Leave No Question Behind!

Broadening the Scope of Machine Comprehension

Daniel Khashabi
Allen Institute for AI

AlphaGo: The Success

AlphaGo: The Not-So-Successful Story

AlphaGo is incapable of solving any other problem in the world.

AlphaGo: The Not-So-Successful Story

- What would AlphaGo say if I ask it:

AlphaGo

AlphaGo is incapable of solving any other problem in the world.

AlphaGo: The Not-So-Successful Story

- What would AlphaGo say if I ask it:

Can you help me with my presentation?

??

AlphaGo

AlphaGo is incapable of solving any other problem in the world.

AlphaGo: The Not-So-Successful Story

- What would AlphaGo say if I ask it:

AlphaGo is incapable of solving any other problem in the world.

AlphaGo: The Not-So-Successful Story

- What would AlphaGo say if I ask it:

Can you help me with my presentation?

??

Can you play poker?

??

AlphaGo

Can you tell me about the game you're good at?

zzZ

AlphaGo is incapable of solving any other problem in the world.

AlphaGo: The Not-So-Successful Story

- What would AlphaGo say if I ask it:

Can you help me with my presentation?

Can you play poker?

Can you tell me about the game you're good at?

??

??

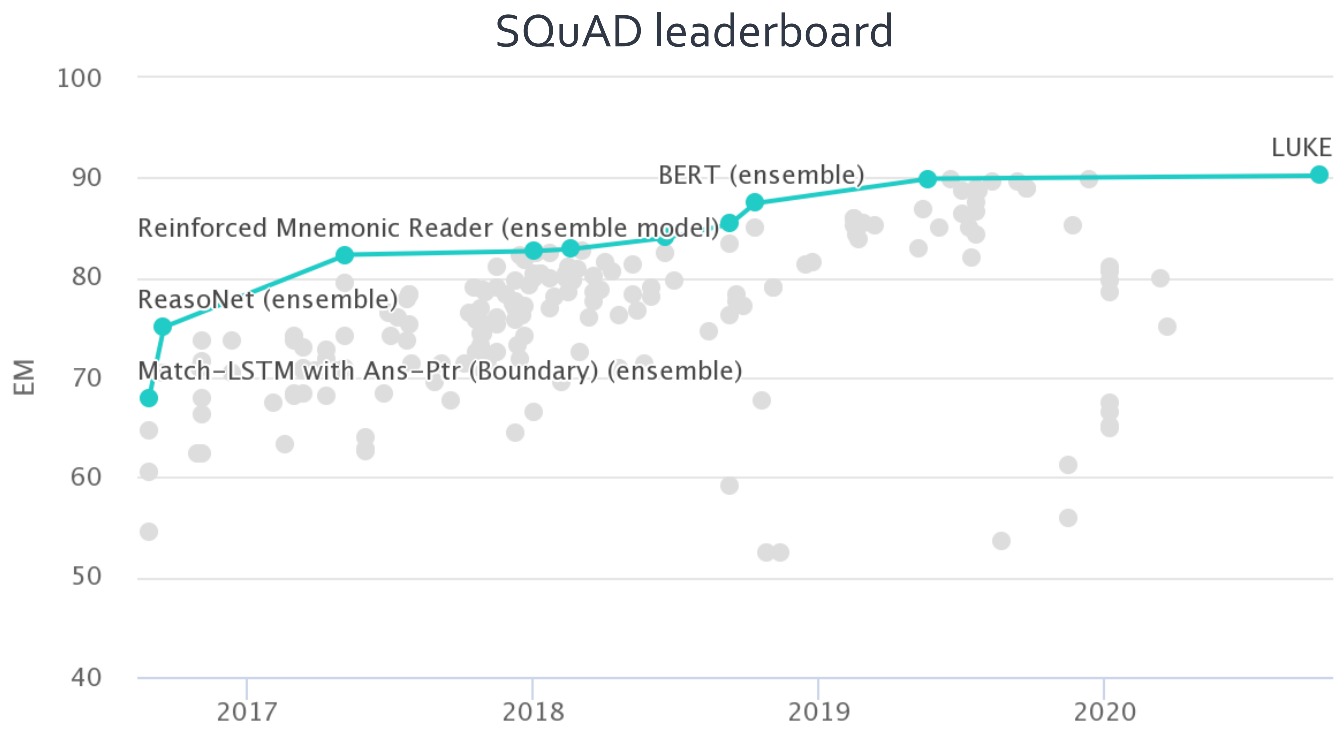
zzZ

AlphaGo

AlphaGo is incapable of solving any other problem in the world.

The Progress in NLP/QA

- Many benchmarks in NLP:
  - SQuAD [Rajpurkar et al. 2016]
  - ARC [Clark et al. 2018]
  - DROP [Dua et al. 2019]
  - …



Limits of Our Progress

- Successes in NLP are focused on niche domains

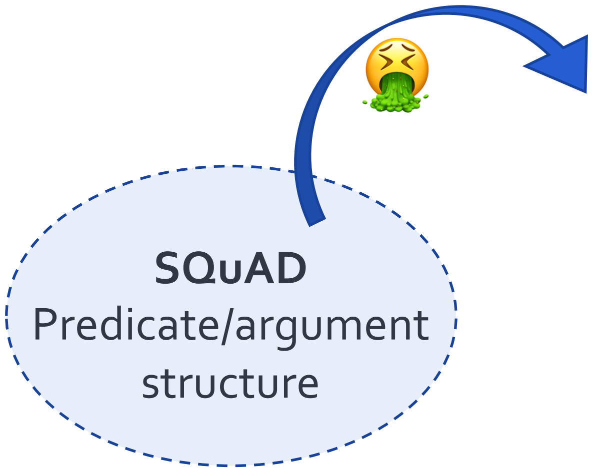
Limits of Our Progress

- Successes in NLP are focused on niche domains

SQuAD
Predicate/argument structure

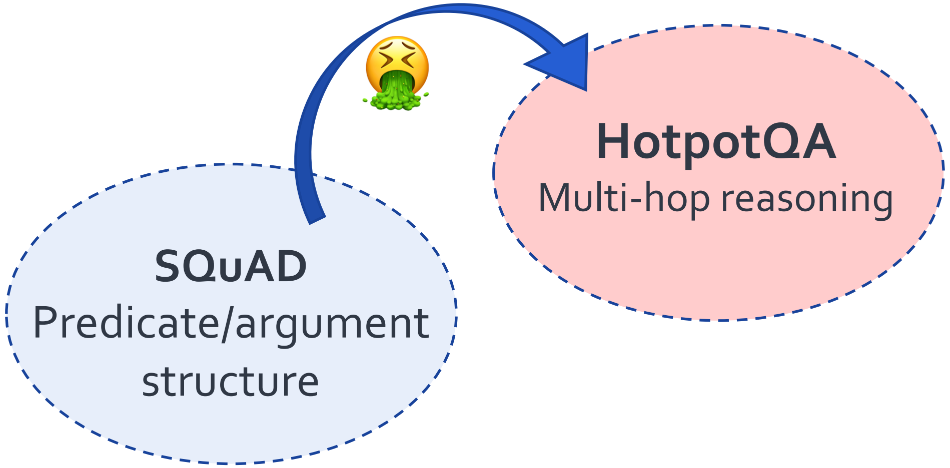
Limits of Our Progress

- Successes in NLP are focused on niche domains



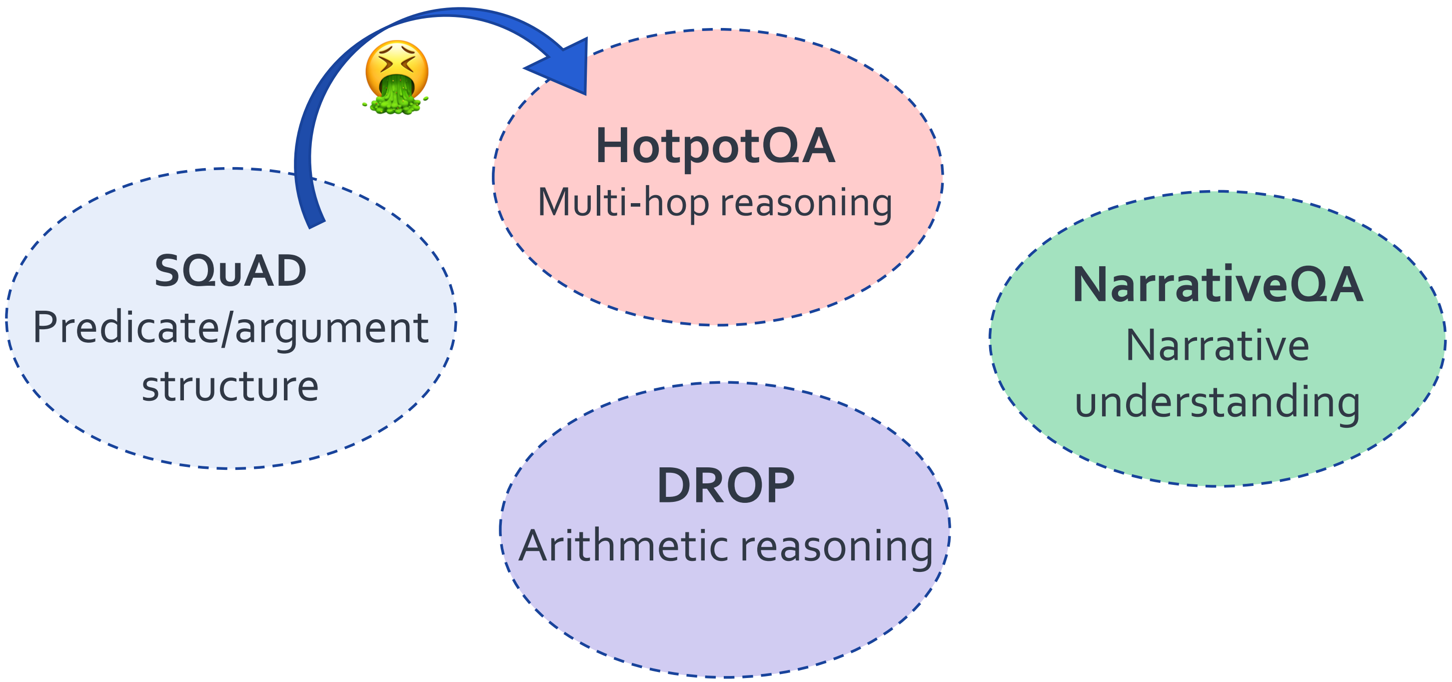
Limits of Our Progress

- Successes in NLP are focused on niche domains



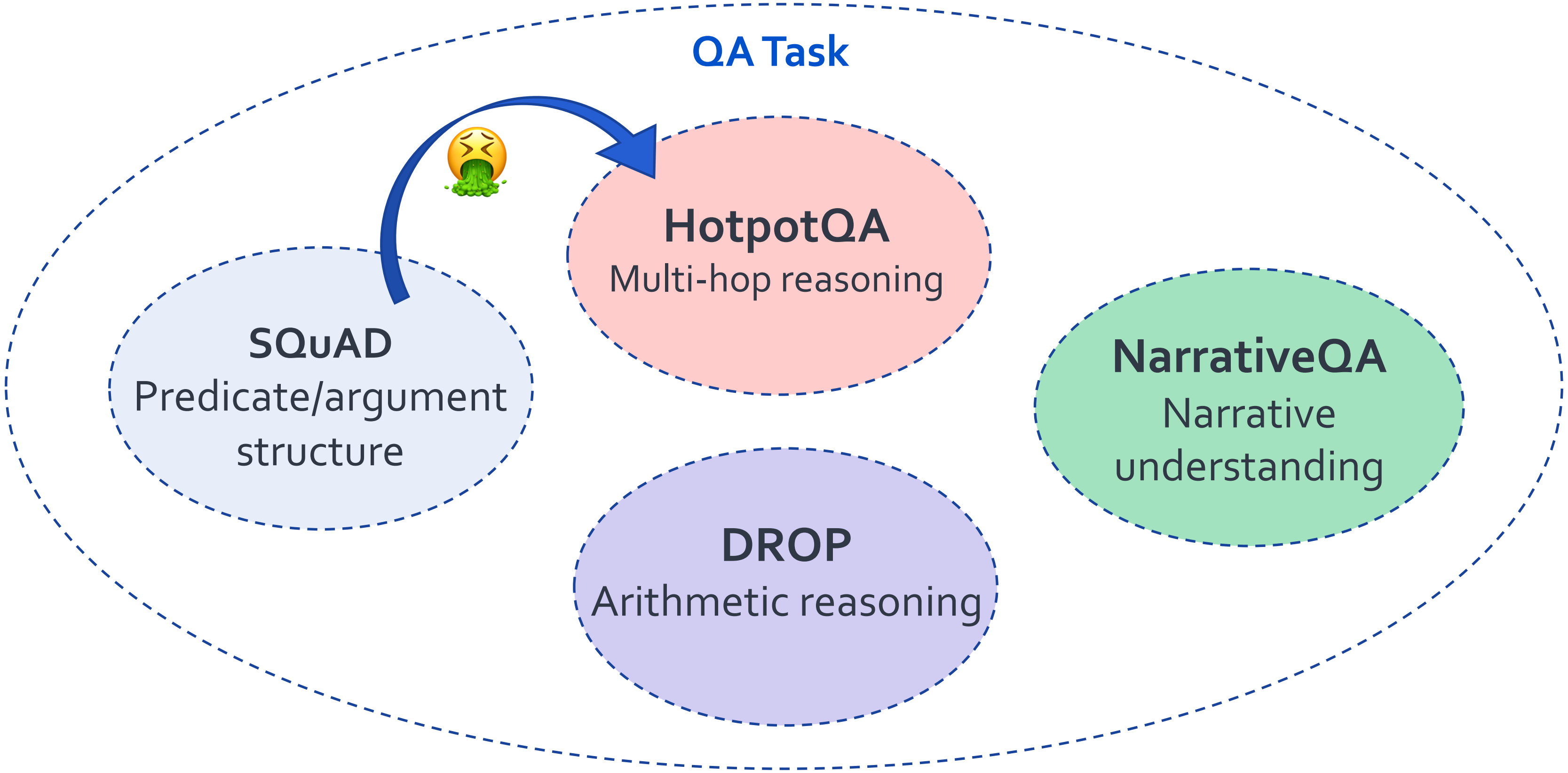
Limits of Our Progress

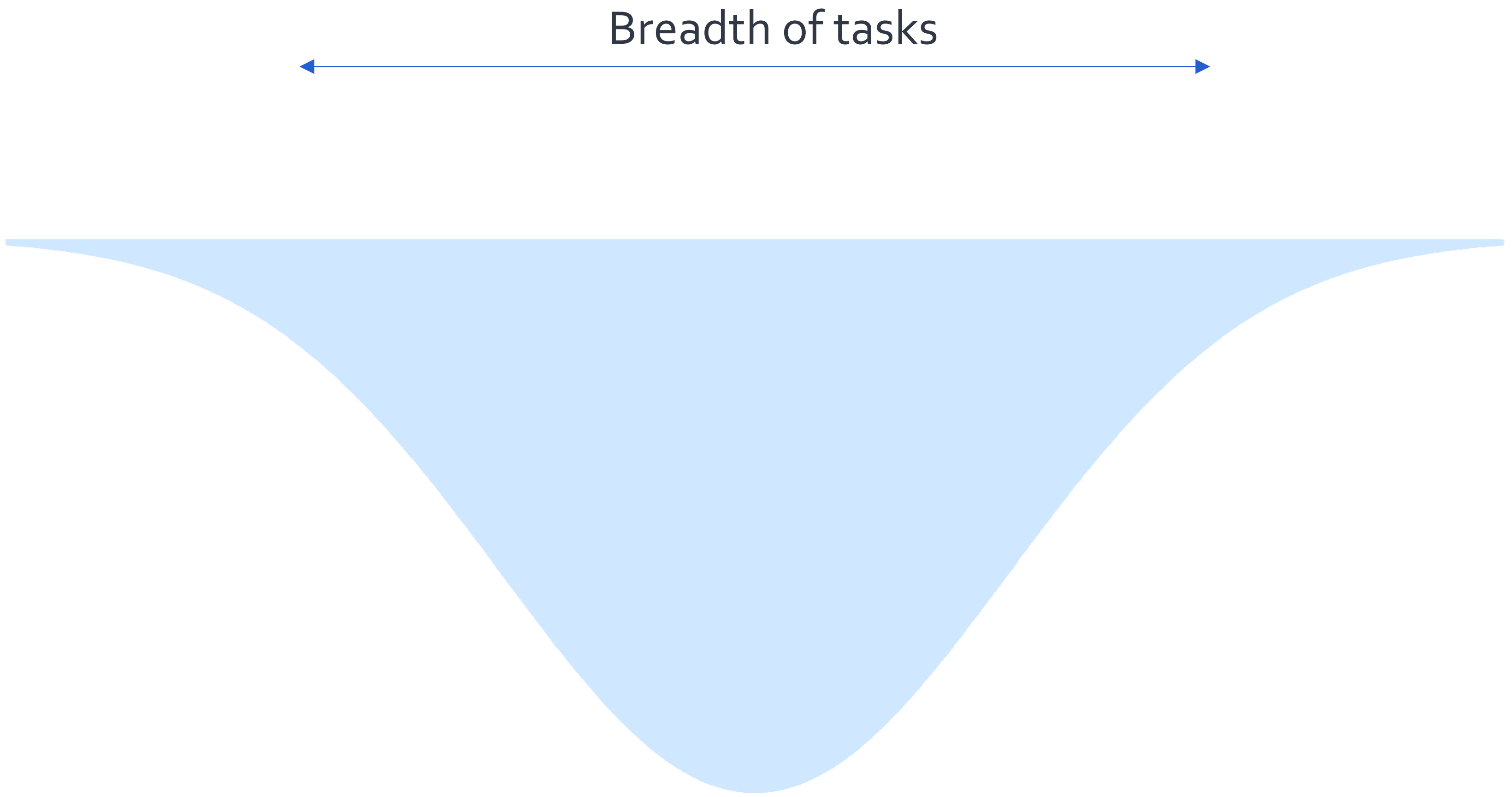
- Successes in NLP are focused on niche domains

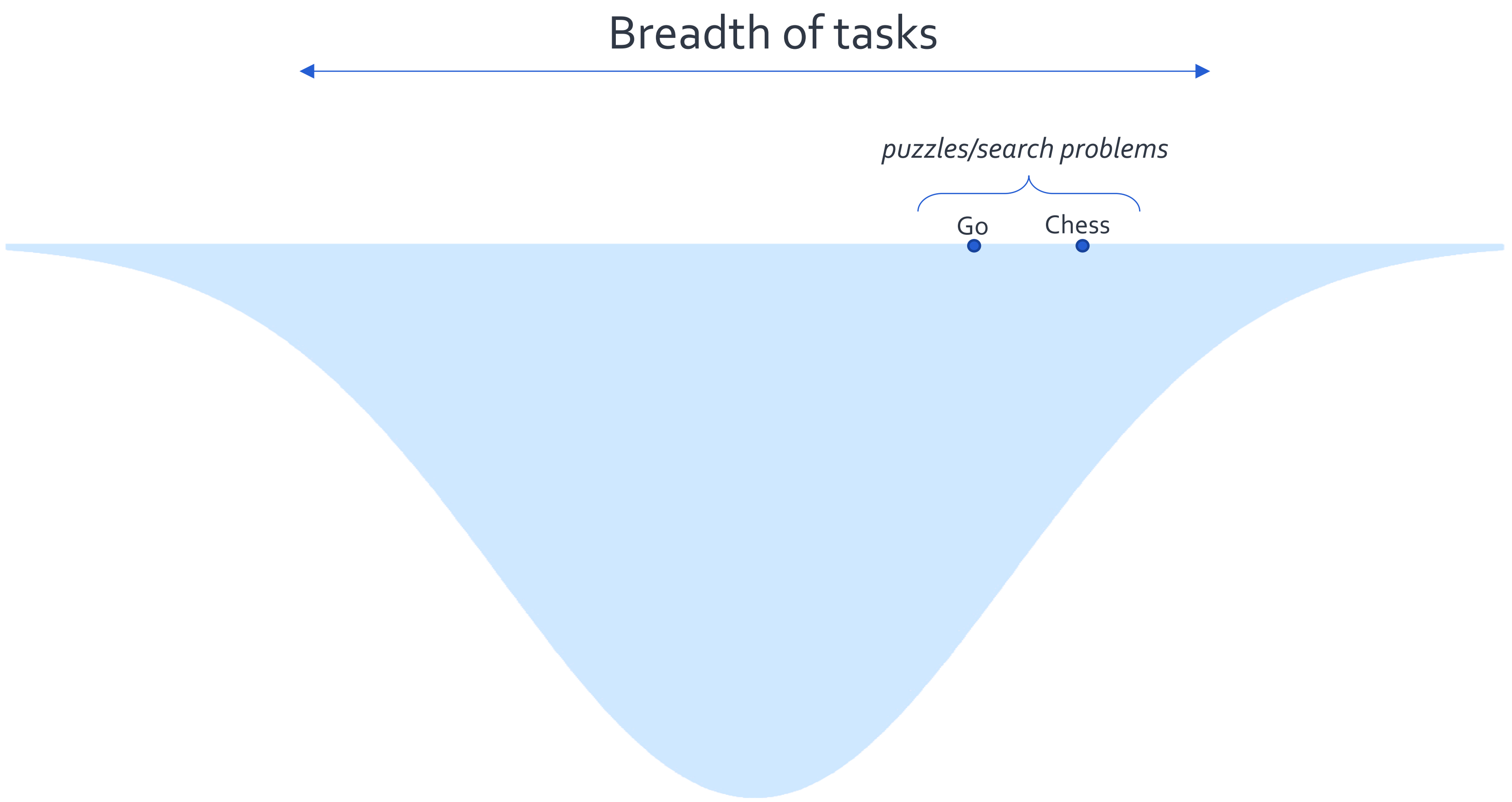


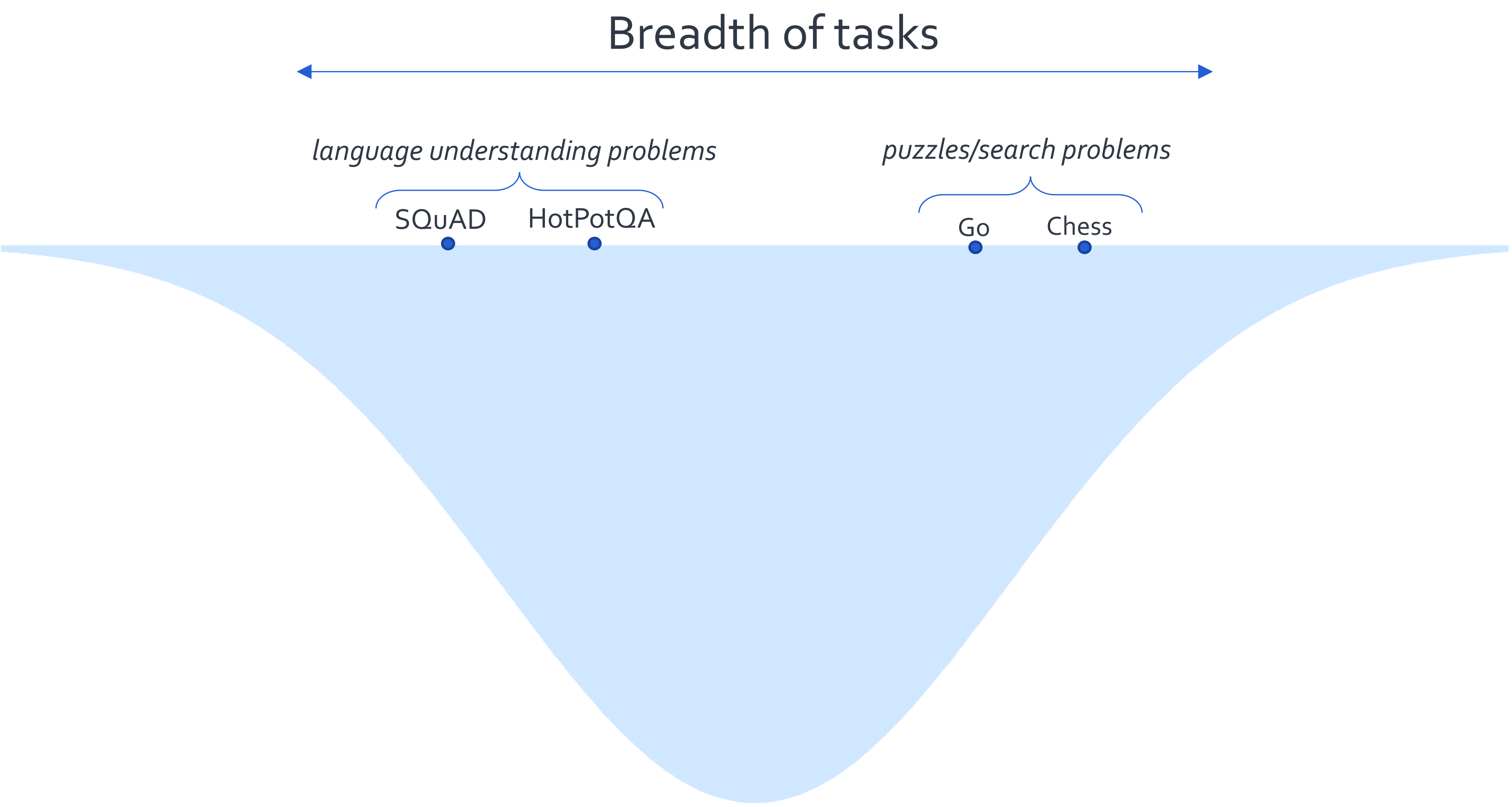
Limits of Our Progress

- Successes in NLP are focused on niche domains

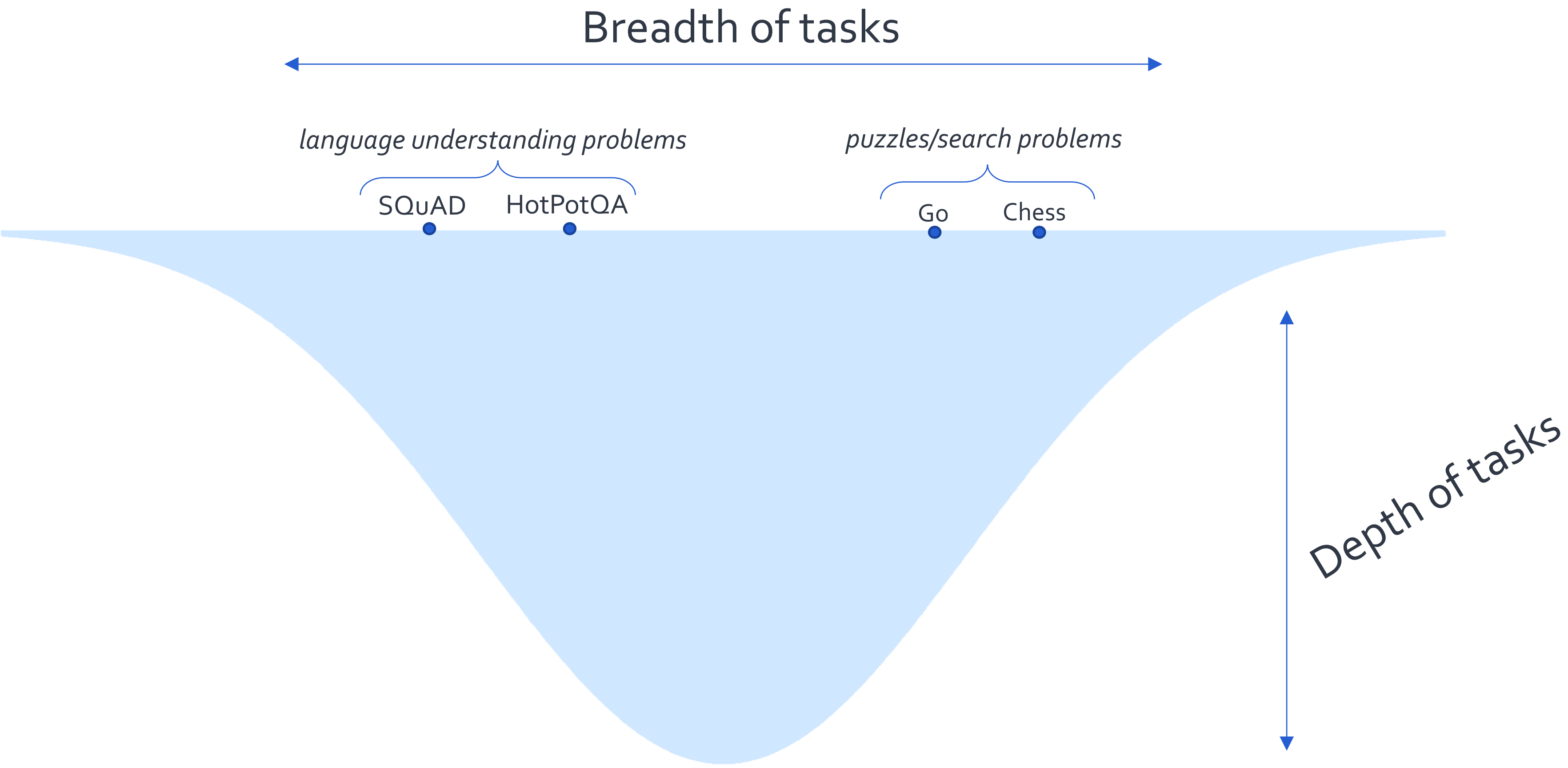


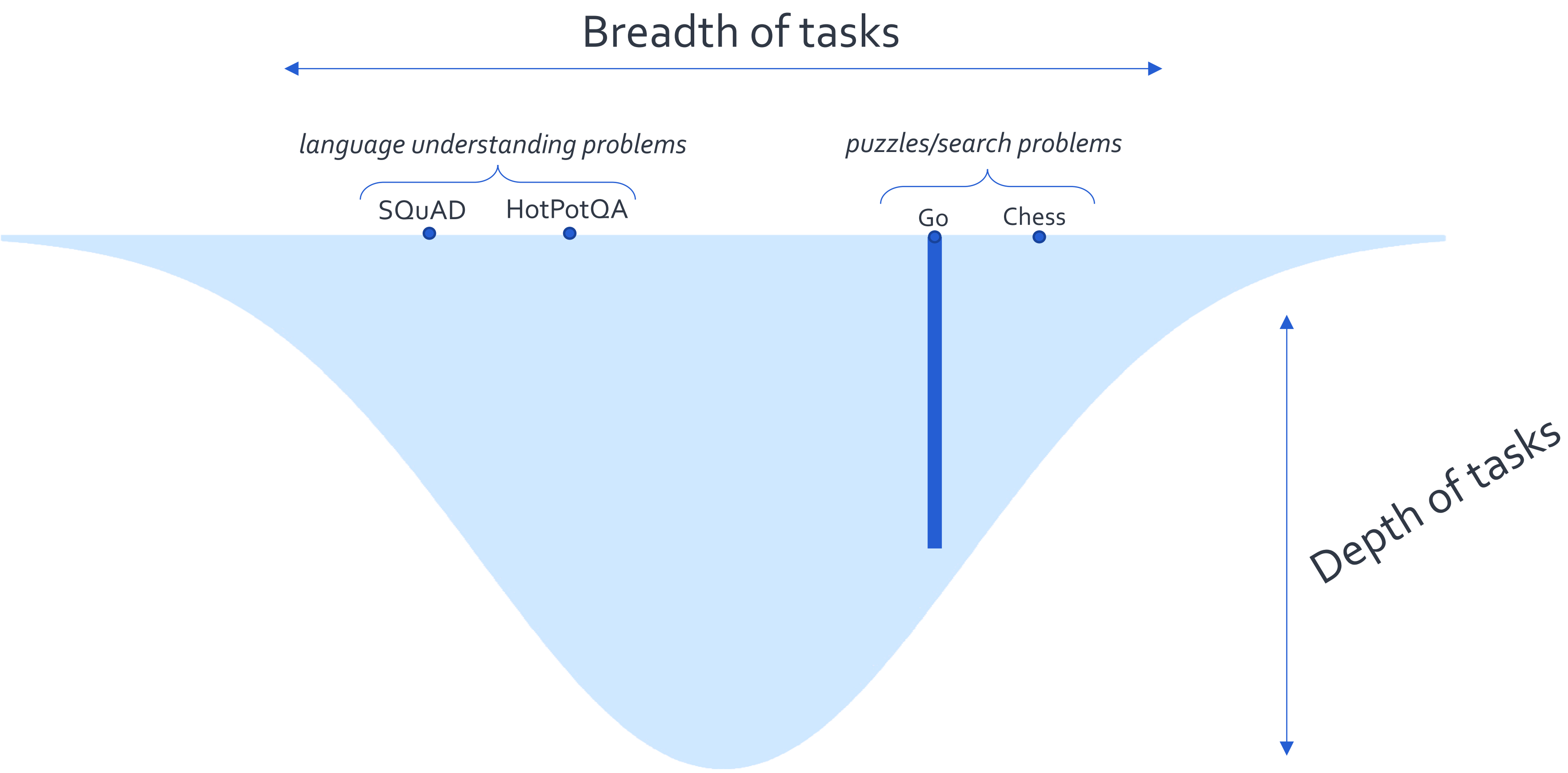

Breadth of tasks

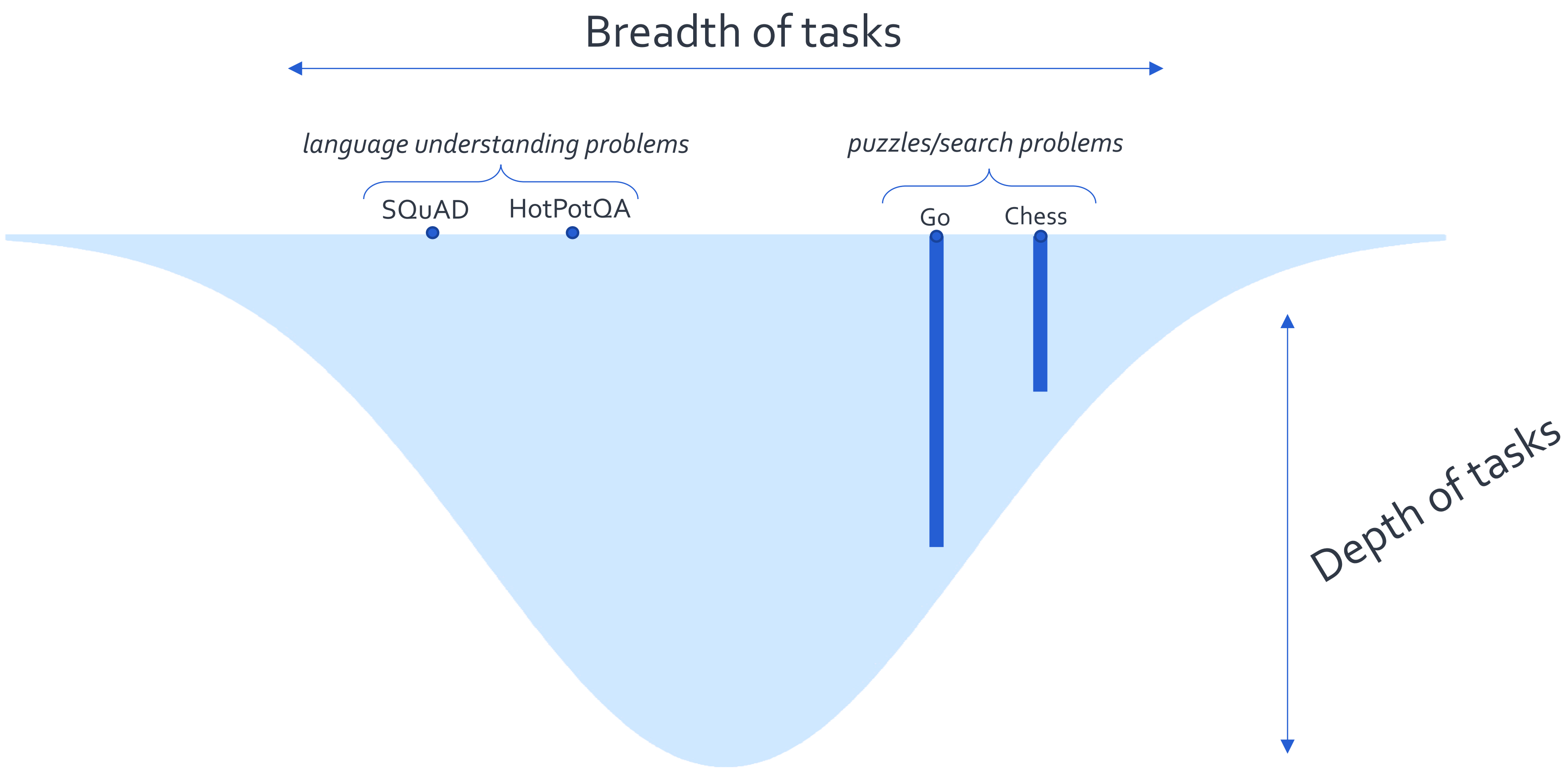

Breadth of tasks
puzzles/search problems
Go
Chess

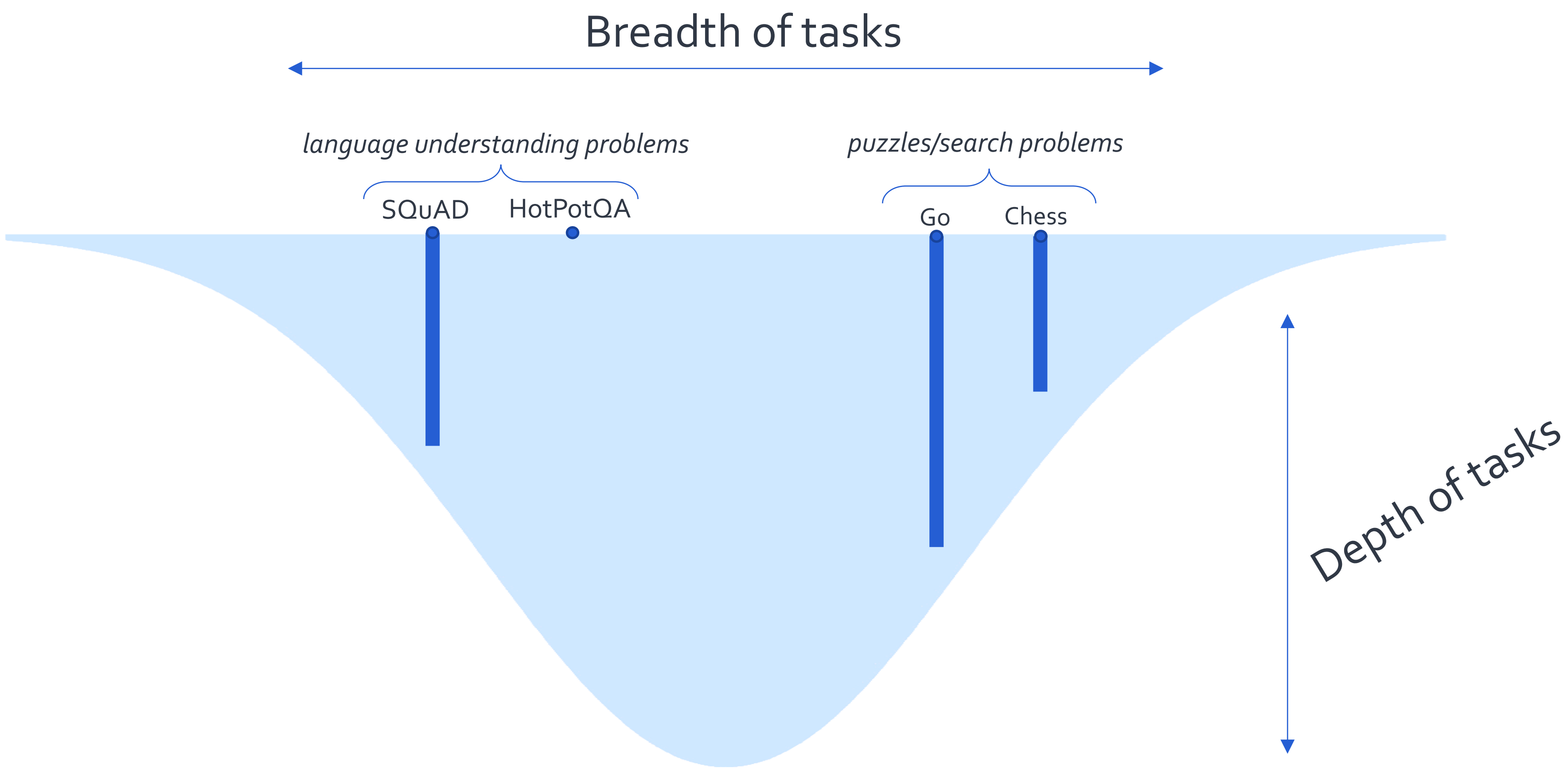


Breadth of tasks
language understanding problems
SQuAD
HotPotQA
puzzles/search problems
Go
Chess

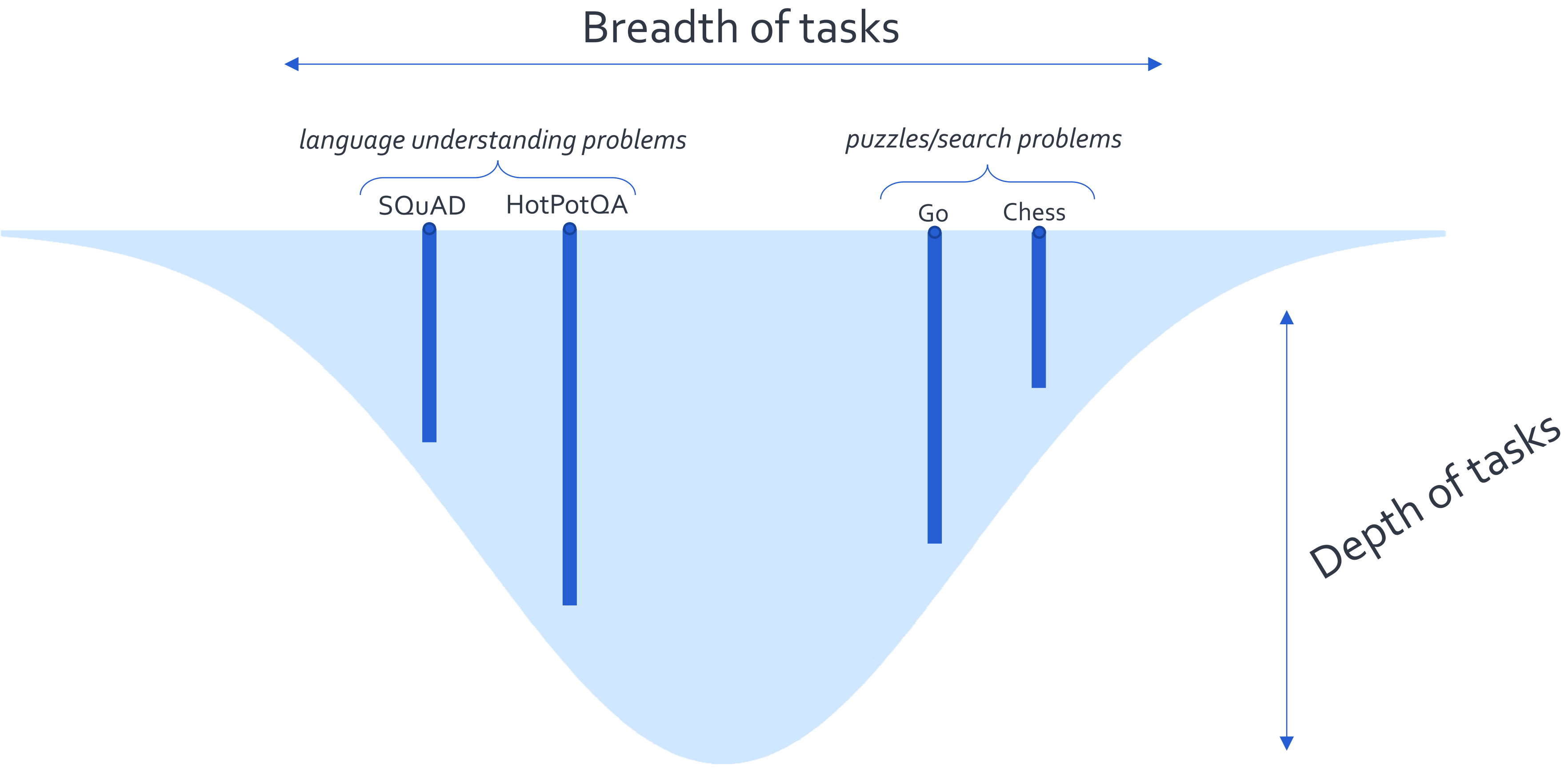

Breadth of tasks
language understanding problems
SQuAD
HotPotQA
puzzles/search problems
Go
Chess
Depth of tasks

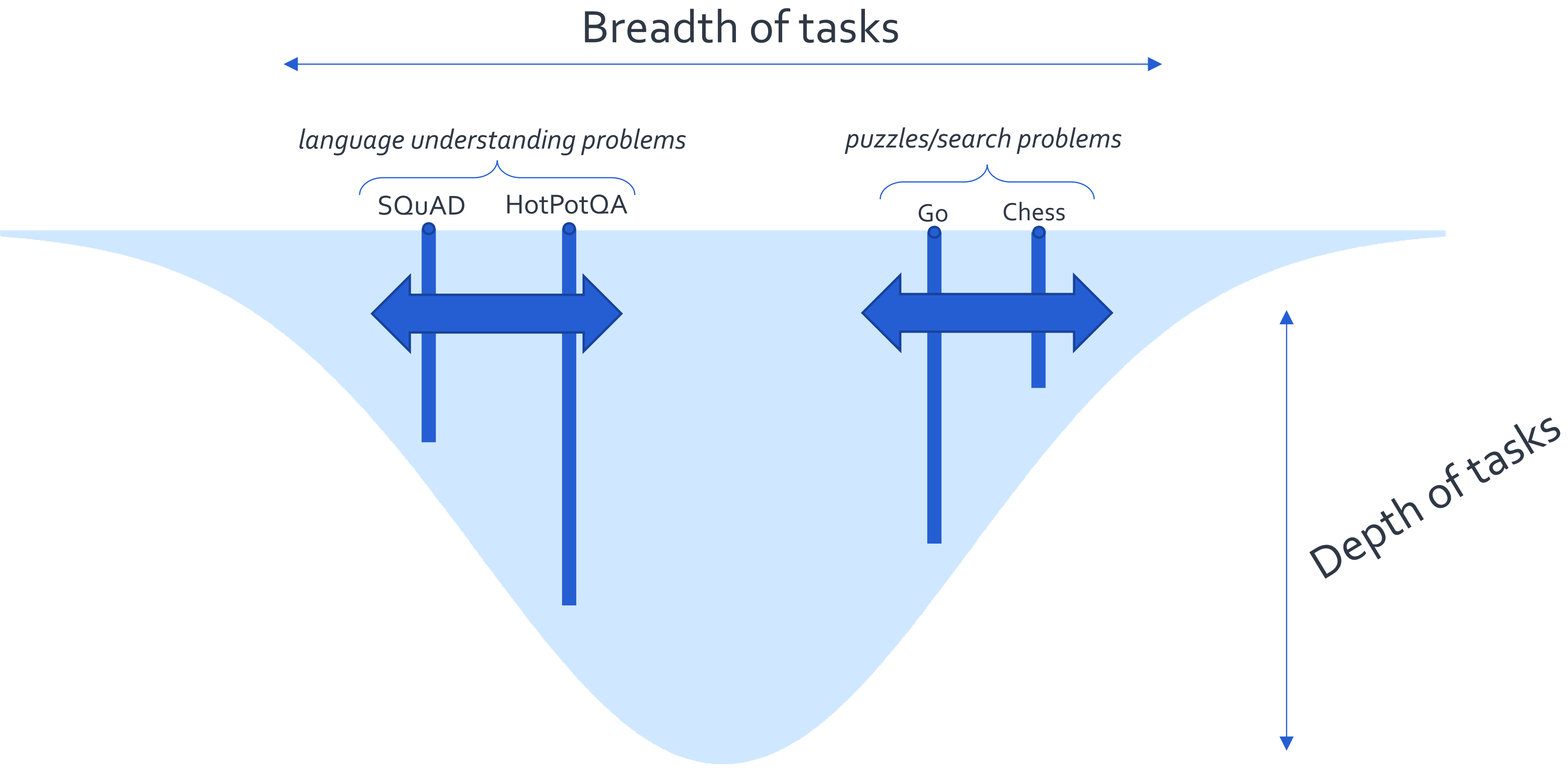

Breadth of tasks
language understanding problems
SQuAD
HotPotQA
puzzles/search problems
Go
Chess
Depth of tasks


Breadth of tasks
language understanding problems
puzzles/search problems
SQuAD
HotPotQA
Go
Chess
Depth of tasks



Breadth of tasks
language understanding problems
SQuAD
HotPotQA
puzzles/search problems
Go
Chess
Depth of tasks


Breadth of tasks
language understanding problems
puzzles/search problems
SQuAD
HotPotQA
Go
Chess
Depth of tasks


Breadth of tasks
language understanding problems
SQuAD
HotPotQA
puzzles/search problems
Go
Chess
Depth of tasks

This Talk

- In the current state of NLP field, we do **not** focus enough on the "breadth" of our progress.

This Talk

- In the current state of NLP field, we do **not** focus enough on the "breadth" of our progress.

Transfer Across Formats

This Talk

- In the current state of NLP field, we do **not** focus enough on the "breadth" of our progress.

Transfer Across Formats

Decomposing Complex Questions

This Talk

- In the current state of NLP field, we do **not** focus enough on the "breadth" of our progress.

Transfer Across Formats

Decomposing Complex Questions

This Talk

- In the current state of NLP field, we do **not** focus enough on the "breadth" of our progress.

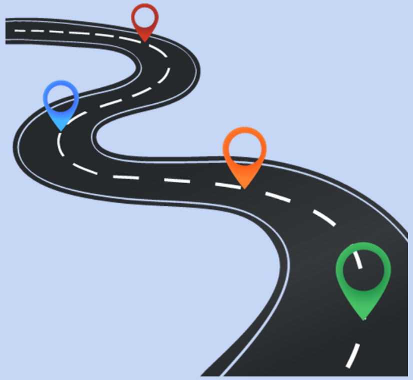
Transfer Across Formats

Decomposing Complex Questions

Broadening scope of QA

This Talk

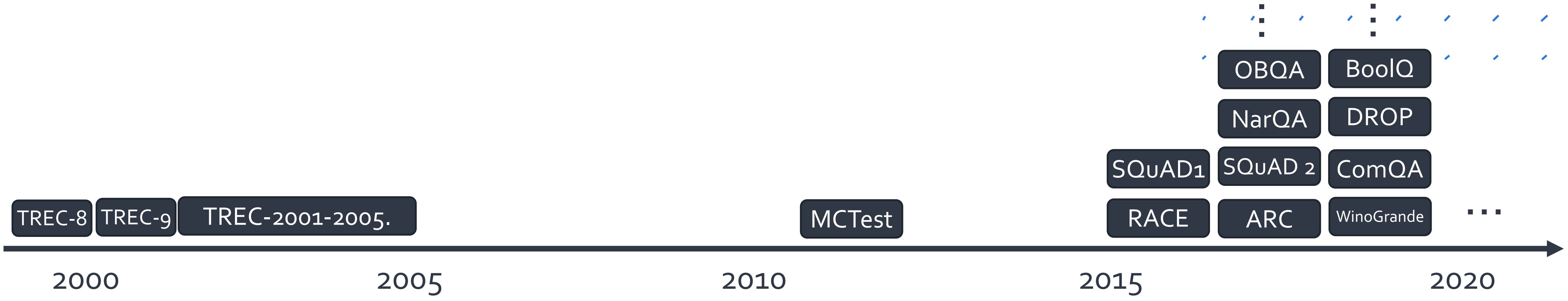
- ▣ Introduction
- ☐ Transfer Across Formats
- ☐ Decomposing Complex Q's
- ☐ Future Work



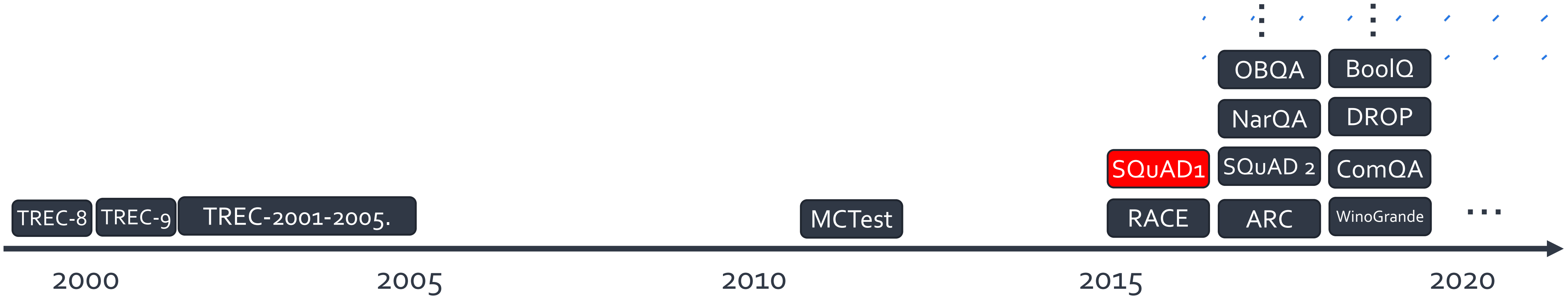
Transfer Across QA Formats

K et al. UnifiedQA: Crossing Format Boundaries With a Single QA System. EMNLP-Findings 20.

Many Flavors of QA

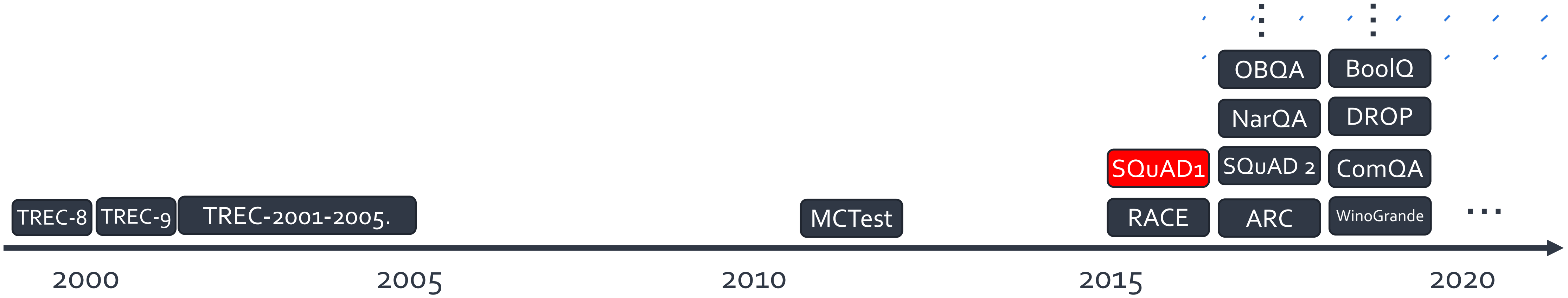


Many Flavors of QA



[Rajpurkar et al. 2016]

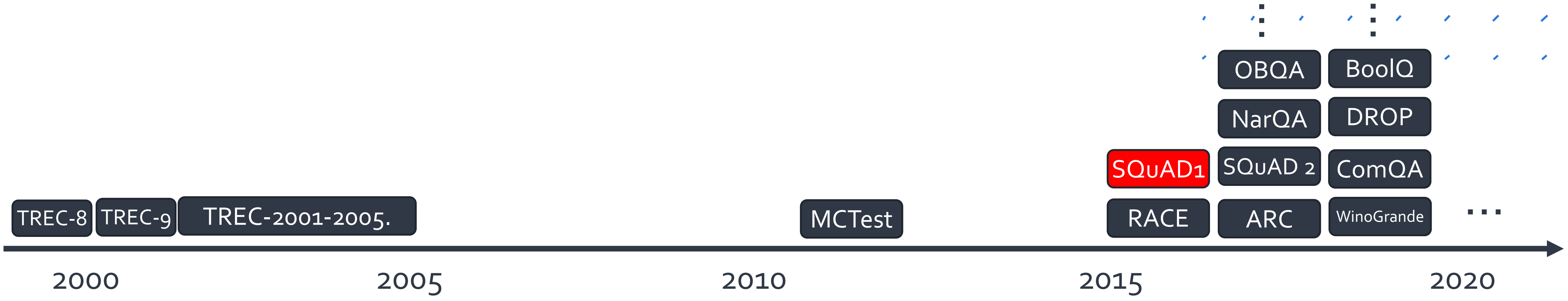
Many Flavors of QA



Question: *"At what speed did the turbine operate?"*

[Rajpurkar et al. 2016]

Many Flavors of QA

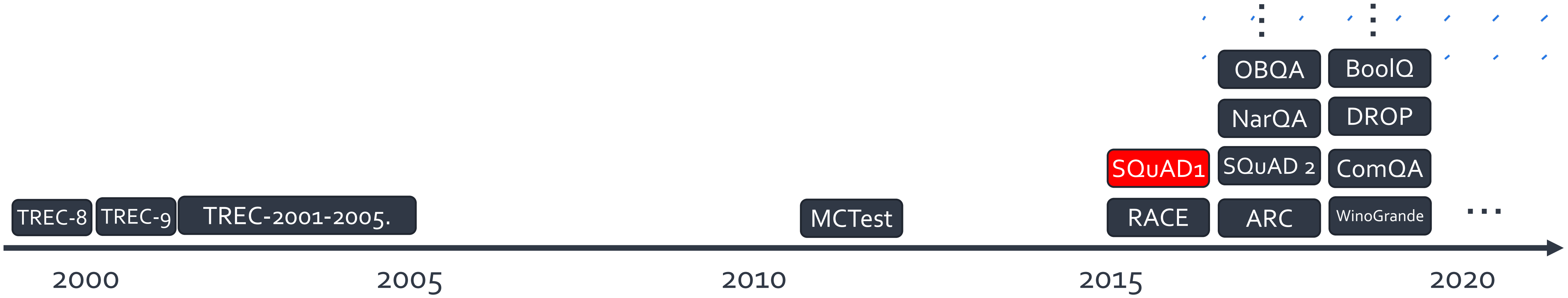


Question: *"At what speed did the turbine operate?"*

Candidates: *(Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...*

[Rajpurkar et al. 2016]

Many Flavors of QA

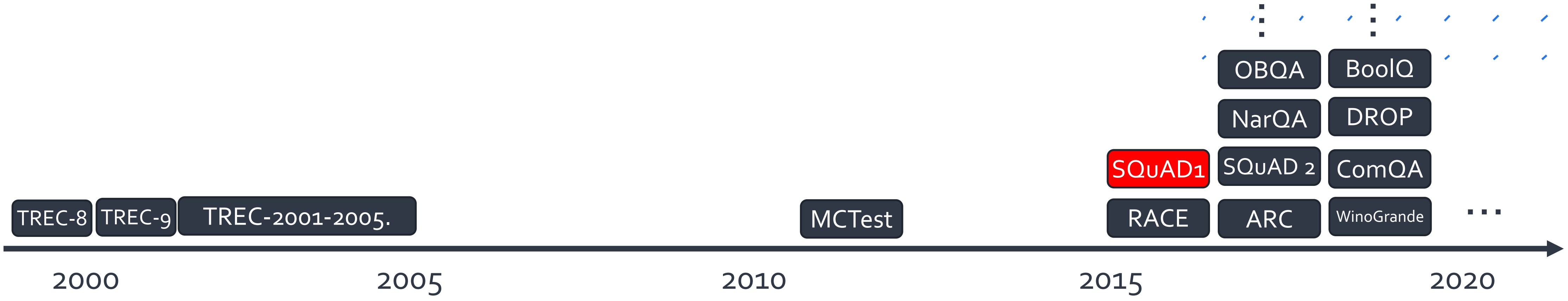


Question: *"At what speed did the turbine operate?"*

Candidates: *(Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...*

[Rajpurkar et al. 2016]

Many Flavors of QA



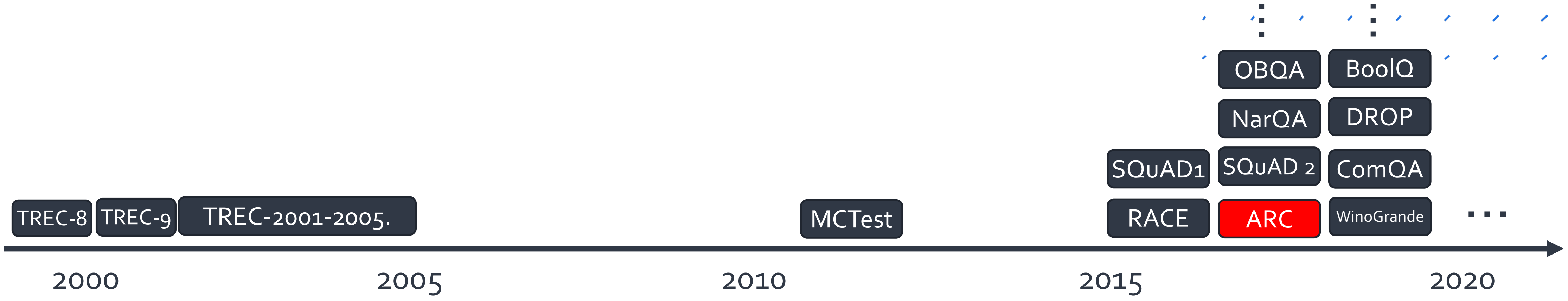
Question: *"At what speed did the turbine operate?"*

Candidates: *(Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...*

"16,000 rpm"

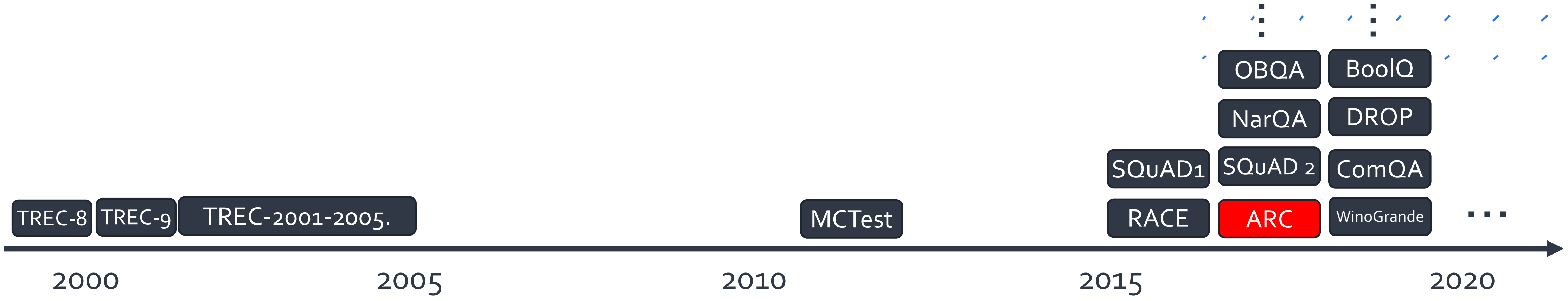
[Rajpurkar et al. 2016]

Many Flavors of QA



[Clark et al. 2018]

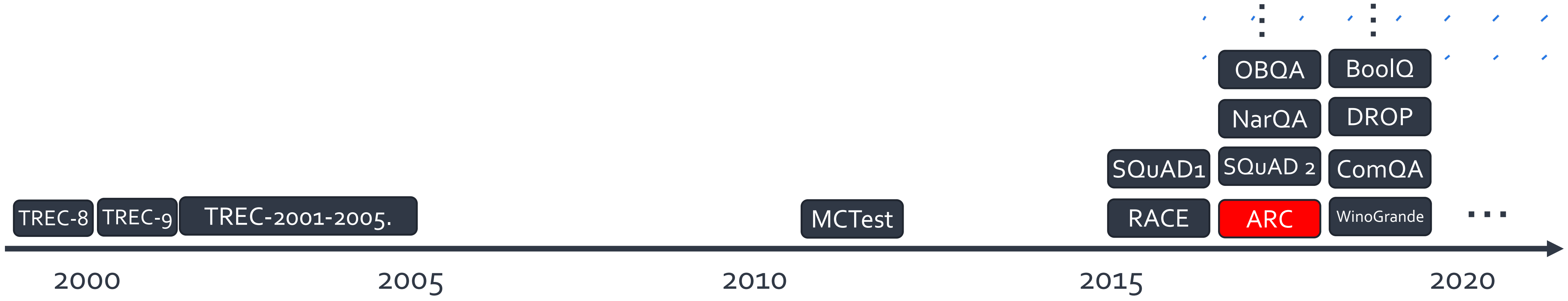
Many Flavors of QA



Question: "*What does photosynthesis produce that helps plants grow?* "

[Clark et al. 2018]

Many Flavors of QA

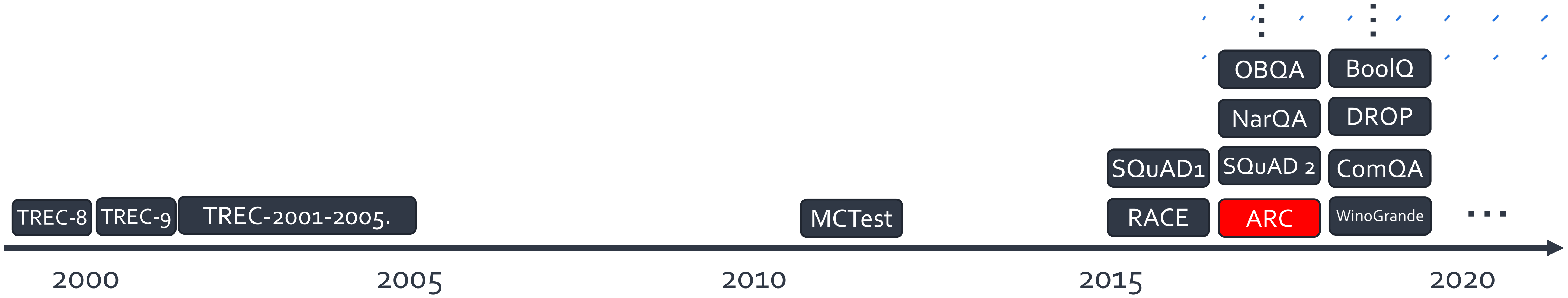


Question: "*What does photosynthesis produce that helps plants grow?* "

Candidates: *(A) water*
(B) oxygen
(C) protein
(D) sugar

[Clark et al. 2018]

Many Flavors of QA

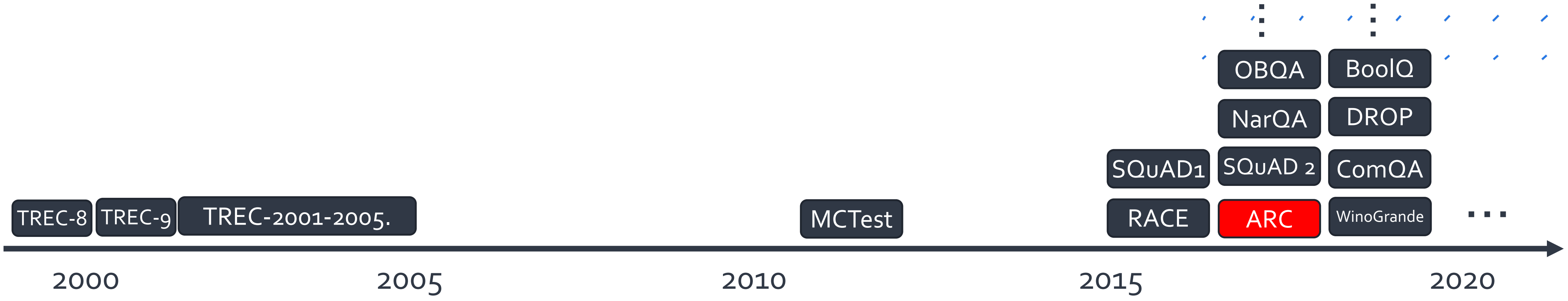


Question: "*What does photosynthesis produce that helps plants grow?*"

Candidates: *(A) water*
(B) oxygen
(C) protein
(D) sugar

[Clark et al. 2018]

Many Flavors of QA



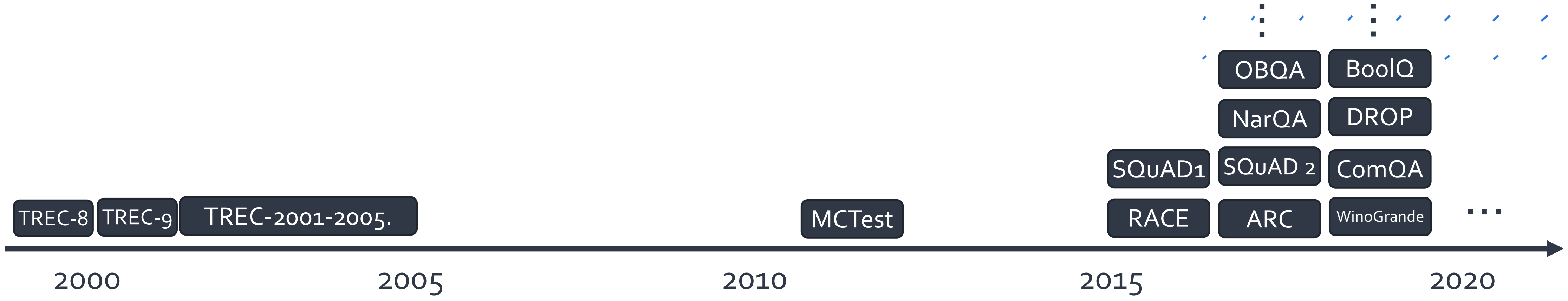
Question: "*What does photosynthesis produce that helps plants grow?* "

Candidates: *(A) water*
(B) oxygen
(C) protein
(D) sugar

"The big kid"

[Clark et al. 2018]

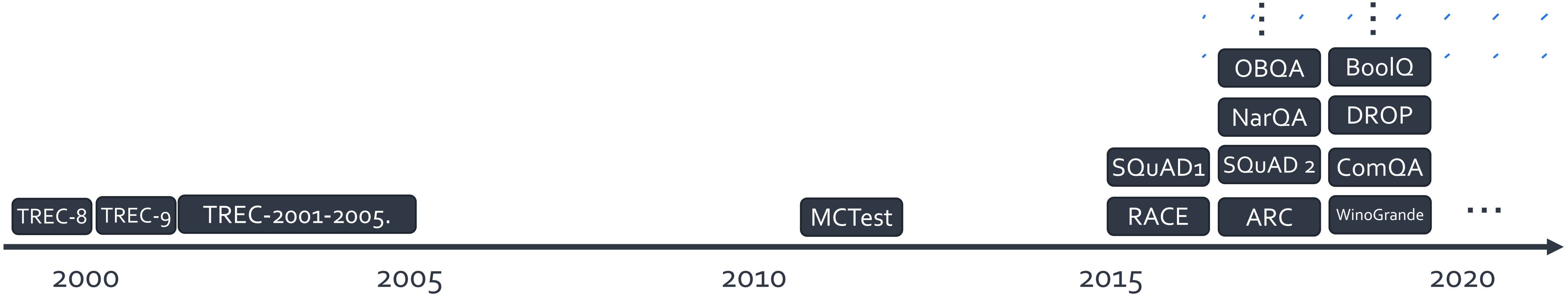
Many Flavors of QA



- Motivations for publishing new datasets:
  - Unexplored reasoning challenges
  - Alternate (better?) evaluation protocols

*But inherently they're all **QA**!*

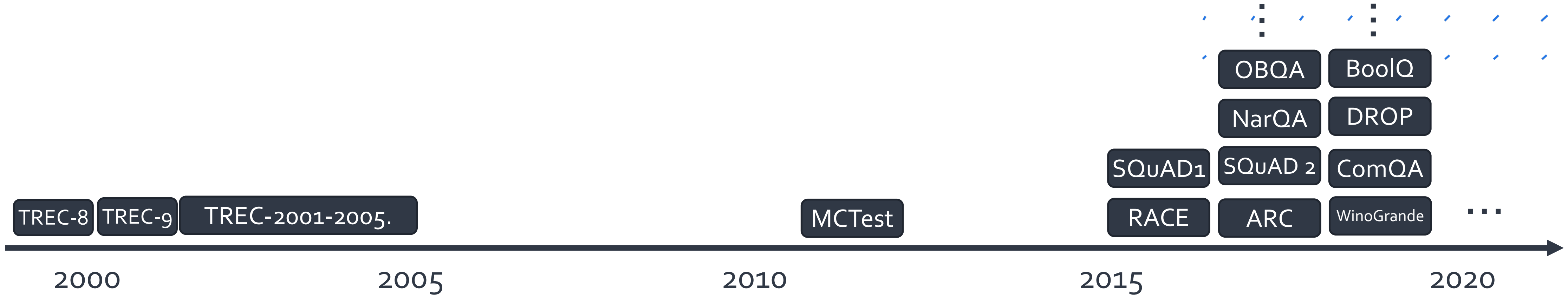
Many Flavors of QA



- Motivations for publishing new datasets:
  - Unexplored reasoning challenges
  - Alternate (better?) evaluation protocols

*But inherently they're all **QA**!*

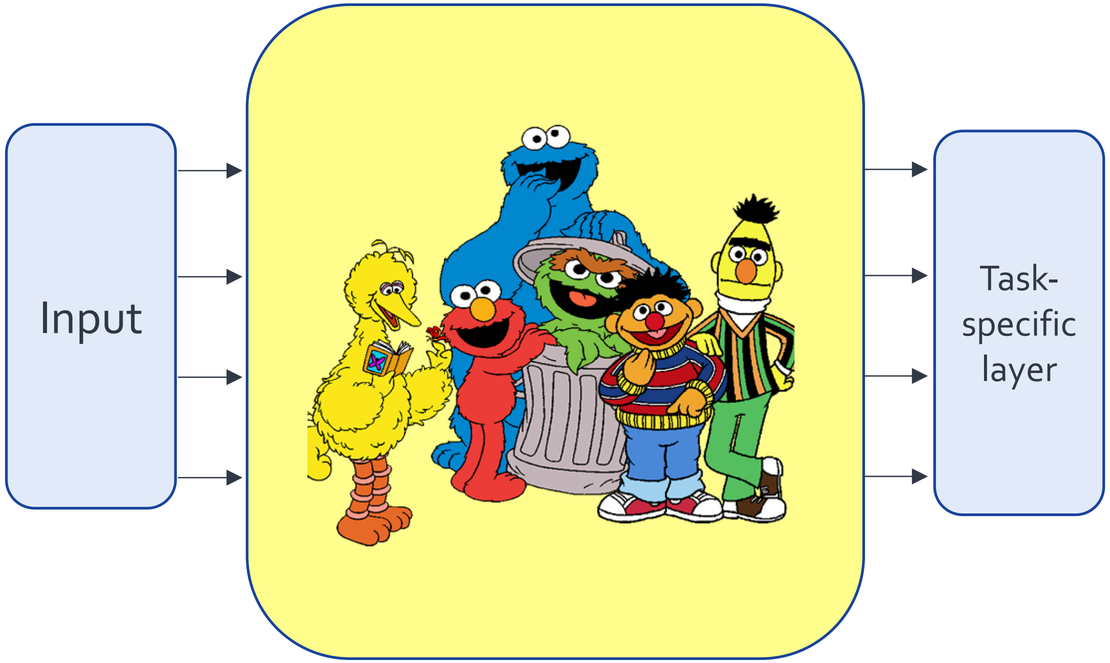
Many Flavors of QA



- Motivations for publishing new datasets:
  - Unexplored reasoning challenges
  - Alternate (better?) evaluation protocols

*But inherently they're all **QA**!*

Format-Specific Model Design



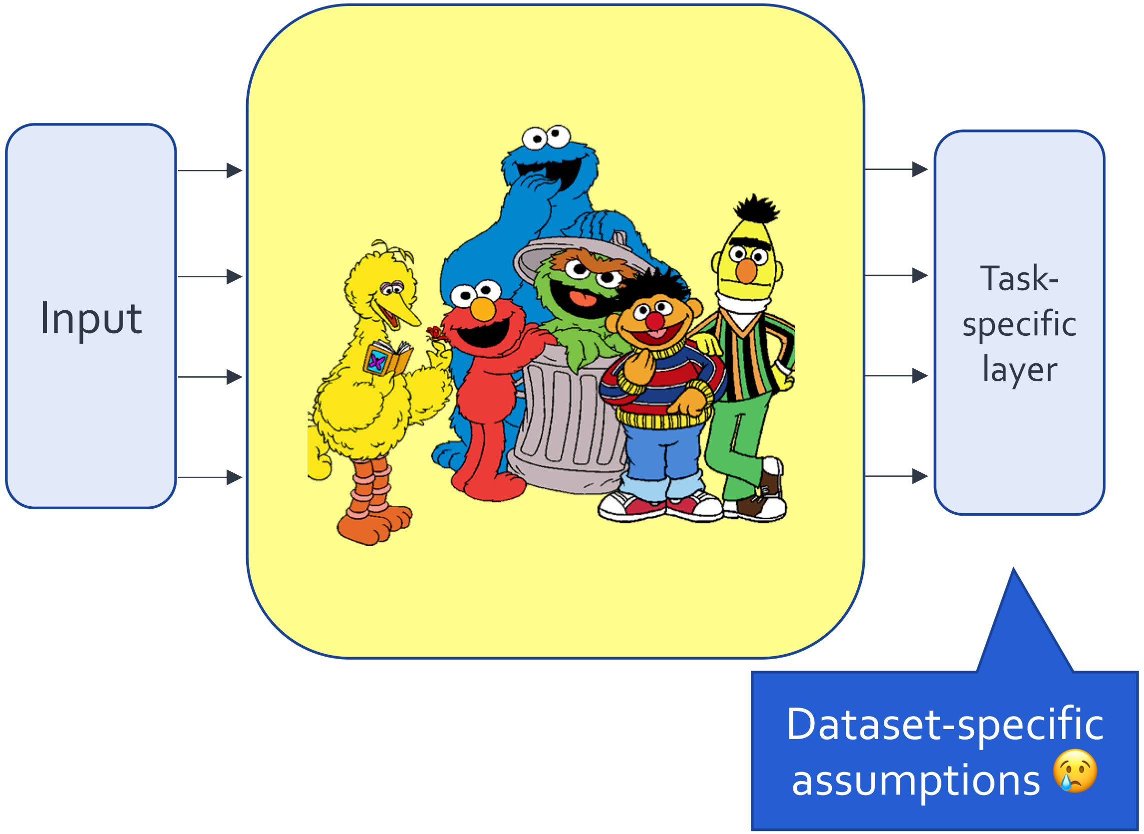
Format-Specific Model Design

Input

Task-specific layer

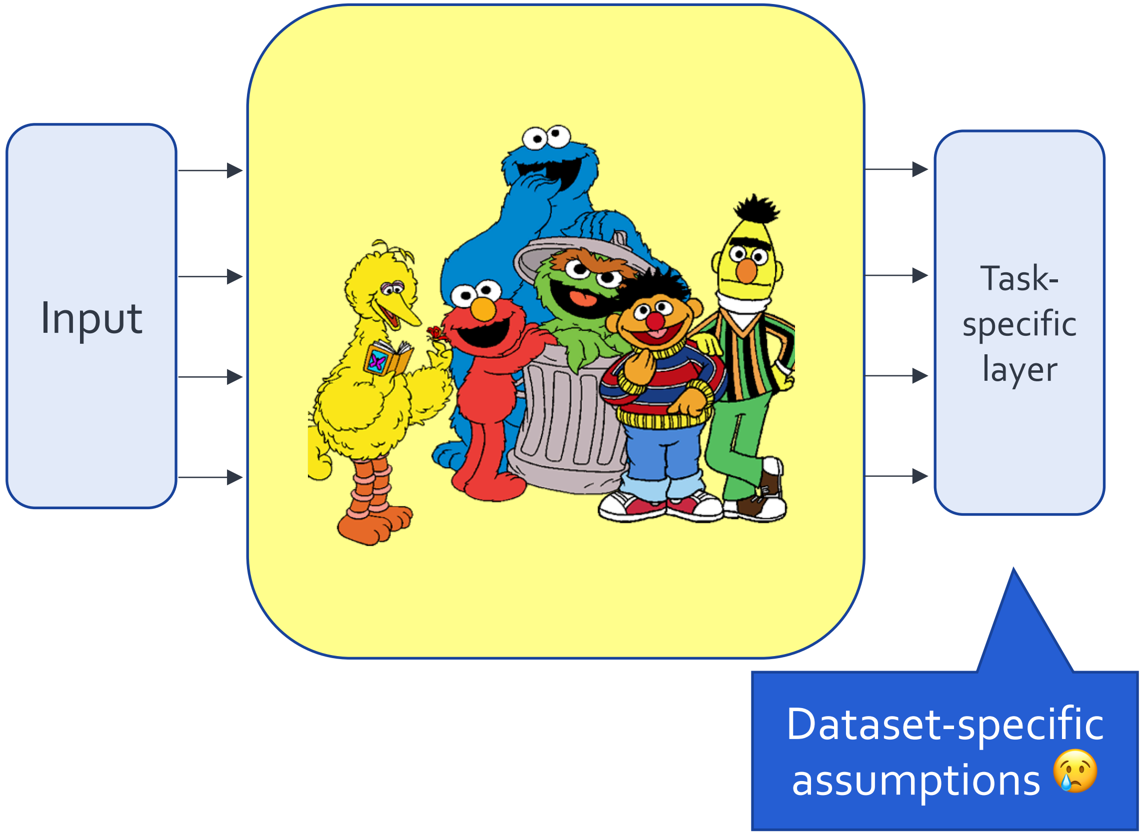
Dataset-specific assumptions 😢

Format-Specific Model Design



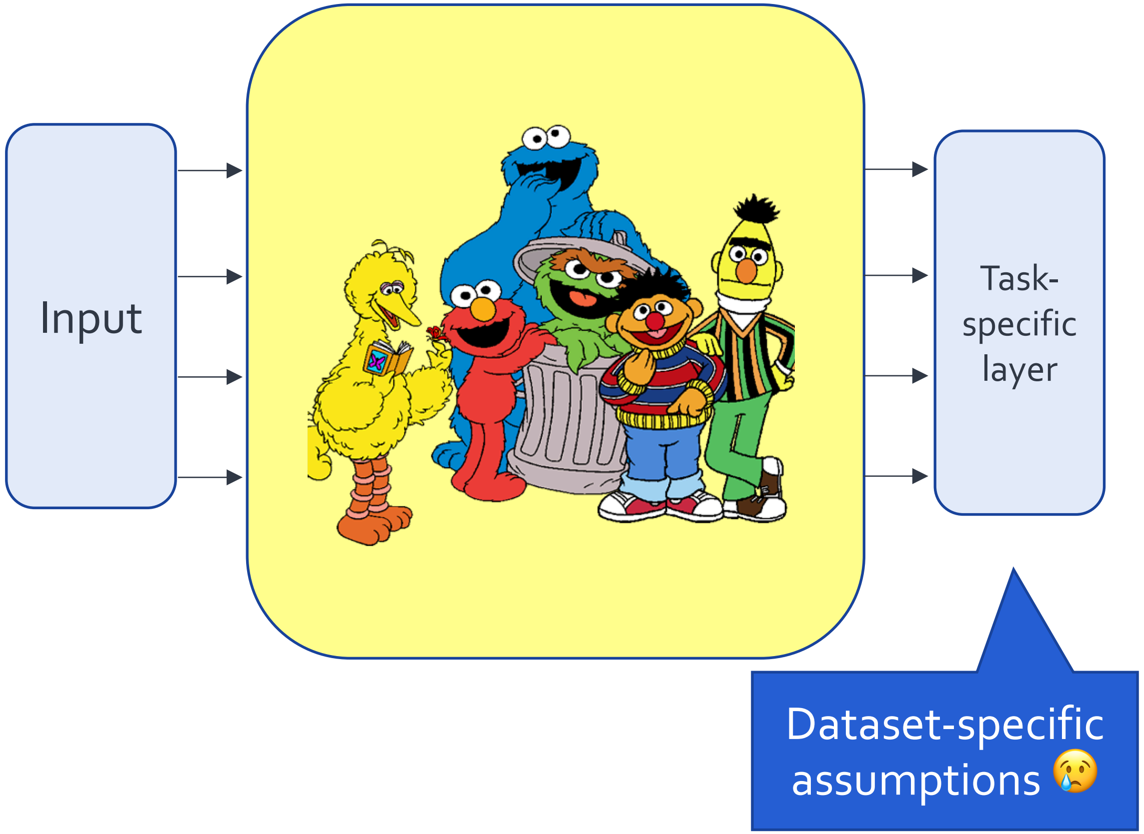
format	assumption
Yes/No QA	
Multiple-choice QA	
Extractive QA	
Abstractive QA	

Format-Specific Model Design



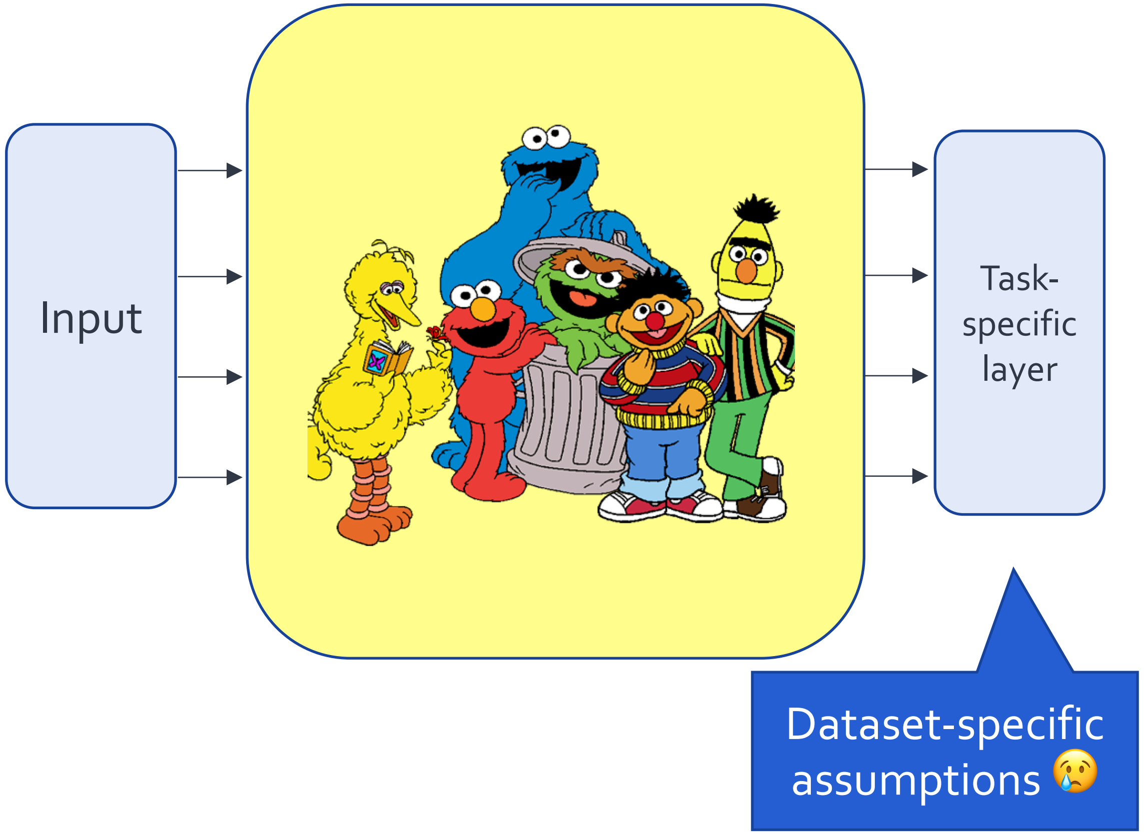
format	assumption
Yes/No QA	*binary output*
Multiple-choice QA	
Extractive QA	
Abstractive QA	

Format-Specific Model Design



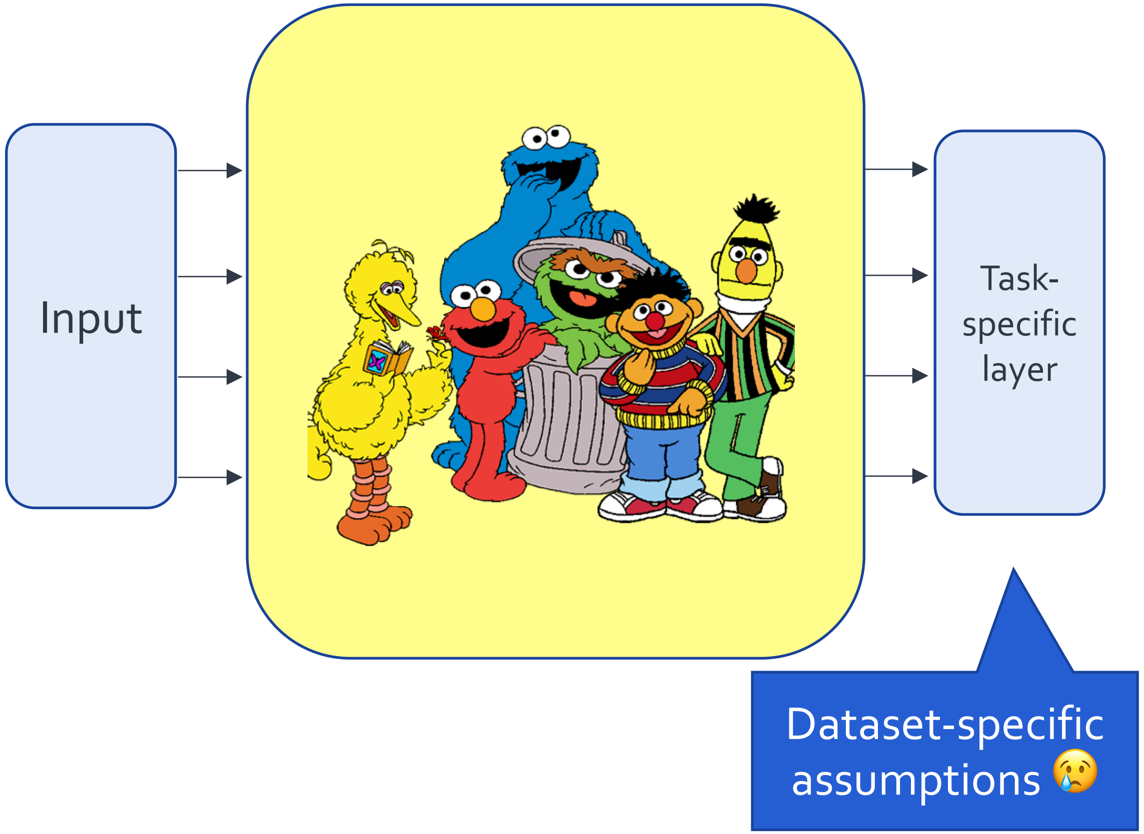
format	assumption
Yes/No QA	*binary output*
Multiple-choice QA	*One correct answer from a list of candidates.*
Extractive QA	
Abstractive QA	

Format-Specific Model Design



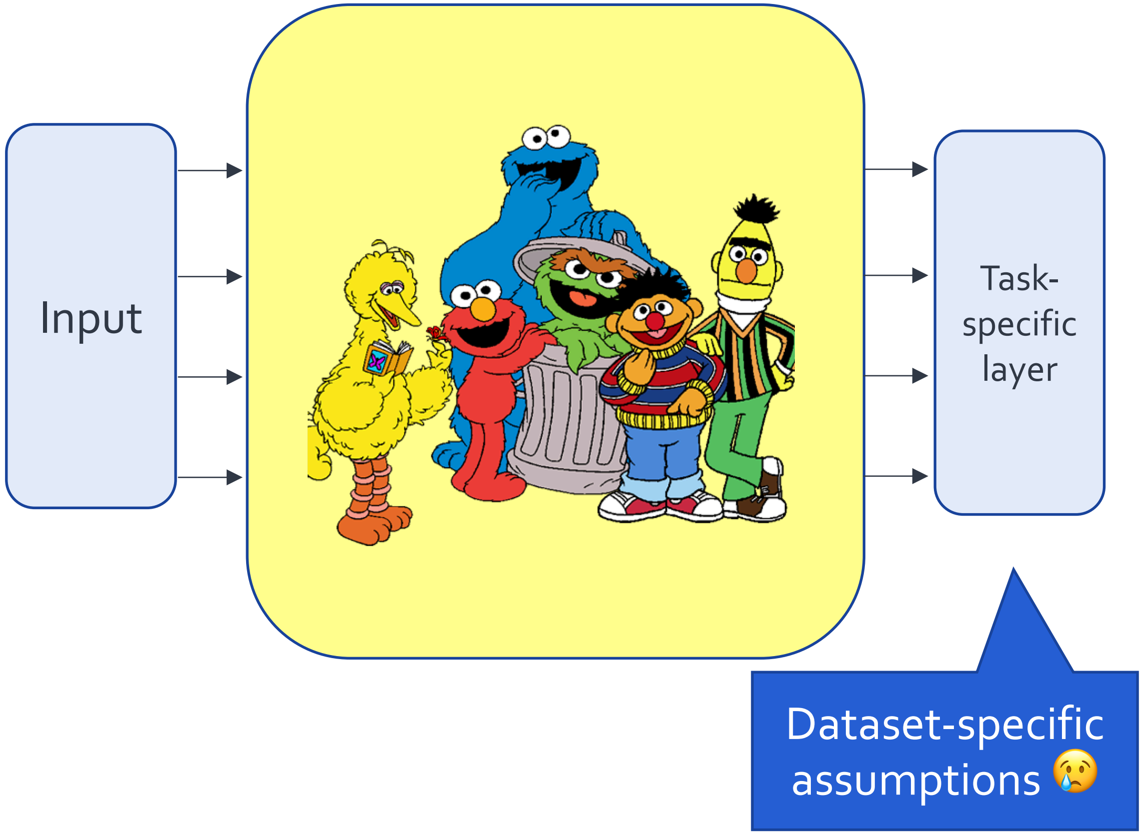
format	assumption
Yes/No QA	*binary output*
Multiple-choice QA	*One correct answer from a list of candidates.*
Extractive QA	*answer is a substring of paragraph*
Abstractive QA	

Format-Specific Model Design



format	assumption
Yes/No QA	*binary output*
Multiple-choice QA	*One correct answer from a list of candidates.*
Extractive QA	*answer is a substring of paragraph*
Abstractive QA	*answer to be inferred from the paragraph*

Format-Specific Model Design



Consequences of format-specific designs:

- **Prevent generalization** across formats
- **Don't benefit** from labeled data of other formats

format	assumption
Yes/No QA	*binary output*
Multiple-choice QA	*One correct answer from a list of candidates.*
Extractive QA	*answer is a substring of paragraph*
Abstractive QA	*answer to be inferred from the paragraph*

Format-Specific Model Design (2)

ExtractiveQA

MultipleChoiceQA

Format-Specific Model Design (2)

ExtractiveQA

Question: *"At what speed did the turbine operate?"*

(Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...

"16,000 rpm"

MultipleChoiceQA

Format-Specific Model Design (2)

ExtractiveQA

Question: *"At what speed did the turbine operate?"*

(Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...

"16,000 rpm"

MultipleChoiceQA

Question: *"What does photosynthesis produce that helps plants grow?"*

(A) water
(B) oxygen
(C) protein
(D) sugar

"sugar"

Format-Specific Model Design (2)

ExtractiveQA

Question: "*At what speed did the turbine operate?*"

(Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. …

"16,000 rpm"

MultipleChoiceQA

Question: "*What does photosynthesis produce that helps plants grow?*"

(A) water
(B) oxygen
(C) protein
(D) sugar

"sugar"

Format-Specific Model Design (2)

ExtractiveQA

Question: *"At what speed did the turbine operate?"*

(Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...

"16,000 rpm"

MultipleChoiceQA

Question: *"What does photosynthesis produce that helps plants grow?"*

(A) water
(B) oxygen
(C) protein
(D) sugar

"sugar"

Format-Specific Model Design (2)

ExtractiveQA

Question: *"At what speed did the turbine operate?"*

(Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...

"16,000 rpm"

MultipleChoiceQA

Question: *"What does photosynthesis produce that helps plants grow?"*

(A) water
(B) oxygen
(C) protein
(D) sugar

"sugar"

Format-Specific Model Design (2)

ExtractiveQA

Question: *"At what speed did the turbine operate?"*

(Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...

"16,000 rpm"

MultipleChoiceQA

Question: *"What does photosynthesis produce that helps plants grow?"*

(A) water
(B) oxygen
(C) protein
(D) sugar

"sugar"

Beyond Format-Specialized Models

ExtractiveQA

Question: *"At what speed did the turbine operate?"*

(Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...

"16,000 rpm"

MultipleChoiceQA

Question: *"What does photosynthesis produce that helps plants grow?"*

(A) water
(B) oxygen
(C) protein
(D) sugar

"sugar"

Beyond Format-Specialized Models

ExtractiveQA

Question: *"At what speed did the turbine operate?"*

(Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...

MultipleChoiceQA

Question: *"What does photosynthesis produce that helps plants grow?"*

(A) water
(B) oxygen
(C) protein
(D) sugar

"16,000 rpm"

"sugar"

Beyond Format-Specialized Models

ExtractiveQA

Question: *"At what speed did the turbine operate?"*

(Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...

MultipleChoiceQA

Question: *"What does photosynthesis produce that helps plants grow?"*

(A) water
(B) oxygen
(C) protein
(D) sugar

"16,000 rpm"

"sugar"

Beyond Format-Specialized Models

ExtractiveQA

Question: *"At what speed did the turbine operate?"*

(Nikola_Tesla) On his 50th birthday in 1906, Tesla demonstrated his 200 horsepower (150 kilowatts) 16,000 rpm bladeless turbine. ...

"16,000 rpm"

MultipleChoiceQA

Question: *"What does photosynthesis produce that helps plants grow?"*

(A) water
(B) oxygen
(C) protein
(D) sugar

"sugar"

UnifiedQA: Definition

1. It's a single system that is supposed to work on a variety of **QA formats**.
2. The input should be *natural*.
   - Minimal-enough for a human solver to infer the format.

UnifiedQA: Definition

1. It’s a single system that is supposed to work on a variety of **QA formats**.
2. The input should be *natural*.
   - Minimal-enough for a human solver to infer the format.

UnifiedQA: Definition

1. It’s a single system that is supposed to work on a variety of **QA formats**.
2. The input should be *natural*.
   - Minimal-enough for a human solver to infer the format.

UnifiedQA: Definition

1. It's a single system that is supposed to work on a variety of **QA formats**.
2. The input should be *natural*.
   - Minimal-enough for a human solver to infer the format.

```
What causes sound?

(A) sunlight (B) vibrations (C) x-rays (D) pitch
```

"vibrations"

UnifiedQA: Definition

1. It's a single system that is supposed to work on a variety of **QA formats**.
2. The input should be *natural*.
   - Minimal-enough for a human solver to infer the format.

```
Is Jamaica considered part of the United States?

(Jamaica) Jamaica (/dʒəˈmeɪkə/ ( listen)) is an island
country situated in the Caribbean Sea...
```

"no"

UnifiedQA: Definition

```
What type of musical instruments did the Yuan bring to China?

(Yuan_dynasty) Western musical instruments were introduced to
enrich Chinese performing arts....
```

“Western musical instruments”

UnifiedQA: Definition

Our encoding:
- *Text-in, text-out*
- *The question always comes first.*
- *Additional info are appended with "\n".*

```
What type of musical instruments did the Yuan bring to China?

(Yuan_dynasty) Western musical instruments were introduced to
enrich Chinese performing arts....
```

“Western musical instruments”

UnifiedQA: Definition

1. It's a single system that is supposed to w
   formats.
2. The input should be *natural*.
   - Minimal-enough for a human solver to infer the for

Our encoding:
- *Text-in, text-out*
- *The question always comes first.*
- *Additional info are appended with "\n".*

```
What type of musical instruments did the Yuan bring to China?

(Yuan_dynasty) Western musical instruments were introduced to
enrich Chinese performing arts....
```

"Western musical instruments"

3. Use text-to-text architectures: T5 [Raffal et al. 2020], BART [Lewis et al. 2019], etc.

Experiment: Mixing Pairs of Formats

- Is there any value in out-of-format training?

Experiment: Mixing Pairs of Formats

- Is there any value in out-of-format training?

Mixing RACE (**Multiple-Choice**)

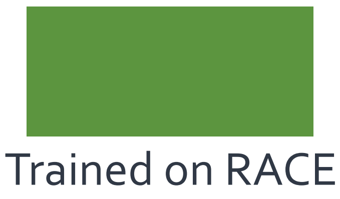
w/ datasets of different formats.

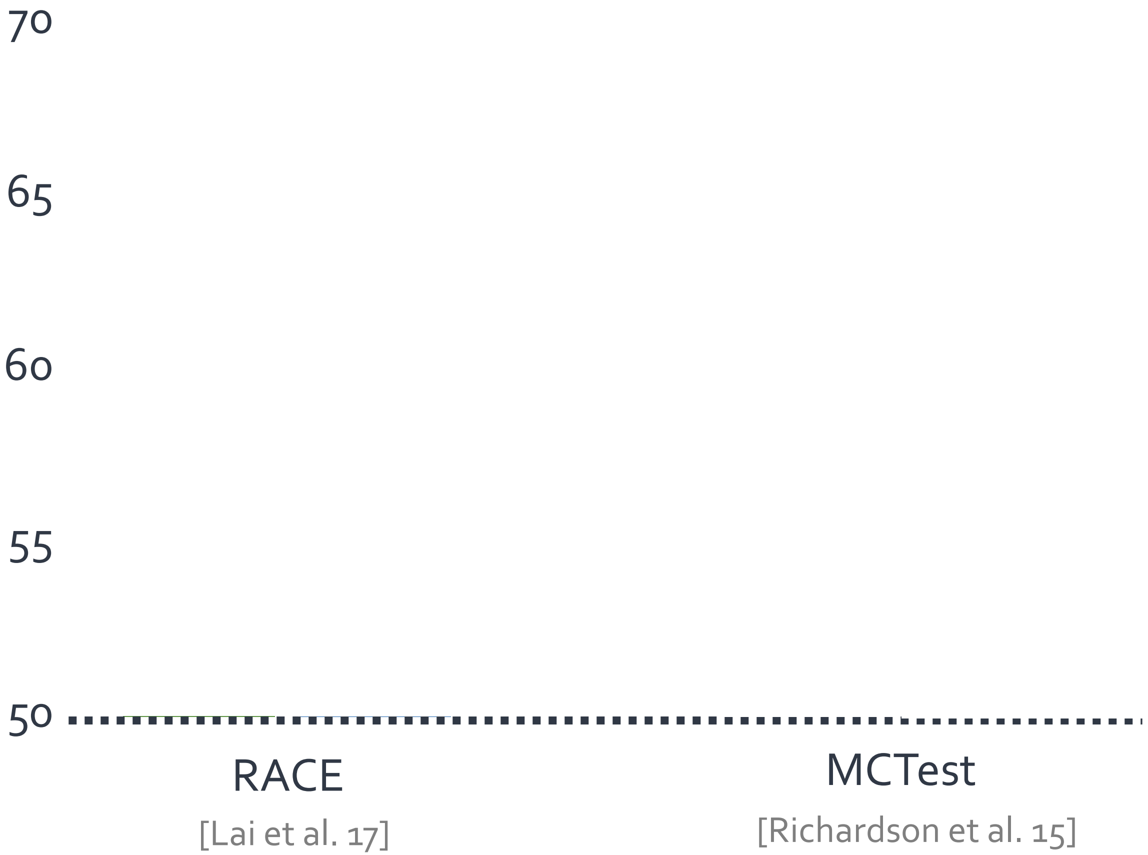
Experiment: Mixing Pairs of Formats

- Is there any value in out-of-format training?

Mixing RACE (**Multiple-Choice**)

w/ datasets of different formats.



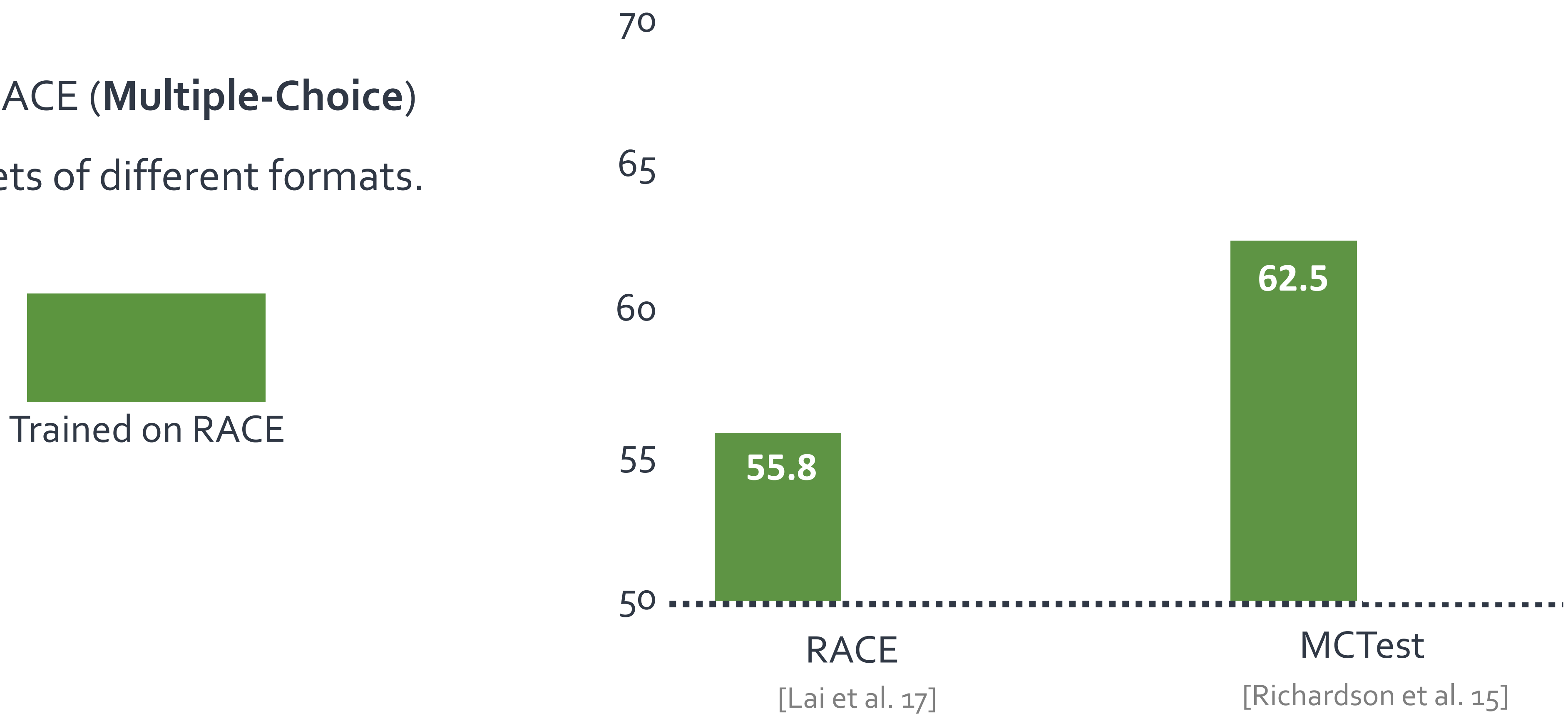


Experiment: Mixing Pairs of Formats

- Is there any value in out-of-format training?

Mixing RACE (**Multiple-Choice**)

w/ datasets of different formats.

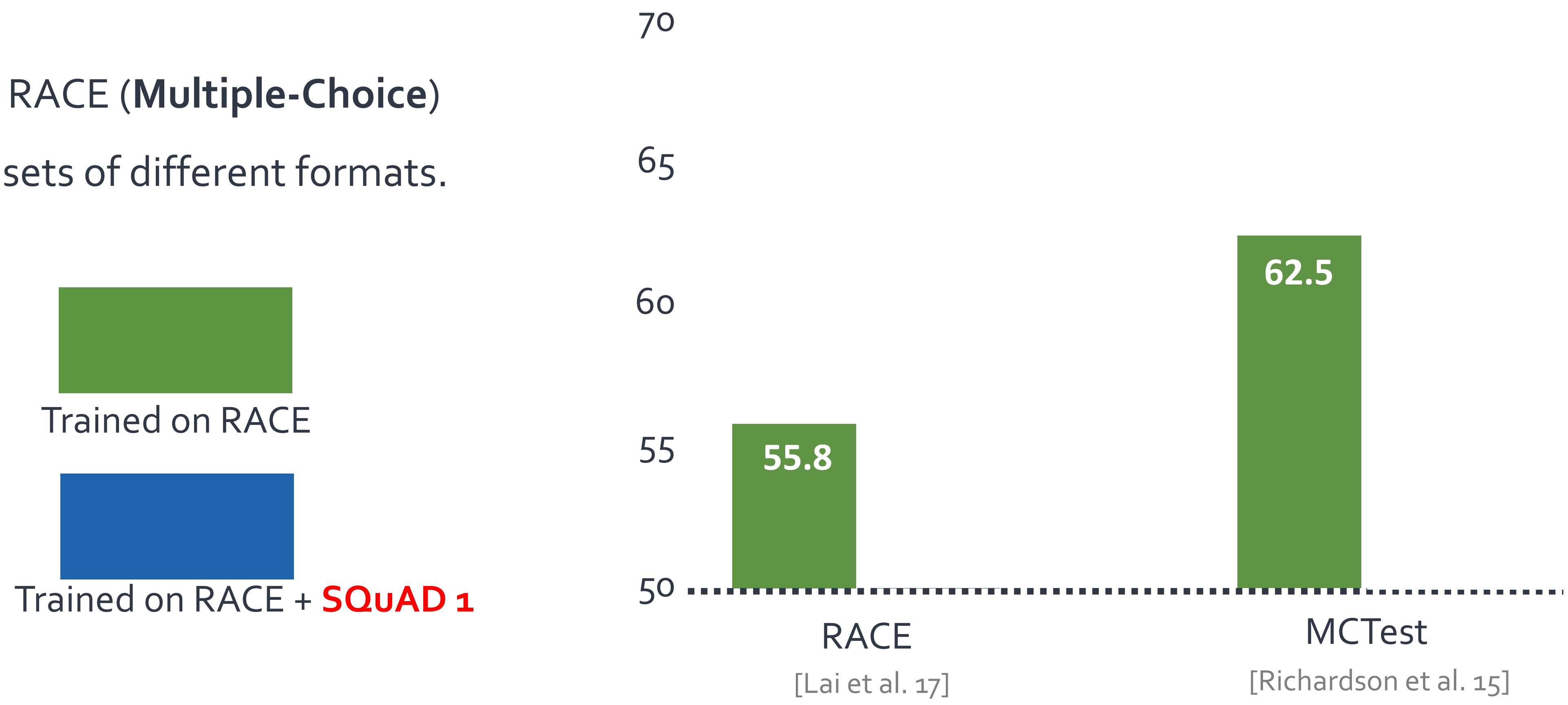


Experiment: Mixing Pairs of Formats

- Is there any value in out-of-format training?

Mixing RACE (**Multiple-Choice**)

w/ datasets of different formats.

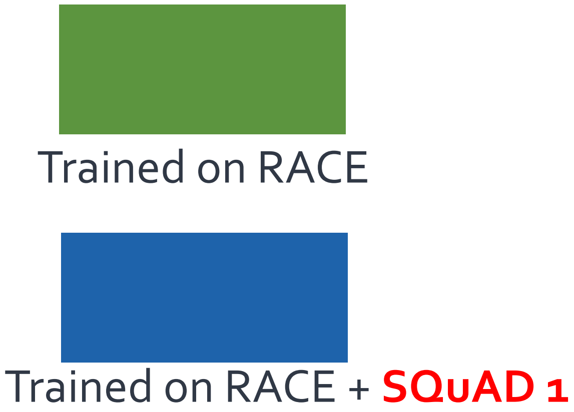


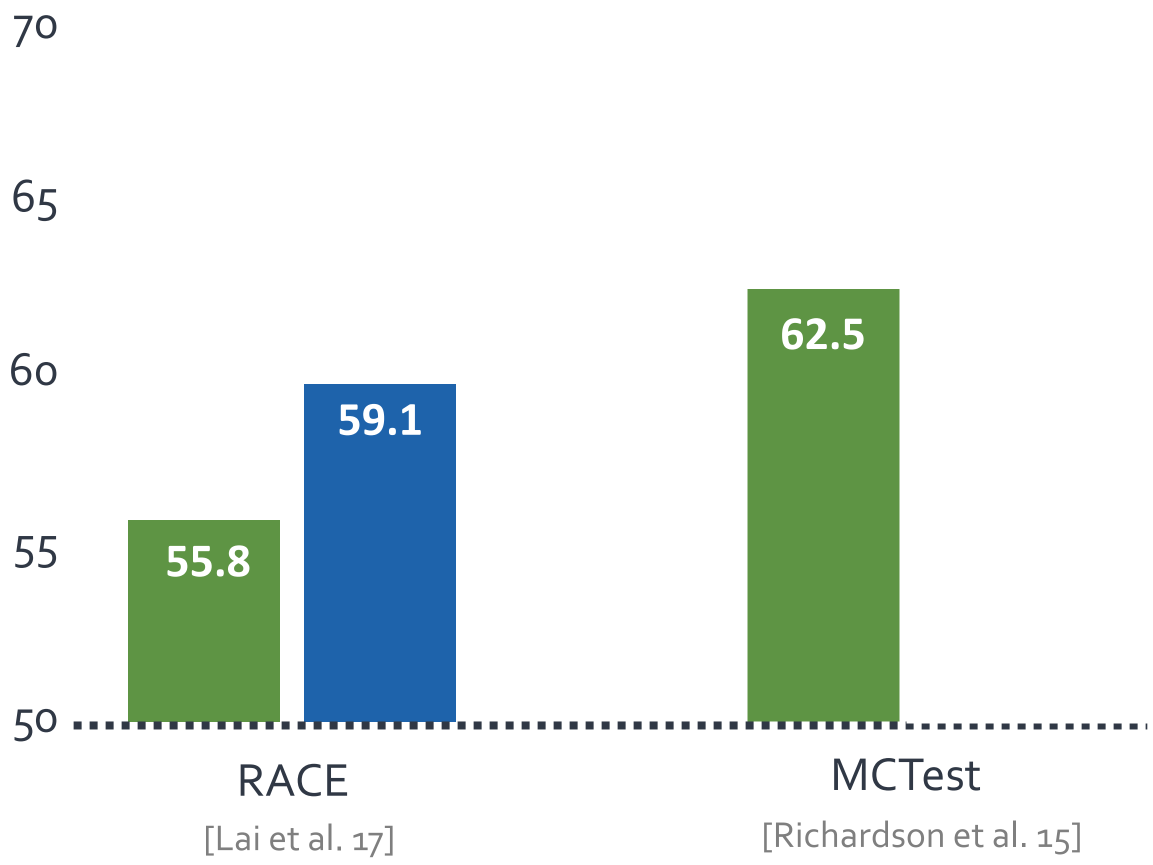
Experiment: Mixing Pairs of Formats

- Is there any value in out-of-format training?

Mixing RACE (**Multiple-Choice**)

w/ datasets of different formats.





Experiment: Mixing Pairs of Formats

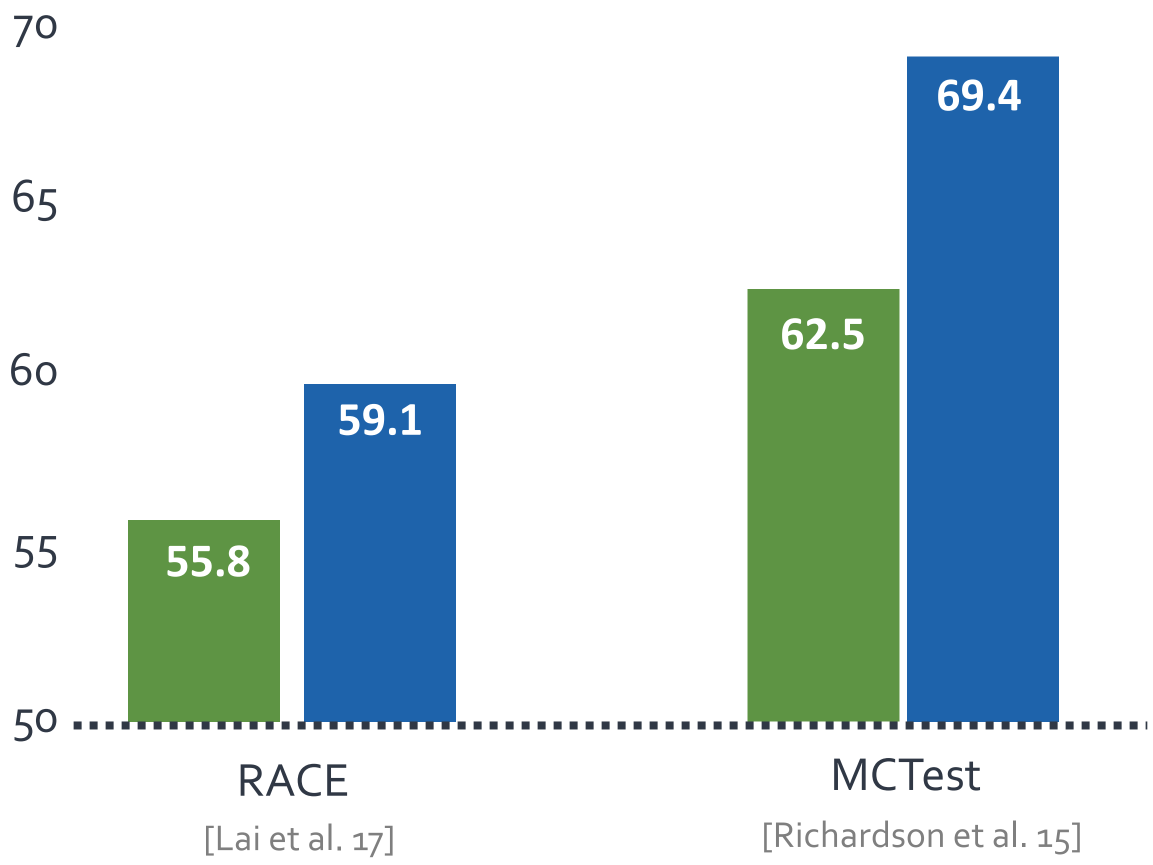
- Is there any value in out-of-format training?

Mixing RACE (**Multiple-Choice**)

w/ datasets of different formats.

Trained on RACE

Trained on RACE + **SQuAD 1**



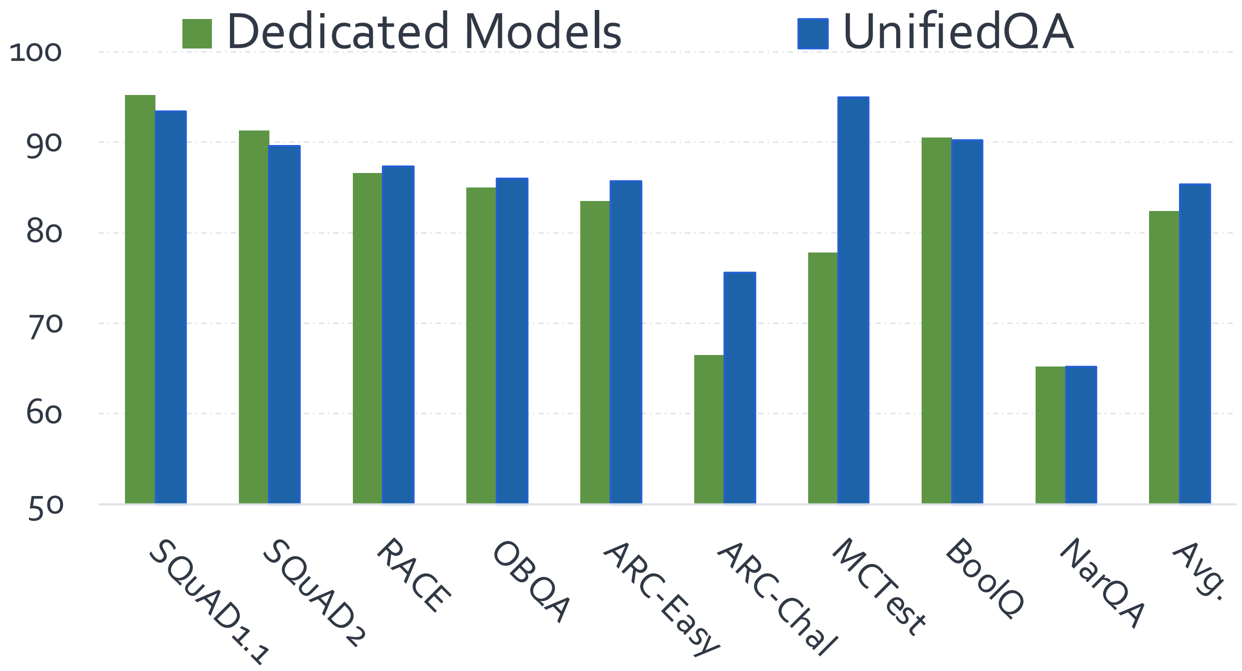
UnifiedQA-v1

- Trained on the union of different formats:
  - **Extractive:** SQuAD 1.1, SQuAD 2.0
  - **Abstractive:** NarrativeQA
  - **Multiple-choice:** RACE, ARC, OBQA, MCTest
  - **YesNo:** BoolQ

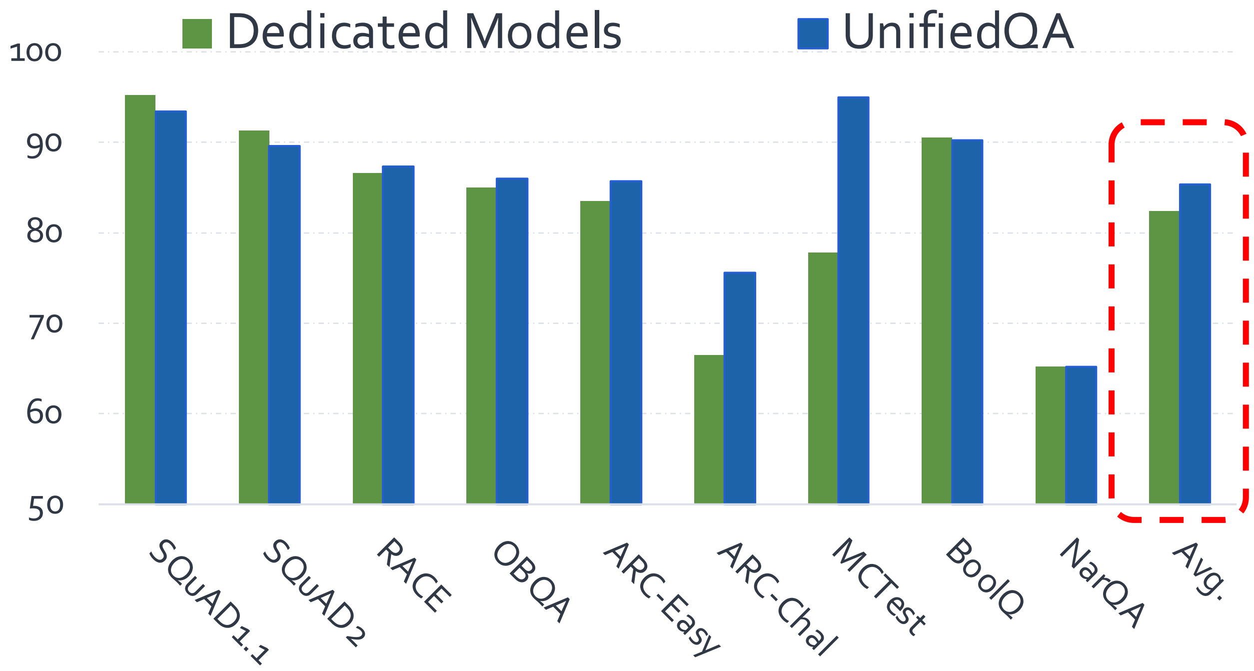
* Rajpurkar et al. '16 & '18; Kociský et al. '18; Lai et al. '17; Clark et al. '18; Mihaylov et al. '18; Richardson et al. '13; Clark et al. '19

Experiment: Comparison w/ Dedicated Models

Experiment: Comparison w/ Dedicated Models

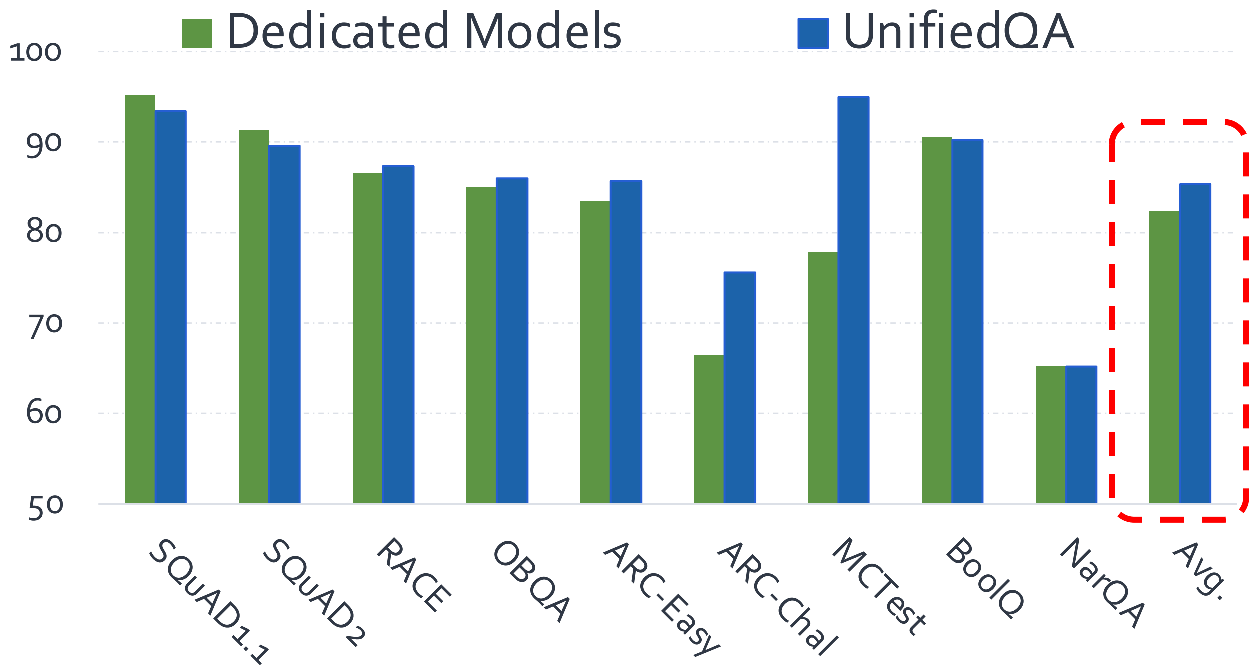


Experiment: Comparison w/ Dedicated Models



Experiment: Comparison w/ Dedicated Models

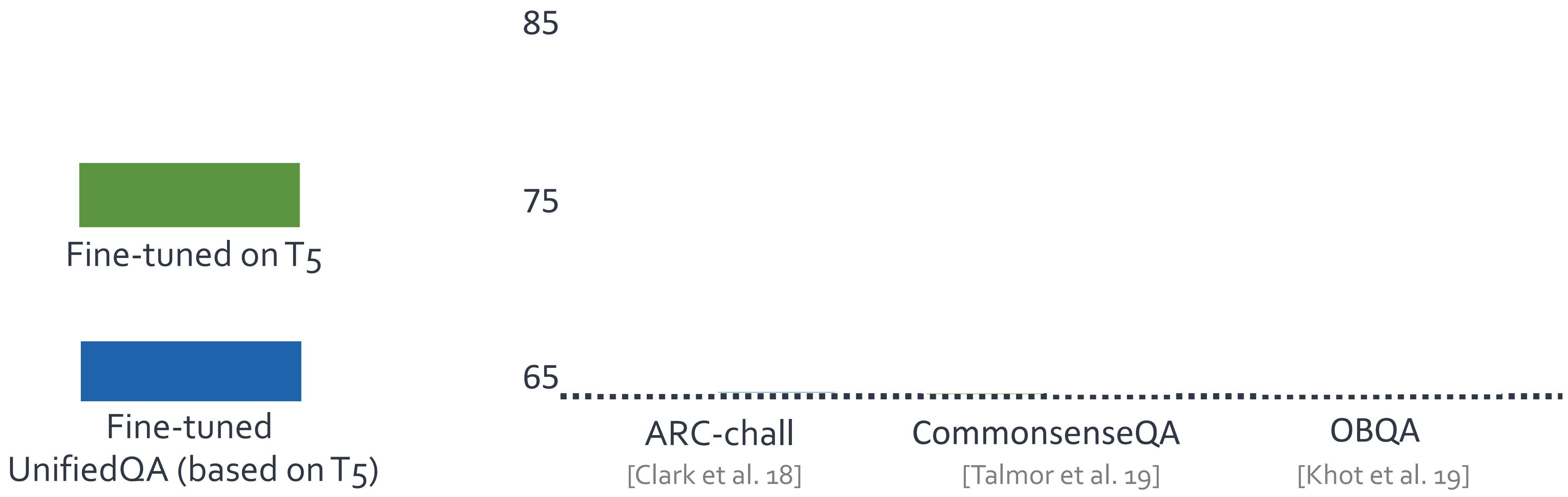
- Is UnifiedQA as good as systems dedicated to individual datasets?



- UnifiedQA performs almost as well as individual T5 models targeted to each dataset.

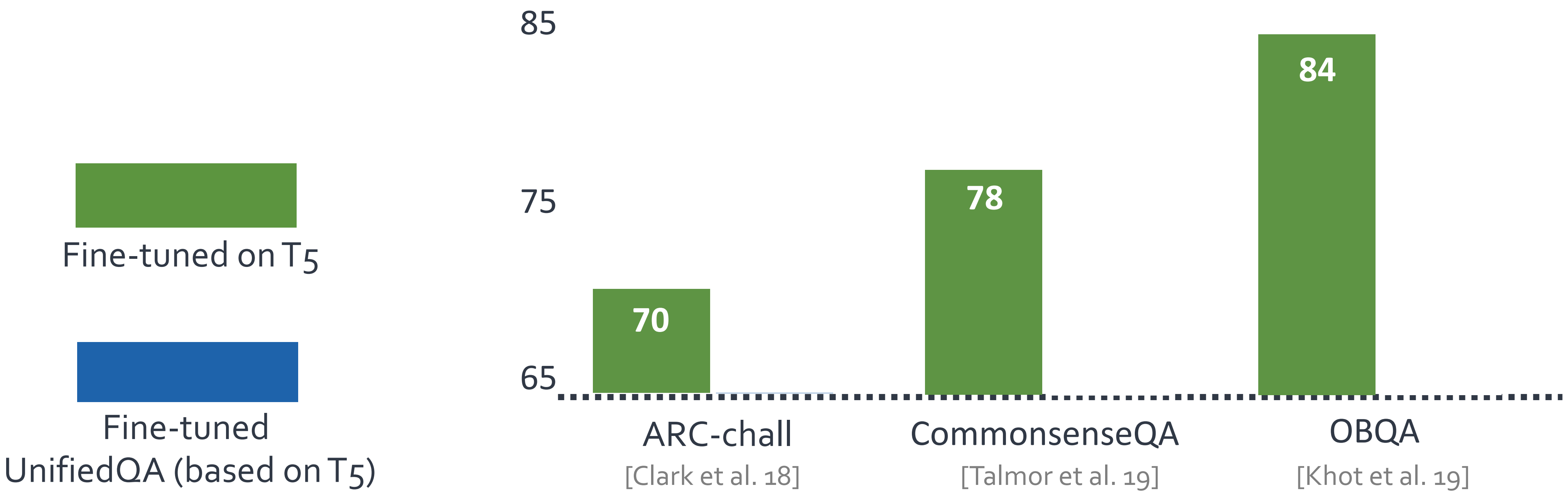
Experiment: Fine-tuning UnifiedQA

- Is there value in using UnifiedQA as a starting point for fine-tuning?
  - Show SOTA on 10 datasets (OBQA, QASC, RACE, WinoGrande, PIQA, SIQA, ROPES)
  - Similar trends for BART



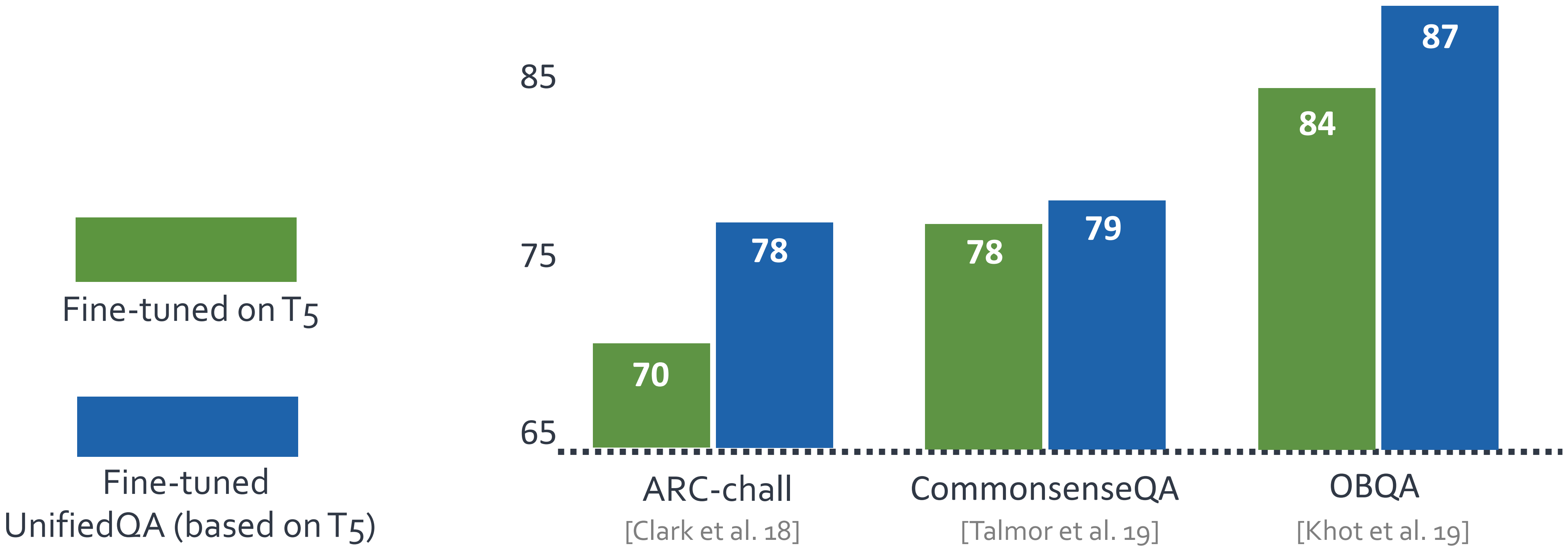
Experiment: Fine-tuning UnifiedQA

- Is there value in using UnifiedQA as a starting point for fine-tuning?
  - Show SOTA on 10 datasets (OBQA, QASC, RACE, WinoGrande, PIQA, SIQA, ROPES)
  - Similar trends for BART



Experiment: Fine-tuning UnifiedQA

- Is there value in using UnifiedQA as a starting point for fine-tuning?
  - Show SOTA on 10 datasets (OBQA, QASC, RACE, WinoGrande, PIQA, SIQA, ROPES)
  - Similar trends for BART



Earlier Work on Multi-task Learning

- In the same spirit as multi-task learning [Caruana '97; McCann et al. ’18]
  - They usually don’t work! 😭
- The choice of tasks is also important.
  - Earlier works select too broad of tasks.
    - E.g., [Raffel et al.’19]: diverse NLP tasks (machine translation, summarization, etc) and conclude that a single model for multiple NLP tasks underperform task-specific models.
- We choose to stay within the boundaries of QA.

Earlier Work on Multi-task Learning

- In the same spirit as multi-task learning [Caruana '97; McCann et al. '18]
  - They usually don't work! 😭
- The choice of tasks is also important.
  - Earlier works select too broad of tasks.
    - E.g., [Raffel et al.'19]: diverse NLP tasks (machine translation, summarization, etc) and conclude that a single model for multiple NLP tasks underperform task-specific models.
- We choose to stay within the boundaries of QA.

Earlier Work on Multi-task Learning

- In the same spirit as multi-task learning [Caruana '97; McCann et al. '18]
  - They usually don't work! 😭
- The choice of tasks is also important.
  - Earlier works select too broad of tasks.
    - E.g., [Raffel et al.'19]: diverse NLP tasks (machine translation, summarization, etc) and conclude that a single model for multiple NLP tasks underperform task-specific models.
- We choose to stay within the boundaries of QA.

Earlier Work on Multi-task Learning

- In the same spirit as multi-task learning [Caruana '97; McCann et al. '18]
  - They usually don't work! 😭

Didn't work before; why would it work now? 🤔

- The choice of tasks is also important.
  - Earlier works select too broad of tasks.
    - E.g., [Raffel et al.'19]: diverse NLP tasks (machine translation, summarization, etc) and conclude that a single model for multiple NLP tasks underperform task-specific models.

- We choose to stay within the boundaries of QA.

Earlier Work on Multi-task Learning

- In the same spirit as multi-task learning [Caruana '97; McCann et al. '18]
  - They usually don't work! 😭

Didn't work before; why would it work now? 🤔

- The choice of tasks is also important.
  - Earlier works select too broad of tasks.
    - E.g., [Raffel et al.'19]: diverse NLP tasks (machine translation, summarization, etc) and conclude that a single model for multiple NLP tasks underperform task-specific models.

- We choose to stay within the boundaries of QA.

Earlier Work on Multi-task Learning

- In the same spirit as multi-task learning [Caruana '97; McCann et al. '18]
  - They usually don't work! 😭

Didn't work before; why would it work now? 🤔

- The choice of tasks is also important.
  - Earlier works select too broad of tasks.
    - E.g., [Raffel et al.'19]: diverse NLP tasks (machine translation, summarization, etc) and conclude that a single model for multiple NLP tasks underperform task-specific models.

- We choose to stay within the boundaries of QA.

Lessons

- The field relies excessively on format-specific assumptions for system design.
  - Creating format-specific QA models distance us from broad QA.
- Instead, we should build more general QA architectures → more breadth!
- Incentive: there is value in mixing QA datasets of different formats.
- UnifiedQA: a single QA system working across four common QA formats
  - Fine-tuning models pre-trained on UnifiedQA yields SOTA results.

https://github.com/allenai/unifiedqa

Lessons

- The field relies excessively on format-specific assumptions for system design.
  - Creating format-specific QA models distance us from broad QA.
- Instead, we should build more general QA architectures → more breadth!
- Incentive: there is value in mixing QA datasets of different formats.
- UnifiedQA: a single QA system working across four common QA formats
  - Fine-tuning models pre-trained on UnifiedQA yields SOTA results.

https://github.com/allenai/unifiedqa

Lessons

- The field relies excessively on format-specific assumptions for system design.
  - Creating format-specific QA models distance us from broad QA.
- Instead, we should build more general QA architectures → more breadth!
- Incentive: there is value in mixing QA datasets of different formats.
- UnifiedQA: a single QA system working across four common QA formats
  - Fine-tuning models pre-trained on UnifiedQA yields SOTA results.

https://github.com/allenai/unifiedqa

Lessons

- The field relies excessively on format-specific assumptions for system design.
  - Creating format-specific QA models distance us from broad QA.
- Instead, we should build more general QA architectures → more breadth!
- Incentive: there is value in mixing QA datasets of different formats.
- UnifiedQA: a single QA system working across four common QA formats
  - Fine-tuning models pre-trained on UnifiedQA yields SOTA results.

https://github.com/allenai/unifiedqa

Lessons

- The field relies excessively on format-specific assumptions for system design.
  - Creating format-specific QA models distance us from broad QA.
- Instead, we should build more general QA architectures → more breadth!
- Incentive: there is value in mixing QA datasets of different formats.
- UnifiedQA: a single QA system working across four common QA formats
  - Fine-tuning models pre-trained on UnifiedQA yields SOTA results.

https://github.com/allenai/unifiedqa

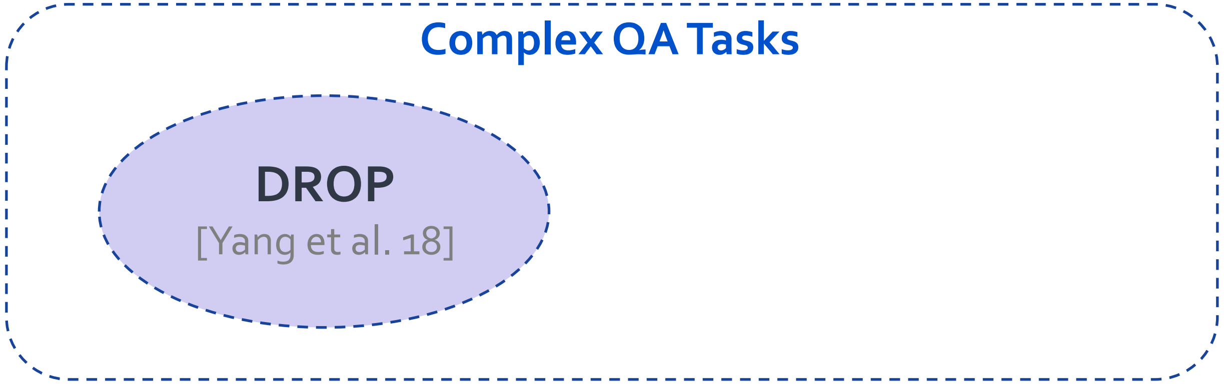
Decomposing Complex Questions in the Terms of Existing QA Models

KKRCS. Text Modular Networks: Learning to Decompose Tasks in the Language of Existing Models. arXiv preprint 20 (under review).

Generalization Across Multi-Hop Tasks

Complex QA Tasks

Generalization Across Multi-Hop Tasks



Generalization Across Multi-Hop Tasks

How many years did it take for the services sector to rebound?

Complex QA Tasks

DROP
[Yang et al. 18]

Generalization Across Multi-Hop Tasks

How many years did it take for the services sector to rebound?

Complex QA Tasks

DROP
[Yang et al. 18]

HotpotQA
[Dua et al. 19]

Generalization Across Multi-Hop Tasks

How many years did it take for the services sector to rebound?

Complex QA Tasks

DROP
[Yang et al. 18]

HotpotQA
[Dua et al. 19]

What was the nationality of the director of "The Simpsons"?

Generalization Across Multi-Hop Tasks

How many years did it take for the services sector to rebound?

Complex QA Tasks

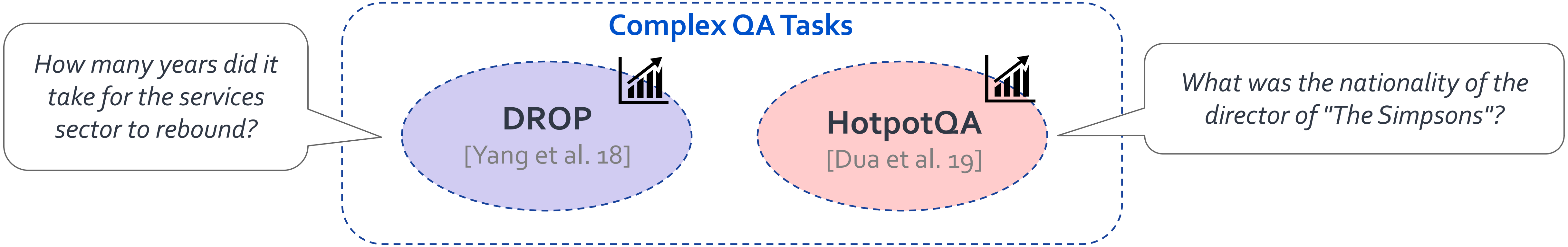
DROP
[Yang et al. 18]

HotpotQA
[Dua et al. 19]

What was the nationality of the director of "The Simpsons"?

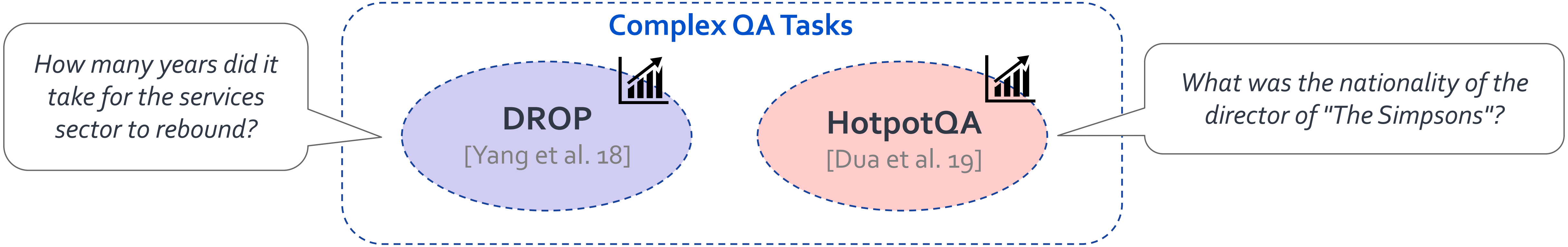
Generalization Across Multi-Hop Tasks

No system with a reasonable generalization across datasets! 😞



Generalization Across Multi-Hop Tasks

No system with a reasonable generalization across datasets! 😞



- **Challenge:** How do we build a system that generalize to **both** datasets? 🤔
- **Hypothesis:** despite having different distributions, their sub-problems are similar.
- **Idea:**
  - Build a framework to **decompose** complex questions into simpler ones.
  - Have a **shared** set of solvers for addressing the sub-questions.

Generalization Across Multi-Hop Tasks

No system with a reasonable generalization across datasets! 😞

How many years did it take for the services sector to rebound?

Complex QA Tasks

DROP
[Yang et al. 18]

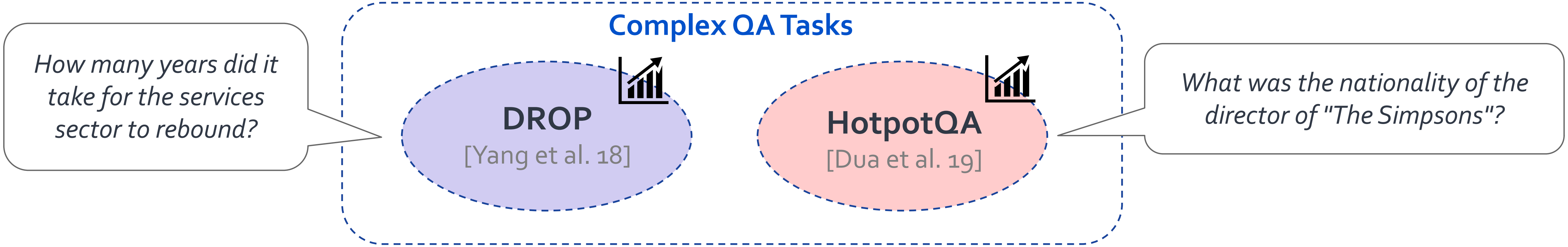
HotpotQA
[Dua et al. 19]

What was the nationality of the director of "The Simpsons"?

- **Challenge:** How do we build a system that generalize to **both** datasets? 🤔
- **Hypothesis:** despite having different distributions, their sub-problems are similar.
- **Idea:**
  - Build a framework to **decompose** complex questions into simpler ones.
  - Have a **shared** set of solvers for addressing the sub-questions.

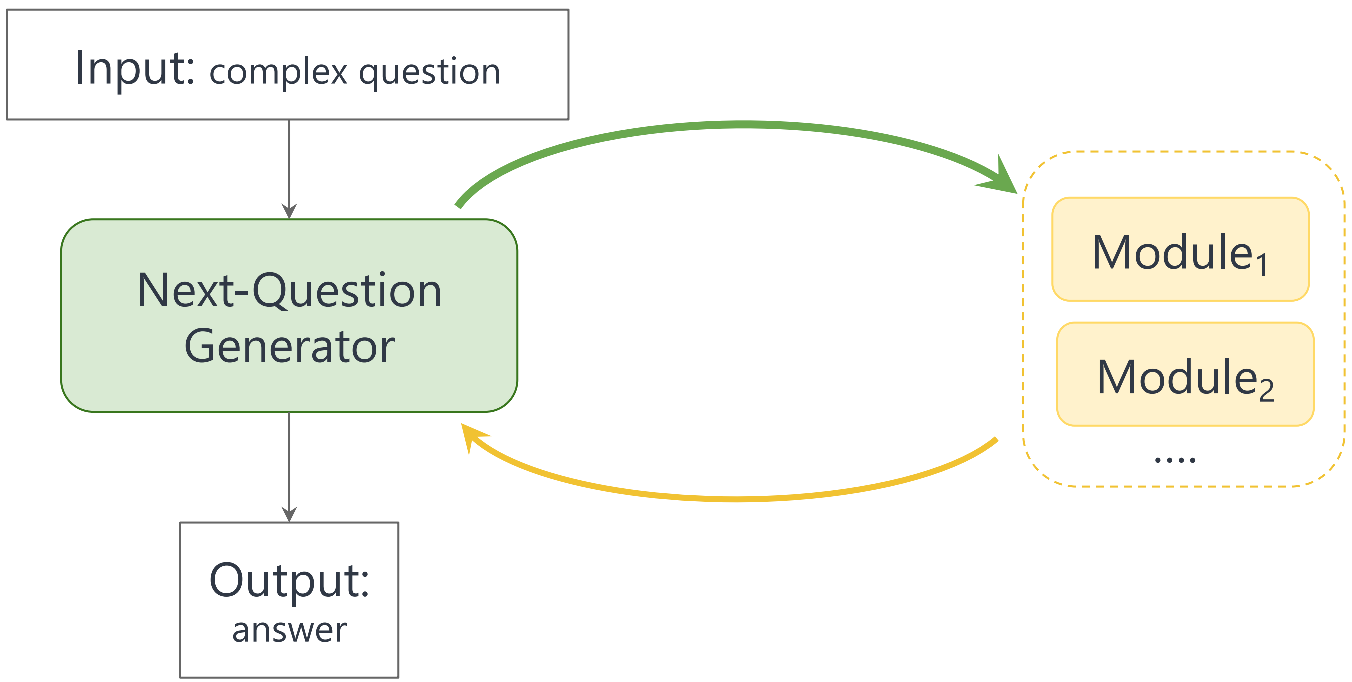
Generalization Across Multi-Hop Tasks

No system with a reasonable generalization across datasets! 😞



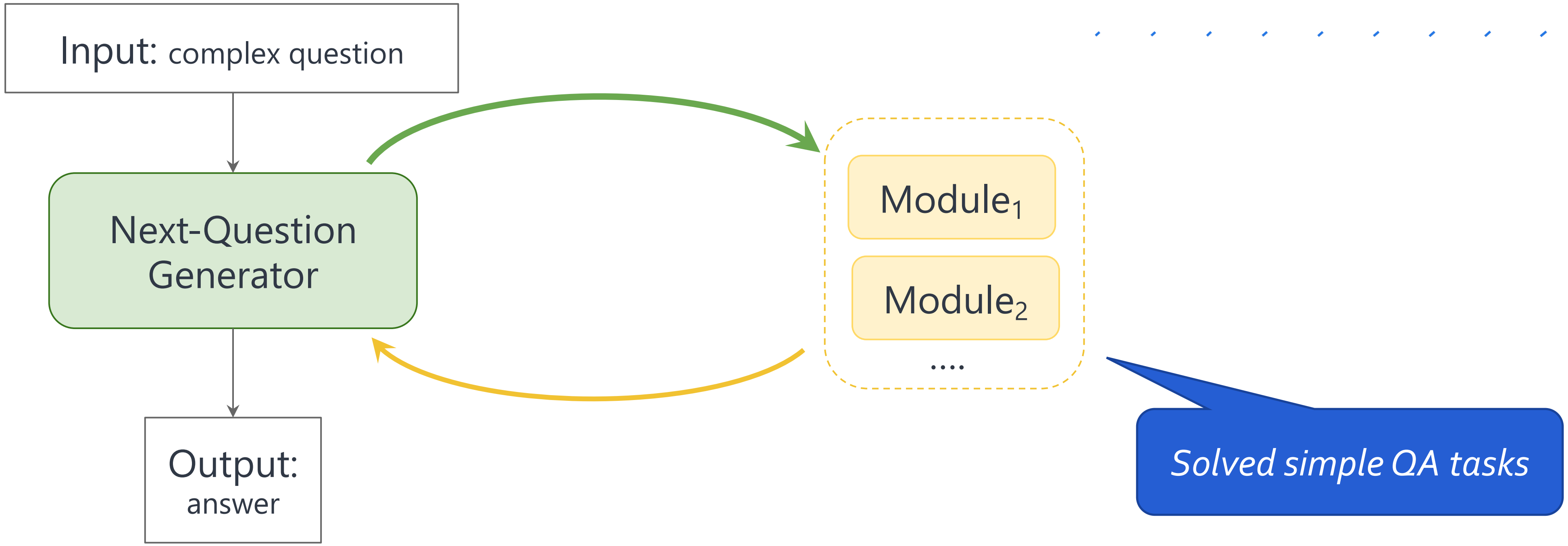
- **Challenge:** How do we build a system that generalize to **both** datasets? 🤔
- **Hypothesis:** despite having different distributions, their sub-problems are similar.
- **Idea:**
  - Build a framework to **decompose** complex questions into simpler ones.
  - Have a **shared** set of solvers for addressing the sub-questions.

Key Idea: Shoulders of Giants



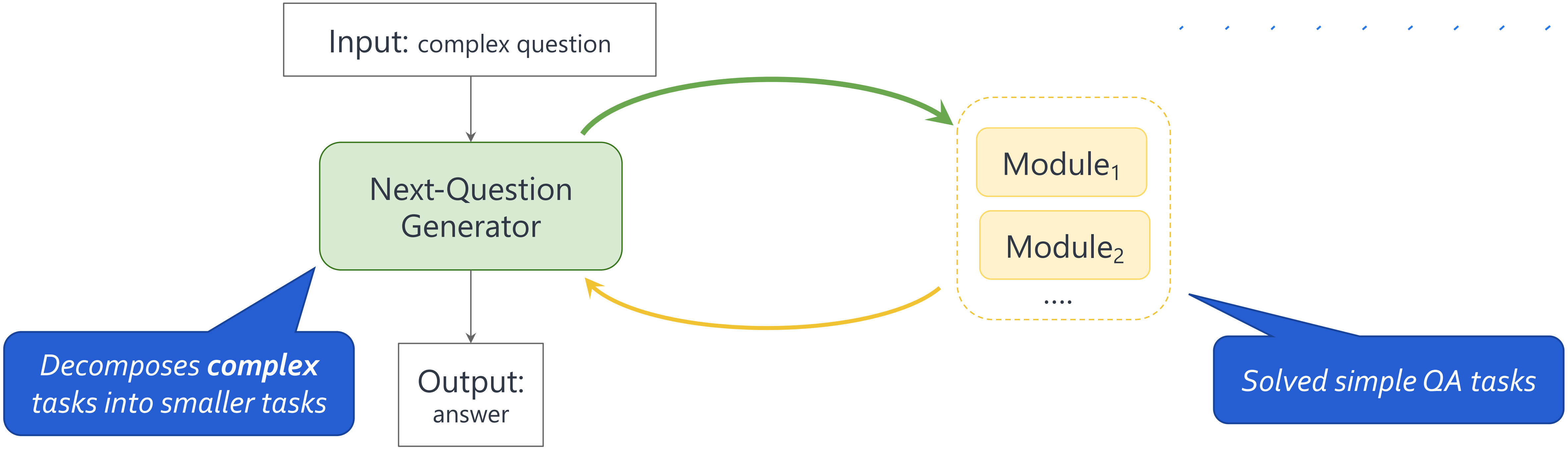
- Text Modular Networks:
  - Modules for different skills
  - Natural language for communication between modules
  - ModularQA: an implementation of this framework

Key Idea: Shoulders of Giants



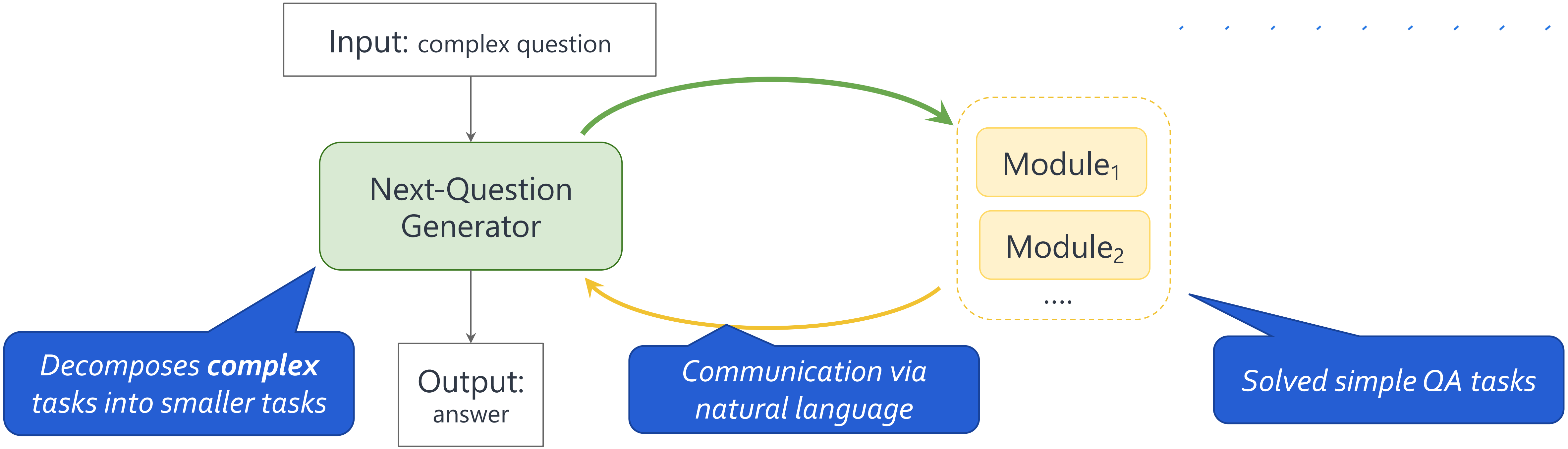
- Text Modular Networks:
  - Modules for different skills
  - Natural language for communication between modules
  - ModularQA: an implementation of this framework

Key Idea: Shoulders of Giants



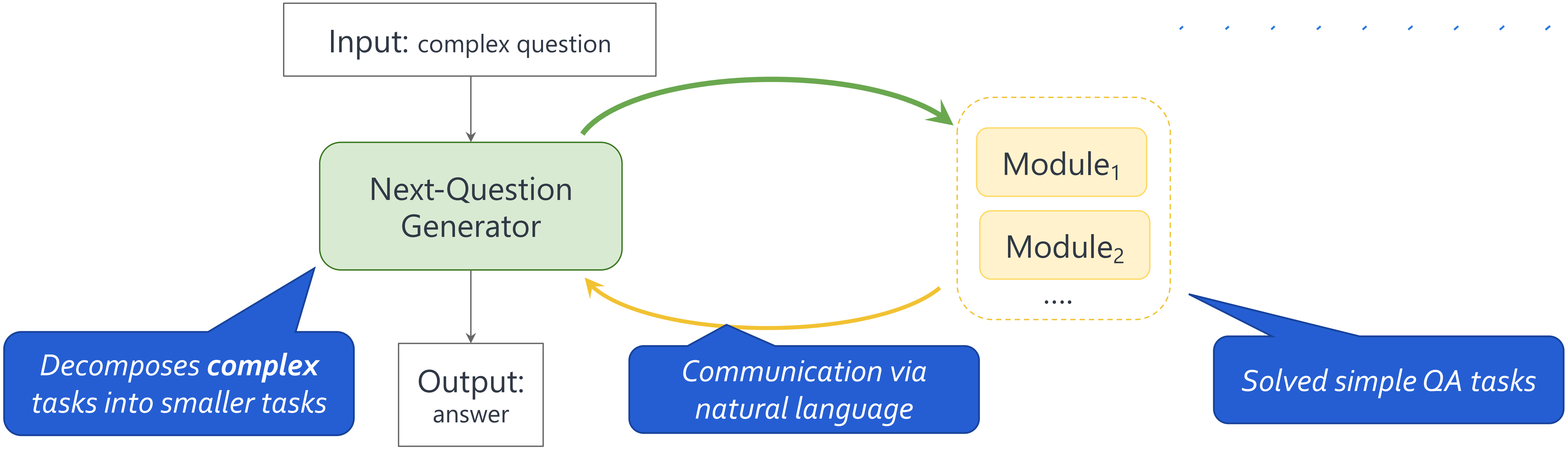
- Text Modular Networks:
  - Modules for different skills
  - Natural language for communication between modules
  - ModularQA: an implementation of this framework

Key Idea: Shoulders of Giants



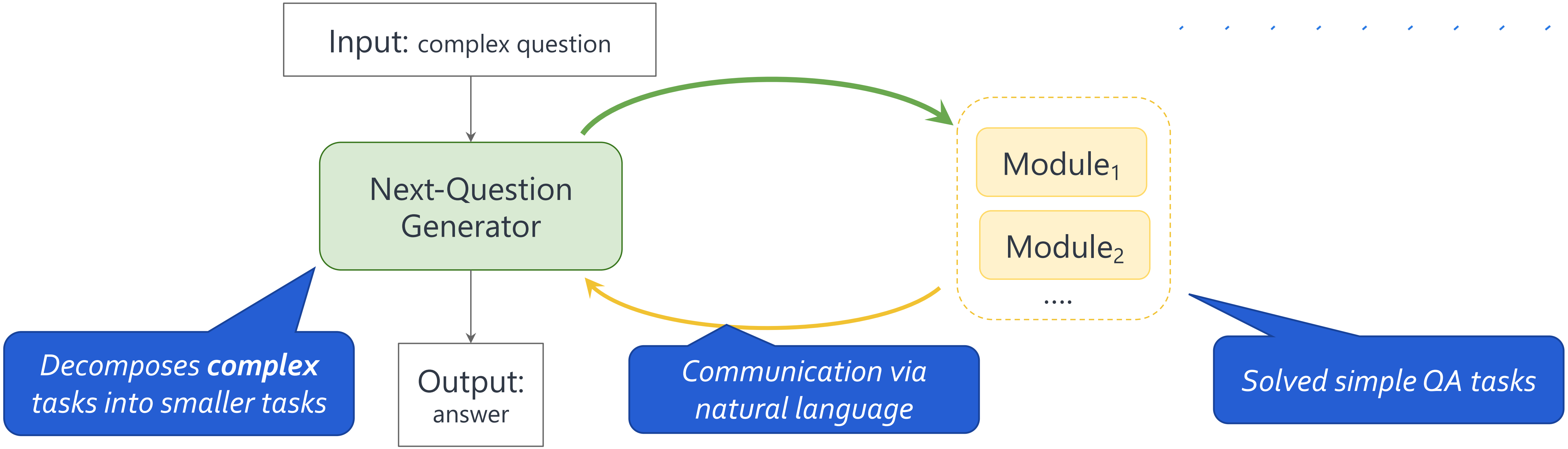
- Text Modular Networks:
  - Modules for different skills
  - Natural language for communication between modules
  - ModularQA: an implementation of this framework

Key Idea: Shoulders of Giants



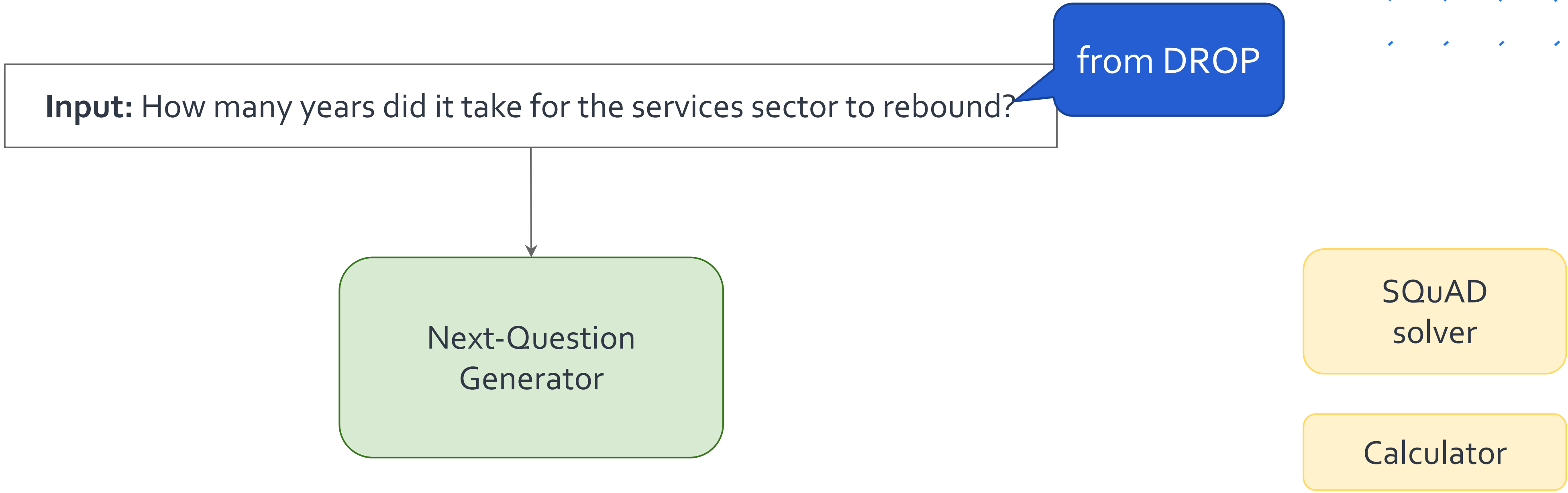
- Text Modular Networks:
  - Modules for different skills
  - Natural language for communication between modules
  - ModularQA: an implementation of this framework

Key Idea: Shoulders of Giants

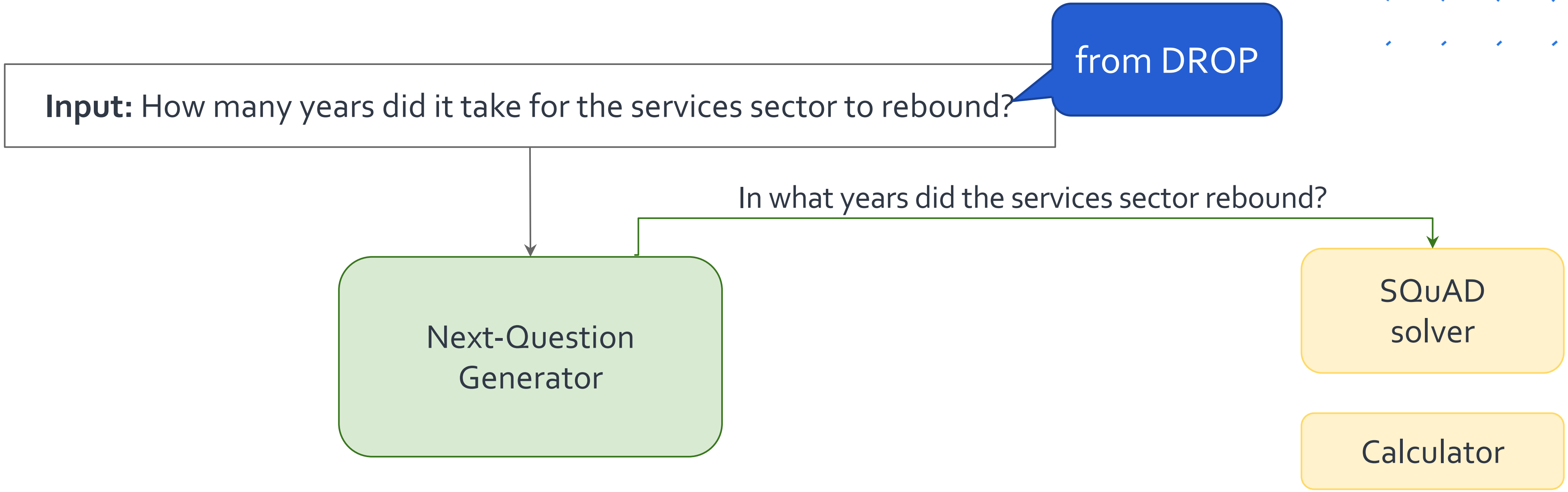


- Text Modular Networks:
  - Modules for different skills
  - Natural language for communication between modules
  - ModularQA: an implementation of this framework

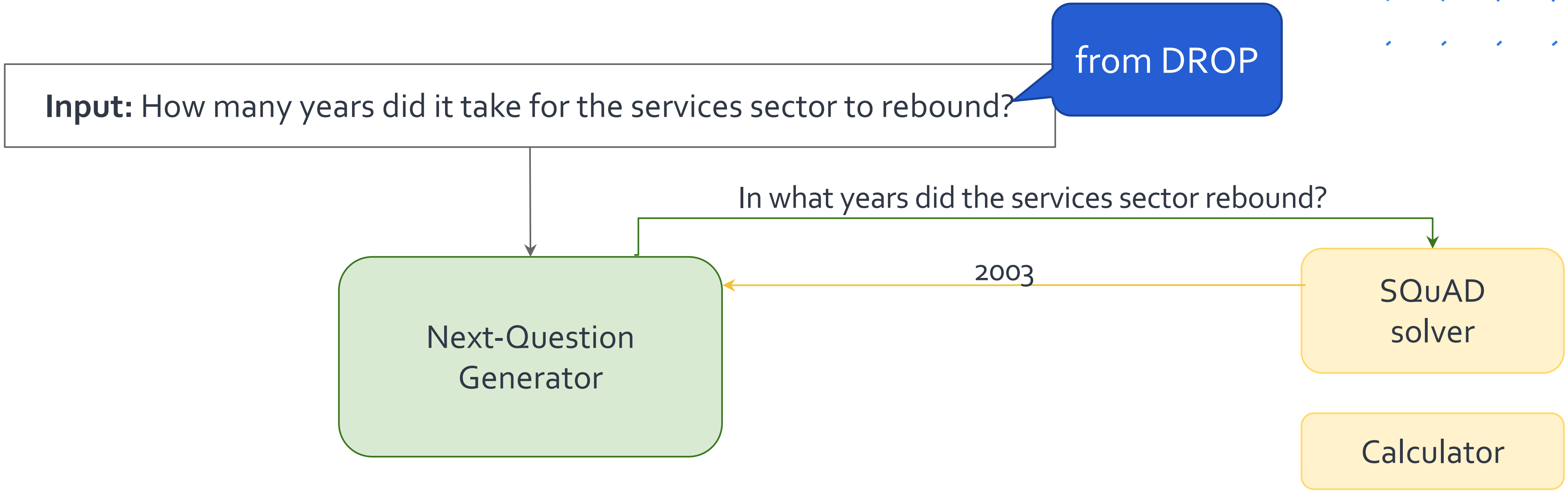
ModularQA: Example



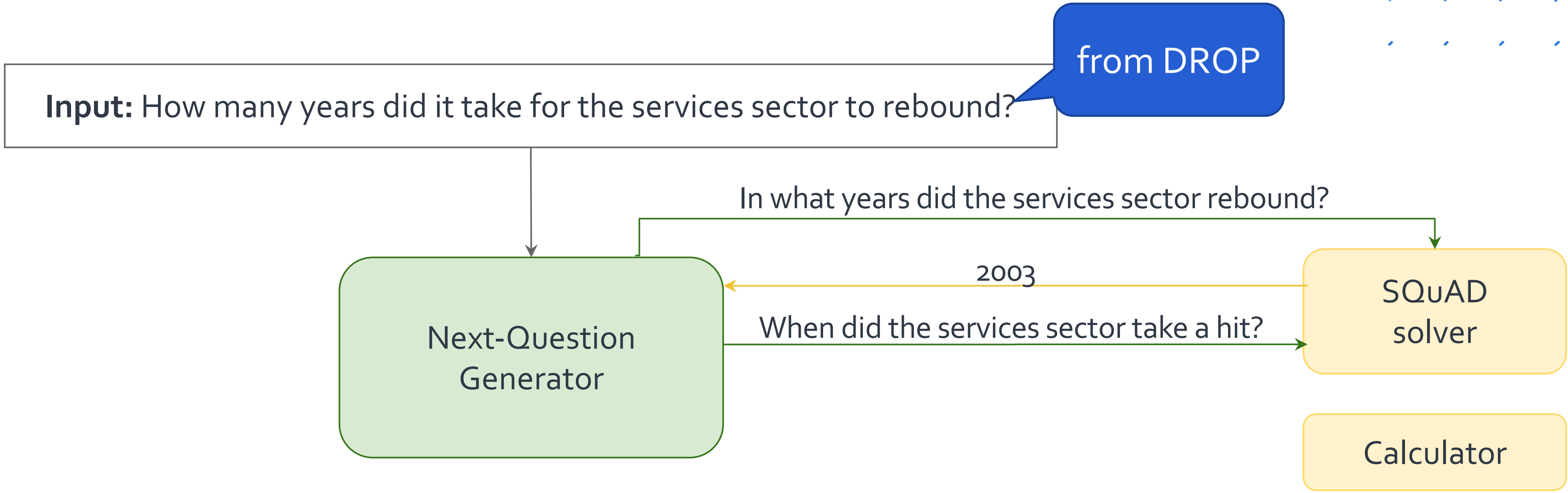
ModularQA: Example



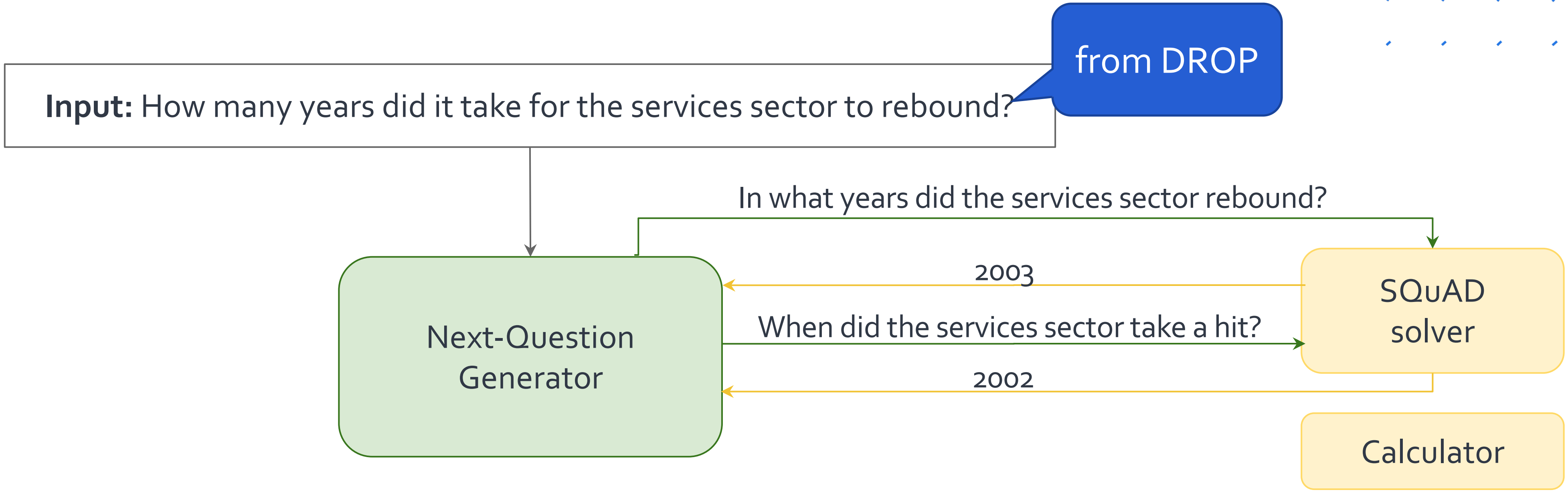
ModularQA: Example



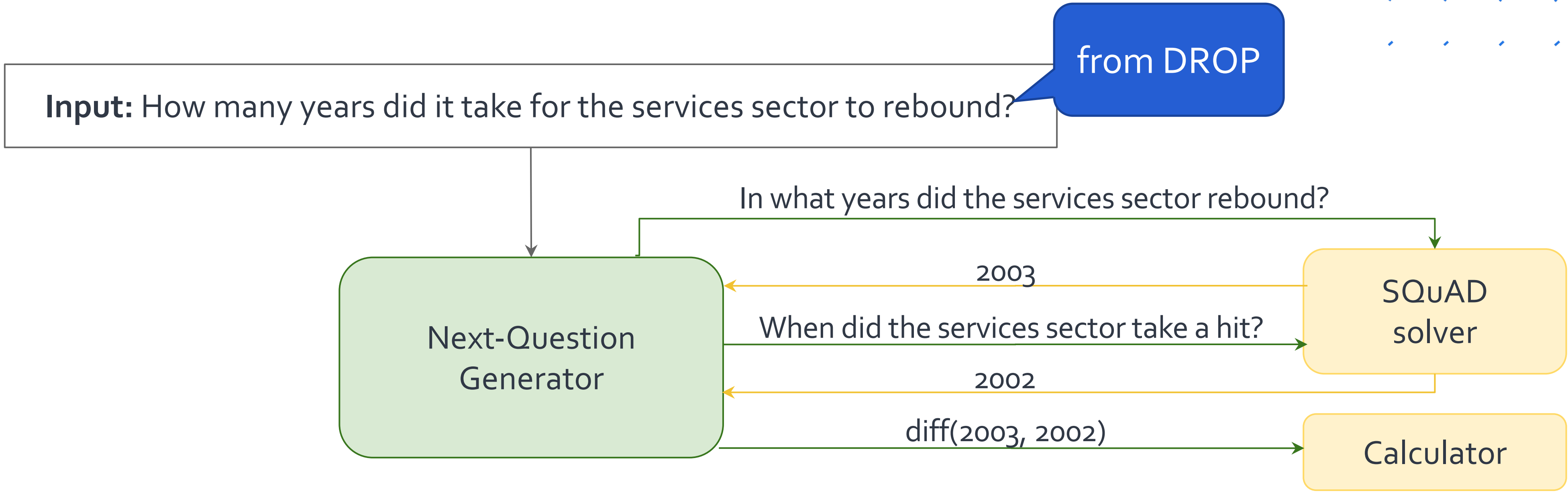
ModularQA: Example



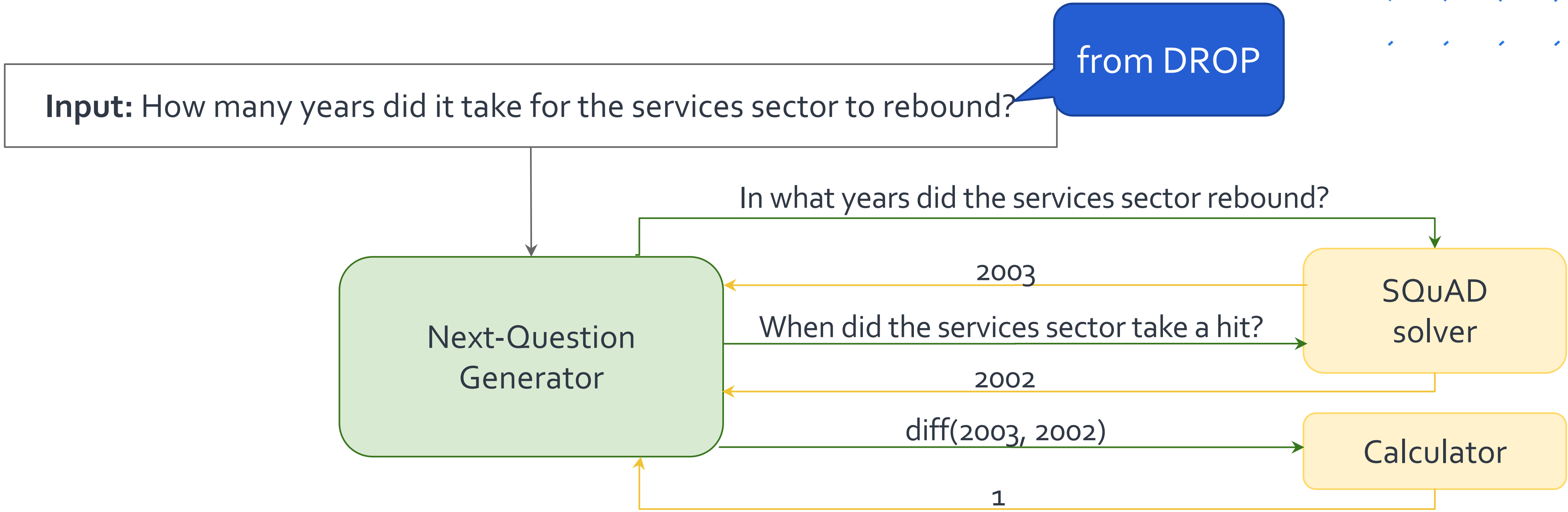
ModularQA: Example



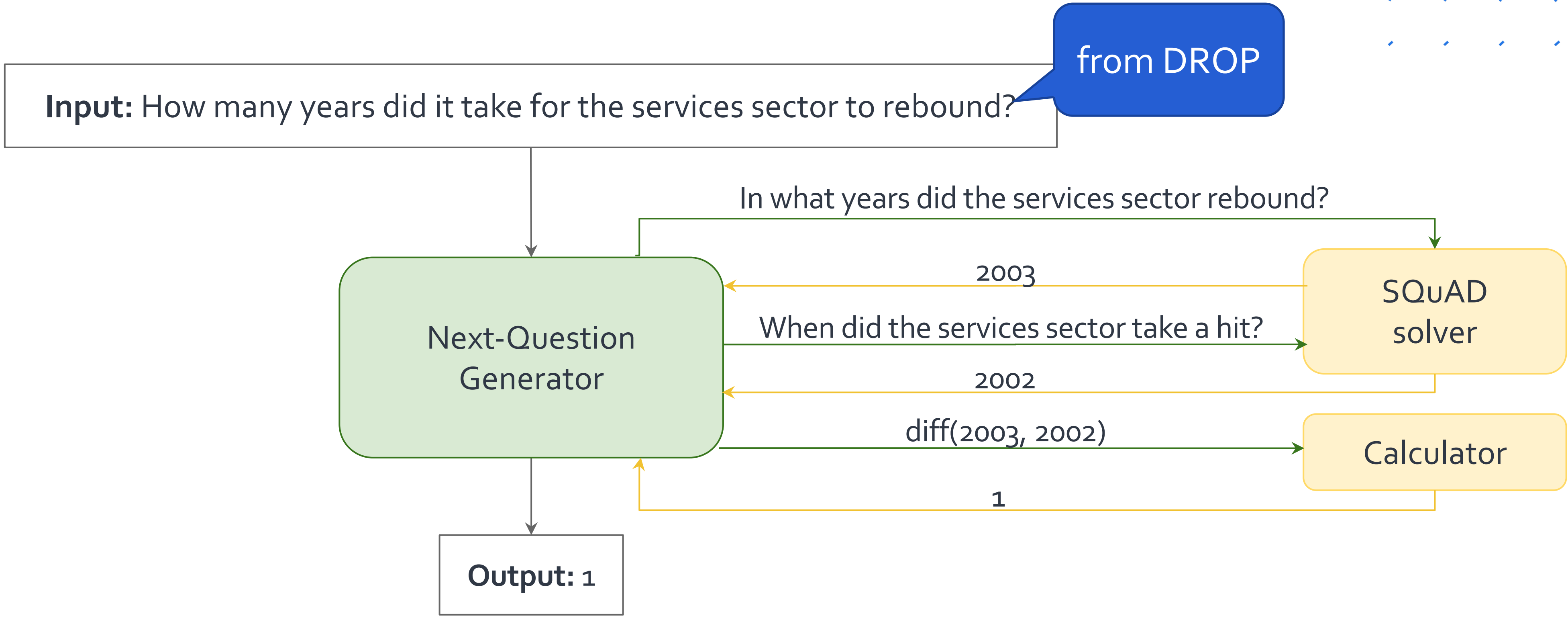
ModularQA: Example



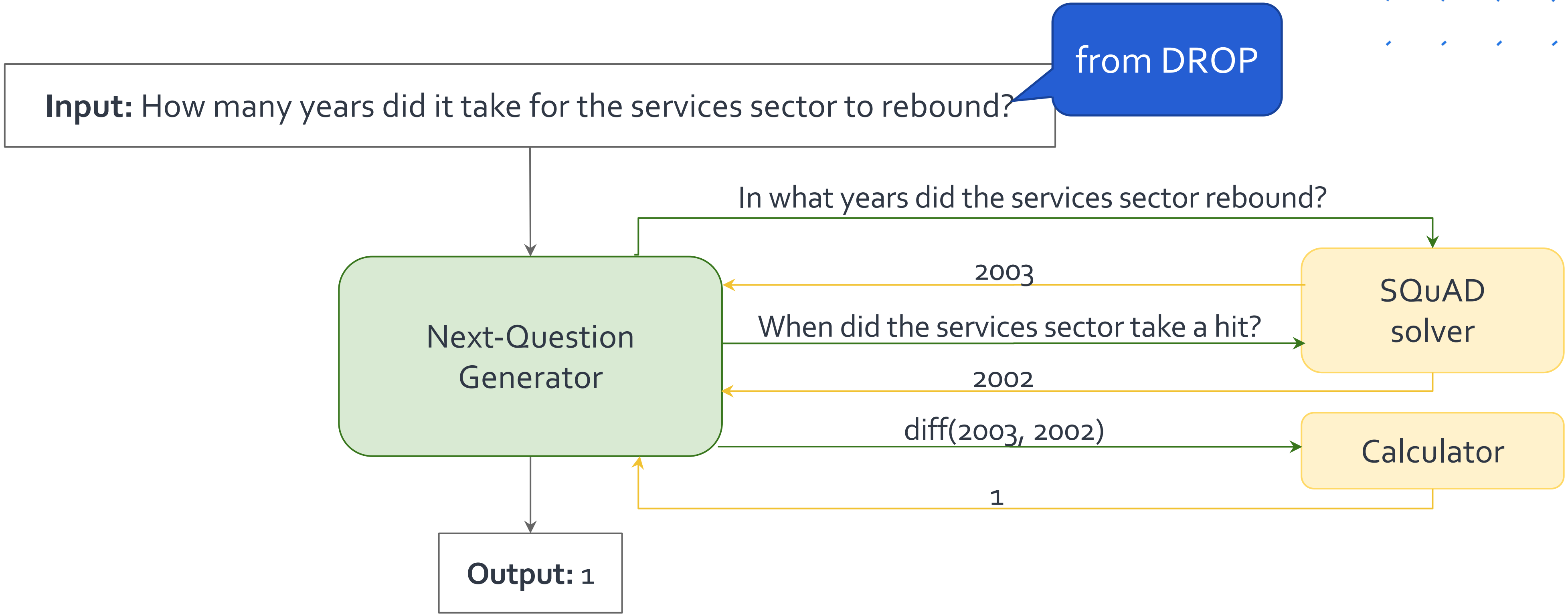
ModularQA: Example



ModularQA: Example

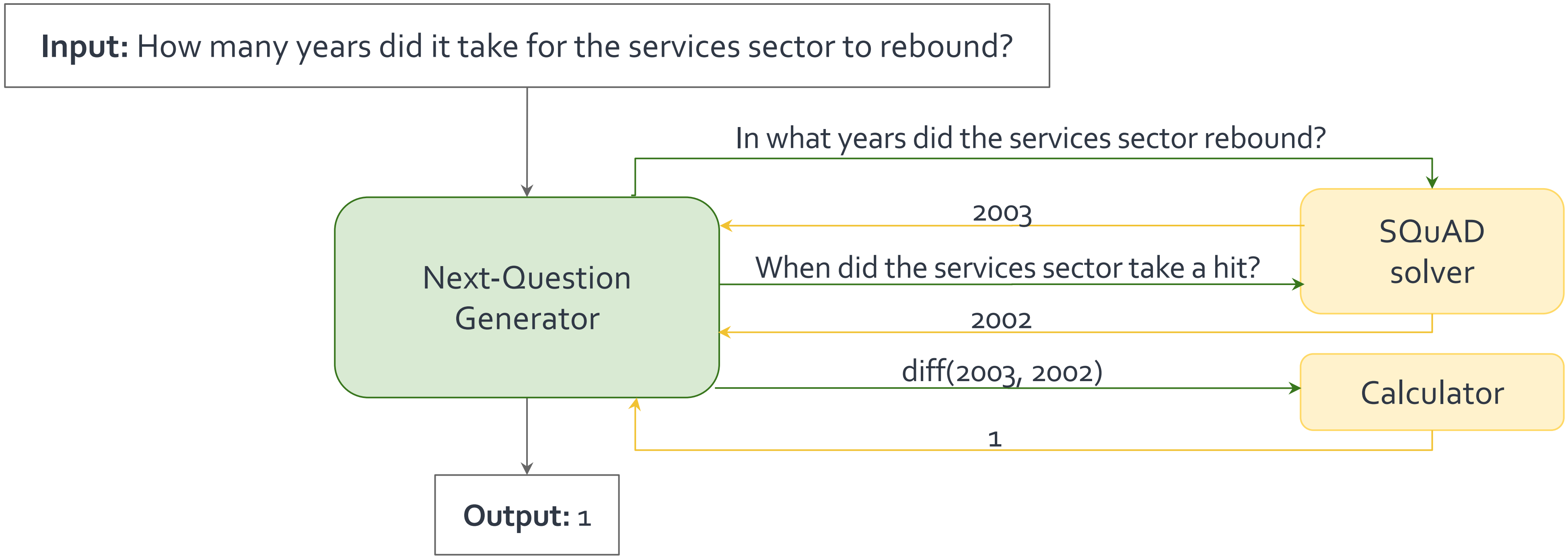


ModularQA: Example

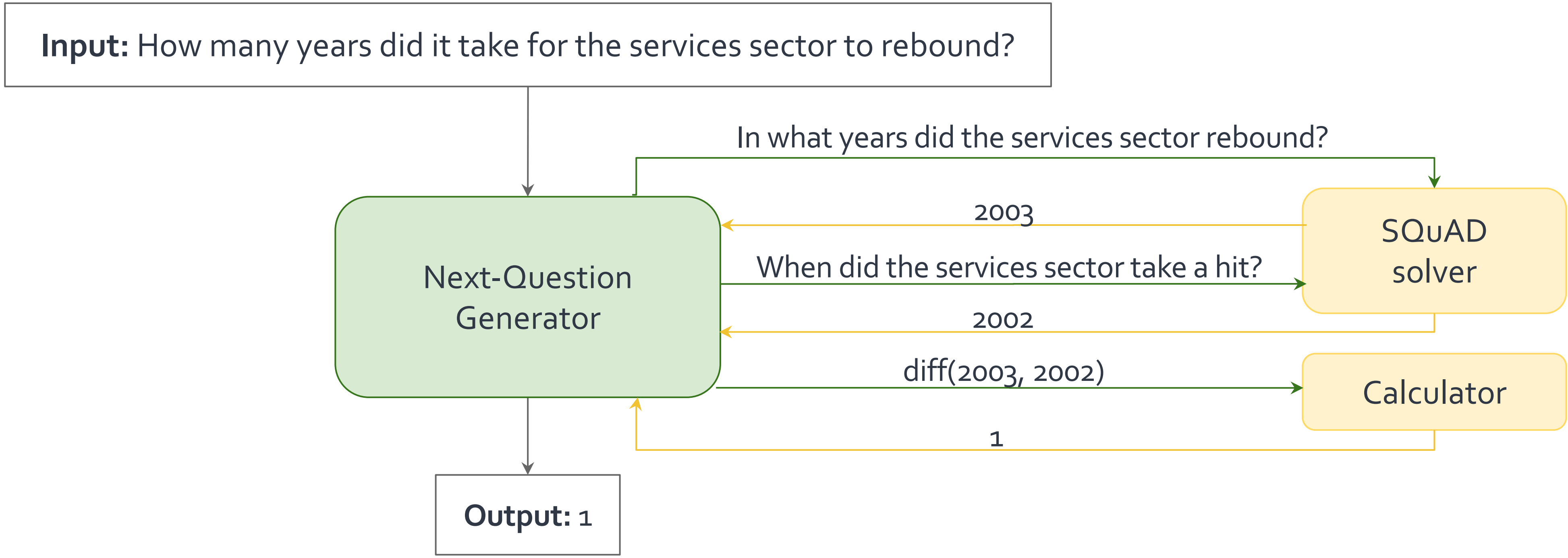


- **Immediate Benefit:**
  - Ease of interpretation

ModularQA: Example

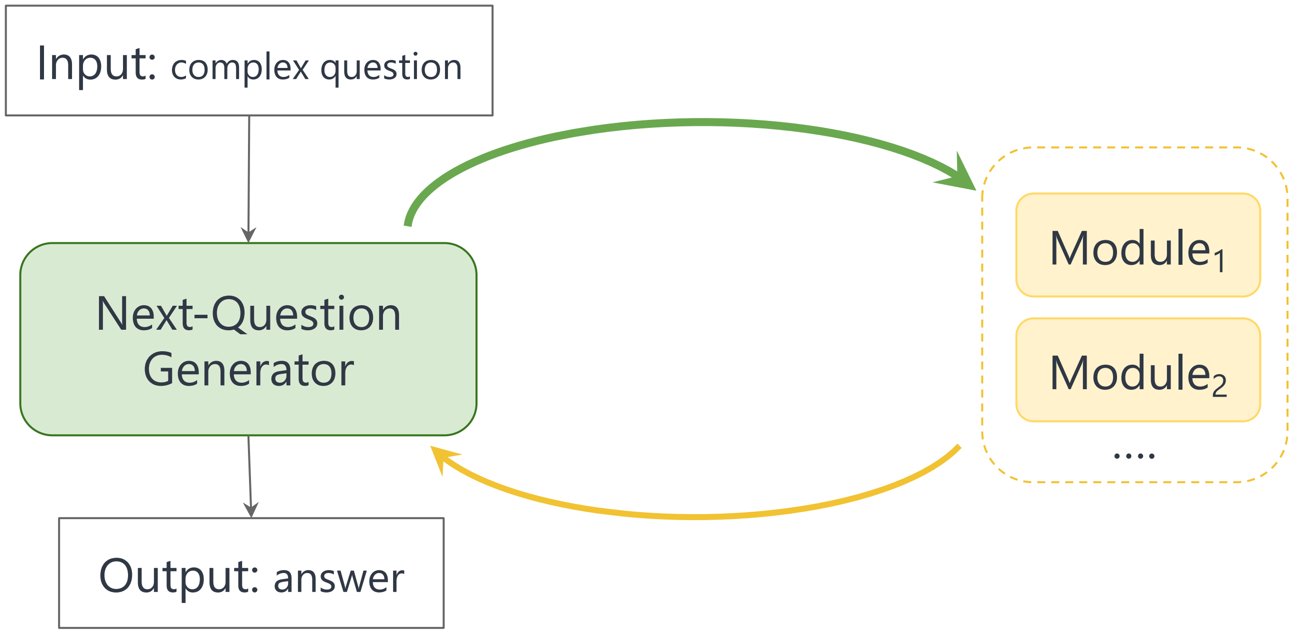


ModularQA: Example

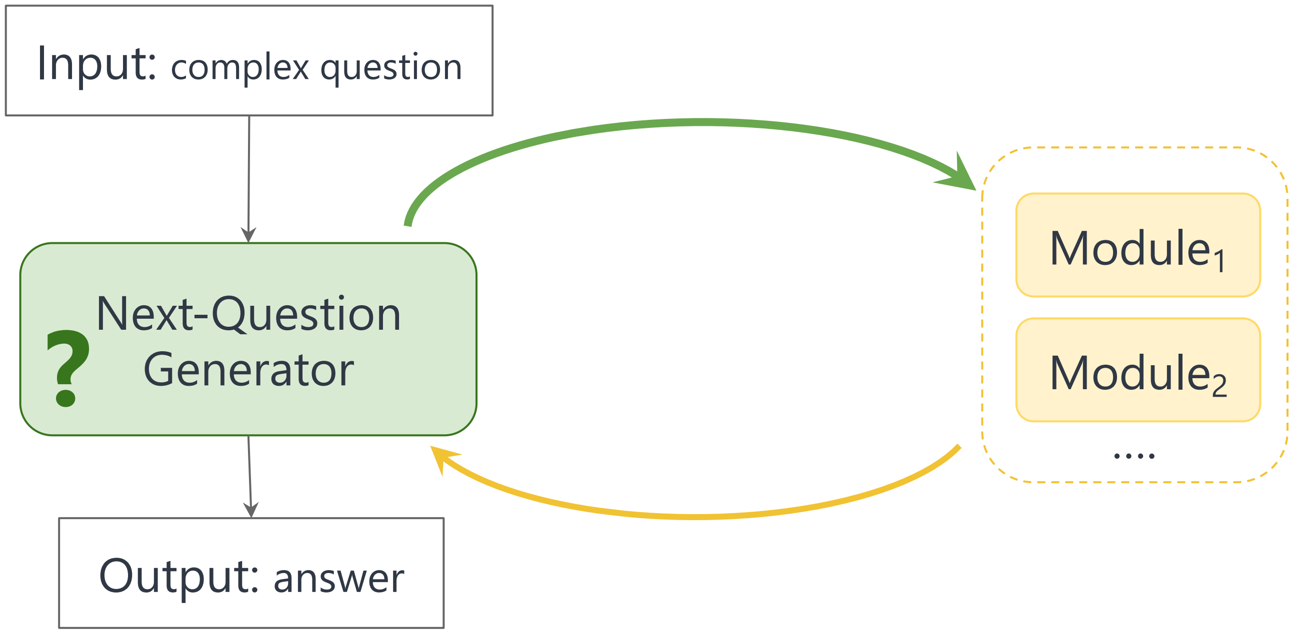


- **Challenge:**
  - How do we build this model (that decomposes the complex tasks into simpler sub-tasks)?

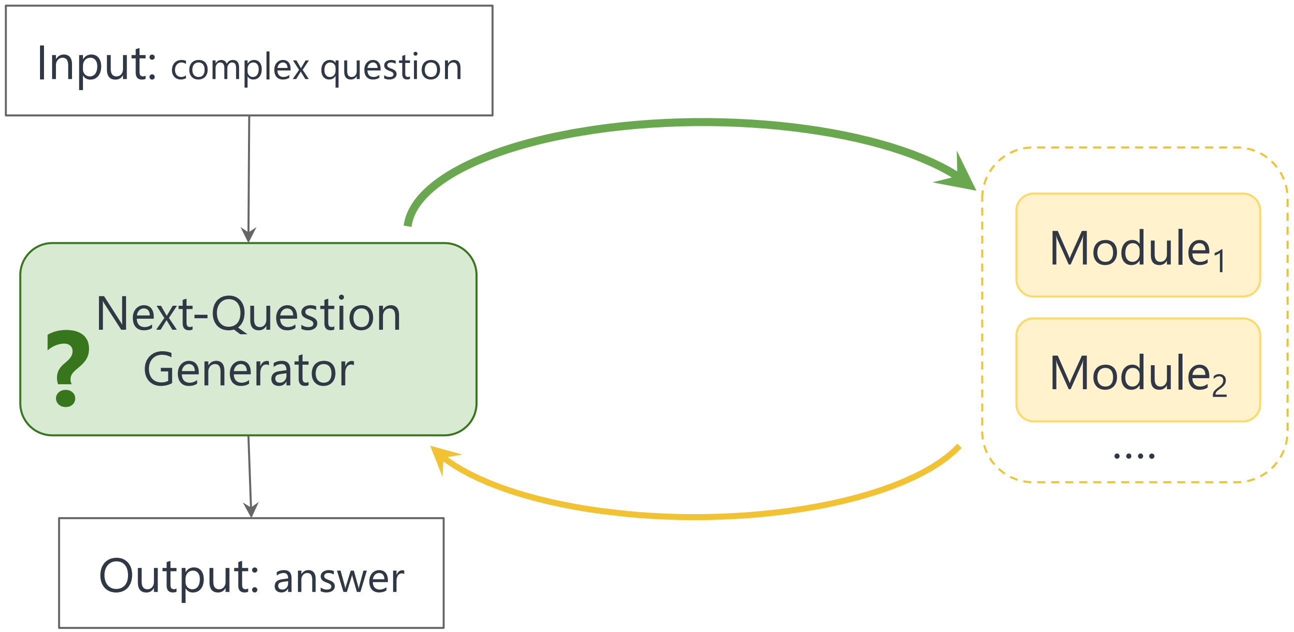
Key Pieces to be Solved



Key Pieces to be Solved



Key Pieces to be Solved



- **Design question:** how to build a "next question" box, s.t.:
    - The generated questions follow the "*language*" of existing QA sub-models (i.e., capabilities)

A Naïve (?) Approach

- Crowdsourcing approach for collecting question decomposition
  [Wolfson et al. 2020]

"From what yard line did Shayne Graham kick two field goals?"

1. Shayne Graham
2. field goals of #1
3. yards of #2
4. number of #2 for each #3
5. #3 where #4 is two

- Costly human annotations
- Not necessarily comprehensible to existing models

A Naïve (?) Approach

- Crowdsourcing approach for collecting question decomposition
  [Wolfson et al. 2020]

"From what yard line did Shayne Graham kick two field goals?"

1. Shayne Graham
2. field goals of #1
3. yards of #2
4. number of #2 for each #3
5. #3 where #4 is two

- Costly human annotations
- Not necessarily comprehensible to existing models

A Naïve (?) Approach

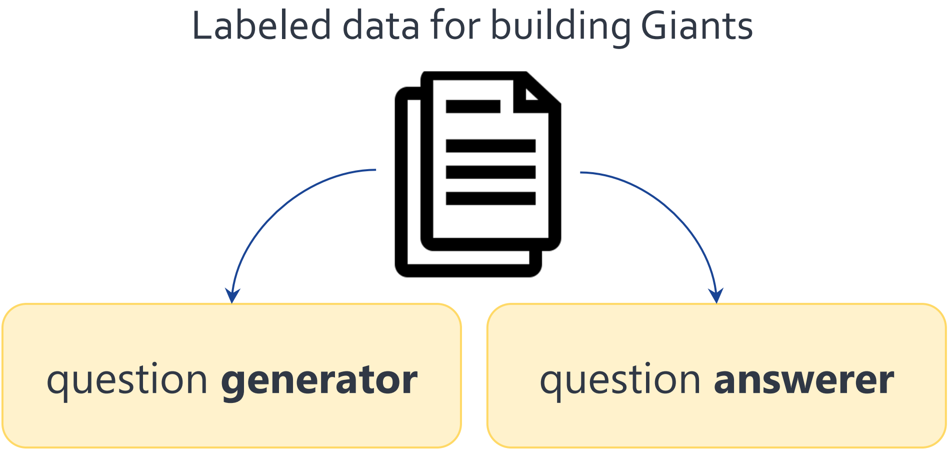
- Crowdsourcing approach for collecting question decomposition
  [Wolfson et al. 2020]

"From what yard line did Shayne Graham kick two field goals?"

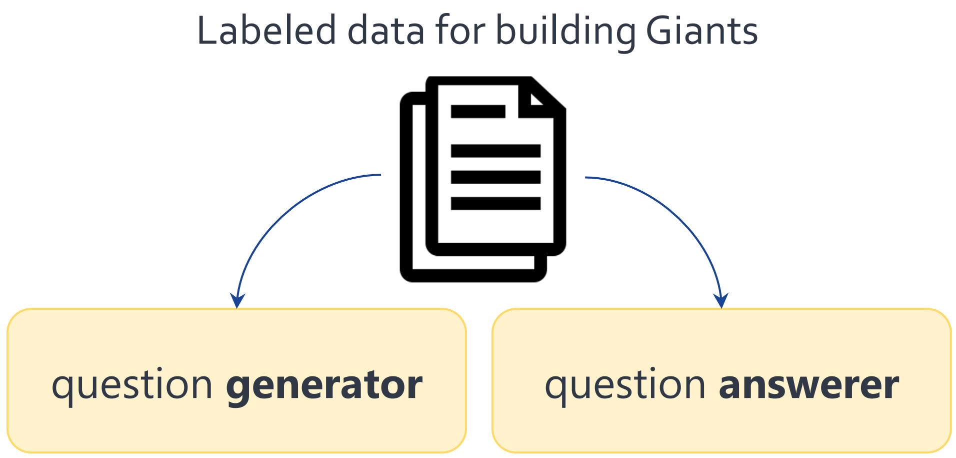
1. Shayne Graham
2. field goals of #1
3. yards of #2
4. number of #2 for each #3
5. #3 where #4 is two

- Costly human annotations
- Not necessarily comprehensible to existing models

Step 1: Language of Existing QA Models

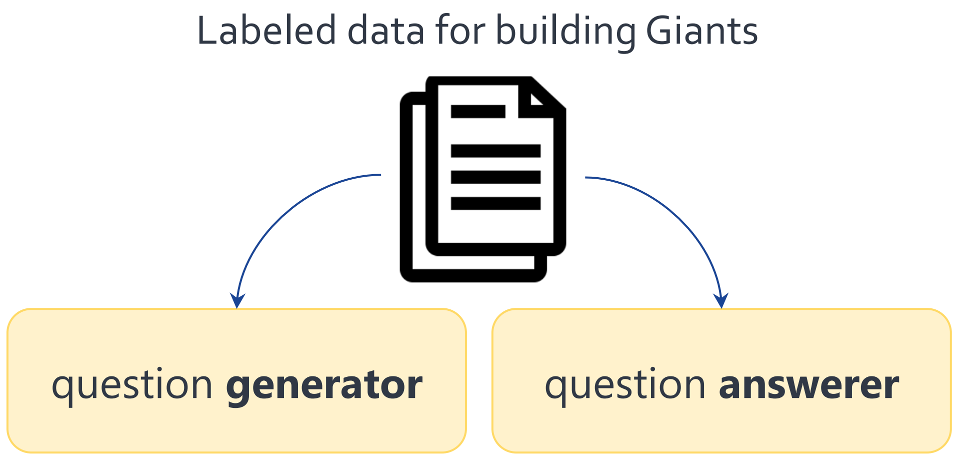


Step 1: Language of Existing QA Models



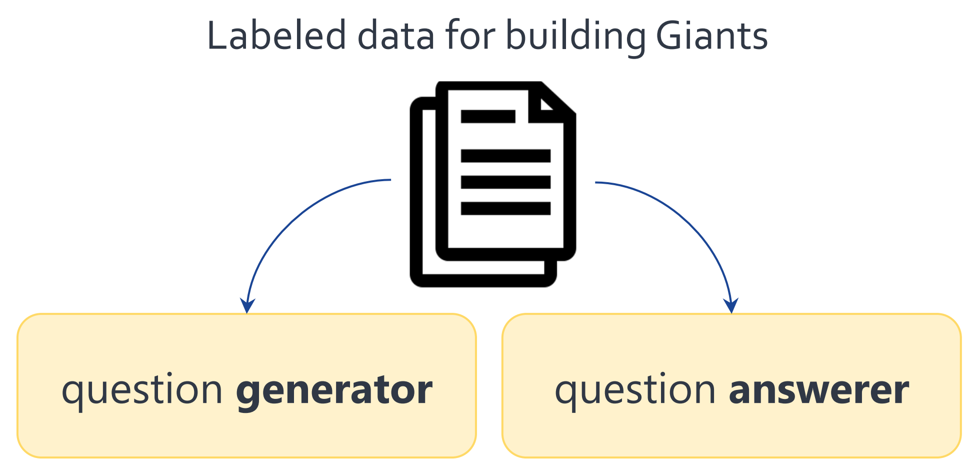
Step 1: Language of Existing QA Models

- How can we build sub-questions that are understandable to the individual modules?
  - Train a model to **generate** the question



Step 1: Language of Existing QA Models

- How can we build sub-questions that are understandable to the individual modules?
  - Train a model to **generate** the question



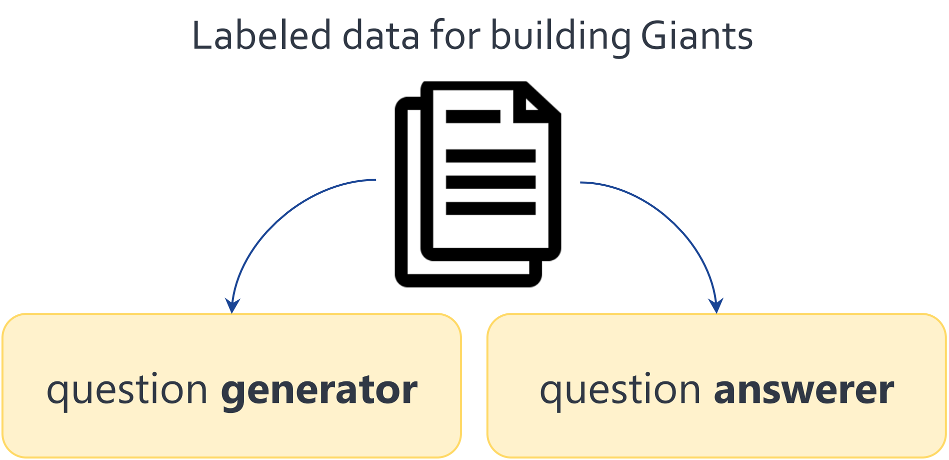

```
QG(q_vocab, exp_ans, doc)
```

Step 1: Language of Existing QA Models

- How can we build sub-questions that are understandable to the individual modules?
  - Train a model to **generate** the question

Suggested vocab to use in Q's

`QG(q_vocab, exp_ans, doc)`



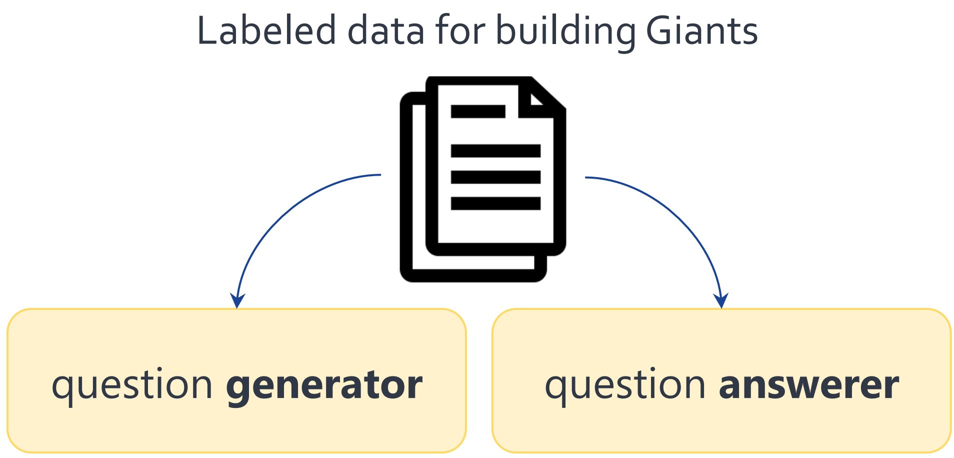
Step 1: Language of Existing QA Models

- How can we build sub-questions that are understandable to the individual modules?
    - Train a model to **generate** the question

Suggested vocab to use in Q's

The expected answer

`QG(q_vocab, exp_ans, doc)`



Step 1: Language of Existing QA Models

- How can we build sub-questions that are understandable to the individual modules?
  - Train a model to **generate** the question

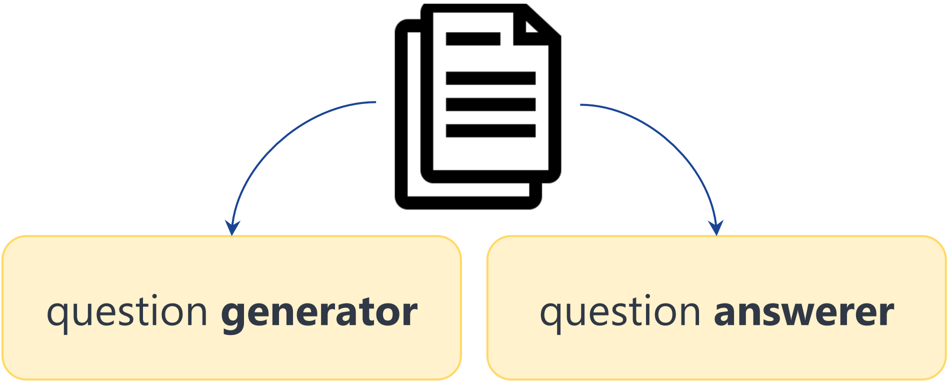
Suggested vocab to use in Q's

The expected answer

Supplemented document

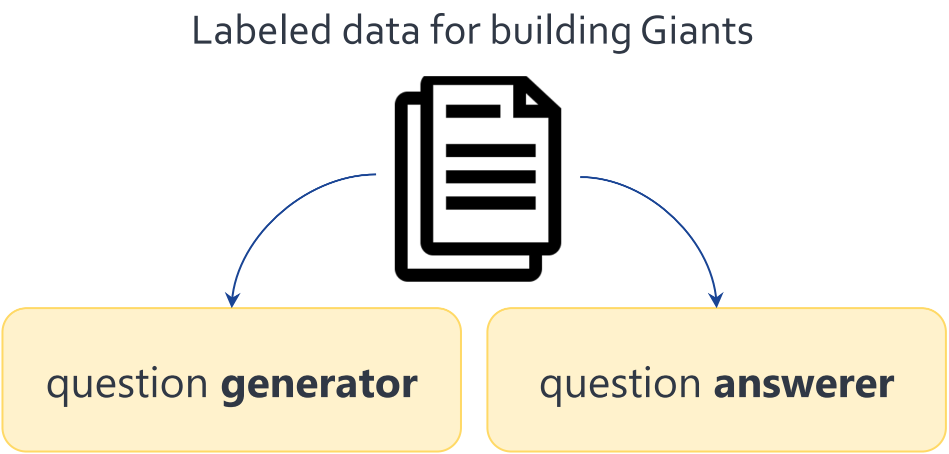
```
QG(q_vocab, exp_ans, doc)
```

Labeled data for building Giants



Step 1: Language of Existing QA Models

- How can we build sub-questions that are understandable to the individual modules?
  - Train a model to **generate** the question



Suggested vocab to use in Q's

The expected answer

Supplemented document

```
QG(q_vocab, exp_ans, doc)
```

```
QG(
    q_vocab=["years","services","sector"],
    exp_ans=2002,
    doc=
)
```

Step 1: Language of Existing QA Models

- How can we build sub-questions that are understandable to the individual modules?
  - Train a model to **generate** the question

Labeled data for building Giants

question **generator**

question **answerer**

Suggested vocab to use in Q's

The expected answer

Supplemented document

```
QG(q_vocab, exp_ans, doc)
```

```
QG(
    q_vocab=["years","services","sector"],
    exp_ans=2002,
    doc=
)
```

When did the services sector take a hit?
When did the services sector take a downturn?
When did the services sector take a big hit?
....

Step 1: Language of Existing QA Models

- How can we build sub-questions that are understandable to the individual modules?
  - Train a model to **generate** the question

Suggested vocab to use in Q's

The expected answer

Supplemented document

QG(q_vocab, exp_ans, doc)

```
QG(
    q_vocab=["years","services","sector"],
    exp_ans=2002,
    doc=
)
```

2003

Labeled data for building Giants

question **generator**

question **answerer**

When did the services sector take a hit?
When did the services sector take a downturn?
When did the services sector take a big hit?
....

Step 1: Language of Existing QA Models

- How can we build sub-questions that are understandable to the individual modules?
  - Train a model to **generate** the question

Labeled data for building Giants

question **generator**

question **answerer**

Suggested vocab to use in Q's

The expected answer

Supplemented document

```
QG(q_vocab, exp_ans, doc)
```

```
QG(
    q_vocab=["years","services","sector"],
    exp_ans=2002,
    doc=
)
```

~~2002~~ 2003

When did the services sector take a hit?
When did the services sector take a downturn?

In what years did the services sector rebound?
In what year did the services sector rebound?
....

Step 2: Typing Complex Questions

- Infer the [complex] question type via heuristics

Step 2: Typing Complex Questions

- Infer the [complex] question type via heuristics

Question type	Example
Difference questions	*How many years did it take for the services sector to rebound?*
Comparison questions	*Which ancestral group is smaller: Irish or Italian?*
Complementation questions	*How many percent of the national population does not live in Bangkok?*
Composition questions	*What was the nationality of the director of the "Little Big Girl" episode of "The Simpsons"?*
Conjunction questions	*Who is a politician and an actor?*

Step 2: Typing Complex Questions

- Infer the [complex] question type via heuristics

Question type	Example
Difference questions	*How many years did it take for the services sector to rebound?*
Comparison questions	*Which ancestral group is smaller: Irish or Italian?*
Complementation questions	*How many percent of the national population does not live in Bangkok?*
Composition questions	*What was the nationality of the director of the "Little Big Girl" episode of "The Simpsons"?*
Conjunction questions	*Who is a politician and an actor?*

High-level and used across datasets

Step 2: Typing Complex Questions

- Infer the [complex] question type via heuristics

Question type	Example
👉 Difference questions	*How many years did it take for the services sector to rebound?*
Comparison questions	*Which ancestral group is smaller: Irish or Italian?*
Complementation questions	*How many percent of the national population does not live in Bangkok?*
Composition questions	*What was the nationality of the director of the "Little Big Girl" episode of "The Simpsons"?*
Conjunction questions	*Who is a politician and an actor?*

High-level and used across datasets

Step 3: Generating [Noisy] Training Data

- Form chains of sub-questions, based on the inferred type

Complex question and its answer:

- **Question:** How many years did it take for the services sector to rebound?
- **Answer:** 1

Step 3: Generating [Noisy] Training Data

- Form chains of sub-questions, based on the inferred type

Complex question and its answer:

- **Question:** How many years did it take for the services sector to rebound?
- **Answer:** `1`

```
Question Type=difference
```

Step 3: Generating [Noisy] Training Data

- Form chains of sub-questions, based on the inferred type

Complex question and its answer:

- **Question:** How many years did it take for the services sector to rebound?
- **Answer:** 1

```
Question Type=difference
```

```
q1 = QG(q_vocab, exp_ans=n1, doc)
q2 = QG(q_vocab, exp_ans=n2, doc)
q3 = calc(diff, n1, n2)
```

Step 3: Generating [Noisy] Training Data

- Form chains of sub-questions, based on the inferred type

Complex question and its answer:

- **Question:** How many years did it take for the services sector to rebound?
- **Answer:** 1

q_vocab=non-stop words from complex question

```
Question Type=difference
```

```
q1 = QG(q_vocab, exp_ans=n1, doc)
q2 = QG(q_vocab, exp_ans=n2, doc)
q3 = calc(diff, n1, n2)
```

Step 3: Generating [Noisy] Training Data

- Form chains of sub-questions, based on the inferred type

Complex question and its answer:

- **Question:** How many years did it take for the services sector to rebound?
- **Answer:** 1

```
Question Type=difference
```

q_vocab=non-stop words from complex question

```
q1 = QG(q_vocab, exp_ans=n1, doc)
q2 = QG(q_vocab, exp_ans=n2, doc)
q3 = calc(diff, n1, n2)
```

n1 and n2: numbers extracted from doc with difference equal to the final answer.

Step 4: Filtering the [Noisy] Training Data

- Filter out undesirable chains:
  - Too many question words **not** used
  - Too many **new** words introduced

Complex question and its answer:

- **Question:** How many years did it take for the services sector to rebound?
- **Answer:** 1

```
Question Type=difference
```

```
q1 = QG(q_vocab, exp_ans=n1, doc)
q2 = QG(q_vocab, exp_ans=n2, doc)
q3 = calc(diff, n1, n2)
```

```
Filtering
noisy
chains
```

Training the Model

Train the model to generate future sub-questions, given the past ones.

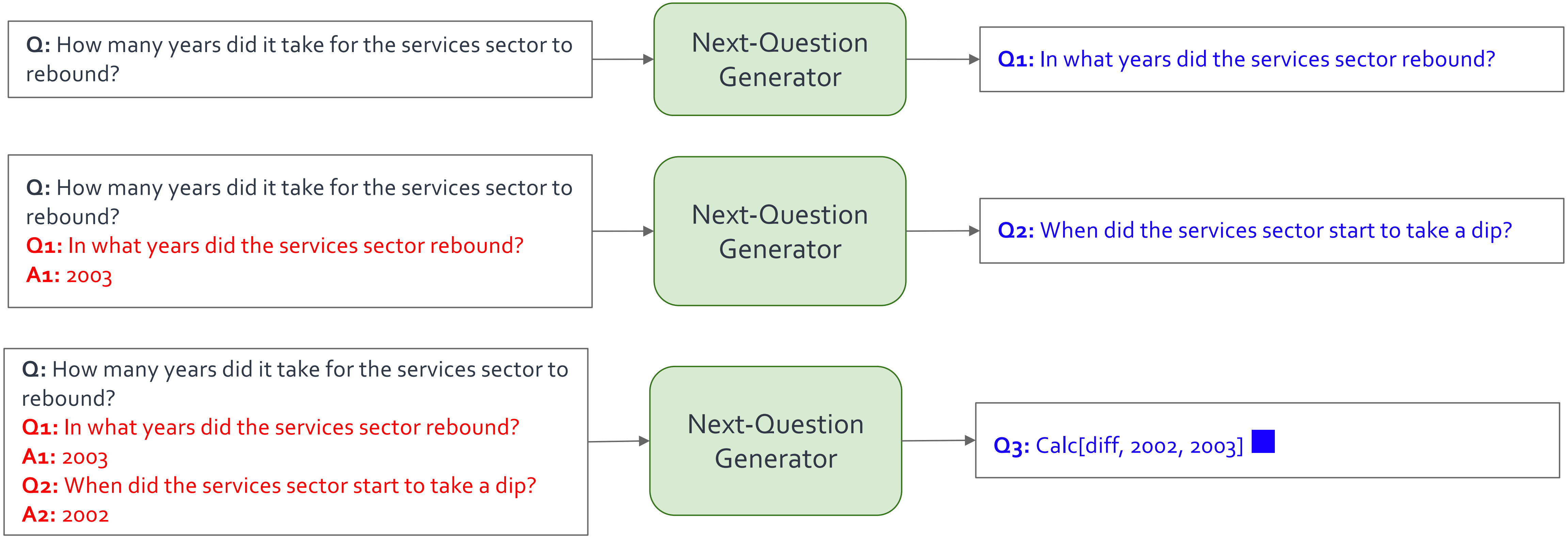
Q: How many years did it take for the services sector to rebound?
→ Next-Question Generator →
Q1: In what years did the services sector rebound?

Q: How many years did it take for the services sector to rebound?
Q1: In what years did the services sector rebound?
A1: 2003
→ Next-Question Generator →
Q2: When did the services sector start to take a dip?

Q: How many years did it take for the services sector to rebound?
Q1: In what years did the services sector rebound?
A1: 2003
Q2: When did the services sector start to take a dip?
A2: 2002
→ Next-Question Generator →
Q3: Calc[diff, 2002, 2003] ■

Training the Model

Train the model to generate future sub-questions, given the past ones.



Training the Model

Train the model to generate future sub-questions, given the past ones.

Q: How many years did it take for the services sector to rebound?

→ Next-Question Generator →

Q1: In what years did the services sector rebound?

Q: How many years did it take for the services sector to rebound?
Q1: In what years did the services sector rebound?
A1: 2003

→ Next-Question Generator →

Q2: When did the services sector start to take a dip?

Q: How many years did it take for the services sector to rebound?
Q1: In what years did the services sector rebound?
A1: 2003
Q2: When did the services sector start to take a dip?
A2: 2002

→ Next-Question Generator →

Q3: Calc[diff, 2002, 2003] ■

Training the Model

Train the model to generate future sub-questions, given the past ones.

Q: How many years did it take for the services sector to rebound? → Next-Question Generator → **Q1:** In what years did the services sector rebound?

Q: How many years did it take for the services sector to rebound?
Q1: In what years did the services sector rebound?
A1: 2003
→ Next-Question Generator → **Q2:** When did the services sector start to take a dip?

Q: How many years did it take for the services sector to rebound?
Q1: In what years did the services sector rebound?
A1: 2003
Q2: When did the services sector start to take a dip?
A2: 2002
→ Next-Question Generator → **Q3:** Calc[diff, 2002, 2003] ■

ModularQA System

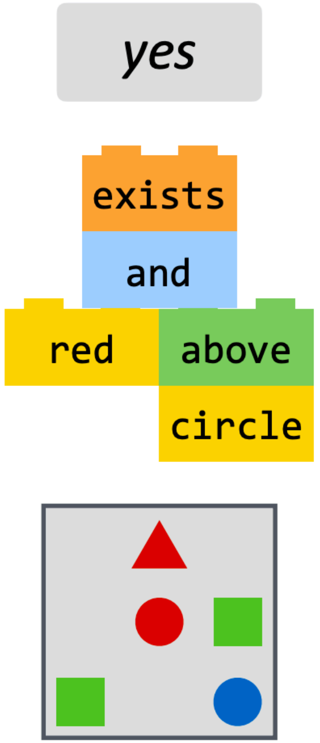
- Uses BART-Large for sub-question generation

- QA modules
  - Roberta model trained on SQuAD 2.0
  - Math Calculator with three key functions: x-y, 100-x, if-then-else

- Target datasets:
  - DROP [Dua et al. 19]
  - HotPotQA [Yang et al. 18]

Existing Modular Architectures

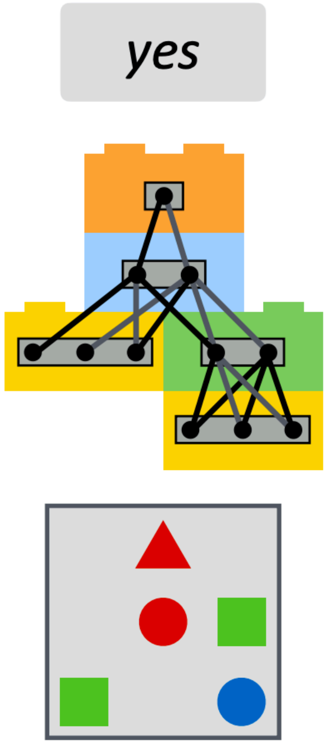
- Neural Module Networks [Andreas et al. 16]
  - Communicate through dense vectors
    - (e.g., attention weights)



Is there a red shape above a circle?

Existing Modular Architectures

- Neural Module Networks [Andreas et al. 16]
  - Communicate through dense vectors
    - (e.g., attention weights)



Is there a red shape above a circle?

Experiment: Comparison w/ Existing Models

System	Evaluated on	
	DROP (F1)	HotPotQA (F1)
NMN-D [Gupta et al. 20]	79.1	?
S-NMN [Jiang and Bansal. 19]	?	63.1
NumNet+V2 [Ran et al. 19]	91.6	?
Quark [Groeneveld et al. 20]	?	75.5
ModularQA [this work]	87.9	61.8

Experiment: Comparison w/ Existing Models

modular baselines

System	Evaluated on	
	DROP (F1)	HotPotQA (F1)
NMN-D [Gupta et al. 20]	79.1	?
S-NMN [Jiang and Bansal. 19]	?	63.1
NumNet+V2 [Ran et al. 19]	91.6	?
Quark [Groeneveld et al. 20]	?	75.5
ModularQA [this work]	87.9	61.8

Experiment: Comparison w/ Existing Models

modular baselines

System	Evaluated on	
	DROP (F1)	HotPotQA (F1)
NMN-D [Gupta et al. 20]	79.1	?
S-NMN [Jiang and Bansal. 19]	?	63.1
NumNet+V2 [Ran et al. 19]	91.6	?
Quark [Groeneveld et al. 20]	?	75.5
ModularQA [this work]	87.9	61.8

Experiment: Comparison w/ Existing Models

modular baselines

System	Evaluated on	
	DROP (F1)	HotPotQA (F1)
NMN-D [Gupta et al. 20]	79.1	?
S-NMN [Jiang and Bansal. 19]	?	63.1
NumNet+V2 [Ran et al. 19]	91.6	?
Quark [Groeneveld et al. 20]	?	75.5
ModularQA [this work]	87.9	61.8

Experiment: Comparison w/ Existing Models

modular baselines

System	Evaluated on	
	DROP (F1)	HotPotQA (F1)
NMN-D [Gupta et al. 20]	79.1	?
S-NMN [Jiang and Bansal. 19]	?	63.1
NumNet+V2 [Ran et al. 19]	91.6	?
Quark [Groeneveld et al. 20]	?	75.5
ModularQA [this work]	87.9	61.8

Experiment: Comparison w/ Existing Models

- ModularQA is **competitive** with other **modular** approaches
- Performs not far from black-box models that use dataset specific assumptions
- More experiments in the paper:
  - Higher Robustness
  - Learning with Less Data

System	Evaluated on DROP (F1)	Evaluated on HotPotQA (F1)
NMN-D [Gupta et al. 20]	79.1	?
S-NMN [Jiang and Bansal. 19]	?	63.1
NumNet+V2 [Ran et al. 19]	91.6	?
Quark [Groeneveld et al. 20]	?	75.5
ModularQA [this work]	87.9	61.8

modular baselines

Experiment: Comparison w/ Existing Models

- ModularQA is **competitive** with other **modular** approaches
- Performs not far from black-box models that use dataset specific assumptions
- More experiments in the paper:
  - Higher Robustness
  - Learning with Less Data

	System	Evaluated on	
		DROP (F1)	HotPotQA (F1)
modular baselines	**NMN-D** [Gupta et al. 20]	79.1	?
modular baselines	**S-NMN** [Jiang and Bansal. 19]	?	63.1
black-box baselines	**NumNet+V2** [Ran et al. 19]	91.6	?
black-box baselines	**Quark** [Groeneveld et al. 20]	?	75.5
	ModularQA [this work]	87.9	61.8

Experiment: Comparison w/ Existing Models

- ModularQA is **competitive** with other **modular** approaches
- Performs not far from black-box models that use dataset specific assumptions
- More experiments in the paper:
  - Higher Robustness
  - Learning with Less Data

	System	Evaluated on: DROP (F1)	Evaluated on: HotPotQA (F1)
modular baselines	**NMN-D** [Gupta et al. 20]	79.1	?
modular baselines	**S-NMN** [Jiang and Bansal. 19]	?	63.1
black-box baselines	**NumNet+V2** [Ran et al. 19]	91.6	?
black-box baselines	**Quark** [Groeneveld et al. 20]	?	75.5
	ModularQA [this work]	87.9	61.8

Lessons

- Text Modular Networks, a general-purpose framework
  - Complex tasks solved as textual interaction between existing modules
  - ModularQA, an insanitation of this framework

- Benefits:
  - First interpretable model for DROP and HotpotQA → more breadth!
  - Competitive with existing approaches

Lessons

- Text Modular Networks, a general-purpose framework
  - Complex tasks solved as textual interaction between existing modules
  - ModularQA, an insanitation of this framework

- Benefits:
  - First interpretable model for DROP and HotpotQA → more breadth!
  - Competitive with existing approaches

Lessons

- Text Modular Networks, a general-purpose framework
  - Complex tasks solved as textual interaction between existing modules
  - ModularQA, an insanitation of this framework

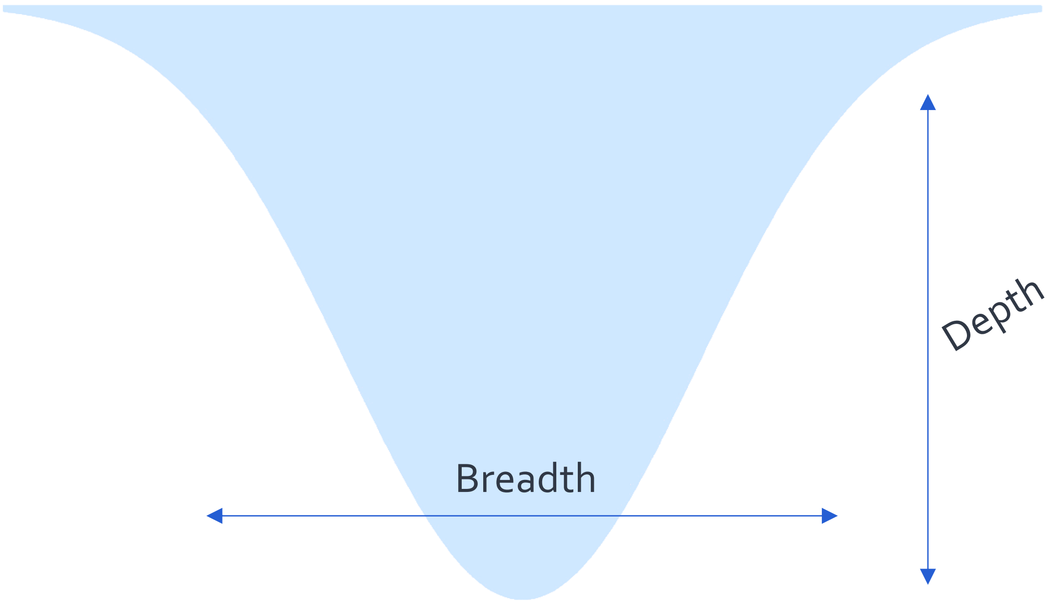
- Benefits:
  - First interpretable model for DROP and HotpotQA → more breadth!
  - Competitive with existing approaches

Tying the Loose Ends

- Currently, we do not focus enough on the "breadth" of our progress.
  - Obsessed with depth (e.g., chasing leaderboards for individual tasks)

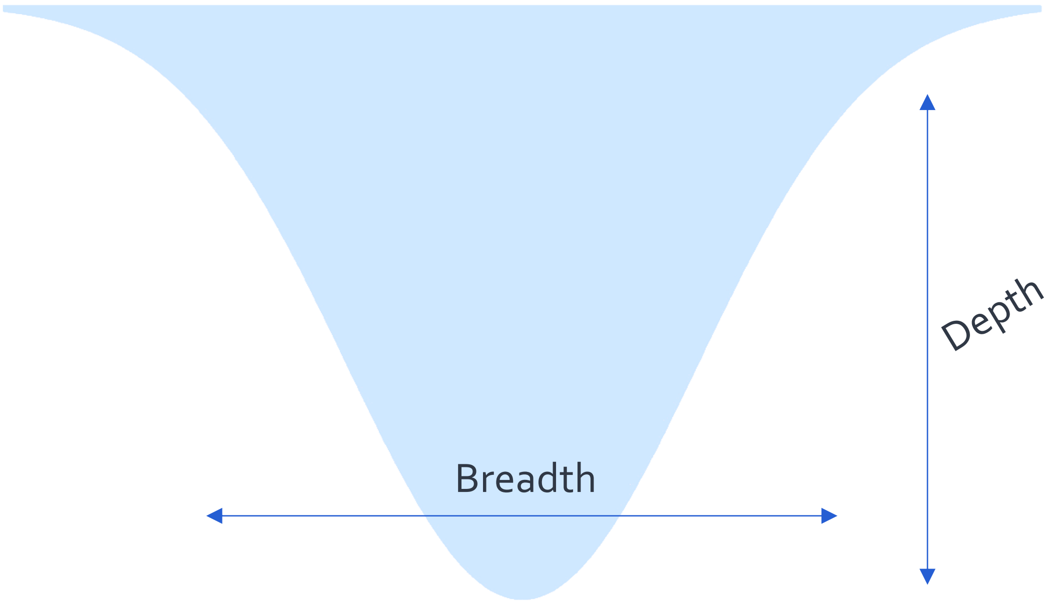
- The two works presented here:
  - UnifiedQA: broader range of tasks
  - ModularQA: utilizing existing modules for more complex tasks

- Not just two systems!



Tying the Loose Ends

- Currently, we do not focus enough on the “breadth” of our progress.
  - Obsessed with depth (e.g., chasing leaderboards for individual tasks)
- The two works presented here:
  - UnifiedQA: broader range of tasks
  - ModularQA: utilizing existing modules for more complex tasks
- Not just two systems!

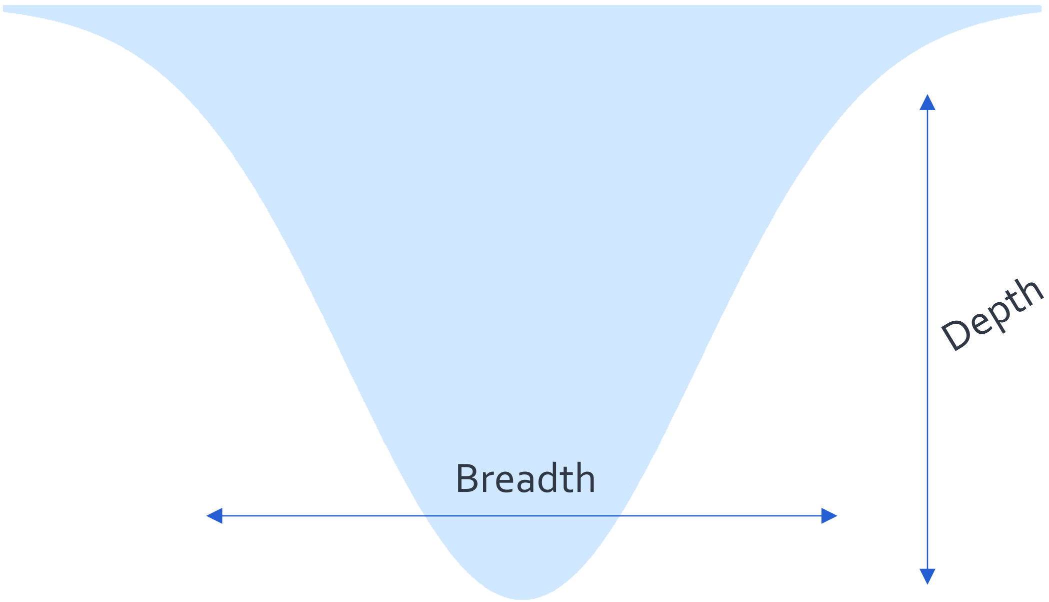


Tying the Loose Ends

- Currently, we do not focus enough on the “breadth” of our progress.
  - Obsessed with depth (e.g., chasing leaderboards for individual tasks)

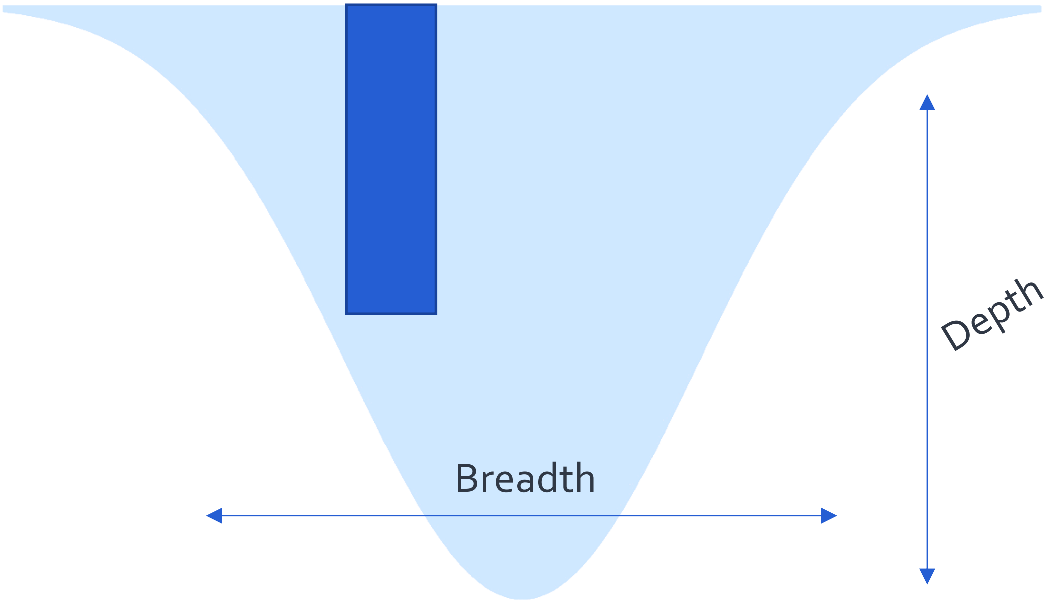
- The two works presented here:
  - UnifiedQA: broader range of tasks
  - ModularQA: utilizing existing modules for more complex tasks

- Not just two systems!



Tying the Loose Ends

- Currently, we do not focus enough on the "breadth" of our progress.
  - Obsessed with depth (e.g., chasing leaderboards for individual tasks)
- The two works presented here:
  - UnifiedQA: broader range of tasks
  - ModularQA: utilizing existing modules for more complex tasks
- Not just two systems!

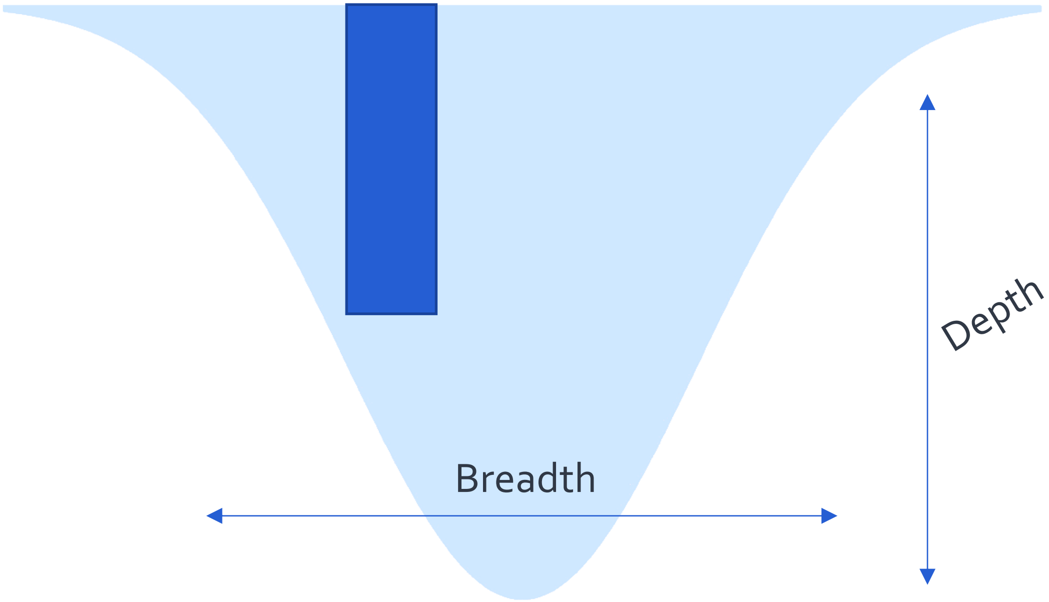


Tying the Loose Ends

- Currently, we do not focus enough on the “breadth” of our progress.
    - Obsessed with depth (e.g., chasing leaderboards for individual tasks)

- The two works presented here:
    - UnifiedQA: broader range of tasks
    - ModularQA: utilizing existing modules for more complex tasks

- Not just two systems!

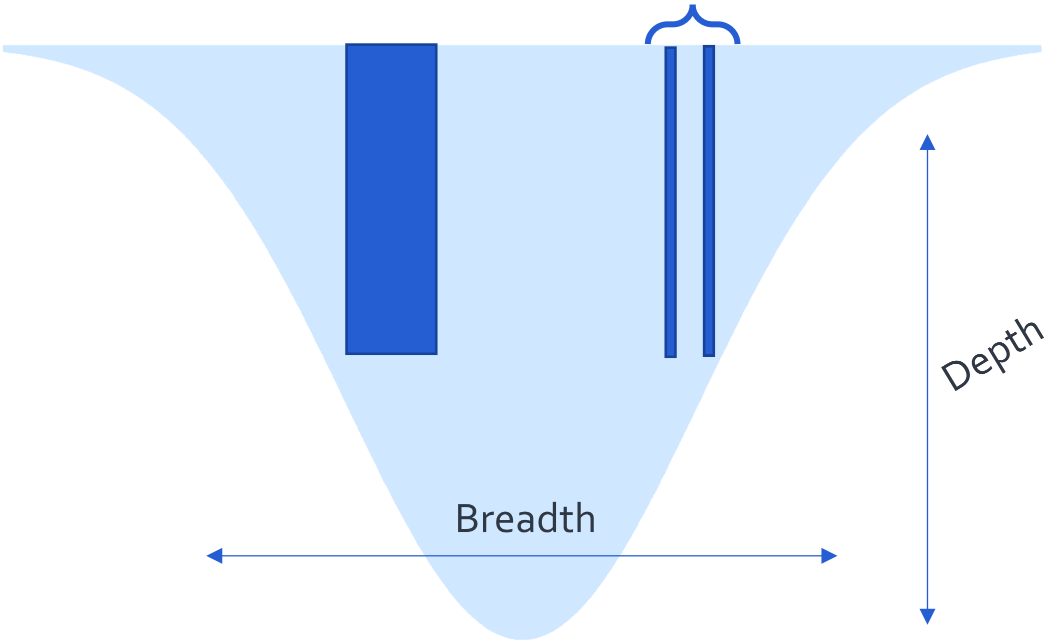


Tying the Loose Ends

- Currently, we do not focus enough on the "breadth" of our progress.
  - Obsessed with depth (e.g., chasing leaderboards for individual tasks)

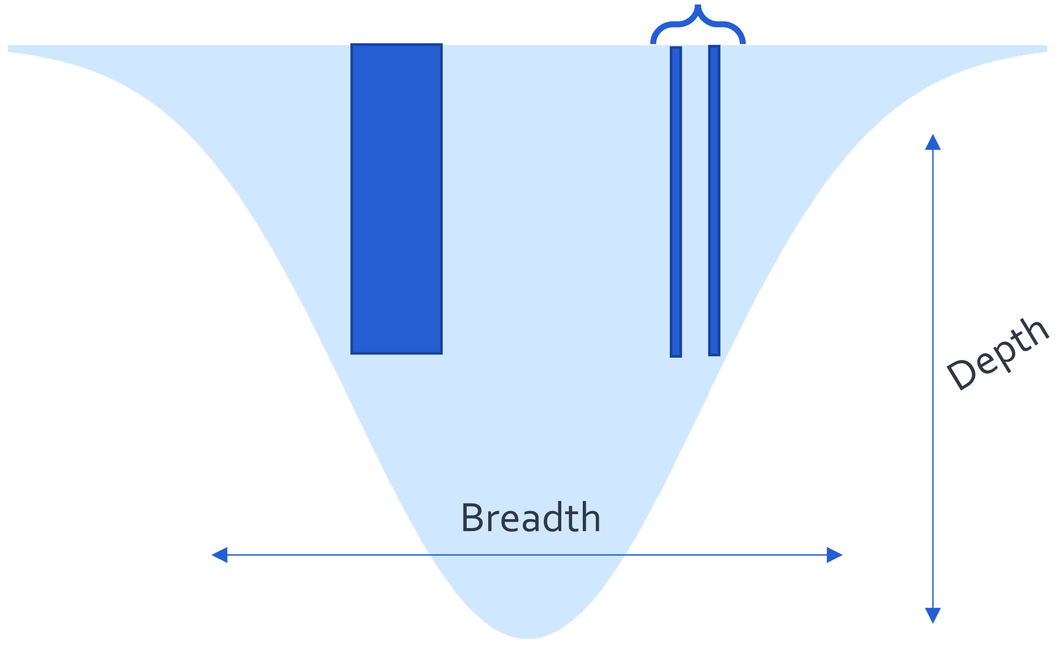
- The two works presented here:
  - UnifiedQA: broader range of tasks
  - ModularQA: utilizing existing modules for more complex tasks

- Not just two systems!



Tying the Loose Ends

- Currently, we do not focus enough on the "breadth" of our progress.
  - Obsessed with depth (e.g., chasing leaderboards for individual tasks)
- The two works presented here:
  - UnifiedQA: broader range of tasks
  - ModularQA: utilizing existing modules for more complex tasks
- Not just two systems!



Big Picture

Models of Language Problems

KMKKTCH. EMNLP-Findings’20

KCRUR. NAACL’18

PKR. NAACL’15

Models of Language Problems

KMKKTCH. EMNLP-Findings'20
KCRUR. NAACL'18
PKR. NAACL'15

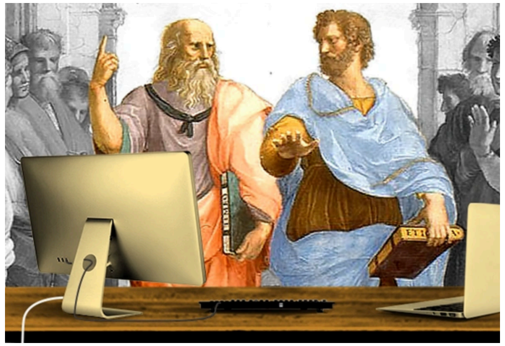
Measuring Our Progress

GKSKRB. TACL'21
ZKQR. EMNLP'19
KCRUR. NAACL'18

Implicit decompositions dataset

[Geva et al. TACL'21]

Did Aristotle Use a Laptop?



Models of Language Problems

KMKKTCH. EMNLP-Findings'20
KCRUR. NAACL'18
PKR. NAACL'15

Measuring Our Progress

GKSKRB. TACL'21
ZKQR. EMNLP'19
KCRUR. NAACL'18

Models of Language Problems

KMKKTCH. EMNLP-Findings'20
KCRUR. NAACL'18
PKR. NAACL'15

Measuring Our Progress

GKSKRB. TACL'21
ZKQR. EMNLP'19
KCRUR. NAACL'18

Models of Language Problems

KMKKTCH. EMNLP-Findings'20
KCRUR. NAACL'18
PKR. NAACL'15

Measuring Our Progress

GKSKRB. TACL'21
ZKQR. EMNLP'19
KCRUR. NAACL'18

Analyses

G et al. EMNLP-Findings'20
KKS. EMNLP'20

Characterizing models' decision boundaries

[Gardner et al. EMNLP-Findings'20]

three differently

~~Two~~ ~~similarly~~-colored and similarly-posed chow ~~dogs~~ are face to face in one image.

cats

Models of Language Problems

KMKKTCH. EMNLP-Findings'20
KCRUR. NAACL'18
PKR. NAACL'15

Measuring Our Progress

GKSKRB. TACL'21
ZKQR. EMNLP'19
KCRUR. NAACL'18

Analyses

G et al. EMNLP-Findings'20
KKS. EMNLP'20

Models of Language Problems

KMKKTCH. EMNLP-Findings’20
KCRUR. NAACL’18
PKR. NAACL’15

Measuring Our Progress

GKSKRB. TACL’21
ZKQR. EMNLP’19
KCRUR. NAACL’18

Analyses

G et al. EMNLP-Findings’20
KKS. EMNLP’20

Models of Language Problems

KMKKTCH. EMNLP-Findings'20
KCRUR. NAACL'18
PKR. NAACL'15

Measuring Our Progress

GKSKRB. TACL'21
ZKQR. EMNLP'19
KCRUR. NAACL'18

Analyses

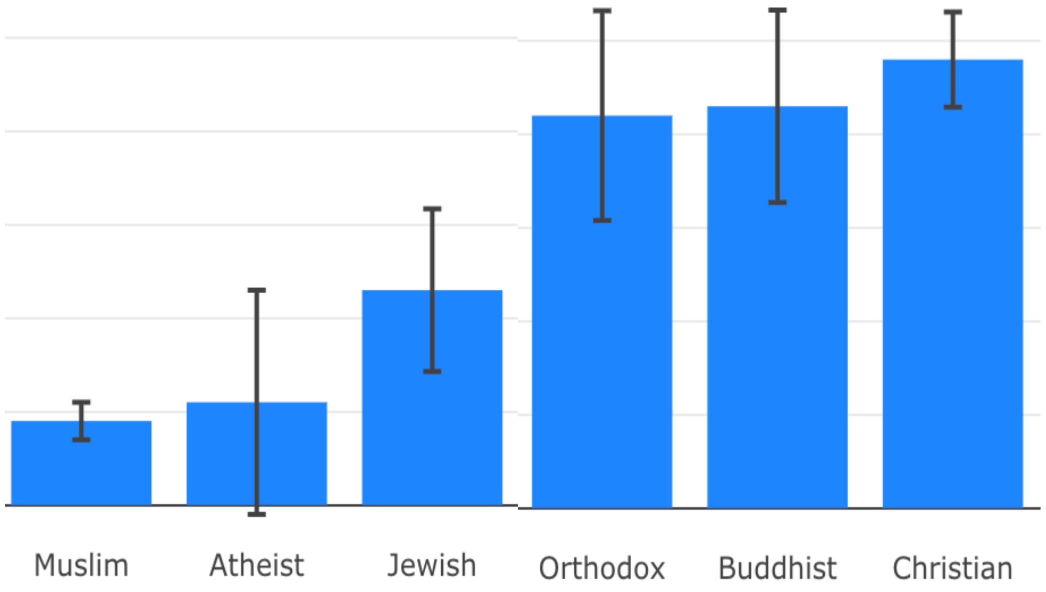
G et al. EMNLP-Findings'20
KKS. EMNLP'20

NLP+Society

LKKSS. EMNLP-Findings'20
CKWCR. NAACL'19

Social Biases in QA Models

[Li et al. EMNLP-Findings’20]



Association of ethylic/religious groups with negative stereotypes

Models of Language Problems

KMKKTCH. EMNLP-Findings’20
KCRUR. NAACL’18
PKR. NAACL’15

Measuring Our Progress

GKSKRB. TACL’21
ZKQR. EMNLP’19
KCRUR. NAACL’18

Analyses

G et al. EMNLP-Findings’20
KKS. EMNLP’20

NLP+Society

LKKSS. EMNLP-Findings’20
CKWCR. NAACL’19

Models of Language Problems

KMKKTCH. EMNLP-Findings'20
KCRUR. NAACL'18
PKR. NAACL'15

Measuring Our Progress

GKSKRB. TACL'21
ZKQR. EMNLP'19
KCRUR. NAACL'18

Analyses

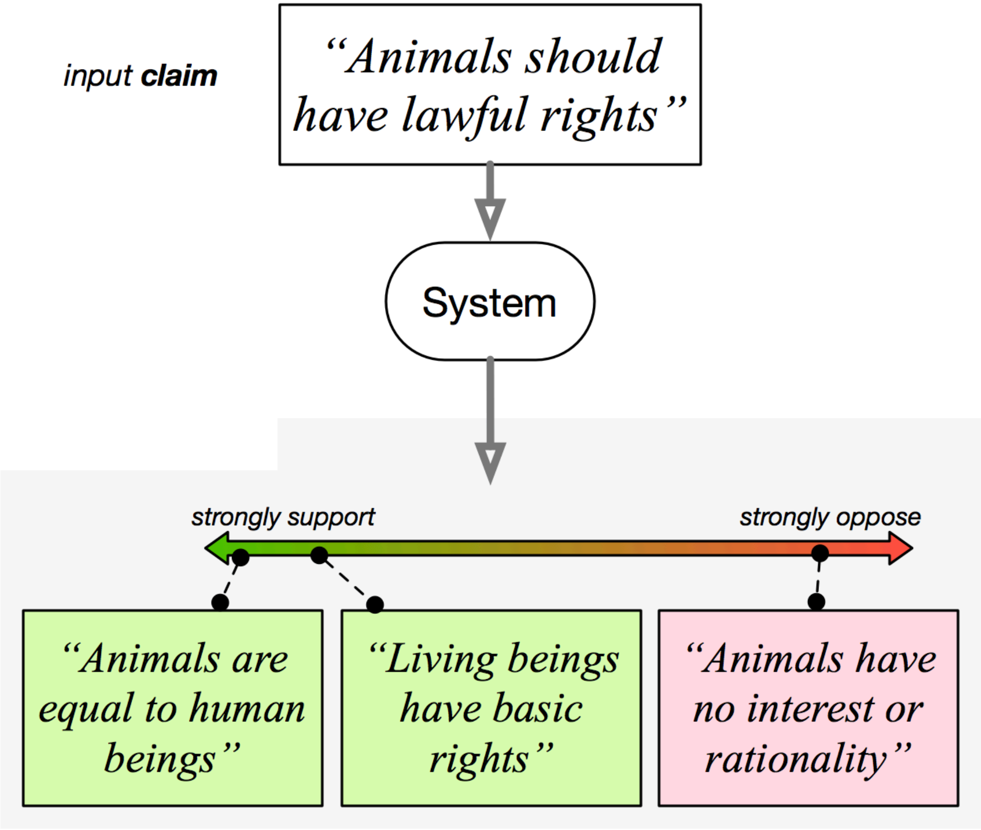
G et al. EMNLP-Findings'20
KKS. EMNLP'20

NLP+Society

LKKSS. EMNLP-Findings'20
CKWCR. NAACL'19

Diverse Perspective Discovery

[Chen et al. NAACL'19]



Diverse perspectives to address the given claim.

Models of Language Problems

KMKKTCH. EMNLP-Findings'20
KCRUR. NAACL'18
PKR. NAACL'15

Measuring Our Progress

GKSKRB. TACL'21
ZKQR. EMNLP'19
KCRUR. NAACL'18

Analyses

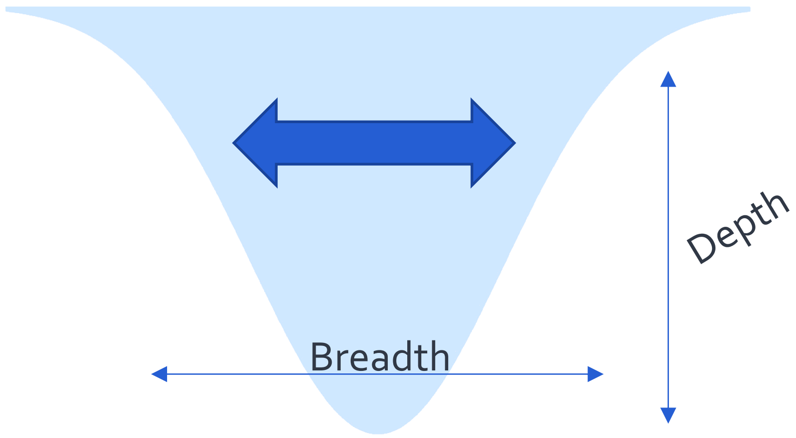
G et al. EMNLP-Findings'20
KKS. EMNLP'20

NLP+Society

LKKSS. EMNLP-Findings'20
CKWCR. NAACL'19

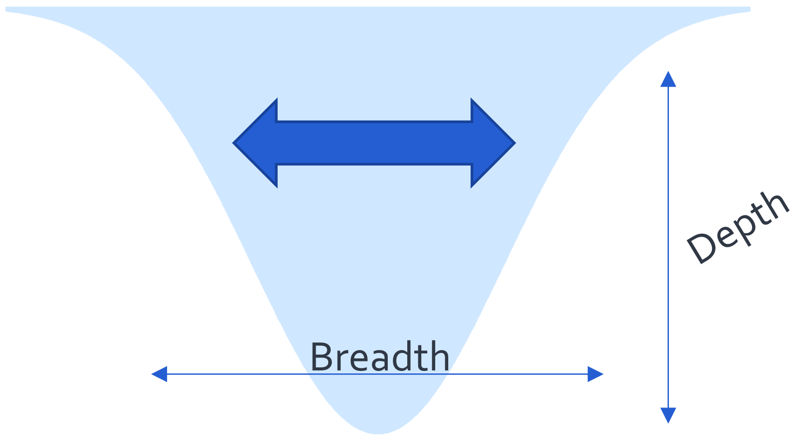
Look Ahead

Better Systems



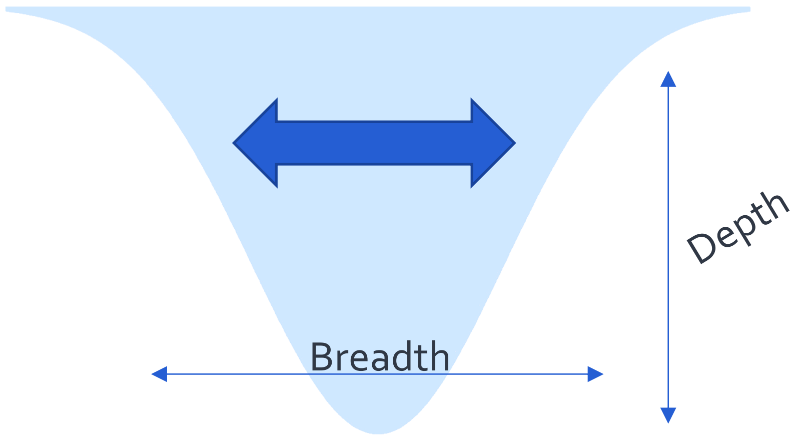
Better Systems

- Continue towards broader scope for QA models
  - **Broadness:** how to cover a larger range of "natural" variations of QA?
  - **Reliability:** we can we quantify what model [un]certainty?
  - **Faithful Explainability:** can we get explanations that are faithful to models' reasoning?
  - **Efficiency:** Can we build small, yet accurate models?



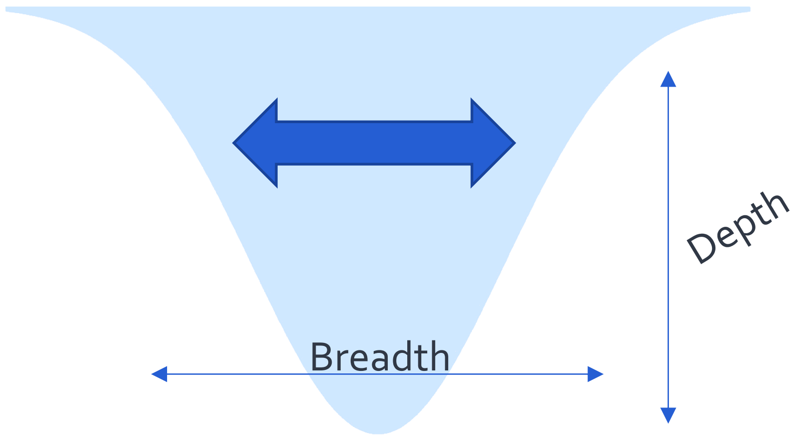
Better Systems

- Continue towards broader scope for QA models
  - **Broadness:** how to cover a larger range of "natural" variations of QA?
  - **Reliability:** we can we quantify what model [un]certainty?
  - **Faithful Explainability:** can we get explanations that are faithful to models' reasoning?
  - **Efficiency:** Can we build small, yet accurate models?



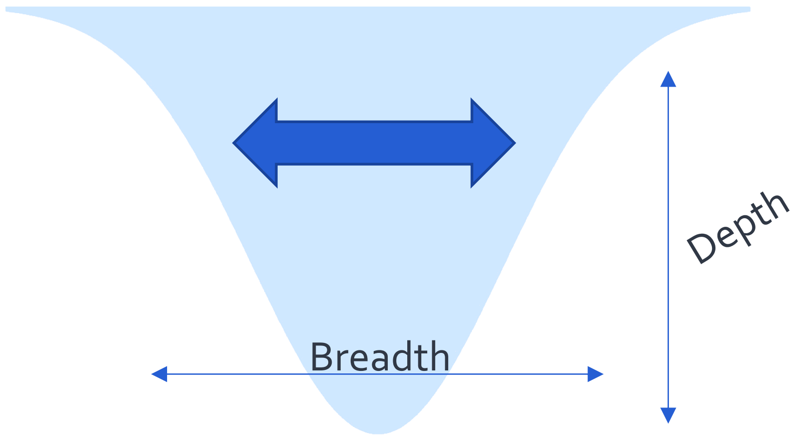
Better Systems

- Continue towards broader scope for QA models
  - **Broadness:** how to cover a larger range of “natural” variations of QA?
  - **Reliability:** we can we quantify what model [un]certainty?
  - **Faithful Explainability:** can we get explanations that are faithful to models’ reasoning?
  - **Efficiency:** Can we build small, yet accurate models?



Better Systems

- Continue towards broader scope for QA models
  - **Broadness:** how to cover a larger range of “natural” variations of QA?
  - **Reliability:** we can we quantify what model [un]certainty?
  - **Faithful Explainability:** can we get explanations that are faithful to models’ reasoning?
  - **Efficiency:** Can we build small, yet accurate models?



Learning from Instructions

Input-output supervision

instructions

Learning from Instructions

$$\{(\mathbf{x},\mathbf{y})\} \Rightarrow$$

( ,"spam")

( , "ham")

( ,"spam")

$\vdots$

Input-output supervision

instructions

Learning from Instructions

$\{(\mathbf{x},\mathbf{y})\}$ →

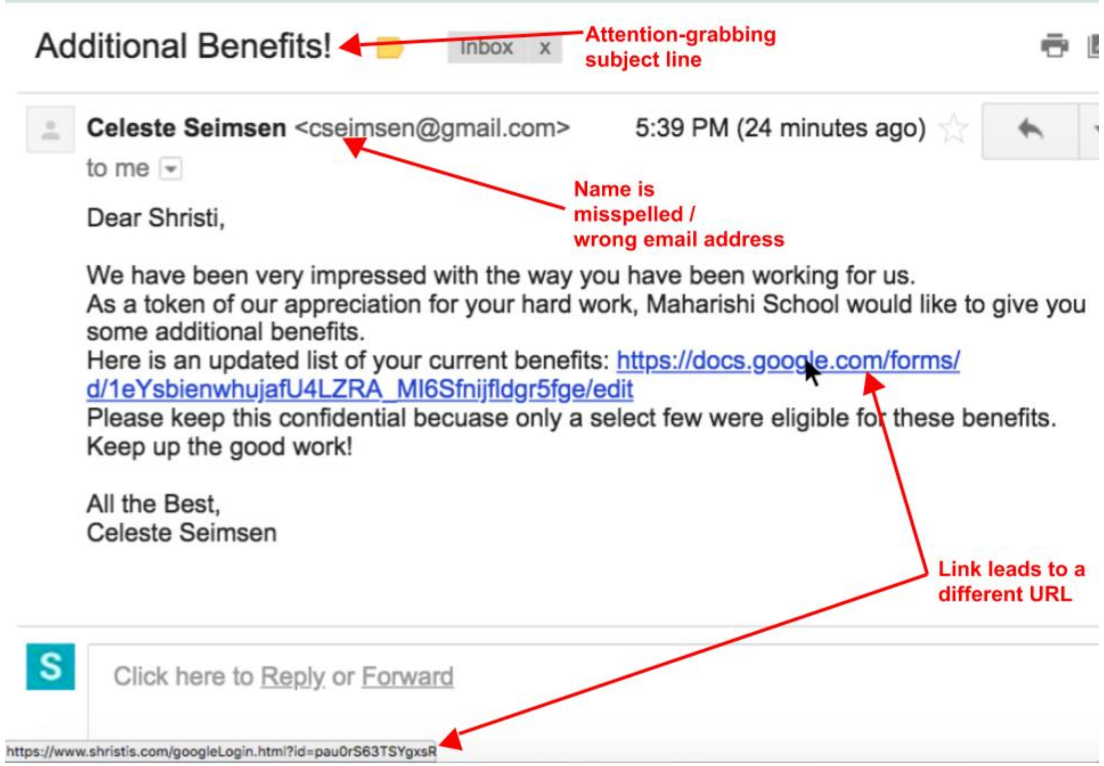
( ,"spam")

( , "ham")

( ,"spam")

⋮

Input-output supervision



instructions

Interactive Semantics

Single-shot evaluation

Learning from interactions

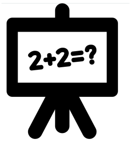
Interactive Semantics

Single-shot evaluation

Learning from interactions

Interactive Semantics

Single-shot evaluation



input problem

evaluation

Learning from interactions

That’s it!