Towards a more Robust Evaluation for Conversational Question Answering

Wissam Siblini, Baris Sayil, Yacine Kessaci
Worldline, France
{wissam.siblini,yacine.kessaci}@worldline.com
baris.sayil@insa-lyon.fr

Abstract

With the explosion of chatbot applications, Conversational Question Answering (CQA) has generated a lot of interest in recent years. Among proposals, reading comprehension models which take advantage of the conversation history (previous QA) seem to answer better than those which only consider the current question. Nevertheless, we note that the CQA evaluation protocol has a major limitation. In particular, models are allowed, at each turn of the conversation, to access the ground truth answers of the previous turns. Not only does this severely prevent their applications in fully autonomous chatbots, it also leads to unsuspected biases in their behavior. In this paper, we highlight this effect and propose new tools for evaluation and training in order to guard against the noted issues. The new results that we bring come to reinforce methods of the current state of the art.

1 Introduction

The ability to automatically answer questions from a set of raw text paragraphs has long been coveted by computer scientists (Woods, 1977). For applications in search engines, one could consider an isolated task where a user formulates a single question (Croft et al., 2010; Siblini et al., 2020). But recently, with usage in conversational agents (e.g. chatbots), a more contextualized variant referred to as Conversational Question Answering (CQA) has attracted a great deal of attention (Reddy et al., 2019; Choi et al., 2018). CQA differs from traditional (extractive) Question Answering (Rajpurkar et al., 2016) because Question-Answer (QA) pairs are not single but come in sequences within conversations. Therefore, models can use previous turns as context to extract the answer of the current question (Zhu et al., 2018; Huang et al., 2018; Qu et al., 2019a). In some cases, the history is even crucial to disambiguate pronouns in the question.

Similarly to other NLP tasks, the state-of-the-art approaches for CQA are variants of the Transformer Encoder (Vaswani et al., 2017), a deep neural network with several self-attention layers that produce contextualized representations of the "tokens" (words, subwords) that compose a text. For instance, models like BERT (Devlin et al., 2019; Lan et al., 2019; Sanh et al., 2019) obtain a more than decent performance on CQA datasets like QuAC (Choi et al., 2018) or CoQA (Reddy et al., 2019). However, they miss the context to fully understand the questions. Proposals have been made to integrate the history in several manners: using a recursive strategy (Huang et al., 2018), appending previous QAs to the current question as input (Zhu et al., 2018), and contextualizing the question-paragraph pair with respect to the history. We can mention in particular BERT-HAE and BERT-PHAE (Qu et al., 2019a,b) which improve BERT in a simple yet efficient way by encoding, in addition to segment and position, the fact that parts of the paragraph's words belonged to previous answers.

2 Motivation and main contributions

Our objective here is not to propose yet another model to try to obtain the best predictive score on CQA leaderboards. Instead, we focus our thinking around the current evaluation/training protocols with regards to the possible application cases. The starting point of our reflection is that currently, when evaluated on CQA datasets, models like BERT-HAE use the ground-truth answers of previous turns as context to answer the current question. This limits the scope of applicability to only a "semi-automatic" bot that would require a human providing supervision at each turn. We also show how it biases the selection of models towards those with an undesired filter behavior.

To make approaches from the literature usable

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Short Papers), pages 1028–1034
August 1–6, 2021.

in more difficult/realistic scenarios like standalone chatbots (in which they can only access the previous questions and their predictions of the answers), we make the following contributions: **(1)** We implement new evaluation tools to first highlight the current unnoticed and undesirable behavior: in ground-truth free conditions, CQA approaches can become even less accurate than baselines like BERT which do not exploit the history at all. **(2)** We develop the analog training protocol to make approaches robust to the observed issues. In particular, this gives back state-of-the-art models the strength to outperform the baseline but this time in a scenario that connects better to real-world conversational agents. Our work comes with an implementation of conversational QA tools, based on the most widely used transformers library (Wolf et al., 2019).

3 Conversational Question Answering

Conversational Question Answering (CQA) is a Natural Language Processing task related to Machine Comprehension (MC) (Zhang et al., 2019; Gupta et al., 2020). MC has grown significantly over the last decade, particularly thanks to (1) large scale datasets such as SQuAD (Rajpurkar et al., 2016) or Natural Questions (Kwiatkowski et al., 2019), (2) the improvement of representation learning models (Joulin et al., 2017), (3) powerful mechanisms such as attention (Yang et al., 2016; Vaswani et al., 2017), and (4) the emergence of several related topics like multi-lingual modeling (Pires et al., 2019; Siblini et al., 2019) or Conversational Question Answering (Choi et al., 2018; Reddy et al., 2019).

In CQA, questions are grouped in conversations and often require the context, i.e. previous QA turns, to be fully understandable. QuAC (Choi et al., 2018) and CoQA (Reddy et al., 2019) are two examples of CQA datasets. They were both generated by humans (a "student" and a "teacher") through conversations where the student asks a series of questions, complementary or not, on a given paragraph and the teacher answers them. In this paper, we focus on QuAC (Question Answering in Context) which is more recent and described as more challenging than CoQA (Choi et al., 2018). It contains 14k conversations and around 100k question-paragraph pairs, split into a training set (11,567 conversations / 83,568 questions), a validation set (1,000 conversations / 7,354 questions) and a test set. It evaluates models with several metrics, the main one being the F1-score (Flach, 2003).

Models proposed for QuAC are similar to those developed for SQuAD (e.g. BiDAF (Seo et al., 2016) or BERT (Devlin et al., 2019)) but they additionally integrate the history. A popular example is BERT-HAE (Qu et al., 2019a). It uses BERT's architecture but modifies the input embedding layer to add a novel component: the History Answer Embedding (HAE). As usual, the input question-paragraph pair is tokenized and marked with positions and segments. Then, an additional History Answer marker is added to indicate whether the tokens belonged to answers of previous questions or not, and the resulting embedding is simply added to the other embedding vectors (token, position, segment) before the self-attention blocks. BERT-HAE was enhanced, in a later publication (Qu et al., 2019b) by BERT-PHAE (Positional HAE) which additionally encodes the turn position of the answers in the history. Although very promising, we note that BERT-HAE and BERT-PHAE, as well as other state of the art models for QuAC, access the ground-truth answers of previous turns during evaluation. Therefore, reported results only reflect the performance within a reduced scope of applicability. In the following, we detail this limitation and propose to complement the current protocol in order to improve both evaluation and training.

4 A more Robust Protocol

Consider a standalone chatbot that successively answers questions from documents. At each turn, it cannot know for sure the ground truth (GT) answers of the previous turns except if the user or another human provides supervision. This could happen in scenarios where the role of the algorithm is only to provide answer suggestions (semi-automatic) to a human agent (e.g. in customer support). However, applications often seek bots where the question-answer loop is automated (standalone). Here we investigate this second setting. We start by reproducing the literature results on the semi-automatic scenario, then we exhibit the limits and propose solutions for our target scenario.

4.1 Reproducing the Regular Evaluation in the Semi-automatic Scenario

To evaluate the baseline performance (semi-automatic), we train BERT-HAE and BERT-PHAE on QuAC using the protocol described by the authors (Qu et al., 2019a) and the same hyperparame-

ters: history markers from up to 6 turns, and specific optimization parameters (12 as batch size, 3e-5 as learning rate with a linear decrease to 0 over 24k training steps). We implement our own training script on the basis of codes pieces from the transformers library (Wolf et al., 2019) and BERT-HAE's authors[1]. Experiments are run with a Nvidia Tesla V100 GPU.

Model	F1	Uses history
BiDAF++ (Choi et al., 2018)	51.8	No
BERT (Qu et al., 2019a)	54.4 (54.8)	No
BERT-HAE (Qu et al., 2019a)	63.1 (63.4)	Yes
BERT-PHAE (Qu et al., 2019b)	64.7 (64.4)	Yes

Table 1: F1-score of BERT, BERT-HAE, BERT-PHAE and a previous baseline on QuAC using the regular evaluation protocol. We display the original results published by the authors and the ones we reproduced (in parentheses).

Our results are roughly equal to those previously reported (Table 1). BERT's F1 score is 54.8, which compares favorably to previous baselines such as BiDAF. By adjusting the representation of the tokens based on the history of answers, BERT-HAE allows a significant improvement to 63.4 (+15.7 %). The position of the turns in the history also has its importance allowing BERT-PHAE to further improve the F1-score to 64.4. This is probably because questions are often related to the answers that directly precede them. To improve the results even further, one can also select a specific subset of turns in the history (Qu et al., 2019b).

4.2 Critical Analysis: The Filtering Behavior

Although promising, the aforementioned results need to be considered with caution. A hasty conclusion is that adding the history allows the model to benefit from a context and hence to better process the current question. However the improvement could also be explained by a bias in the dataset at hand. Indeed, this question answering task is extractive, i.e. answers are selected from a paragraph. In the course of a conversation in QuAC, an average of 7 questions are successively asked on the same rather small paragraph. Thus simply filtering the paragraph tokens with the answer history provides the advantage of reducing considerably the list of possible remaining answers. Note however that such a filtering could also have a negative effect, in the presence of overlap between answers.

[1] https://github.com/prdwb/bert_hae

To get better insights of the impact of a filtering behavior in practice, we run three experiments.

Model	F1	F1 w/ post filtering
BEST	**95.6**	92.7
BERT	54.8	**56.9**
BERT-HAE	**63.4**	62.5

Table 2: Evaluation of the impact of post-filtering on BEST, BERT and BERT-HAE.

Experiment 1: The negative impact of filtering due to overlap We first compute the best reachable F1-score (that we refer to as BEST) as if we had a model that always predicts the expected answer. Then we compute "BEST w/ post filtering" with the same predictions except that we post-filter all tokens that belong the 6 previous turns' answers, except for the "Cannot Answer" tokens (reserved for unanswerable questions). BEST F1 score is 95.6[2] while "BEST F1 w/ post filtering" is lower but very close: 92.7 (Table 2). This tells us that the maximal negative impact of a filtering strategy on QuAC is weak. We find an explanation by doing proportion measurements in QuAC's eval set: in particular, the percentage of overlapping tokens (resp. non overlapping tokens) between answers is low (resp. high): 5.7% (resp. 74.1%), the other 20.2% being the "Cannot Answer" tokens.

Experiment 2: Global impact of a post filtering on the models After 6 turns, sometimes almost half of the paragraph tokens belong to the history of answers. Even if experiment 1 suggests a negative impact of filtering due to overlap, the positive impact on our baselines (due to the significant reduction of the number of candidate answers) could counterbalance. We therefore re-evaluate the models trained in section 4.1, but this time we apply a post processing of their predictions: the start/end logits of tokens that belong to the answers of previous turns are set to $-\infty$, except for the "Cannot Answer" tokens. This forces previous answers to be excluded from the final predicted span text. This simple strategy to integrate the history in BERT allows an improvement to an F1-score of 56.9 (Table 2). On the contrary, it globally reduces the score

[2] Intuitively, it should be 100. But this value is unreachable in practice. The reason is that questions in QuAC have several acceptable answers (span texts of various length) and we select one randomly as BEST prediction. And, QuAC's official evaluation script computes, for each sample, the average F1 between the prediction and all possible answers.

of BERT-HAE to 62.5 (a reduction factor slightly lower than with BEST). These results suggest that access to ground-truth answers of previous turns allows in QuAC, in which the overlap is weak, a filtering mechanism to be a positive way of integrating history. Results also suggest that BERT-HAE might already implicitly integrate a filtering behavior. Unquestionably, it does it in a more expressive manner than our hard post-processing, since the history markers are passed as inputs to the model.

Experiment 3: Does BERT-HAE exhibit a filter behavior? Although suggested by the previous experiment, we want to answer this question more clearly. We consider an experiment aligned with the philosophy of adversarial attacks (Akhtar and Mian, 2018; Morris et al., 2020). During evaluation, we systematically modify the history answer markers so that the tokens of the current expected answer are marked as if they belonged to the history. The results obtained from this evaluation protocol are displayed under the column "F1 w/ Adv" in Table 3. F1 w/ Adv allows to measure, with the F1 metric, the ability of the models to answer a question when its answer has already appeared in the conversation before. In this condition, we observe a dramatic drop in BERT-HAE's performance (from 63.4 to 41.7), and an even worse for BERT-PHAE. This confirms that these models tend to output lower probabilities for tokens that are in the history, which suggests a filtering behavior and makes their usage potentially counter productive.

4.3 Proposed Evaluation for the Standalone Scenario

The current evaluation protocol on QuAC's validation set can bias model selection towards those able to implement a filtering behavior, which seems to be the case for BERT-(P)HAE. Thus, it does not guarantee a robust behavior in a fully autonomous bot. Here we propose an extension.

Inspired by the literature of recurrent models, we refer to the regular evaluation protocol, which access to ground truth answers of previous turns, as the **"Teacher Forcing" (w/ TF) protocol**. Analogically, we consider a mode **"without Teacher Forcing" (w/o TF)** where models process a conversation in the natural order and only use their predictions as history. The latter is outlined in Algorithm 1, where "build_mark" refers to a function that computes the new HAE markers given the previous ones and the new answer. Note that the algorithm for evaluation w/ TF simply replaces "build_mark(HAE,$\text{answer}_{\text{pred}}$)" with "build_mark(HAE,$\text{answer}_{\text{GT}}$)".

Algorithm 1 Evaluation w/o TF

```
s ← 0
for conversation ∈ valid set do
    HAE ← None
    for turn ∈ conversation do
        question ← turn['question']
        answer_GT ← turn['answer']
        answer_pred ← model(question, HAE)
        HAE ← build_mark(HAE,answer_pred)
        s ← s + F1(answer_pred,answer_GT)
    end for
end for
return s / card(valid set)
```

When we take the models trained in section 4.1 (w/ TF) and evaluate them with the new standalone protocol (w/o TF), Table 3 shows that the performance drops from 63.4 to 53.5 with BERT-HAE and from 64.4 to 54.2 with BERT-PHAE. Concretely, although unsuspected with the original protocol, the approaches do not necessarily seem advantageous compared to BERT here. This in no way detracts the interest of these proposals, which implement clever architectures to integrate the history. It only prevents their application, as is, in the standalone scenario. Nevertheless, now that this issue is identified, we can try to design an appropriate strategy to avoid it from the start, by taking measures at the training phase.

4.4 Training for the Standalone Scenario

To complement the proposed evaluation protocol with a training one, we propose to apply a recipe inspired by the most popular defense mechanism against adversarial attacks called adversarial training (Ren et al., 2020), i.e. we introduce the disruptive element (here the mode without Teacher Forcing) at training time. We consider three heuristics: (1) we disable TF during all the training steps (Robust), (2) we disable TF randomly based on a Coin Flip (Robust-CF), (3) we progressively disable TF from 0% of the steps to 100% of the steps over the training iterations (Robust-P). The new training process is detailed in Algorithm 2, where "update" refers to the optimization algorithm that updates the model based on the loss and "heuristic.condition" is a condition that depends on the

heuristic (e.g. always true for heuristic (1)). Note that both heuristics (2) and (3) are inspired from scheduled sampling methods (Bengio et al., 2015) adapted to the context of CQA.

Algorithm 2 Robust Training

```
for conversation ∈ train set do
    HAE ← None
    for turn ∈ conversation do
        question ← turn['question']
        answer_GT ← turn['answer']
        answer_pred ← model(question, HAE)
        l ← loss(answer_pred,answer_GT)
        update(model,l)
        if heuristic.condition then
            answer_add ← answer_pred
        else
            answer_add ← answer_GT
        end if
        HAE ← build_mark(HAE,answer_add)
    end for
end for
return model
```

We obtain encouraging results (Table 3). In particular, BERT-PHAE Robust-P reaches a F1-score of 58.1 in the standalone scenario which is better than BERT's F1. Besides, "F1 /w Adv" for BERT-(P)HAE Robust seems to indicate that, the less we apply TF, the less the entailed model exhibits a filtering behavior. In fact, all the robust variants exhibit a weaker filtering behaviour than the original methods.

Model	F1 w/ TF	F1 w/o TF	F1 w/ Adv
BERT	-	54.4	-
BERT-HAE	**63.4**	53.5	41.7
BERT-HAE Robust	59.5	56.6	**51.7**
BERT-HAE Robust-CF	61.6	55.9	47.4
BERT-HAE Robust-P	60.7	**56.7**	50.7
BERT-PHAE	**64.4**	54.2	40.7
BERT-PHAE Robust	60.5	57.4	**53.3**
BERT-PHAE Robust-CF	62.2	56.4	47.7
BERT-PHAE Robust-P	62.4	**58.1**	51.6
BERT-AH	-	**58.3**	-

Table 3: Evaluation of BERT, BERT-HAE, BERT-PHAE, BERT-AH and the robust variants with different validation protocols.

Our experiment and results leave room for improvement with additional considerations on protocols/parameters/models. For instance, contrary to answers, standalone models can have access to the exact history of questions. What if we integrated the latter instead of the answer history in the model's input? We tested this by implementing a simple model that we refer to as BERT-AH (Appended History) in which previous questions are added to the regular BERT's inputs, and marked with a special embedding. BERT-AH obtains an F1-score of 58.3 (whatever the evaluation protocol, since answer history is not used). Thus, our guess is that the best direction for standalone CQA lies towards both the integration of previous questions and the robust integration of previous answers.

5 Conclusion

The work presented in this paper comes to complement the current training and evaluation protocols for CQA. It allows (1) highlighting unnoticed and undesirable behavior in existing approaches from the literature and (2) more robustness for their application in autonomous chatbots. We hope that this will encourage additional proposals in the same direction. Several improvements could be made in the future. First, because without Teacher Forcing the history is now predicted and not fixed, we could explore the impact of updating the model by back-propagating an answer's error through all previous turns and not only the current one. This would be analog to backpropagation through time. Second, we could augment the current CQA datasets or propose new ones to prevent the biases we observed: for example, QuAC could have conversations including wrong answers, since this occurs in real-life, so that models could be properly trained for. The associated turns would of course only be used as a part of histories. Finally, we should perform user tests to evaluate the robustness of models in real-life, because when a model's answer is wrong, we expect it to impact the next user's question(s). And this cannot be taken into account with the current protocol since the datasets are static.

References

Naveed Akhtar and Ajmal Mian. 2018. Threat of adversarial attacks on deep learning in computer vision: A survey. *Ieee Access*, 6:14410–14430.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wentau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question answering in con-

text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2174–2184, Brussels, Belgium. Association for Computational Linguistics.

W Bruce Croft, Donald Metzler, and Trevor Strohman. 2010. *Search engines: Information retrieval in practice*, volume 520. Addison-Wesley Reading.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Peter A Flach. 2003. The geometry of roc space: understanding machine learning metrics through roc isometrics. In *Proceedings of the 20th international conference on machine learning (ICML-03)*, pages 194–201.

Somil Gupta, Bhanu Pratap Singh Rawat, and Hong Yu. 2020. Conversational machine comprehension: a literature review. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2739–2753, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Hsin-Yuan Huang, Eunsol Choi, and Wen-tau Yih. 2018. Flowqa: Grasping flow in history for conversational machine comprehension. In *International Conference on Learning Representations*.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

John Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020. TextAttack: A framework for adversarial attacks, data augmentation, and adversarial training in NLP. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 119–126, Online. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Chen Qu, Liu Yang, Minghui Qiu, W Bruce Croft, Yongfeng Zhang, and Mohit Iyyer. 2019a. Bert with history answer embedding for conversational question answering. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1133–1136.

Chen Qu, Liu Yang, Minghui Qiu, Yongfeng Zhang, Cen Chen, W Bruce Croft, and Mohit Iyyer. 2019b. Attentive history selection for conversational question answering. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 1391–1400.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Kui Ren, Tianhang Zheng, Zhan Qin, and Xue Liu. 2020. Adversarial attacks and defenses in deep learning. *Engineering*, 6(3):346–360.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.

Wissam Siblini, Mohamed Challal, and Charlotte Pasqual. 2020. Delaying interaction layers in transformer-based encoders for efficient open domain question answering. *arXiv preprint arXiv:2010.08422*.

Wissam Siblini, Charlotte Pasqual, Axel Lavielle, Mohamed Challal, and Cyril Cauchois. 2019. Multilingual question answering from formatted text applied to conversational agents. *arXiv preprint arXiv:1910.04659*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, pages arXiv–1910.

William A. Woods. 1977. Lunar rocks in natural english: Explorations in natural language question answering. *Linguistic Structures Processing*, pages 521–569.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, San Diego, California. Association for Computational Linguistics.

Xin Zhang, An Yang, Sujian Li, and Yizhong Wang. 2019. Machine reading comprehension: a literature review. *arXiv preprint arXiv:1907.01686*.

Chenguang Zhu, Michael Zeng, and Xuedong Huang. 2018. Sdnet: Contextualized attention-based deep network for conversational question answering. *arXiv preprint arXiv:1812.03593*.