REIGN: Robust Training for Conversational Question Answering Models using REInforced Reformulation GeneratioN

Magdalena Kaiser, Rishiraj Saha Roy and Gerhard Weikum
Max Planck Institute for Informatics, Saarbrücken, Germany
Conversational Question Answering (ConvQA)

Consider context

- Sequential, multi-turn QA
- Incomplete follow-up questions
- Challenges:
  - Implicit context
  - Ad hoc formulations

Q1: What’s the 2022 LOTR TV series called?
A1: The Rings of Power (TROP)

Q2: TROP airing on?
A2: Netflix

Q3: Which actor plays Isildur in the series?
A3: Harry Sinclair
Conversational Question Answering (ConvQA)

Consider diverse formulations

- Common solution: Data augmentation

  Q1: What’s the 2022 LOTR TV series called?
  A1: The Rings of Power (TROP)

  Q2: TROP airing on?
  Q21: Which streaming service showed TROP?
  Q22: TROP available on which network?
  Q23: On which platform is the Rings of Power airing?
  Q24: Rings of Power broadcasted where?
  Q25: Where can I stream the LOTR TV series?
Conversational Question Answering (ConvQA)

Consider diverse formulations

- Common solution: Data augmentation
- Drawbacks with classical data augmentation:
  - not model-specific
  - can be inefficient
  - challenging for ConvQA

Q1: What’s the 2022 LOTR TV series called?
A1: The Rings of Power (TROP)

Q2: TROP airing on?
Q21: Which streaming service showed TROP?
Q22: TROP available on which network?
Q23: On which platform is the Rings of Power airing?
Q24: Rings of Power broadcasted where?
Q25: Where can I stream the LOTR TV series?
Conversational Question Answering (ConvQA)

Consider diverse formulations

- Common solution: Data augmentation

- Drawbacks with classical data augmentation:
  - not model-specific
  - can be inefficient
  - challenging for ConvQA

→ Only select **subset of reformulations most helpful** for specific model

Q1: What’s the 2022 LOTR TV series called?
A1: The Rings of Power (TROP)

Q2: TROP airing on?

Q21: Which **streaming service** showed TROP?

Q22: TROP available on which **network**?

Q23: On which **platform** is the Rings of Power airing?

Q24: Rings of Power broadcasted where?

Q25: Where can I stream the LOTR TV series?
Conversational Question Answering (ConvQA)

Consider diverse formulations

Goal: **Train** a more **robust**
ConvQA model using a
**model-specific set of**
reformulations

Q1: What’s the 2022 LOTR TV series called?
A1: The Rings of Power (TROP)

Q2: TROP airing on?
A2: Amazon Prime Video

Q3: Which actor plays Isildur in the series?
A3: Harry Sinclair
Contributions
Towards robust training and evaluation of ConvQA models

● Taxonomy of question reformulations for ConvQA over KGs based on string-edit distance
● RL model with Deep Q-Network to select helpful reformulations guided towards better QA performance
● About 335k question reformulations of test cases in two ConvQA benchmarks
● REIGN framework with reusable components to judiciously augment benchmark training tailored to specific ConvQA models
The REIGN pipeline
Start with (Q, A) pair from benchmark

Conversation Question 2: TROP airing on? [Gold answer: Amazon Prime Video]
The REIGN pipeline

Reformulation taxonomy

Taxonomy of ConvQA Reformulation Categories

Level 0
- ALL OPERATIONS
  - INS (part)
  - DEL (part)
  - SUBS (part)
  - RETAIN (whole)

Level 1
- Level 0 subdivisions

Level 2
- Level 1 subdivisions
  - Completions
  - Ellipses
  - Paraphrases

Level 3
- Level 2 subdivisions
  - Entity coreferences
The REIGN pipeline
The core: Reformulation Category Selector

Taxonomy of ConvQA Reformulation Categories

Reformulation Category Selector (RCS) with reinforcement learning (Deep Q-Network)
The REIGN pipeline

The core: Reformulation Category Selector

Reformulation Category Selector (RCS) with reinforcement learning (Deep Q-Network)
The REIGN pipeline
Reformulation generator creates reformulations

Taxonomy of ConvQA Reformulation Categories

Reformulation Category Selector (RCS) with reinforcement learning (Deep Q-Network)
The REIGN pipeline
Pass reformulations through ConvQA model …

[Diagram showing the process of reformulation generation and selection]

System responses

Taxonomy of ConvQA Reformulation Categories

Reformulation Category Selector (RCS) with reinforcement learning (Deep Q-Network)
The REIGN pipeline
... to collect rewards ...

Taxonomy of ConvQA Reformulation Categories

Reformulation Category Selector (RCS) with reinforcement learning (Deep Q-Network)

System responses

Rank 1 Answer: Middle Earth
Rank 2 Answer: Hobbits
Rank 3 Answer: Amazon Prime Video
Rank 4 Answer: Netflix
Rank 5 Answer: New Zealand

Rewards (reciprocal rank, or metric proxies)
The REIGN pipeline

... to train the RCS

Taxonomy of ConvQA Reformulation Categories

Reformulation Category Selector (RCS) with reinforcement learning (Deep Q-Network)

Training of RCS with rewards based on ConvQA model performance or proxies
The REIGN pipeline

Repeat for all questions

---

Training of RCS with rewards based on ConvQA model performance or proxies
Components in REIGN

Two-step training

RCS training: Learning to select reformulation category

Agent

RCS

Sampled categories

RG

Generated reformulations

Rewards based on QA performance

ConvQA

orig
Components in REIGN

Two-step training

**RCS training:** Learning to select reformulation category

- **Agent**
  - RCS
- **Environment**
  - RG
  - ConvQA\textsubscript{orig}

- Sampled categories
- Rewards based on QA performance
- Generated reformulations

**ConvQA training:** Robust learning to answer questions

- Trained RCS
- RG
- ConvQA\textsubscript{robust}

- Predicted categories
- Generated reformulations paired with gold answers
- Augmented ConvQA training data
Large-scale Evaluation
Increasing robustness at inference time

- Small test sets not enough
- Reformulate questions with GPT-3.5-turbo
  - 10x with conversation history
  - 10x without conversation history
- 100k-200k questions in total
Experimental Setup
REIGN coupled with ConvQA models

• REIGN applied to two ConvQA models: CONQUER, EXPLAIGNN;
• REIGN applied on two benchmarks: ConvQuestions, ConvMix
• Results on original testsets and 20x larger GPT-augmented testsets (indicated with GPT-ConvMix / GPT-ConvQuestions)
Results

Improves performance of underlying ConvQA model

Models coupled with REIGN are able to answer more questions correctly
Results

Improves robustness to different surface forms

New metric Robust: average of #answerable reformulations per original test question (0-21)

Models coupled with REIGN are able to answer more reformulations per question intent correctly
REIGN: Wrap-up

Takeaways

- **Improved** ConvQA models by **training** with **reformulations**
- Reformulations **generated at scale** in **systematic way** by **reformulation taxonomy**
- More **robust** and **efficient training** by selecting **set** of most **helpful reformulations** for underlying model
- **Enlarged test set** generated with LLM for model **stress-testing**

reign.mpi-inf.mpg.de

Thank you!
Backup slides
# REIGN

## Detailed results: Main results, domain-wise, turn-wise

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Method ↓</td>
<td>P@1</td>
<td>MRR</td>
<td>Hit@5</td>
<td>P@1</td>
</tr>
<tr>
<td>Conquer [35]</td>
<td>0.218</td>
<td>0.272</td>
<td>0.337</td>
<td>0.173</td>
</tr>
<tr>
<td>Conquer [35] + Reign</td>
<td>0.245*</td>
<td>0.292*</td>
<td>0.346*</td>
<td>0.190*</td>
</tr>
<tr>
<td>ExplaIGNN [15]</td>
<td>0.370</td>
<td>0.438</td>
<td>0.526</td>
<td>0.278</td>
</tr>
<tr>
<td>ExplaIGNN [15] + Reign</td>
<td>0.384*</td>
<td>0.446*</td>
<td>0.531</td>
<td>0.295*</td>
</tr>
</tbody>
</table>

Table 5: Main results comparing REIGN-enhanced ConvQA models with their standalone versions. GPT-augmented test sets are 20x original sizes. REIGN is applied zero-shot on ConvQUESTIONS. The higher value per column per QA model is in bold.

<table>
<thead>
<tr>
<th>Method ↓ / Domain →</th>
<th>Books</th>
<th>Movies</th>
<th>Music</th>
<th>TV series</th>
<th>Soccer</th>
<th>Method ↓ / Turn →</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6-10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conquer [35]</td>
<td>0.227</td>
<td>0.175</td>
<td>0.159</td>
<td>0.141</td>
<td>0.163</td>
<td>Conquer [35]</td>
<td>0.205</td>
<td>0.193</td>
<td>0.177</td>
<td>0.184</td>
<td>0.160</td>
<td>0.133</td>
</tr>
<tr>
<td>Conquer [35] + Reign</td>
<td>0.239*</td>
<td>0.200*</td>
<td>0.167*</td>
<td>0.160*</td>
<td>0.184*</td>
<td>Conquer [35] + Reign</td>
<td>0.210*</td>
<td>0.214*</td>
<td>0.194*</td>
<td>0.204*</td>
<td>0.184*</td>
<td>0.147*</td>
</tr>
<tr>
<td>ExplaIGNN [15]</td>
<td>0.298</td>
<td>0.287*</td>
<td>0.265</td>
<td>0.274</td>
<td>0.265</td>
<td>ExplaIGNN [15]</td>
<td>0.333</td>
<td>0.297</td>
<td>0.286</td>
<td>0.292</td>
<td>0.277</td>
<td>0.205</td>
</tr>
<tr>
<td>ExplaIGNN [15] + Reign</td>
<td>0.333*</td>
<td>0.283</td>
<td>0.301*</td>
<td>0.281*</td>
<td>0.275*</td>
<td>ExplaIGNN [15] + Reign</td>
<td>0.350*</td>
<td>0.318*</td>
<td>0.311*</td>
<td>0.305*</td>
<td>0.291*</td>
<td>0.216*</td>
</tr>
</tbody>
</table>

Table 6: Domain-wise P@1 results on GPT-ConvMix testset.  
Table 7: Turn-wise P@1 results on GPT-ConvMix testset.
REIGN

Detailed results: Category predictions, design choices

Figure 4: Common category predictions by the RCS DQN.

Table 8: Large-scale effects of design choices in REIGN (with Conquer on GPT-ConvMIX, all differences systematic).
### REIGN

**Detailed results: GPT test sets, prompts**

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>Train</th>
<th>Dev</th>
<th>Test</th>
<th>GPT-Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>ConvMix [14]</td>
<td>8.4k (1680)</td>
<td>2.8k (560)</td>
<td>4.8k (760)</td>
<td>100.8k (760)</td>
</tr>
<tr>
<td>ConvQuestions [12]</td>
<td>33.6k (6720)</td>
<td>11.2k (2240)</td>
<td>11.2k (2240)</td>
<td>235.2k (2240)</td>
</tr>
</tbody>
</table>

Table 2: Benchmark sizes as #questions (#conversations). Reformulations are also counted as individual questions to be answered. Questions for the GPT-Test sets subsume the original test questions.

Reformulate the ‘Question’ 10 times in a short, informal way. Assume third person singular if not obvious from the question.

‘History’: {CONVERSATION HISTORY}

‘Question’: {QUESTION}

‘Reformulation’:

---

**[Books] History:** How many Pulitzer Prizes has John Updike won? 2.

**Question:** Which was the first book to win him the award?

**Ref 1:** What book earned John Updike his first Pulitzer Prize?

**Ref 2:** What was the author’s first book to win a Pulitzer?

**Ref 3:** Title of John Updike’s first Pulitzer Prize-winning book?


**Question:** What is the book based on?

**Ref 1:** What’s the book about?

**Ref 2:** What’s the book’s topic?

**Ref 3:** What’s the book’s subject?


**Question:** Was Kanye West a composer of the song?

**Ref 1:** Did Kanye West contribute to the lyrics of the song?

**Ref 2:** Did Kanye West perform the song with Beyonce?

**Ref 3:** Was Kanye West featured in the song?

**[TV series] History:** What is the release year of the TV series See? 2019.

**Question:** created by?

**Ref 1:** Who’s responsible for it?

**Ref 2:** Who’s the mastermind?

**Ref 3:** Who’s the author?

**[Soccer] History:** Pele scored how many goals in international play? 77. Has he scored the most goals? No.

**Question:** Did Messi beat his goal total?

**Ref 1:** Did Messi surpass Pele’s international goal record?

**Ref 2:** Has Messi scored more international goals than Pele?

**Ref 3:** Did Messi break Pele’s goal-scoring record?

Table 3: Examples of GPT reformulations for test sets.
REIGN

Detailed results: REIGN reformulations