SYPHON 

Constraint-based Learning of Phonological Processes

Shraddha Barke, Rose Kunkel, Eric Meinhardt, Nadia Polikarpova, Eric Bakovic, Leon Bergen

UC San Diego

ENGLISH VERBS PAST TENSE

zip is phonetically [zɪp]
beg is phonetically [bɛg]

[zɪpt] (zipped)

[bɛgd] (begged)

ENGLISH VERBS PAST TENSE

zip is phonetically [zɪp]
beg is phonetically [bɛg]

/zɪp + d/ → [zɪpt]

(zipped)

/bɛg + d/ → [bɛgd]

(begged)

ENGLISH VERBS PAST TENSE

zip is phonetically [zɪp]
beg is phonetically [bɛg]

/zɪp + d/ → [zɪpt]

/bɛg + d/ → [bɛgd]

/d/ → [t] if it occurs after voiceless sounds

RESEARCH PROBLEM

zip is phonetically [zɪp]
beg is phonetically [bɛg]

/zɪp + d/ → [zɪpt]

/bɛg + d/ → [bɛgd]

Automatic Inference of Phonological rules

/d/ → [t] if it occurs after voiceless sounds

WORD FORMS

/zɪp + d/ → [zɪpt]

/bɛg + d/ → [bɛgd]

/stem + suffix/ → [surface form]

WORD FORMS

/zɪp + d/ → [zɪpt]

/bɛg + d/ → [bɛgd]

/stem + suffix/ → [surface form]

WORD FORMS

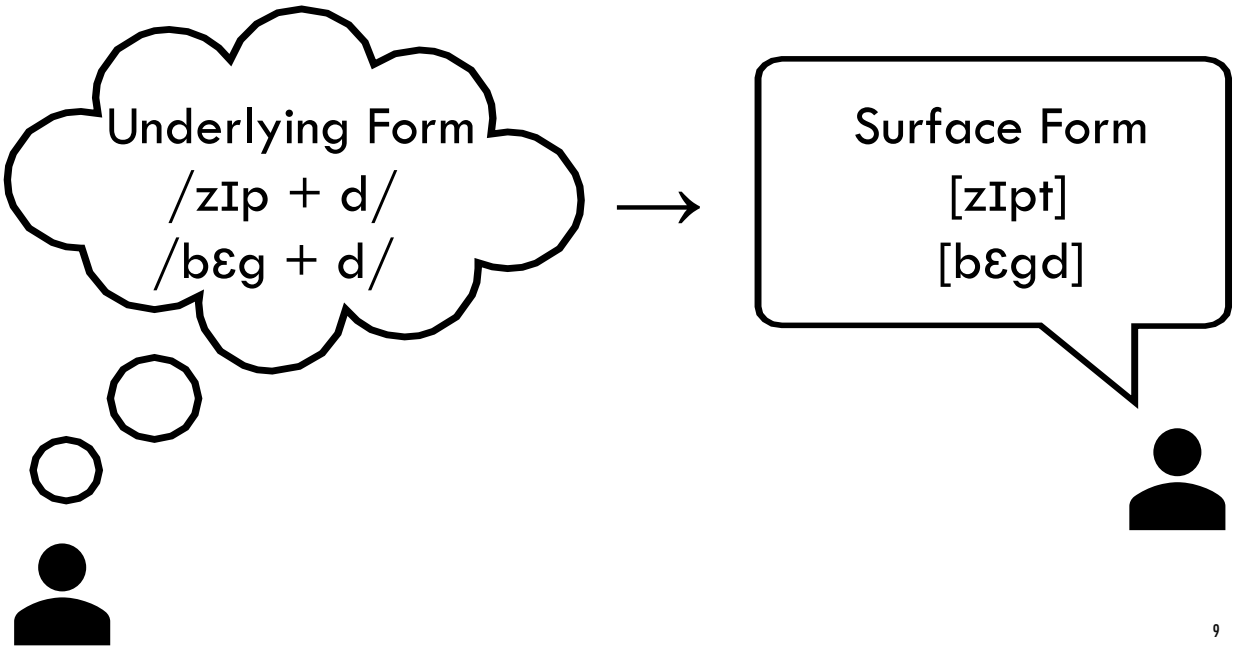
/zɪp + d/ → [zɪpt]

/bɛg + d/ → [bɛgd]

/underlying form/ → [surface form]

PHONOLOGICAL PROCESS

Goal – Infer function from the underlying form to surface form.



PHONOLOGICAL REWRITE RULES

A → B / L _ R

- Any sound that matches A and occurs between sounds that match left context L and right context R will be rewritten to B.

PHONOLOGICAL REWRITE RULES

Surface forms	A → B / L _ R
[zɪpt]	/d/ → [t] / [p] _ ø
[bɛgd]	No change
[zɪps]	/z/ → [s] / [p] _ ø
[bɛgz]	No change

PHONOLOGICAL REWRITE RULES

Surface forms	A → B / L _ R
[zIpt]	/d/ → [t] / [p] _ ø
[zIps]	/z/ → [s] / [p] _ ø

/d/, /z/ → [t] , [s] / [p] _ ø

PHONOLOGICAL REWRITE RULES

Surface forms	A → B / L _ R
[zIpt]	/d/ → [t] / [p] _ ø
[zIps]	/z/ → [s] / [p] _ ø

/d/, /z/ → [t] , [s] / [p] _ ø

voiceless

PHONOLOGICAL REWRITE RULES

Surface forms	A → B / L _ R
[zIpt]	/d/ → [t] / [p] _ ø
[zIps]	/z/ → [s] / [p] _ ø

/d/, /z/ → [t] , [s] / [p] _ ø

[-voice]

PHONOLOGICAL REWRITE RULES

Surface forms	A → B / L _ R
[zIpt]	/d/ → [t] / [p] _ ø
[zIps]	/z/ → [s] / [p] _ ø

/d/, /z/ → [t] , [s] / [p] _ ø

[-sonorant] → [-voice]

PHONOLOGICAL REWRITE RULES

Surface forms	A → B / L _ R
[zIpt]	/d/ → [t] / [p] _ ø
[zIps]	/z/ → [s] / [p] _ ø

/d/, /z/ → [t] , [s] / [p] _ ø

[-sonorant] → [-voice] / [-voice] _ ø

OUTLINE

1. Problem Statement
2. Our Solution
3. Experimental Results

SYPHON

SYNTHESIS OF PHONOLOGICAL RULES

SYPHON

SYPHON

PAST TENSE	PRESENT TENSE
zɪpt	zɪps
bɛgd	bɛgz
rod	roz
lɪvd	lɪvz
æskt	æsks

SYPHON

PAST TENSE	PRESENT TENSE
zɪp + d	zɪp + z
bɛg + d	bɛg + z
ro + d	ro + z
lɪv + d	lɪv + z
æsk + d	æsk + z

PAST TENSE	PRESENT TENSE
zɪpt	zɪps
bɛgd	bɛgz
rod	roz
lɪvd	lɪvz
æskt	æsks

SYPHON

[-sonorant] ⟶ [-voice] / [-voice] _

PAST TENSE	PRESENT TENSE
zɪp + d	zɪp + z
bɛg + d	bɛg + z
ro + d	ro + z
lɪv + d	lɪv + z
æsk + d	æsk + z

PAST TENSE	PRESENT TENSE
zɪpt	zɪps
bɛgd	bɛgz
rod	roz
lɪvd	lɪvz
æskt	æsks

RESEARCH GOALS

1. Interpretability - Inferred rules should be human readable
2. Data efficiency – Few shot learning
3. Interactivity - Inference at interactive speeds

INTERPRETABILITY AND INTERACTIVITY

MOTIVATION

Phonologists spend lot of time manually analyzing language datasets

INTERPRETABILITY AND INTERACTIVITY

MOTIVATION

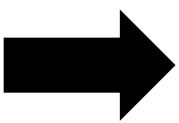
Phonologists spend lot of time manually analyzing language datasets

OUR SOLUTION

Automated approach to phonological rule inference

DATA EFFICIENCY

Large amounts of data unavailable



Few Shot Learning

OUTLINE

1. Problem Statement
2. Our Solution
3. Experimental Results

OBJECTIVE FUNCTION

$$\mathrm{F}(R, U, X) = \begin{cases} \mathrm{length}(R) + \mathrm{fit}(R, U, X) & \text{if } \mathrm{consistent}(R, U, X) \\ \infty & \text{otherwise} \end{cases}$$

R - Rules
U - Underlying forms
X - Surface forms

OBJECTIVE FUNCTION

Correctness constraint

$$\mathrm{F}(R,U,X) = \begin{cases} \mathrm{length}(R) + \mathrm{fit}(R,U,X) & \text{if } \mathrm{consistent}(R,U,X) \\ \infty & \text{otherwise} \end{cases}$$

R - Rules
U - Underlying forms
X - Surface forms

OBJECTIVE FUNCTION

Simplicity constraint

Correctness constraint

$$\mathrm{F}(R,U,X) = \begin{cases} \mathrm{length}(R) + \mathrm{fit}(R,U,X) & \text{if } \mathrm{consistent}(R,U,X) \\ \infty & \text{otherwise} \end{cases}$$

R - Rules
U - Underlying forms
X - Surface forms

OBJECTIVE FUNCTION

Simplicity constraint

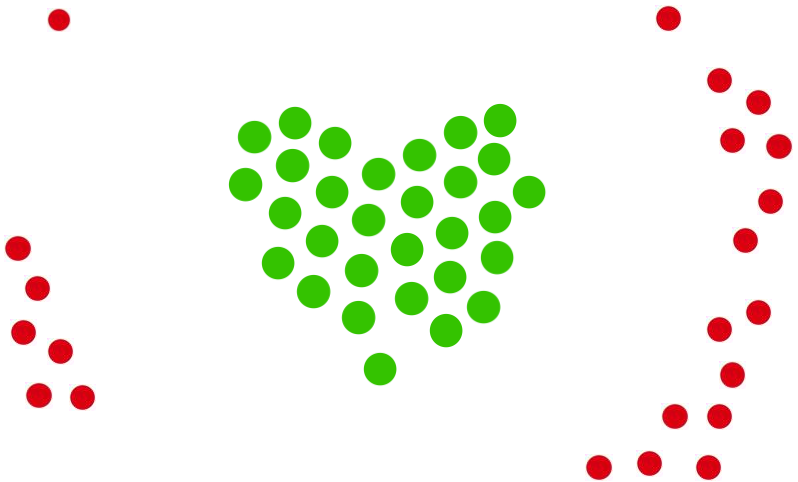
Correctness constraint

$$\mathrm{F}(R,U,X) = \begin{cases} \mathrm{length}(R) + \mathrm{fit}(R,U,X) & \text{if } \mathrm{consistent}(R,U,X) \\ \infty & \text{otherwise} \end{cases}$$

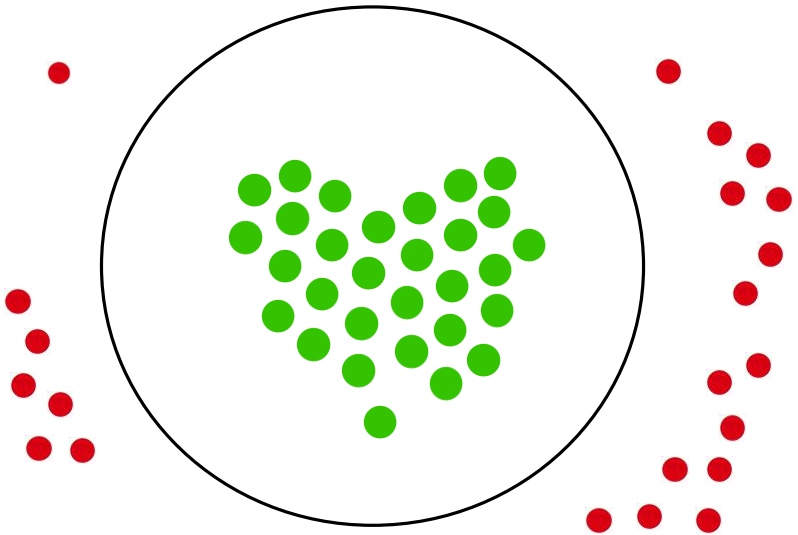
Specificity constraint

R - Rules
U - Underlying forms
X - Surface forms

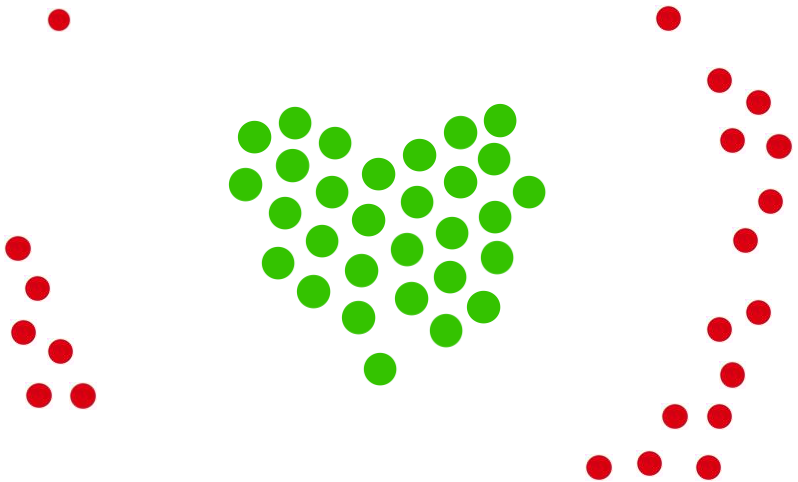
OBJECTIVE FUNCTION SIMPLICITY



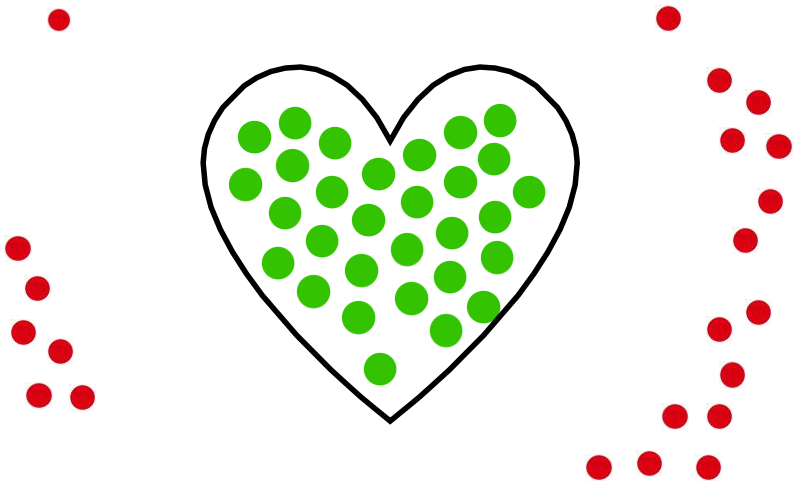
OBJECTIVE FUNCTION SIMPLICITY



OBJECTIVE FUNCTION SPECIFICITY



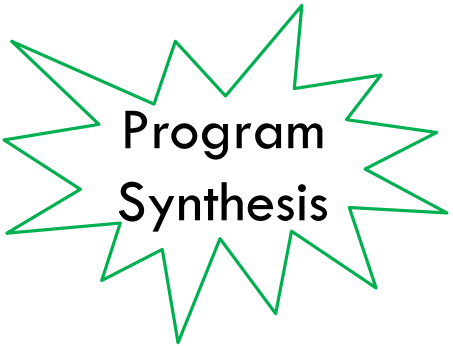
OBJECTIVE FUNCTION SPECIFICITY



RESEARCH GOALS

1. Interpretability - Inferred rules should be human readable
2. Data efficiency – Few shot learning
3. Interactivity - Inference at interactive speeds

RESEARCH GOALS



1. Interpretability - Inferred rules should be human readable
2. Data efficiency – Few shot learning
3. Interactivity - Inference at interactive speeds

CONSTRAINT BASED PROGRAM SYNTHESIS

Represent rule A → B / L _ R as a program

CONSTRAINT BASED PROGRAM SYNTHESIS

Represent rule A → B / L _ R as a program

$$\mathrm{F}(R, U, X)$$

CONSTRAINT BASED PROGRAM SYNTHESIS

Represent rule A → B / L _ R as a program

$\mathrm{F}(R, U, X)$

Consistent program

CONSTRAINT BASED PROGRAM SYNTHESIS

Represent rule A → B / L _ R as a program

$\mathrm{F}(R, U, X)$

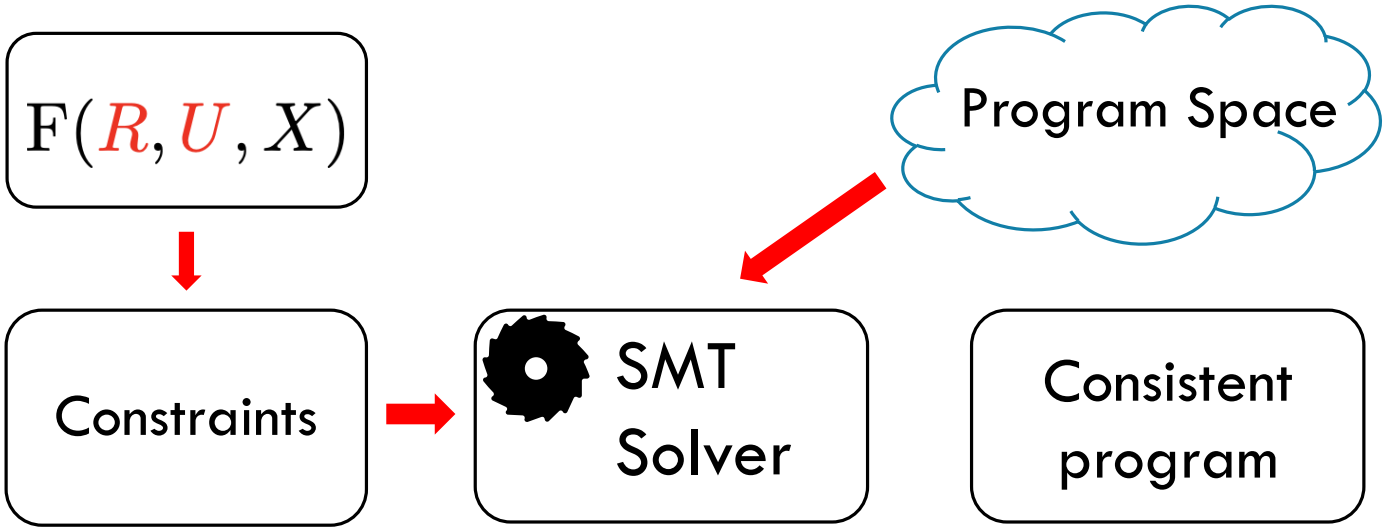
Constraints

Program Space

Consistent program

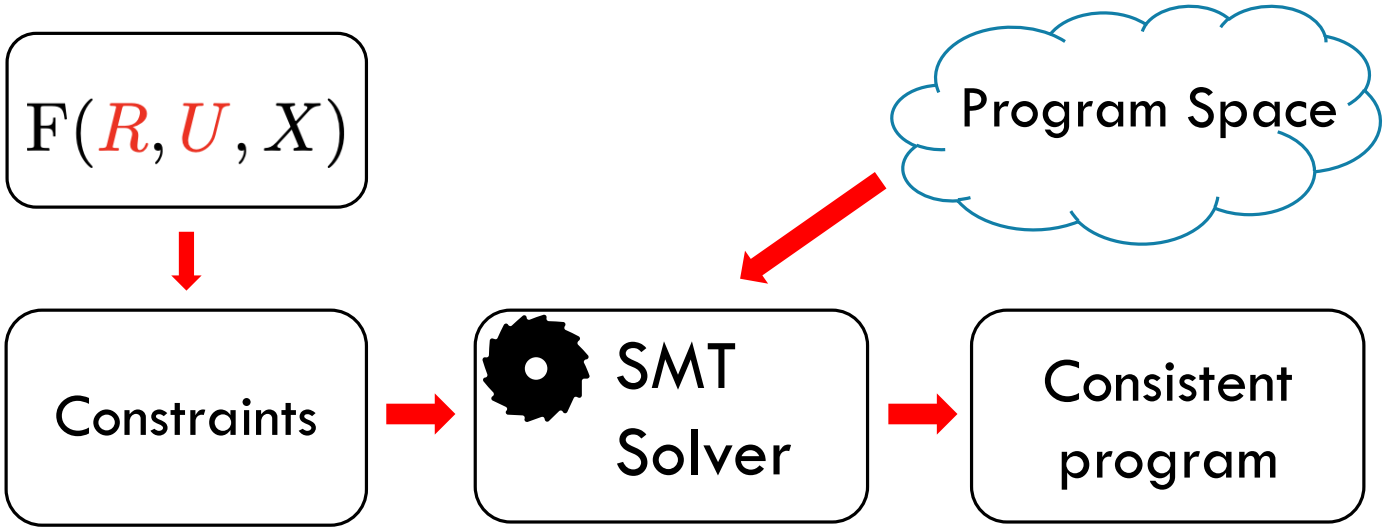
CONSTRAINT BASED PROGRAM SYNTHESIS

Represent rule A → B / L _ R as a program

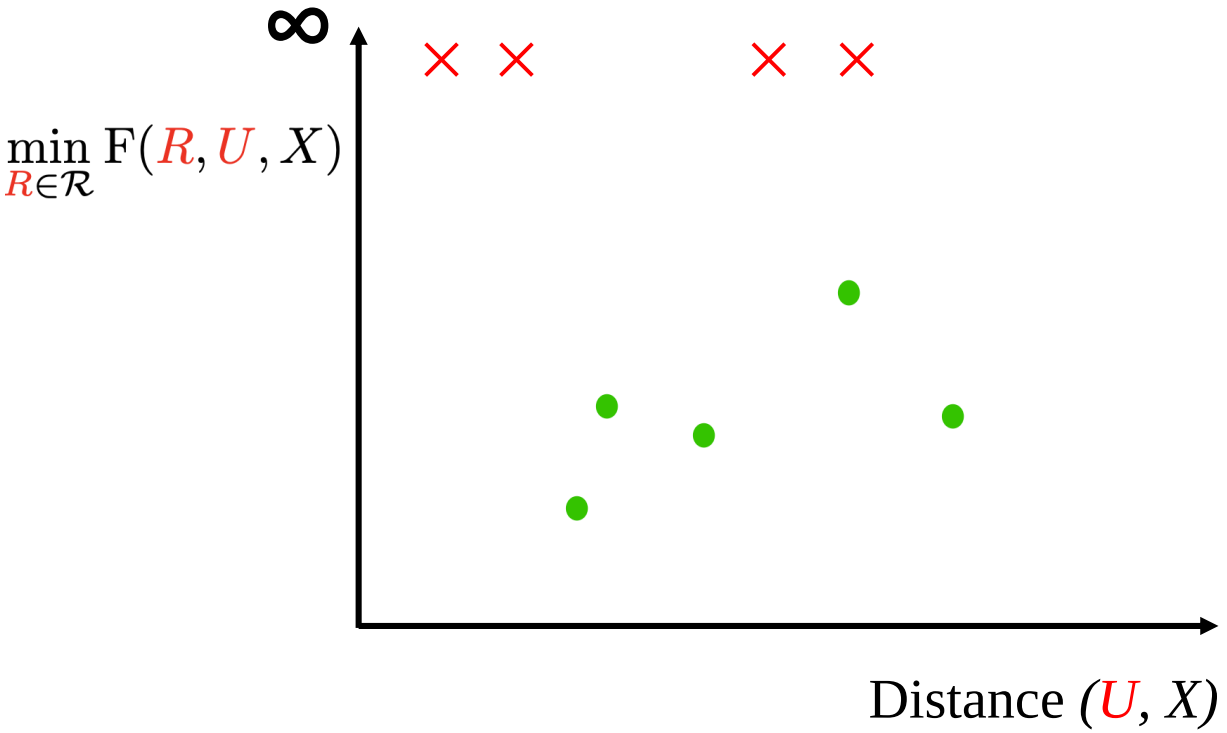


CONSTRAINT BASED PROGRAM SYNTHESIS

Represent rule A → B / L _ R as a program

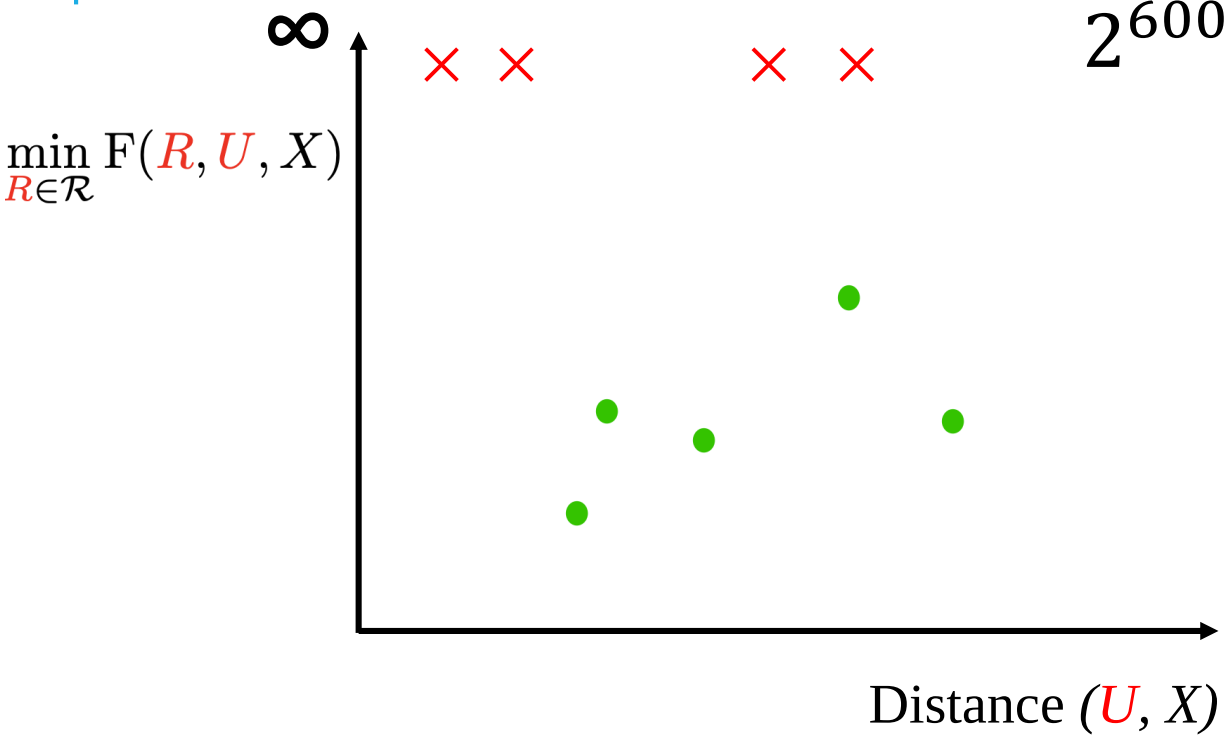


HYPOTHESIS SPACE



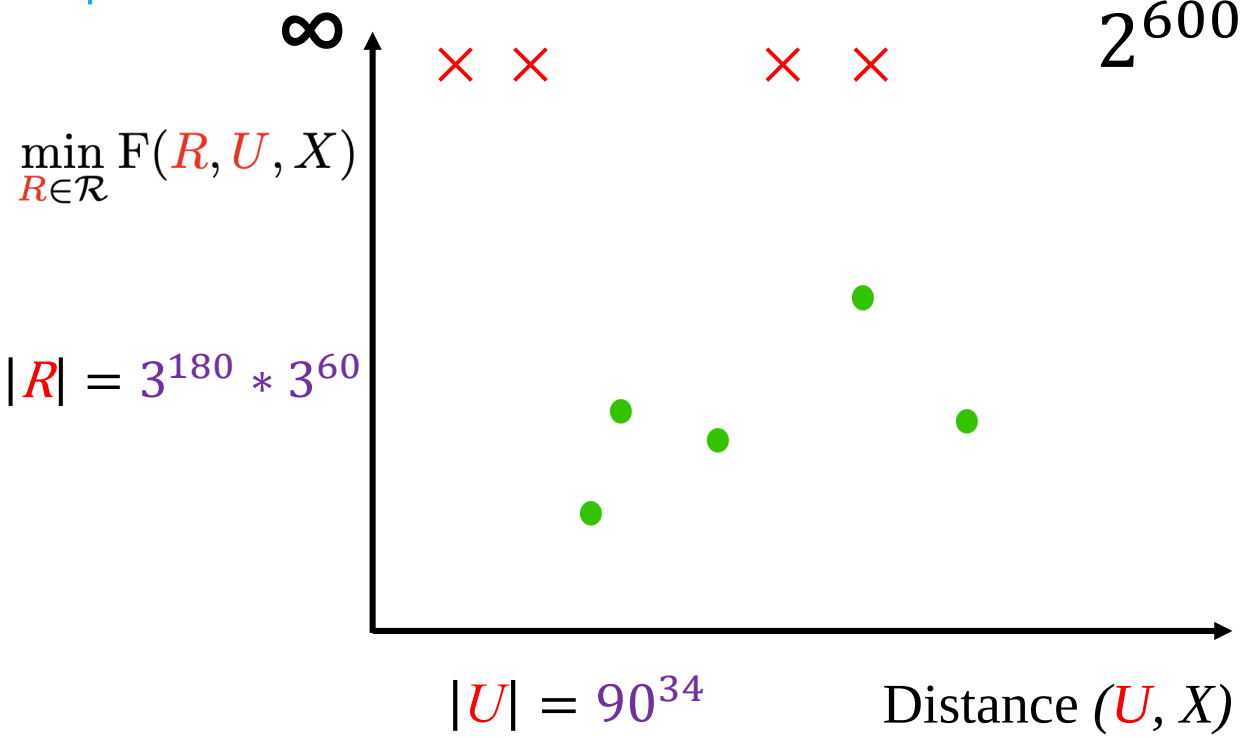
HYPOTHESIS SPACE

2^{600}



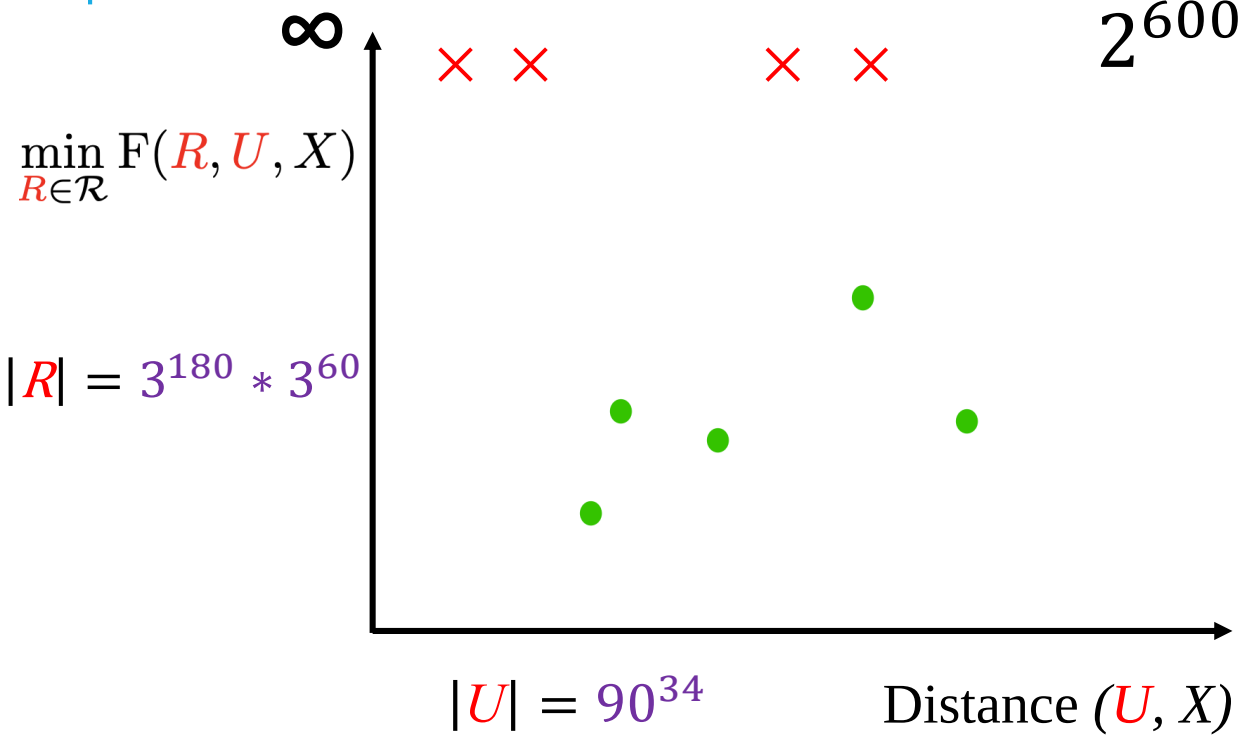
HYPOTHESIS SPACE

2^{600}

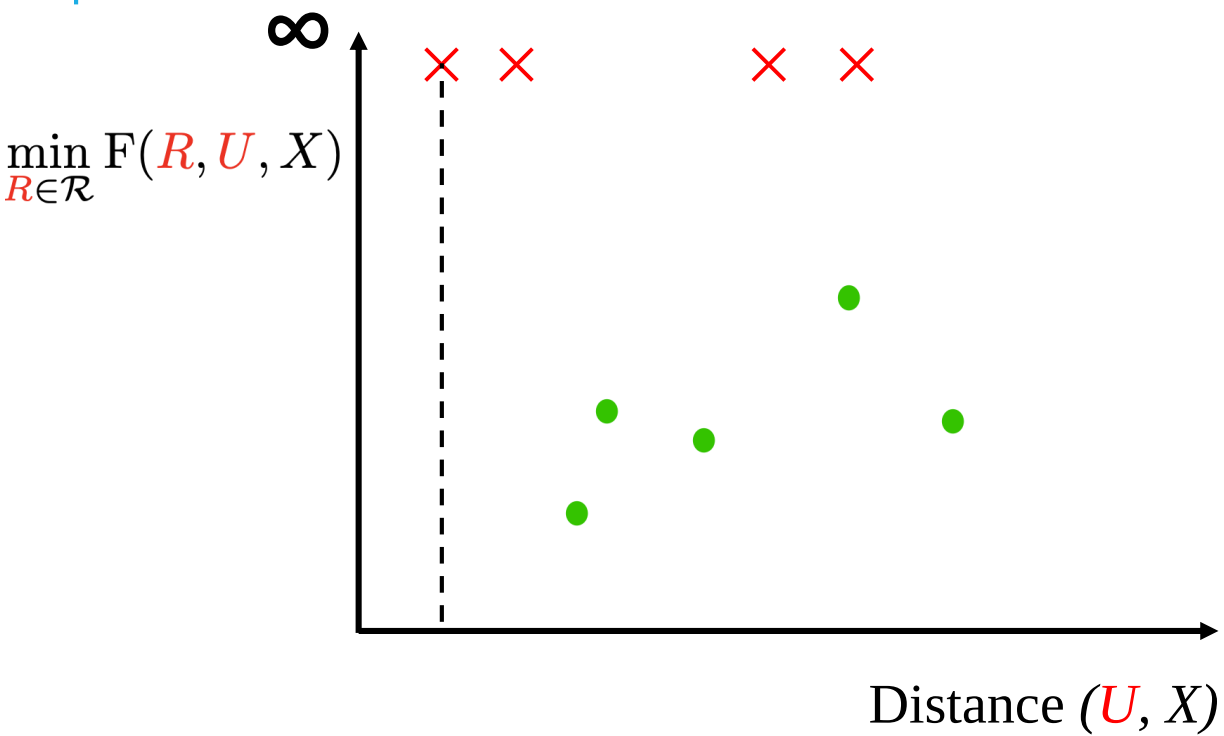


GLOBAL BASELINE

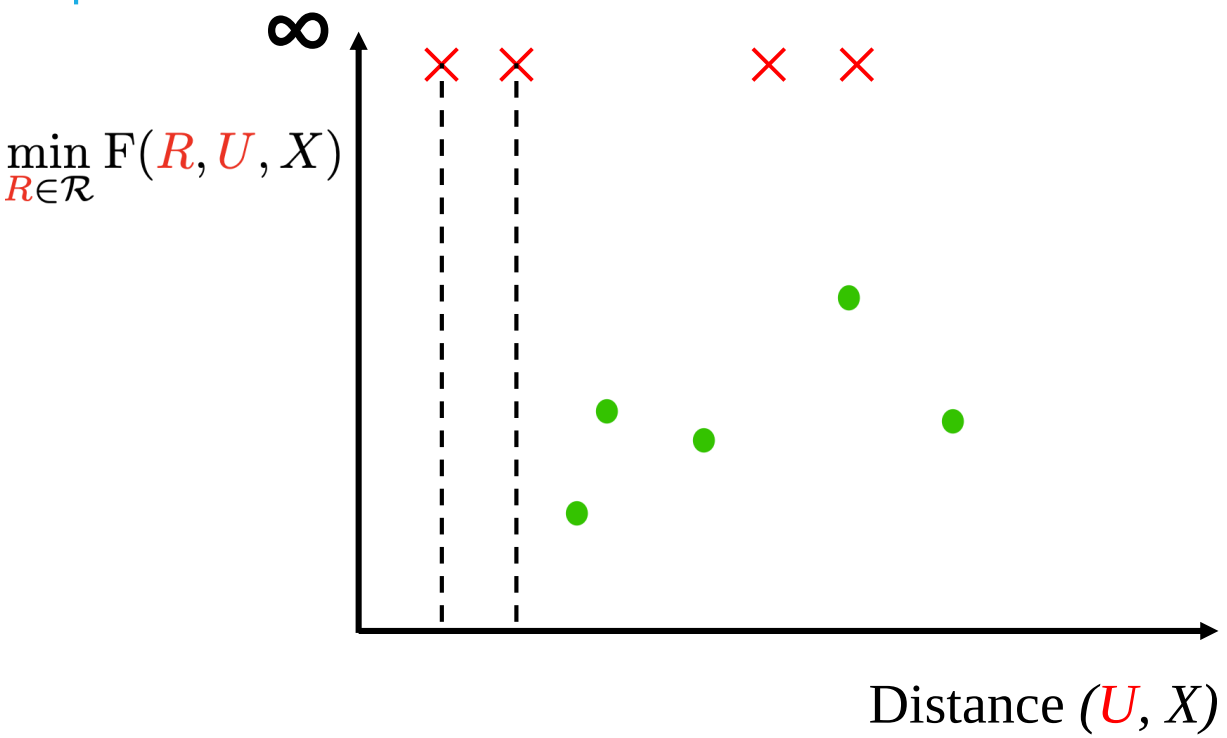
2^{600}



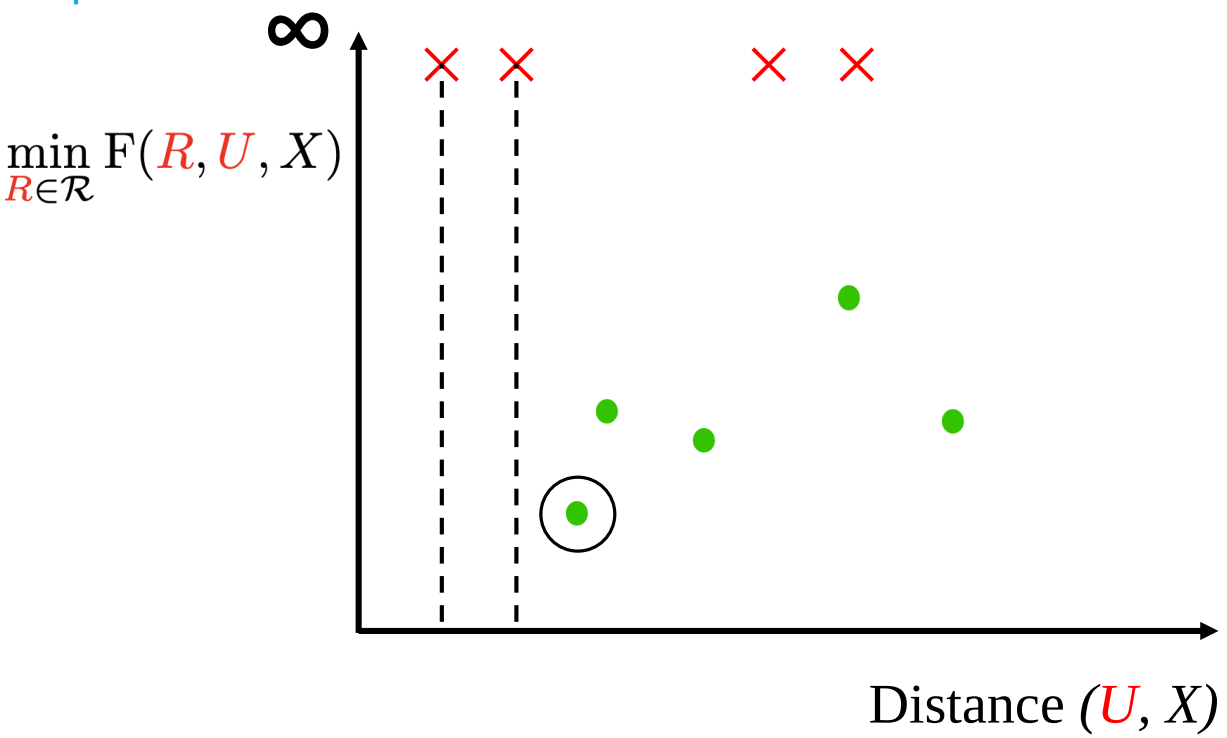
OUR SOLUTION



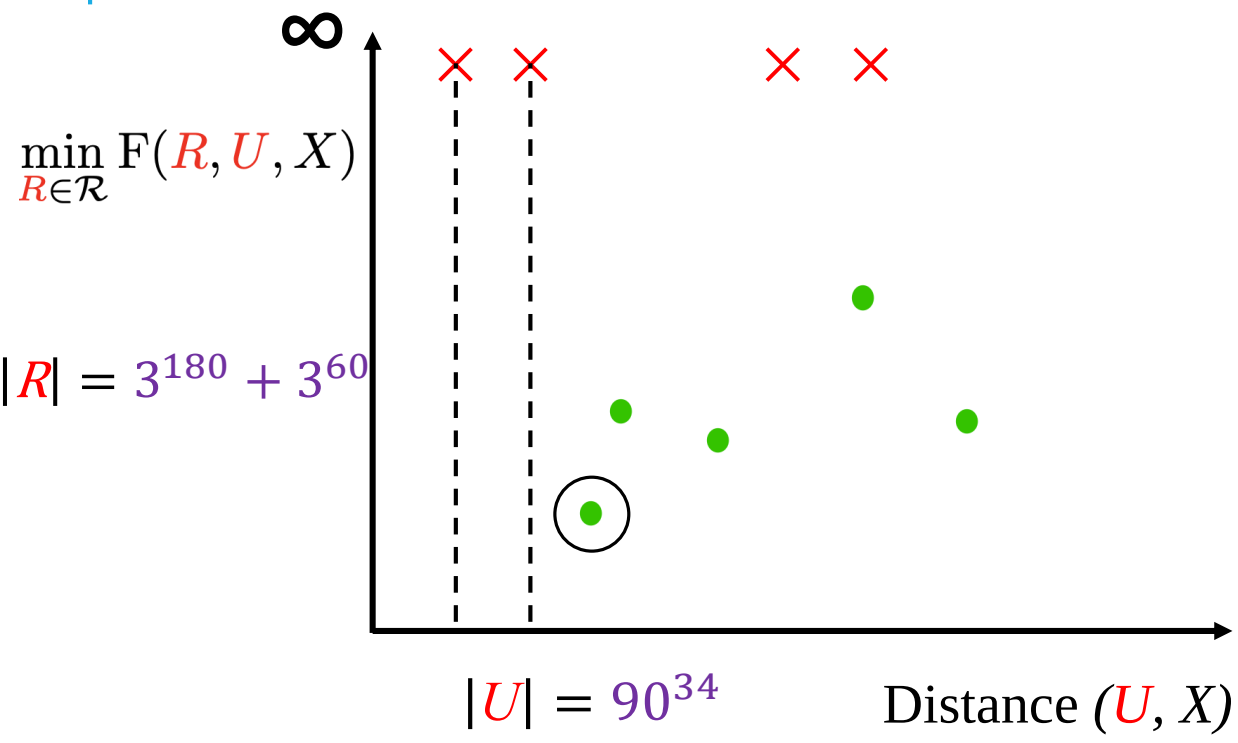
OUR SOLUTION



OUR SOLUTION



OUR SOLUTION



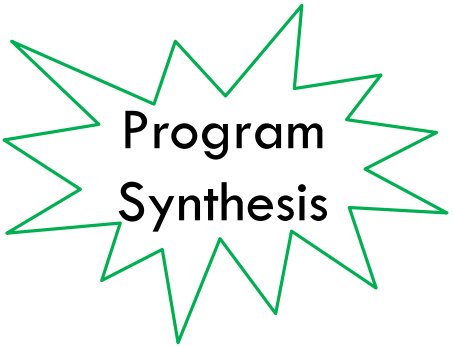
OUR CONTRIBUTION

1. Decomposition of the rule learning problem
   1. Underlying form inference
   2. Change inference
   3. Condition inference

OUR CONTRIBUTION

1. Decomposition of the rule learning problem
   1. Underlying form inference
   2. Change inference
   3. Condition inference
2. Efficient SMT encoding

RESEARCH GOALS



1. Interpretability – Represent rules as programs
2. Data efficiency – Hard constraints
3. Interactivity – Novel problem decomposition and efficient SMT encoding

OUTLINE

1. Problem Statement
2. Our Solution
3. Experimental Results

EXPERIMENTAL DATA

1. Textbook Problems : 34 (~20 Datapoints)
2. Lexical Datasets : 2 (~6000 Datapoints)

32 Languages

RUSSIAN TEXTBOOK PROBLEM (ODDEN 2015)

Gen. Plural	Nom. Singular
vagon	vagona
xlep	xleba
ras	raza
porok	poroga
soldat	soldata
golos	golosa

RUSSIAN DEVOICING RULE

/b/, /z/, /g/ → [p] , [s], [k] / _ #

[-sonorant] → [-voice] / _ #

LEXICAL DATASETS

1. English Flapping

Processed CMU pronouncing dictionary to create underlying and surface form pairs exemplifying flapping.

2. English Verbs

Combined morphological information extracted from CELEX-2 with CMU transcriptions to create a database of regular verbs.

ENGLISH VERB RULES

Devoicing rule

[-sonorant] → [-voice] / [-voice] _

Insertion rule

ø → ə / [αstrident] _ [αstrident]

TEXTBOOK PROBLEM LANGUAGES

Magyar русский English Lietuvių Polski

Türkçe فارسی 한국어 Deutsch Nederlands

EVALUATION METRICS

Learn rule set from 20, 50 and 100 data points

	Accuracy	Rule Match	
		Precision	Recall
Flap 20	76	50	31
Flap 50	93	86	86
Flap 100	100	100	100
Verb 20	86	48	83
Verb 50	88	52	92
Verb 100	95	62	100

EVALUATION METRICS

Learn rule set from 20, 50 and 100 data points

Accuracy evaluated on held out data points

	Accuracy	Rule Match	
		Precision	Recall
Flap 20	76	50	31
Flap 50	93	86	86
Flap 100	100	100	100
Verb 20	86	48	83
Verb 50	88	52	92
Verb 100	95	62	100

EVALUATION METRICS

Learn rule set from 20, 50 and 100 data points

Accuracy evaluated on held out data points

Syntactic comparison of rule set against the gold standard rules

	Accuracy	Rule Match	
		Precision	Recall
Flap 20	76	50	31
Flap 50	93	86	86
Flap 100	100	100	100
Verb 20	86	48	83
Verb 50	88	52	92
Verb 100	95	62	100

LEXICAL DATASETS

Learn rule set from 20, 50 and 100 data points

Accuracy evaluated on held out data points

Syntactic comparison of rule set against the gold standard rules

	Accuracy	Rule Match	
		Precision	Recall
Flap 20	76	50	31
Flap 50	93	86	86
Flap 100	100	100	100
Verb 20	86	48	83
Verb 50	88	52	92
Verb 100	95	62	100

EVALUATION TEXTBOOK PROBLEMS

Classes of textbook problems
of different complexity

	Accuracy	Rule Match	
		Precision	Recall
10 MAT	100	70	77
20 ALT	100	66	71
4 SUP	100	63	71
10 TEST	100	54	61

EVALUATION TEXTBOOK PROBLEMS

Classes of textbook problems
of different complexity

	Accuracy	Rule Match	
		Precision	Recall
10 MAT	100	70	77
20 ALT	100	66	71
4 SUP	100	63	71
10 TEST	100	54	61

Held out test problems

EVALUATION TEXTBOOK PROBLEMS

Classes of textbook problems
of different complexity

	Accuracy	Rule Match	
		Precision	Recall
10 MAT	100	70	77
20 ALT	100	66	71
4 SUP	100	63	71
10 TEST	100	54	61

Held out test problems

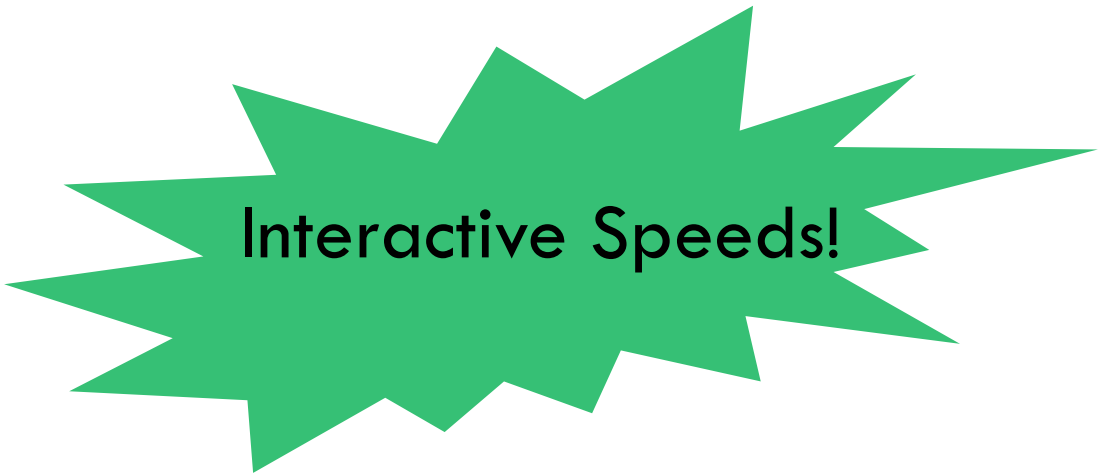
INFERENCE TIME SPEEDUP

SYPHON = BASELINE / 10^2

	Inference Time (secs)		
	SYPHON	Baseline	Speedup
MAT	30.0	3100	124.6
ALT	10.7	N/A	N/A
SUP	5.3	6333	378.3
TEST	8.3	N/A	N/A

INFERENCE TIME SPEEDUP

$$\text{SYPHON} = \text{BASELINE} \; / \; 10^{\,2}$$



CONCLUSION

1. Novel problem decomposition leads to interactivity
2. Phonologists can use our system for automated scientific investigation

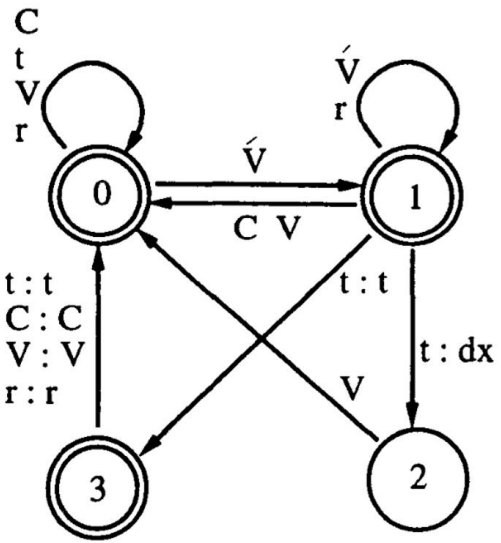
SYPHON 

QUESTIONS?

1. Novel problem decomposition leads to interactivity
2. Phonologists can use our system for automated scientific investigation

SYPHON 

STRING TRANSDUCERS



Ex: batter

Underlying:

b	ae1	t	er

Surface:

b	ae1	dx	er

STRING TRANSDUCERS

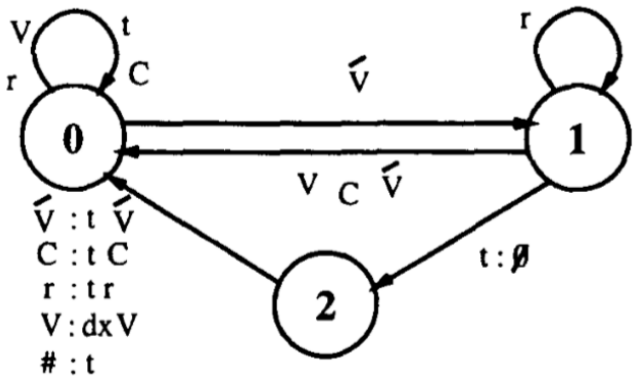


Figure 18
Flapping transducer induced from 50,000 samples (same as Figure 14).