule-based pattern discovery, and probabilistic modeling, and novel solutions for insurance risk management, targeted marketing, and text mining. This paper presents an overview of the group's major technical accomplishments.

Introduction

Predictive modeling, which is perhaps the most-used subfield of data mining, draws from statistics, machine learning, database techniques, pattern recognition, and optimization techniques.

Just what exactly is data mining? At a broad level, it is the process by which one extracts accurate and previously unknown information from large volumes of data. This information should be in a form that can be understood, acted upon, and used for improving decision processes in business and industrial settings. With this definition, data mining is an activity that encompasses a broad set of technologies, including data warehouses, database management, data analysis algorithms, and visualization.

The data analysis (or mining) algorithms can be divided into three major categories on the basis of the nature of their information extraction. These include predictive modeling, clustering (segmentation), and frequent pattern extraction. Data is considered to be a collection of records, with each record being a collection of fields. Using this tabular data model, the data mining algorithms are designed to operate on the contents, under differing assumptions, and to deliver results in differing formats.

Predictive modeling is based upon techniques used for classification and regression modeling. One field in the tabular data set is designated as the target (response or class) variable, and these algorithms produce models for target variables as a function of the other fields in the data set, that are pre-identified as explanatory variables (features). The principal problem addressed by this family of algorithms is to produce predictively accurate functional approximations for the target variables, using potentially noisy data as examples of the relations that exist between instances of explanatory variables and target variables. Once produced, these models can be used to predict the values of target variables, given values for the explanatory variables.

Predictive modeling has its roots in classical statistics. One of the earliest methods developed for classification modeling is the technique of linear discriminants [1]. Similarly, for regression modeling, one of the earliest methods developed is linear regression [2]. Recently, related advances have occurred in other disciplines including pattern recognition, information theory, and machine learning. Because of the increasing availability of massive volumes of data, a preferential shift is taking place toward computational search-based nonparametric modeling techniques, which make no prior assumptions about any underlying distributions in the data. These methods include techniques such as neural networks, decision trees, and decision rules. Recent references that describe these methods in detail can be found [3–5]. Two

0018-8646/03/$5.00 © 2003 IBM

key technical aspects common to all predictive modeling algorithms are the ability to generate models in the presence of noise in the data, and the emphasis on producing accurate error estimates for the models that are generated. A variety of techniques (e.g., cross-validation) have been developed for handling noise and performing error estimation. These techniques provide the foundational basis for most modern predictive modeling methods.

Decision-tree-based and rule-based modeling techniques have found particular favor in the data mining community. These techniques produce models with interpretable structures, making them highly amenable to explanation and human inspection. This characteristic allows business end users and analysts to understand the implications of the models and to then take actions based on these implications. Although other techniques such as those utilizing neural networks may produce predictively accurate models, the model structure does not relate directly to the features of the underlying problem, thereby making them difficult to interpret. However, when evaluating and determining a modeling technique to use from a given set of alternatives, end users have to weigh the compromise among predictive accuracy, level of understandability, and computational demand. Very often, these alternatives must be traded off against one another, because algorithms often compromise one to gain performance in another.

A key factor that influences predictive modeling analytics research in the data mining community and differentiates it from classical statistical modeling is the emphasis on automation and scalability. The goal of most data mining work is to produce easy-to-use software tools for non-expert users. This goal requires increased focus on designing modeling algorithms that can automatically produce robust models from data without any expert tuning or intervention of experienced statisticians. This goal also requires algorithms that can handle very large volumes of data, without the need for experts to extract appropriate samples for modeling. These requirements have led to closer coupling of predictive modeling techniques with database techniques, particularly those that permit data-partitioned parallelism and efficient analysis of high-dimensional data.

DNF rule induction and feature analysis

An early research direction of our group was the development of a classification rule generation algorithm, R-MINI [6]. This algorithm generates complete and consistent classification rules, using a technique based on multivalued disjunction normal form (DNF) minimization. This technique attempts to generate a near-minimal set of rules from training examples by iterating the generalization/specialization heuristic that has been successfully used in a logic minimization system, MINI. Since R-MINI takes as input only nominal (e.g., discrete-valued) features, it requires prediscretization of numerical feature values. Since the rules are consistent, noisy data is handled by developing an ensemble of rule sets from sampled data, and the classification output is obtained by voting among the rule sets. A new concept for the importance of a feature, called contextual merit, was developed [7] that takes into account the presence of other features in determining how a given feature contributes to the discrimination of classes. The technique lends itself to a discretization method so as to maximize the resultant contextual merit of the discretized version of the feature by dynamic programming.

An important step in a classification model building task is to compute the "merit" of a feature, i.e., the value a feature brings to building a good predictive model. Many existing techniques tend to overestimate the merit of categorical features that take on a large number of discrete values, a problem known as variety bias. While many *ad hoc* remedies have been proposed for this problem, none have been very effective. We have explored new ways of accurately estimating the unbiased merit of a feature. We have shown that by comparing the feature merit against the estimated average merit of a random feature with the same distribution as the feature substituted in its place, one can more accurately and efficiently compute the true value of a feature for classification model building [8].

These techniques for feature analysis, feature discretization, and classification rule modeling were implemented in the RAMP (rules abstraction for modeling and prediction) system [9], which was successfully used in an early experimental application for predicting equity returns from securities data [10].

Automated methods for text categorization

An important goal of text mining is to sift through large volumes of text to extract patterns and models that can then be incorporated in intelligent applications, such as automatic text categorizers and routers, for automatically classifying electronic documents. Decision rules and decision-tree-based approaches to learning from text are particularly appealing, since rules and trees provide explanatory insight to end users and text application developers. Our research has focused on applying these methods to automatic text categorization and, more recently, developing methodologies for maximizing their predictive performance. From a predictive modeling perspective, text categorization problems become those of modeling from high-dimensional but sparse training data. Such problems require special handling.

Our initial methodology for automatic text categorization was built around the use of rule induction, coupled with a new approach to constructing feature

vectors that emphasized the use of local dictionaries and numerical features [11, 12]. More recently, we have begun exploring methods for maximizing the predictive accuracy of the models constructed from the text mining process. Maximizing predictive accuracy is an important requirement, particularly in real-world applications in which noisy and limited samples are a pervasive problem.

Benchmark data, such as the Reuters-21578 test collection, has been used by researchers to measure advances in automated text categorization. Conventional methods such as decision trees have been competitive but have not produced the best predictive performance. Using the Reuters collection, we demonstrated that a) adaptive sampling techniques can be used to boost the performance of decision trees, and b) relatively small pooled local dictionaries are effective [13]. Adaptive sampling is an approach to building multiple models (ensembles) from data and applying the ensemble in a weighted combination scheme to produce a new categorization. Results on the Reuters benchmark were superior, surpassing all previously reported results at that point.

We see today a significant increase in research and development activities related to incorporating predictive modeling analytics in text mining, information retrieval, and natural language solutions.

Probabilistic rule estimation modeling

ProbE (for probabilistic estimation) is a customizable data mining engine that is being developed to enhance IBM predictive modeling products and services [14]. ProbE takes as input a collection of records (potentially in the millions) in which each record can be a collection of real, integer, ordinal, or nominal attributes (potentially in the thousands). A specific attribute is pre-identified as a target or dependent variable, and the remaining attributes as explanatory or independent variables. ProbE mines this data to produce a segmentation-based if-then model. The segments are defined by conditions that appear in the "if" part of the rule, which can be range tests on real numbers, integers, and ordinal numbers, or subset tests on nominal values. Predictions are made by statistical models that appear in the "then" part of rules, which can be linear regression with feature selection, logistic regression with feature selection, or more specialized models.

Viewed from the broadest perspective, ProbE can be described as an extensible, embeddable, and scalable segmentation-based modeling engine. Although virtually any predictive modeling technique can be implemented within the ProbE software environment, the ProbE application programming interfaces (APIs) are particularly well suited for implementing segmentation-based modeling techniques in which data records are partitioned into segments and separate predictive models are developed for each segment. This style of modeling is popular in the fields of data analysis and applied statistics; however, it is usually approached as a sequential process in which data is first segmented (using, for example, unsupervised clustering algorithms) and predictive models are then developed for those segments. The drawback of this sequential approach is that it ignores the strong influence that segmentation exerts on the predictive accuracies of the models within each segment. Good segmentations tend to be obtained only through trial and error by varying the segmentation criteria.

ProbE, on the other hand, is able to perform segmentation and predictive modeling within each segment simultaneously, thereby optimizing the segmentation so as to maximize overall predictive accuracy. The benefit of this optimizing approach is that it can produce better models than might otherwise be obtained. ATM-SE (advanced targeted marketing for single events) is an application built on top of ProbE that exploits this approach for mining high-dimensional customer interaction and promotion histories for modeling customer profitability and response likelihood for the retail industry [15]. An evaluation of ATM-SE was recently conducted with a leading U.S. direct-mail retailer that is considered to be a sophisticated user of predictive analytics in its targeted marketing efforts. The segmentation-based response models produced by ProbE outperformed the retailer's proprietary models, which had also been developed using a segmentation-based methodology. The outcome of this evaluation is significant because numerous vendors and consultants had attempted to surpass the retailer's in-house modeling capability in the past, but none had previously succeeded. Moreover, ProbE achieved this result in a fully automated mode of operation with no manual intervention. Although further development and testing is needed, early indications are that ProbE will be able to consistently produce high-quality models for this application on a fully automated basis without requiring costly manual adjustments of the models or the mining parameters by data mining experts, a necessary step in making data mining attractive to medium-sized businesses.

A key feature of ProbE is that it can readily be extended so as to construct virtually any kind of predictive model within a segment. For example, in the UPA (underwriting profitability analysis) application [16], a joint Poisson/log-normal statistical model is used to simultaneously model the frequency with which insurance claims are filed and the amounts (i.e., severities) of those claims for each segment. The segments identified by ProbE thus correspond to distinct risk groups whose loss characteristics (i.e., claim frequency and severity) are estimated in accordance with standard actuarial practices. A second example is found in the ATM-SE application for predicting customer response to promotional mailings. For

this application, segment models were constructed using least-squares linear regression with forward stepwise feature selection to select the variables that appear in the regression equations. In this case, ProbE constructs piecewise-linear models in which the segments correspond to regions of the response surface that are approximately linear and the boundaries between segments correspond to nonlinearities detected in that surface. No matter what kind of predictive models are used within each segment, the same segmentation algorithms can be used to optimize the predictive accuracies of the resulting ensemble of models without regard to their internal details.

In addition to being extensible with respect to segment models, ProbE also permits extensions to be made to its segmentation algorithms, i.e., the procedures used for determining the segments within which models are fit. This degree of extensibility was achieved through careful design of ProbE APIs. In particular, a single API is used to implement all predictive modeling algorithms, including segmentation algorithms. This model API is general enough to permit virtually any predictive modeling technique to be implemented within ProbE. Another important aspect of the API is that it enables one type of model to be embedded within another type of model without either knowing the internal details of the other. The model API also facilitates the development of solution-specific mining methods for generating highly accurate predictive models from data for specialized applications. It can also support the development of customer-specific models and, potentially, third-party models that could be dynamically linked with ProbE without modifications to ProbE itself.

In addition to being extensible, ProbE is designed to be an embedded system that can be incorporated into industry-specific application environments. For example, ProbE does not have a graphical user interface (GUI) of its own; instead, one would have to be supplied by the host application if so desired, as is done in the UPA and ATM-SE applications. The interface to ProbE has been kept as simple as possible. Host applications provide ProbE with specifications of data mining tasks to be performed, and ProbE returns the results of those tasks upon completion. At present, communications are conducted through specification and results files; however, future extensions to ProbE will permit full integration with relational database systems, with task specifications and mining results communicated through database tables. Moreover, the current specification and results files are similar to XML documents in terms of their structures, so future extensions to incorporate XML-based communications could also be readily accomplished.

A third key consideration in the design of ProbE is scalability. ProbE is designed to work with very large, out-of-core data sets. ProbE is also designed to be data-partition parallelized (i.e., large data sets are partitioned across multiple processors, with each processor accessing data only in the partition assigned to it and with only statistical summary information being exchanged among processors). Because this approach minimizes the amount of communication among processors, it is anticipated that it will achieve near-linear improvements in execution speed (i.e., increasing the number of processors by a factor of n decreases the execution time by a factor of n).

The ProbE model API is already designed to support parallelization when it is finally implemented. Moreover, some of the implementation details required by the parallelization scheme (i.e., aggregation of statistical summary information) are already being exploited in ProbE segmentation procedures to reduce the computation of those algorithms. Our ongoing research in this area is centered around designing ProbE as a parallelizable database extender to make its capabilities available to database application developers. Data mining is having its greatest impact in building database-centric vertical solutions. By coupling ProbE tightly to a database, one can avoid data-access bottlenecks, and at the same time leverage query optimization capabilities that are offered in parallel databases, for parallelizing the mining analytics of ProbE.

Exploratory directions

The core effort of the group continues to be centered around predictive modeling analytics, in the context of building innovative solutions for business and industry, and integration of these analytics into middleware components. Additionally, we investigate new approaches and techniques in machine learning and statistical modeling that can provide future bases for robust and accurate predictive mining analytics. Our approach is to make advances in core methods and to evaluate these advances in specific application contexts. Areas specifically being explored include collaborative filtering and item set recommendation algorithms, ensemble-based methods for predictive modeling, and applications of data mining analytics to unstructured information, such as text and document collections.

Collaborative filtering and item set recommenders

Recommendation systems provide a type of customization that has become popular on the Internet. Most search engines use them to group relevant documents. Some newspapers allow news customization. E-commerce sites recommend purchases on the basis of their other customers' preferences. The main advantages of recommendation systems stem from ostensibly better-targeted promotions. This promises higher sales, more advertising revenues, less search by customers to get what they want, and consequently, greater customer loyalty.

Collaborative filtering [17] is one class of recommendation system that mimics word-of-mouth recommendations. A related task is to compare two people and assess how closely they resemble each other. The general concept of nearest-neighbor methods, matching a new instance to similar stored instances, is well known [18]. Collaborative filtering methods use this fundamental concept, but differ in how data are encoded, how similarity is computed, and how recommendations are computed.

We have developed computationally efficient methods for collaborative filtering that processes binary-encoded data. Examples of transactions that can be described in this manner are items purchased by customers or web pages visited by individuals. As with all collaborative filtering, the objective is to match one customer's records with similar records of other customers. For example, on the basis of a customer's prior purchases, one might recommend new items for purchase by examining stored records of other customers who made similar purchases. Because the data are binary (true-or-false)-encoded, and not ranked preferences on a numerical scale, it is possible to develop efficient and lightweight schemes for compactly storing the data, computing similarities between new and stored records, and making recommendations tailored to individuals. Our preliminary results are promising and are competitive with published benchmarks [19].

We have also developed a new statistical modeling approach to the item set recommendation problem [20]. A market basket is a set of items that are purchased together in a single commerce transaction. A partial basket represents a market basket that contains some items, but not the final set that will be purchased. A problem of analytic interest is in determining items to recommend, given the identity of items in a partial basket. Our main contribution here is the notion of explicit separation of the associated next choice given the current partial basket from the independent (renewal) item choice in modeling the next-choice probabilities for each item. It was shown that the new approach is more accurate than existing techniques when the size of partial baskets is not large.

Document matching and FAQ generation

We have developed a fast document matcher that matches new documents to those stored in a database [21]. The matcher lists in order those stored documents that are most similar to the new document. The new documents are typically detailed problem descriptions or free-form textual queries of unlimited length, and the stored documents are potential answers, such as responses to frequently asked questions, or service tips. The method uses minimal data structures and lightweight scoring algorithms that perform efficiently even in restricted environments, such as mobile or small desktop computers. Evaluations on benchmark document collections demonstrate that predictive performance for multiple document matches is competitive with that of more computationally expensive procedures.

Utilizing the lightweight document-matcher algorithm, we developed a prototype solution for FAQ (frequently asked questions) generation. The methodology relies upon a new lightweight document-clustering method [22] that operates in high dimensions, processes tens of thousands of documents, and groups them into several thousand clusters or, by varying a single parameter, into a few dozen clusters. The method uses a reduced indexing view of the original documents in which only the k best keywords of each document are indexed. An efficient procedure for clustering is specified in two parts: a) compute the k most similar documents for each document in the collection, and b) group the documents into clusters using these similarity scores. For FAQ generation, we applied this method to more than 50 000 customer service problem reports. These reports were reduced to 3000 clusters, from which 5000 exemplar documents (FAQs) were generated. Results demonstrate efficient clustering performance with excellent group similarity measures.

Ensemble methods for predictive modeling

The predictive performance of a model can sometimes be not nearly as strong on unseen data as that obtained on the training data. In this phenomenon, often described as overfitting, the model is too specialized to the training data. Overfitting may arise because of the existence of high variance in the data. Researchers have observed that the variance can be greatly reduced by inducing multiple models (an ensemble) from the same data. The classification of an unseen case is then determined by a weighted combination of the classifications assigned by the multiple models.

Techniques such as stacking [23], bagging [24], and boosting [25] have been developed to utilize ensembles of models in predictive analytics. These methods either combine models developed using different techniques on the same data, or combine models developed using the same technique but on different subsets of the data.

In the case of decision-tree building, the key step in inducing multiple trees is in the sampling of the training data. In the simpler approach, called bagging [24], a sample of size n is taken with replacement from the original set of n examples. Some examples are repeated in the sample, while others may not occur. Thus, it is possible to generate many samples and induce trees from each sample. An alternative method of sampling, boosting [25], usually performs better than bagging. Instead of sampling all cases randomly, so that each data point has a $1/n$ chance of being drawn from the sample, an

incremental approach is used to bias the selection. The objective is to increase the odds of selecting examples that have been erroneously classified by the trees that have been induced in previous iterations.

The concept of boosting, i.e., adaptive resampling, applies to many learning methods. However, resampling is particularly advantageous when used in conjunction with decision trees for two key reasons: Decision-tree algorithms are relatively efficient in high dimensions, and decision trees tend to have a bigger component of model variance than other methods such as nearest neighbors or neural nets. We have been exploring applications of adaptive resampling to many problems in predictive modeling and their applications, and have observed dramatic improvements in the accuracy of results in many instances. These include financial portfolio management [10], text categorization [13], active learning [26], and fast methods for rule-based classification and regression [27, 28].

Conclusion

Increased attention and focus on decision support solutions using data mining techniques has refueled interest in classification and regression modeling, particularly in techniques [29] that permit the automatic generation of interpretable models from high-dimensional data. Our goal continues to be the exploration, application, and improvement of this technology, motivated by key business problems such as insurance risk management, targeted marketing, financial credit scoring, text mining, and consumer recommendation systems.

While some aspects of this technology have reached maturity and have become stable, there are many issues that remain open. Symbolic modeling approaches that remain consistently robust across a wide variety of data sets are not yet well understood. Additionally, though some of these techniques are conceptually robust and elegant, they prove to be computationally challenging when applied to large-scale business and industrial data sets. Another open issue that continues to be explored is the characterization of data sets, using simple measures, statistical measures, or information-theoretic measures, that will allow an educated mapping of the most appropriate mining technique to a data set for maximizing the accuracy of the resulting solution.

Research in the underlying algorithms is far from done. If one is to use scalability, accuracy, robustness, and interpretability as the criteria by which to judge data mining algorithms, no existing algorithms simultaneously excel in all criteria. Can existing techniques be modified, or new algorithms designed, that are scalable (so that the size of data does not pose a problem), robust (work well in a wide variety of domains), accurate (so that information extracted from the data continues to hold up outside and beyond the immediate data), and interpretable (providing insight and value to the end user)? No existing algorithms excel in all criteria. Continuing research is also needed to extend and adapt data mining algorithms so that they can operate on richer collections of data types. Data is no longer just numerical or discrete. It can be unstructured text, video clips, or audio clips, and the amount of data in these newer data types is growing dramatically.

Developing scalable and automated predictive mining techniques for extracting useful knowledge from diverse sources of data is expected to be the motivating thrust for research as we move forward.

Acknowledgment

We thank the many individuals at IBM, within the Research Division as well as in IBM Industry Sectors, Global Services, and the Software Group, from whose collaborations we have immensely benefited.

References

1. M. James, *Classification Algorithms*, John Wiley & Sons, Inc., New York, 1985.
2. H. Scheffe, *The Analysis of Variance*, John Wiley & Sons, Inc., New York, 1959.
3. D. Michie, D. Spiegelhalter, and C. Taylor, *Machine Learning, Neural and Statistical Classification*, Ellis Horwood, Chichester, UK, 1994.
4. B. Ripley, *Pattern Recognition and Neural Networks*, Cambridge University Press, Cambridge, UK, 1996.
5. S. M. Weiss and C. A. Kulikowski, *Computer Systems That Learn*, Morgan Kaufmann, San Francisco, 1991.
6. S. J. Hong, "R-MINI: An Iterative Approach for Generating Minimal Rules from Examples," *IEEE Trans. Knowledge Discovery & Data Eng.* **9,** No. 5, 709–717 (1997).
7. S. J. Hong, "Use of Contextual Information for Feature Ranking and Discretization," *IEEE Trans. Knowledge Discovery & Data Eng.* **9,** No. 5, 718–730 (1997).
8. S. J. Hong, J. R. M. Hosking, and S. Winograd, "Use of Randomization to Normalize Feature Merits," *Proceedings of the Conference on Information, Statistics, and Induction in Science (ISIS'96)*, 1996, pp. 10–19.
9. C. Apte, S. J. Hong, J. Lepre, S. Prasad, and B. Rosen, "RAMP: Rules Abstraction for Modeling and Prediction," *Research Report RC-20271*, IBM Thomas J. Watson Research Center, Yorktown Heights, NY, 1996.
10. C. Apte and S. J. Hong, "Predicting Equity Returns from Securities Data with Minimal Rule Generation," *Advances in Knowledge Discovery and Data Mining*, AAAI Press/ The MIT Press, Cambridge, MA, 1995, pp. 541–560.
11. C. Apte, F. Damerau, and S. M. Weiss, "Automated Learning of Decision Rules for Text Categorization," *ACM Trans. Info. Syst.* **12,** No. 3, 233–251 (July 1994).
12. C. Apte, F. Damerau, and S. M. Weiss, "Towards Language Independent Automated Learning of Text Categorization Methods," *Proceedings of the ACM Special Interest Group on Information Retrieval (SIGIR'94)*, 1994, pp. 23–30.
13. S. M. Weiss, C. Apte, and F. Damerau, "Maximizing Text-Mining Performance," *IEEE Intell. Syst.* **14,** No. 4, 63–70 (1999).

14. C. V. Apte, R. Natarajan, E. P. D. Pednault, and F. A. Tipu, "A Probabilistic Estimation Framework for Predictive Modeling Analytics," *IBM Syst. J.* **41,** No. 3, 438–448 (2002).
15. C. Apte, E. Bibelnieks, R. Natarajan, E. P. D. Pednault, F. Tipu, D. Campbell, and B. Nelson, "Segmentation-Based Modeling for Advanced Targeted Marketing," *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2001)*, August 2001, pp. 408–413.
16. C. Apte, E. Grossman, E. P. D. Pednault, B. Rosen, F. Tipu, and B. White, "Probabilistic Estimation Based Data Mining for Discovering Insurance Risks," *IEEE Intell. Syst.* **14,** No. 6, 49–58 (November/December 1999).
17. D. Goldberg, D. Nichols, B. Oki, and D. Terry, "Using Collaborative Filtering to Weave an Information Tapestry," *Commun. ACM* **35,** No. 12, 61–70 (1992).
18. T. Cover and P. Hart, "Nearest Neighbour Pattern Classification," *IEEE Trans. Info. Theory* **13,** 21–27 (1967).
19. S. M. Weiss and N. Indurkhya, "Lightweight Collaborative Filtering Method for Binary-Encoded Data," *Proceedings of the Fifth European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD)*, September 2001, pp. 484–491.
20. S. J. Hong, R. Natarajan, and I. Belitskaya, "A New Approach for Item Choice Recommendations," *Proceedings of the Third International Conference on Data Warehousing and Knowledge Discovery* (*DaWak-01*), September 2001, pp. 131–140.
21. S. M. Weiss, B. F. White, C. Apte, and F. Damerau, "Lightweight Document Matching for Help Desk Applications," *IEEE Intell. Syst.* **15,** No. 2, 57–61 (March/April 2000).
22. S. M. Weiss and C. Apte, "Automated Generation of Model Cases for Help-Desk Applications," *Research Report RC-22061*, IBM Thomas J. Watson Research Center, Yorktown Heights, NY, 2001.
23. D. Wolpert, "Stacked Generalization," *Neural Networks* **5,** No. 2, 241–259 (1992).
24. L. Breiman, "Bagging Predictors," *Machine Learning* **24,** No. 2, 123–140 (1996).
25. Y. Freund and R. Schapire, "Experiments with a New Boosting Algorithm," *Proceedings of the International Machine Learning Conference*, 1996, pp. 148–156.
26. V. Iyengar, C. Apte, and T. Zhang, "Active Learning Using Adaptive Resampling," *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2000)*, August 2000, pp. 91–98.
27. S. M. Weiss and N. Indurkhya, "Lightweight Rule Induction," *Proceedings of the International Conference on Machine Learning (ICML)*, 2000, pp. 1135–1142.
28. N. Indurkhya and S. M. Weiss, "Solving Regression Problems with Rule-Based Ensemble Classifiers," *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2001)*, August 2001, pp. 287–292.
29. U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, Eds., *Advances in Knowledge Discovery and Data Mining*, AAAI Press/The MIT Press, Cambridge, MA, 1995.

Received October 2, 2001; accepted for publication July 17, 2002

Chidanand V. Apte *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (apte@us.ibm.com).* Dr. Apte is Manager of the Data Abstraction Research Group at the IBM Thomas J. Watson Research Center. He received his Ph.D. degree in computer science from Rutgers University in 1984. His research interests include knowledge discovery and data mining, applied machine learning and statistical modeling, and business intelligence automation.

Se June Hong *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (sejune@us.ibm.com).* Dr. Hong is a Research Staff Member in the Data Abstraction Research Group at the IBM Thomas J. Watson Research Center. He received his Ph.D. degree in electrical engineering from the University of Illinois at Urbana–Champaign in 1969. His research interests are in machine learning and statistical modeling. Dr. Hong is currently working on probabilistic classification methods and their applications to complex business problems.

Ramesh Natarajan *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (nramesh@us.ibm.com).* Dr. Natarajan is a Research Staff Member in the Data Abstraction Research Group at the IBM Thomas J. Watson Research Center. He received his Ph.D. degree in chemical engineering from the University of Texas at Austin in 1984. Dr. Natarajan is currently working on statistical data mining applications, modeling algorithms for massive data sets, and database analytic extenders.

Edwin P. D. Pednault *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (pednault@us.ibm.com).* Dr. Pednault is a Research Staff Member in the Data Abstraction Research Group at the IBM Thomas J. Watson Research Center. He received his Ph.D. degree in 1987 at Stanford University, working with Robert C. Moore at SRI International on the mathematical foundations of automatic planning. His current research interests include data mining, statistical learning theory, and reinforcement learning.

Fateh A. Tipu *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (fateh@us.ibm.com).* Mr. Tipu is an Advisory Software Engineer in the Data Abstraction Research Group at the IBM Thomas J. Watson Research Center. He received his M.S. degree in electrical engineering from the University of Wisconsin at Madison in 1991. Mr. Tipu's technical interests include software development, data mining, and CAD tools for logic design.

Sholom M. Weiss *IBM Research Division, Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598 (sholom@us.ibm.com).* Dr. Weiss is a Research Staff Member in the Data Abstraction Research Group at the IBM Thomas J. Watson Research Center. He received his Ph.D. degree in computer science from Rutgers University in 1974. His research interests are in machine learning. Dr. Weiss is currently working on classification pattern generation techniques and their applications to Market Intelligence solutions.