IntrospectQA: Building Self-reflecting, Consistent Question Answering Models

Maya Srikanth, Rachael Wang, *Mentor: Eric Mitchell*
Department of Computer Science, Stanford University

Motivation

- Large pretrained language models excel on a variety of NLP tasks, but often suffer from a fundamental weakness: **logical inconsistency**
- **Example of logical inconsistency:**
  - **Is the apple a fruit? Yes.**
  - **Is fruit a plant? Yes.**
  - **Is apple a plant? No**

LOGICAL CONSISTENCY DEFINITION
If an LM assigns true to X and Y, and X ∧ Y => Z, then the LM should assign true to Z.

- **Can we use past predictions and NLI model output to build a self-reflecting, consistent pretrained QA model?**
- **Our approach:**
  - Augmenting a pre-trained QA model with an external memory for storing past model prediction
  - Integrate supervisory signals from a large pretrained NLI model to encourage consistency between past and future QA model predictions
  - **Evaluate model with adversarially sampled batches to increase conflict probability**

Background

- Existing approaches for enforcing logical consistency rely on constraint solving algorithms like MAXSAT, which operate on **hand-engineered constraint graphs** and are **confined to a finite set of entities and facts**
  - BeliefBank[1] adds a novel memory layer on top of pretrained T5 Macaw QA model to track model beliefs over time and modify raw PTLM answers to improve consistency

CONSTRAINT DEFINITION

A constraint comes in two forms:
1. positive implications: X = T (true)
   "X is a dog.T →"X has a tail.".T
2. mutual exclusivities: pair with X = F (false)
   "X is a dog".T →"X is a bird.".F
   "X is a bird".T →"X is a dog.".F
   This entity cannot be both a dog and a bird

A constraint is comprised of **condition → conclusion**

DATASET

We used the BeliefBank dataset of 85 entities, 4998 facts and 12,147 constraints. For this experiment, we use only questions that align with some constraint, resulting in subset of ~5500 questions.

Methods

RELATION → QUESTION + ANSWER PREPROCESSING

1. IsA: albatross, (IsA, bird: yes)→
   Q. Is an albatross a bird? A. Yes.
2. HasA: albatross, (HasA, feathers: yes) →
   Q. Does an albatross have feathers? A. Yes.
3. HasPart: albatross, (HasPart, face: yes) →
   Q. Does an albatross have a face? A. Yes
4. CapableOf: albatross, (CapableOf, fight for life: yes) →
   Q. Can an albatross fight for life? A. Yes.
5. HasProperty: albatross, (HasProperty, alive: yes) →
   Q. Is an albatross alive? A. Yes.

CONSISTENCY DEFINITION

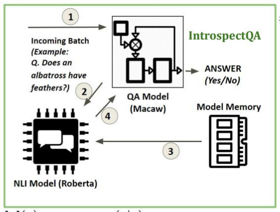
Define a constraint c_i as a 5-tuple of the form $(s_i.l_i \rightarrow s_j.l_j, w_i)$, where s_i, s_j are sentences $\in S$, $l_i, l_j \in True(T), False(F)$, and w_i denotes the weight of the constraint c_i.

$$\tau = \frac{|\{c_i|\neg(s_i.l_i \rightarrow s_j.l_j)\}|}{|c_i|s_i.l_i|}$$

$$consistency = 1 - \tau.$$

- Consistency (in words): the fraction of constraints whose **condition is believe, but whose conclusion is not**

ALGORITHM 1: CONFLICT BATCHING
conds←list of conditions corresponding to entity e
concl←list of conclusions corresponding to entity e
(1) Sample condition c_i from conds
(2) Sample 2 conclusions a_i from concl
such that $c_i \rightarrow a_i$ is specified by a constraint
Repeat (1), (2) until batch is built
Each run focuses on a single entity e



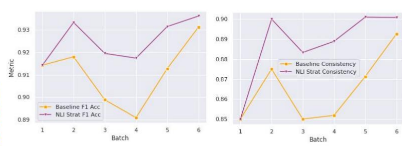
ALGORITHM 2: NLI Strategy 1
1. Compute latest batch of predictions with $\hat{y} = \mathbf{QA\_model}(x) = argmax_y p(y|x)$
2. Add latest batch $(x, \hat{y})$ to model_memory
   revised_predictions ← [ ]
   For each hypothesis, hyp_pred in model_memory:
      max_contr, max_entail ← [ ], [ ]
      For each premise, prem_pred in model_memory:
         hyp_NLI ← format (hypothesis, hyp_pred) into a declarative sentence
         prem_NLI ← format (premise, prem_pred) into a declarative sentence
         contr_logprob, neutral_logprob, entail_logprob ← append **NLI_model**(prem_NLI, hyp_NLI)
      max_contr ← contr_logprob ≥ -0.0015
      max_entail ← entail_logprob ≥ -0.0015
      If len(max_entail) < len(max_contr):
         y_{new} ← ¬ hyp_pred
      Else:
         y_{new} ← hyp_pred
      revised_predictions.append(y_{new})
   return revised_predictions

ALGORITHM 3: NLI Strategy 2
Same as NLI Strategy 1, except replace:
   If len(max_entail) < len(max_contr)
with
   sum(max_entail) < sum(max_contr)

Experiments & Analysis

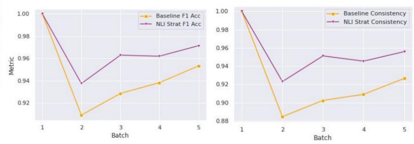
We evaluate whether our method can outperform a Macaw QA baseline on BeliefBank[1] yes/no questions across a variety of configurations (# entities, # batches, batch size), in a streaming setting.

#1. **6 entities (at least 100 questions each), 6 batches of size 20**



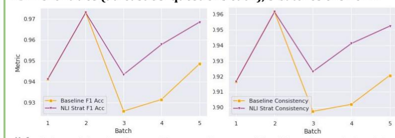
+1.6% avg F1
+2.6% avg consistency

#2. **11 entities (at least 70 questions each), 5 batches of size 13**



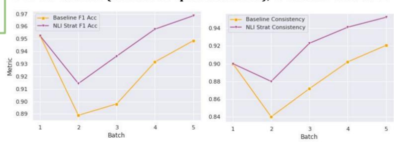
+2.2% avg F1
+3.3% avg consistency

#3. **25 entities (at least 65 questions each), 5 batches of size 12**



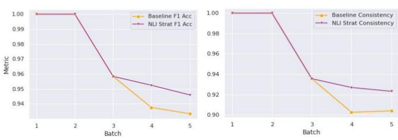
+1.3% avg F1
+2.1% avg consistency

#4. **35 entities (at least 60 questions each), 5 batches of size 12**



+2.4% avg F1
+3.7% avg consistency

#5. **74 entities (at least 50 questions each), 5 batches of size 10**



+0.6% avg F1
+1% avg consistency

- IntrospectQA can boost performance (F1, consistency) in a streaming setting across a variety of BeliefBank topics **without any fine-tuning or constraint data**
  - Other considerations: sensitivity to threshold, performance on specific entities may vary
- **Our baselines & NLI strategy are not directly comparable to related work**[1]: **(i)** our baseline is significantly better; **(ii)** differences in eval data (we focus only on constraints); **(iii)** differences in batch sampling (we use conflict batching)
  - **Noisy comparison:** IntrospectQA outperforms BeliefBank F1 accuracy (their best-performing model achieves F1 86.6)

Conclusion

- IntrospectQA performance suggests that augmenting a QA model with NLI model + model memory can improve logical consistency and F1 accuracy in a **streaming setting** on **any topic**
- IntropectQA is **a promising substitute** for MaxSAT solver in scenarios where a constraint graph is not available for topics of interest
- Future steps: improve strategies, extend method to more complex NLP tasks (QA, dialogue, etc)

References

[1] Nora Kassner, Oyvind Tafjord, Hinrich Schütze, and Peter Clark. Beliefbank: Adding memory to a pre-trained language model for a systematic notion of belief, 2021.
[2] Tao Li, Vivek Gupta, Maitrey Mehta, and Vivek Srikumar. A logic-driven framework for consistency of neural models. CoRR , abs/1909.00126, 2019.
[3] Oyvind Tafjord and Peter Clark. General-purpose question-answering with macaw, 2021