Human-in-the-Loop Parsing

Luheng He, Julian Michael, *Mike Lewis, Luke Zettlemoyer

University of Washington

EMNLP 2016

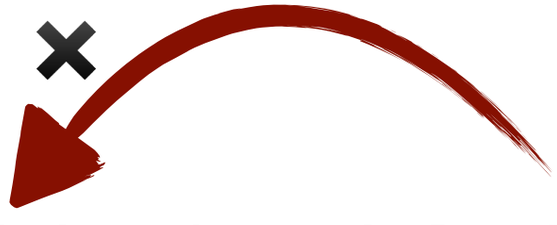
*Now at Facebook AI Research

Our key hypothesis:
Anyone who **understands the meaning of a sentence** should be able to correct **parser mistakes**.

Our key hypothesis:
Anyone who **understands the meaning of a sentence** should be able to correct **parser mistakes**.

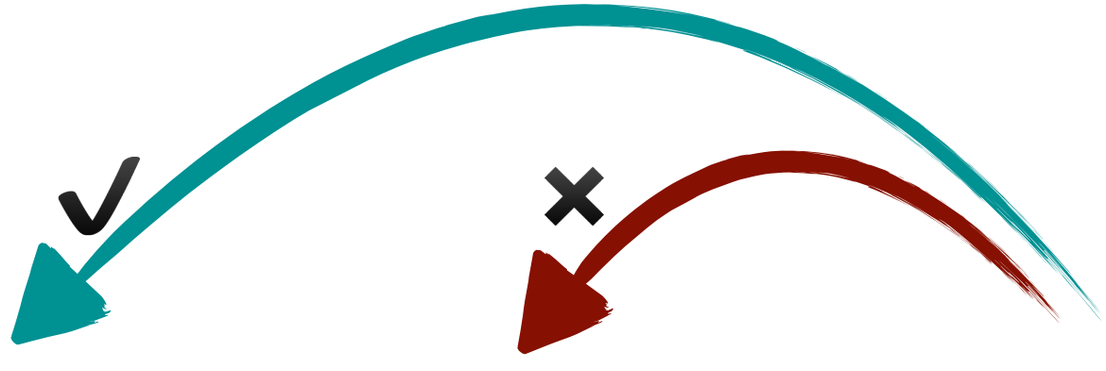
Pat ate the cake on the table that I ***baked*** last night.

Our key hypothesis:
Anyone who **understands the meaning of a sentence** should be able to correct **parser mistakes**.



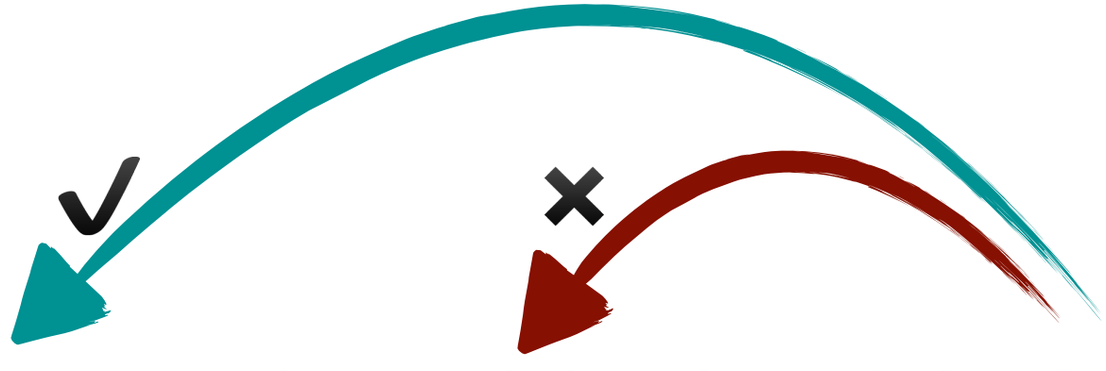
Pat ate the cake on the table that I ***baked*** last night.

Our key hypothesis:
Anyone who **understands the meaning of a sentence** should be able to correct **parser mistakes**.



Pat ate the cake on the table that I ***baked*** last night.

Our key hypothesis:
Anyone who **understands the meaning of a sentence** should be able to correct **parser mistakes**.

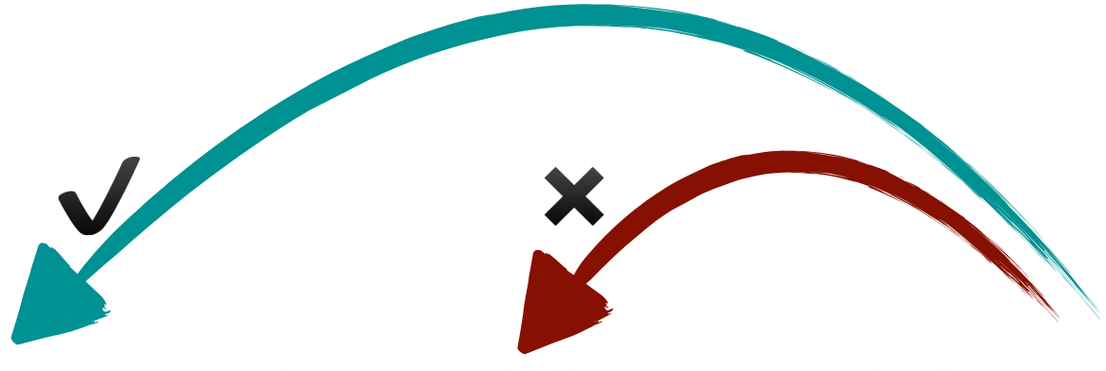


Pat ate the cake on the table that I ***baked*** last night.

Parser: I baked **table**

Human understanding: I baked **cake**

Our key hypothesis:
Anyone who **understands the meaning of a sentence** should be able to correct **parser mistakes**.



Pat ate the cake on the table that I ***baked*** last night.

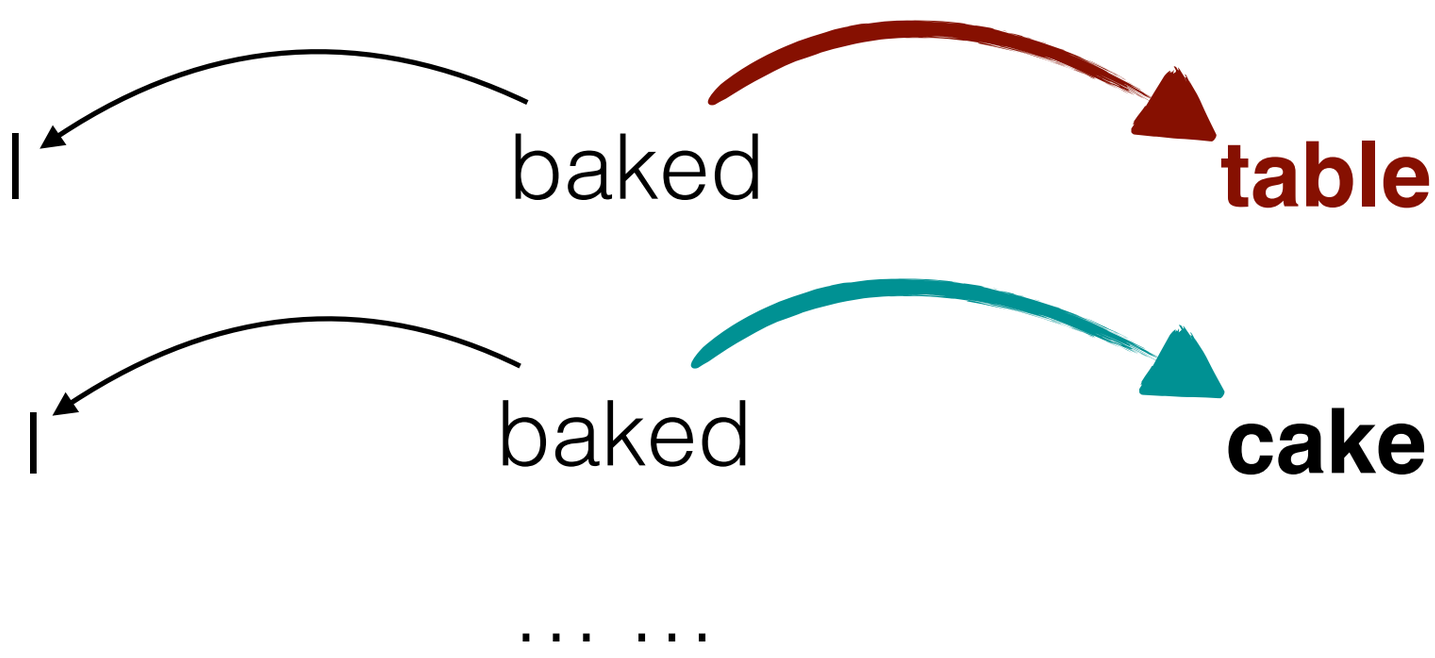
Parser: I baked **table**

Human understanding: I baked **cake**

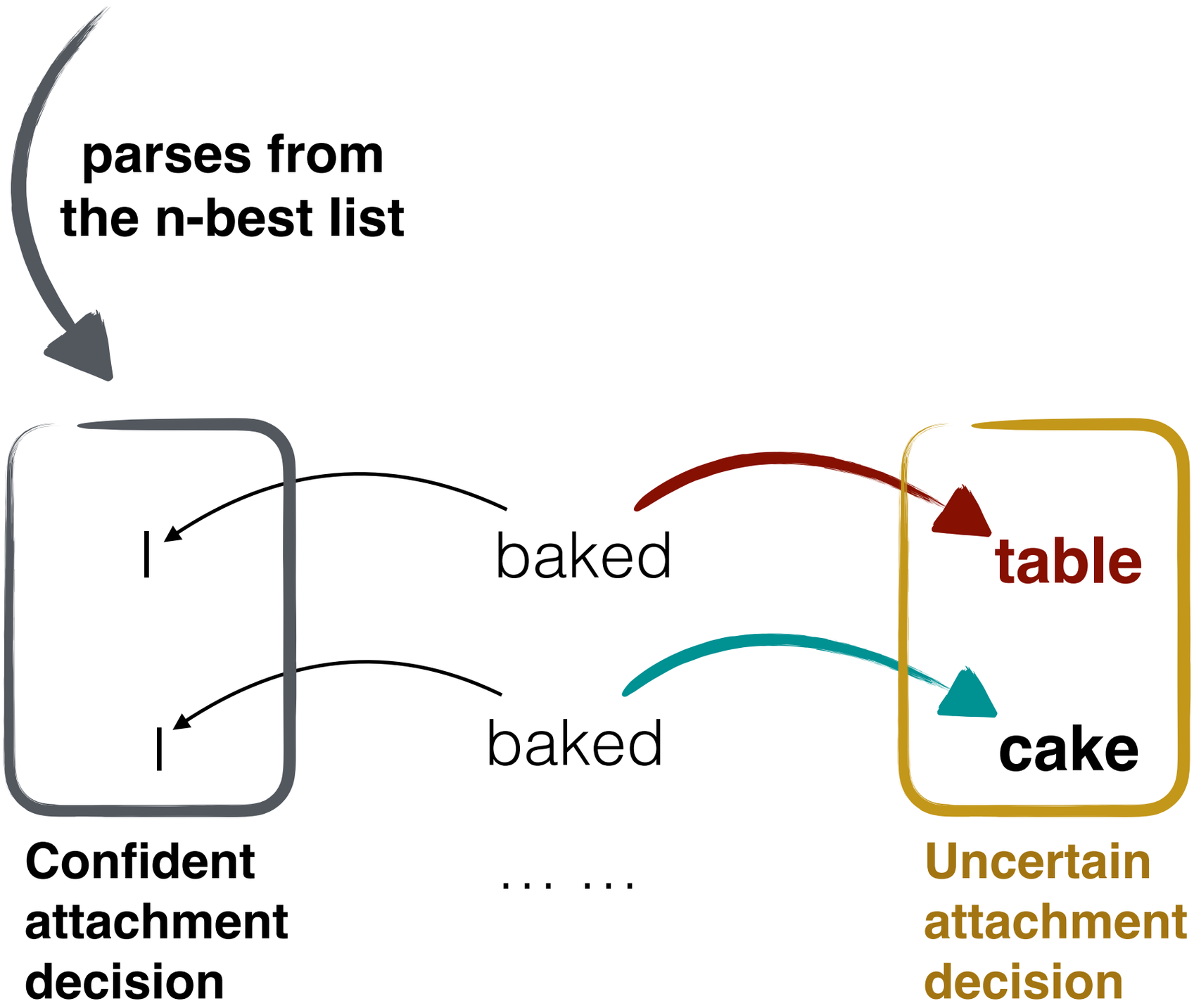
How can we use this kind of human knowledge?

Pat ate the cake on the table that I ***baked*** last night.

parses from the n-best list



Pat ate the cake on the table that I ***baked*** last night.



Pat ate the cake on the table that I ***baked*** last night.

parses from the n-best list

human judgment

I

I

baked

baked

… …

table

cake

Confident attachment decision

Uncertain attachment decision

Pat ate the cake on the table that I ***baked*** last night.

parses from the n-best list

Q: What did someone ***bake***?

1. table 2. cake

human judgment

I

baked

table

I

baked

cake

Confident attachment decision

… …

Uncertain attachment decision

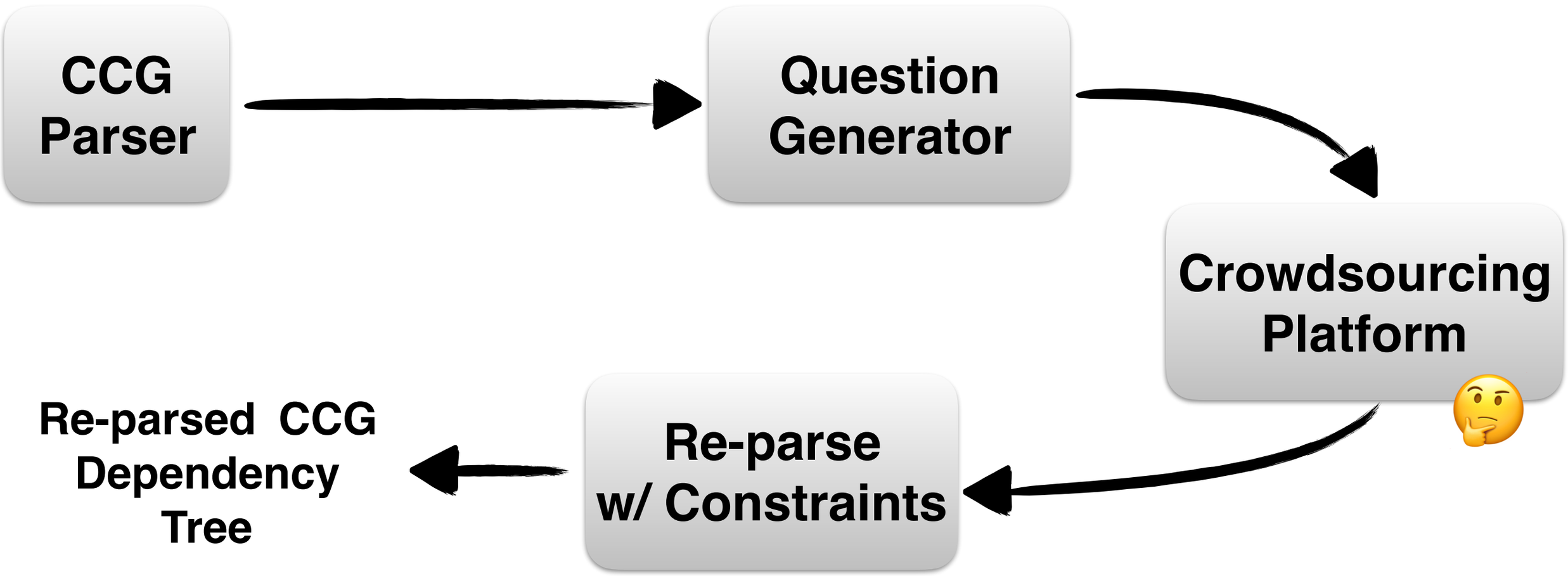
Related Work

	Form of Supervision	Data Collection	Usage
Jha et al., 2010	PP Attachment Decisions	Crowdsourced	/
Choe and McClosky, 2015	Paraphrases	In-house Annotator	Re-parsing
Duan et al., 2016	Paraphrases	Crowdsourced	Re-training

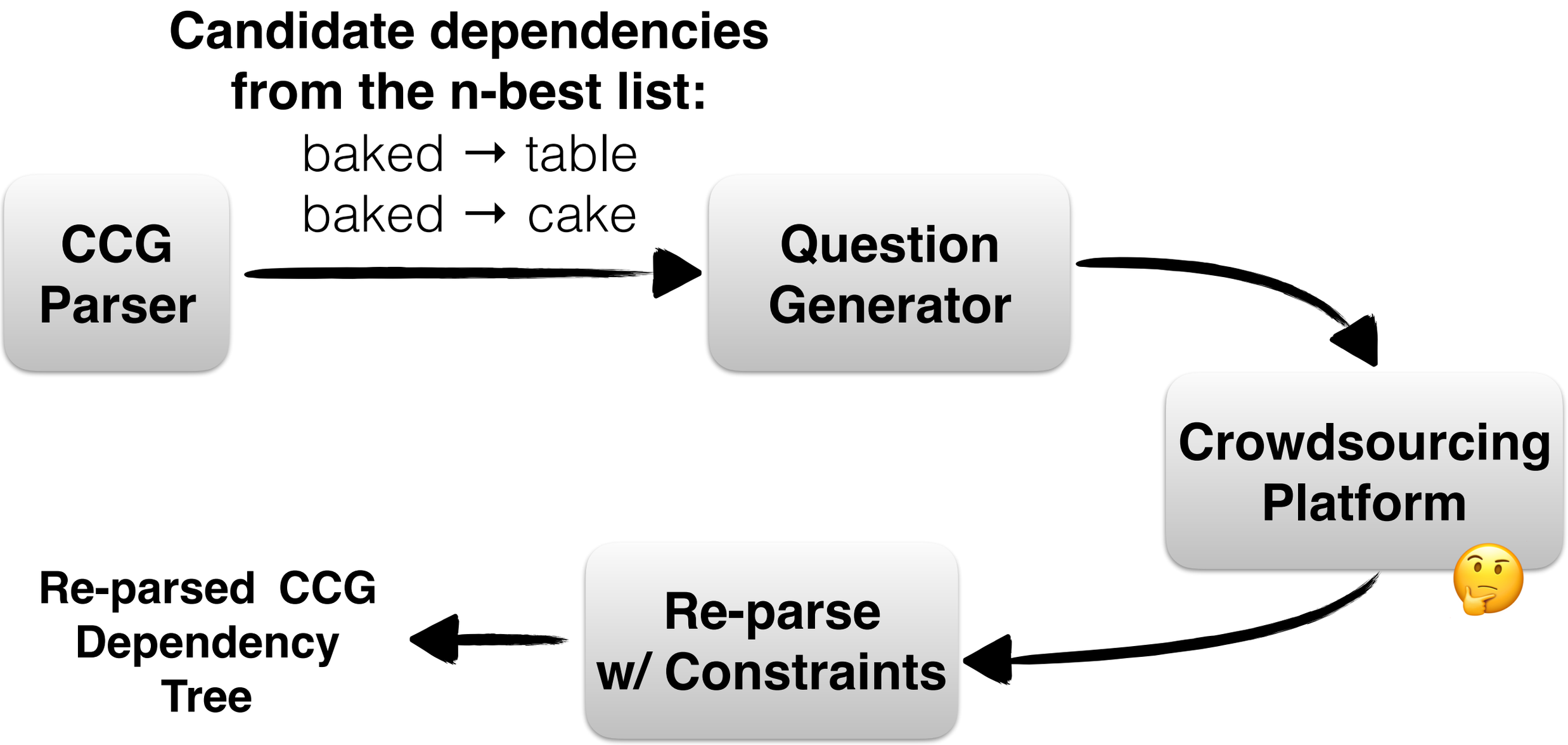
Scope of this Work

- Target core arguments of verbal predicates.
- Use human judgments to fix parser mistakes at decoding time.
- Use CCG (Combinatory Categorial Grammar) as the underlying syntactic formalism.
- Use the Neural CCG Parser (Lewis et al. 2016) as our base parser.

Workflow


CCG Parser
Question Generator
Crowdsourcing Platform
Re-parse w/ Constraints
Re-parsed CCG Dependency Tree

Workflow


Candidate dependencies from the n-best list:
baked → table
baked → cake
CCG Parser
Question Generator
Crowdsourcing Platform
Re-parse w/ Constraints
Re-parsed CCG Dependency Tree

Workflow

CCG Parser

Candidate dependencies from the n-best list:
baked → table
baked → cake

Question Generator

Q: "What did someone ***bake***?"
1) table **2)** cake

Crowdsourcing Platform 🤔

Re-parse w/ Constraints

Re-parsed CCG Dependency Tree

Workflow

CCG Parser

Candidate dependencies from the n-best list:
baked → table
baked → cake

Question Generator

Q: "What did someone ***bake***?"
1) table **2)** cake

Crowdsourcing Platform 🤔

cake (4 votes)
table (1 vote)

Re-parse w/ Constraints

Re-parsed CCG Dependency Tree

Workflow

CCG Parser

Candidate dependencies from the n-best list:
baked → table
baked → cake

Question Generator

Q: "What did someone ***bake***?"
1) table **2)** cake

Crowdsourcing Platform 🤔

cake (4 votes)
table (1 vote)

Re-parse w/ Constraints

C_pos (bake → cake)
C_neg (bake → table)

Re-parsed CCG Dependency Tree

Workflow

CCG Parser

Candidate dependencies from the n-best list:
baked → table
baked → cake

Question Generator

Q: "What did someone ***bake***?"
1) table **2)** cake

Crowdsourcing Platform 🤔

cake (4 votes)
table (1 vote)

Re-parse w/ Constraints

C_pos (bake → cake)
C_neg (bake → table)

Re-parsed CCG Dependency Tree

Not re-training the model

Generate Q/A Pairs from CCG Dependencies

Predicted CCG category of ***baked***: $(S\backslash NP_1)/NP_2$

Generate Q/A Pairs from CCG Dependencies

Predicted CCG category of ***baked***: $(S\backslash NP_1)/NP_2$

Convert to template: | NP_1 | bake | NP_2 |

Generate Q/A Pairs from CCG Dependencies

Predicted CCG category of ***baked***: (S**NP**$_1$)/**NP**$_2$

Convert to template:

NP$_1$	bake	**NP**$_2$

Filling-in the Slots:

what	bake	sth.

Generate Q/A Pairs from CCG Dependencies

Predicted CCG category of ***baked***: (S**NP$_1$**)/**NP$_2$**

Convert to template: **NP$_1$** bake **NP$_2$**

Filling-in the Slots:

what bake sth.

What baked something?

— **I**

Generate Q/A Pairs from CCG Dependencies

Predicted CCG category of ***baked***: $(S\backslash NP_1)/NP_2$

Convert to template: [**NP_1**] bake [**NP_2**]

Filling-in the Slots:

[**what**] bake [sth.]

What baked something?

— **I**

[sth.] bake [**what**]

What did someone bake?

— **the table**

Generate Q/A Pairs from CCG Dependencies

Predicted CCG category of ***baked***: (S**NP$_1$**)/**NP$_2$**

Convert to template: **NP$_1$** bake **NP$_2$**

Filling-in the Slots:

what bake sth. — **What** baked something? — **I**

sth. bake **what** — **What** did someone bake? — **the table**

Infer **someone/something** and the **answer spans** based on the n-best parses

Used "**what**" for all questions

Generate Q/A Pairs from CCG Dependencies

Predicted CCG category of ***baked***: (S**NP$_1$**)/**NP$_2$**

Convert to template: **NP$_1$** bake **NP$_2$**

Filling-in the Slots:

what bake sth.

What baked something?

— **I**

sth. bake **what**

What did someone bake?

— **the table**

Infer **someone/something** and the **answer spans** based on the n-best parses

Used "**what**" for all questions

Generate Q/A Pairs from CCG Dependencies

Predicted CCG category of ***baked***: (S**NP**$_1$)/**NP**$_2$

Convert to template: [**NP**$_1$] bake [**NP**$_2$]

Filling-in the Slots:

[**what**] bake [sth.]

What baked something?
— **I**

What baked something?
— **I**

[sth.] bake [**what**]

What did someone bake?
— **the table**

What did someone bake?
— **the cake**

Infer **someone/something** and the **answer spans** based on the n-best parses

Used "**what**" for all questions

Group Q/A Pairs into Queries

Questions	Answers
What baked something?	**I**
What did someone bake?	**the table**
	the cake
What was baked something something?	**the table**

Group Q/A Pairs into Queries

Questions	Answers	Scores
What baked something?	**I**	1.0
What did someone bake?	**the table**	0.7
	the cake	0.3
What was baked something something?	**the table**	0.1

Group Q/A Pairs into Queries

Questions	Answers	Scores	Question Confidence
What baked something?	**I**	1.0	**1.0**
What did someone bake?	**the table**	0.7	**1.0**
	the cake	0.3	
What was baked something something?	**the table**	0.1	**0.1**

Group Q/A Pairs into Queries

Questions	Answers	Scores	Question Confidence	Answer Uncertainty (Entropy)
What baked something?	**I**	1.0	**1.0**	**0.0**
What did someone bake?	**the table**	0.7	**1.0**	**0.88**
	the cake	0.3		
What was baked something something?	**the table**	0.1	**0.1**	**0.0**

Group Q/A Pairs into Queries

Questions	Answers	Scores	Question Confidence	Answer Uncertainty (Entropy)
What baked something?	**I**	1.0	**1.0**	**0.0**
What did someone bake?	**the table**	0.7	**1.0**	**0.88**
	the cake	0.3		
What was baked something something?	**the table**	0.1	**0.1**	**0.0**

Non-sensical question

No uncertainty

Our Annotation Task

Sentence:

Pat ate the cake on the table that I **baked** last night.

Question:

What did someone bake?

Check one or more

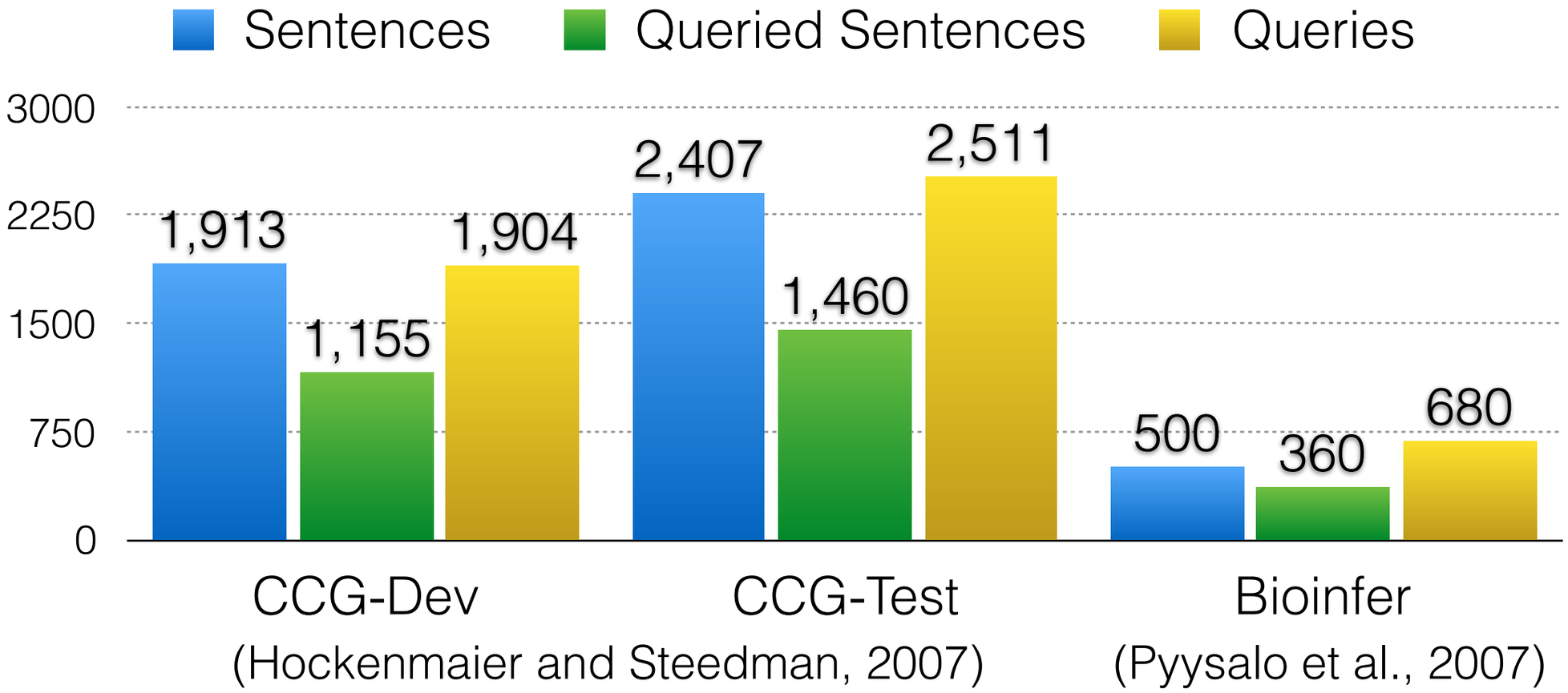
- ☐ the cake
- ☐ the table
- ☐ None of the above.

Comment:

- Annotators are instructed to choose options that "***explicitly and directly***" answer the question.
- Multiple answers are allowed.
- 5 judgements per query.

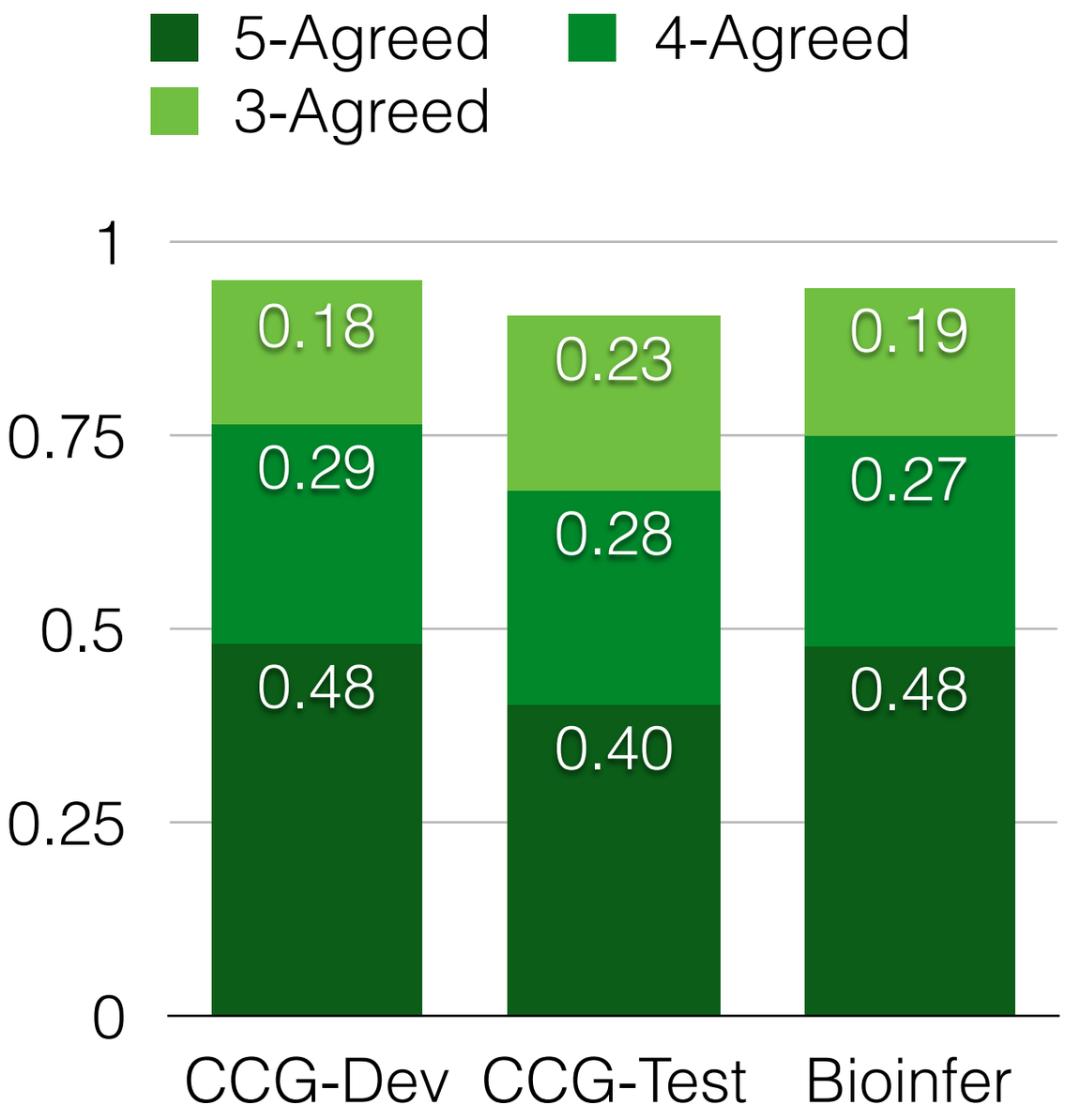
* Crowdsourcing platform: https://www.crowdflower.com/.

Data Collection with Crowdsourcing



- All developments are done on CCG-Dev only.
- Less than 2 queries per sentence, for about 60% of the sentences.
- **Cost:** 46 cents per query.
- **Speed:** 200 queries per hour.

Inter-Annotator Agreement



- Agreement is computed only for matching the exact set of answers. i.e. (A, B) and (B) are considered disagreement.
- Unanimous agreement for over 40% of the queries.
- Over 90% absolute majority.

Putting our hypothesis to the test:
How well does annotators' **human understanding** align with the **gold syntax?**

- Successes: Long-range attachment decisions
- Challenges: **Syntax-semantics mismatch**
- Use heuristics to fix the mismatch problems at re-parsing time.

Success - Long-range Dependency

Temple also said Sea Containers' plan raises numerous legal, regulatory, financial and fairness issues, but didn't ***elaborate***.

What *didn't* ***elaborate*** something?

Temple

Sea Containers' plan

None of the above.

Success - Long-range Dependency

Temple also said Sea Containers' plan raises numerous legal, regulatory, financial and fairness issues, but didn't ***elaborate***.

What *didn't* ***elaborate*** something?

4 Temple

1 Sea Containers' plan

0 None of the above.

Challenge - Coreference

Kalipharma is a New Jersey-based pharmaceuticals concern that ***sells*** products under the Purepac label.

What ***sells*** something?

Kalipharma

a New Jersey-based pharmaceuticals concern

None of the above.

Challenge - Coreference

Kalipharma is a New Jersey-based pharmaceuticals concern that ***sells*** products under the Purepac label.

What ***sells*** something?

5	Kalipharma
0	a New Jersey-based pharmaceuticals concern
0	None of the above.

Challenge - Coreference

Kalipharma is a New Jersey-based pharmaceuticals concern that ***sells*** products under the Purepac label.

What ***sells*** something?

5 Kalipharma

0 a New Jersey-based pharmaceuticals concern

0 None of the above.

- Syntax-semantics mismatch
- Also happens with pronouns and appositives.
- Some cases are heuristically fixed during reparsing.

Challenge - Headedness

Timex had requested duty-free treatment for many types of watches, **covered** by 58 different U.S. tariff classifications.

What would be ***covered*** ?

Timex

duty-free treatment

None of the above.

many types of watches

watches

Challenge - Headedness

Timex had requested duty-free treatment for many types of watches, **covered** by 58 different U.S. tariff classifications.

What would be ***covered*** ?

0 Timex

0 duty-free treatment

0 None of the above.

2 many types of watches

3 watches

Challenge - Headedness

Timex had requested duty-free treatment for many types of watches, **covered** by 58 different U.S. tariff classifications.

What would be ***covered*** ?

0 Timex

0 duty-free treatment

0 None of the above.

2 many types of watches

3 watches

Challenge - Headedness

Timex had requested duty-free treatment for many types of watches, **covered** by 58 different U.S. tariff classifications.

What would be ***covered*** ?

0 Timex

0 duty-free treatment

0 None of the above.

2 many types of watches

3 watches

- Annotators tend to struggle with headedness.
- We add “disjunctive constraint”, forcing the re-parser to produce either of the two dependencies.

Re-Parsing with Crowdsourced Constraints

Q1: What did someone ***bake***?

votes(*cake*) = 4

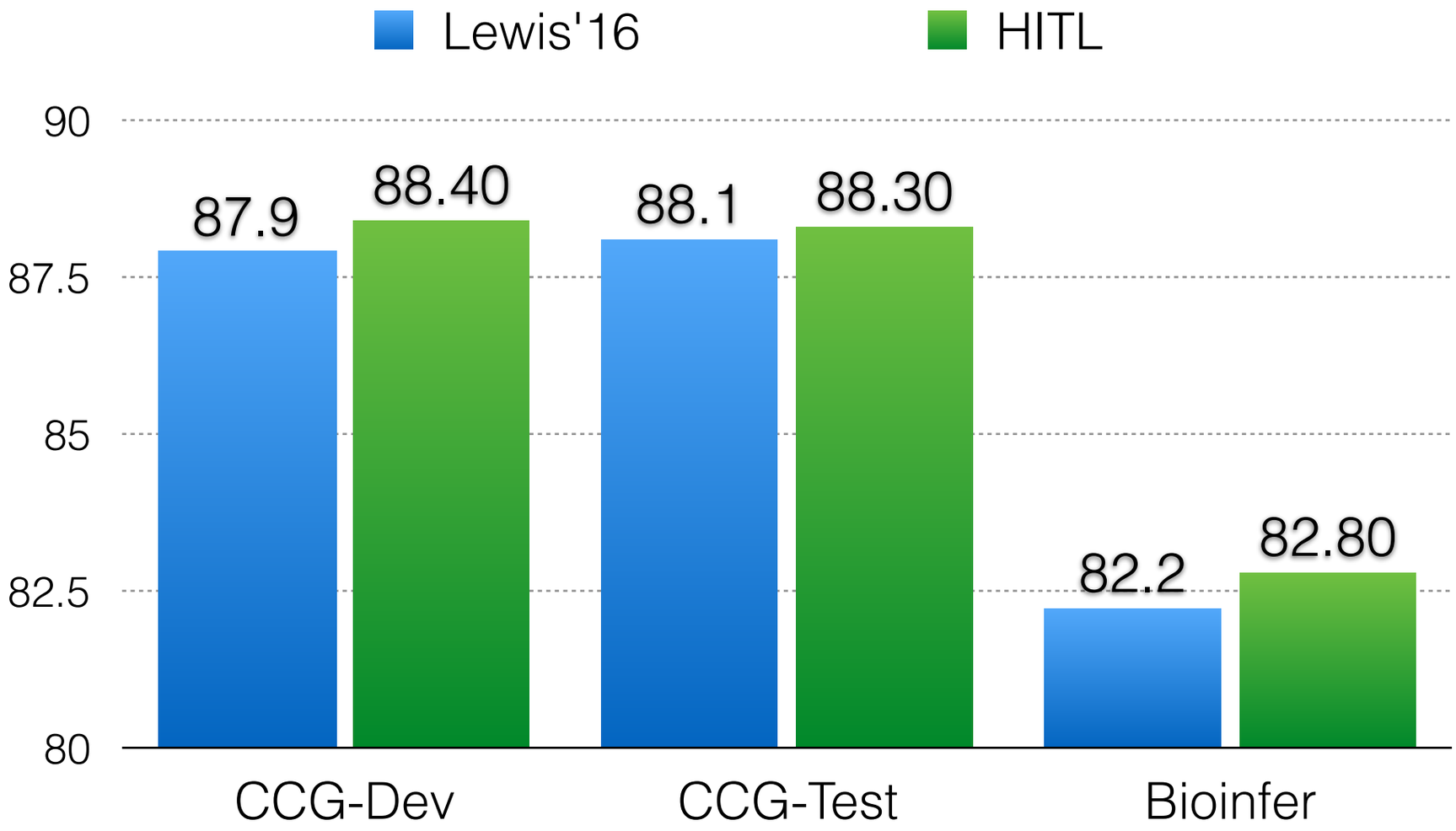
votes(*table*) = 1

votes(*None of the above*) = 0

$$y^{\text{new}} = \arg\max_{y} \quad \text{base\_parser\_score}(y)$$
$$-T^{+} \times \mathbb{1}(\text{baked} \rightarrow \text{cake} \in y)$$
$$-T^{-} \times \mathbb{1}(\text{baked} \rightarrow \text{table} \in y)$$

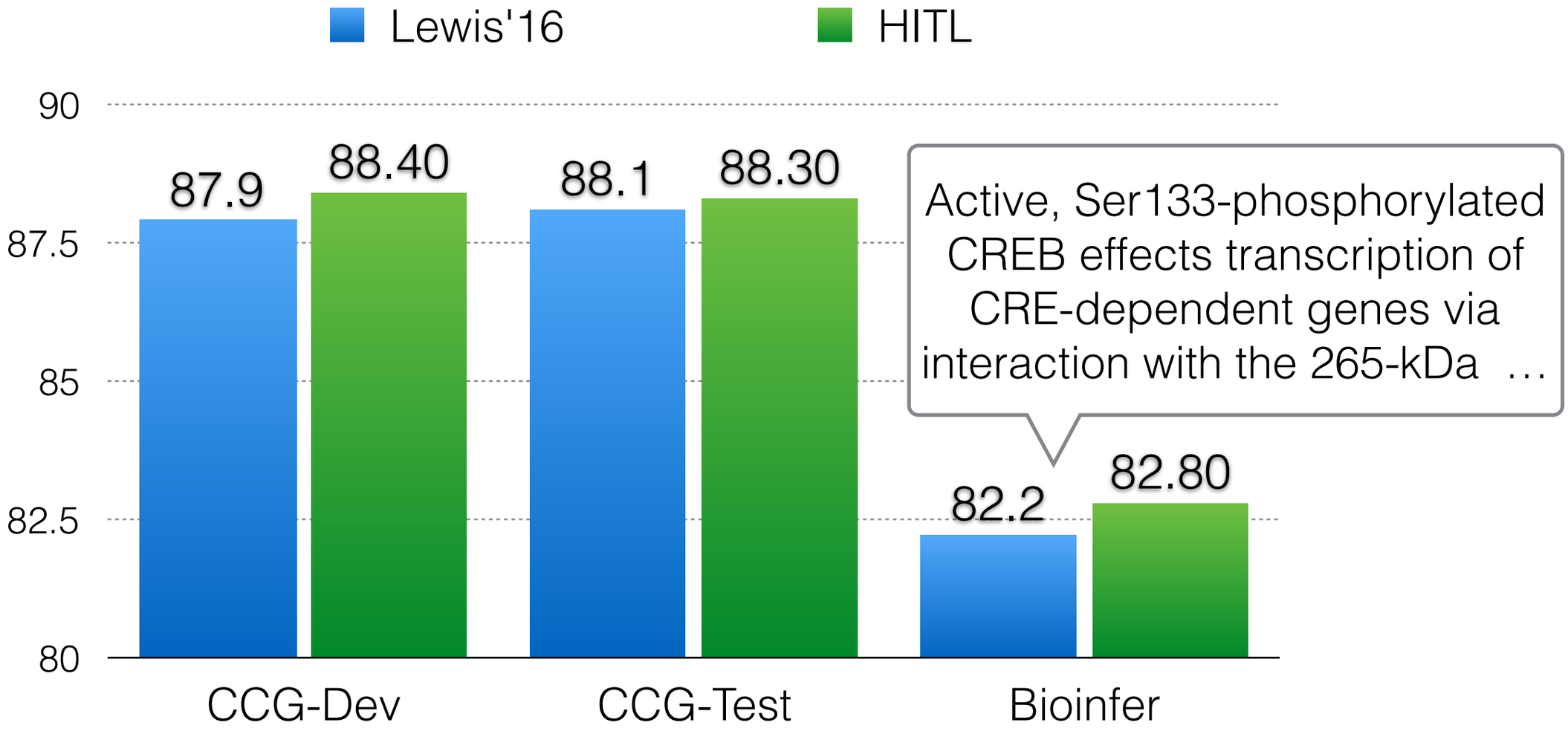
- Penalizes parses that disagree with crowdsourced judgments.
- Constraints are decomposed by dependencies.
- Thresholds and penalties are tuned on CCG-Dev.

Re-parsing Results (Labeled F1)



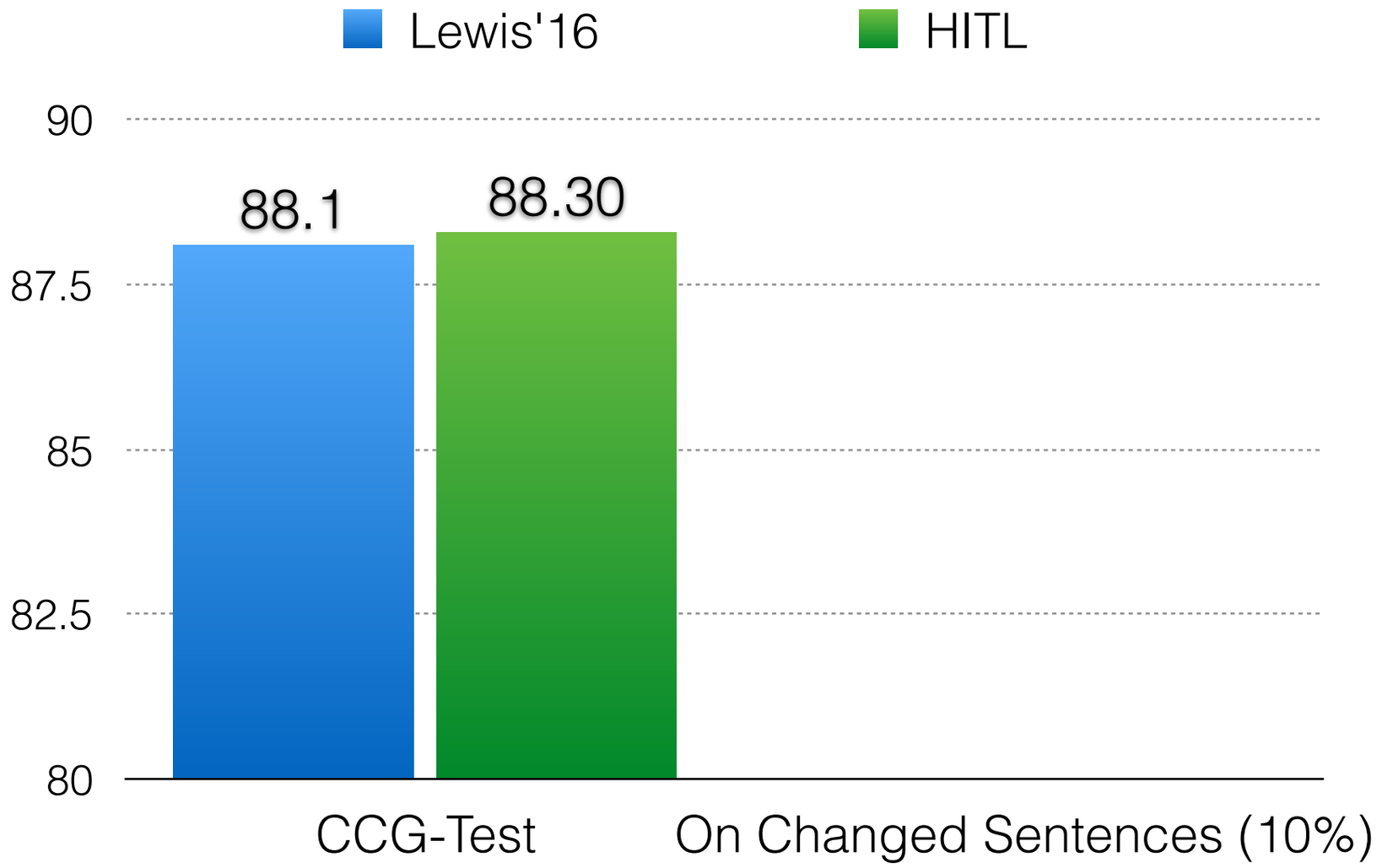
- Modest improvement due to syntax-semantics mismatch.
- Larger improvement on out-of-domain data.

Re-parsing Results (Labeled F1)



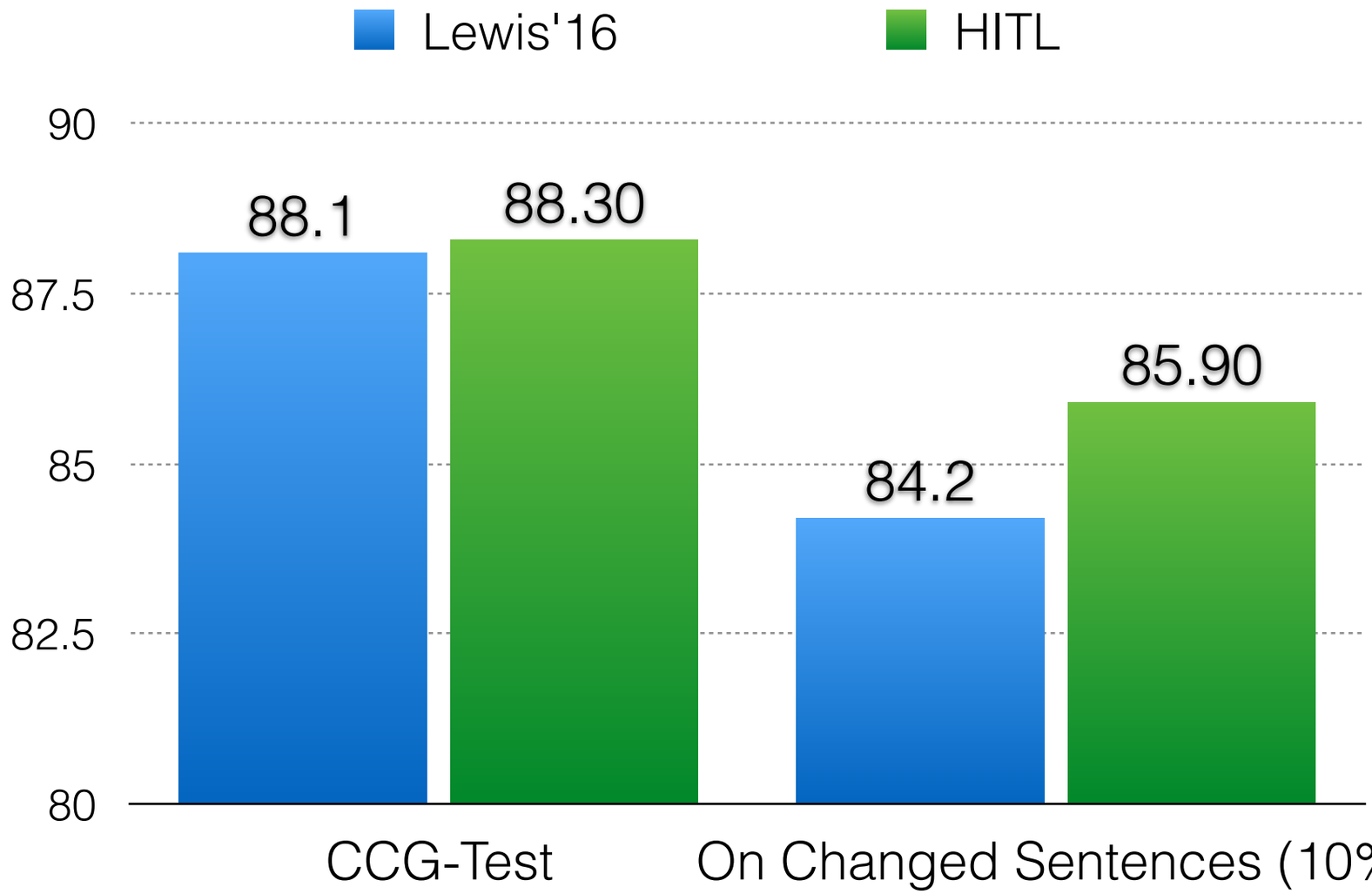
- Modest improvement due to syntax-semantics mismatch.
- Larger improvement on out-of-domain data.

Re-parsing Results



- Modified parse trees for about 10% of the sentences after incorporating human judgments.
- Larger gain on changed sentences.
- Changed sentences are “more difficult” on average.

Re-parsing Results



- Modified parse trees for about 10% of the sentences after incorporating human judgments.
- Larger gain on changed sentences.
- Changed sentences are “more difficult” on average.

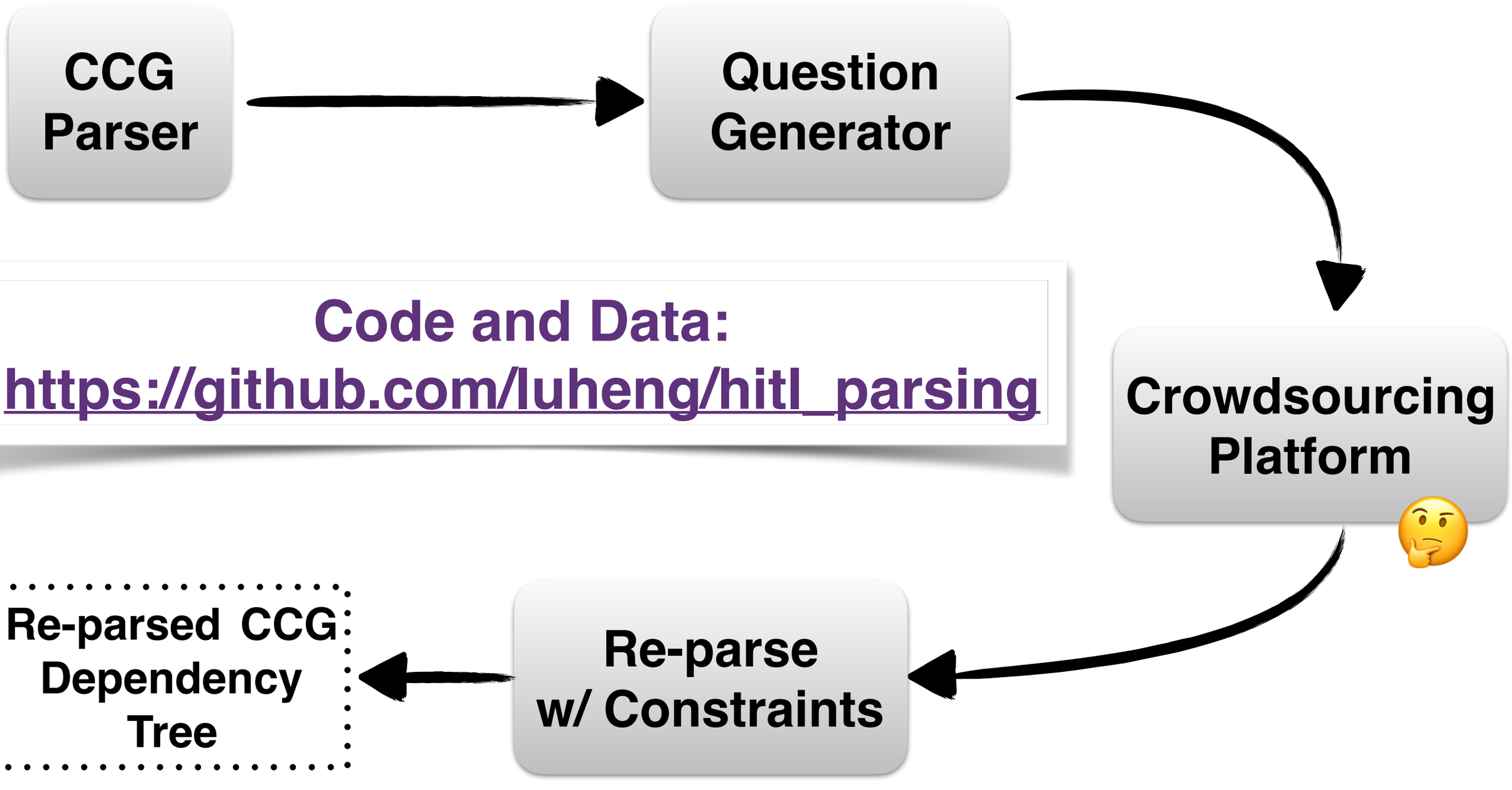
Future Work

- Improve coverage by adding new types of questions:
  - Modifiers: when, where, why …
  - PP attachments with natural language queries.
- Bootstrap a parser in a low-resource domain.
- Focus on downstream applications (e.g. Information Extraction).

Contributions

- Use non-expert annotation to improve a parser.
- Crowdsourced Q/A data for further exploration of active learning/reinforcement learning techniques.
- Code and data available online: https://github.com/luheng/hitl_parsing

Thank You!


CCG Parser
Question Generator
Crowdsourcing Platform
Code and Data:
https://github.com/luheng/hitl_parsing
Re-parsed CCG Dependency Tree
Re-parse w/ Constraints