Sentence Level Learning For Unanswerable Question Predictions

Yuling Chen, BK (Baokui) Yang, Daniel Ce, Sid J Reddy
UC Berkeley School of Information
{yulingchen54, baokui}@berkeley.edu, daniel.cer@gmail.com, sid@conversica.com

1. Abstract

Unanswerable questions create big challenges to Machine Reading Comprehension (MRC) models and systems. After SQuAD 2.0 [1] [7] introduced human-written adversarial unanswerable (NoAns) questions, the overall performance of any MRC model has been largely impacted by its accuracy on NoAns questions. Accurately predicting unanswerable questions becomes critical to overall Q&A model performance.

In this paper, we analyze and compare the unanswerable questions with answerable questions in SQuAD 2.0. We discover that most unanswerable questions involve more complex linguistic phenomena. At the same time, most of the answerable questions can be answered on sentence level, and most of the unanswerable questions do not need cross sentence context for the prediction. Moreover, we also observe that sentence level trained models are more capable of learning and identifying the linguistic elements of the language. Based on these observations, we create a novel approach that can be adapted to any MRC models and systems. We break down paragraphs into sentences to train and predict the questions, trim the short sentences in the training set to achieve a balanced unanswerable question ratio, and aggregate the sentence level results into paragraph level as the final output. Using the BERT[2] base model as the MRC reader baseline, with more than 24% F1 improvement on unanswerable question prediction, our model pushes the overall performance of EM score from 76.3 to 90.4(14% improvement), and F1 score from 79.4 to 94.2(15% improvement).

2. Introduction

Q&A with NLP is an active research field in recent years. SQuAD[1] is a popular public data set for Q&A model training and evaluation. In the past couple of years, BERT[2] has gained popularity in Q&A systems. Despite the tremendous improvements brought by BERT, Q&A tasks remain a challenging task for computers. To be applicable to real-world problems, the model needs to not only answer the answerable questions (a.k.a HasAns questions) correctly but also detect the unanswerable ones (a.k.a NoAns questions).

After SQuAD 2.0 introduced unanswerable questions[7], the performance of existing models have been largely impacted. A strong neural system that gets 86% F1 on SQuAD 1.1 achieves only 66% F1 on SQuAD 2.0. BERT model also shows 10.7% lower F1 performance on unanswerable questions compared with its performance on answerable questions. The low performance on unanswerable questions dragged down the overall model performance to a large extent.

Why are unanswerable questions creating such a challenge for MRC systems? When the unanswerable questions were created by crowdworkers based on the same paragraphs in SQuAD 1.1 data set, a wide range of linguistic language elements were used intentionally or unconsciously to make the questions unanswerable. This includes negation, antonymy, word sequence change, and entity exchange between questions and paragraphs. Table 1 shows an unanswerable question example of entity exchange (people vs. households) between the question and the passage. Before unanswerable questions are introduced, assuming there always exists an answer to the question, the models only need to choose a most plausible text span based on the question. Now systems and models must learn to identify a wide range of those linguistic phenomena. This makes the Q&A tasks much harder than before.

Question: *What year was 366,273 people in Jacksonville?*
Paragraph: *"... As of 2010[update], there were 366,273 households out of which 11.8% were vacant. ..."*

Table 1: Entity Exchange Example of Unanswerable Questions in SQuAD 2.0

When we compare unanswerable questions with answerable questions in SQuAD 2.0, one interesting pattern we observe is that most of the answerable questions can be answered by one single sentence. At the same time, for unanswerable question prediction, in most cases, sentence level data can decide whether this question is answerable or not. We do not need cross sentence information to make the prediction.

Motivated by this observation, we create an approach of breaking down paragraphs into sentences, train and predict each answer on sentence level, and then aggregate the prediction result back to paragraph level. This approach can be adapted to any MRC models and systems. Using the BERT base model as the baseline MRC reader on SQuAD 2.0 data set, we achieve significantly better performance on unanswerable questions as well as the overall model performance compared with that of the BERT base model.

The major contributions that we make to Q&A and the related research areas are follows:

- Our research and analysis results on why the unanswerable question is harder to predict, and most unanswerable questions in SQuAD2.0 can be predicted on sentence level.
- The methodology together with the reference implementation based on the BERT base model to break down paragraphs into sentences for training and prediction, and aggregate back to paragraph level as the final output.
- 24% F1 improvement on unanswerable question prediction, and 15% F1 overall model performance gain compared with BERT base model as our baseline MRC reader.

3. Related Work

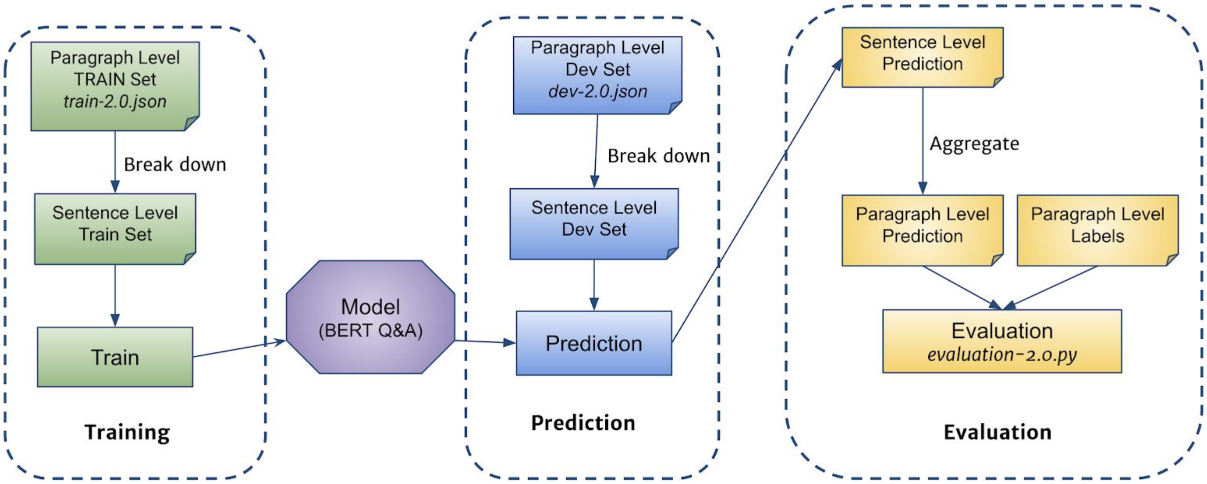
Since SQuAD 2.0 was released, many MRC models have been adapted to work on this new data set and dedicated efforts to unanswerable questions prediction[3][4][5][6][8][9][10]. (Huang et al.

2019)[4] used a relational module that is adapted to existing MRC models to extract semantic and context objects from both questions and passages. Then a relation network is used to generate relationship scores that are utilized to determine whether a question is unanswerable. (Hu et al., 2019)[3] added a separately trained answer verifier for no-answer detection with their MnemonicReader. The answer result proposed by the reader together with the question are passed to verifiers to check if the answer is legitimate. (Sunet al., 2018)[6] proposed the U-net with a universal node that encodes the fused information from both the question and passage, which helps to predict if the question is answerable.

All these efforts obtained similar or slightly better performance than that of the BERT base model. In this paper, we present our approach of sentence level training and prediction that achieves significant performance improvement on unanswerable question prediction compared with the BERT base model.

4. Methods

We introduce a sentence level model for the reading comprehension task. Firstly, we break down the training set from paragraphs into sentences, and then train the model. Secondly, we do a similar sentence breakdown on the dev set, and run the prediction for each sentence. Lastly, to get the prediction for each paragraph, we aggregate the prediction results of all its sentences.



Paragraph Level TRAIN Set *train-2.0.json*
Break down
Sentence Level Train Set
Train
Training
Model (BERT Q&A)
Paragraph Level Dev Set *dev-2.0.json*
Break down
Sentence Level Dev Set
Prediction
Prediction
Sentence Level Prediction
Aggregate
Paragraph Level Prediction
Paragraph Level Labels
Evaluation *evaluation-2.0.py*
Evaluation

Figure 1: Sentence Model Flow

4.1 Breaking Paragraph Into Sentences

The following is an example of how we break down the paragraph into sentences. Basically we split the paragraph into individual sentences. Then only for the sentence containing the answer, we label it with the correct answer; for all the other sentences, we label them with no answer.

```
question: When did Beyonce start becoming popular?
```

Paragraph:	**Sentences:**
context: Beyoncé Giselle Knowles-Carter ... is an American singer... and actress. Born and raised in Houston... and rose to fame in **the late 1990s** as lead singer ... Child. Managed by her father... all time. Their hiatus saw ... and "Baby Boy". **Label:** **the late 1990s**	**context:** Beyoncé Giselle Knowles-Carter ... is an American singer... and actress. **Label: UnAnswerable** **context:** Born and raised in Houston... and rose to fame in **the late 1990s** as lead singer ... Child. **Label:** **the late 1990s** **context:** Managed by her father... all time. **Label: UnAnswerable** ...

Table 2: Sample breakdown from paragraph to sentence, the answer is highlighted in **blue**.

4.2 Further cleanup of the data

Original SQuAD 2.0 data has about 130K questions in the train set. Blindly breaking them down into sentences will end up with around 667K questions, among which 87% are not answerable. This makes the NoAns ratio too high and might significantly bias the model.

To solve the problem, we filter out short sentences. More specifically, for HasAns examples, we filter out sentences with less than 50 characters; for NoAns examples, we filter out sentences with less than 200 characters. Please note that we are using a higher bar for NoAns questions, so as to reduce the ratio between NoAns and HasAns questions. After the filtering, we have about 190K questions, among which 55% are not answerable.

We conduct a similar process to the dev set. The only difference is that we filter out sentences with less than 50 characters for both HasAns and NoAns questions.

4.3 Aggregate Sentence Predictions

After training and predicting on sentence level using the BERT base model, we aggregate the sentence level predictions to paragraph level. If all sentences are predicted as NoAns, the paragraph would be predicted as NoAns; otherwise, the first sentence level answer will be selected as the answer for the whole paragraph. This simple approach works pretty well since we have not seen many cases where an answer is predicted for multiple sentences in one paragraph.

5. Results and Discussion

5.1 Impact of NoAns questions

The model performance is sensitive to NoAns proportions in the training set. We trained a base

BERT model multiple rounds with different proportions of unanswerable questions. The original train set has about 131K questions, among which 33% are unanswerable. When we increase the NoAns questions ratio in the training set, the model performs better on the NoAns prediction. Please note that in the entire process, we did not touch the dev set and kept it the same as that in SQuAD 2.0.

Figure 2 shows how model performance changes when we increase the NoAns questions ratios. For the last two data points in the chart, we over-sampled NoAns questions and down-sampled HasAns questions to go beyond this ratio.

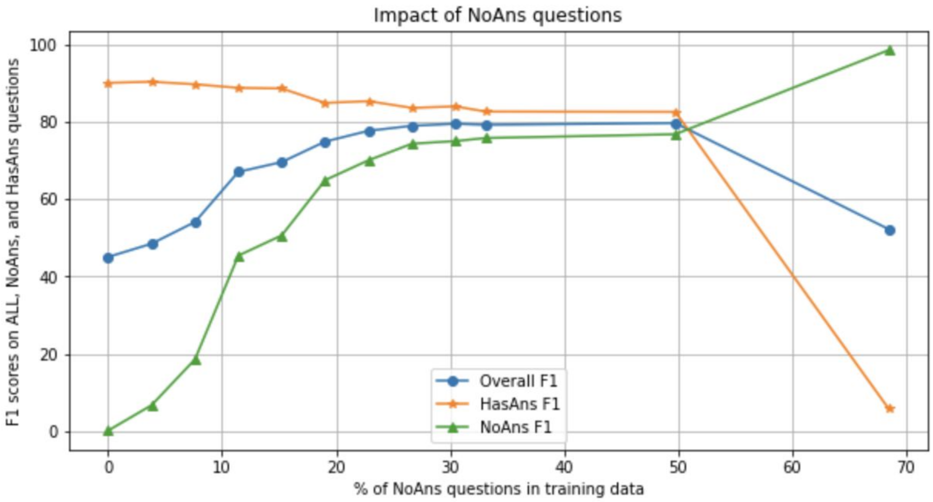


Figure 2: Model performance with different proportions of NoAns questions.

5.2 Model Evaluation

Table 3 presents the performance comparison between Paragraph Model and Sentence Model. The Sentence Model shows huge improvements on all the metrics, especially for the NoAns questions.

Models	**EM**	**F1**	**HasAns EM**	**HasAns F1**	**NoAns EM**	**NoAns F1**
Paragraph Model	76%	79%	77%	83%	75%	75%
Sentence Model	**90%** **(+14%)**	**94%** **(+15%)**	81% (+14%)	88% (+5%)	**99%** **(+24%)**	**99%** **(+24%)**

Table 3: Paragraph Model vs Sentence Model.

It's worth pointing out that the Sentence Model is so strong in predicting whether one question is answerable or not. When being evaluated as a binary classifier, the accuracy reaches 98.7%. For more details, please refer to Appendix 2.

Table 4 is a breakdown of Paragraph Model vs. Sentence Model performance in HasAns and NoAns categories. We can see that for the 1,368 of the total 5,945 NoAns questions(23%), the Sentence Model predicted correctly as NoAns, while the Paragraph Model predicted wrong.

This contributes a lot to the performance gain on unanswerable questions as well as the overall model performance improvement.

SQuAD 2.0 DEV SET Questions		**Paragraph Model Prediction**	**Sentence Model Prediction**	**# of questions**	**Paragraph Model**	**Sentence Model**
Total 11,873	**HasAns** 5,928	Wrong	Wrong	841	**Predicted HasAns** 6,708	**Predicted HasAns** 5,841
		Wrong	Correct	712		
		Correct	Wrong	436		
		Correct	Correct	3,939		
	NoAns 5,945	Wrong	Wrong	7	**Predicted NoAns** 5,165	**Predicted NoAns** 6,032
		Wrong	**Correct**	**1,368**		
		Correct	Wrong	31		
		Correct	Correct	4,539		

Table 4: Detailed breakdown of Paragraph Model vs Sentence Model

5.3 Why Sentence Model Outperform Paragraph Model

The reasons for the Sentence Model to outperform Paragraph Model are in two folds. First, most of the unanswerable questions from the SQuAD 2.0 data set can be decided on sentence level, meaning we do not need paragraph level context to decide whether the question is answerable. Second, the model trained on sentence level is more capable of identifying a series of complex linguistic phenomena that are particularly involved in unanswerable questions. This includes negation, antonymy, entity exchange, and word sequence change between questions and paragraphs.

Table 5 shows a negation example between the question and the paragraph. The question asks what issues were addressed, while the paragraph mentions that the discussions on the issues reached no decision. The sentence structure based negation makes the question unanswerable. In this case, the sentence level model predicted the question correctly while the paragraph level model predicted it wrong.

Question: What issues **were addressed** in the Treaty of Aix-la-Chapelle? **Paragraph:** ... The issues ... were turned over to a commission to resolve, **but** it reached no decision...	
Paragraph Model Result: **conflicting …**	**Sentence Model Result:** **NoAns**

Table 5: More Accurate Negation Detection in Sentence Level Trained Model

Appendix 3 shows more examples on word sequence change and entity exchange between the question and the context in the paragraph. In both cases and many more other cases indicated in Table 3, the sentence level trained model outperforms the paragraph level model.

When the same model is trained on sentences instead of paragraphs, since for each labeled sample, the data is smaller and more focused, it is easier for the model to learn and identify the subtle differences between the question and context, which widely exists in SQuAD 2.0 unanswerable questions. Therefore, the sentence level trained models are more capable of identifying linguistic phenomena of the language, while the paragraph level trained models are much weaker from this perspective. On the other hand, since most of the SQuAD 2.0 unanswerable questions can be judged on sentence level, sentence level model performs much better than that on the paragraph level.

6. Conclusion

In this paper, we analyzed and compared the unanswerable questions and answerable questions in SQuAD 2.0. We discovered that most SQuAD 2.0 questions can be answered by a single sentence, and sentence level trained models are more capable of learning and identifying complex linguistic phenomena involved in unanswerable questions. Based on these analysis results, we proposed a methodology with implementation to breakdown paragraphs into sentences for MRC readers to train and predict. We achieved significant performance improvements compared with that of the BERT base model.

As the next step for future research, we suggest evaluating SQuAD 2.0 data set to see if the unanswerable questions are aligned with the real world applications. In everyday life, is it the case that unanswerable questions are more involved in complex linguistic phenomena than answerable questions? Is it true that in real life, most answerable and unanswerable questions can be decided on sentence level without cross sentence paragraph context? If the answer is true, we would highly recommend sentence level learning for MRC applications using the proposed approach in this paper. Otherwise, SQuAD 2.0 data set needs to be revisited to accurately reflect the real world application needs. Furthermore, we should look into how to leverage sentence level models for more accurate question prediction for those that only need sentence level context, while leaving paragraph level models to focus on those that need cross sentence paragraph level context.

References

[1] SQuAD: 100,000+ Questions for Machine Comprehension of Text, Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, Percy Liang, arXiv:1606.05250, 2016, https://arxiv.org/abs/1606.05250,
[2] BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, arXiv:1810.04805, 2018, https://arxiv.org/abs/1810.04805

[3] Read + Verify: Machine Reading Comprehension with Unanswerable Questions, Minghao Hu, Furu Wei, Yuxing Peng, Zhen Huang, Nan Yang, Dongsheng Li,arXiv:1808.05759,2018, https://arxiv.org/pdf/1808.05759.pdf
[4] Relation Module for Non-answerable Prediction on Reading Comprehension,Kevin Huang, Yun Tang, Jing Huang, Xiaodong He, and Bowen Zhou, Proceedings of the 23rd Conference on Computational Natural Language Learning,2019. *(CoNLL)*https://www.aclweb.org/anthology/K19-1070.pdf
[5] Stochastic answer networks for machine reading comprehension.Xiaodong Liu, Yelong Shen, Kevin Duh, and Jian-feng Gao. 2018b. CoRR,abs/1712.03556. arXiv:1712.03556,2018.https://arxiv.org/abs/1712.03556
[6] U-net: Machine reading comprehension with unan-swerable questions.Association for the Advance-ment of Artificial Intelligence,Fu Sun, Linyang Li, Xipeng Qiu, and Yang Liu. arXiv:1810.06638, 2018.https://arxiv.org/abs/1810.06638
[7]Know What You Don't Know: UnAnswerable Questions, Pranav Rajpurkar, Robin Jia, Percy Liang, arXiv:1806.03822, 2018. (https://arxiv.org/abs/1806.03822)
[8]Simple and Effective Multi-Paragraph Reading Comprehension,Christopher Clark, Matt Gardner,arXiv:1710.10723, 2017.https://arxiv.org/abs/1710.10723
[9]Zero-Shot Relation Extraction via Reading Comprehension, Omer Levy, Minjoon Seo, Eunsol Choi, Luke Zettlemoyer, arXiv:1706.04115 ,2017. https://arxiv.org/abs/1706.04115
[10]A Nil-Aware Answer Extraction Framework for Question Answering,Souvik Kundu, Hwee Tou Ng,10.18653/v1/D18-1456,2018.https://www.aclweb.org/anthology/D18-1456
[11]Attention Is All You Need,Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin,arXiv:1706.03762, 2017.https://arxiv.org/abs/1706.03762
[12] NLP Tutorial: Question Answering System Using BERT + SQuAD On Colab TPU.

Appendix

1. GitHub with code and data

https://github.com/baokuiyang/unanswerable_question_detection_QA

2. Performance of the Sentence Model as a binary classifier

When evaluated as a binary classifier for HasAns and NoAns questions, the Sentence Model has accuracy of 98.7%.

Label	Prediction	# of questions	% of questions
NoAns	NoAns	5,916	49.8%
NoAns	HasAns	29	0.24%
HasAns	NoAns	126	1.06%
HasAns	HasAns	5,802	48.87%

Table 5: Sentence Model as a binary classifier.

3. More examples of unanswerable question prediction with complex linguistic phenomena.

Table 6 presents a word sequence change example between the question and the context in the paragraph. The Earth around the Moon in the question is different from the Moon around the Earth, which makes this question unanswerable. In this case, the paragraph level trained model predicted the wrong answer, while the sentence level trained model predicted correctly as no answer.

Question: Newton said that the acceleration of **the Earth around the Moon** represented what? **Paragraph:** ...In particular, Newton determined that the acceleration of **the Moon around the Earth** could be ascribed ...	
Paragraph Model Result:the same force...	**Sentence Model Result: NoAns**

Table 6: More Accurate Word Sequence Change Detection

Table 7 describes an example of entity exchange between the question and paragraph. 366,273 **households** in the question and 366,277 **people** in the paragraph context makes entity exchange between the question and paragraph, causing this question unanswerable. Again, the paragraph level trained model predicted the wrong answer, while the sentence level trained model predicted correctly as with no answer.

Question: What year was 366,273 people in Jacksonville? **Paragraph:** As of 2010[update], there were **366,273 households** out of...	
Paragraph Level Model Result:2010	**Sentence Level Model Result: NoAns**

Table 7: More Accurate Entity Exchange Detection

4. Other methods and approaches we tried

In order to achieve our goal of improving model performance on unanswerable questions, we also tried another approach of adding a binary classifier on top of the existing state-of-the-art MRC models. We used the open source BERT classification model to train a classifier on whether the question is answerable using SQuAD 2.0 data set. However, we obtained very low EM and F1 scores on both training and dev sets.

The reasons why we got such low model score could be in two folds:

- Predicting whether a question is answerable or not is more difficult, which is aligned with all the observations and analysis that we have conducted in this paper.
- It could be possible that the last layers in the classifier were not wired correctly, which we did not have enough time to confirm with the time constraints in this project.