Discussion 5: Experiment Results & Tool Demo

June 16, 2020

Experiment Results

General Result:

	2-Multiphase	5-Multiphase	SVMRanker	LassoRanker
FINITE	39	42	30	24
INFINITE	34	34	34	37
UNKNOWN	61	58	70	73
TIME	1107s	5709s	162s	1695s

Experiment Results

Time Details:

	2-Multiphase	5-Multiphase	SVMRanker
sampling	79.6s	472.0s	11.6s
training	52.4s	396.7s	4.8s
z3-solving	229.2s	4016.1s	22.6s

Experiment Results

Examples in SV-COMP15/Mumeric & Crafted:

	2-Multiphase	5-Multiphase	SVMRanker
FINITE	22	22	20
INFINITE	7	7	7
UNKNOWN	23	23	25

Experiment Results

Num of linear and non-linear loops

	2-Multiphase	5-Multiphase	SVMRanker	LassoRanker
linear	21	23	14	22
non-linear	18	19	16	2

Tool Demo

```
clexma@clexma-ThinkPad-P52s:~/Desktop/Disk_D/gitRepos/RankingExp/src$ python3 CLIMain.py
SVMRanker --- Version 1.0
Usage: CLIMain.py [OPTIONS] COMMAND [ARGS]...

Options:
  --help  Show this message and exit.

Commands:
  lmulti           SOURCE: path of source program file LOG: path of log...
  lnested          SOURCE: path of source program file LOG: path of log...
  parseboogie      SOURCE: path of source boogie file PARSEOUTFILE: ath of...
  parsectoboogie   SOURCE: path of source c file OUTFILE: path of output...
  parsectopy       SOURCE: path of source c file OUTFILE: path of output...
```

Tool Demo

```
clexma@clexma-ThinkPad-P52s:~/Desktop/Disk_D/gitRepos/RankingExp/src$ python3 CLIMain.py lmulti --help
SVMRanker --- Version 1.0
Usage: CLIMain.py lmulti [OPTIONS] SOURCE [LOG] [DEPTH_BOUND]

  SOURCE: path of source program file  LOG: path of log folder, default set
  to ./Log_temp DEPTH_BOUND: depth bound of multiphase ranking function,
  default set to 2

Options:
  --filetype [C|BOOGIE]           --file C: input is c file. --file BOOGIE:
                                  input is boogie file

  --sample_strategy [ENLARGE|CONSTRAINT]
                                  --sample_strategy ENLARGE: enlarge the
                                  sample zone when sample num not enough.
                                  --sample_strategy CONSTRAINT: find feasible
                                  points by constraint if sample num not
                                  enough

  --cutting_strategy [MINUS|MINI|POS]
                                  use f(x) < b to cut --cutting_strategy POS:
                                  b is a postive number --cutting_strategy
                                  MINUS: b is a negative number
                                  --cutting_strategy MINI: b is the minimum
                                  value of sampled points

  --template_strategy [SINGLEFULL|FULL]
                                  templates used for learning
                                  --template_strategy SINGLEFULL: templates
                                  are either single variable or combination of
                                  all variables

  --print_all [T|F]               --print_all T: print all the information of
                                  the learning --print_all F: only print the
                                  result information of the learning
```

Tool Demo

```
clexma@clexma-ThinkPad-P52s:~/Desktop/Disk_D/gitRepos/RankingExp/src$ python3 CLIMain.py lnested --help
SVMRanker --- Version 1.0
Usage: CLIMain.py lnested [OPTIONS] SOURCE [LOG]

  SOURCE: path of source program file LOG: path of log folder, default set
  to ./Log_temp

Options:
  --filetype [C|BOOGIE]           --file C: input is c file. --file BOOGIE:
                                  input is boogie file

  --sample_strategy [ENLARGE|CONSTRAINT]
                                  --sample_strategy ENLARGE: enlarge the
                                  sample zone when sample num not enough.
                                  --sample_strategy CONSTRAINT: find feasible
                                  points by constraint if sample num not
                                  enough

  --print_all [T|F]               --print_all T: print all the information of
                                  the learning --print_all F: only print the
                                  result information of the learning

  --help                          Show this message and exit.
```

Tool Demo

```
clexma@clexma-ThinkPad-P52s:~/Desktop/Disk_D/gitRepos/RankingExp/src$ python3 CL
IMain.py  lmulti  ./CParserTest/a.08_true-termination.c --filetype C
```

```
  warnings.warn("Liblinear failed to converge, increase "
/home/clexma/.local/lib/python3.8/site-packages/sklearn/svm/_base.py:976: Conver
genceWarning: Liblinear failed to converge, increase the number of iterations.
  warnings.warn("Liblinear failed to converge, increase "
/home/clexma/.local/lib/python3.8/site-packages/sklearn/svm/_base.py:976: Conver
genceWarning: Liblinear failed to converge, increase the number of iterations.
  warnings.warn("Liblinear failed to converge, increase "
/home/clexma/.local/lib/python3.8/site-packages/sklearn/svm/_base.py:976: Conver
genceWarning: Liblinear failed to converge, increase the number of iterations.
  warnings.warn("Liblinear failed to converge, increase "
/home/clexma/.local/lib/python3.8/site-packages/sklearn/svm/_base.py:976: Conver
genceWarning: Liblinear failed to converge, increase the number of iterations.
  warnings.warn("Liblinear failed to converge, increase "
--------------------LEARNING MULTIPHASE SUMMARY-------------------
MULTIPHASE DEPTH:  1
LEARNING RESULT:  FINITE
-----------RANKING FUNCTIONS----------
0.9 *  x^1 - 0.9 *  y^1 + 0.7 * 1
```

TODOs

- Use translation of C to Boogie in Ultimate.
- Use AST instead of Python array for program input.
- Recurrent Set.