Weakly Supervised Semantic Parsing with Execution-based Spurious Program Filtering

Kang-il Lee [1]
4bkang@snu.ac.kr

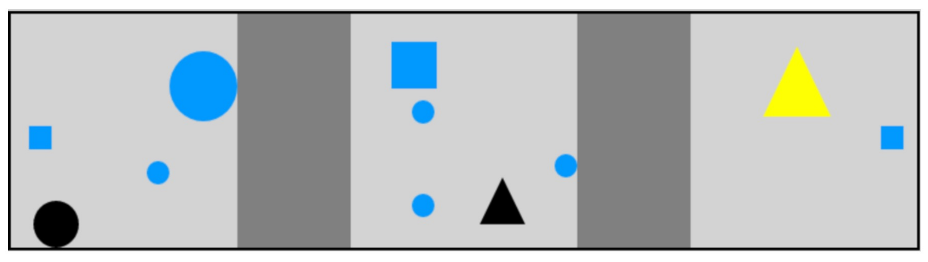
Segwang Kim [2]
ksk5693@snu.ac.kr

Kyomin Jung [1,3]
kjung@snu.ac.kr

[1] Department of ECE, Seoul National University, [2] Samsung Electronics Mobile eXperience, [3] IPAI, Seoul National University

EMNLP 2023

Task & Contributions



x : There is a blue square
w : [[{color: blue, shape: square}, {color: black, shape: circle}…], …]
z : objExists(square(blue(all_objects)))
z': objExists(black(circle(all_objects)))
y : True

Rank	Nation	Gold	Silver	Bronze	Total
1	Soviet Union	50	27	22	99
2	United States	33	31	30	94
3	East Germany (GDR)	20	23	23	66
4	West Germany (FRG)	13	11	16	40
5	Japan	13	8	8	29
6	Australia	8	7	2	17
7	Poland	7	5	9	21
8	Hungary	6	13	16	35
9	Bulgaria	6	10	5	21
10	Italy	5	3	10	18

x : How many nations won more than ten silver medals?
w : [[{Rank: 1}, {Nation: Soviet Union}, {Gold: 50}…], …]
z : count(filterNumberGreater(allRows, column:Silver, 10))
z': select(filterIn(allRows, column:Nation, Japan), column:Rank)
y : 5

Weakly Supervised Semantic Parsing

- Goal: **map *x* into *z***
- The dataset includes only utterance *x*, world *w* and denotation *y*. **Ground truth program *z* is not given.**
- During the training, a search algorithm **generates a pool of likely programs**, and filters out programs with incorrect execution results.
- **Spurious programs** like *z'*, **whose meaning is wrong but execution result is coincidentally correct**, are major challenges of the task.

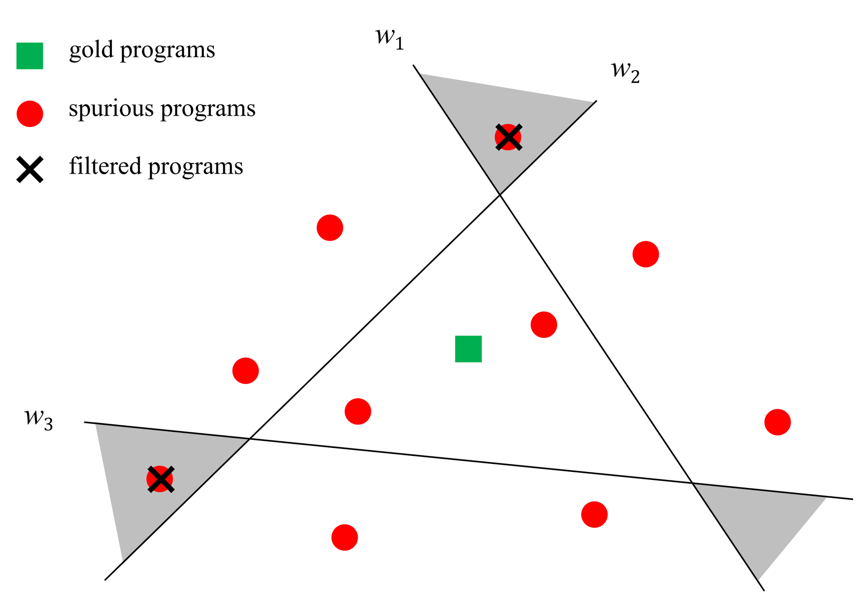
Contributions

- We propose **a novel program representation scheme** base on programs' execution results on various input worlds from the training set.
- We show that **running majority vote over execution results** and **filtering out programs with low vote score** consistently improves base parser performance on NLVR and WTQ.

Motivation

Execution-based Program Representation

- Retrieved worlds from training set (w_j's) partition the programs into several groups by their execution results.
- **Intuition: Correct programs lie near the centroid** and **spurious programs lie far from the centroid.**



Execution result space

Proposed Method

Filtering Programs with Majority Vote

Hard Vote

- First get the **centroid program representation r_*** with majority vote.
- Each program's contribution is weighted by some metric $W(\cdot)$.
- **Program score** is calculated based on the **distance from the centroid** representation r_* (higher the closer).

$$r_*^j = \underset{e \in E}{\operatorname{argmax}} \sum_{i=1}^{k} W(z_i)\mathbb{1}(r_i^j = e)$$

$$s_i = \frac{1}{n}\sum_{j=1}^{n} \mathbb{1}(r_i^j = r_*^j)$$

Soft Vote

- Instead of using centroid representation, each program contributes to the results based on the proportion it occupies in the execution results.

$$s_i = \sum_{j=1}^{n}\sum_{l=1}^{k} W(z_l)\mathbb{1}(r_i^j = r_l^j)$$

Collecting Execution Results

Domain 1 - NLVR

- "Informative" worlds in the training set are retrieved based on the BLEU score between *x* and each w_j's corresponding utterances.

Domain 2 - WikiTableQuestions

- Each program in WTQ is conditioned on a specific table and therefore cannot be used on others.
- We modify programs so that they can be executed on any table, while maintaining the semantic relationship between the programs.

Source table

Team	Wins	Losses	Win %	GB
Detroit Tigers	104	58	.642	0
Toronto Blue Jays	89	73	.549	15.0
New York Yankees	87	75	.537	17.0
Boston Red Sox	86	76	.531	18.0
Baltimore Orioles	85	77	.525	19.0
Cleveland Indians	75	87	.463	29.0
Milwaukee Brewers	67	94	.416	36.5

Target table

Rank	Nation	Gold	Silver	Bronze	Total
1	France	1	3	0	4
2	England	1	2	1	4
3	Ireland	1	1	0	2
–	Sweden	1	1	0	2
5	Belgium	1	0	2	3
6	Hungary	1	0	0	1
–	Netherlands	1	0	0	1
–	Spain	1	0	0	1
9	Scotland	0	1	0	1
10	Czechoslovakia	0	0	2	2
–	Italy	0	0	2	2
12	Denmark	0	0	1	1

z_1 : select(argmax(allRows, column:Wins), column:Team)
z_2 : count(filterNumberGreater(allRows, column:Wins, 100))

z_1' : select(argmax(allRows, column:Silver), column:Nation)
z_2' : count(filterNumberGreater(allRows, column:Silver, 2))

Experiments & Analysis

Approach	Dev.		Test-P		Test-H		Test
	Acc.	Con.	Acc.	Con.	Acc.	Con.	Con.
Abs. Sup. + ReRank (Goldman et al., 2018)	85.7	67.4	84.0	65.0	82.5	63.9	64.5
Iterative Search (Dasigi et al., 2019)	85.4	64.8	82.4	61.3	82.9	64.3	62.8
LLD (Gupta et al., 2021)	88.2	73.6	86.0	69.6	87.2	70.1	69.9
LLD + CR (Gupta et al., 2021)	89.6	75.9	86.3	71.0	89.5	74.0	72.5
LLD (w/ modified beam search)	90.8	77.8	88.3	73.4	89.0	74.6	74.0
+ Execution-based Filtering	90.5	**78.8**	**89.4**	74.2	**89.4**	**76.3**	**75.2**
LLD + CR (w/ modified beam search)	90.3	77.5	87.8	72.8	87.8	72.2	72.5
+ Execution-based Filtering	**90.9**	78.7	88.7	**74.9**	88.8	72.5	73.7

Approach	Dev.	Test
Zhang et al. (2017)	40.4	43.7
Liang et al. (2018)	42.3	43.1
Dasigi et al. (2019)	42.1	43.9
Agarwal et al. (2019)	43.2	44.1
Wang et al. (2019)	**43.7**	44.5
+ Execution-based Filtering	43.2	**44.8**

Main results on NLVR and WTQ

- **Our method improves the performance of base parsers consistently.**
- Our method is **domain-agnostic** and can augment existing weakly supervised semantic parser.

Score-spuriousness correlation

- Pearson correlation: **0.358**
- ROC-AUC: **0.738**
- Correct program scores: mean **0.997**, std **0.029**
- Spurious program scores: mean **0.899**, std **0.155**

τ	Precision	Recall	F1-score
0.8	99.5	40.0	49.5
0.9	99.6	57.8	66.3
1.0	99.4	82.0	85.7

- Spurious program detection performance on 30 NLVR training examples with various thresholds τ.

(Successful case) Sentence: There is at least one black item closely touching the bottom of a box.

Score	Program
1.0	**((* (* (object_count_greater_equals 1) black) touch_bottom) all_objects)**
1.0	**((* (* object_exists black) touch_bottom) all_objects)**
0.85	((* (* (* (object_count_greater_equals 1) black) touch_bottom) bottom) all_objects)
0.58	((* (* (object_count_greater_equals 2) black) touch_bottom) all_objects)
0.50	(box_count_greater_equals 2 (box_filter all_boxes (* (* (object_count_greater_equals 1) black) touch_bottom)))

Qualitative example on NLVR

- Boldfaced programs are semantically correct programs and the others are spurious programs.