Generating Natural Language Proofs with Verifier-Guided Search

Kaiyu Yang, Jia Deng, and Danqi Chen
Department of Computer Science, Princeton University

---

**Generating Natural Language Proofs**

- **Input:** a hypothesis $h$ and a set of supporting facts $C = \{s_{ent_1}, s_{ent_2}, ..., s_{ent_n}\}$ in natural language
- **Output:** a proof tree $P$ for deriving $h$ from a subset of $C$
  - The root node is $h$; the leaf nodes are sentences in $C$
  - Others are intermediate conclusions generated by the model

**Our Method: NLPoofs**

- **NLPoofs (Natural Language Proof Search)**
  - A new method for stepwise proof generation
  - Prover: Generate relevant steps conditioning on the hypothesis
  - Verifier: Mitigate hallucination by training an independent network to check the proof steps
  - Proof search: Use the prover/verifier to generate the final proof with the optimal validity score

**Prover**

- Fine tune a T5 model to predict the next proof step
- Generate multiple candidate steps via beam search

**Verifier**

- Input: A proof step (multiple premises, one conclusion)
- Output: A score in $[0, 1]$ calculated by finetuning RoBERTa
- Step scores are aggregated to calculate proof scores

**Proof Search**

1. Initialization: a proof generated by the prover alone
2. Iteration: expand the graph iteratively
   - Using steps proposed by the prover
   - Checked by the verifier
   - Average verifier/prover scores
3. Extraction: proof tree with the best score

---

**Experiments**

- **State-of-the-art on two benchmarks**
  - RuleTaker [Tajfard et al. 2021]: Simple, synthetic proofs
  - EntailmentBank [Dalvi et al. 2021]: Challenging, human-written proofs

- **Main Results on EntailmentBank**
  - Single-shot baselines: EntailmentWriter
  - Stepwise baselines: IRGR [Ribeiro et al. 2022], MetGen [Hong et al. 2022]

- **Ablations: Importance of Stepwise Verifier-Guided Search**

- **Large Language Models w/ In-Context Learning**
  - Single-shot proof generation with 7 in-context examples
  - GPT-3 and Codex cannot solve the task out of the box

---

**Challenges in Generating Valid and Relevant Steps**

- Existing stepwise methods struggle to generate proof steps that are both valid and relevant
  - **Condition on premises only:**
    - Many valid steps are irrelevant (not useful for proving the hypothesis)

  - **Condition on premises + hypothesis:**
    - Hallucinate invalid steps

---