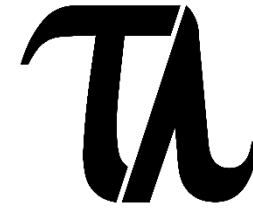# Weakly Supervised Semantic Parsing with Abstract Examples

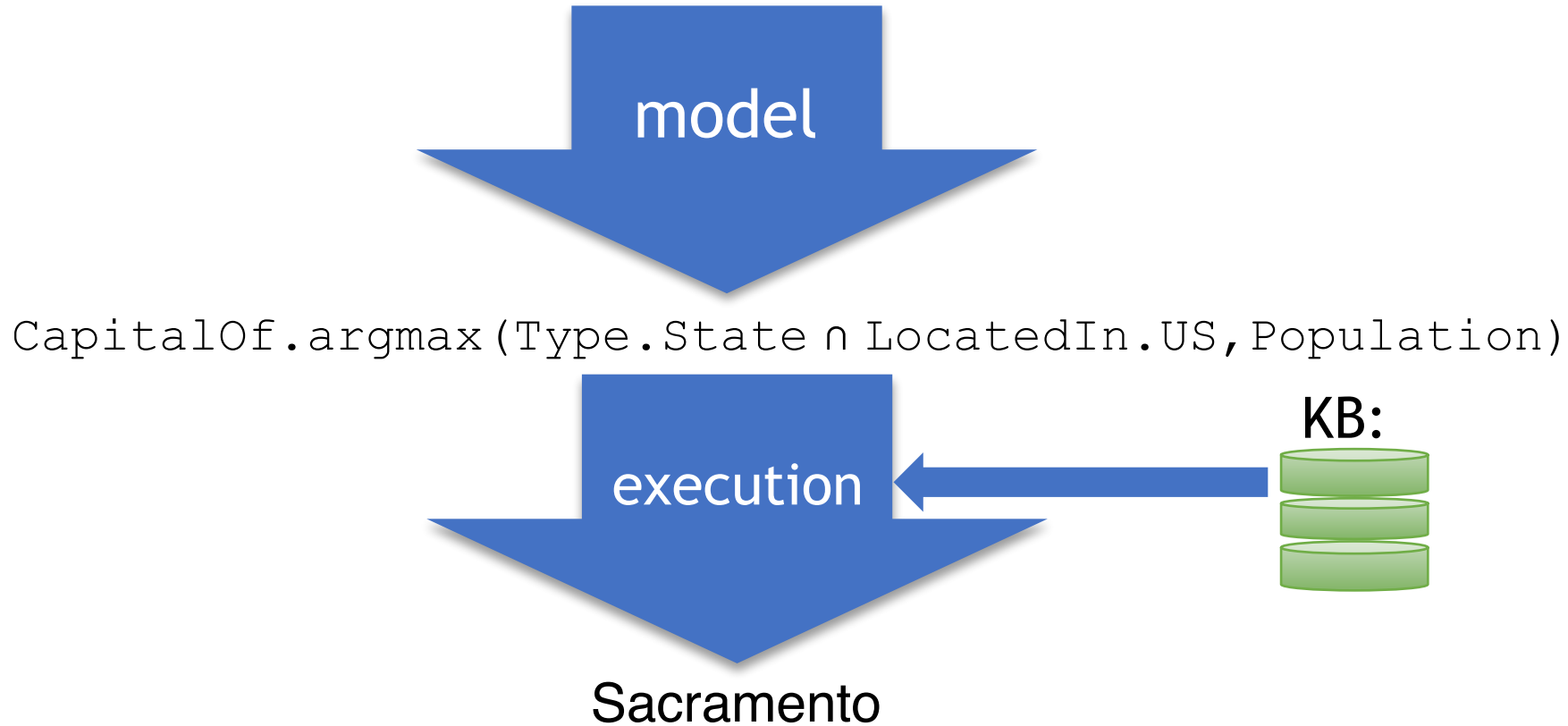**Omer Goldman**, Veronica Latcinnik, Udi Naveh, Amir Globerson, Jonathan Berant

Tel Aviv University

ACL, July 2018

TEL AVIV UNIVERSITY

TAU NLP

Semantic Parsing

What is the capital of the largest US state?

model

`CapitalOf.argmax(Type.State ∩ LocatedIn.US,Population)`

execution

KB:

Sacramento

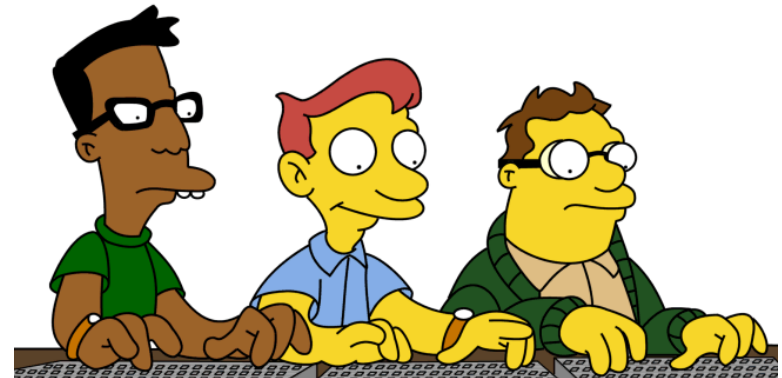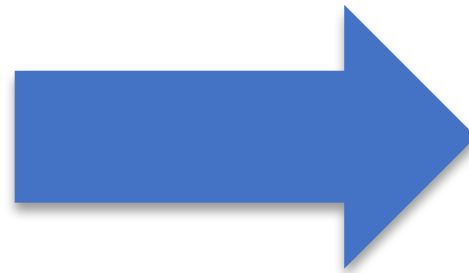# Training with Full Supervision

▶ Training examples:

x:

y: `CapitalOf.argmax(Type.State ∩ LocatedIn.US,Population)`

Training with Weak Supervision

▶ Training examples:

x:
y: Sacramento

Problems with Weak Supervision

▶ Exponential search space
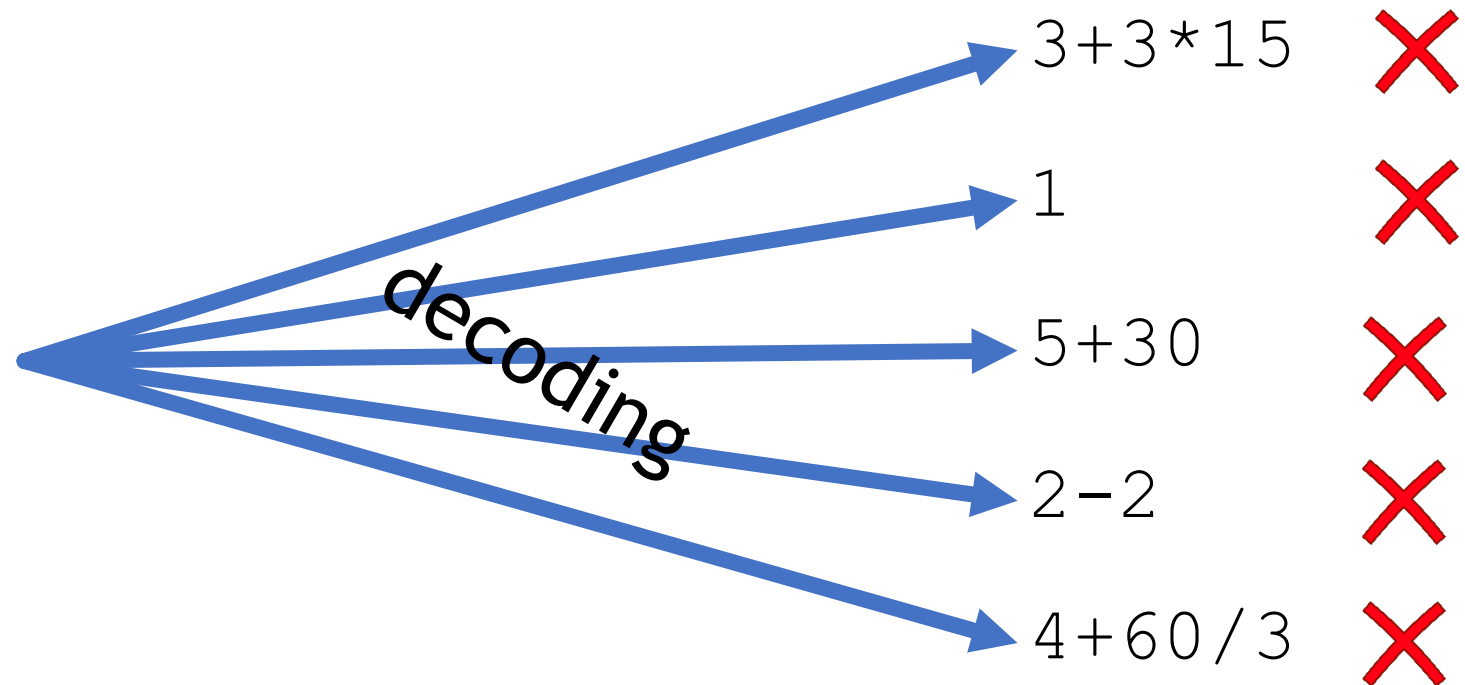
Problems with Weak Supervision

▶ **Spurious programs** (Pasupat and Liang, 2016; Guu et al., 2017)
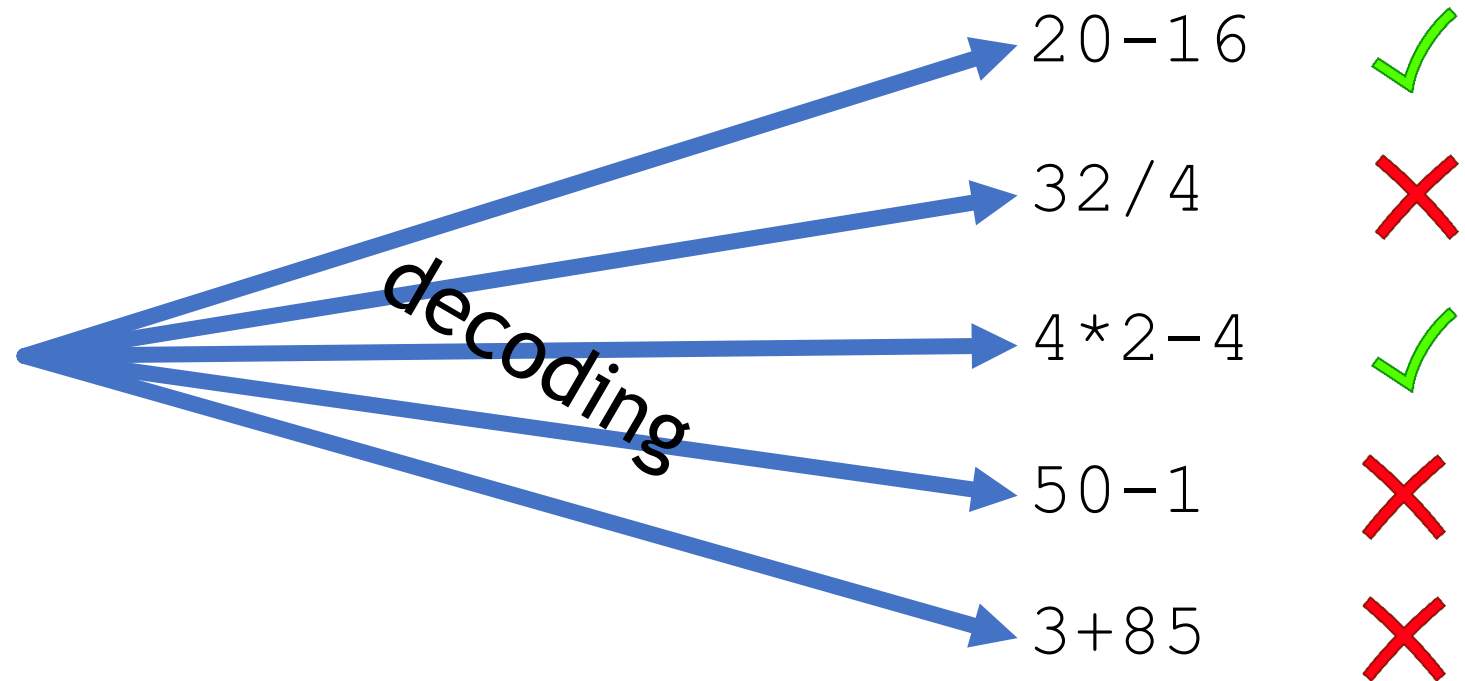
Correct program: `2*2`

CNLVR (Suhr et al., 2017)

I :

k :[[{y_loc: ..., color: 'Black', type: 'square', x_loc: ...
    size: 20}, ...}]]

x : *There is a small yellow item not touching any wall*

y :True

Semantic Parsing

There is a blue square

model

$$exist(filter(ALL\_ITEMS, \lambda x.IsBlue(x) \cap IsSquare(x)))$$

execution

KB:
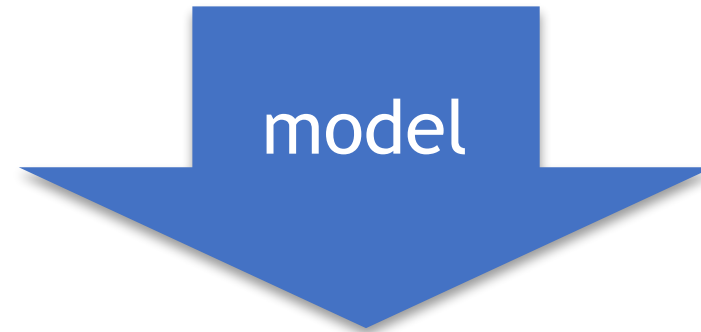
Binary!
50% spurious

True

Insight

There is exactly one black circle not touching the edge

▶ Equal(**1**, (filter(ALL_ITEMS,λx.IsBlack(x) ∩ IsCircle(x) ∩ ¬IsTouchingWall(x)))

▶ GreaterEqual(**3**,(filter(ALL_ITEMS,λx.IsBlue(x) ∩ IsTriangle(x) ∩ ¬IsTouchingWall(x)))

▶ GreaterEqual(**1**,(filter(ALL_ITEMS,λx.IsBlue(x) ∩ IsTriangle(x) ∩ ¬IsTouchingWall(x)))

▶ LessEqual(**3**,(filter(ALL_ITEMS,λx.IsYellow(x) ∩ IsRectangle(x) ∩ ¬IsTouchingWall(x)))

Contributions

There is a $\textcolor{cyan}{yellow}$ $\textcolor{orange}{circle}$

`exist(filter(ALL_ITEMS,`λ`x.`$\textcolor{cyan}{\texttt{IsYellow}}$`(x)`$\wedge$$\textcolor{orange}{\texttt{IsCircle}}$`(x)))`

There is a $\textcolor{cyan}{C-COLOR}$ $\textcolor{orange}{C-SHAPE}$

`exist(filter(ALL_ITEMS,`λ`x.`$\textcolor{cyan}{\texttt{IsC-COLOR}}$`(x)`$\wedge$$\textcolor{orange}{\texttt{IsC-SHAPE}}$`(x)))`

Data augmentation
helps search

Abstract cache
tackles spuriousness

CNLVR improvement:
67.8 → 82.5

Semantic Parsing

Logical Program

x:

z: `exist(filter(ALL_BOXES,λx.ge(3,count(filter(x,λy.IsBlue(y)))))`

Function
Set() → Bool

Constant
Set(Set(Item))

Function
Set() → Int

Variable
Set(Item)

Variable
Item

Function
Set() BoolFunc() → Set()

Model

"There is a yellow triangle" → Encoder – Decoder → Beam search →

```
Exist(Filter(ALL_ITEMS,λx.And(IsYellow(x), IsTriangle(x))))

Exist(Filter(ALL_ITEMS,λx.IsYellow(x)))

GraeterEqual(1,count(Filter(ALL_ITEMS,λx.And(IsYellow(x),IsTriangle(x)))))
```

✓
✗
✓
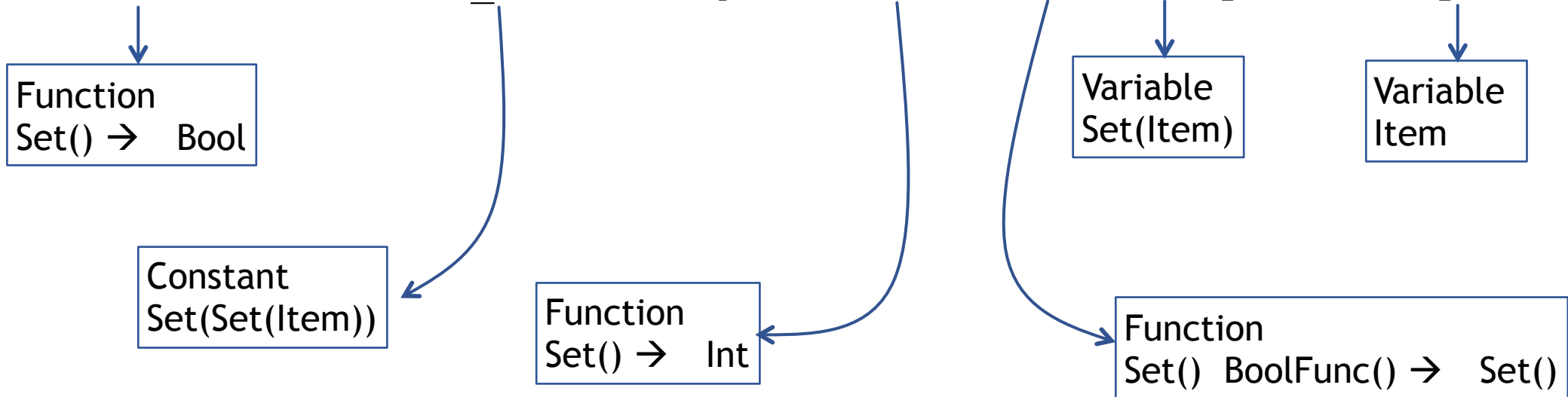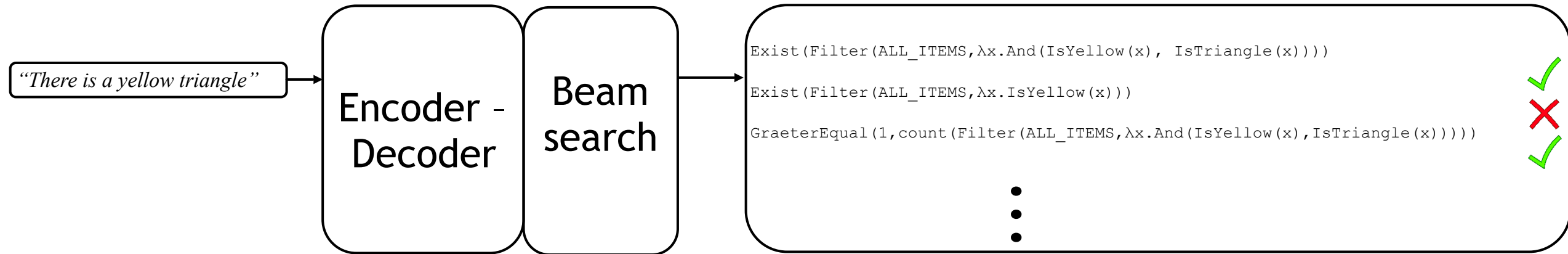
▶ Training maximizes log-likelihood of correct programs

▶ + discriminative re-ranker

Abstract Examples

Abstraction

There is a yellow circle

`exist(filter(ALL_ITEMS,λx.IsYellow(x)∧IsCircle(x)))`

There is a blue square

`exist(filter(ALL_ITEMS,λx.IsBlue(x)∧IsRectangle(x)))`

Utterance	Program	Cluster	#
"yellow"	IsYellow	C-Color	3
"big"	IsBig	C-Size	3
"square"	IsSquare	C-Shape	4
"3"	3	C-Num	2
"exactly"	EqualInt	C-QuantMod	5
"top"	Side.Top	C-Location	2
"above"	GetAbove	C-SpaceRel	6
		Total:	**25**

There is a C − COLOR C − SHAPE

`exist(filter(ALL_ITEMS,λx.IsC-COLOR(x)∧IsC-SHAPE(x)))`

Analysis

3163 CNLVR sentences

- There is............
- One of the.......
- There are........
- Exactly two.....
- There is...........
- In two of.........
- There is..........
- There are........
- There is..........
- One square.....
- There is..........
- One of the.......
- There are........
- There is.........
- Two towers.....
- There are........
- There is..........
- One circle.......
- There is..........
- Last one..........

abstraction

~1300 abstract sentences

- There is..............
- There are...........
- C-Num of.............
- There is...........
- One tower..........
- There are...........
- C-Num C-Shape...
- There is..............
- C-Num towers.....
- Another last.........

▶ ~150 abstract sentences cover 50% of CNLVR.

Abstraction

▶ Data augmentation

▶ Abstract cache

Data Augmentation

There is a *yellow* *circle*

There is a *C − COLOR* *C − SHAPE*

exist(filter(ALL_ITEMS,λx.IsC-COLOR(x)∧IsC-SHAPE(x)))

There is a *blue* *rectangle*
exist(filter(ALL_ITEMS,λx.IsBlue(x)∧IsSquare(x)))

There is a *yellow* *triangle*
exist(filter(ALL_ITEMS,λx.IsYellow(x)∧IsTriangle(x)))

There is a *black* *circle*
exist(filter(ALL_ITEMS,λx.IsBlack(x)∧IsCircle(x)))
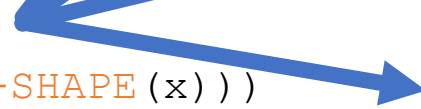
Training Procedure

~100 Abstract examples
(abs. sent., abs. prog.)

| ↓ Generation

~6000 Instantiated examples ──Supervised training──▶ | Supervised model |
(sentence, program)

| ↓ initialization
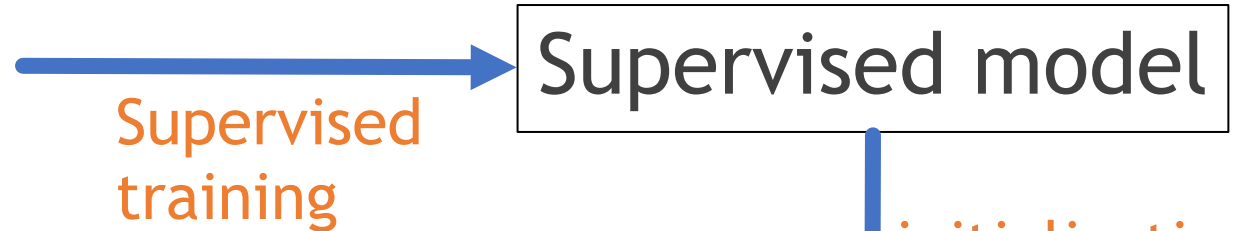
3163 CNLVR training examples ──Weakly supervised training──▶ | Weakly-supervised model |
(sentence, answer)

Abstract Cache

"There is a yellow triangle" —— *decoding* ——▶ **Beam:**

	$t = 0$	$t = 1$	$t = 2$	$t = 3$	$t = 4$	$t = 5$
	Exist	Filter	ALL_BOXES	λx.And	Count	Filter
	Exist	Filter	ALL_ITEMS	λx.And	IsYellow	IsYellow
	GreaterThan	1	Count	Filter	ALL_BOXES	λx.Equal
	Equal	2	Count	Filter	ALL_BOXES	λx.Equal
	Exist	Filter	ALL_ITEMS	λx.And	IsYellow	IsTriangle
	GreaterEqual	1	Count	Filter	ALL_ITEMS	λx.And

Exist	Filter	ALL_ITEMS	λx.And	IsYellow	IsTriangle
GreaterEqual	1	Count	Filter	ALL_ITEMS	λx.And

abstraction

"There is a C-Color C-Shape"

retrieval

CACHE

Exist(Filter(ALL_ITEMS,λx.And(IsC-COLOR(x),IsC-SHAPE(x))))	95%
GreaterEqual(1,count(Filter(ALL_ITEMS,λx.And(IsC-COLOR(x),IsC-SHAPE(x)))))	90%
Exist(Filter(ALL_ITEMS,λx.IsCOLOR(x)))	30%

Reward Tying

I :

k :[[{y_loc: ..., color: 'Black', type: 'square', x_loc: ...
size: 20}, ...}]]

x :*There is a small yellow item not touching any wall*

50% spurious ⟶ y :True

Reward Tying

I :

k :[[{y_loc: ..., color: 'Black', ty
size: 20}, ...}]]

x : *There is a small yellow item*

y :True

I :

k :[[{y_loc: ..., color: 'Black', typ
size: 20}, ...}]]

x : *There is a small yellow item*

y :False

I :

k :[[{y_loc: ..., color: 'Black', ty
size: 20}, ...}]]

x : *There is a small yellow item*

y :False

I :

k :[[{y_loc: ..., color: 'Black', type: 'square', x_loc: ...
size: 20}, ...}]]

x : *There is a small yellow item not touching any wall*

y :True

6.25% spurious

z :Exist(Filter(ALL_ITEMS, λx.And(And(IsYellow(x),
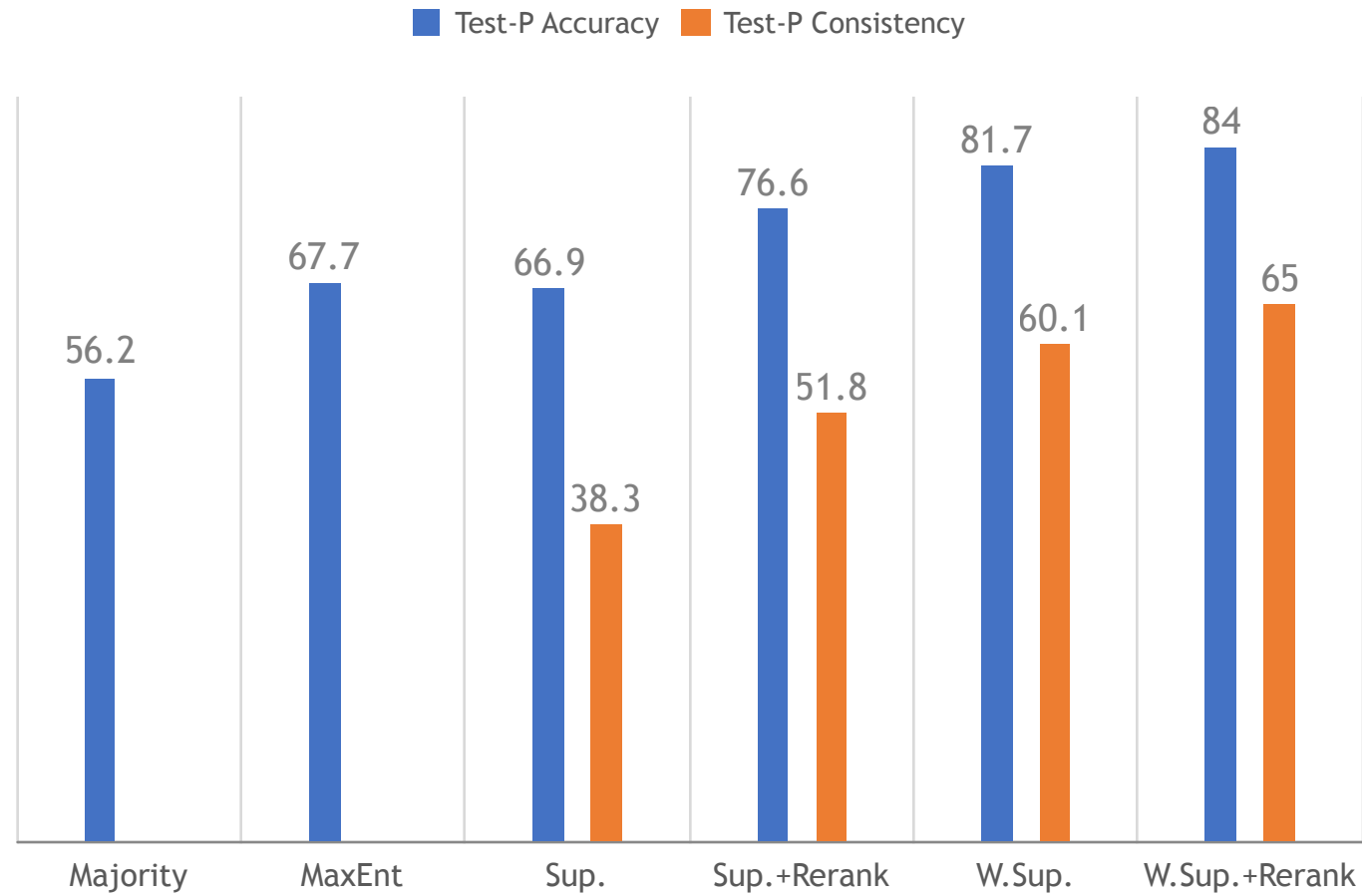IsSmall(x)), Not(IsTouchingWall(x, Side.Any)))))

Results

Models

▶ Majority label (**True**)

▶ Max Entropy classifier on extracted features

Baselines (Suhr et al., 2017)

▶ Supervised trained model (+Re-ranker)

▶ Weakly supervised trained model (+Re-ranker)

Results - Public test set

Legend: ■ Test-P Accuracy ■ Test-P Consistency

Method	Test-P Accuracy	Test-P Consistency
Majority	56.2	
MaxEnt	67.7	
Sup.	66.9	38.3
Sup.+Rerank	76.6	51.8
W.Sup.	81.7	60.1
W.Sup.+Rerank	84	65

Ablations

No data
mentation

ract weakly
rvised parser

Ablations

Data augmentation addition

- Dev Accuracy
- Dev Consistency

	-Abstraction	-Data augment.	-Beam cache	W.Sup.+Rerank
Dev Accuracy	58.2	71.4	77.2	85.7
Dev Consistency	7.1	41.2	56.1	67.4

Cache addition
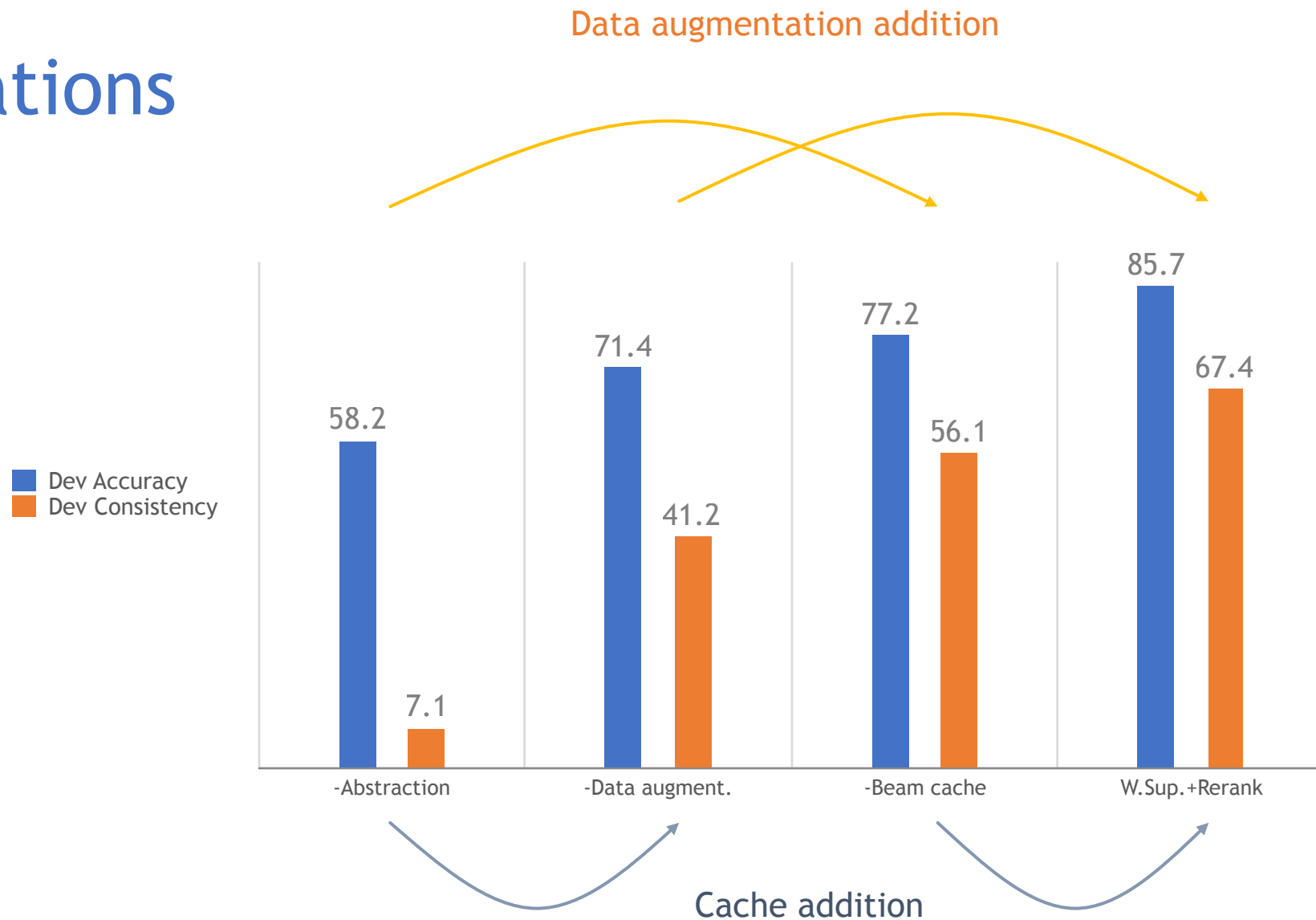
Conclusions

Conclusions

Closed domains

Weak supervision

Abstraction ➡ **Better training**

Facilitate information sharing
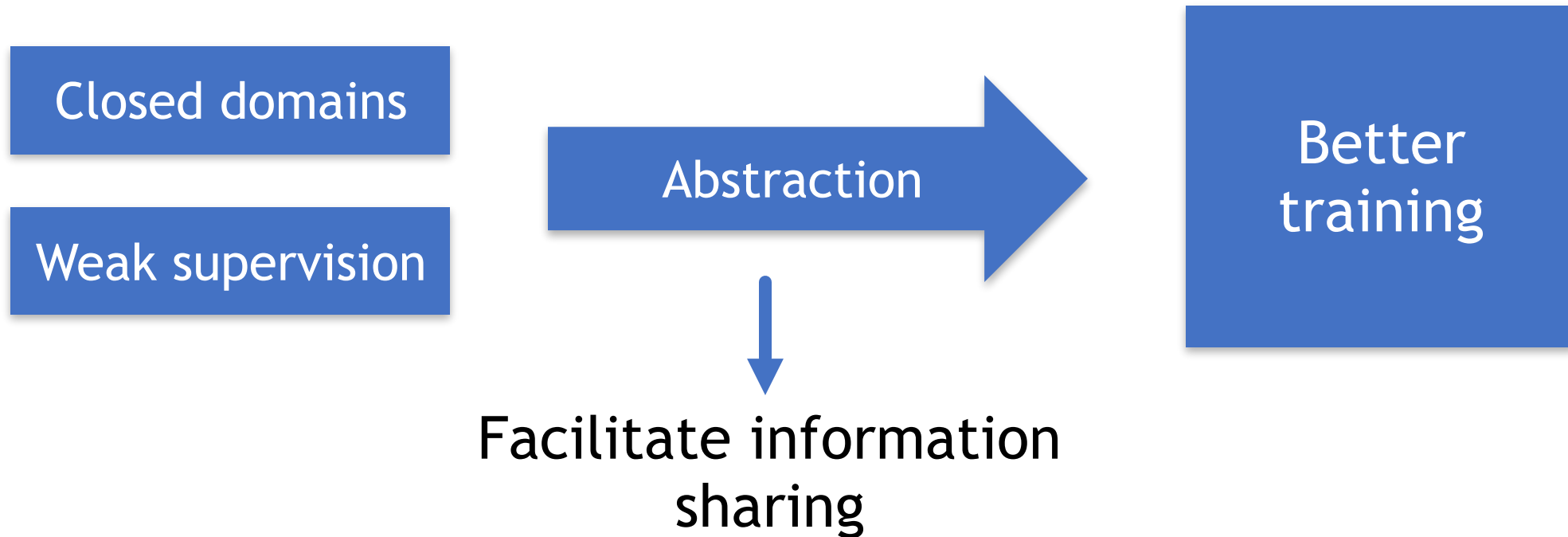
▶ Similar ideas in: Dong and Lapata (2018) and Zhang et al. (2017)

▶ Automation would be useful

Thank you

https://github.com/udiNaveh/nlvr_tau_nlp_final_proj