三篇论文

SCIENCE AND TECHNOLOGY

《The Missing Links: Bugs and Bug-fix Commits》	2010	FSE/ESEC
《Is better data better than better data miners?: on the benefits of tuning smote for defect prediction》	2018	ICSE

GRADUATION DEFENSE

The Missing Links: Bugs and Bug-fix Commits

2010 FSE/ESEC

Given the wide use of linked defect data, it is vital to gauge the nature and extent of the bias, and try to develop testable theories and models of the bias.

Link

Bug Reports ⟷ Commit Logs

Issues

Bug-feature Bias

where only the fixes of certain types of defects are linked

Commit-feature Bias

where only the certain kinds of fixes, or fixes to certain kinds of files, are linked.

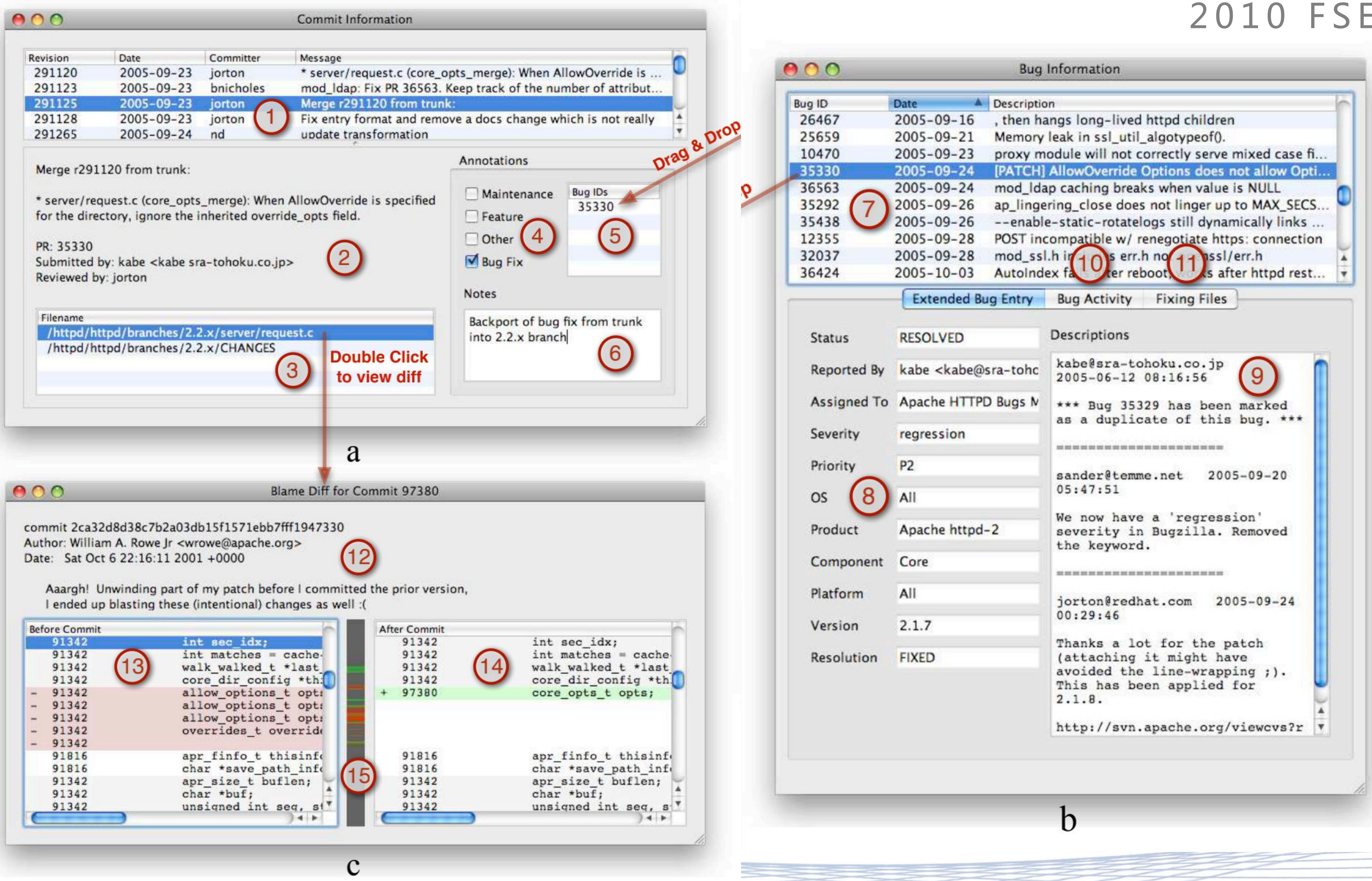
Contribution

1. Present ***Linkster***
   A tool to facilitate link reverse-engineering.
2. Evaluate this tool
3. Analyze the comprehensive data set
   Apache HTTP web server project

The Missing Links: Bugs and Bug-fix Commits

2010 FSE/ESEC

Linkster



The Missing Links: Bugs and Bug-fix Commits

2010 FSE/ESEC

- RQ 1: *Do the bug reporting and fixing practices of developers correspond to the assumptions commonly made by researchers?*

A so-called "bug" is not always a bug; neither is a "commit" always a commit.

- RQ 2: *How well does the automated approach of finding links between commits and bug reports work?*

The automated approach finds virtually all the commit log messages which contain a link to the bug tracking database

- RQ 3: *Is there any evidence of systematic bias in the linking of bug-fix commits to bug reports?*

Find that reporting bias affects the performance of a bug prediction algorithm .

The Missing Links: Bugs and Bug-fix Commits

2010 FSE/ESEC

Finding 1. Not all fixed bugs are mentioned in the bug tracking database. Some are discussed (only) on the mailing list.

Finding 2. To fix a bug in an Apache release, multiple similar commits by different developers are needed.

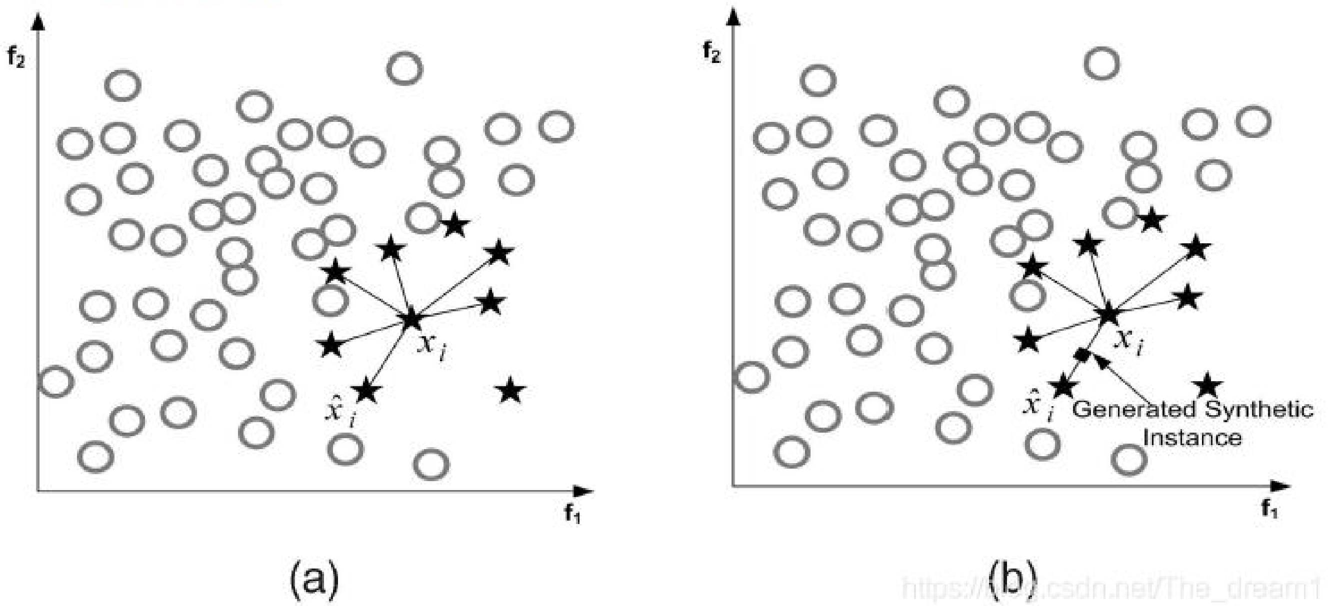
Finding 3. Developers sometimes fix bugs that are only reported in some other projects' bug tracker, rather than in their own; and vice-versa.

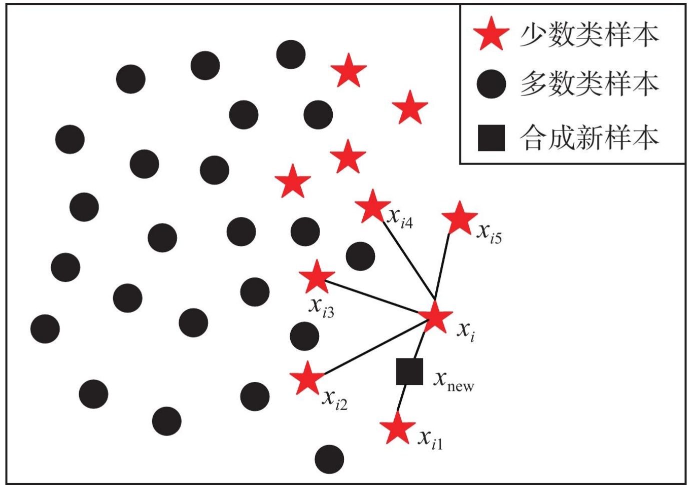
Finding 4. Even if we annotate all commits, the cause of a commit still remains unspecified in some cases.

Is better data better than better data miners?: on the benefits of tuning smote for defect prediction

2018 ICSE

Smote






```
def SMOTE(k=2, m=50%, r=2): # defaults
  while Majority > m do
    delete any majority item # random
  while Minority < m do
    add something_like(any minority item)

def something_like(X0):
  relevant = emptySet
  k1 = 0
  while(k1++ < 20 and size(found) < k) {
     all = k1 nearest neighbors
     relevant += items in "all" of X0 class}
  Z = any of found
  Y = interpolate (X0, Z)
  return Y

def minkowski_distance(a,b,r):
  return (Σ_i abs(a_i − b_i)^r)^(1/r)
```

Figure 3: Pseudocode of SMOTE

Is better data better than better data miners?: on the benefits of tuning smote for defect prediction

2018 ICSE

Smotuned

```
def DE( n=10, cf=0.3, f=0.7): # default settings
  frontier = sets of guesses (n=10)
  best = frontier.1 # any value at all
  lives = 1
  while(lives−− > 0):
    tmp = empty
    for i = 1 to |frontier|: # size of frontier
      old = frontier_i
      x,y,z = any three from frontier, picked at random
      new= copy(old)
      for j = 1 to |new|: # for all attributes
        if rand() < cf # at probability cf...
          new.j = x.j + f ∗ (z.j − y.j) # ...change item j
      # end for
      new = new if better(new,old) else old
      tmp_i = new
      if better(new,best) then
        best = new
        lives++ # enable one more generation
      end
    # end for
   frontier = tmp
  # end while
  return best
```

Figure 4: SMOTUNED uses DE (differential evolution).

Table 5: SMOTE parameters

Para	**Defaults used by** SMOTE	**Tuning Range (Explored by** (SMOTUNED)	**Description**
k	5	[1,20]	Number of neighbors
m	50%	{50, 100, 200, 400}	Number of synthetic examples to create. Expressed as a percent of final training data.
r	2	[0.1,5]	Power parameter for the Minkowski distance metric.

Is better data better than better data miners?: on the benefits of tuning smote for defect prediction

2018 ICSE

- RQ1: Are the default "off-the-shelf" parameters for SMOTE appropriate for all datasets?
- RQ2: Is there any benefit in tuning the default parameters of SMOTE for each new dataset?
- RQ3: In terms of runtimes, is the cost of running SMOTUNED worth the performance improvement?
- RQ4: How does SMOTUNED perform against more recent class imbalance technique?

Is better data better than better data miners?: on the benefits of tuning smote for defect prediction

2018 ICSE

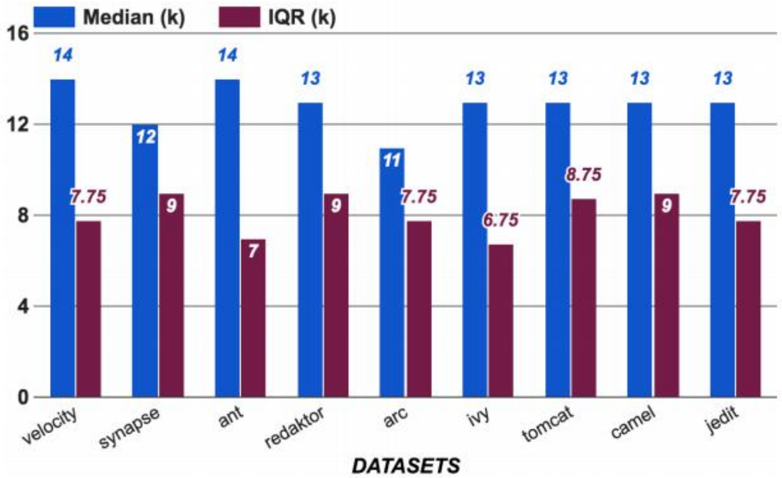


Figure 5a: Tuned values for k (default: $k = 5$).

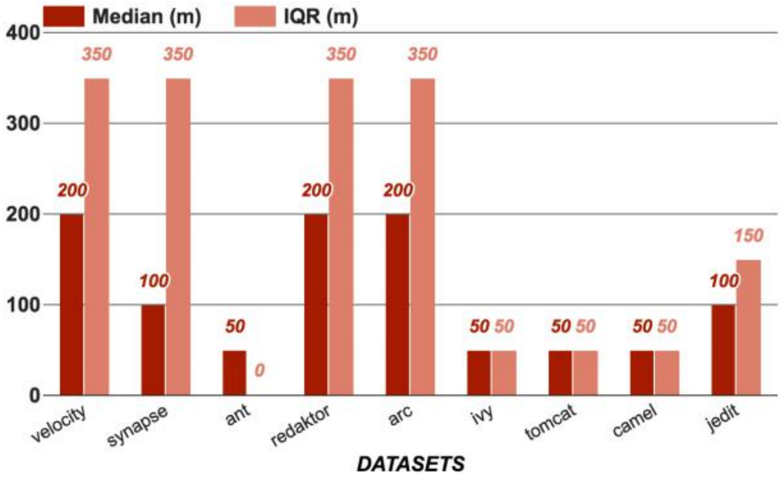


Figure 5b: Tuned values for m (default: $m = 50\%$).

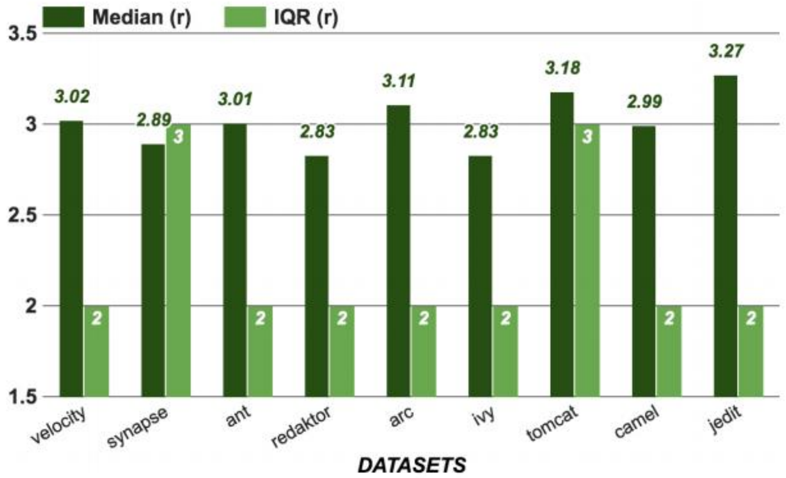


Figure 5c: Tuned values for r (default: $r = 2$).

Figure 5: Data sets vs Parameter Variation when optimized for recall and results reported on recall. "Median" denotes 50th percentile values seen in the 5*5 cross-validations and "IQR" shows the intra-quartile range, i.e., (75-25)th percentiles.

	number of wins			
Treatments	AUC	Recall	Precision	False Alarm
MAHAKIL	1/9	0/9	**6/9**	**9/9**
SMOTE	0/9	1/9	0/9	0/9
SMOTUNED	**8/9**	**8/9**	3/9	0/9

Is better data better than better data miners?: on the benefits of tuning smote for defect prediction

2018 ICSE

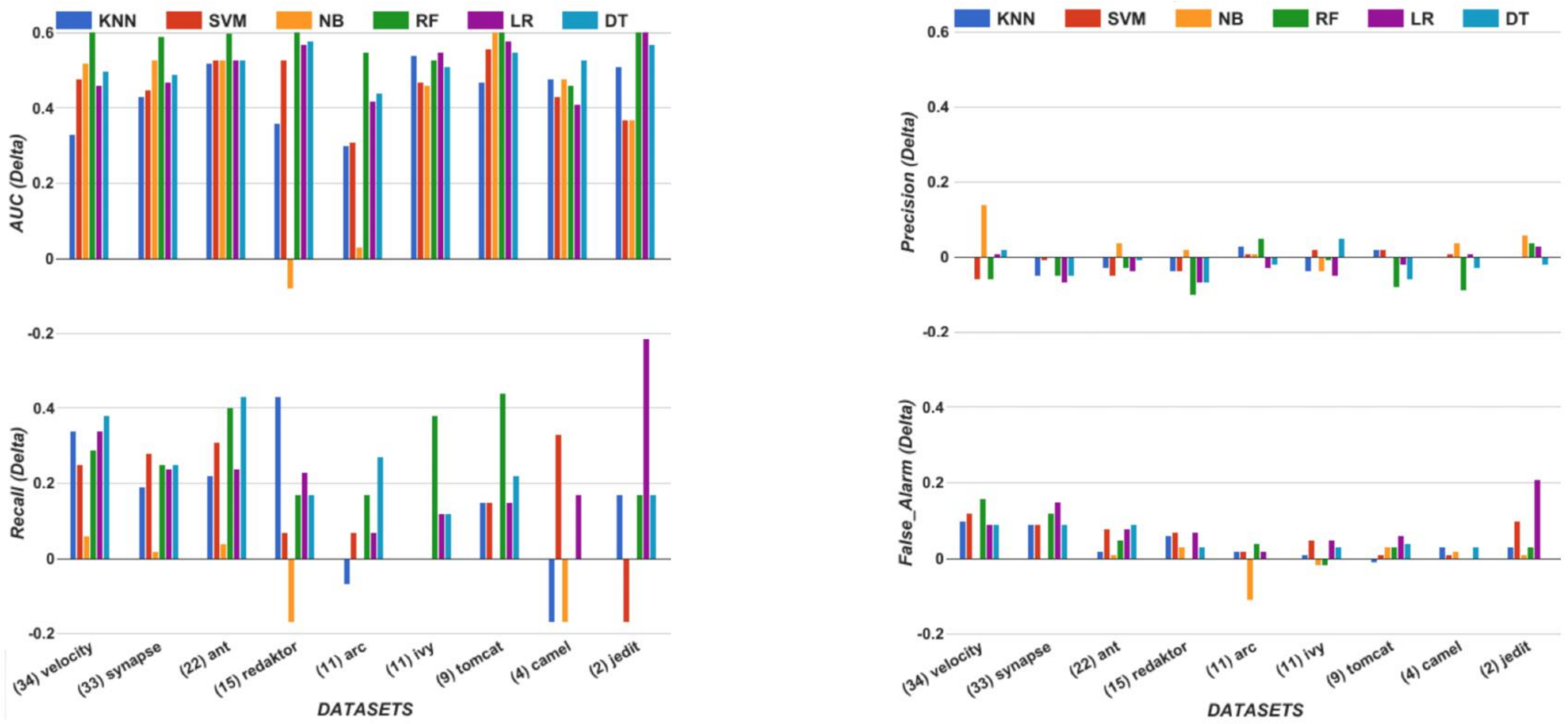


Figure 6: SMOTUNED **improvements over** SMOTE. **Within-Measure assessment (i.e., for each of these charts, optimize for performance measure** M_i**, then test for performance measure** M_i**). For most charts,** ***larger*** **values are** ***better*****, but for false alarm,** ***smaller*** **values are** ***better*****. Note that the corresponding percentage of minority class (in this case, defective class) is written beside each data set.**