Improving Logical Consistency in Pre-Trained Language Models using Natural Language Inference

Ananth Agarwal   Cameron Tew   Anthony Tzen

Problem

Current state-of-the-art **pre-trained language models (PTLMs)** contain rich and vast amounts of world knowledge, demonstrating an ability to extrapolate information from contextual texts and to accurately answer questions. However, the latent factual understanding captured by PTLMs can be irrational and incohesive, causing PTLMs to be prone to generating logically inconsistent statements.

Macaw, a PTLM built on T5, outputs the following inconsistent result:

- Q: Is a puppy a vertebrate? A: Yes
- Q: Is a vertebrate a crustacean? A: No
- Q: Is a puppy a crustacean? A: Yes

We aim to improve accuracy and logical consistency of PTLMs using **natural language inference (NLI)** and a heuristic function to revise contradictory PTLM answers within a batch of input questions.

Dataset

We are using the BeliefBank dataset curated by Kassner et al. to tune and evaluate our model. The dataset contains the following:

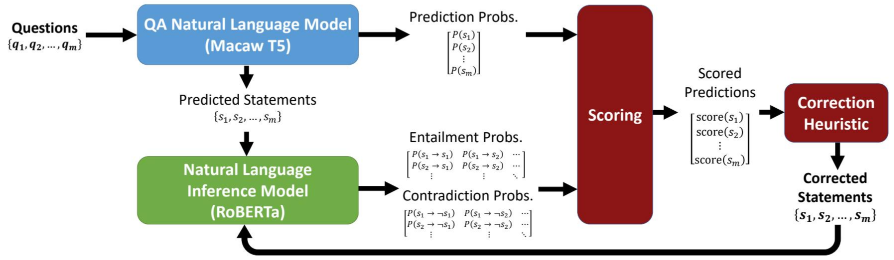
- **Constraint graph:** Directed graph derived from the ConceptNet semantic knowledge graph. Nodes are modeled as statements of the form `(<relation>,<target>:<truth>)`, and edges capture **directional implications** between nodes.
- **Test silver facts:** 12,636 facts harvested from the constraint graph consisting of 85 animal and plant entities (e.g., "puppy", "daisy"). Silver facts can be represented as `(<entity>,<relation>,<target>,<truth>)`.
- **Development silver facts:** Facts used to tune model hyperparameters

We sample facts for each entity to create one batch of facts per entity. Dev batch size is 50, and test batch size is 100.

Metrics

$$\mathrm{F1} = \frac{\mathrm{TP}}{\mathrm{TP} + 0.5(\mathrm{FP} + \mathrm{FN})} \qquad \mathrm{Consistency} = \frac{1 - |\{c_i \mid \neg(s_p \rightarrow s_h)\}|}{|\{c_i \mid s_p\}|}$$

The denominator of consistency is the number of constraints with a true premise s_p contained in the batch. The numerator is the number of these constraints that are violated (where $s_p \rightarrow s_h$ is false). Thus, consistency is defined as the complement of the fraction of all violated constraints.



Methodology

- For each question q_i, a PTLM predicts an answer a_i with an associated **prediction probability**, $P_{PTLM}(a_i)$
- Each question+answer (q_i, a_i) is translated into a statement s_i. The prediction prob. of s_i would be the same: $P_{PTLM}(s_i) = P_{PTLM}(a_i)$
- For each ordered pair of statements (s_h, s_p) where s_h is the hypothesis and s_p is the premise, a NLI model returns an **entailment probability** and a **contradiction probability**. We thus have various ways to estimate how the probability of s_h relates to the probability of s_p:

Single Constraint Scenario (only one s_p):

$$P_{NLI}(s_h) = P\left(s_p \wedge (s_p \rightarrow s_h)\right) = P_{PTLM}(s_p)P(s_p \rightarrow s_h) = P_{PTLM}(s_p)P_e(s_h, s_p)$$

$$P_{NLI}(\neg s_h) = P\left(s_p \wedge (s_p \rightarrow \neg s_h)\right) = P_{PTLM}(s_p)P\left(\neg(s_p \wedge s_h)\right) = P_{PTLM}(s_p)P_c(s_h, s_p)$$

Maximum

$$P_{NLI}(s_h) \coloneqq \max_{p \neq h}\left(P_{PTLM}(s_p)P_e(s_h, s_p)\right)$$

$$P_{NLI}(\neg s_h) \coloneqq \max_{p \neq h}\left(P_{PTLM}(s_p)P_c(s_h, s_p)\right)$$

Weighted Average

$$P_{NLI}(s_h) \coloneqq \frac{1}{\sum_{p \neq h} P_{PTLM}(s_p)} \sum_{p \neq h} P_{PTLM}(s_p)P_e(s_h, s_p)$$

$$P_{NLI}(\neg s_h) \coloneqq \frac{1}{\sum_{p \neq h} P_{PTLM}(s_p)} \sum_{p \neq h} P_{PTLM}(s_p)P_c(s_h, s_p)$$

- To compute a final confidence score for s_h, we balance the NLI estimates and the PTLM estimate of $P(s_h)$:

$$\mathbf{score}(s_h) \coloneqq \lambda\left(0.5 \cdot P_{NLI}(s_h) + 0.5 \cdot \left(1 - P_{NLI}(\neg s_h)\right)\right) + (1 - \lambda)P_{PTLM}(s_h)$$

- To correct the original PTLM predictions, the statement with the lowest score is inverted ("flipped") if it is under a minimum score. This is iteratively repeated, with the scores being updated after each flip.

Results and Analysis

	Hyperparameters			Metrics	
Method	**Min. Score**	**Max Flips**	λ	**F1**	**C**
Baseline	-	-	-	0.787	0.826
Max	0.573	9	0.422	0.807	0.836
Average	0.543	6	0.519	0.812	0.846
Weighted Avg.	0.367	7	0.832	0.833	0.858

Table 1. Approach vs. Baseline Performance

- Baseline scores are taken from the PTLM's raw output
- Score increases after flipping incorrect statements
- The Weighted Average produces higher variance during scoring, which may allow for easier identification of statements to flip
- High λ for Weighted Average indicates NLI score is a good signal
- Lower λ for Max suggests the max is noisy, so the model learns to weigh PTLM prediction probability higher
- Q: Is a puppy a crustacean? PTLM: Yes Our model: No

Conclusion

Combining NLI output and the PTLM's confidence in its original predictions through a heuristic function to identify and revise contradictory statements **improves both F1 score and logical consistency** without needing hand-written constraints.