Enhancing Supervised Classifications with Metamorphic Relations

Liming Xu
International Doctoral Innovation Centre
University of Nottingham Ningbo
Ningbo, Zhejiang, China
liming.xu@nottingham.edu.cn

Dave Towey
AIOP Group, School of Computer Science
University of Nottingham Ningbo
Ningbo, Zhejiang, China
dave.towey@nottingham.edu.cn

Andrew P. French
School of Computer Science
University of Nottingham
Nottingham, UK
andrew.p.french@nottingham.ac.uk

Steve Benford
School of Computer Science
University of Nottingham
Ningbo, Zhejiang, China
steve.benford@nottingham.ac.uk

Zhi Quan Zhou
School of Computing and Information Technology
University of Wollongong
Wollongong, NSW, Australia
zhiquan@uow.edu.au

Tsong Yueh Chen
Dept. of Computer Science and Software Engineering
Swinburne University of Technology
Hawthorn, VIC, Australia
tychen@swin.edu.au

ABSTRACT
We report on a novel use of metamorphic relations (MRs) in machine learning: instead of conducting metamorphic testing, we use MRs for the augmentation of the machine learning algorithms themselves. In particular, we report on how MRs can enable enhancements to an image classification problem of images containing hidden visual markers (“Artcodes”).

Working on an original classifier, and using the characteristics of two different categories of images, two MRs, based on separation and occlusion, were used to improve the performance of the classifier. Our experimental results show that the MR-augmented classifier achieves better performance than the original classifier, algorithms, and extending the use of MRs beyond the context of software testing.

CCS CONCEPTS
- Software and its engineering → Software testing and debugging;
- Computing methodologies → Supervised learning by classification, Bagging;

KEYWORDS
Metamorphic testing, metamorphic relations, supervised classification, Artcodes, random forests

1 INTRODUCTION
Over the past two decades, machine learning techniques have been increasingly adopted by the research community to solve a range of practical problems. For researchers in the machine learning and software testing communities, building accurate learning models and verifying their quality are major topics. Due to the nature of machine learning programs, test oracles (mechanisms to determine if software behaviour is correct) are generally hard to define. Metamorphic testing (MT) has been used to alleviate the oracle problem in testing machine learning software [25, 31, 32]. Machine learning techniques have also been used to identify metamorphic relations (MRs) [17]. MT has been used to further analyse classification results of machine learning systems [4].

In the literature, MRs have been used in software verification and validation, and to assess the quality of software [34] — in this paper, we report on expanding the existing role of MRs to use as a kind of post adjustor to a machine learning program, to build a more accurate learning model. Compared to the reported use of MT in [4], in this paper, MR is used to adjust the both the inputs and outputs of a machine learning system. To the best of our knowledge, this is the first time MRs have been extended to such a use. Using an example of the Artcodes classification problem [33] — similar to QR codes [30], Artcodes are visual codes where bespoke designs can be scanned — we identify MRs for each category of inputs, and use them to augment the original classifier, improving its performance.

The rest of this paper is organised as follows. Section 2 gives a brief explanation of metamorphic relations. Section 3 describes the Artcode classification. Section 4 presents the details of the MR-augmented classifier. The experimental evaluation of the MR-augmented classifier’s performance is given in Section 5. Finally, Section 6 concludes the paper.

2 METAMORPHIC RELATIONS
In software testing, an inability to determine if software is behaving correctly, or producing the correct output, is called the oracle problem [1]. Metamorphic testing (MT) is an approach that can alleviate the oracle problem [6, 10], MT has been investigated and adopted by a growing number of researchers and practitioners [7, 16, 20, 21, 28], successfully uncovering software problems, even in extensively
Background imagery

Recognisable foreground

blob

region

Figure 1: Illustration of the components of an Artcode.

tested systems [8, 9, 19]. Central to MT is a set of metamorphic relations (MRs), which are expected relations among the inputs and outputs of multiple executions of the intended program’s functionality. Instead of examining the behaviour or output for an individual input, MT checks the SUT against selected MRs, with violations of an MR indicating the presence of a fault.

An example MR for a database management system is that the system should return the same results for a query with the search condition “A and B” and a query with the search condition “B and A”.

4 AUGMENTED CLASSIFIER

Typically, the first step with conventional classifiers involves creating feature vectors that distinctively describe each class. Machine learning models can then be used to predict the class of individual inputs. To date, to the best of our knowledge, no attempt has been made to make use of the metamorphic properties or MRs inherent in classification problems to enhance or rectify the classification outputs. Inspired by the various successes of MT, we examined the Artcode classification domain to identify MRs which we then used to augment and enhance the original classifier.

The MRs were identified through observation of the impact on classification results among different classes (or labels, e.g. Artcode or non-Artcode) when feeding in predefined inputs. In particular, the MRs allowed us to express probabilistically the likelihood of a modified input being Artcode or not — labelled Artcode or non-Artcode classes, in this study. The Artcode class follows the topological definition of Artcodes, whereas the non-Artcode class comprises images that do not conform to these topological rules.
describing the geometry or structure of fixed objects, SOH is used to describe the topological structure of images through analysis of the symmetry and smoothness of the orientation histogram. The SOH is constructed by quantifying these two aspects of the orientation histogram using similarity measurements such as procrustes [24] or Chi-squared ($\chi^2$) distance [26].

After calculation of the SOH feature vector of each image, random forests were trained and used to predict the newly input image. As an ensemble learning method, a random forest has a number of decision trees (nTrees) in the forests. Therefore, it is an appropriate method to be used in Artcode classification.

### 4.2 Metamorphic relations

As described in Section 3, Artcodes are composed of a number of connected regions. Each region is an independent entity that contains several solid blobs, and therefore has a complete topological structure. Additionally, an Artcode image might contain several independent Artcodes, which means that parts of Artcode images are likely to be complete regions or Artcodes themselves. In other words, parts of Artcodes are "simplified" Artcodes which will be classified as "Artcodes" by the original classifier with a relatively high probability. On the other hand, non-Artcode images (ideally) should not have those characteristics: parts of non-Artcode images do not have the predefined topology, and they will be treated as "non-Artcodes" by the original classifier. Therefore, parts of Artcode images are more likely to be classified as "Artcodes" than parts of non-Artcode images. Based on this observation, and the characteristics of the original classifier, we identified two MRs: Separation and Occlusion.

**MR1-Separation.** Separation involves splitting the input image uniformly into a number of sections, or blocks. For example, Figure 4(a) presents separation masks to generate four uniform blocks by intersecting them with input images. This MR is based on the observation that the blocks of Artcodes could be classified as "Artcode" with a higher probability than the blocks of non-Artcode images. If we select the number of blocks appropriately, this difference in the total probability of all blocks may provide more clues for classification. MR1-Separation can be formulated as:

$$\sum_{i=1}^{n} \Pr(B_{S_i}^i) \geq \sum_{i=1}^{n} \Pr(B_{O_i}^i) \quad (1)$$

where $n$ is the number of image blocks; $\Pr()$ is the probability to be classified as Artcode by the original classifier; and $B_{S_i}^i$ and $B_{O_i}^i$ denote the $i$th block of the Artcode and non-Artcode image generated after MR-Separation, respectively.

**MR2-Occlusion.** Occlusion is similar to Separation, but the image blocks are not separated uniformly — blocks with overlapped areas are permitted. As shown in Figure 4(b), four occlusion masks are provided to intersect with the input image and output the image blocks outlined by white regions. Based on this, we have:

$$\sum_{i=1}^{m} \Pr(B_{O_i}^i) \geq \sum_{i=1}^{m} \Pr(B_{O_i}^i) \quad (2)$$

Occluded images generally keep half of the properties of the input images: half Artcode images have a high probability to be classified as "Artcode" by the original classifier, occluded images of non-Artcodes are still as likely to be labeled as "non-Artcode:" 

$$\Pr(B_{O_a}^i) \geq \Pr(B_{O_d}^i) \quad (3)$$

where $m$ is the number of masks; $B_{O_d}^i = \cap(I_a, M_i)$ and $B_{O_a}^i = \cap(I_a, M_i)$ outputs the overlapped areas of Artcode and non-Artcode images $I_a$ and $I_a$ and the $i$th mask $M_i$; and $B_{O_a}^i$ and $B_{O_d}^i$ denote the
Figure 5: The MR-augmented classifier framework. This framework includes three stages: Follow-up input generation, Prediction, and Rectification. Follow-up input generation involves splitting the input image into blocks based on the two MRs; Prediction makes predictions of these image blocks using the original classifier; and Rectification involves the calculation of the total probability \( p \) of belonging to the Artcode, and then rectifies the predictions of the original classifier on the basis of the \( p \) value and the two predefined thresholds \( t_1 \) and \( t_2 \).

5 EXPERIMENTAL EVALUATION

5.1 Dataset

In order to study the Artcodes classification problem, we created a dataset containing 47 Artcode and 116 non-Artcode images. The non-Artcode images (comprising logos, drawings, and graphics) were all created by humans, and were deliberately selected such that they would appear very similar to actual Artcode images. This means that this dataset is very challenging for Artcodes classification. Because Artcodes are manually created by designers, the number of available Artcodes is currently small, but work is ongoing to extend the dataset.

5.2 Cross validation

Cross-validation is a commonly used model validation technique for assessing how a learning model will generalize to a dataset [12, 18]. One of the main reasons for using cross-validation rather than conventional validation (partitioning the dataset into two sets of 70% for training and 30% for testing) is that there is not sufficient data available to partition into separate training and test sets without losing significant modeling or testing capability. In these cases, a fair way to properly estimate model prediction performance is to use cross-validation [29]. We used \( k \)-fold cross-validation, which involves randomly partitioning a dataset into \( k \) equally-sized subsets, and keeping one single subset as the validation data for testing the trained model, and using the remaining \( k-1 \) subsets as training data. The cross-validation process is then repeated \( k \) times (the folds).

5.3 Experimental setting

We implemented an augmented classifier based on the framework shown in Figure 5 using Matlab, and evaluated its performance using cross-validation on the Artcodes dataset. As there is no existing research on Artcodes classification, we only compare the
Figure 6: Performance comparison between the original and MR-augmented classifier with different \( n\text{Trees} \) values and \( t_1, t_2 = 0.2 \).

Table 1: Confusion matrix

<table>
<thead>
<tr>
<th>Actual</th>
<th>Predicted</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Artcode</td>
<td>True Positive (TP)</td>
<td>False Negative (FN)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>non-Artcode</td>
<td>False Positive (FP)</td>
<td>True Negative (TN)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

MR-augmented classifier with the original classifier presented in Section 4.1. Because random forests are used in the classifier, the performance exhibits a certain level of variation on each execution due to the random variable selection from the feature vector. Ten runs of cross-validation were therefore conducted to calculate the average performance.

Considering the imbalance of the dataset (with more non-Artcode images) we selected five performance metrics to provide an informative view of the augmented classifier’s performance: Precision, Recall, Accuracy, \( F_2 \) measure, and MCC (Matthews correlation coefficient) [22]. These five measures are calculated based on a confusion matrix (Table 1). A confusion matrix, also known as an error matrix, is a specific table layout that allows visualization of the performance of an algorithm — each row represents the instances in an actual class while each column represents the instances in a predicted class (or vice versa) [27].

\[
\text{Precision} = \frac{TP}{TP + FP} \tag{4}
\]
\[
\text{Recall} = \frac{TP}{TP + FN} \tag{5}
\]
\[
\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \tag{6}
\]
\[
F_2 = \frac{(1 + \beta^2) \cdot \text{Precision} \cdot \text{Recall}}{\beta^2 \cdot \text{Precision} + \text{Recall}} \tag{7}
\]

where \( \beta = 2 \). As shown in Equations 4 and 5, Precision is the proportion of true positives among the all predicted positives, and Recall is the proportion of true positives over the total amount of actual positives. The \( F_2 \) measure uses a weighted average of Precision and Recall to evaluate the classification effectiveness, giving twice as much importance to recall as to precision.

Compared with Accuracy, the \( F_2 \) measure provides more insight into the performance of a classifier, but can be sensitive to data distributions. MCC (Equation 8) is in essence a correlation coefficient
between the observed and predicted classifications, incorporating true and false positives and negatives.

\[
MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{8}
\]

MCC is generally regarded as one of the best measures for classifier performance evaluation [27], and remains effective even if the dataset is imbalanced. The tuning parameters — the number of decision trees (nTrees) in the random forests, and the thresholds \( t_1 \) and \( t_2 \) — were studied in the experiment, as was their impact on the classifier.

The values of \( t_1 \) and \( t_2 \) are strongly related to the given values in the weight vector, and, according to Equation 3, the weights of blocks generated by Occlusion are greater than those for Separation. We separated the input images uniformly into four blocks, and overlapped with four occlusion masks, and thus had 8-dimensional blocks generated by Occlusion that are greater than those for Separation.

All performance metric values reported are the average values calculated from ten executions of \( k \)-fold cross-validation [18]. As explained in Section 5.2, because of the limited number of samples in the Artcodes dataset, a 5-fold cross-validation was used to ensure sufficient training and testing set sizes for the performance evaluation. Two combinations of the thresholds \( t_1 \) and \( t_2 \) were used to study the classifier’s tuning parameters. In all graphs in Figures 6 and 7, higher values indicate better performance.

The impact of nTrees on the augmented classifier’s performance is illustrated in Figures 6 and 7, which show a stable performance across different numbers of nTrees in terms of the five evaluation metrics. This means that the augmented classifier is not sensitive to changes in the number of nTrees, a property it inherits from the original classifier.

Figure 7: Performance comparison between the original and MR-augmented classifier with different nTrees values and \( t_1 = 0.15, t_2 = 0.3 \).

To simplify calculations, we also used 1 and 0 in the prediction vector \( p \) to represent the “Artcode” and “non-Artcode” classes, respectively.

5.4 Experimental results

All performance metric values reported are the average values calculated from ten executions of \( k \)-fold cross-validation [18]. As explained in Section 5.2, because of the limited number of samples in the Artcodes dataset, a 5-fold cross-validation was used to ensure sufficient training and testing set sizes for the performance evaluation. Two combinations of the thresholds \( t_1 \) and \( t_2 \) were used to study the classifier’s tuning parameters. In all graphs in Figures 6 and 7, higher values indicate better performance.

The impact of nTrees on the augmented classifier’s performance is illustrated in Figures 6 and 7, which show a stable performance across different numbers of nTrees in terms of the five evaluation metrics. This means that the augmented classifier is not sensitive to changes in the number of nTrees, a property it inherits from the original classifier.
For various numbers of nTrees and fixed thresholds $t_1$ and $t_2$, the augmented classifier outperforms the original classifier in terms of all five metrics. The augmented classifier performs better in terms of both precision (Figures 6(a) and 7(a)) and recall (Figures 6(b) and 7(b)) than the original classifier, with an average of about 10-15% improvement for both threshold combinations. As shown in Figures 6(c) and 7(c), for both threshold combinations, the augmented classifier has slightly better Accuracy than the original classifier, about 2-3% improvement on average. Although the MR-augmented classifier shows improved performance with the Artcodes class, the small percentage of artcodes in the dataset does not contribute strongly to the overall evaluation of Accuracy, which is influenced by both true positives and true negatives.

In contrast, $F_2$ measure and MCC are more informative measures of overall performance, even when the dataset is imbalanced. As shown in Figures 6(d)(e) and 7(d)(e), the augmented classifier obtains better values (approximately 15-20% improvement) than the original for different numbers of nTrees, showing an overall improved performance of the MR-augmented classifier. However, due to the imbalance of the dataset, the MCC values for the two classifiers are relatively low.

Overall, the MR-augmented classifier achieves improved performance according to the five evaluation metrics. This improved performance is sensitive to the threshold values for $t_1$ and $t_2$, but not to nTrees. The impact of nTrees on the classifier’s performance is relatively small, but larger numbers of nTrees require more time to train the classifier and make the classification predictions. Thus, careful selection of the tuning parameter values is necessary to ensure the performance improvement of the original classifier.

6 CONCLUSION

In this paper, we have reported on a study using metamorphic relations (MRs) to improve binary classification in machine learning. To the best of our knowledge, this is the first use of MRs in such an application. Two MRs were identified based on properties of the input data and the usage of the classification model, and an augmented classifier using these two MRs was designed to show the applicability of the technique. Experimental evaluation showed the performance improvement across certain aspects of the original classifier, demonstrating the potential to apply MT theories and techniques to machine learning applications. The experimental evaluation also showed the importance of the tuning parameters $t_1$ and $t_2$ on the performance of the augmented classifier. Our future work will include further examination of other parameters, including the number of blocks in the separation and occlusion MRs, the given values of the weight vector, and the adaptive values of thresholds $t_1$ and $t_2$.

Although this has been a preliminary study, the results are very promising, and clearly demonstrate the potential for MR-augmentation of classifiers. More practical and theoretical work will be necessary to fully investigate this new research direction, including more case studies examining application of MRs to other well-known machine learning problems, such as face and object detection.

ACKNOWLEDGMENTS

The authors acknowledge the financial support from the International Doctoral Innovation Centre, Ningbo Science and Technology Bureau, and the University of Nottingham. This work was supported by grants from the UK Engineering and Physical Science Research Council (Grant No. EP/L015463/1), the National Natural Science Foundation of China (Grant No. 71471092) and the Ningbo Science and Technology Bureau (Grant No. 2014A35006).

REFERENCES


Enhancing Supervised Classifications with Metamorphic Relations Conference’17, July 2017, Washington, DC, USA


[26] Karl Pearson. 1900. X. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science 50, 302 (1900), 157–175.


