

# Selected Prior Research

Robert L. Grossman

March, 2006

## Data Mining Systems

- **1996 - scaled tree-based classifiers to very large data sets.** A fundamental challenge in data mining is to mine data sets that are so large that they do not fit into a computer's memory. This is important for a wide variety of applications ranging from homeland defense to identifying fraudulent credit card transactions. One of the most accurate techniques in data mining is tree-based classifiers and predictors. Our 1996 paper [16] described a method for computing tree-based classifiers on data sets that are too large to fit into a computer's memory. The first idea is to partition the data, build individual trees on each partition, and then combine the trees using an ensemble, or collection, of classifiers. The second idea is to use stratified sampling to oversample rare events and distribute them over the various partitions. This was essentially a variant of a type of sampling called bootstrapping. This technique was implemented in Magnify's 1996 version of the PATTERN data mining system and was called Averaged Classification Trees/Averaged Regression Trees or ACT/ART. PATTERN was the first data mining system to build very accurate classifiers on data sets that could not fit into a computer's memory, allowing classifiers in 1996 to be built on terabyte size data sets when memory was measured in megabytes and disks in gigabytes. The 1996 paper by Breiman [1] presented a complementary idea called bagging in which ensembles of trees are built over small data sets by repeated sampling with replacement (another variant of bootstrapping). Building ensembles of trees via partitioning and appropriate bootstrapping is still considered by many to be the most effective algorithm for detecting rare events in large data sets.
- **1997 - decreased the time and cost to deploy new data mining models.** Although companies in the 1990's began quite a few data mining projects, many were not as successful as anticipated. One of the reasons for this lack of success is that although a lot of time and energy was spent building statistical and data mining models, it was often very difficult to deploy these models in operational systems and to update them. In 1995-1997, I worked with members of the Terabyte Challenge Testbed to introduce what are now called scoring engines. The basic idea is to

separate architecturally producers and consumers of statistical and data mining models and to define an XML format so that a complete description of the model can be passed easily and safely between the producer and consumer. With this approach, scoring engines can be integrated once into operational systems and analytical models can be updated instantly, simply by reading an XML file. Prior to this, most operational deployments of analytics required re-coding the analytics when updates were required. With scoring engines, the time to deploy new models was reduced from months to weeks, or even days. Magnify introduced the first scoring engine in 1997, and today scoring engines are provided by most data mining vendors including SAS, SPSS, IBM, Oracle and Microsoft. As more vendors introduced scoring engines, it became desirable to develop a XML standard for statistical modeling, data mining and business intelligence, and so in 1998, I co-founded the Data Mining Group (DMG) for this purpose. Today, most vendors of data mining and statistical tools belong to the DMG and work together to support one standard XML language for analytics and business intelligence called the Predictive Model Markup Language or PMML. PMML is described in [18] and [24]. There have also been three ACM KDD Workshops on Data Mining Standards, Services and Platforms (DM-SSP), in 2004, 2005, and 2006, each of which had several papers on PMML. Prior to KDD 2001, 2002, and 2003, there was also a co-located workshop on PMML.

- **1998 - introduced data webs to simplify the exploration and integration of remote and distributed data.** Today, there are many good algorithms for *building* statistical and data mining models. Scoring engines are good infrastructure for *deploying* statistical and data mining models in operational systems. On the other hand, there is no good infrastructure for discovering relevant data, exploring it efficiently, and *integrating* it to produce the learning sets required for data mining. In 1998, I worked with a team of software engineers and built a prototype of a data web — just as the world wide web today allows remote and distributed documents to be accessed with a point and a click, data webs are designed to do the same thing for data. The basic idea is to introduce a protocol that directly supports data, metadata, and universal keys, and standard operations involving them, such as range queries on distributed columns, or integrating two distributed columns of data using a common key. The NSF-funded DataSpace Project (1998-2003) developed several open source tools for creating data webs that can be used to lower the cost and improve the speed of data integration when working with distributed scientific, engineering, or business data. Data webs were introduced in [23]. During this period, we developed several data web applications, including applications in astronomy [22], earth science [27], and proteomics [28]. A framework for data integration is described in [26].

## High Performance Computing and Networking

- **1991 - developed distributed computing infrastructures for data sharing, analysis and collaboration.** From 1991-1995, I was the co-director of the DOE sponsored PASS Project that developed technology for the Superconducting SuperCollider. The goal of the PASS project was to develop new technologies so that scientists working around the world could easily collaboratively analyze large and rapidly growing scientific data sets. On the positive side, several important milestones were achieved, including showing how terabyte size data sets could be mined and how virtual organizations could work together using distributed computing and data resources [6], [10]. On the negative side, the cancellation of Superconducting SuperCollider in 1993 ultimately meant that much of the technology developed by the PASS project was orphaned. Over two dozen publications describing this work appeared. The project helped popularize an architecture in which it was natural to view scientific data as distributed collections of objects, accessed via queries and services [6], [8], [15], rather than as distributed files, which was the dominant view at that time. From this viewpoint, technical challenges included: how to scale the indexing, accessing, querying and processing of very large amounts of complex distributed data. Addressing these problems led to a number of publications describing how to scale persistent object stores to very large data sets and how to parallelize the querying and analysis of high energy physics event data, such as the publications [11], [12], [13] [14].
- **1994 - developed grid computing technology for data analysis and data mining.** In 1994, I co-founded a project called the National Scalable Cluster Project (NSCP) which was funded by the National Science Foundation and linked high performance computing clusters in Chicago, Philadelphia and College Park, Maryland using high performance wide area networks [25]. This allowed us to create one of the earliest grid based computers, which we called the NSCP Meta-Cluster, since it was a cluster of clusters. During the period 1994-1998, we developed a grid based system for data analysis and data mining called Papyrus [19] [21], and developed several data intensive grid applications involving high energy physics data, health care outcomes data, and medical imaging data. The NSCP created several open source tools for grid computing: over time, commercial applications were brought to the market by Magnify in financial services and by i3Archive in medical imaging. Three of the fundamental technical challenges faced by the project were: i) how to develop cluster-based versions of some of the common data mining algorithms; ii) how to transport large data sets over high performance networks with high bandwidth delay products; and iii) how to integrate more efficiently distributed data. The first challenge led to the work described above on building tree-based classifiers over workstation clusters. The second chal-

lenge led to the work described next in network protocols. The third challenge led to the work described above in data integration.

- **2000 - introduced network striping to improve high performance data transport.** In data mining, interesting data is almost always in another location. Unfortunately, moving large data sets over long distances is a challenge, even with today's high performance networks. In [20], H. Sivakumar, S. Bailey and I introduced network striping to improve high performance data transport over wide area networks, roughly analogous to how striping in RAID disks improves the performance of local disk systems. We also developed an open source tool employing network striping called Pockets, and built a number of applications using Pockets. A similar idea was introduced at the same time in [2] by A. Chervenak, I. Foster, C. Kesselman and S. Tuecke. Today, network striping is used in GridFTP, which is part of the Globus toolkit for grid computing, and is quite common.
- **2003 - developed a new network protocol for networks with high bandwidth delay products.** Today, there is a fundamental need to find new network protocols so that emerging wide area high performance 1 and 10 Gbps can work effectively with large remote and distributed data sets. In 2002, Gu, Hong and I developed a new protocol called SABUL [31], [29] which for some time held the application record for high performance data transport — for example, it was used to move over 1.4 Terabytes of astronomical data across the Atlantic in November, 2002 in less than thirty minutes. In 2004, we developed a successor to SABUL called UDT [36], [33], [34], [38]. UDT is a type of algorithm that employs additive increases when there is no negative event from the receiver, such as a packet loss or increasing delay time for the packet, and multiplicative decreases when there is such an event. These are called additive increases, multiplicative decreases algorithms or AIMD. TCP is the most famous example of this type of algorithm. With UDT, we introduced a new type of additive increases called decreasing additive increases and showed that congestion control algorithms, such as UDT, employing decreasing additive increases effectively used the available bandwidth for high bandwidth delay product networks, and yet were fair to other high volume flows and friendly to TCP flows. UDT set a number of milestones for high performance data transport at SC 04, 05 and 06. We have used UDT to build a variety of data intensive applications, including applications for transporting and analyzing earth science data [32] and astronomy data. UDT has been downloaded over 5000 times and is used by several collaborations. In particular, it is used by the Sloan Digital Sky Survey (SDSS) to distribute its data sets to collaborators world wide. We have also integrated UDT with user-controlled light paths and developed several data intensive applications over this stack [35].

## Algorithms

- **1989 - discovered a natural multiplication on data structures of trees.** Trees are one of the most useful data structures in computer science and are used as the underlying data structures for many algorithms. In [4], Larson and I showed that there is a natural multiplication on trees. More precisely, if  $H$  is the vector space whose basis consists of rooted trees, then we showed that  $H$  has a natural product on it. We also showed that the dual  $H^*$  of  $H$  also has a natural product and that these are compatible in such a way that  $H$  has the structure of what is called a Hopf algebra. Using this multiplication, it is easy to derive a number of new algorithms: Larson and I derived efficient algorithms for the symbolic computation of derivations in [7]; Crouch, Larson and I derived a new type of numerical integration algorithm for Lie groups which generalizes Runge-Kutta type algorithms [9]; Grayson and I derived explicit formulas for flows on nilpotent Lie groups [5]. More recently, Larson and I showed how to use a connection so that families of trees can be given a natural differential algebra structure. In 1998, Connes and Kreimer showed that an algebra of trees is important in quantum field theory [3]. This algebra, called the Connes-Kreimer algebra, is the dual to the Hopf algebra we discovered.
- **2003 - discovered a simple way to assign unique IDs to chemical compounds.** In 2003, Kasturi, Hamelberg, Liu and I showed that there is a natural graph algorithm that can assign essentially unique IDs to chemical compounds. We call these unique chemical IDs or UCKs [30]. The basic idea was to define certain classes of *natural operations* on labeled graphs. For example, one can assign a new (long) label to a node by forming the set of all labeled paths of depth  $d$  or less originating from the node, lexicographically ordering the resulting set, and then concatenating the results. Using a few such operations one can define keys for labeled graphs, which we showed are unique for the types of graphs associated with chemical compounds. In contrast, today, most databases of chemical compounds assign ascension numbers to new chemical compounds when they are added to the database (1, 2, 3, etc.). We showed that over 10% of the chemical compounds in one of the most popular databases of chemical compounds were duplicates and were not detected because of the defects in the current ascension numbers [30]. The lack of keys for chemical compounds makes it extremely difficult to compare chemical compounds across two different databases. Using UCKs, data can be integrated easily from multiple databases containing chemical compounds, facilitating the discovery of new drugs and therapeutic treatments.

## For More Information

More information and drafts of many of the papers referenced can be found online at [www.rgrossman.com/articles.htm](http://www.rgrossman.com/articles.htm).

## References

- [1] Leo Breiman, Bagging Predictors, *Machine Learning*, Volume 24, No. 2, pp. 123-140, 1996.
- [2] A. Chervenak, I. Foster, C. Kesselman and S. Tuecke, Protocols and Services for Distributed Data-Intensive Science, *ACAT2000 Proceedings*, pages 161-3, 2000.
- [3] A. Connes and D. Kreimer, Hopf algebras, renormalization and noncommutative geometry, *Communication in Mathematical Physics*, Volume 199, pages 203-242, 1998.
- [4] R. Grossman and R. Larson, Hopf algebraic structures of families of trees, *Journal of Algebra*, Volume 26, 1989, pages 184-210.
- [5] M. Grayson and R. Grossman, Models for free, nilpotent Lie algebras, *Journal of Algebra*, Vol. 35, 1990, pages 177-191.
- [6] A. Baden and R. Grossman, Database computing and high energy physics, *Computing in High-Energy Physics 1991*, edited by Y. Watase and F. Abe, Universal Academy Press, Inc., Tokyo, 1991, pp. 59-66.
- [7] R. Grossman and R. Larson, The symbolic computation of derivations using labeled trees, *Journal of Symbolic Computation*, Volume 13, pages 511-523, 1992.
- [8] C. T. Day, S. Loken, J. F. MacFarlane, E. May, D. Lifka, E. Lusk, L. E. Price, A. Baden, R. Grossman, X. Qin, L. Cormell, P. Liebold, D. Liu, U. Nixdorf, B. Scipioni, T. Song, Database Computing in HEP – Progress Report, *Proceedings of the International Conference on Computing in High Energy Physics '92*, C. Verkerk and W. Wojcik, editors, CERN-Service d'Information Scientifique, 1992, ISSN 0007-8328, pp. 557-560.
- [9] P. Crouch and R. Grossman, Numerical integration of ordinary differential equations on manifolds, *Journal of Nonlinear Science*, Volume 3, pages 1-33, 1993.
- [10] D. R. Quarrie, C. T. Day, S. Loken, J. F. Macfarlane, D. Lifka, E. Lusk, D. Malon, E. May, L. E. Price, L. Cormell, A. Gauthier, P. Liebold, J. Hilgart, D. Liu, J. Marstaller, U. Nixdorf, T. Song, R. Grossman, X. Qin, D. Valsamis, M. Wu, W. Xu, A. Baden, The PASS Project: A Progress Report, *Proceedings of the Conference on Computing in High Energy Physics 1994*, edited by S. C. Loken, pages 229-232, 1995.
- [11] R. L. Grossman, Working With Object Stores of Events Using PTool, 1993 Cern Summer School in Computing, C. E. Vandoni and C. Verkerk, editors, CERN-Service d'Information Scientifique 94-06, pages 66-97, 1994.

- [12] R. L. Grossman and X. Qin, Ptool: a scalable persistent object manager, Proceedings of SIGMOD 94, ACM, 1994, page 510.
- [13] E. N. May, D. Lifka, D. Malon, L. E. Price L. Cormell, A. Gauthier, J. Marsteller, S. Mestad, U. Nixdorf R. Grossman, X. Qin, D. Valsamis, M. Wu, W. Xu A Demonstration of a Multi-level Object Store and its Application to the Analysis of High Energy Physics Data, Proceedings of the Conference on Computing in High Energy Physics 1994, edited by S. C. Loken, pages 236-238, 1995.
- [14] R. L. Grossman, N. Araujo, X. Qin, and W. Xu, Managing physical folios of objects between nodes, Persistent Object Systems (Proceedings of the Sixth International Workshop on Persistent Object Systems), M. P. Atkinson, V. Benzaken and D. Maier, editors, Springer-Verlag and British Computer Society, 1995, pages 217-231.
- [15] D. R. Quarrie, C. T. Day, S. Loken, J. F. Macfarlane, D. Lifka, E. Lusk, D. Malon, E. May, L. E. Price, L. Cormell, A. Gauthier, P. Liebold, J. Hilgart, D. Liu, J. Marstaller, U. Nixdorf, T. Song, R. Grossman, X. Qin, D. Valsamis, M. Wu, W. Xu, A. Baden, The PASS Project Architectural Model, Proceedings of the Conference on Computing in High Energy Physics 1994, edited by S. C. Loken, pages 233-235, 1995.
- [16] R. L. Grossman, H. Bodek, D. Northcutt, and H. V. Poor, Data Mining and Tree-based Optimization, Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, E. Simoudis, J. Han and U. Fayyad, editors, AAAI Press, Menlo Park, California, 1996, pp 323-326.
- [17] Robert Grossman, and Marco Mazzucco, DataSpace - A Web Infrastructure for the Exploratory Analysis and Mining of Data, IEEE Computing in Science and Engineering, July/August, 2002, pages 44-51.
- [18] P. Hallstrom, I. Pulley and X. Qin, The Management and Mining of Multiple Predictive Models Using the Predictive Modeling Markup Language (PMML), Information and Software Technology, Volume 41, 1999, pages 589-595.
- [19] R. L. Grossman, S. Bailey, A. Ramu, B. Malhi and H. Sivakumar, A. Turinsky, Papyrus: A System for Data Mining over Local and Wide Area Clusters and Super-Clusters, Proceedings of Supercomputing 1999, IEEE.
- [20] H. Sivakumar, S. Bailey, R. L. Grossman, Pockets: The Case for Application-level Network Striping for Data Intensive Applications using High Speed Wide Area Networks, Proceedings of the 2000 ACM/IEEE Conference on Supercomputing (CDROM), IEEE Computer Society, Washington, DC, USA, 2000, page 38.
- [21] R. L. Grossman, S. Bailey, A. Ramu, B. Malhi and A. Turinsky, The Preliminary Design of Papyrus: A System for High Performance, Distributed

- Data Mining over Clusters, in *Advances in Distributed and Parallel Knowledge Discovery*, H. Kargupta and P. Chan, editors, AAAI Press/The MIT Press, Menlo Park, California, 2000, pages 259-275.
- [22] A DataSpace Infrastructure for Astronomical Data, Robert Grossman, Emory Creel, Marco Mazzucco, Roy Williams in R. L. Grossman, C. Kamath, W. Philip Kegelmeye, V. Kumar, and R. Namburu, *Data Mining for Scientific and Engineering Applications*, Kluwer Academic Publishers, 2001, pages 115-123.
  - [23] Robert Grossman, and Marco Mazzucco, DataSpace — A Web Infrastructure for the Exploratory Analysis and Mining of Data, *IEEE Computing in Science and Engineering*, July/August, 2002, pages 44-51.
  - [24] Robert Grossman, Mark Hornick, and Gregor Meyer, Data Mining Standards Initiatives, *Communications of the ACM*, Volume 45-8, 2002, pages 59-61.
  - [25] R. L. Grossman and R. Hollebeek, The National Scalable Cluster Project: Three Lessons about High Performance Data Mining and Data Intensive Computing, in *Handbook of Massive Data Sets*, J. Abello, P. M. Pardalos, and M. G. C. Resende, editors, Kluwer Academic Publishers, 2002.
  - [26] Ian Foster and Robert L. Grossman, Data Integration in a Bandwidth Rich World, *Communications ACM*, Volume 46, Issue 11, November, 2003, pages 50-57.
  - [27] Asvin Ananthanarayan, Rajiv Balachandran, Yunhong Gu, Robert Grossman, Xinwei Hong, Jorge Levera, Marco Mazzucco, *Data Webs for Earth Science Data*, *Parallel Computing*, Volume 29, 2003, pages 1363-1379.
  - [28] Robert Grossman, Donald Hamelberg, Pavan Kasturi, and Bing Liu, Experimental Studies of the Universal Chemical Key (UCK) Algorithm on the NCI Database of Chemical Compounds, *Proceedings of the 2003 IEEE Computer Society Bioinformatics Conference (CSB 2003)*, IEEE Computer Society, Los Alamitos, California, pages 244-250.
  - [29] A. Chien, T. Faber, A. Falk, J. Bannister, R. Grossman, J. Leigh, Transport Protocols for High Performance: Whither TCP?, *Communications ACM*, Volume 46, Issue 11, November, 2003, pages 42-49.
  - [30] Robert L. Grossman, Pavan Kasturi, Donald Hamelberg, Bing Liu, An Empirical Study of the Universal Chemical Key Algorithm for Assigning Unique Keys to Chemical Compounds, *Journal of Bioinformatics and Computational Biology*, 2004, Volume 2, Number 1, 2004, pages 155-171.
  - [31] Yunhong Gu and Robert L. Grossman, SABUL: A Transport Protocol for Grid Computing, *Journal of Grid Computing*, Volume 1, pages 377-386, 2004

- [32] Robert L. Grossman, Yunhong Gu, Dave Hanley, Xinwei Hong and Parthasarathy Krishnaswamy, Experimental Studies of Data Transport and Data Access of Earth Science Data over Networks with High Bandwidth Delay Products, *Computer Networks*, Volume 46, 2004, pages 411-421.
- [33] Yunhong Gu, Xinwei Hong, and Robert Grossman, Experiences in Design and Implementation of a High Performance Transport Protocol, *ACM/IEEE SC 2004 Conference (SC'04)*, 2004, page 22.
- [34] Yunhong Gu, Xinwei Hong and Robert Grossman, An Analysis of AIMD Algorithms with Decreasing Increases, *Proceedings of GridNets 2004*, IEEE Press, 2004.
- [35] Robert L. Grossman, Yunhong Gu, Dave Hanley, Xinwei Hong, Dave Lillithun, Jorge Levera, Joe Mambretti, Marco Mazzucco, and Jeremy Weinberger, Photonic Data Services: Integrating Path, Network and Data Services to Support Next Generation Data Mining Applications, *Data Mining: Next Generation Challenges and Future Directions*, H. Kargupta, A. Joshi, K. Sivakumar, and Y. Yesha, editors, AAAI Press, 2004.
- [36] Robert L. Grossman, Yunhong Gu, Xinwei Hong, Antony Antony, Johan Blom, Freek Dijkstra, and Cees de Laat, Teraflows over Gigabit WANs with UDT, *Journal of Future Computer Systems*, Elsevier Press, Volume 21, Number 4, 2005, pages 501-513.
- [37] Robert L. Grossman and Richard G. Larson, Differential Algebra Structures on Families of Trees, *Advances in Applied Mathematics*, Volume 35, pages 97-119, 2005.
- [38] Yunhong Gu and Robert L. Grossman, Optimizing UDP-Based Protocol Implementations, *Proceedings of the Third International Workshop on Protocols for Fast Long-Distance Networks PFLDnet 2005*, 2005.
- [39] Robert L. Grossman, Yunhong Gu, David Handley, and Michal Sabala Joe Mambretti, Alex Szalay and Ani Thakar, Kazumi Kumazoe and Oie Yuji, Minsun Lee, Yoonjoo Kwon, and Woojin Seok, Data Mining Middleware for Wide Area High Performance Networks, *Journal of Future Generation Computer Systems (FGCS)*, 2006.