Paper Review: Automating String Processing in Spreadsheets Using Input-Output Examples

Gaurav Mahajan

FlashFill uses input-output examples as behavioral constraints and built-in string expression language as structural constraints. To search the space of candidate expressions, it first enumerates the set of all trace expressions (trace set) for an input-output pair. If the trace sets for all examples have a non empty intersection, then that would be the set of all target expressions. Otherwise, it first partitions the space such that each partition has a non empty intersection and then looks for conditionals.

The following program in the FlashFill string expression language extracts conference years for the given examples: $Substr(v, p1, p2)$, where

$$p1 \equiv Pos(TokenSeq(UpperTok, SpaceTok), NumTok, -1)),$$
$$p2 \equiv Pos(NumTok, \epsilon, -1)$$

Without the regular expression restriction, Lines 2 and 3 in GeneratePosition would be infinite loops since we would have infinite number of regular expressions matching any string.

EUSolver's condition abduction mechanism learns a decision tree with terms as labels, predicates as attributes and points as samples. FlashFill on the other hand, first greedily finds the best partition and then learns the boolean classifier. Consequently, it only allows conditionals at top level.