First-Order Logic-Based Dataset for Question-Answering and Classification Evaluation

Rachel Blumenthal[1] and Dragomir Radev PhD[1]

[1]Department of Computer Science, Yale University, New Haven, CT

LILY Lab

Introduction

Following the train of human thought in a piece of text is one of the most complicated tasks NLP models can be trained on. NLP machines' ability to understand, follow, and even generate text with propositional logic has been thoroughly studied and developed, but propositional logic is still far simpler than the logic used in most written pieces. The next step in the reasoning development of NLP machines is the development of their ability to follow first-order logic. This field is a "new and exciting direction for neural NLP … yet to be well addressed" but known to be "important for building interpretable and knowledge-driven neural NLP models to handle complex tasks" (Zhou, Duan, Liu, & Shum, 2020).

First-order logic differs from propositional logic in its ability to understand references and pro-form words. Propositional logic only allows for very concrete statements and references by name, while first-order logic allows for references to objects or subjects and for discussing groups of objects without individually specifying each object in the group (First-Order Logic). The limitations in place when using only propositional logic limit the ability of NLP machines to follow and understand human-written text and greatly impacts the natural flow of computer-generated statements.

Dataset

The final dataset consists of 254 short "stories" or informational blurbs with an even distribution of aggregation, subsumption, and unification problems. In order to create flexibility for researchers and make the dataset useful for different kinds of NLP models, each entry in the dataset includes both a classification problem and a generation problem.

The classification questions are simple statements about the contents of the dataset story and a classification of True or False. The dataset contains an even distribution of true and false statements both overall and within each task category.

The generation questions are simple questions that require the NLP machine to know the correct answer and actually come up with the word, number, or name by itself. This is a different kind of processing than the previously discussed classification problems.

In order to allow researchers to test the robustness of their machines, the dataset also includes several lightly perturbed short stories. The perturbed entries similar to the kinds of mistakes common in human typed text. These mistakes are so small that most human readers wouldn't even notice them, but even if they were noticed, the changes wouldn't impact a person's understanding of the story.

Test Methodology

In order to demonstrate the usefulness of the finished dataset, I ran the dataset through three NLP models. Two classification models from the Allen Institute for Artificial Intelligence, ProofWriter (Tafjord, Mishra, & Clark, 2020) and its predecessor, RuleTaker (Clark, Tafjord, & Richardson, 2020), were tested. I also tested the generative questions and answers on the OpenAI GPT-3 Question Answering Beta (Brown, et al., 2020).

I analyzed the responses of the two classification NLP models by their Accuracy, Precision, Recall, and F1 score from their confusion matrices. The RuleTaker model provides additional quantitative data in the form of a confidence score for each classification it gives. In addition to the previously listed metrics, I analyzed the average confidence of the RuleTaker response to different types of problems. Unlike the RuleTaker, ProofWriter can classify statements as UNKNOWN if it is not confident in the classification, so I analyzed the responses of ProofWriter both ignoring the UNK responses and counting the UNK tokens towards the False tally, essentially making it a True or Not True classification. Finally, the GPT-3 generative responses were evaluated only for accuracy, as the previously described metrics are all reliant on classification relationships.

ProofWriter Performance (Combining False and Unknown)

	Accuracy	Precision	Recall	F1
Overall	45.24%	46.93%	66.14%	54.90%
Aggregation	30.95%	36.84%	48.83%	42.00%
Subsumption	54.76%	52.31%	82.93%	64.15%
Unification	50.60%	50.88%	67.44%	58.00%

RuleTaker Performance

	Accuracy	Precision	Recall	F1
Overall	43.65%	40.50%	25.19%	31.07%
Aggregation	33.33%	15.79%	6.98%	9.68%
Subsumption	50.00%	48.57%	41.46%	44.74%
Unification	48.19%	48.00%	27.91%	35.29%

GPT-3 Performance

	Overall Dataset	Normal	Perturbed
Overall Accuracy	37.69%	36.32%	43.14%
Aggregation Accuracy	14.29%	16.42%	5.88%
Subsumption Accuracy	51.19%	49.25%	58.82%
Unification Accuracy	47.62%	43.28%	64.71%

RuleTaker Average Confidence on Correct Classifications

	Overall	Normal	Perturbed
Overall	93.25%	93.93%	90.55%
Aggregation	95.43%	96.86%	90.17%
Subsumption	90.09%	91.06%	86.00%
Unification	95.05%	94.97%	95.38%

Standard	Task	Story	Classification Question	Classification	Generation Question	Answer	Answer (Num.)
Normal	Aggregation	Crofton reads five books. The next day he reads two books.	Crofton reads eight books.	FALSE	How many books did Crofton read?	Seven	7
Normal	Subsumption	Anton draws two circles, two triangles, a square, and an oval.	Anton draws six shapes.	TRUE	How many shapes does Anton draw?	Six	6
Perturbed	Aggregation	Clarissa steals nine stop signs. The police recover one of of the stop signs.	Clarissa has nine stop signs.	FALSE	How many stop signs does Clarissa have?	Eight	8

Results

The GPT-3 model's accuracy was very dependent on the task of the story-question pair. It was 51.19% accurate and 47.62% accurate on subsumption and unification tasks, respectively, but struggled on aggregation. The GPT-3 was only 14.29% accurate on aggregation tasks, bringing its overall accuracy on the dataset down to 37.7%. These same trends continue when the dataset is split into Normal and Perturbed stories – subsumption and unification are similarly successful and the model far underperforms at aggregation tasks.

The classification models RuleTaker and ProofWriter displayed similar trends to the GPT-3, as both struggled with aggregation but performed fairly similarly on subsumption and unification problems. Additionally, the distinction between Normal and Perturbed stories did not change the trends or dramatically impact the models' performance. The following tables show the accuracy, Precision, Recall, and F1 scores for the entire dataset, broken down by task.

Despite RuleTaker's better performance in subsumption and unification than in aggregation, the confidence scores indicate that, on average, the model was the least confident in its correct answers on subsumption task questions. Additionally, this was the metric where the difference between Normal and Perturbed stories is consistent – RuleTaker was consistently less confident on the correct answers it gave to perturbed questions. The following tables highlight the main results.

Conclusion

The RuleTaker, ProofWriter, and GPT-3 models' performance on this dataset, though analyzed in depth in this project, overall demonstrated that the task of first-order logic is still extremely difficult for these advanced models. Even in their best categories, the highest accuracy in any category was 54.76% for ProofWriter answering subsumption problems, and that number does not reflect the reality that the model classified many of the statements as Unknown. Though these are three of the most popular NLP logic models, this research demonstrates a need for further training and development on first-order logic. While the dataset created is only around 250 entries, it can serve as a tool to highlight where more work needs to be done and what kinds of questions and problems should be put into larger datasets when they are created.

Acknowledgement

I would like to thank Professor Dragomir Radev and Simeng Han for all of their help and guidance on this project.