Iteratively Learning Data Transformation Programs from Examples

Bo Wu

Ph.D. defense

2015-10-21

Agenda

- **Introduction**
- Previous work
- Our approach
  - Learning conditional statements
  - Synthesizing branch transformation programs
  - Maximize user correctness with minimal effort
- Related work
- Conclusion and future work

Programming by example

Accession	Credit	Dimensions	Medium	Name
01.2	Gift of the artist	5.25 in HIGH x 9.375 in WIDE	Oil on canvas	John Mix Stanley
05.411	Gift of James L. Edison	20 in HIGH x 24 in WIDE	Oil on canvas	Mortimer L. Smith
06.1	Gift of the artist	Image: 20.5 in. HIGH x 17.5 in. WIDE	Oil on canvas	Theodore Scott Dabo
06.2	Gift of the artist	9.75 in|16 in HIGH x 13.75 in|19.5 in WIDE	Oil on canvas	Leon Dabo
…				
09.8	Gift of the artist	12 in|14 in HIGH x 16 in|18 in WIDE	Oil on canvas	Gari Melchers

Programming by Example

	Raw Value	Target Value
R1	5.25 in HIGH x 9.375 in WIDE	9.375
R2	20 in HIGH x 24 in WIDE	**24**
R3	9.75 in|16 in HIGH x 13.75 in|19.5 in WIDE	null 19.5
R4	Image: 20.5 in. HIGH x 17.5 in. WIDE	17.5 17.5
	. . .	
R5	12 in|14 in HIGH x 16 in|18 in WIDE	null 18

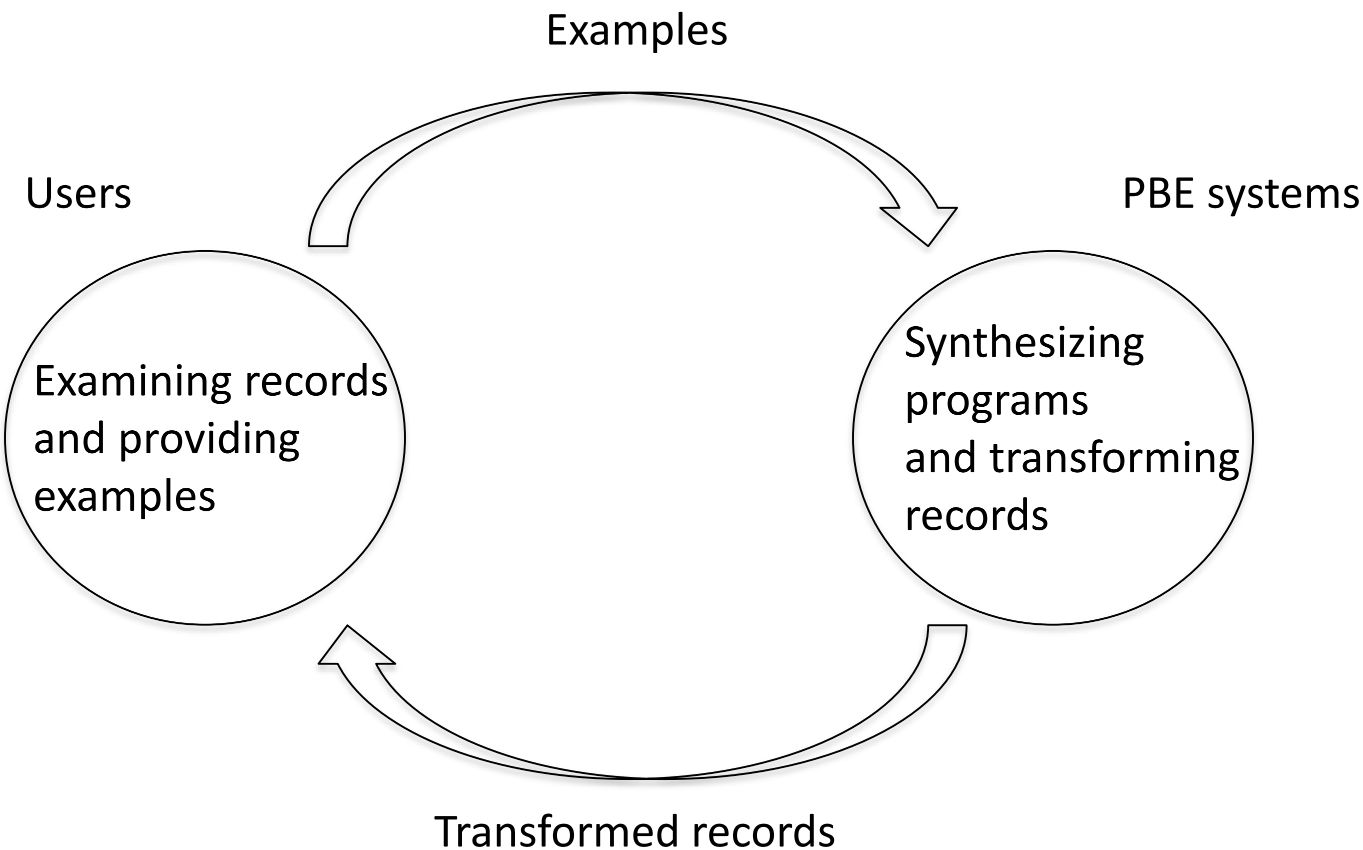
Challenges

- Various formats and few examples
- Stringent time limits
- Verifying the correctness on large datasets

Research problem

Enabling PBE approaches to efficiently generate correct transformation programs for large datasets with multiple formats using minimal user effort

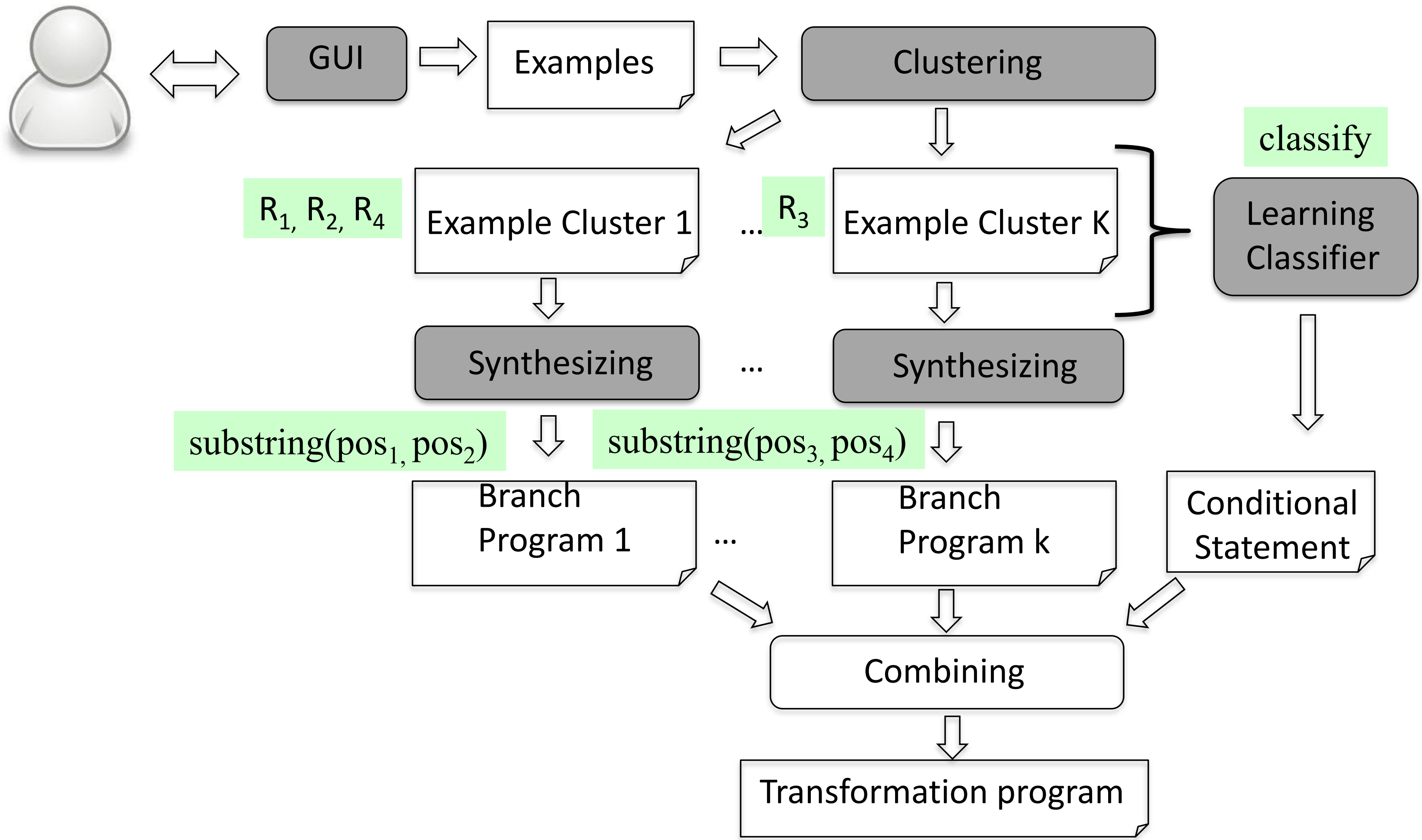
Iterative Transformation



Agenda

- Introduction
- **Previous work**
- Our approach
  - Learning conditional statements
  - Synthesizing branch transformation programs
  - Maximize user correctness with minimal effort
- Related work
- Conclusion and future work

R_1	5.25 in HIGH x 9.375 in WIDE	9.375
R_2	20 in HIGH x 24 in WIDE	24
R_3	9.75 in|16 in HIGH x 13.75 in|19.5 in WIDE	19.5
R_4	Image: 20.5 in. HIGH x 17.5 in. WIDE	17.5



Transformation Program

BNK: blankspace
NUM([0-9]+): 98
UWRD([A-Z]): I
LWRD([a-z]+): mage
WORD([a-zA-Z]+): Image
START:
END:
VBAR: |
…

Segment program:
return a substring

Position program:
return a position in the input

Conditional statement

Branch transformation program

Branch transformation program

Transform(*value*)

switch (classify(value)) :

case $format_1$:

$pos_1 = value.indexOf(BNK,$ NUM, -1)

$pos_2 = value.indexOf(NUM,$ BNK, 2)

output=value.substr(pos_1, pos_2)

case $format_2$:

$pos_3 = value.indexOf($"|", NUM, 2)

$pos_4 = value.indexOf(NUM, BNK,$ -1)

output=value.substr(pos_3, pos_4)

return output

9.75 in|16 in HIGH x 13.75 in|19.5 in WIDE➔ 19.5

Creating Hypothesis Spaces

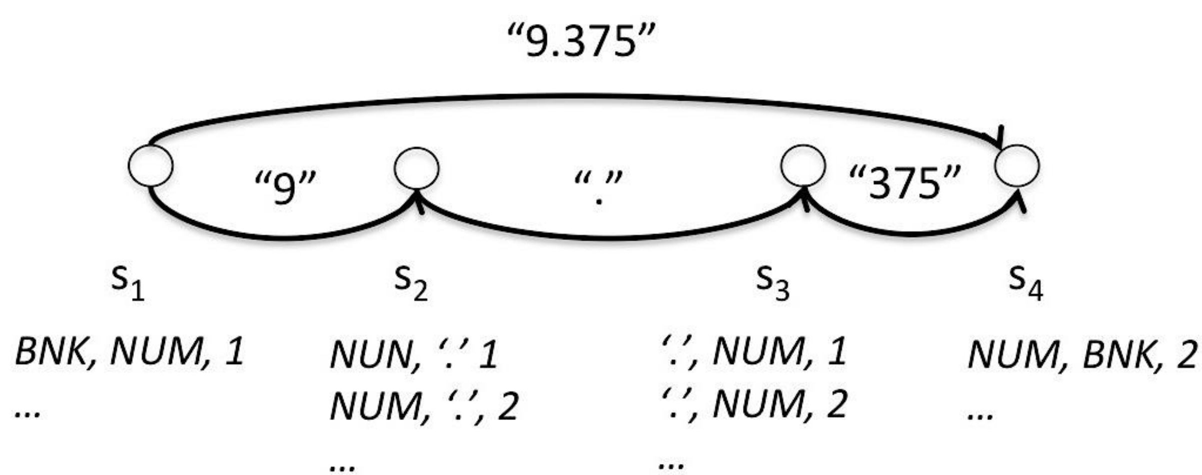
- Create traces

Traces: *A trace here defines how the output string is constructed from a specific set of substrings from the input string.*

Original: 5.25 in HIGH x 9.375 in WIDE

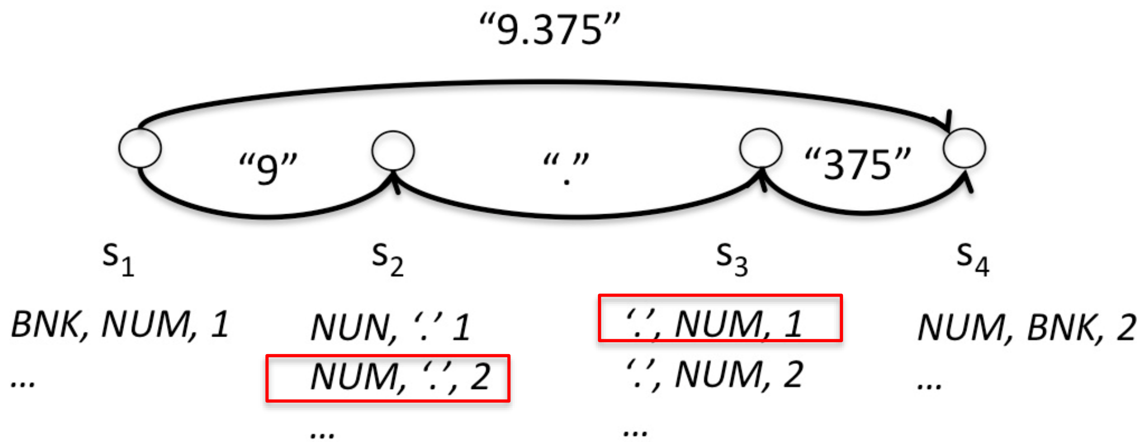
Target: 9.375

- Derive hypothesis spaces



Generating Branch Programs

- Generate programs from hypothesis space
  - Generate-and-test



  - Generate simpler programs first

Programs with one segment programs

earlier than

Programs with three segment programs

Learning Conditional Statements

- Cluster examples

R_1	5.25 in HIGH x 9.375 in WIDE	9.375
R_2	20 in HIGH x 24 in WIDE	24
R_4	Image: 20.5 in. HIGH x 17.5 in. WIDE	17.5

Cluster1-format$_1$

R_3	9.75 in|16 in HIGH x 13.75 in|19.5 in WIDE	19.5

Cluster2-format$_2$

- Learn a multiclass classifier
  - Recognize the format of the inputs

R_5	Image: 20.5 in. HIGH x 17.5 in. WIDE	format$_1$
R_6	12 in|14 in HIGH x 16 in|18 in WIDE	format$_2$

Agenda

- Introduction
- Previous work
- Our approach
  - Learning conditional statements
  - Synthesizing branch transformation programs
  - Maximize user correctness with minimal effort
- Related work
- Conclusion and future work

Our contributions

- Efficiently learning accurate conditional statements [DINA, 2014]
- Efficiently synthesizing branch transformation programs [IJCAI, 2015]
- Maximizing the user correctness with minimal user effort [IUI, 2014; IUI, 2016(submitted)]

Agenda

- Introduction
- Previous work
- Our approach
  - Learning conditional statements
  - Synthesizing branch transformation programs
  - Maximize user correctness with minimal effort
- Related work
- Conclusion and future work

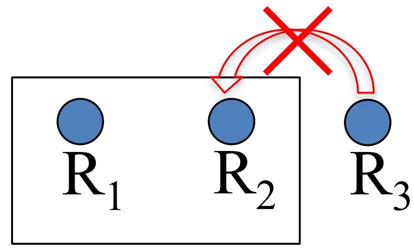
Motivation

- Example clustering is time consuming
  - Many ways (2^n) to cluster the examples
  - Many examples are not compatible

R_1	5.25 in HIGH x 9.375 in WIDE	9.375
R_2	20 in HIGH x 24 in WIDE	24
R_3	9.75 in|16 in HIGH x 13.75 in|19.5 in WIDE	19.5

  - Verifying compatibility is expensive
- Learned conditional statement is not accurate
  - Users are willing to provide a few examples

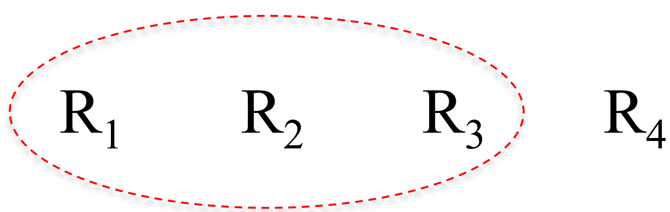
Utilizing known compatibilities



After providing
3 examples

R_1	5.25 in HIGH x 9.375 in WIDE	9.375
R_2	20 in HIGH x 24 in WIDE	24
R_3	9.75 in|16 in HIGH x 13.75 in|19.5 in WIDE	19.5

After providing
4 examples



Constraints

- Two types of constraints:
  - Cannot-merge constraints:

Ex:

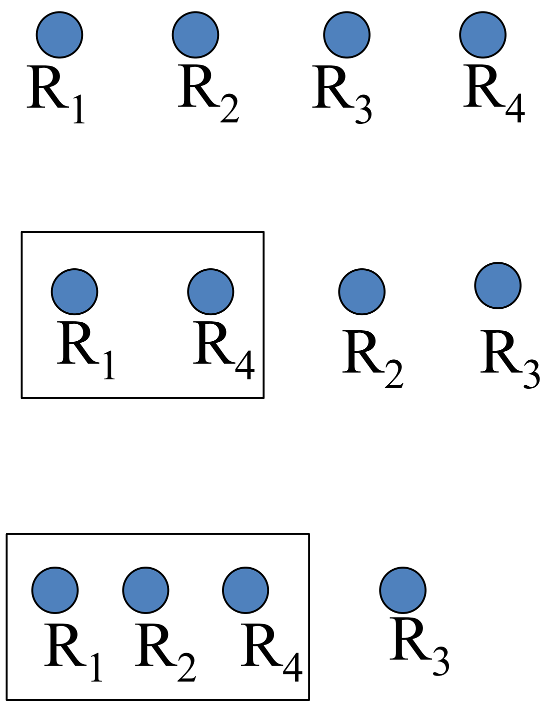
5.25 in HIGH x 9.375 in WIDE	9.375
9.75 in|16 in HIGH x 13.75 in|19.5 in WIDE	13.75
20 in HIGH x 24 in WIDE	24

  - Must-merge constraints:

Ex:

5.25 in HIGH x 9.375 in WIDE	9.375
20 in HIGH x 24 in WIDE	24

Constrained Agglomerative Clustering



Distance between clusters (p_i and p_j) :

$$d(p_i, p_j) = \min\{d(e_x, e_y) \mid e_x \in p_i, e_y \in p_j\}$$

R_1	5.25 in HIGH x 9.375 in WIDE
R_4	Image: 20.5 in. HIGH x 17.5 in. WIDE

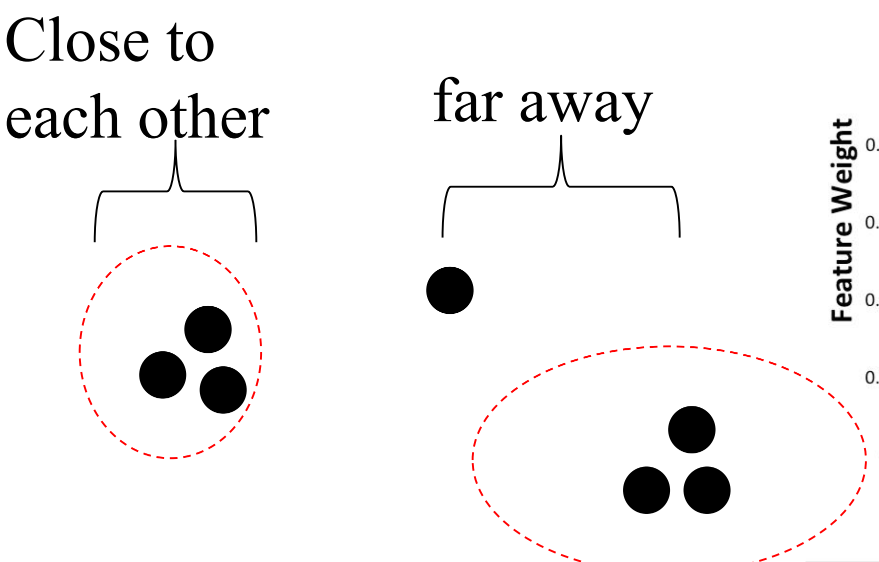
R_1	5.25 in HIGH x 9.375 in WIDE
R_2	20 in HIGH x 24 in WIDE
R_3	9.75 in|16 in HIGH x 13.75 in|19.5 in WIDE

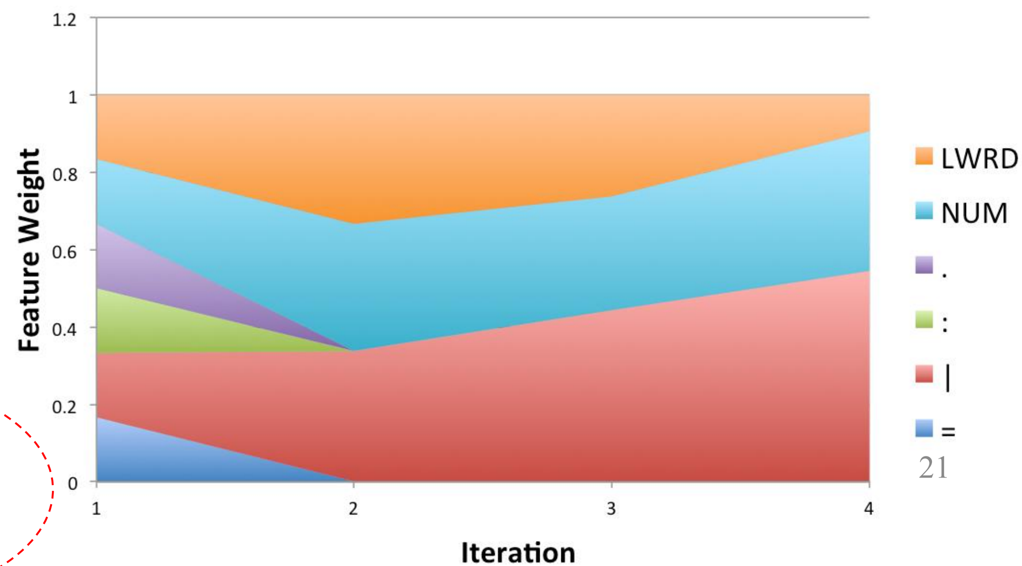
Distance Metric Learning

- Distance metric learning

$$d(x,y) = \|x - y\|_w = \sqrt{\sum_i w_i (x_i - y_i)^2}$$

- Objective function





Utilizing Unlabeled data

Partition 1

Examples	5.25 in HIGH x 9.375 in WIDE	9.375
	20 in HIGH x 24 in WIDE	24
	Image: 20.5 in. HIGH x 17.5 in. WIDE	17.5

Unlabeled	26 in. HIGH x 23 in. WIDE
	19.75 in HIGH x 22.75 in WIDE x 0.25 in DEEP
	33.5 in HIGH x 39 in WIDE
	...

Partition 2

Examples	9.75 in|16 in HIGH x 13.75 in|19.5 in WIDE	13.75

Unlabeled	12 in|14 in HIGH x 16 in|18 in WIDE
	20.25 in|19.75 in HIGH x 15.75 in|15.875 in WIDE
	55 in HIGH x 46 in|290 in WIDE
	...

Evaluation

- Dataset:
  - 30 editing scenarios collected from student course projects

Avg records	Min formats	Max formats	Avg formats
350	2	12	4.4

- Methods:
  - **SP**
    - The state-of-the-art approach that uses compatibility score to select partitions to merge
  - SPIC
    - Utilize previous constraints besides using compatibility score
  - DP
    - Learn distance metric
  - DPIC
    - Utilize previous constraints besides learning distance metric
  - **DPICED**
    - Our approach in this paper

Results

Time and Examples:

	Total Time (seconds)	Examples
DPICED	3.9	5.4
DPIC	6.4	6.8
DP	8.3	6.8
SPIC	21.3	6.8
SP	26.5	6.9

Agenda

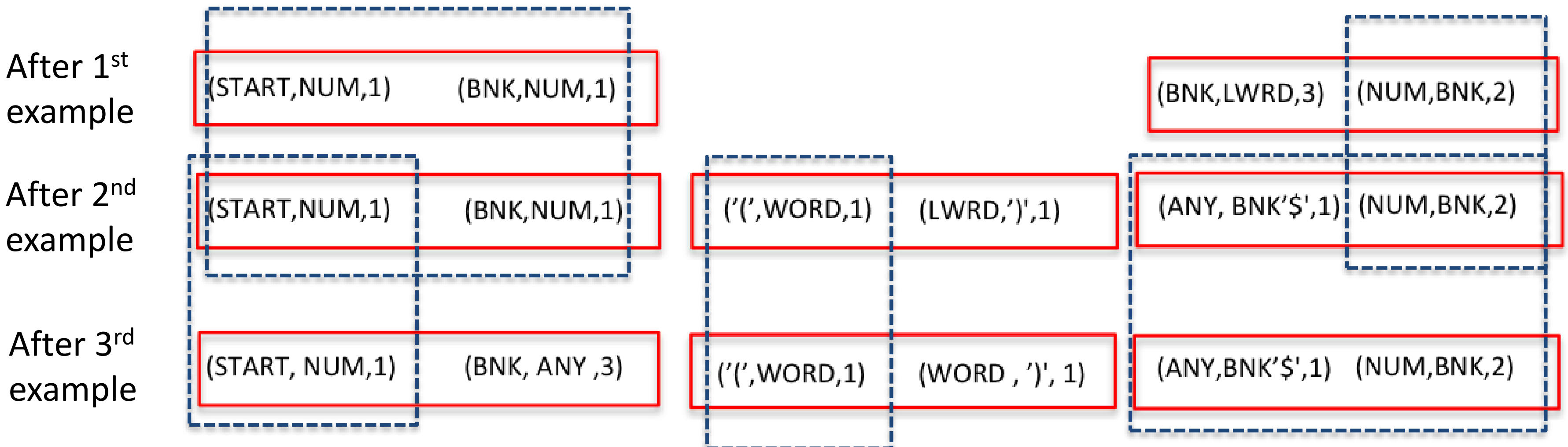
- Introduction
- Previous work
- Our approach
  - Learning conditional statements
  - Synthesizing branch transformation programs
  - Maximize user correctness with minimal effort
- Related work
- Conclusion and future work

Learning Transformation Programs by Example

Input Data	Target Data
2000 Ford Expedition 11k runs great los angeles $4900 (los angeles)	2000 Ford Expedition los angeles $4900
1998 Honda Civic 12k miles s. Auto. - $3800 (Arcadia)	1998 Honda Civic Arcadia $3800
2008 Mitsubishi Galant ES $7500 (Sylmar CA) pic	2008 Mitsubishi Galant Sylmar CA $7500
1996 Isuzu Trooper 14k clean title west covina $999 (west covina) pic	1996 Isuzu Trooper west covina $999
...	...

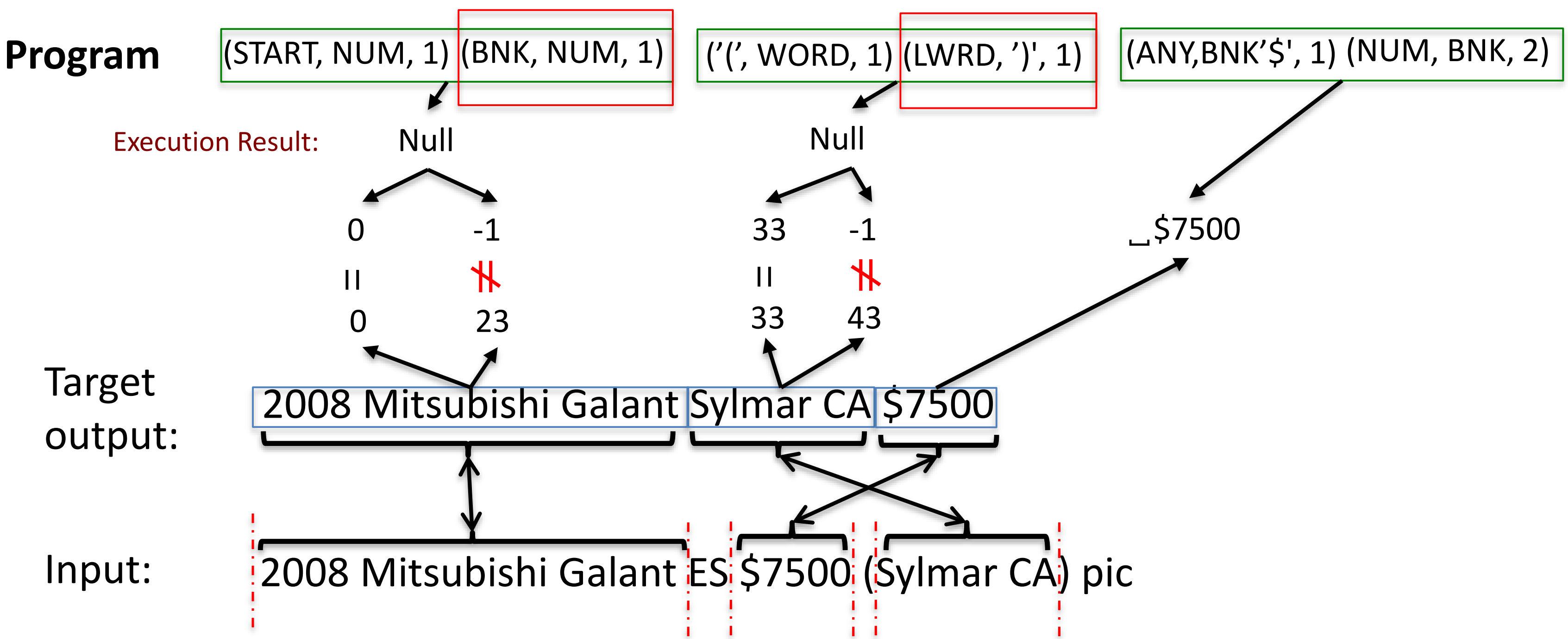
Time complexity is **exponential** in the **number** and a **high polynomial** in the **length** of examples

Reuse subprograms



Identify incorrect subprograms

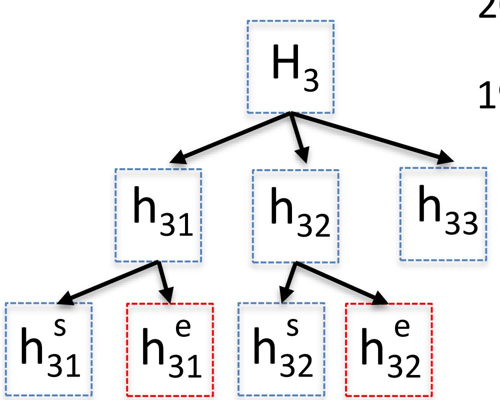
Input	Output
2000 Ford Expedition 11k runs great los angeles $4900 (los angeles)	2000 Ford Expedition los angeles $4900
1998 Honda Civic 12k miles s. Auto. - $3800 (Arcadia)	1998 Honda Civic Arcadia $3800



Update hypothesis spaces

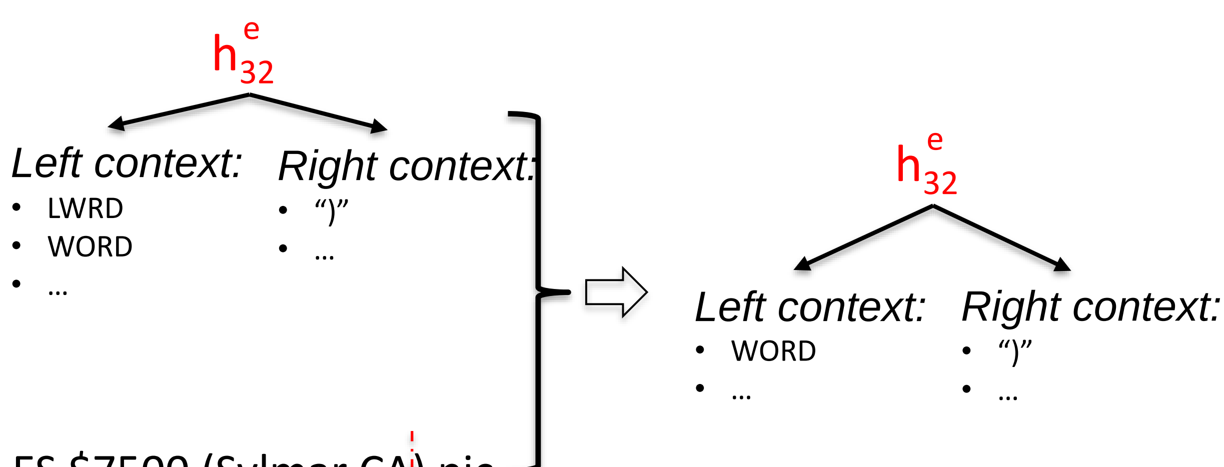
Program (START, NUM, 1) (BNK, NUM, 1) ('(', WORD, 1) (LWRD, ')', 1) (ANY,BNK'$', 1)

Hypothesis H_3



2000 Ford Expedition 11k runs great los angeles $4900 (los angeles)

1998 Honda Civic 12k miles s. Auto. - $3800 (Arcadia)



2008 Mitsubishi Galant ES $7500 (Sylmar CA) pic

Evaluation

- Dataset
  - **D1**: 17 scenarios used in (Lin et al., 2014)
    - 5 records per scenario
  - **D2**: 30 scenarios collected from student data integration projects
    - about 350 records per scenario
  - **D3**: synthetic dataset
    - designed to evaluate scale-up
- Alternative approaches
  - **Our implementation of Gulwani's approach**: (Gulwani, 2011)
  - **Metagol:** (Lin et al., 2014)
- Metric
  - Time (in **seconds**) to generate a transformation program

Program generation time comparisons

Table: time (in seconds) to generate programs on D1 and D2 datasets

		Min	Max	Avg	Median
D1	IPBE	0	5	0.34	0
	Gulwani's approach	0	8	0.59	0
	Metagol	0	213.93	55.1	0.14
D2	IPBE	0	1.28	0.20	0
	Gulwani's approach	0	17.95	4.02	0.33
	Metagol	∼	∼	∼	∼

D3

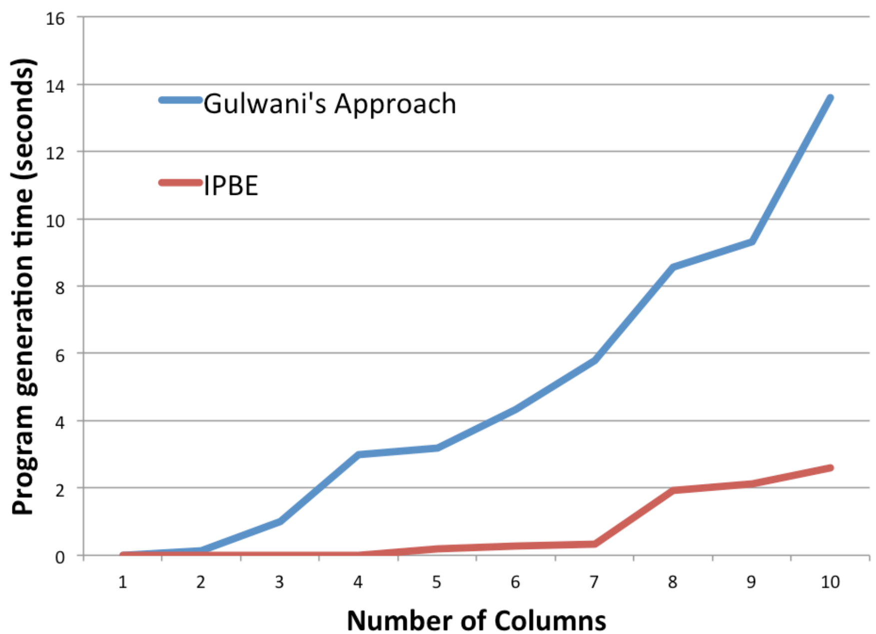


Figure: scalability test on D3

Agenda

- Introduction
- Previous work
- Our approach
  - Learning conditional statements
  - Synthesizing branch transformation programs
  - Maximize user correctness with minimal effort
- Related work
- Conclusion and future work

Motivation

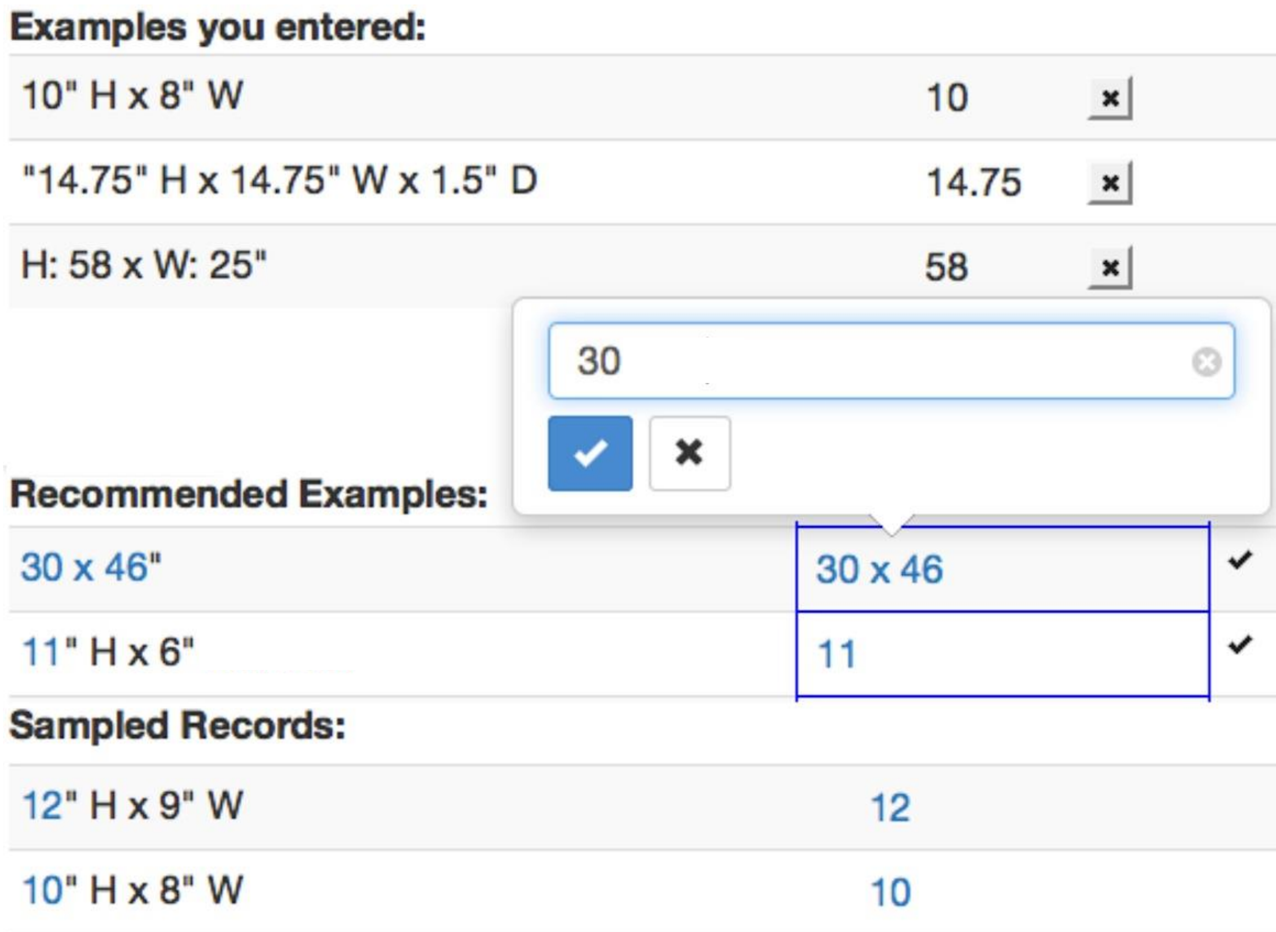
- Thousands of records in datasets
- Various transformation scenarios

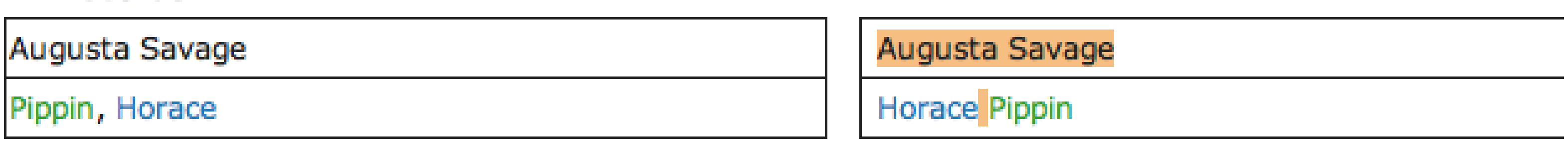
Raw (Input)	Transformed (Output)
300 or more	3
Between 100 and 299	2
Fewer than 100	3
…	…

Raw (Input)	Transformed (Output)
10” x 8	10
26" H x 24" W x 12.5”	26
3 x 6"	3 x 6
…	…

- Overconfident users

User Interface





Learning from various past results

Raw	Transformed
26" H x 24" W x 12.5	26
Framed at 21.75" H x 24.25” W	21
12" H x 9"	12
…	

Raw	Transformed
Ravage 2099#24 (November, 1994)	November, 1994
Gambit III#1 (September, 1997)	September, 1997
(comic) Spidey Super Stories#12/2 (September, 1975)	comic
…	

Examples

Incorrect records

Correct records

…

Approach Overview

Entire dataset

Raw	Transformed
10“ H x 8” W	10
H: 58 x W:25”	58
12”H x 9”W	12
11”H x 6”	11
…	…
30 x 46”	30 x 46

Random Sampling

Sampled records

Raw	Transformed
10“ H x 8” W	10
11”H x 6”	11
…	…
30 x 46”	30 x 46

Verifying records

Raw	Transformed
11”H x 6”	11
30 x 46”	30 x 46
…	…

Sorting and color-coding

Raw	Transformed
30 x 46”	30 x 46
11”H x 6”	11
…	…

Verifying Records

- Recommend records causing runtime errors
  - Records cause the program exit abnormally

Program: (LWRD, ‘)’, 1)

Input: 2008 Mitsubishi Galant ES $7500 (Sylmar CA) pic

- Recommend potentially incorrect records
  - Learn a binary meta-classifier

Ex:

Raw	Transformed
11”H x 6”	11
30 x 46”	30 x 46
…	…

Learning the Meta-classifier

$$F(r) = \mathrm{s}ign(\sum_i w_i * f_i(r)) = \begin{cases} 1, & \text{if r is correct} \\ -1, & \text{if r is incorrect} \end{cases}$$

Learn an ensemble of classifiers using ADABOOST:

(1) Select a f_i from a pool of binary classifiers

(2) Assign weight w_i to f_i

(3) Loop until error below a threshold

Evaluation

Dataset:

30 scenarios

350 records per scenario

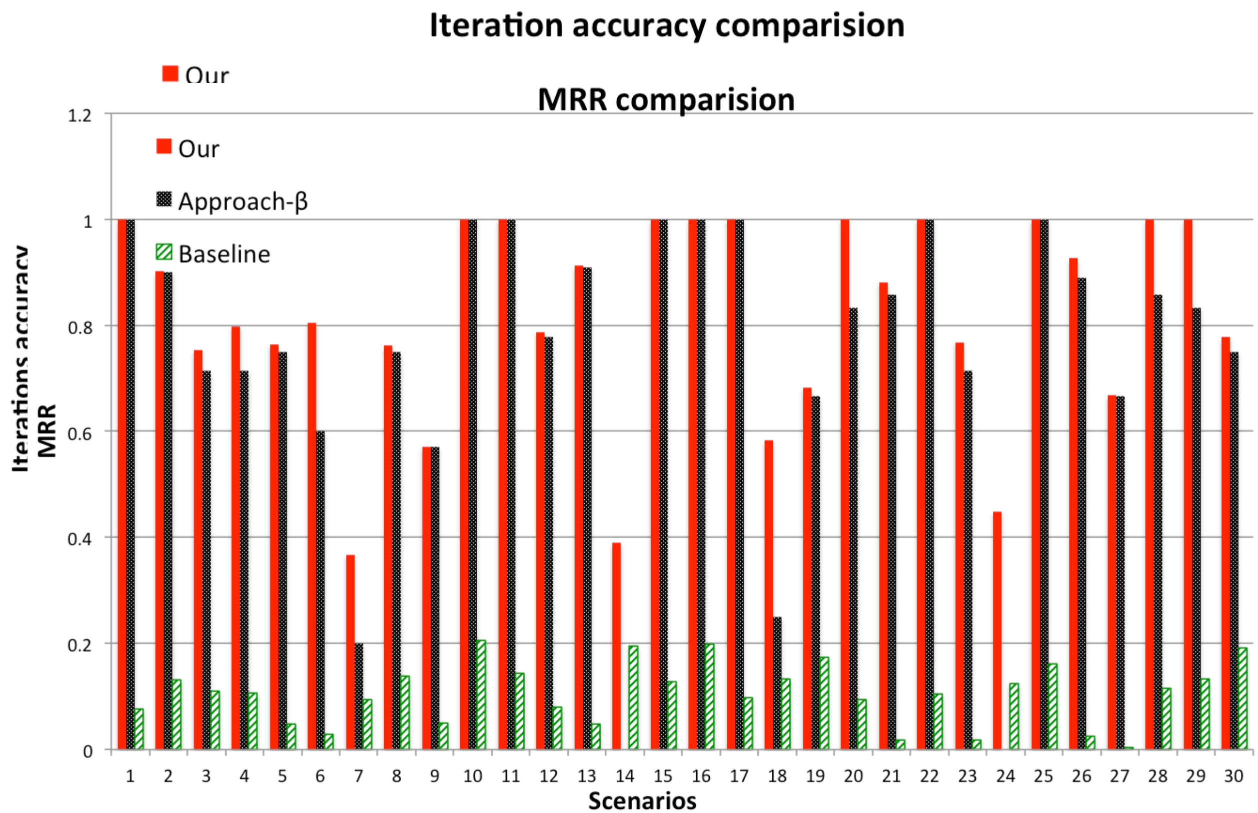
Experiment setup:

- Approach-β
- Baseline

Metrics:

- Iteration correctness
- MRR

$$MRR = \frac{1}{Q}\sum_{i=1}^{Q}\frac{1}{Rank_i}$$



Agenda

- Introduction
- Previous work
- Our approach
  - Learning conditional statements
  - Synthesizing branch transformation programs
  - Maximize user correctness with minimal effort
- Related work
- Conclusion and future work

Related Work

- Approaches not focusing on data transformation
  - Wrapper induction
    - Kushmerick,1997; Hsu and Dung, 1998; Muslea et al., 1999
  - Inductive programming (we learn)
    - Summers, 1977; Kitzelmam and Schmid, 2006;Shaprio, 1981; Muggleton and Lin, 2013
- Approaches not learning program iteratively
    - FlashFill (Gulwani, 2011); SmartPython (Lau, 2001), SmartEdit (Lau, 2001); Singh and Gulwani 2012; Raza et al., 2014; Harris, et al., 2011
  - Approaches learning part of the programs iteratively
    - $Metagol_{DF}$ (Lin et al., 2014); Preleman, et al 2014

Conclusion: contributions

- Enable users to generate complicated programs in real time
- Enable users to work on large datasets
- Improve the performance of other PBE approaches

Conclusion: future work

- Managing user expectation

- Incorporating third-party functions

- Handling user errors

Questions ?