SGP-DT: Towards Effective Symbolic Regression with a Semantic GP Approach Based on Dynamic Targets

Stefano Ruberto  
Institute for Biomedical Informatics  
Philadelphia, USA  
stefano.ruberto@pennmedicine.upenn.edu

Valerio Terragni  
Faculty of Informatics  
Lugano, Switzerland  
valerio.terragni@usi.ch

Jason H. Moore  
Institute for Biomedical Informatics  
Philadelphia, USA  
jjhmoore@upenn.edu

ABSTRACT
Semantic Genetic Programming (SGP) approaches demonstrated remarkable results in different domains. SGP-DT is one of the latest of such approaches. Notably, SGP-DT proposes a dynamic-target approach that combines multiple GP runs without relying on any form of crossover. On eight well-known datasets SGP-DT achieves small RMSE, on average 25% smaller than state-of-the-art approaches.

CCS CONCEPTS
- Theory of computation → Bio-inspired optimization; Genetic programming;

KEYWORDS
Genetic Programming, Semantic GP, Natural Selection, Symbolic Regression, Residuals, Linear Scaling, Crossover, Mutation

ACM Reference Format:

1 INTRODUCTION
Semantic Genetic Programming (SGP) uses richer feedback during the evolution that incorporates semantic awareness, which has the potential to improve the power of genetic programming [7, 8].

Symbolic Regression is characterized by a set of training cases, defined as a pair of inputs \( x \) and desired output \( y \). Following most SGP approaches [8], we intend the semantics of an individual \( I \) as a vector \( \text{sem}(I) = (y_1, y_2, \ldots, y_m) \) of responses to the \( m \) inputs of the training cases [4]. Let \( \text{sem}(\hat{y}) = (\hat{y}_1, \hat{y}_2, \cdots, \hat{y}_m) \) denote the semantic vector of the target (as defined in the training set), where \( \hat{y}_1, \hat{y}_2, \cdots, \hat{y}_m \) are the desired outputs. SGP defines the semantic space [8] equipped with a distance between the semantic vectors of the individuals \( \text{sem}(I) \) and the target \( \text{sem}(\hat{y}) \). The effectiveness of SGP depends on the availability of GP operators that can move in the semantic space towards the global optimum. Both classic and semantic-aware crossovers suffer from several drawbacks [4, 8].

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

GECCO ’20 Companion, July 8–12, 2020, Cancún, Mexico
© 2020 Copyright held by the owner/author(s).
https://doi.org/10.1145/3377929.3397486

Our recent paper proposes SGP-DT [6], a SGP technique that effectively navigates the semantic space without relying on any form of crossover. SGP-DT invokes multiple GP runs, each of which is guided by a (dynamic) target that focuses on a particular characteristic of the problem at hand. SGP-DT combines with linear scaling the models returned by each GP run into a final model. In this paper, we summarize the technique and major accomplishments.

2 SGP-DT
Figure 1 gives an overview of our approach. SGP-DT runs a predefined number of GP algorithms \( GP_1, GP_2, \cdots, GP_N \). We call these runs external iterations. As opposed to the internal iterations (i.e., GP generations) that the GP algorithm performs at each run, each run has associated a dynamic target \( dt \) that changes at each external iteration. The dynamic target dictates the fitness of the individuals, defined as the variance of the difference between \( \text{sem}(I) \) and the current target (i.e., \( \sigma^2(\text{sem}(I) - dt) \)). Each GP run performs a fixed number of internal iterations and returns the fittest solution that we call partial model (PM). For each individual (and thus for each partial model) we compute the coefficients \( a \) and \( b \) of the linear scaling technique [2], which entails a bound on the worsening of the offspring at each internal and external iteration [6].

The dynamic target of the first external iteration is the desired output \( \hat{y} \) as specified by the training set. The dynamic target of the \( i \)-th iteration is the difference between the previous target and the semantic of the partial model returned by the previous iteration (i.e., residual error): \( dt_{i-1} \cdot \text{sem}(a_{i-1} + b_{i-1} \cdot PM_{i-1}) \). As such, SGP-DT leads to dynamic targets that change at each external iteration incorporating the semantic information. Each partial model focuses on a different characteristic of the problem that the fitness function recognized to be important (at that iteration). This makes the search more efficient because the evolution focuses on a single characteristic at a time leaving unaltered the already optimized ones (and thus preserving the already discovered functionalities).
SGP-DT obtains the final solution with a linear combination \( \sum_{i=1}^{n} \theta_i + b_i \cdot P_{Mi} \). Notably, SGP-DT does not rely on any form of crossover, neither semantic nor classic, and thus avoiding their intrinsic limitations. SGP-DT implicitly recombinates different functionalities when it assembles the partial models into the final one.

3 EVALUATION RESULTS

We now summarize a series of experiments that we conducted to evaluate SGP-DT.

Datasets. We performed our experiments on eight well-known datasets of regression problems that have been used to evaluate most of the relate techniques. Table 1 shows their names, number of attributes, and number of instances.

<table>
<thead>
<tr>
<th>Data set</th>
<th># attributes</th>
<th># instances</th>
<th>source</th>
</tr>
</thead>
<tbody>
<tr>
<td>concrete</td>
<td>5</td>
<td>1,503</td>
<td>UCI</td>
</tr>
<tr>
<td>enc</td>
<td>8</td>
<td>768</td>
<td></td>
</tr>
<tr>
<td>enh</td>
<td>8</td>
<td>768</td>
<td></td>
</tr>
<tr>
<td>housing</td>
<td>14</td>
<td>586</td>
<td></td>
</tr>
<tr>
<td>tower</td>
<td>25</td>
<td>3,135</td>
<td></td>
</tr>
<tr>
<td>yacht</td>
<td>6</td>
<td>399</td>
<td></td>
</tr>
<tr>
<td>uball5d</td>
<td>5</td>
<td>6,024</td>
<td>[5]</td>
</tr>
</tbody>
</table>

Techniques under comparison. We compared SGP-DT with two techniques: LASSO [1] and \( \epsilon \)-LEXICASE [3].

LASSO [1] is a regression analysis method that uses the least square regression to linearly combine solution components. More specifically, LASSO incorporates a regularization penalty into least-squares regression and uses an \( \ell_1 \) norm of the model coefficients and uses a tuning parameter \( \lambda \) to specify the weight of this regularization [1].

\( \epsilon \)-LEXICASE [3] is an evolutionary technique that adapts the lexicase selection operator for continuous domains. The idea behind \( \epsilon \)-LEXICASE selection is to promote candidate solutions that perform well on unique subsets of samples in the training set, and thereby maintain and promote diverse building blocks of solutions [3].

For SGP-DT and \( \epsilon \)-LEXICASE we set a population size of 1,000 and a budget of 1,000 generations. For SGP-DT, we divided the 1,000 generations in 20 external iterations \( N_{\text{ext}} = 20 \), and thus the number of internal iterations of each GP run is 50. To cope with the stochastic nature of GP, we ran 50 trials for every technique on each dataset using 25% of the data for testing and 75% for training.

Root Mean Square Error (RMSE) comparison. Table 2 shows the median RMSE and the RMSE percentage decrease of SGP-DT with respect to LASSO and \( \epsilon \)-LEXICASE. A positive value means that SGP-DT has a lower (better) RMSE median.

4 CONCLUSION

SGP-DT presents a novel SGP approach that yields to low approximation error and computational cost when applied to the symbolic regression domain. On eight well-known datasets, SGP-DT outperforms both LASSO and \( \epsilon \)-LEXICASE. This is an important result as \( \epsilon \)-LEXICASE outperforms many GP-inspired algorithms [5].

GP techniques evaluated to produce the final solution (all comparisons are statistically significant). Comparing with \( \epsilon \)-LEXICASE, SGP-DT reduces the amount of node evaluations by a factor between 4.01\( \times \) and 9.26\%, obtaining better RMSE values than \( \epsilon \)-LEXICASE for seven out of eight datasets (see Table 2).

REFERENCES