Incremental Tree-edit Distance

Kieran O'Brien

It was initially proposed that code distance (a measure of how different the structure of one program is from another) would be a useful metric for gamification. However, performing this calculation is too slow to be responsive in a gamification context. Instead, it iss proposed that the distance could be updated as changes are made, using the information in the diff between program versions.

Two approaches were designed. One is Distance Update and the other is Distance Inference. Both of these require a shortest edit sequence (to change one tree into another) to be generated. Distance Update reuses previously calculated results as much as it can (making decisions based on the edit sequence) and Distance Inference compares edit sequences and looks for evidence that programs are diverging or converging.

These methods are tested on a variety of input data of different sizes. Accuracy and efficiency are recorded. Distance Inference was found to be mostly accurate for very small trees with occasional spikes of error. It was unreliable on large trees with error between fifty and one hundred percent. Distance Update was able to make modest savings on both sizes of tree but the overhead involved in building the edit sequence caused it to take longer than the standard algorithm.

It is concluded that further work needs to go into making Distance Inference more reliable, in which case it will prove useful in the gamification context in which it was originally imagined. The overhead involved in building the edit sequence needs to be dealt with for Distance Update to become useful, but even so the modest savings make it ill suited to a gamification context.