Investigating Language Models' Temporal Reasoning Through Adversarial Training

Akhil Iyer
Department of Computer Science
The University of Texas at Austin
akhil.iyer@utexas.edu

Abstract

When humans read passages for QA, they learn temporal relations between events by reasoning the chronology of the entire passage. This reasoning ability is not well-performed by state-of-the-art models such as ELECTRA-small. To address models' poor temporal reasoning, I propose a fix to the model training by introducing numbers of adversarial examples into the SQuAD dataset in order to help the model filter out events irrelevant to the chronology of all of the events and gain a better understanding of the chronology of the entire passage as a result. The proposed training data is a combination of portions of examples from SQuAD (Rajpurkar et al., 2016) and examples from adversarial_qa (Bartolo et al., 2020). Different versions of the training data with different ratios between squad and adversarial examples were tested using checklist sets (Ribeiro et al., 2020) and I found that a dataset consisting . Quantitative analysis of the results reveals a significant improvement in understanding when one event happens before or after another by 35% in exact-match score.

1 Introduction

For several years, pre-trained models have had impressive performance on par with human performance in question-answering tasks for datasets such as SQuAD (Rajpurkar et al., 2016) that focus on synthesis of the relationships among events within a single passage. This synthesis requires a deep understanding of the temporal relationships among events because temporal information in real-world contexts is most often given implicitly, and temporal reasoning is often integrated with other types of reasoning including causal reasoning and spatial reasoning. As a result, instead of gaining proper temporal reasoning abilities, models learn spurious correlations that make them perform well with question-answering datasets but struggle in other contexts. A common approach to help models generalize in different contexts is to introduce adversarial examples into the training set. The AdversarialQA dataset (Bartolo et al., 2020), for example, is dataset in which human annotators develop questions for passages and use model-in-the-loop feedback from BERT, BiDAF, and RoBERTa to fine-tune their questions to intentionally stump these models, leading to stronger generalization for non-adversarial datasets. Additionally, there exists work in creating checklist datasets (Ribeiro et al., 2020) that allow the creation of large datasets that provide better insights into the fine-grained errors produced by language models, including errors related to temporal reasoning. Because of this existing work, I specifically investigated the effect of introducing adversarial examples into SQuAD training data on language models' abilities to answer questions related to temporal reasoning. I created training and validation datasets of combinations between the SQuAD and AdversarialQA datasets while varying the split ratio of number of SQuAD examples to the number of adversarial examples. I also built a small checklist dataset composed of QA examples that explicitly test for temporal reasoning in order to evaluate the models trained on each dataset. Overall, when the model is trained on more adversarial examples, the model performs better in correctly answering questions in the checklist dataset, suggesting that models trained on more adversarial data improve in temporal reasoning ability.

2 Experiments

2.1 Experimental Setup

2.1.1 Training and Validation Datasets

I use a combination of SQuAD and AdversarialQA for my datasets. Each training dataset is the same size as the original SQuAD dataset (about 87600 examples) with varying numbers of examples coming from the AdversarialQA. Adversarial examples were randomly selected from the AdversarialQA training set, and I used these examples to create a dataset with 10% adversarial examples, a dataset with 25% adversarial examples, and a dataset 50% examples. In the rest of the paper, these datasets will be denoted as D_{10}, D_{25}, and D_{50}. Additionally, D_0 will denote the original SQuAD training dataset. These four training datasets were used to train the ELECTRA-small language model over 3 epochs, processing about 32500 batches of training data. To evaluate the overall accuracy of each dataset, I evaluate each version of the model using three validation sets: V_0, V_{50}, and V_{100}. The SQuAD validation set is V_0, the AdversarialQA validation set is V_{100}, and V_{50} is a combination of both validation sets in which 50% of the examples are randomly selected from the AdversarialQA validation set. The exact match score and F1 word-overlap score are computed for each combination of training set and validation set.

2.1.2 Checklist Set

I generate a single checklist set by generating 90 synthetic examples of context-question pairs that require some type of temporal reasoning using GPT-4. I designed the checklist set with four categories in mind: change in profession, before-and-after relations, synchronous event relations, time point identifications, and relative time expressions. "Change in profession" examples are events in which two people have a job followed by one person changing their job along with the question asking which one has the new job. For "before-and-after" event relations, the context provides two events, and the question asks which event came either or last. The "time point identification" examples assess the models' ability to answer the time an event occurred based on dates provided in the context. Examples with synchronous events assess the models' ability to understand what events are occurring simultaneously. Lastly, examples for relative time expressions assess how models perform when a specific amount of time passes between two events such in terms of weeks or months. I evaluate all of the models on each fine-grained error type individually for a more fine-grained assessment between the models in reasoning ability, and I compute the exact-match and F1 score for all of the models on the combined checklist set for an overall performance comparison.

Exact Match Scores	V_{100}	V_{50}	V_0
D_{100}	78.39	61.57	18.70
D_{90}	77.60	62.29	23.4
D_{75}	77.52	63.16	26.17
D_{50}	76.87	62.60	26.70

Table 1: Exact match scores for all combinations of training and validation datasets

F1-Scores	V_{100}	V_{50}	V_0
D_{100}	86.20	69.99	29.04
D_{90}	85.63	70.77	33.51
D_{75}	85.39	71.53	36.34
D_{50}	84.85	71.21	36.96

Table 2: F1 scores for all combinations of training and validation sets

3 Analysis

3.1 Overall Performance

The exact-match scores and F1-scores of every combination of training set and validation set are given in Table 1 and Table 2. Overall, there is no significant improvement in the accuracy of the ELECTRA-small model for any of the validation sets when trained on more adversarial examples. The only noticeable trend in the results is that the model improved by 8% in exact match score when evaluated on V_0 when trained on more adversarial examples. Despite no significant improvement, there is still no performance drop when evaluated on V_{100} as a result of introducing adversarial examples, meaning that the model was not overfitted on the adversarial examples. This, in combination with the improvement for V_0, suggests that training with adversarial examples improves the ELECTRA-small model's generalizability.

3.2 Checklist Performance

The exact-match scores and F1-scores for the evaluation of each model on the full checklist suite

EM and F1 Scores for Checklist Set	D_{100}	D_{90}	D_{75}	D_{50}
Exact Match Scores	42	49	55	55
F1-Scores	57.49	62.36	68.91	67.99

Table 3: Exact-Match and F1 Word Overlap Scores for Evaluating the Models on Checklist Sets

Number of Correct Predictions Per Fine-Grained Checklist Task	D_{100}	D_{90}	D_{75}	D_{50}
Change in Profession	20	20	20	20
Before and After	9	17	23	21
Relative Time Expressions	0	0	0	0
Time Point Identification	10	10	9	10
Synchronous Events	3	2	3	4

Table 4: The number of correct predictions per fine-grained temporal category. There are 20 change in profession examples, 40 before and after examples, 10 relative time expression examples, 10 time point identification examples, and 10 synchronous event examples.

are displayed in Table 3. Overall, when there are more adversarial examples, there is a significant improvement of about 15% in the model's ability to answer questions related to temporal reasoning. Every model scored perfectly on the change in profession task, and for the relative time expression task, every models' predictions did not match any of the example answers. Additionally, it is worth noting that there are no tasks in the checklists in which the model trained on D_{100} outperformed any of the models trained on adversarial examples.

3.2.1 Change in Profession

For this task, all of the models scored perfectly in both exact-match score and F1-score. This is likely due to the same templated format being used for the context for each example. Two examples from this task are shown as follows: "Both Alice and Bob were teachers, but there was a change in Alice, who is now a lawyer," and "Both Sarah and John were doctors, but there was a change in John, who is now a chef." Because of this format, the answer is always in the same absolute position in the sentence. Thus, it is possible that the model can perfectly answer these questions due to seeing similar relative positional encodings frequently in the SQuAD training data.

3.2.2 Before and After Relations

This task is where there is a huge improvement in performance between the model trained on D_{75} and the model trained on D_{100} from initially getting only 9 examples correct to getting 23 examples correct. The following example is an example for which the D_{100} model got wrong but the D_{50} right is shown as follows: "Michael became a chef before Lisa did." When asked "who became a chef first," the D_{100} model answered "Michael became a chef before Lisa," whereas the D_{75} answered "Michael." Because the D_{100} extracted the entire phrase, the incorrect literal text extraction suggests that the model struggled in entity recognition. Temporal recognition requires not only an understanding of the events in a passage but also the entities, so poor entity recognition implicitly leads to poor temporal reasoning as evident in this example. Adversarial examples are generally more complex than SQuAD examples, so the model may have better learned entity recognition due to seeing more variation in its training data. However, adding more adversarial examples does not explicitly cause an improvement in temporal reasoning. For D_{50}, the model actually predicted two fewer examples correctly than the D_{75} model possibly due to overfitting on adversarial examples.

3.2.3 Relative Time Expressions

There was no improvement at all among the models in either exact-match score or F1-score, which is likely due to the limitations of the ELECTRA-small itself rather than the training datasets. The ELECTRA-small model is designed for extractive question-answering, whereas this task requires answering questions with information that is not explicitly mentioned in the context. For instance, this is an example for which all of the models answered incorrectly: "John started working at the company in 2010. Three years later, he was

Number of Correct Predictions Per Fine-Grained Checklist Task	D_{100}	D_{90}	D_{75}	D_{50}
Change in Profession	100.00	100.00	100.00	100.00
Before and After	29.88	43.33	57.5	53.33
Relative Time Expressions	16.11	11.67	19.19	16.33
Time Point Identification	100.00	100.00	96.00	100.00
Synchronous Events	81.81	76.24	75.00	82.29

Table 5: The F1 word-overlap scores per fine-grained temporal category for each dataset.

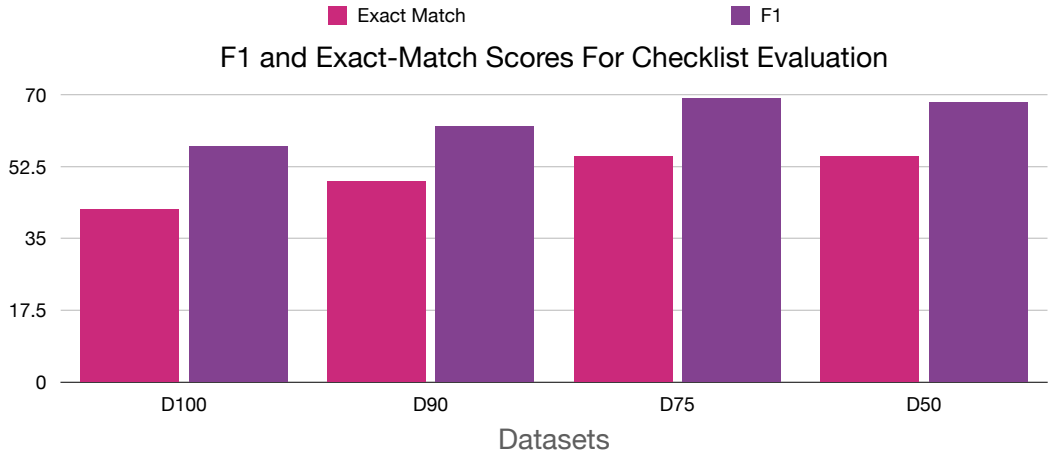

Figure 1: The exact-match scores and F1 word-overlap scores after evaluating all of the trained models on the entire checklist set.

promoted to manager." When prompted with the question "when was John promoted to manager," the model answered "three years later" instead of "2013." The model is only capable of answering "three years later" since it is an extractive model, so it inherently struggles for questions that require a real-world understanding of time.

3.2.4 Time Point Identification

All of the models got a perfect score for this task. In all of the examples for this task, the model simply has to find the date in the context. In an example, for the context "the first iPhone was released by Apple on June 29, 2007, revolutionizing the smartphone industry," the model can easily recognize dates since they frequently occur in its training data. Therefore, this task is already trivial for the base model.

3.2.5 Synchronous Events

The synchronous events task resulted in models achieving high F1 word-overlap scores but low exact-match scoring, suggesting that the example labels may be in a format inconsistent with the way that ELECTRA-small extracts its answers. Upon manual review, I found it common that the model would predict the correct answer in a format that does not exactly match the gold standard. When asked "what happened when the concert was supposed to start" for the context "the thunderstorm hit the town at the same time the concert was supposed to start," the model answered "the thunderstorm" while the gold standard answered "thunderstorm hit." The F1-scores show that the D_{75} and D_{90} models performed poorly in comparison to the D_{100} and D_{50} models. The baseline D_{100} may have already been fitted well to understand words that indicate synchronous events, so it would not have been as necessary to augment adversarial examples. Additionally, the examples randomly selected for D_{75} and D_{90} may not have been relevant to understanding when events occur simultaneously. Since D_{50} has significantly more adversarial data, the dataset possi-

Figure 2: The exact-match scores and F1 word-overlap scores after evaluating all of the trained models on the entire checklist set.

bly includes more relevant examples that allow the model to learn synchronous events better.

4 Limitations

Although multiple proportions of adversarial examples were used to investigate any possible improvements in temporal reasoning, the reasoning capabilities are significantly hindered by the abilities of the model. The ELECTRA-small model is a masked language model that performs text extraction, so it inherently struggles with questions that require a real-world understanding of time and expressions related to time. Additionally, the adversarial examples augmented to the SQuAD dataset were not based on feedback from the ELECTRA-small model but only on BERT, BiDAF, and RoBERTa. So, it may be the case that the adversarial questions in AdversarialQA may not lead the ELECTRA-small model to struggle as much as the other three models would. Furthermore, the checklist sets assessed temporal reasoning only with rudimentary context. In the real world, events have much more complex temporal relationships with each other, so the minor improvements found in this paper are not generalizable to real-world contexts.

5 Conclusion

In this paper, I investigated the augmentation of adversarial training data into the SQuAD dataset to assess whether adversarial training can lead to better performance in temporal reasoning tasks. I created three new datasets with varying levels of adversarial examples retrieved from AdversarialQA and assessed its overall temporal reasoning abilities as well as its performance in five fine-grained temporal reasoning QA tasks using a checklist set. My experiment results show improvements in overall temporal reasoning ability, specifically for helping models understanding when one event happens before or after another.

References

Max Bartolo, Alastair Roberts, Johannes Welbl, Sebastian Riedel, and Pontus Stenetorp. 2020. Beat the ai: Investigating adversarial human annotation for reading comprehension. *Transactions of the Association for Computational Linguistics*, 8:662–678.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. *arXiv e-prints*, page arXiv:1606.05250.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of nlp models with checklist. In *Association for Computational Linguistics (ACL)*.