Data Mining In Design and Test Processes - Basic Principles and Promises *

Li-C. Wang
University of California, Santa Barbara

ABSTRACT

This talk discusses several application examples to illustrate the basic principles of applying data mining in design and test. Two types of data mining are seen in most of the applications: novelty detection and feature-based rule learning. The experience of developing a practical data mining flow is summarized. Promises are demonstrated with positive experimental results based on industrial settings.

Categories and Subject Descriptors

B.7 [**Integrated Circuits**]: Miscellaneous; H.2.8 [**Database Management**]: Database Applications—*Data mining*

General Terms

Design

Keywords

Computer-Aided Design, Data Mining, Test, Verification

1. INTRODUCTION

Data mining in design and test has become a fast-growing research area in recent years. In design and test, tremendous amounts of simulation and measurement data are generated. These data present opportunities for applying data mining.

In formulating an application, it is important to ask the question: Mining for what purpose? Many problems encountered in design automation and test are NP-hard problems. Data mining does not make a NP-hard problem easier. In fact, the power of learning is so limited that even learning a simple 3-Term DNF formulae is NP-hard [1].

Then, what is the power of data mining for, if not for solving a difficult problem? Learning a Boolean function with an almost perfect accuracy is hard [1]. However, learning with a high percentage of accuracy is feasible [2]. In other words, if one demands an almost guaranteed result, the problem becomes hard. Therefore, a practical data mining application should be formulated in such a way that guaranteed result is not required for the application to be meaningful.

An application can only be formulated based on the data availability. If the data is not readily available, the cost of collecting the data will become an important consideration. One has to demonstrate that the benefit of mining the data out-weights the cost in order for the application to make practical sense. Furthermore, the size of the data has to be large enough for data mining to make sense.

Data mining is most useful when it is applied to improve the efficiency of an existing process. For example, a person needs to examine a large number of plots to identify a few interesting ones for further analysis. It is a tedious manual process. Data mining can be used to speed up the process by automatically identifying the interesting plots.

Introduction of a data mining flow should make a target task easier for its user, not harder. Hence, design of an effective usage model is crucial. This includes effective presentation of the mining results with easy visualization to facilitate user interaction and decision making.

2. EXAMPLES AND PROMISES

Data mining algorithms operate on a collection of *samples*. Given N samples, novelty detection intends to find a few samples that are "novel" with respect to others. A popular algorithm to implement novelty detection is the SVM one-class [3]. In feature-based rule learning, two classes of samples are given, a large number of known non-novel samples and a small number of known novel samples. The objective is to uncover special properties of novel samples and represent them as rules. A popular algorithm to implement rule learning is the subgroup discovery [4].

In pre-silicon design, functional verification remains a key bottleneck. In a design cycle, the design evolves over time. Consequently, functional verification is an iterative process in which extensive simulation is run on a few relatively stable versions of the design. From one iteration to another, two assets are kept. The first comprises the test templates refined and accumulated up to the previous iteration. The second comprises the important tests identified so far.

In this context, data mining can be employed in two applications, to reduce the simulation time required to find an important test and to improve a test template for generating additional important tests. Here, a sample is a test.

The work in [5] proposes a novel test detection framework that can filter out a large number of unimportant tests before simulation, effectively reducing the simulation time by up to 90%. The data mining approach is novelty detection. The work in [6] presents a feature-based analysis approach to extract special properties of novel tests. These properties are then used to improve the test template for achieving a better coverage. The data mining approach is feature-based rule learning [7]. In both works, a test is an assembly

*This work is supported in part by Semiconductor Research Corporation, projects 2010-TJ-2093 and 2012-TJ-2268

ISPD'13, March 24–27, 2013, Stateline, Nevada, USA.
ACM 978-1-4503-1954-6/13/03.

program. The experiments were run on a low-power 64-bit Power Architecture-based processor core.

Lithography simulation is another example in pre-silicon design where simulation time can be a major concern. In this context data mining can be used to reduce the need of lithography simulation. The objective is to identify potential problematic layout spots that are to be checked with lithography simulation. Majority of the layout areas do not need lithography simulation and hence simulation time is reduced. The work in [8] proposes two approaches for such a purpose, the *supervised* approach where two classes of training samples are first labeled by a lithograph simulation and then used to learn a classifier, and the *unsupervised* approach where potential problematic samples are identified as novel samples by novelty detection. Here, a sample is a small piece of layout image based on a raster scan of a layout [8].

For processor design, one important task in post-silicon is to identify speed limiting paths as guides for performance improvement. Data mining can be applied in two applications, facilitating the identification of potential speed paths [9][10] and understanding known speed paths [11]-[14]. In these applications, a sample is a path. The work in [14] summarizes the speed path analysis research in an industrial application. Design issues were uncovered by analyzing top speed paths against a large number of non-speed paths, which otherwise were difficult to find without the proposed feature-based data mining approach [14].

In production, test cost and/or quality continue to be major concerns. In die/chip-level analysis, a sample is a die/chip. In wafer-level analysis, a sample is a wafer. The work in [15] discusses how to predict potential defective parts as novel samples. Because novelty depends on the tests used in the analysis, the work [16] discusses the test selection problem. Both works were based on test data from an SoC production line for the automotive market where quality requirement is extremely high. Higher quality usually demands more sophisticated test processes and hence, higher cost. One expensive test process is burn-in. The work in [17] discusses the potential of burn-in reduction for another product line also for the automotive market, by predicting parts that do not need long hours of burn-in (or equivalently (novel) samples that need the burn-in). Data mining is applied on test data collected after a short period of burn-in to predict the outcomes after a long period of burn-in.

3. SUMMARY

In the above applications, data mining begins by defining what a sample is. For example, a sample can be an assembly program, a layout image, a path, a die/chip or a wafer. To apply a data mining algorithm, a sample needs to be encoded with a set of features. The effectiveness of the mining largely depends on this encoding. For example, the feature set used to describe the characteristics of an assembly programs or a path can take weeks to develop. Fortunately this development effort can be seen as one-time cost. In die/chip-level analysis, features are tests. Hence, test selection becomes an important problem [16].

In all applications, we are interested in finding and/or understanding a small set of novel samples in a large population of samples. While one can formulate the problem as binary classification if examples of novel samples are provided, it is usually not effective due to the extreme imbalance between the size of novel sample set and the size of non-novel sample set, i.e. the data usually contains a lot more information on the non-novel samples while our interest is on the novel samples. To overcome this issue, the feature set needs to incorporate some domain knowledge such that the analysis can be directed to a more refined space of interest. User intervention may also be needed to guide the selection of relevant features and the mining process. Data mining is rarely a one-time task. It is often seen as an iteratively knowledge discovery process where results are interpreted and actions are taken by user from one iteration to the next.

Modern learning algorithms such as SVM are designed with guaranteed asymptotic behavior when the data size approaches infinity [18]. Its effectiveness on a sample size of thousands or tens of thousands may vary, depending on the accuracy requirement for the application. With limited data, the quality of the feature set becomes more important. Hence, in many applications it is expected that a significant part of the effort will be spent on feature set development.

4. REFERENCES

[1] Michael J. Kearns and Umesh V. Vazirani. An Introduction to Computational Learning Theory, *MIT Press*, 1994.

[2] Onur Guzey, et. al. Extracting a Simplified View of Design Functionality Based on Vector Simulation. Lecture Note in Computer Science, *LNCS*, Vol 4383, 2007, pp. 34-49.

[3] Bernhard Schölkopf, and Alexander J. Smola. Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. The MIT Press, 2001.

[4] N. Lavrač, B. Kavšek, P. Flach, and L. Todorovski. Rule induction for subgroup discovery with CN2-SD *Journal of Machine Learning Research*, 5:153–188, Dec. 2004.

[5] Wen Chen, et. al. Novel Test Detection to Improve Simulation Efficiency A Commercial Experiment. *ACM/IEEE ICCAD*, 2012.

[6] Wen Chen, et. al. Novel Test Analysis to Improve Structural Coverage A Commercial Experiment. *IEEE VLSI Design Automation and Test Symposium*, 2013.

[7] Li-C. Wang, Data Learning Based Diagnosis. *ASP-DAC*, 2010, pp 247-254.

[8] Dragoljub (Gagi) Drmanac, Frank Liu, Li-C. Wang. Predicting Variability in Nanoscale Lithography Processes. *ACM/IEEE DAC*, 2009, pp. 545-550.

[9] Pouria Bastani, et. al. Speedpath Prediction Based on Learning from a Small Set of Examples. *ACM/IEEE DAC*, June 2008, pp. 217-222.

[10] Nicholas Callegari, Li-C. Wang, Pouria Bastani. Feature based similarity search with application to speedpath analysis. *IEEE ITC*, 2009.

[11] Li-C. Wang, Pouria Bastani, Magdy S. Abadir. Design-silicon timing correlation — a data mining perspective. In *ACM/IEEE DAC* 2007, pp. 384-389.

[12] P. Bastani, et. al. Statistical Diagnosis of Unmodeled Timing Effect. *ACM/IEEE DAC*, 2008, pp. 355-360.

[13] Nicholas Callegari, et. al. Classification rule learning using subgroup discovery of cross-domain attributes responsible for design-silicon mismatch. *ACM/IEEE DAC*, 2010, pp. 374-379.

[14] Janine Chen, et. al. Mining AC Delay Measurements for Understanding Speed-limiting Paths. *IEEE ITC*, 2010.

[15] Nik Sumikawa, et. al. Screening Customer ReturnsWith Multivariate Test Analysis. *IEEE ITC*, 2012.

[16] Nik Sumikawa, et. al. Important Test Selection For Screening Potential Customer Returns. *IEEE VLSI Design Automation and Test Symposium*, 2011, pp. 171-174.

[17] Nik Sumikawa, Li-C.Wang, and Magdy S. Abadir. An Experiment of Burn-In Time Reduction Based On Parametric Test Analysis. *IEEE ITC*, 2012.

[18] V. Vapnik, The nature of Statistical Learning Theory. 2nd ed., *Springer*, 1999.