Test Set Diameter: Quantifying the Diversity of Sets of Test Cases

2016 IEEE International Conference on Software Testing, Verification and Validation

Robert Feldt, Simon Pdoulding, David Clark, Shin Yoo

Introduction: Diversity of Test Cases

Idea
A small set of diverse test cases are considered to be most efficiently tests in testing because differing are more like cover software behaviors.

Problem
However, how to quantify diversity among test cases? This paper propose a test diversity metric based on information distance.

Kolmogorov Complexity

Example
Which of these is more complex or random?

- 111111111111
- 11001101110001

Intuition
- The first has a simple description:” print 1 16 times”. 
- There is no obvious description for the second string that is shorter than listing it digits.
- This intuition leads to definition of Kolmogorov Complexity.
**Definition of Kolmogorov Complexity**

The Kolmogorov Complexity of a string \( w \), denoted \( K(w) \) is the length of
the shortest program which outputs \( x \) given no input.

**Definition of conditional Kolmogorov Complexity**

The Kolmogorov Complexity of a string \( x \) given \( y \), denoted \( K(x|y) \) is the
length of the shortest program which outputs \( x \) given the input \( y \).

**Information Distance (ID)**

They calculate the similarity by inverse of diversity based on Kolmogorov
Complexity.

**Definition of ID**

For two strings, \( x \) and \( y \), the information distance is:
\[
ID(x, y) = \max \{ K(x|y), K(y|x) \}
\]
The similarity between \( x \) and \( y \) is the length of the shortest program that
converts \( x \) to \( y \), or \( y \) to \( x \).

**Normalized Information Distance (NID)**

**Purpose**

They want to compare similarities between pairs of strings across a range of
sizes.

**Problem**

However, the ID between two long strings larger than that between two short strings.

**NID**

Normalized information distance (NID) that normalizes the information
distance of two strings:
\[
NID(x, y) = \frac{\max(K(x|y), K(y|x))}{\max(K(x), K(y))}
\]
NID takes values in \([0,1]\), where 0 indicate greater similarity.

**Normalized Copression Distance (NCD)**

In practice, it hard to determine the shortest program that outputs a given string.

**NCD**

The output size of compression a string approximates its Kolmogorov
complexity.

**NCD definition**

\[
NCD(x, y) = \frac{C(xy) - \min\{C(x), C(y)\}}{\max\{C(x), C(y)\}}
\]
\( C(x) \) is the compressing size of string \( x \)
\( xy \) is the concatenation of \( x \) and \( y \). NCD takes values in \([0,1+\epsilon]\), \( \epsilon \) is a
positive value depends on what compression program we use.
NCD for Multisets

For a multiset $X$, the NCD is calculated via an intermediate measure:

$$NCD_1(x) = \frac{C(x) - \min_{x \in X} \{C(x)\}}{\max_{x \in X} \{C(x)\}} \quad (1)$$

$$NCD(x) = \max \{NCD_1(X), \max_{y \in X} \{NCD(Y)\}\}$$

$C(x)$ is the length of compressing the string. However, NCD requires time complexity $O(2^N)$.

Alternative algorithm with time complexity $O(N^2)$

Using $NCD_1$ defined above. Multiset $Y_0 = X = \{x_1, x_2, ..., x_n\}$

1. Find index $i$ that maximizes $C\{Y_k \{x_i\}\}$.
2. Let $Y_{k+1} = Y_k \{x_i\}$.
3. Repeat [1] until the subset contains only two strings.
4. Calculate NCD($X$) as: $\max_{0 \leq k \leq n-2} \{NCD_1(Y_K)\}$

TSDm is the diversity of a test set measured by the extension of NCD for multisets.

TSDm Procedure

1. A pool of potential test cases.
2. At each iteration, a subset of the pool is created by removing a test cases which maximize the $NCD_1$
3. A decreasing size sequence is created and we choose the desired size from sequence as the diverse test set.

Software Subjects and Measures

Software

1. JEuclid, ROME and NanoML are java libraries.
2. Replace is a C application.

Greedy Coverage Algorithm

We consider Greedy method achieves the best coverage because it selects test cases from the whole pool.

Coverage

JaCoCo was used to measure the structural coverage of the java libraries. Replace was measured by the detection of seeded faults in the application.
Q1: Are higher levels of I-TSD associated with higher levels of coverage?

Answer:
Test sets with higher input diameter on average have higher code coverage.

Table 1: Spearman rank correlation values between I-TSDm and instruction coverage for three different SUTs and three different test set sizes.

<table>
<thead>
<tr>
<th>SUT</th>
<th>Test Set Size 10</th>
<th>Test Set Size 25</th>
<th>Test Set Size 50</th>
</tr>
</thead>
<tbody>
<tr>
<td>J丝路</td>
<td>0.59</td>
<td>0.67</td>
<td>0.52</td>
</tr>
<tr>
<td>NanoXML</td>
<td>0.50</td>
<td>0.60</td>
<td>0.26</td>
</tr>
<tr>
<td>ROME</td>
<td>0.60</td>
<td>0.57</td>
<td>0.82</td>
</tr>
</tbody>
</table>

Positive Correlation
Overall the correlation can be considered a moderate positive correlation.

Q2: Does I-TSD selection lead to higher code coverage than randomly selection?

Fig. 3: Instruction coverage (normalized) of the ROME library against size for test sets selected using the greedy algorithm (red), I-TSDm (green), and random algorithm (blue) from an initial pool of 250 randomly-generated MatML inputs. The plots show the average values over 10 runs.
Q2: Does I-TSD selection lead to higher code coverage than randomly selection?

**Answer**

Test sets selected for highest test input diameter lead to higher code coverage than randomly selected test sets.

---

Q3: Does I-TSDm selection lead to higher code coverage than random selection with size constraints?

They redo experiment 2 with size constraints.
Structural Coverage Ability with Size Constraints

Q3: Does I-TSDm selection lead to higher code coverage than random selection with size constraints?

Answer

Test sets selected for highest test input diameter lead to higher code coverage than randomly selected test sets with size constraints.

Fault Finding Ability

Q4: Does I-TSD selection lead to higher fault coverage than random selection?

Answer

Test sets with larger test set diameter (I-TSDm) may have better fault-finding ability.

Selection Time

Running Time

By collecting the running time of I-TSDm selection procedure, they are able to model its performance. They found a model of form $S_{avg} \times N^2$ where $S_{avg}$ is the average length of strings being selected and $N$ is the number of elements.
Selection Time

Running Time
By collecting the running time of I-TSDm selection procedure, they are able to model its performance. They found a model of form $S_{avg} \times N^2$ where $S_{avg}$ is the average length of strings being selected and $N$ is the number of elements.

Answer
The TSDm test selection procedure scales quadratically in the size of the initial pool of tests to select from, and linearly with the average length of the tests.

Discussion

Limitations
1. It is unclear what is the basis for a difference between two test cases.
2. They do not understand the effect that the choice of compression algorithm has on NCD metrics.

Conclusion
In this paper, a diversity test set selection method called TSDm is proposed based on NCD.
1. moderate to high positive correlation to instruction coverage
2. higher fault coverage
3. higher code coverage
4. higher code coverage with control size
5. time complexity $O(N^2)$

Questions
Any question?