



Beyond Data Mining

Tim Menzies



THE PREDICTIVE MODELING community applies data miners to artifacts from software projects. This work has been very successful—we now know how to build predictive models for software effects and defects and many other tasks such as learning developers’ programming patterns (see the extended version of this article at <http://menzies.us/pdf/13idea.pdf> for more detail).

That said, to truly impact the work of industrial practitioners, we need to change the predictive modeling community’s focus. To date, it has spent too much time on *algorithm mining* when the field is moving into what I call *landscape mining*. To support industrial practitioners, we’re going to have to move on to something I call *decision mining* and then *discussion mining*.

This article compares and contrasts these four kinds of miners shown in Figure 1:

- Algorithm miners explore tuning parameters in data mining algorithms.
- Landscape miners reveal the shape of the decision space.
- Decision miners comment on how best to change a project.
- Discussion miners help the community debate trade-offs between the different decisions.

Note that algorithm and landscape mining are more research-focused activities that explore the miners’ internal details. However, decision and discussion miners are more practitioner-oriented because they’re focused on how a community can use conclusions.

Algorithm Mining

While it’s rarely stated, the original premise of predictive modeling was that predictions should guide software management—in other words, once upon a time, the aim of a prediction was a decision.

Sadly, that original aim seems to be forgotten. Too many researchers in the field are stuck in a rut, publishing papers that spend very little time exploring the data and much more time on the data algorithms. Most of these papers focus on exploring configuration options with the algorithms, rather than reflecting on the underlying data. Recent papers report that there’s little to be gained from such algorithm mining because the “improvements” found from this approach are marginal, at best—for example, for effort estimation and defect prediction, simpler data miners do just as well or better than more elaborate ones.^{1,2}

Landscape Mining

Algorithm mining is a “leap before your look” approach in which researchers throw algorithms at data and then see what comes out. A second approach is the “look before you leap” option—mining the data to find the space of possible inferences before leaping in with the learners. This is the data’s “landscape.”

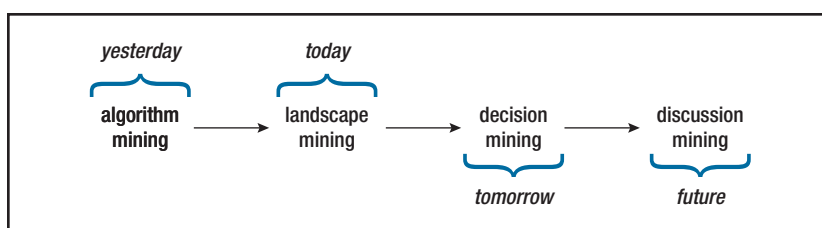


FIGURE 1. Four kinds of miners shown left to right, past to future.

continued on p. 90

continued from p. 92

Consider the W1 case-based reasoning (CBR) system, also known as “Dub-ya” or the “the decider.”³ CBR makes conclusions by inspecting the nearest similar historical cases. To make W1 a landscape miner (which we’ll call W2), we can cluster the training data into a tree of clusters, where child nodes contain subclusters of the parents. Then, a feature selector runs over the data to reject features whose values can’t distinguish between the clusters. Specifically, we’re checking the entropy of each attribute value over all clusters and deleting those with highest entropy. Finally, we can replace all leaf clusters with the median of each cluster. The resulting space of features and examples is very small: dozens of features reduce down to just a handful, and hundreds of examples reduce down to just one example per cluster.

By restricting inference to just some subtree of clusters (where the leaves now contain just one representative example), we can quickly build many local models specialized to particular contexts.

sible for the predictive community modeling community to refocus and redirect its tools toward an interesting new goal.

Decision Mining

At a recent panel on software analytics at ICSE 2012, industrial practitioners reviewed the state of the art in data mining.⁴ Panelists commented that “prediction is all well and good, but what about decision making?” Data mining is useful because it focuses an inquiry onto particular issues, but data miners are subroutines in a higher-level decision process.

To convert W2 into a decision miner (what we’ll call W3), we add contrast set learning. While classifiers report what’s true about different regions of data, contrast set learners report how those regions differ. Contrast sets can be much smaller than classification rules, particularly if they’re generated as a postprocessor to some decision tree process. Contrast sets learned high in a decision tree tend to wipe out most possibilities and select for few classes—they do this by using fewer extra constraints.

W3 uses the same clusters as found by W2, but applies the principle of

sults in that cluster. In a recent *IEEE Transactions on Software Engineering* paper, I showed that such envy-based “local learning” can result in models much better than if we overgeneralize by learning from all the data.⁵

The lesson of W3 is the same as W2: new and innovative approaches to predictive modeling can be achieved by refactoring our current tools.

Discussion Mining

Pablo Picasso once said “computers are stupid; they only give you answers.” Discussion miners aren’t stupid; they know that while predictions and decisions are important, so too are the questions and insights generated on the way to those conclusions. In my view, discussion mining is the next great challenge for the predictive modeling community. In the coming century’s heavily digital world, such discussion tools are going to be essential. Without them, humans will be unable to navigate and exploit the ever-increasing quantity of readily-accessible digital information.

In some sense, discussion miners are the very opposite of the Web:

- The Web was designed for information transport and access, with a primary goal of rapid sharing of new information.
- If the Web were a discussion miner, it would be possible to instantly query each webpage to find other pages with similar (or disputing) beliefs, find the contrast set between then agreeing and disputing pages, and then run queries that helped the reader assess the plausibility of each item in that contrast set.

Note that much of the current predictive modeling research wouldn’t qualify as a discussion miner because, in the usual case, most of that literature is still struggling with methods to

Prediction is all well and good,
but what about decision making?

W2 has two important features. First, it’s a landscape miner in that it maps out different regions of data inside of which we might build different models. Second, while the assembly of ideas is somewhat unique, each part of W2 is a known tool to the predictive modeling community. That is, it’s pos-

sible for the predictive community modeling community to refocus and redirect its tools toward an interesting new goal. W3 then applies a contrast set learner to the neighboring cluster to find best practices for achieving those better re-

Internals of a discussion miner.


Level	What	Task	Uses
0	Do	Predict, decide	Regression, classification, nearest neighbor reasoning
1	Say	Summarize, plan, describe	Instance section, feature selection, contrast sets
2	Reflect	Trade-offs, envelopes, diagnosis, monitoring	Clustering, multi-objective optimization, anomaly detectors
3	Share	Privacy, data compression, integrate old and new rules, recognize and debate deltas between competing models	Contrast set learning, transfer learning
4	Scale	Do all of the above, quickly	?

create one model, let alone updating a model as time progresses.

One fascinating open issue with discussion miners is how they should be assessed. In discussion mining, the model's goal is to find its own flaws and replace itself with something better, which brings to mind a quote from Susan Sontag: "The only good answers are the ones that destroy the questions." In other words, we shouldn't assess such models by accuracy, recall, or precision—rather, we should assess the audience engagement they engender. No, I don't know how to do that either, but I find it exciting that there are such clear and important problems waiting for us to solve tomorrow.

In terms of engineering principles, Table 1 shows the internals of a discussion miner. Note that the predictive

modeling community already has the parts needed to assemble this and other new kinds of miners.

We must move on, and we can. Enough already with algorithm mining: it's time to do other things. Industrial practitioners aren't really concerned with the internal details of our algorithms or how our data divides into regions. They're more concerned with the tools needed to help push the community debate different possible decisions. 

References

1. K. Dejaeger et al., "Data Mining Techniques for Software Effort Estimation: A Comparative Study," *IEEE Trans. Software Eng.*, vol. 28, no. 2, pp. 375–397.
2. T. Hall et al., "A Systematic Review of Fault Prediction Performance in Software Engineer-

ing," *IEEE Trans. Software Eng.*, vol. 38, no. 6, pp. 1276–1304.

3. A. Brady and T. Menzies, "Case-Based Reasoning vs. Parametric Models for Software Quality Optimization," *Proc. 6th Int'l Conf. Predictive Models in Software Eng.* (PROMISE 10), ACM, 2010; <http://doi.acm.org/10.1145/1868328.1868333>.
4. T. Menzies and T. Zimmermann, "Goldfish Bowl Panel: Software Development Analytics," *Proc. 2012 Int'l Conf. Software Eng.* (ICSE 2012), IEEE Press, 2012, pp. 1032–1033.
5. T. Menzies et al., "Local vs. Global Lessons for Defect Prediction and Effort Estimation," *IEEE Trans. Software Eng.*, preprint, published online Dec. 2012; <http://goo.gl/k6qno>.

TIM MENZIES is a full professor in computer science at the Lane Department of Computer Science and Electrical Engineering, West Virginia University. Contact him at tim@menzies.us.



Selected CS articles and columns are also available for free at <http://ComputingNow.computer.org>.

IEEE Software (ISSN 0740-7459) is published bimonthly by the IEEE Computer Society. IEEE headquarters: Three Park Ave., 17th Floor, New York, NY 10016-5997. IEEE Computer Society Publications Office: 10662 Los Vaqueros Cir., Los Alamitos, CA 90720; +1 714 821 8380; fax +1 714 821 4010. IEEE Computer Society headquarters: 2001 L St., Ste. 700, Washington, DC 20036. Subscription rates: IEEE Computer Society members get the lowest rate of US\$56 per year, which includes printed issues plus online access to all issues published since 1984. Go to www.computer.org/subscribe to order and for more information on other subscription prices. Back issues: \$20 for members, \$209.17 for nonmembers (plus shipping and handling).

Postmaster: Send undelivered copies and address changes to *IEEE Software*, Membership Processing Dept., IEEE Service Center, 445 Hoes Lane, Piscataway, NJ 08854-4141. Periodicals Postage Paid at New York, NY, and at additional mailing offices. Canadian GST #125634188. Canada Post Publications Mail Agreement Number 40013885. Return undeliverable Canadian addresses to PO Box 122, Niagara Falls, ON L2E 6S8, Canada. Printed in the USA.

Reuse Rights and Reprint Permissions: Educational or personal use of this material is permitted without fee, provided such use: 1) is not made for profit; 2) includes this notice and a full citation to the original work on the first page of

the copy; and 3) does not imply IEEE endorsement of any third-party products or services. Authors and their companies are permitted to post the accepted version of IEEE-copyrighted material on their own web servers without permission, provided that the IEEE copyright notice and a full citation to the original work appear on the first screen of the posted copy. An accepted manuscript is a version which has been revised by the author to incorporate review suggestions, but not the published version with copyediting, proofreading, and formatting added by IEEE. For more information, please go to: http://www.ieee.org/publications_standards/publications/rights/paperversionpolicy.html. Permission to reprint/republish this material for commercial, advertising, or promotional purposes or for creating new collective works for resale or redistribution must be obtained from IEEE by writing to the IEEE Intellectual Property Rights Office, 445 Hoes Lane, Piscataway, NJ 08854-4141 or pubs-permissions@ieee.org. Copyright © 2013 IEEE. All rights reserved.

Abstracting and Library Use: Abstracting is permitted with credit to the source. Libraries are permitted to photocopy for private use of patrons, provided the per-copy fee indicated in the code at the bottom of the first page is paid through the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923.