Having Your Cake and Eating it Too: Training Neural Retrieval for Language Inference without Losing Lexical Match

Vikas Yadav
University of Arizona
vikasy@email.arizona.edu

Steven Bethard
University of Arizona
bethard@email.arizona.edu

Mihai Surdeanu
University of Arizona
msurdeanu@email.arizona.edu

ABSTRACT

We present a study on the importance of information retrieval (IR) techniques for both the interpretability and the performance of neural question answering (QA) methods. We show that the current state-of-the-art transformer methods (like RoBERTa) encode poorly simple information retrieval (IR) concepts such as lexical overlap between query and the document. To mitigate this limitation, we introduce a supervised RoBERTa QA method that is trained to mimic the behavior of BM25 and the soft-matching idea behind embedding-based alignment methods. We show that fusing the simple lexical-matching IR concepts in transformer techniques results in improvement a) of their (lexical-matching) interpretability, b) retrieval performance, and c) the QA performance on two multi-hop QA datasets. We further highlight the lexical-chasm gap bridging capabilities of transformer methods by analyzing the attention distributions of the supervised RoBERTa classifier over the context versus lexically-matched token pairs.

CCS CONCEPTS

• **Information systems → Question answering**;

KEYWORDS

Semantic alignment; Question answering; Interpretability

ACM Reference Format:
Vikas Yadav, Steven Bethard, and Mihai Surdeanu. 2020. Having Your Cake and Eating it Too: Training Neural Retrieval for Language Inference without Losing Lexical Match. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20), July 25–30, 2020, Virtual Event, China.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3397271.3401311

1 INTRODUCTION

In recent years, the "deep learning tsunami"[11] has proved to be more successful in learning complex inference required for several QA tasks. On the other hand, conventional information retrieval (IR) techniques are widely used for simple retrieval tasks. Although transformer-based [14] methods have achieved groundbreaking performance in many NLP tasks (especially QA [6]), they have shown to perform worse than IR techniques on simple retrieval based QA tasks [12, 16]. We argue that a generalizable neural QA approach should solve both the simple lexical-matching (or retrieval) based questions and the complex-inference based questions altogether.

In this work, we analyze the state-of-the-art transformer based QA method - RoBERTa[10] for simpler lexical-matching and contextual reasoning propoerties. We first show that the pretrained transformer methods lack the simple lexical matching capabilities in both the retrieval and the QA task. This is intuitive, as none of the components in the pretraining objectives of transformers motivate simple lexical matching between two sentences[6]. We also show that when the transformer methods are finetuned for learning a QA task, they somewhat learn to align the lexical matching words but distribute substantial attention on the context tokens, thus bridging the "lexical chasm" [1] necessary for complex inference and reasoning. Hence, we propose a simple strategy to infuse the lexical matching capabilities within RoBERTa using transfer learning framework. Specifically, we propose a two-stage training process where the simpler things are learned first i.e., we first fine-tune RoBERTa by training it to predict *unsupervised* IR scores, which motivates learning to lexical match. In the second step, we transfer this (IR) learned knowledge into the final task by continuing training the model on the gold labels of the final task. We show that this approach results in (a) end-task performance improvement, and (b) substantial increase in attention from lexically matched token pairs on the final prediction. Our key contributions are:

1) We present a simple strategy[1] to infuse lexical matching IR concepts in state-of-the-art transformer method (RoBERTa) and show that it directly improves the end-task performance.
2) After infusing lexical-matching, we show 3%F1 improvement on evidence selection task and 4%F1 improvement on the QA task in MultiRC [9]. These improvements also establish the new state-of-the-art evidence retrieval results on MultiRC, hence directly improving the explainability of the QA process. We also achieve 1.6% F1 improvement on AI2 reasoning challenge (ARC) dataset [4] where the improvement on Easy partition of ARC (having simple retrieval based questions[4]) is 2%. Importantly, the performance on the challenge partition of ARC (having questions which require complex inference and reasoning) is nearly unaffected suggesting that our approach enhances the simple question solving capabilities without affecting the other deep inference properties of RoBERTa.
3) We present an attention based interpretability analysis showing that the RoBERTa model infused with IR concepts indeed pays more attention to the lexically matching tokens in the QA pair.

2 RELATED WORK

Transformer methods have achieved state-of-the-art (SOTA) performance on several complex reasoning based QA and IR tasks[6, 15,

[1] Codes - https://github.com/vikas95/Lexical_Match_Interpretability

SIGIR '20, July 25–30, 2020, Virtual Event, China

ACM ISBN 978-1-4503-8016-4/20/07…$15.00
https://doi.org/10.1145/3397271.3401311

17]. However, they have also achieved low performance on simpler retrieval based QA tasks [12, 16]. Analyzing attention weights have been widely studied for interpreting the behavior of neural methods [3]. Especially the attention contribution on lexical-matched tokens is often studied as a proxy to interpret neural models on (retrieval based) QA [7] and natural language inference task[8]. Thus, to show the lexical-matching interpretability of the models in our experiments, we analyze the attention from lexical-matched token pairs on the final prediction scores of a given task.

IR methods are also widely used for retrieving the knowledge text which are then fed to a QA method [2]. Recently, the quality of these knowledge text are given substantial importance as they provide interpretability of the QA process[9, 15] to humans. We also experiment with interpretable multi-hop QA (MultiRC[9]) in this work to analyze the strength/weakness of neural methods on qualitative knowledge text retrieval task.

Several works have utilized IR scores to train neural methods as an alternative to annotations in the retrieval tasks [5] often along with denoising rules[15]. Some of our results also show similar patterns but the main focus of our work is on inducing and interpreting the lexical matching concepts within the neural methods. Further, our work focuses on improving the performance of neural methods on simpler lexical-matching based QA or retrieval task while maintaining their deep inference properties.

3 TASKS AND APPROACH

We considered two end tasks for evaluating our approach - 1) retrieval of evidence text for better explainable QA process and 2) QA task itself. We experiment with 2 multi-hop QA datasets which require knowledge aggregation from >= 2 facts[17].

3.1 Data

Multi-sentence reading comprehension (MultiRC) is a reading comprehension dataset provided in the form of multiple-choice QA task [9]. Every question is based on a paragraph, which contains the gold justification sentences for each question. There are two main tasks in MultiRC - *evidence selection* and *QA*. In the evidence selection task, only the question relevant sentences (known as justifications) are needed to be retrieved which provide QA-pair linking explanation to the end user. We use all the sentences of a paragraph as candidate justifications for a given question where gold justification sentences receive the positive label and the remaining sentences receive the negative label. The retrieved evidence text is then fed to another answer classification module [2] which predicts if the candidate answer is correct/incorrect based on the justification sentences. In MultiRC experiments, we focus on inducing lexical match just for the evidence selection task. We use the original MultiRC dataset[3] for our experiments.

AI2 Reasoning Challenge (ARC) is a multiple-choice question answering (MCQA) dataset, constructed from science exam questions. [4]. The dataset has two partitions: Easy and Challenge, where the latter partition contains the more difficult questions that require reasoning. ARC also includes a textual knowledge base (KB) which contains 14.3M sentences suitable for solving ARC questions. We use off-the-shelve BM25[4] for retrieving question relevant knowledge setences from this KB. Then, we consider the MCQA task as a binary classification task to find similarity between two text for each candidate answer. Here, the first text is the concatenated text of question and the candidate answer (referred to as query) and the second text is the knowledge sentence retrieved from the KB. Similar to Yadav et al. [17], we consider k knowledge sentences for each candidate answer and compute the average of k scores as the final similarity score. The candidate answer with the maximum similarity score is predicted as the final answer. In ARC experiments, we focus on inducing lexical match just for the QA task.

3.2 Approach

Our main approach focuses on inducing lexical-matching in neural methods by training them directly on the scores returned by the IR methods. Importantly, utilizing unsupervised IR scores is more pragmatic as IR techniques do not require annotations or any specific structured resources. Our training strategy is divided into two steps where, lexical-matching is induced in the first step, and the end-task specific training is continued in the second step.

Step 1 - We first train the RoBERTa for predicting the IR score. In the evidence retrieval task of MultiRC, the (scaled) IR retrieval score between the query and the candidate justification becomes the labelled score. In MCQA task of ARC, the labelled QA score is the (scaled) averaged IR scores between the query and k knowledge sentences. Specifically, we use the following two standard IR approaches for computing the retrieval score:

BM25: We use the Lucene's BM25 for computing similarity scores between a query and the document [13], using the default values of hyperparameters.

Alignment: We use the alignment approach of Yadav et al. [16]. The alignment method [18] computes the cosine similarity between the word embeddings of each token in the query and each token in the given KB sentence, resulting in a matrix of cosine similarity scores. For each query token, the algorithm selects the most similar token in the evidence text using max-pooling. At the end, the element-wise dot product between this max-pooled vector of cosine-similarity scores and the vector containing the IDF values of the query tokens is calculated to produce the overall alignment score s for the given query Q and the supporting paragraph P_j:

$$s(Q, P_j) = \sum_{i=1}^{|Q|} idf(q_i) \cdot align(q_i, P_j) \quad (1)$$

$$align(q_i, P_j) = \max_{k=1}^{|P_j|} cosSim(q_i, p_k) \quad (2)$$

where q_i and p_k are the i^{th} and k^{th} terms of the query (Q) and justification sentence (P_j) respectively. We train the RoBERTa to predict this score after scaling it (see eq. (3))

Step 2 - In the second step, we transfer and continue training the model from the first step on the end task. In this step, the model is trained on annotated gold labels of justifications in MultiRC and the annotated correct/incorrect candidate answers in ARC.

We consider the tasks in both steps as a regression task because of the continuous values of the IR scores. In the first step, the IR scores are scaled between 0 and 5 using eq. (3).

[2]Similar to [15, 17], we use a seperate RoBERTa as binary classifer for the answer classification module

[3]https://cogcomp.seas.upenn.edu/multirc/

[4]https://lucene.apache.org

#	Approach	Justification Selection			Question Answering			Attention Scores (12^{th} layer)	
		P	R	F1	$F1_m$	$F1_a$	EM0	Lexical-matched	Context
	Unsupervised baselines								
1	BM25 (IR) (Khashabi et al. [9])	42.6	56.1	48.4	64.3	60.0	-	-	-
2	Alignment	62.4	55.6	58.8	72.6	69.6	25.9	-	-
3	Pretrained RoBERTa	-	-	-	-	-	-	41.0	59.0
4	RoBERTa ($BM25_{score}$)	53.8	54.8	54.3	70.1	67.9	21.8	54.8	45.2
5	RoBERTa ($Align_{score}$)	63.9	60.9	62.4	72.0	69.8	24.6	54.6	45.4
	Supervised baselines and previous SOTA								
6	RS (GPT-2) (Wang et al. [15])	-	-	60.7	73.1	70.5	20.8	-	-
7	AutoROCC (BERT) (Yadav et al. [17])	63.4	61.1	62.3	72.9	69.6	24.7	-	-
8	RoBERTa-retriever	64.5	64.6	64.6	70.5	68.0	24.9	47.9	52.1
	Our approach								
9	RoBERTa ($BM25_{score}$)-retriever	**65.4**	67.1	66.2	73.1	71.2	25.7	52.1	47.9
10	RoBERTa ($Align_{score}$)-retriever	65.1	**69.6**	**67.3**	**74.3**	**72.7**	**27.1**	52.4	47.6

Table 1: Performance on the MultiRC development set with official evaluation metrics[9]. RoBERTa ($BM25_{score}$) and RoBERTa ($Align_{score}$) are the RoBERTa model trained on unsupervised BM25 and alignment scores respectively. RoBERTa-retriever is the RoBERTa model trained for retrieving justification sentences from gold annotated labels. RoBERTa ($BM25_{score}$)-retriever is first fine-tuned on predicting BM25 scores and then further fine-tuned on predicting the gold justification selection labels. The numbers in bold indicate the new state-of-the-art results.

$$Scale(s(Q, P_j)) = \frac{s(Q, P_j) - S_{min}}{S_{max} - S_{min}} * 5 \quad (3)$$

where S_{max} and S_{min} are the maximum and the minimum justification retrieval score for a given question.

For the second step, we give the score of 5 for the positive label and the score of 0 for the negative label. Hence, the first step has continuous values between 0 and 5 and the second step (end task) have just two values (i.e., 0 and 5). The hyperparameters used are common for both the steps except the number of epochs. We found that training the model for 3 epochs in the first step and 4 epochs in the second step leads to consistent improvement across all the settings. The other common hyperparameters are batch size = 32, maximum sequence length = 128, gradient accumulation step of 8.

4 ANALYSIS AND RESULT DISCUSSION

To understand the changes within the model from infusion of lexical-matching, we analyze the attention weights of RoBERTa.

4.1 Attention Analysis

The representation of [*CLS*] token is fed into the linear layer for the regression task[6]. Hence, we analyze the average contributions from attention of lexical-matched tokens on the [*CLS*] representation versus the average contributions from non-lexical-matching tokens. We compute the attention on the [*CLS*] from a given token by summing up the attention scores from all the 12 heads in each layer[3]. We repeat the same for all the tokens of text 1 (query) and text 2 (knowledge sentence). Then we average the attention scores from all the token pairs matching lexically in both text 1 and text 2, refering it to as lexically-matched attention (LMA) scores. Similarly, we compute the average attention scores from the other remaining words in text 1 and text2, refering it to as context attention (CA) scores. We normalize the attention scores on [*CLS*] in each layer such that LMA and CA sum up to 1 for every layer. The LMA scores for MultiRC justification selection task and ARC MCQA task are presented in table 1 and fig. 1 respectively.

4.2 Analysis of performance gains

We analyze the gains achieved in table 1 and table 2 after infusing lexical-matching. As shown in table 2 (rows 11-13), the performance of RoBERTa($Align_{score}$)-QA improves majorly on ARC easy which contains retrieval based questions[4]. Notably, the performance on ARC challenge which requires complex inference and reasoning, remains largely unaffected . Hence, our approach that uses transfer learning allows the model to strengthen its lexical-matching capabilities on the easy questions while still maintaining the complex reasoning properties for the challenging questions.

In MultiRC[5], the justifications correctly predicted by RoBERTa-retriever (row 8, table 1) overlap with only 48.5% and 47.6% of the justifications correctly predicted by BM25 (row 1) and the alignment method (row 2) respectively. This overlap of correctly predicted justifications increases to 55.6% between {RoBERTa($BM25_{score}$)-retriever and BM25} and 55.1% between {RoBERTa ($Align_{score}$)-retriever and alignment method} (row 9-10) indicating that the gains come from solving the simpler lexical-matching based justifications.

4.3 Result discussion

We draw several observations from table 1 and table 2:

• Inducing lexical-matching with training on unsupervised IR scores improves both the retrieval performance (upto 2.7% F1) (line 9-10 vs line 8 in table 1) and the QA performance (2% F1) (line 11-13 vs lines 5-7 in table 2). We did not observe gains from RoBERTa ($BM25_{score}$)-QA (rows 8-10) in table 2 highlighting that QA is a more challenging task and imposing strict lexical matching may (slightly) effect the performance. But, we see consistent improvements with RoBERTa($Align_{score}$)-QA which uses soft-matching of token pairs in embedding space rather than strict lexical matching.
• The lexically-matched attention (LMA) scores improve by 14% after training the RoBERTa on IR scores (row 3 vs rows 4-5 in table 1). When the training is continued on the gold justification labels,

[5]The test set of MultiRC is hidden which is changed periodically[9]. For fair comparison with previous works, we present the analysis on publicly available dev set

#	Approach	ARC All (F1)	ARC Challenge (F1)	ARC Easy (F1)
	Unsupervised Baselines			
1	BM25 (IR solver) [4]	-	23.98	59.99
2	Alignment [17]	-	26.56	58.36
3	RoBERTa($BM25_{score}$) ($k = 3$)	40.28	26.19	47.22
4	RoBERTa($Alignment_{score}$) ($k = 3$)	44.06	30.11	50.92
	Supervised Baselines			
5	RoBERTa-QA ($k = 3$)	57.36	**42.66**	64.56
6	RoBERTa-QA ($k = 4$)	56.72	41.30	64.30
7	RoBERTa-QA ($k = 5$)	55.99	41.47	63.13
	BM25 fine-tuned RoBERTa			
8	RoBERTa ($BM25_{score}$)-QA ($k = 3$)	56.33	39.42	64.64
9	RoBERTa ($BM25_{score}$)-QA ($k = 4$)	56.30	39.60	64.52
10	RoBERTa ($BM25_{score}$)-QA ($k = 5$)	54.39	37.97	62.46
	Alignment fine-tuned RoBERTa			
11	RoBERTa($Align_{score}$)-QA ($k = 3$)	**58.47**	41.94	**66.34**
12	RoBERTa($Align_{score}$)-QA ($k = 4$)	58.25	41.89	66.29
13	RoBERTa($Align_{score}$)-QA ($k = 5$)	56.27	41.55	63.51

Table 2: Performance on ARC test dataset. Notations are same as in table 1 except RoBERTa-QA is the model trained on gold labels of MCQA task of ARC. RoBERTa ($BM25_{score}$)-QA and RoBERTa ($Align_{score}$)-QA are first fine-tuned for predicting BM25 and alignment scores respectively and then further fine-tuned on the gold MCQA labels. k indicates the number of KB sentences. Bold numbers indicate the best performance amongst transformer models.

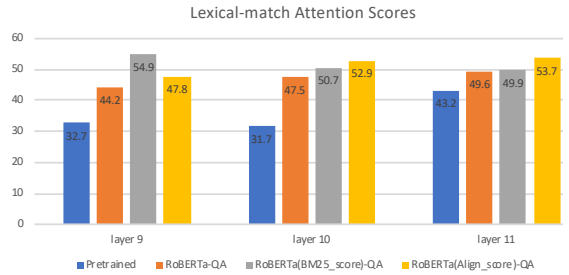

Figure 1: Histogram depicting the lexical-match attention(LMA) scores on the $[CLS]$ token across the last 3 layers of RoBERTa trained on ARC.

model tends to attend more on the contextual words and lesser on lexically-matched words(rows 4-5 vs rows 9-10), hence bridging the lexical-chasm for reasoning. We see similar attention score patterns in ARC. Importantly, these patterns are more consistent across the last 3-4 layers of RoBERTa (as shown in fig. 1) but differences amongst attention scores tend to diminish in the bottom few layers which are farthest away from the decision layer.

- RoBERTa($Align_{score}$)-QA achieves consistent improvements on ARC easy and maintains close performance (or minute gains) within the challenge partition (row 5-7 vs row 11-13 in table 2) suggesting that infusing lexical-matching in QA can benefit the simpler word-match question and simultaneously maintain the complex inference properties required for solving challenging questions.
- Similarly, the gains in MultiRC are likely from the justifications that can be retrieved with lexical matching(4.2). Overall, the increase in LMA scores point towards the increased performance of justification retrieval and retrieval based questions, hence improving the interpretability of RoBERTa on simpler retrieval tasks.

5 CONCLUSION

We highlighted the lack of lexical-matching in neural methods. We presented a simple approach to infuse lexical matching using unsupervised IR methods into a state-of-the-art transformer method - RoBERTa. We show that infusing lexical-matching improves the performance on simpler retrieval based question and the (justification) retrieval task itself. Importantly, our proposed strategy does not effect the deeper inference properties of RoBERTa necessary for bridging the lexical-chasm in QA. Finally, we show that the best retrieval and QA performance can be achieved by creating a balance of attention from both the lexical-matching and contextual-inference.

ACKNOWLEDGMENTS

This work was supported by the Defense Advanced Research Projects Agency (DARPA) under the World Modelers program, grant number W911NF1810014. Mihai Surdeanu declares a financial interest in lum.ai. This interest has been properly disclosed to the University of Arizona Institutional Review Committee and is managed in accordance with its conflict of interest policies.

REFERENCES

[1] Adam Berger, Rich Caruana, David Cohn, Dayne Freytag, and Vibhu Mittal. 2000. Bridging the Lexical Chasm: Statistical Approaches to Answer Finding. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research & Development on Information Retrieval*. Athens, Greece.

[2] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051* (2017).

[3] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. 2019. What Does BERT Look at? An Analysis of BERT's Attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. 276–286.

[4] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge. *arXiv preprint arXiv:1803.05457* (2018).

[5] Mostafa Dehghani, Hamed Zamani, Aliaksei Severyn, Jaap Kamps, and W Bruce Croft. 2017. Neural ranking models with weak supervision. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 65–74.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805* (2018).

[7] Andrea Galassi, Marco Lippi, and Paolo Torroni. 2019. Attention, please! a critical review of neural attention models in natural language processing. *arXiv preprint arXiv:1902.02181* (2019).

[8] Reza Ghaeini, Xiaoli Z Fern, and Prasad Tadepalli. 2018. Interpreting recurrent and attention-based neural models: a case study on natural language inference. *arXiv preprint arXiv:1808.03894* (2018).

[9] Daniel Khashabi, Snigdha Chaturvedi, Michael Roth, Shyam Upadhyay, and Dan Roth. 2018. Looking beyond the surface: A challenge set for reading comprehension over multiple sentences. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics*. 252–262.

[10] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* (2019).

[11] Christopher D Manning. 2015. Computational linguistics and deep learning. *Computational Linguistics* 41, 4 (2015), 701–707.

[12] Harshith Padigela, Hamed Zamani, and W Bruce Croft. 2019. Investigating the successes and failures of BERT for passage re-ranking. *arXiv preprint arXiv:1905.01758* (2019).

[13] Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval* 3, 4 (2009), 333–389.

[14] 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*. 5998–6008.

[15] Hai Wang, Dian Yu, Kai Sun, Jianshu Chen, Dong Yu, David McAllester, and Dan Roth. 2019. Evidence Sentence Extraction for Machine Reading Comprehension. In *Proceedings of the 23rd CoNLL)*. 696–707.

[16] Vikas Yadav, Steven Bethard, and Mihai Surdeanu. 2019. Alignment over Heterogeneous Embeddings for Question Answering. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, (Long Papers)*.

[17] Vikas Yadav, Steven Bethard, and Mihai Surdeanu. 2019. Quick and (not so) Dirty: Unsupervised Selection of Justification Sentences for Multi-hop Question Answering. In *Proceedings of the 2019 EMNLP-IJCNLP*. 2578–2589.

[18] Vikas Yadav, Rebecca Sharp, and Mihai Surdeanu. 2018. Sanity Check: A Strong Alignment and Information Retrieval Baseline for Question Answering. (2018).