MEG: Multi-objective Ensemble Generation for Software Defect Prediction (HOP GECCO'23)

Rebecca Moussa, Giovani Guizzo, Federica Sarro
University College London, London, UK

ABSTRACT

This Hot-off-the-Press abstract aims at disseminating our recent work titled "MEG: Multi-objective Ensemble Generation for Software Defect Prediction" published in the proceedings of the 16th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM) [4]. We believe this work is of interest for the GECCO community as it proposes a novel way to automatically generate ensemble machine learning models leveraging the power of evolutionary computation: MEG introduces the concept of *whole-ensemble generation* as opposed to the well known *Pareto-ensemble generation.* While we evaluate the effectiveness of MEG for Software Defect Prediction in our work, MEG can be applied to any classification or regression problem and we invite both researchers and practitioners to further explore its effectiveness for other application domains. To this end, we have made MEG's source code publicly available.

KEYWORDS

Multi-objective Ensamble, Search-Based Software Engineering

ACM Reference Format:
Rebecca Moussa, Giovani Guizzo, Federica Sarro. 2023. MEG: Multi-objective Ensemble Generation for Software Defect Prediction (HOP GECCO'23). In *Genetic and Evolutionary Computation Conference Companion (GECCO '23 Companion), July 15–19, 2023, Lisbon, Portugal.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3583133.3595850

1 INTRODUCTION

Early software defect identification is considered an important step towards software quality assurance. To this end, research in software defect prediction (DP) aims at the automation and early identification of problematic software components, in order to direct the most of the testing effort towards them. A variety of automated approaches, ranging from traditional classification models to more sophisticated learning approaches, have been explored. Among these, recent studies have found the use of ensemble prediction models (i.e., aggregation of multiple base classifiers) achieves more accurate results than those that would have been obtained by relying on a single classifier. However, designing an ensemble, specifically the choice of base classifiers, their hyper-parameter values, and the choice of the strategy used to aggregate the predictions requires a non-trivial amount of effort and expertise. An inappropriate choice can have a detrimental effect on the performance of the ensemble. On the other hand, given the size of the search space, examining all possible combinations is not computationally affordable, especially that these design choices are interconnected, therefore cannot be optimised separately. Such a large search space renders Search-Based Software Engineering a suitable solution for the problem of automatically generating effective ensembles for DP. In our recent work [4], we propose MEG, a novel use of multi-objective evolutionary algorithms to automatically generate defect prediction ensembles. Our proposal introduces the concept of *whole-ensemble generation* as opposed to the existing *Pareto-ensemble* one. Moreover, our study is the first to investigate the effectiveness of evolutionary ensemble for defect prediction. In the remaining of this article we introduce the main aspects of MEG and summarise the results we obtained when assessing its effectiveness in a large-scale empirical study.

2 MEG

MEG aims at improving the performance of existing machine learners by automating the process of generating ensemble predictors, tuning their hyperparameters and selecting a suitable aggregation strategy. MEG is a Multi-objective Evolutionary Algorithm that evolves a population of ensembles across multiple generations and outputs the ensembles with the best trade-off among the fitness functions which guide the search. While existing evolutionary algorithms for ensemble generation like DIVACE [1] work as a Pareto-ensemble technique (i.e., evolving each base classifier individually as a chromosome and aggregating the non-dominated ones in one optimal ensemble at the end of the evolutionary process), MEG takes a different approach where each chromosome in a population is an ensemble itself which is evolved during the search.

Representation: A chromosome consists of three parts: i) a set of machine learners; ii) a set of hyper-parameters; and a set of Ensemble/Aggregation Strategy. The machine learner set is encoded as a binary array consisting of bits, each one corresponding to a single base model. If, at a specific index i, the bit denotes 1, then this signifies that the i-th machine learner is active and will be included in the ensemble, otherwise, it will not be considered. Any existing machine learner can potentially be part of this set. In our work we explored three base classifiers (Naive Bayes, k-Nearest Neighbors and Support Vector Machines) because the most related work [6] to ours used these specific learners. This allows a fair comparison in our empirical study, however future work can extend MEG to incorporate other machine learners (classifier/regressor) depending on the problem at hand. The parameter set consists of the hyper-parameter values of each of the ML models considered. For example, if one of the models considered is Random Forest (RF), then the parameter list would include values for the number of trees, number of features, the maximum depth of the trees, etc. The third set consists of a single integer value (later converted into a categorical one) representing the aggregation strategy to be used by the ensemble to aggregate the predictions of all constituent models. Given that we aim at investigating ensembles composed by different types of base machine learners, our set is composed by four heterogeneous aggregation strategies: majority voting, weighted majority voting, stacking, and average voting.

Fitness Functions: MEG uses two fitness functions to guide the search for ensembles: accuracy and diversity. The accuracy of an ensemble depicts how well it can predict the labels of the instances under consideration, which in the context of this work are "defective" (true) or "non-defective" (false). Naturally, the more accurate the ensemble, the better. On the other hand, the diversity measures assess how different the predictions of the classifiers in the ensemble are. Diversity is an important factor when designing ensembles, since a diverse ensemble is more likely to predict "corner cases" instances. However, in a classification problem, the more the classifiers disagree in their predictions, the less accurate the ensemble tends to be. For instance, for a given instance with the true label "defective", if two classifiers each predict "defective" and "non-defective" respectively, then we obtain diverse results, but with 50% accuracy. Hence, these two measures are conflicting, and MEG aims at optimising both to strike an optimal trade-off. We use Mathews Correlation Coefficient (MCC) [5] as the accuracy objective, and Disagreement [2] as the diversity measure. MCC represents the correlation coefficient between the actual and predicted classifications. MCC values range in $[-1, 1]$, where +1 indicates a perfect prediction, 0 signifies that the prediction is no better than random guessing, and -1 represents a completely miss-classified output. We use MCC as this measure has been recommend in alternative to other previously popular measures, such as F-measure, which have been shown to be biased [5] when the data is imbalanced (as it is frequently the case in DP). MCC is a more balanced measure which, unlike the other measures, takes into account all the values of the confusion matrix [5]. Diversity, is computed based on the disagreement measure [2], which captures the prediction disagreement between groups of classifiers. We use this measure as it was used in previous work for DP [6] to which we compare MEG. In short, disagreement measures how many of the base classifiers' predictions contradict each other. Its values range in $[0, 1]$, where the higher the value, the more two classifiers differ. To calculate the diversity of the ensemble, we find the average pairwise disagreement between all pairs of its constituent members. An ideal ensemble for the defect prediction problem would be the one that yields a high MCC and a high disagreement. However, this is hard to achieve in practice, thus the need of using a MOEA to strike an optimal trade-off between these two competing goals.

Genetic Operators Since our chromosome is constituted of three arrays of different type, the crossover and mutation occur in three parts, each of which is adapted to work for the type of array at hand. MEG uses a Single Point Crossover operator with 95% probability. This crossover operator takes two parents and combines their genes to generate two children. After crossover, the children undergo mutation with a lower probability. Since the classifiers array (bit array) and parameters array (double array) have 15 indexes, the mutation probability is set to 0.07. Thus, it is expected that each child will have one of its bit/double gene mutated. For the ensemble strategy (int array) with one index, the probability is set to 0.25. Hence, it is expected that the ensemble aggregation strategy is mutated once in every four children. MEG uses a Bit Flip Mutation operator for the classifier array, and a Simple Random Mutation operator for the parameter and strategy arrays. The former simply flips the bit by changing it to 1 if the gene is 0, or to 0 otherwise. The latter generates a random number for a mutated gene. In our experiment, we set the population size to 100 and the stopping condition to 10,000 fitness evaluations (i.e., it stops after 100 generations).

3 EMPIRICAL STUDY SUMMARY

To assess the effectiveness of MEG, we conduct a thorough large-scale empirical study involving a total of 24 real-world software versions and 16 cross-version defect prediction scenarios, assessed according to the latest best practice for the evaluation of defect prediction and search-based approaches. We compare MEG's performance against (1) traditional base classifiers (as a sanity check), (2) a state-of-the-art multi-objective ensemble approach proposed by Petrić et al. [6] (the only one to use a diversity measure for ensemble DP), and (3) DIVACE [1] (a seminal work for Pareto-ensemble generation). Our results show that MEG is able to generate ensembles with similar or more accurate predictions than those achieved by all the other approaches considered in 73% of the cases (with large effect sizes in 80% of them). Moreover, MEG's performance demonstrates that it is possible to relieve engineers from the error-prone, burdensome, and time-consuming task of manually designing and experimenting with different ensemble configurations in order to find an optimal one for the problem at hand.

4 WHAT'S NEXT

MEG introduces the concept of *Whole-ensemble generation* as opposed to the well known *Pareto-ensemble generation*. The results of our empirical study highlights that MEG is highly effective for Software Defect Prediction, and encourage further studies on the effectiveness of MEG in other application domains. In fact, the set of machine learners and fitness functions is customisabile and the idea proposed herein is applicable to any classification or regression problem. To facilitate future studies, we have made MEG's source-code publicly available online, along with a replication package, containing the datasets, the raw results and the scripts we realised to evaluate the results [3].

ACKNOWLEDGEMENTS

This research is supported by the ERC grant no. 741278 (EPIC).

REFERENCES

[1] A. Chandra and X. Yao. 2006. Ensemble Learning Using Multi-Objective Evolutionary Algorithms. *Journal of Math. Modelling and Alg.* 5, 4 (2006), 417–445.

[2] L.I. Kuncheva and C.J. Whitaker. 2003. Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine learning* 51, 2 (2003).

[3] R. Moussa, G. Guizzo, and F. Sarro. 2022. https://github.com/SOLAR-group/MEG

[4] R. Moussa, G. Guizzo, and F. Sarro. 2022. MEG: Multi-objective Ensemble Generation for Software Defect Prediction. In *Procs. of the 16th ACM/IEEE ESEM.*

[5] R. Moussa and F. Sarro. 2022. On the Use of Evaluation Measures for Defect Prediction Studies. In *Procs. of the 31st ACM SIGSOFT ISSTA.*

[6] J. Petrić, D. Bowes, T. Hall, B. Christianson, and N. Baddoo. 2016. Building an Ensemble for Software Defect Prediction Based on Diversity Selection. In *Procs. of the 10th ACM/IEEE ESEM.*