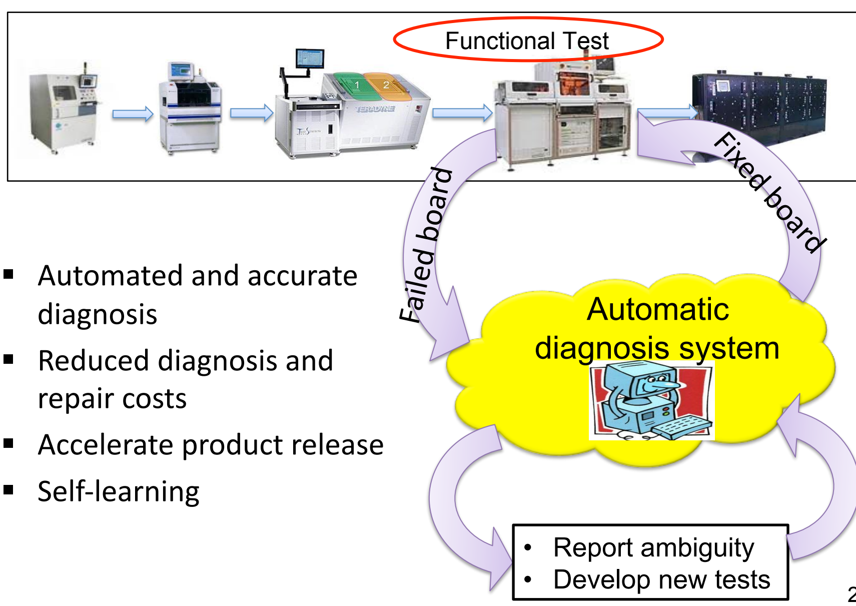
Board-Level Functional Fault Diagnosis Using Learning Based on Incremental Support-Vector Machines

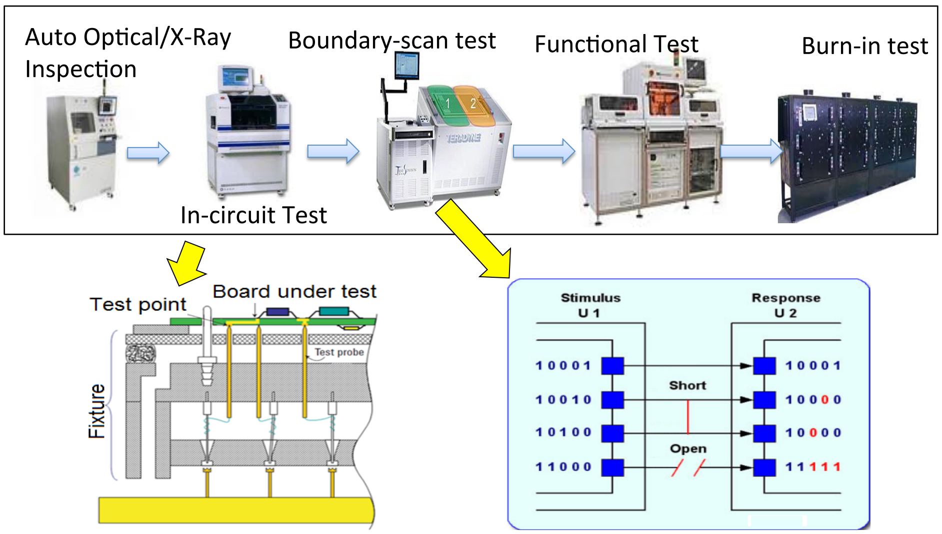
Motivation



- Automated and accurate diagnosis
- Reduced diagnosis and repair costs
- Accelerate product release
- Self-learning

2

Board-level Test Techniques



Background | Prior work | Diagnosis framework | iSVM | Experimental results | Conclusions

3

Case-based Learning

- Bypass bottleneck of rule-based learning
  - Difficult to acquire knowledge needed to build rules
- Bypass bottleneck of model-based learning
  - Difficult to construct model for complex system
- Ease of implementation
- Diagnostic accuracy improves with continuous learning

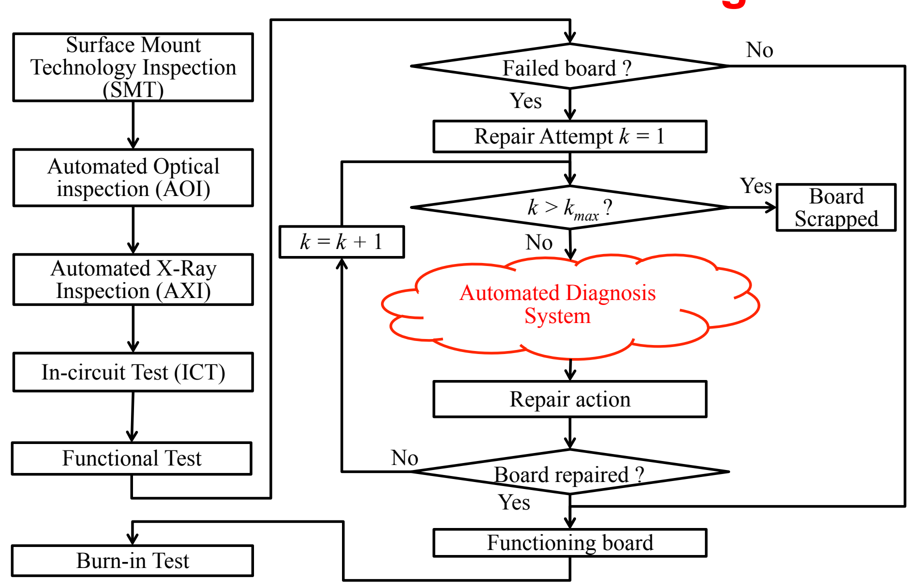
Background | Prior work | Diagnosis framework | iSVM | Experimental results | Conclusions

4

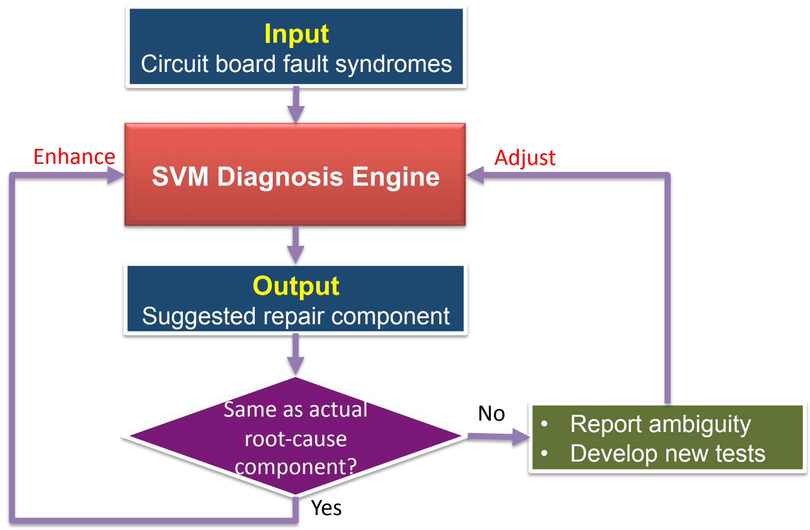
Learning For Board-level Functional Diagnosis

- Bayesian inference [Zhang VTS'10]
- Artificial neural networks [Zhang ITC'11]
- Support vector machines [Zhang ETS'12]
- Decision trees [Ye ATS'12]

Flowchart for Automated Diagnosis



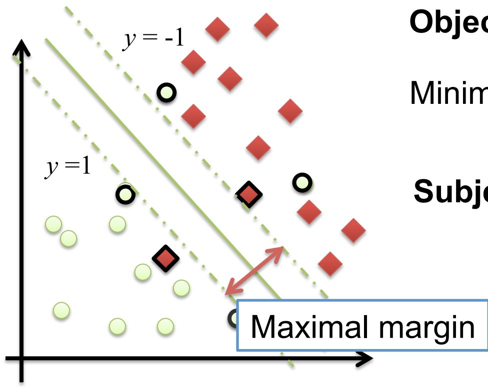
Diagnosis System Update Mechanism



Incremental SVMs

- Key ideas:
  - Dynamic learning, diagnosis system updates
  - Rely on SVMs
- Advantage:
  - Reduce training/computation time
  - Appropriate for online diagnosis in manufacturing line
  - Scalable for diagnosis during high-volume production

Support Vector Machines (SVMs)



Objective function:

Minimize $W = \frac{1}{2}\|w\|^2 + C\sum_{i}^{L} \xi_i$

Subject to:

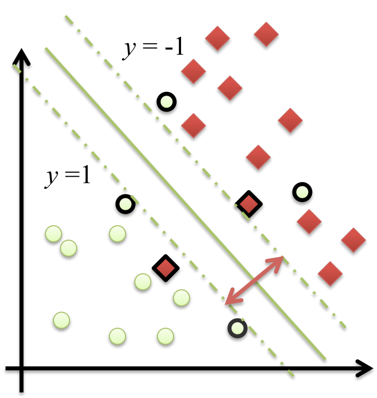
- $\|y_i(w \cdot x_i + b)\| \leq 1 - \xi_i, \quad \forall i$
- $\xi_i > 0$

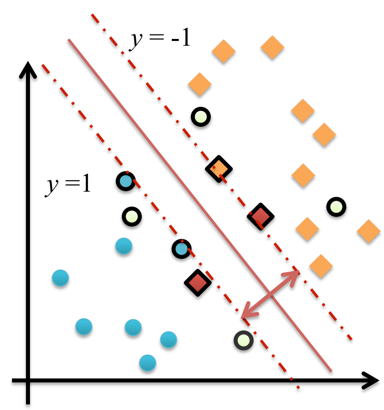
Elements in Class 1
Elements in Class 2
Support Vectors
Optimal classifier

Background | Prior work | Diagnosis framework | iSVM | Experimental results | Conclusions

9

Incremental SVMs





Object function $W' = \frac{1}{2}\|w'\|^2 + C\sum_{i}^{l} (L \cdot \xi_i + \xi_i')$

Elements in Class 1
Support Vectors
Elements in Class 2
Optimal classifier

10

Fault Syndromes And Repair Actions

- A segment of the log file of traffic test

```
## Summary: Interfaces<   r2d2 -- metro   > counts - Fail(mismatch)
......464. (00000247) ERR    EG    R2D2_ARIC_CP_DBUS_CRC_ERR
......
Error: (0000010A) DIAGERR_ERRISO_INVALID_PKT_CNT: Packet count invalid
```

- Syndromes are parsed in multiple dimensions
  - Error ID; mismatched interface; drop counter; component with interrupts; interrupt bits, etc.
  - E.g. Error ID: Mismatched interface: r2d2 – metro, etc.
- Actions are replaced components, e.g. U37

Syndromes →
$$\left[\begin{array}{ccc|c} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & -1 \\ 0 & 0 & 1 & -1 \end{array}\right]$$
← Action

Example of iSVMs

Initial training set

$$\mathcal{A} = [\mathcal{B}|\mathcal{C}] = \left[\begin{array}{ccc|c} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & -1 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 1 & -1 \end{array}\right]$$

Support vector extraction

Extracted support vectors

$$\mathcal{A}' = \left[\begin{array}{ccc|c} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & -1 \\ 0 & 0 & 1 & -1 \end{array}\right]$$

+

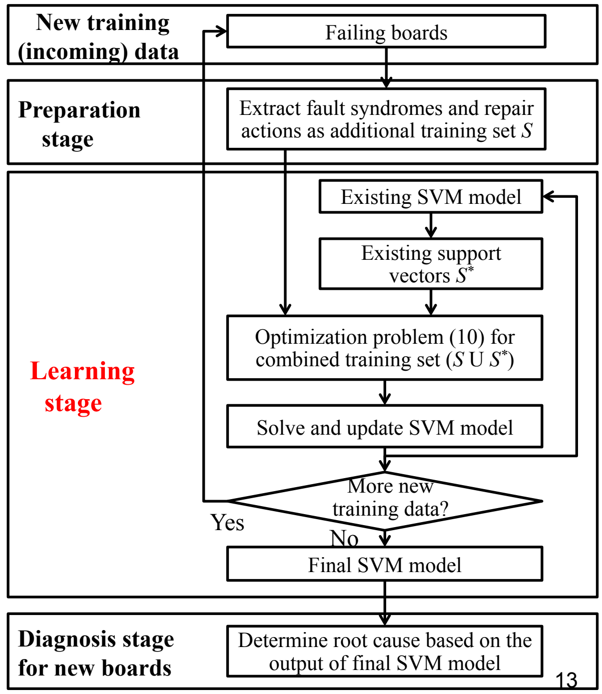
New training cases

$$\mathcal{N} = \left[\begin{array}{ccc|c} 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & 0 & 1 & -1 \end{array}\right]$$

⇒ Combined Training Set

$$\mathcal{D} = \left[\frac{\mathcal{A}'}{\mathcal{N}}\right] = \left[\begin{array}{ccc|c} 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & -1 \\ 0 & 0 & 1 & -1 \\ \hline 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & 0 & 1 & -1 \end{array}\right]$$

Incremental Learning Flowchart

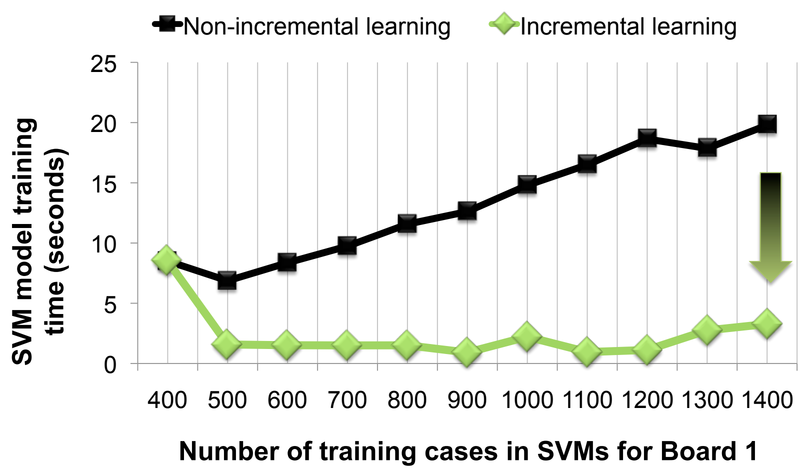


Experiments

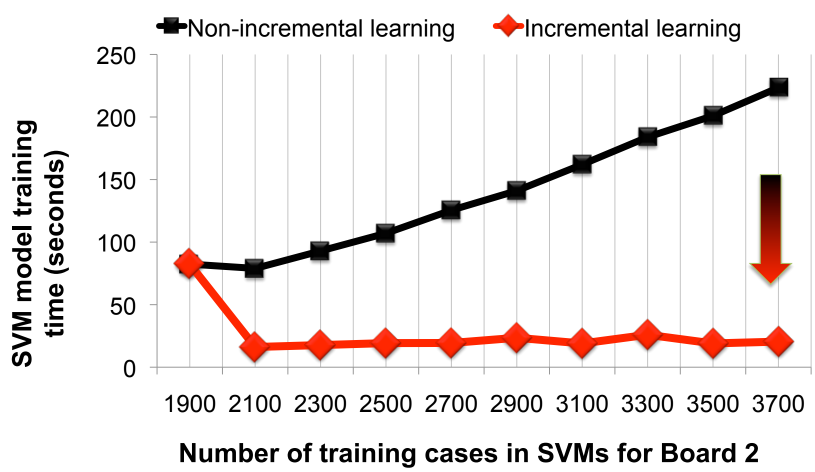
- Experiments performed on boards currently in production
  - Tens of ASICs, hundreds of passive components
- All the boards under analysis failed traffic test
  - A comprehensive functional test set for fault isolation, run through all components

	Board 1	Board 2
Number of syndromes	207	420
Number of root-cause component	14	37
Number of failed boards	1400	3700

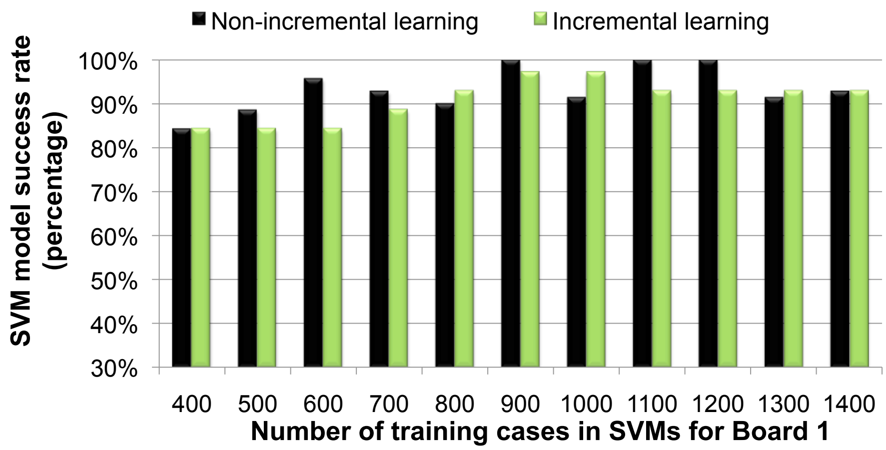
Comparison Of Training Time



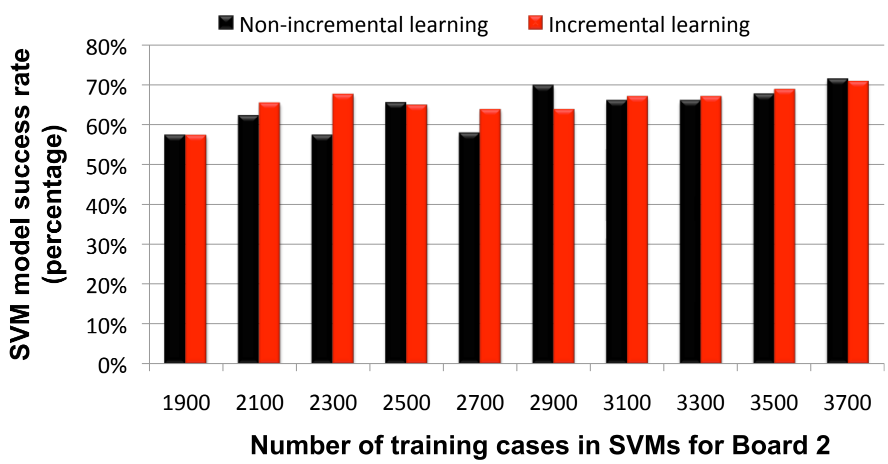
Comparison Of Training Time



Comparison of Success Rates



Comparison of Success Rates



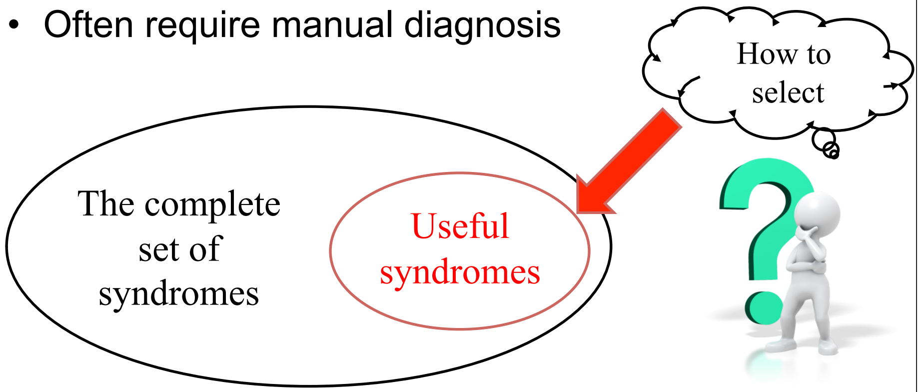
Conclusions

- Manufacturing test and fault diagnosis affect product quality, time-to-market, yield, and cost
- Proposed diagnose system based on incremental SVMs can achieve high diagnosis accuracy
- Reduced diagnose-system update time
  - Scalable to production in high volume

Adaptive Board-Level Functional Fault Diagnosis Using Decision Trees

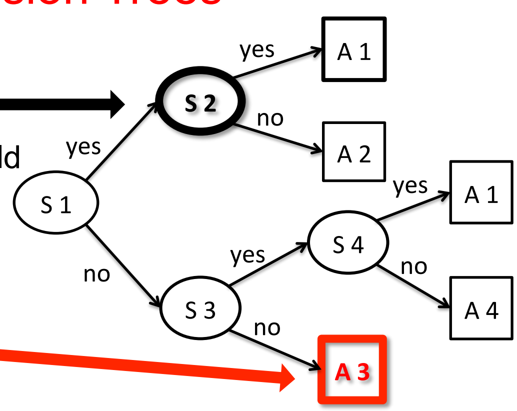
Current Diagnosis System

- Number of syndromes (up to 1,000 per board)
- Diagnosis time (up to several hours per board)
- Often require manual diagnosis



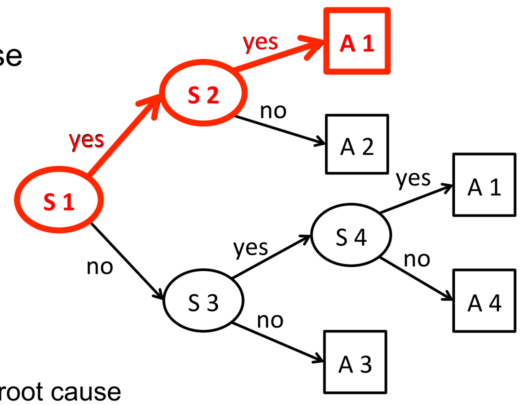
Decision Trees

- Internal Nodes
  - Can branch to two child nodes
  - Represent syndromes
- Terminal Nodes
  - Do not branch
  - Contain class information



Decision Trees

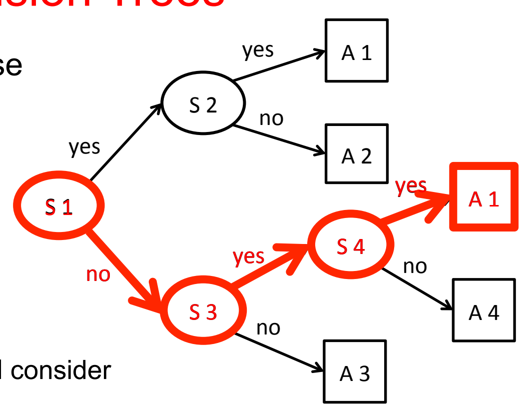
- We may reach root cause A1 in two different test sequences.



1) Start from the most discriminative syndrome S1
2) If S1 manifests itself, we then consider syndrome S2
3) If S2 manifests itself, we can determine A1 to be the root cause

Decision Trees

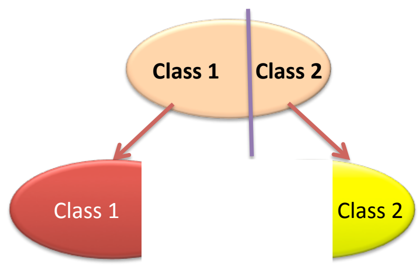
- We may reach root cause A1 in two different test sequences.



1) Start from the most discriminative syndrome S1
2) If S1 pass, we will consider syndrome S3
3) If S3 manifests itself, we will consider syndrome S4
4) If S4 manifests itself, then we can determine A1 to be the root cause

Training Of Decision Trees (Syndrome Identification)

- Goals:
  - Rank syndromes
  - Minimize ambiguity
  - Reduce tree depth
- Three popular criteria can be used for training decision trees
  - Information Gain
  - Gini Index
  - Twoing



Information Gain

Symbol	Description
C	A set of training cases (failed boards)
A	A set of root cause component $\{A_1, A_2 .... A_j\}$
S	A set of syndromes $\{S_1, S_2 .... S_m\}$

$$\boldsymbol{IG(C, S_i) = E(C) - E(C|S_i)}$$

- $E(C)$: entropy of C
- $E(C|S_i)$: entropy of C given a syndrome S_i
- $p(A_j)$: probability of class A_j in C
- s_i: event that S_i manifest itself; $\overline{s_i}$ otherwise

Information Gain (Example)

S_1	S_2	S_3	Root Cause
1	1	0	A_1
1	1	1	A_1
1	1	0	A_1
1	0	1	A_2
0	0	1	A_2

- First calculate the entropy of $\boldsymbol{C}$

$$E(C) = E(3{:}2)$$

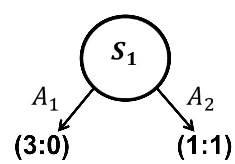
$$= -\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5}$$

$$= 0.673$$

Information Gain (Example)

S_1	S_2	S_3	Root Cause
1	1	0	A_1
1	1	1	A_1
1	1	0	A_1
1	0	1	A_2
0	0	1	A_2

- Consider S_1



$$E(A_1|S_1) = E(3{:}0)$$

$$= -\frac{3}{3}\log_2\frac{3}{3} - \frac{0}{3}\log_2\frac{0}{3} = 0$$

$$E(A_2|S_1) = E(1{:}1)$$

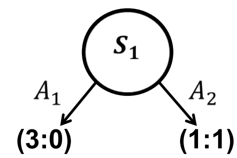
$$= -\frac{1}{2}\log_2\frac{1}{2} - \frac{1}{2}\log_2\frac{1}{2}$$

$$= 0.69$$

Information Gain (Example)

S_1	S_2	S_3	Root Cause
1	1	0	A_1
1	1	1	A_1
1	1	0	A_1
1	0	1	A_2
0	0	1	A_2

- Consider S_1



$E(A_1|S_1) = 0$
$E(A_2|S_1) = 0.69$

$$E(C|S_1) = 0 \times \frac{3}{5} + 0.693 \times \frac{2}{5}$$
$$= 0.277$$
$$IG(C, S_1) = E(C) - E(C|S_1)$$
$$= 0.396$$

Information Gain (Example)

S_1	S_2	S_3	Root Cause
1	1	0	A_1
1	1	1	A_1
1	1	0	A_1
1	0	1	A_2
0	0	1	A_2

- Also consider S_2, S_3

  $IG(C, S_1) = 0.396$
  $IG(C, S_2) = 0.673$
  $IG(C, S_3) = 0.298$

- Since S_2 has the highest information gain, we choose S_2 to be the most discriminative syndrome

Gini Index

$$GI(C, S_i) = Gini(C|S_i) - Gini(C)$$

- $Gini(C)$ is the Gini index of $\boldsymbol{C}$
- $Gini(C|S_i)$ is the Gini index of $\boldsymbol{C}$ given syndrome S_i

Gini Index (Example)

S_1	S_2	S_3	Root Cause
1	1	0	A_1
1	1	1	A_1
1	1	0	A_1
1	0	1	A_2
0	0	1	A_2

- Consider Gini index of $\boldsymbol{C}$

$$Gini(C) = E(3:2)$$

$$= \frac{3}{5}\left(1 - \frac{3}{5}\right) + \frac{2}{5}\left(1 - \frac{2}{5}\right)$$

$$= 0.48$$

Gini Index (Example)

S_1	S_2	S_3	Root Cause
1	1	0	A_1
1	1	1	A_1
1	1	0	A_1
1	0	1	A_2
0	0	1	A_2

- Consider Gini Index of S_1, S_2, and S_3

$GI(C, S_1) = -0.28$
$GI(C, S_2) = -0.48$
$GI(C, S_3) = -0.21$

Diagnosis Using Decision Trees

Training Data Preparation
Extract all the fault syndromes and the repair actions from historical data

⇩

DT Architecture Design
Design inputs, outputs, splitting criterion, pruning

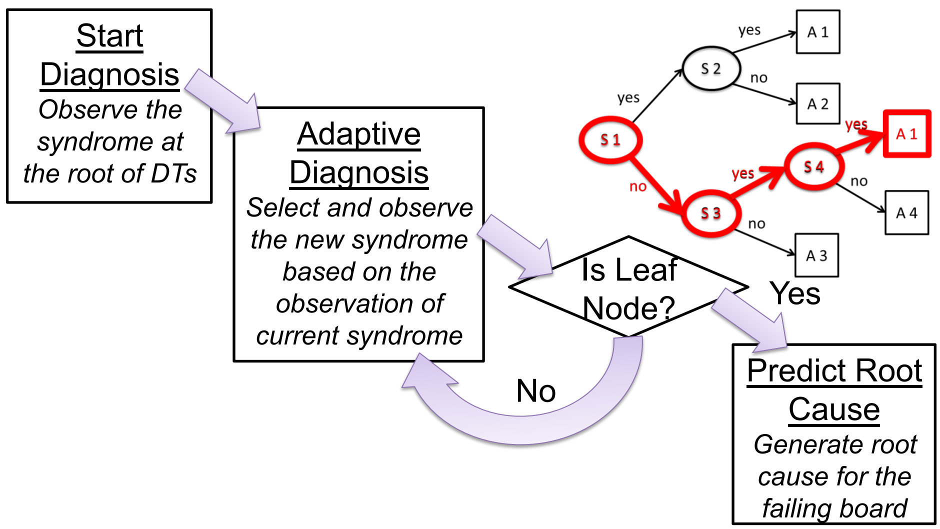
⇩

DT Training
Generate a tree-based predictive model and assess the performance

⇩

DT-based Diagnosis
Traverse from the root node of DTs and obtain the root cause at the leaf node

Diagnosis Using Decision Trees



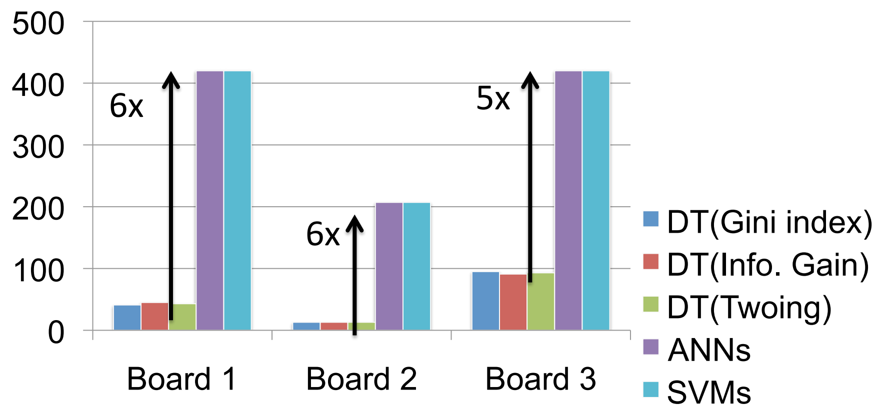
Experiments

- Experiments performed on industrial boards currently in production
  - Tens of ASICs, hundreds of passive components
- All the boards under analysis failed traffic test
  - A comprehensive functional test set for fault isolation, run through all components

	Board 1	Board 2	Board 3
Number of test items	420	207	420
Number of root cause components	10	14	10
Number of failed boards	130	40	1000

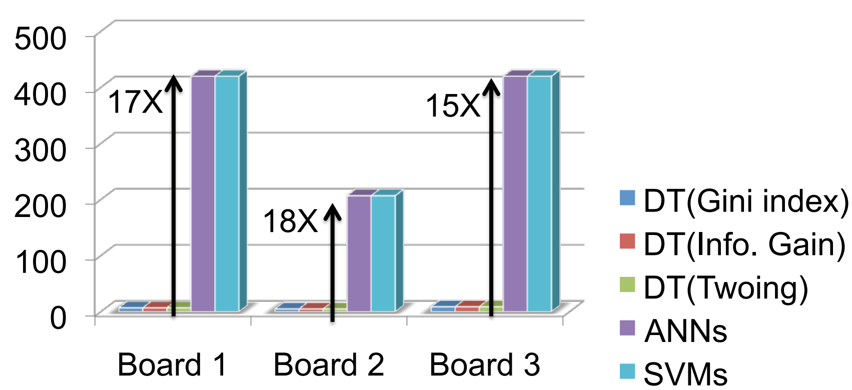
Comparison Of Different Decision-Tree Architectures

Total number of syndromes used for diagnosis



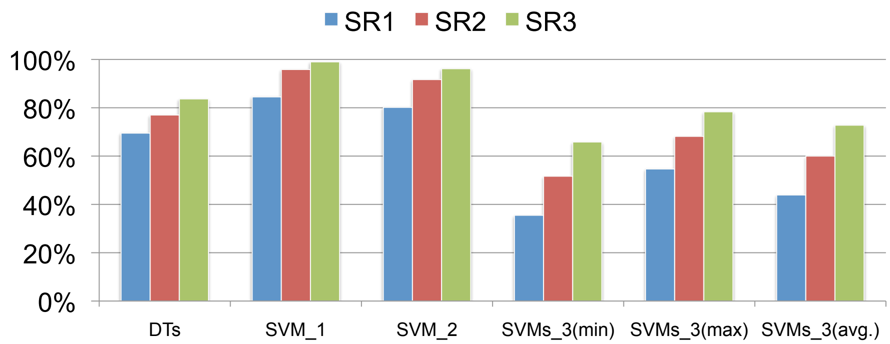
Comparison Of Different Decision-Tree Architectures

Average Number of syndromes used for diagnosis



Comparison Between DTs And SVMs

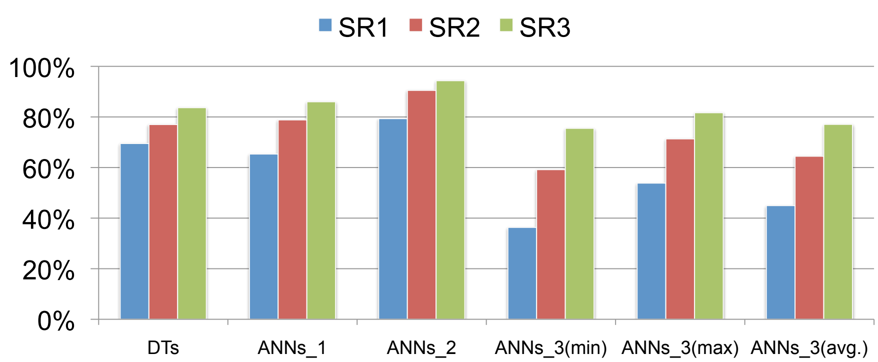
- Success rates (SR) obtained for Board 3



- SR obtained by DTs are similar to SR obtained by SVMs

Comparison Between DTs And ANNs

- Success rates (SR) obtained for Board 3



- SR obtained by DTs are similar to SR obtained by ANNs

Conclusions

- Decision tree simplifies board diagnosis
  - Simple structure, less time for training and on-line diagnosis
  - Bypass “test item” bottleneck of existing methods
- Reduced number of syndromes for industry boards
  - A total of 92 test items (syndromes) in DT diagnosis compared to a total of 420 test items in ANNs/SVMs diagnosis
- Different architectures available based on information theory measures
- High success rates
  - Similar success rates obtained using DTs compared to success rates obtained with ANNs/SVMs
- Scalable to diagnosis for production in volume