Exploring Contrast Consistency of Open-Domain Question Answering Systems on Minimally Edited Questions

Zhihan Zhang, Wenhao Yu, Zheng Ning, Mingxuan Ju, Meng Jiang

University of Notre Dame, Notre Dame, IN, USA

{zzhang23, wyu1, zning, mju2, mjiang2}@nd.edu

Abstract

Contrast consistency, the ability of a model to make consistently correct predictions in the presence of perturbations, is an essential aspect in NLP. While studied in tasks such as sentiment analysis and reading comprehension, it remains unexplored in open-domain question answering (OpenQA) due to the difficulty of collecting perturbed questions that satisfy factuality requirements. In this work, we collect minimally edited questions as challenging contrast sets to evaluate OpenQA models. Our collection approach combines both human annotation and large language model generation. We find that the widely used dense passage retriever (DPR) performs poorly on our contrast sets, despite fitting the training set well and performing competitively on standard test sets. To address this issue, we introduce a simple and effective query-side contrastive loss with the aid of data augmentation to improve DPR training. Our experiments on the contrast sets demonstrate that DPR's contrast consistency is improved without sacrificing its accuracy on the standard test sets.[1]

1 Introduction

Contrast consistency (Gardner et al., 2020) is a crucial aspect for neural models in NLP. Models are expected to identify perturbations in the text input and decide whether such a semantic shift leads to a different label. To evaluate this consistency, contrast sets have been introduced in various tasks such as sentiment analysis (Wu et al., 2021), natural language inference (Ross et al., 2022), and reading comprehension (Longpre et al., 2021) by minimally modifying the original input to reverse the original label. However, to our best knowledge, there is no study on the contrast consistency in open-domain question answering (OpenQA). In OpenQA, even a

slight modification of a word or two can alter the meaning of the question, which leads to a completely different answer. To maintain contrast consistency, models are expected to predict the corresponding answer when such semantic shift occurs.

Studying contrast consistency in OpenQA poses unique challenges. Firstly, collecting appropriate contrast sets is difficult. While contrast sets have been developed for reading comprehension (Longpre et al., 2021; Li et al., 2022), they typically replaced an entity (*e.g.*, Barack Obama was born in *Hawaii*) in given context with another entity (*e.g.*, Barack Obama was born in *New York*), leading to a different answer to the given question (*e.g.*, Where was Barack Obama born?). Constructing such contrast sets does not necessitate the factuality of the perturbed context, as the answer depends solely on the context rather than world knowledge. However, in the absence of evidence context, the perturbed questions in OpenQA must be factually answerable in accordance with world knowledge, which is beyond what rule-based methods can do. Secondly, achieving contrast consistency is challenging for OpenQA models, which usually follow the ''retrieve-then-read'' pipeline (Lewis et al., 2020). In addition to the challenge of predicting answers from a contrast context as in reading comprehension, models also face the challenge of mapping the perturbed question with its corresponding evidence passage in a large corpus. The latter requires the retriever to distinguish the minimal semantic difference between embeddings of the perturbed question and the original question, which is ignored in typical retriever training.

To fill this gap in OpenQA, we propose to create contrast sets using **M**inimally **E**dited **Q**uestions (MEQs). Given a question q and its answer a, an MEQ q' is defined as a question that possesses high lexical and semantic similarity with q, while having a distinct answer a' ($a' \neq a$). For

---

[1] Data and code are available at https://github.com /ytyz1307zzh/Minimally_Edited_Questions.

q_1^+

ε

q_1 → p_1
("Pet Sematary 2")

q_2 ⊗

p_2
("Pet Sematary")

q_3 → p_3
("Australian one-cent coin")

(a) Original DPR: Not contrast consistent

q_1^+

ε

q_1 → p_1
("Pet Sematary 2")

q_2 ✓ → p_2
("Pet Sematary")

q_3 → p_3
("Australian one-cent coin")

(b) Our improved DPR: Contrast consistent!

q_1 Q: Where was the movie Pet Sematary 2 filmed? A: Coweta County, Georgia

q_1^+ Q: Where was the movie Pet Sematary 2 taken? A: Coweta County, Georgia ← Paraphrase

Contrast

q_2 Q: Where was the movie Pet Sematary filmed? A: Maine

q_3 Q: When did Australia stop using one cent coins? A: 1992

○ Original question (training set)
● Paraphrase question
● Random question (standard test set)
● Minimally edited question (contrast set)

▨ ▨ ▨ Evidence passages to corresponding questions

Ranking MRR: 0.921, 0.784, 0.563, 0.507
Retrieval Recall: 93.2, 70.0, 50.0, 57.3
QA Accuracy: 63.2, 43.2, 19.7, 25.8
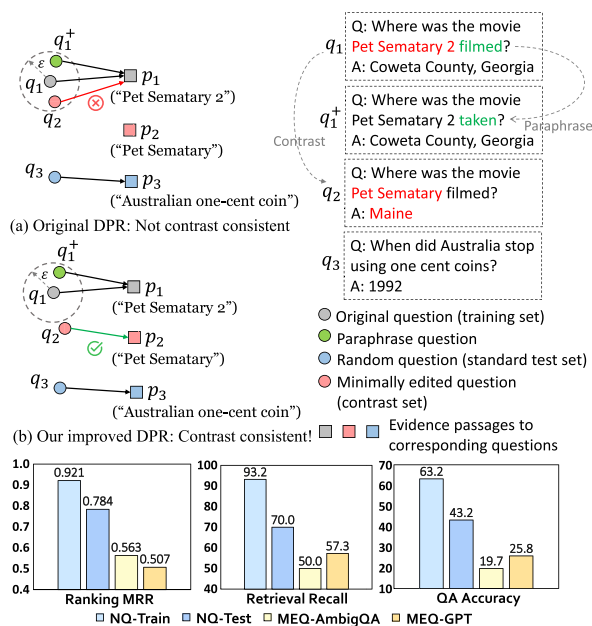
▢ NQ-Train ▢ NQ-Test ▢ MEQ-AmbigQA ▢ MEQ-GPT

Figure 1: Above: Trained on question q_1 but not a contrast one q_2, DPR generated an overly similar embedding of q_2 with q_1's and thus falsely retrieved p_1. We aim to identify q_2 as a distinct question and retrieve p_2 instead. Below: The performance of DPR-based OpenQA models on the standard NQ question set and our contrast set of minimally edited questions (MEQs).

example, in Figure 1, changing ''Pet Sematary 2'' to ''Pet Sematary'' generates an MEQ that resembles the original question but has a distinct answer (''Coweta County, Georgia''→ ''Maine''). We use the training set of an existing benchmark as the original questions because neural OpenQA models exhibit high performance on them. Thus, we are able to evaluate the models' ability of distinguishing MEQs by measuring their performance on the MEQ contrast set. Specifically, we collect MEQs for training questions in the Natural Questions (NQ) benchmark (Kwiatkowski et al., 2019) from two sources, namely, (1) InstructGPT-based question generation (Ouyang et al., 2022) then crowdsource annotation and (2) the AmbigQA dataset (Min et al., 2020).

We find that the state-of-the-art OpenQA models which employ the dense passage retriever (DPR) (Karpukhin et al., 2020) struggle on our MEQ contrast sets. As shown in Figure 1, DPR-retrieved passages lead to 63% downstream QA accuracy on the training set and 43% on standard test set. However, the accuracy drops to 20%∼25% on our MEQ contrast sets. The

problem lies in the contrastive training process of DPR. The model is trained to optimize question embeddings to be closer to their positive passage embeddings[2] than negative passage embeddings. This paradigm does not provide explicit signals for understanding the relationships between questions, which causes the generated question embeddings to be insensitive to minimal discrepancies. As a result, the model generates overly similar embeddings for the MEQ and the original question, leading to incorrect passage retrieval for the MEQ. In fact, the overlap between the retrieved passages of the original question and those of its MEQ is as high as ∼70%, which reflects DPR's limited ability in distinguishing the questions. To overcome such limitations, it is necessary to complement DPR training with signals on inter-question relationships. Besides building the mapping between questions and passages, DPR needs to know which questions are the same and which are different.

In this pioneering study, we propose a simple and effective method based on a query-side contrastive loss to improve the performance of DPR on MEQs. Specifically, in order to learn inter-question relationships, DPR is trained to distinguish between paraphrase questions and semantically different questions. To achieve this, we obtain synthetic MEQs for training questions from the machine-created QA corpus, PAQ (Lewis et al., 2021), as augmented data. Experiments demonstrate that learning the query-side contrastive loss on the augmented MEQs improves the performance of DPR on contrast sets, without sacrificing its performance on standard open-domain questions in the NQ test set.

2 Related Work

2.1 Open-Domain Question Answering

OpenQA is a task that aims to answer user questions without any specified context, thereby testing the ability of QA systems to retrieve, comprehend, and utilize world knowledge (Zhu et al., 2021). The state-of-the-art approach in OpenQA is a two-stage pipeline, consisting of *evidence retrieval* and *answer prediction* (Chen et al., 2017).

In the evidence retrieval stage, a retriever model finds evidence passages from a large corpus (*e.g.*, Wikipedia) based on their relevance

---

[2]A passage that provides evidence for answering the question is its positive passage, otherwise a negative passage.

to the question. Traditional retrievers like BM25 (Robertson and Zaragoza, 2009) perform lexical matching to measure such relevance scores. Recently, DPR (Karpukhin et al., 2020) revolutionized the field by employing dual BERT (Devlin et al., 2019) encoders to compute embeddings for the question and the passage, respectively. It searches evidence passages based on the inner product of question and passage embeddings. Despite subsequent approaches having sought to improve the architecture of the retriever by using fine-grained question-passage interactions (Khattab and Zaharia, 2020) or enhancing global embedding training (Gao and Callan, 2021), DPR remains the most widely-used model due to its simplicity and efficiency. However, the capability of DPR in distinguishing contrastive information has not been thoroughly studied. In this work, we use MEQs as contrast sets and show that DPR has limited contrast consistency when solving MEQs.

In the answer prediction stage, a reader model encodes and fuses the representations of all passages, then predicts an answer by extracting a span (Kedia et al., 2022), generating a free-form sequence (Izacard and Grave, 2021), or using a hybrid approach (Fajcik et al., 2021). While answer prediction is also challenging on MEQs, our approach mainly focuses on the retrieval part which is the bottleneck of solving the MEQs in OpenQA.

2.2 Contrast Sets

NLP Benchmark datasets are typically composed of *i.i.d.* examples that are randomly divided into training and test sets. Conversely, contrast sets refer to data created from small yet label-changing modifications to the existing examples (Gardner et al., 2020). Such characteristics make contrast sets an ideal testbed for evaluating contrast consistency. For example, Gardner et al. (2020) and Kaushik et al. (2020) employed humans to modify linguistic patterns on tasks like syntactic parsing, relation extraction, and claim verification. On sentiment analysis and language inference tasks, controlled text modification models could automatically generate contrast sets (Wu et al., 2021; Ross et al., 2022). In reading comprehension, rule-based algorithms created contrast sets by replacing the answer with another entity (Longpre et al., 2021; Ye et al., 2021; Li et al., 2022). In video-to-text matching, a pre-trained

T5 model was used to find replacements for verbs and entities in the original caption (Park et al., 2022).

Nevertheless, building contrast sets to evaluate contrast consistency in OpenQA has not been explored yet, where data collection must guarantee the factuality of MEQs. The most relevant work is Paranjape et al. (2022) which automatically generated perturbed questions for data augmentation on QA datasets. However, we focus on collecting challenging MEQs to evaluate model consistency instead of data augmentation. Moreover, their generated questions did not meet the requirements of MEQs. The limited accuracy of the question generation model would lead to lots of noise instead of perfect factuality. Also, their method did not ensure the minimality of edits. Therefore, their generated data cannot be used as challenging contrast sets to evaluate contrast consistency in OpenQA.

3 Task: Contrast Consistency on MEQs

3.1 Problem Formulation

In this work, we study minimally edited questions (MEQ) as challenging contrast sets in OpenQA. Suppose we have two questions q and q' with answers a and a', respectively, where q is the original question in the training set and q' is an MEQ of q. In this study, the minimality of edits is measured in two aspects: lexical distance $d_\ell(q, q')$ and semantic distance $d_s(q, q')$. That is to say, q' needs to satisfy $d_\ell(q, q') \leq \epsilon_\ell$, $d_s(q, q') \leq \epsilon_s$ and $a' \neq a$, where ϵ_ℓ and ϵ_s are distance thresholds.

3.2 Evaluation Metrics

To evaluate DPR on MEQ contrast sets, we consider metrics on both ranking and retrieval evaluation. Additionally, we run end-to-end QA experiments using the passages retrieved by DPR.

Ranking evaluation measures the model's ability to differentiate a positive passage from negative passages, by ranking a set of candidate passages based on the relevance score to the question. We collect 50 candidates for each question, including a positive passage, 30 hard negative passages, and 19 random negative passages. Hard negatives are the top-ranked passages in BM25

retrieval that do not contain the answer. We report Mean Rank (MR) and Mean Reciprocal Rank (MRR) of the positive passage.

Retrieval evaluation tests the model's ability to retrieve passages relevant to answering the question from a large corpus. Our retrieval corpus contains $\sim$21M passages from Wikipedia. We calculate Recall@k, the number of passages containing the answer in top-k retrieved passages.

End-to-end QA evaluation checks whether the retrieved passages contain useful information for predicting the correct answer. The retrieved passages are fed into a Fusion-in-Decoder (FiD) reader (Izacard and Grave, 2021) trained on NQ. We calculate Exact Match between model predictions and answers.

4 Data: MEQ Contrast Sets

4.1 Dataset Construction

Based on the above evaluation metrics, we collect two MEQ contrast sets to evaluate models' contrast consistency. The first set, referred to as MEQ-GPT, is generated using InstructGPT (Ouyang et al., 2022) then manually filtered and annotated with answers by crowdsource workers. The second set, named MEQ-AmbigQA, is sourced from the AmbigQA dataset (Min et al., 2020). The construction of our contrast sets consists of four phases: question collection, MEQ filtering, answer annotation, and evidence passage annotation.

4.1.1 Collection of Candidate MEQs

MEQ-InstructGPT Generating answerable MEQs is very difficult for crowdsource workers who are not domain experts. It is hard for them to determine which modifications to the original question result in an answerable MEQ without extensive Internet searches. However, recent GPT-3 models have demonstrated their ability to possess vast amount of knowledge through massive pre-training (Brown et al., 2020). Therefore, we first utilize the InstructGPT model (*text-davinci-002*) to generate a set of MEQ candidates, and leave the answer annotation task to crowdsource workers. The input to InstructGPT is of the form $[I, x_1, \cdots, x_t, q, a]$, where I is the instruction ''Generate a similar question that has a different answer''.

$\{x_i\}_{i=1}^{t}$ are in-context demonstrations that are manually created, where each x_i is a tuple $[q_i, a_i, q_i', a_i']$ (q_i' is the MEQ of q_i). The original question q and answer a are appended to the input, prompting InstructGPT to generate a new question q' and its answer a' to complete the sequence. For each input q, we sample 10 completions from InstructGPT to generate a set of candidate MEQs.

MEQ-AmbigQA The AmbigQA dataset initially targeted a subset of NQ consisting of ambiguous questions. The dataset was introduced to decompose each ambiguous question into multiple disambiguated questions, each of which is a slight modification of the original question. For each NQ question covered in AmbigQA, its corresponding disambiguated questions are considered as its candidate MEQs and are delivered to the subsequent filtering phase (§4.1.2). However, such questions are limited as we set strict criteria for MEQs, so we need more data generated by InstructGPT for solid evaluation.

4.1.2 MEQ Filtering

To build challenging contrast sets, a series of criteria are applied to eliminate unqualified candidates and select MEQs based on the definition in §3.1.

1. **Quality control**: We do not allow q and q' to differ in question words (*e.g.*, how, what), or if the only word that q' adds to q falls into {first, last, new, next, original, not}. We have found that InstructGPT frequently adds these words to create MEQs, but they usually lead to unanswerable questions.

2. **Lexical distance**: Word-level edit distance is used as $d_\ell(q, q')$, and we remove q' if $d_\ell(q, q') = 0$ or $d_\ell(q, q') > 3$.

3. **Semantic distance**: The cosine similarity of semantic embeddings is used to measure $d_s(q, q')$. We remove q' if $\cos(\mathbf{h}_q, \mathbf{h}_{q'}) < 0.95$ which indicates non-negligible semantic discrepancy. The semantic embedding $\mathbf{h}$ should be generated by a sentence embedding model. Here we use the question encoder of the unsupervised dense retrieval model Contriever (Izacard et al., 2021).

1085

4. **Paraphrase filtering**: q' is discarded if it is determined to be a paraphrase to q by a paraphrase detection model. Here we use a RoBERTa-large (Liu et al., 2019) fine-tuned on the Quora Question Pairs dataset (Wang et al., 2019) for paraphrase classification.

5. **Answer difference**: q' is discarded if $a' = a$. For AmbigQA questions, since they are originally human-annotated, we ask human volunteers to check whether a' and a are aliases to the same entity. For GPT-generated questions, the inspection of answer difference is included in the answer annotation process, which we will elaborate in §4.1.3.

Among GPT-generated questions, for a certain original question q, there may be multiple MEQ candidates that pass the above filtering. In such cases, the question that is generated most frequently across 10 samples is selected as the most confident MEQ by InstructGPT. This is similar to the self-consistency idea in Wang et al. (2022).

4.1.3 Answer Annotation

Due to the limited accuracy of InstructGPT in directly answering open-domain questions (Yu et al., 2023), we recruit crowdsource workers to annotate the answer of each candidate MEQ generated by InstructGPT. Before human annotation, we first check the answer generated by InstructGPT via Google Search. If Google Search returns a highlighted answer box which matches the InstructGPT-generated answer, we skip the subsequent human labeling step. For the remaining questions, we recruit human annotators from Surge AI[3] for data labeling. We ask them the following questions:

Q1. Is q' a good variation to q? Bad variations include being unanswerable or having the same answer with q, and are discarded from our dataset.

Q2. If q' is deemed a good variation, find the answer a' using search engines. If necessary, the question may have multiple answers.

Quality Control To ensure answer correctness, each question is answered by two different annotators. If the annotators disagree on the answer

or if either annotator determines the question is an bad variation, the question is discarded. Since the answers are free-form responses, we manually check whether the answers given by two annotators are aliases to the same entity. If the response of the first annotator matches exactly with the answer provided by InstructGPT, we do not recruit a second annotator to reduce costs.

4.1.4 Gold Evidence Passages

As mentioned in §3.2, ranking evaluation on MEQs needs gold evidence passages as positive examples, so we collect them from Wikipedia for our contrast sets. For MEQ-AmbigQA, we utilize the semi-oracle evidence documents[4] provided by the original authors, dividing them into 100-word passages. Then, we identify the first passage that contains the gold answer. For MEQ-GPT, our initial step involves finding candidate evidence passages that include the gold answer. This is achieved by retrieving Wiki passages with BM25 and selecting the top 3 passages that contain the answer. Next, we recruit human annotators from Surge AI to assess whether any of these passages provide sufficient evidence for answering the question. The highest-ranked passage that passed human annotation is chosen as the gold evidence passage. Finally, both contrast sets have a subset of questions paired with a corresponding gold evidence passage.

4.2 Dataset Analysis

The full dataset is composed of 3,343 MEQs (2,293 from InstructGPT and 1,050 from AmbigQA). Each of these MEQs has its original question in the NQ training set. Among them, 1,229 (53.6%) InstructGPT questions and 625 (59.5%) AmbigQA questions are paired with a gold evidence passage from Wikipedia. We use this subset in ranking evaluation and the full set in retrieval and end-to-end QA evaluation.

Data Statistics We summarize basic statistics of the MEQ contrast sets compared to the original NQ questions. As shown in Table 1, MEQ-GPT is similar to NQ regarding the average length of questions and answers. Questions in MEQ-AmbigQA are longer because the original AmbigQA annotators usually added conditions to

---

[3]https://www.surgehq.ai.

[4]https://github.com/shmsw25/AmbigQA/blob/main/evidence.md.

Statistics	NQ		MEQ	
	Train	Test	AmbigQA	GPT
Size	79,168	3,610	1,050	2,293
With Gold Passage	58,880	1,766	625	1,229
Question Length	9.17	9.22	10.73	9.69
Answer Length	2.16	2.22	2.62	1.96
#Answers	1.22	1.79	1.47	1.18
Edit Distance	9.10	9.16	**2.39**	**1.18**
Semantic Similarity	30.12	29.87	**96.47**	**97.96**

Table 1: Dataset statistics. Question lengths, answer lengths, and edit distances are all measured in *words*. Semantic similarity is computed by Contriever (Izacard et al., 2021). For NQ-train and NQ-test, edit distance and semantic similarity are computed between random question pairs. For MEQ contrast sets, they are computed between the original question and its MEQ.

disambiguate the original NQ questions. Besides, AmbigQA does not impose a limit on the answer length, while we limit each answer in MEQ-GPT to at most 5 words, consistent with NQ. The number of answers per question is lower in MEQ-GPT than in MEQ-AmbigQA, because most answers are obtained through strict text matching on candidate answers from two sources. In addition, we observe that MEQ-GPT has a smaller edit distance and higher semantic similarity between q and q', making it hard for models to distinguish them.

Types of Edits We review and categorize different types of minimal edits that are used to create MEQs. Since MEQ-AmbigQA primarily consists of edits that add specifications to the original NQ question, we consider MEQ-GPT as a more natural representation of minimal edits. As shown in Table 2, the edits in MEQ-GPT involve nouns (28.0%), verbs (18.5%), adjectives (18.2%), numbers (14.2%), ordinals (9.2%), dates (6.6%), prepositions/conjunctions (2.9%), and others (2.4%). A word cloud of the edited words is given in Figure 2. We also observe that 22.5% of the total edits are antonym edits where a word in the original question is replaced by its antonym. Our dataset of diverse MEQs provides a comprehensive evaluation of contrast consistency.

4.3 Challenges of MEQ Contrast Sets

The collected MEQ contrast sets are challenging for the widely-used DPR-based OpenQA sys-

tem, although these perturbed questions are only minimal edits to the well-learned training questions. As shown in Figure 1, the model significantly underperforms on the contrast sets, where the passage ranking score of DPR decreases by 39% and 45% compared to NQ-train, and by 29% and 18% compared to NQ-test. This makes a substantial impact on the QA performance, with the accuracy being 69% and 60% lower on the two contrast sets compared to NQ-train, and 54% and 40% lower than NQ-test. The results show that the collected MEQs are much harder to solve than random test questions, which indicates our contrast sets can serve as testbeds for evaluating the contrast consistency of OpenQA.

5 Method: Training DPR with Query-Side Contrastive Loss

5.1 Preliminary: DPR

As a dense retriever, DPR includes a question encoder $E_Q(\cdot)$ and a passage encoder $E_P(\cdot)$. Both encoders map the input sequence to a dense vector as its semantic representation. The relevance score $s(q, p)$ between a question q and a passage p is defined as the dot product of their representations:

$$s(q, p) = E_Q(q)^\mathsf{T} E_P(p)$$

DPR is trained via a contrastive loss. Given a positive passage p^+ and a set of negative passages $\{p_i^-\}_{i=1}^n$ to a certain question, the model is trained to maximize the relevance score between q and p^+, while minimizing the relevance score between q and each p_i^-. The loss function is:

$$\mathcal{L}_{QP} = -\log \frac{\exp(s(q, p^+))}{\exp(s(q, p^+)) + \sum_{i=1}^n \exp(s(q, p_i^-))}$$

The above training paradigm works well on retrieving passages for random test questions, but does not perform as effectively on MEQ contrast sets, as discussed in §1 and §4.3. The training loss $\mathcal{L}_{QP}$ does not provide explicit signals for DPR to learn the relationships between questions. As a result, the question embeddings are insensitive to minimal discrepancies, which prevents the model from identifying the MEQ as a distinct question after seeing the original question in training. This causes DPR to generate an overly similar embedding for the MEQ, leading to a high
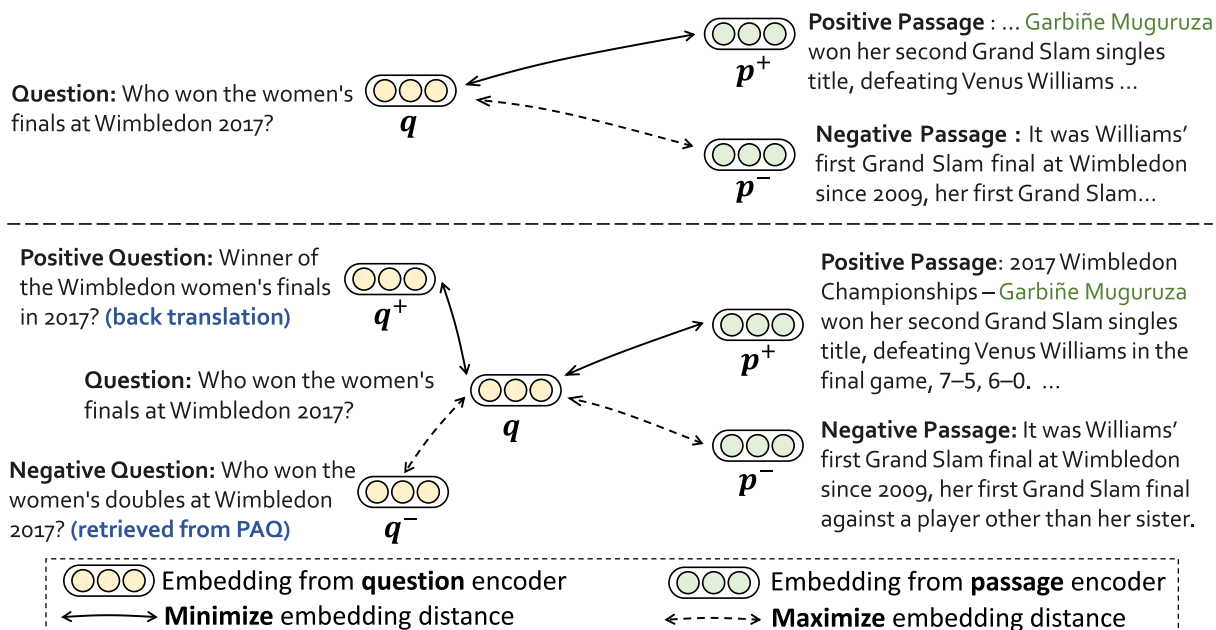
1087

Edit Type	Proportion	Antonym Edits	Example
Nouns	641(28.0%)	151 (24%)	Q: Who wrote the music for the national anthem? A: John Stafford Smith Q: Who wrote the lyrics for the national anthem? A: Francis Scott Key
Verbs	425 (18.5%)	176 (41%)	Q: When did Australia stop using one cent coins? A: 1992 Q: When did Australia start using one cent coins? A: 1966
Adjectives	418 (18.2%)	146 (35%)	Q: How many islands are in Andaman and Nicobar? A: 572 Q: How many inhabited islands are in Andaman and Nicobar? A: 37
Numbers	326 (14.2%)	–	Where did season 2 of Jersey Shore take place? A: Miami Beach, Florida Q: Where did season 3 of Jersey Shore take place? A: Seaside Heights, New Jersey
Ordinals	211 (9.2%)	30 (14%)	Q: Highest scoring NBA players of all time in one game? A: Wilt Chamberlain Q: Second highest scoring NBA players of all time in one game? A: Kobe Bryant
Dates	152 (6.6%)	–	Q: Who ruled the Holy Roman Empire in 1509? A: Maximilian I Q: Who ruled the Holy Roman Empire in 1519? A: Charles V
Prepositions Conjunctions	66 (2.9%)	14 (21%)	Q: Where did the Titanic make its maiden voyage from? A: Southampton Q: Where did the Titanic make its maiden voyage to? A: New York

Table 2: Different MEQ edit types in MEQ-GPT with their proportions of antonym edits and examples. The remaining 2.4% of the instances are of miscellaneous types. The first line in each example is the original question and the second line is the MEQ. Words in green and red are the deleted and added words, respectively.

Figure 2: Word cloud of the edited words. Words in green and red are the deleted and added words, respectively. Larger font sizes indicate higher frequencies.

overlap in the retrieved passages and low contrast consistency.

5.2 Proposed Method

We propose to improve the contrast consistency of DPR by introducing a query-side contrastive loss to distinguish between paraphrase questions and MEQs which are positive and negative question examples for an original question, respectively. We devise a data augmentation approach to collect synthetic question examples to train this loss.

5.2.1 Data Augmentation

For a training question q, its positive example q^+ is a synthetic paraphrase question which is slightly different from q and has the same answer; its negative question q^- is a synthetic MEQ with a different answer.

To obtain q^+, we leverage back translation provided by the nlpaug[5] package. The original question q is translated to another language and then translated back to produce a new phrasing of q. We used translation models of 6 languages provided by Ng et al. (2019) and Tiedemann and Thottingal (2020). Questions that are identical to q (*i.e.*, edit distance = 0) or classified as ''not paraphrase'' by the paraphrase detection model used in §4.1.2 are eliminated. The remaining questions constitute a candidate set of positive questions from which a random q^+ is sampled in each epoch.

To obtain q^-, synthetic MEQs are retrieved from the machine-built QA corpus PAQ (Lewis et al., 2021). All questions in PAQ that are similar to q are retrieved by the question retriever in the work of PAQ. Then, the MEQ requirements specified in §4.1.2 are applied to filter the retrieved synthetic questions. The remaining questions constitute a candidate set of negative questions from which a random q^- is sampled in each epoch.

Apart from learning the relationships among q, q^+, and q^-, the loss $\mathcal{L}_{QP}$ can be augmented to learn the relevance between synthetic questions and their corresponding passages. Because q^+ is a paraphrase question mapping the passages of q, it does not have to be involved in $\mathcal{L}_{QP}$. To

---

[5] https://github.com/makcedward/nlpaug.

1088

Figure 3: Above: the original contrastive training of DPR. Below: our improved DPR with the query-side contrastive loss, where q^+ and q^- are obtained through data augmentation.

train on q^-, its positive passage is the Wikipedia passage that was used to generate the question during the construction of PAQ; its negative passages are collected from the top-ranked passages retrieved by BM25 which do not contain the answer.

5.2.2 Model Training

To provide more supervision signals and prevent overfitting, we randomly sample q^+, q^-, and p^- for each training question q in each epoch. This means that while the original training questions remain fixed, a different set of augmented questions is used. For explicit supervision on inter-question relationships, given q, DPR is trained to assign higher relevance scores to its paraphrase question (q^+) and lower relevance scores to its MEQ (q^-). The relevance score of any pair of questions (q_1, q_2) is calculated as the inner product of their embeddings: $s(q_1, q_2) = E_Q(q_1)^{\intercal} E_Q(q_2)$. Specifically, we consider three forms of query-side constrastive loss functions in experiments:

(1) InfoNCE Loss (van den Oord et al., 2018), which differentiates the positive question from a set of m negative questions. Besides the synthetic MEQ which is considered as a hard negative, the other questions in the same batch are included as random negatives. The loss function is:

$$\mathcal{L}_{QQ} = -\log \frac{\exp(s(q, q^+))}{\exp(s(q, q^+)) + \sum_{j=1}^{m} \exp(s(q, q_j^-))}.$$

(2) Dot Product Loss, which directly penalizes the relevance score between a sample question q and its augmented MEQ:

$$\mathcal{L}_{QQ} = s(q, q^-).$$

(3) Triplet Loss (Schroff et al., 2015), which trains the model to assign a higher relevance score to q^+ compared to q^-, enfored by a margin α:

$$\mathcal{L}_{QQ} = \max\left(0, \alpha - s(q, q^+) + s(q, q^-)\right).$$

The final training loss of our improved DPR is $\mathcal{L} = \mathcal{L}_{QP} + \lambda \mathcal{L}_{QQ}$, where the hyperparameter λ weights the trade-off between the loss terms.

6 Experiments

In experiments, we compare our proposed training method against the original training setting of DPR. After training the models on the NQ training set, we test them on the standard NQ test set

Model	Augmentation	NQ		MEQ-AmbigQA		MEQ-GPT	
		MR↓	MRR↑	MR↓	MRR↑	MR↓	MRR↑
DPR$_{BASE}$	None	2.36	0.784	5.09	0.563	5.44	0.507
DPR$_{BASE}$	Random	2.36	0.781	5.09	0.557	5.25	0.524
DPR$_{BASE}$	MEQs	2.34	0.783	5.09	0.543	5.10	0.529
DPR$_{BASE}$	MEQs + $\mathcal{L}_{QQ}$	**2.25**	**0.791**	**4.85**	**0.569**	**4.88**	**0.547**
DPR$_{LARGE}$	None	2.31	0.780	4.84	0.569	5.46	0.515
DPR$_{LARGE}$	Random	2.20	0.797	4.98	0.554	5.18	0.533
DPR$_{LARGE}$	MEQs	2.17	0.797	4.79	0.561	5.00	0.544
DPR$_{LARGE}$	MEQs + $\mathcal{L}_{QQ}$	**2.14**	**0.804**	**4.59**	**0.592**	**4.61**	**0.565**

Table 3: Ranking evaluation results. MR and MRR stand for mean rank and mean reciprocal rank, respectively. A lower MR or higher MRR indicates better performance. BM25 is not listed because sampling hard negatives from top-ranked passages in BM25 retrieval lowers the ranking performance of BM25 in return.

as well as two MEQ contrast sets that we collected in this work.

6.1 Models

We augment the training set with M = 33k synthetic MEQs and train DPR with both $\mathcal{L}_{QP}$ and $\mathcal{L}_{QQ}$. We consider the following baselines:

- **Vanilla DPR**. This is the original training setting of DPR, proposed by Karpukhin et al. (2020). The model is trained only with $\mathcal{L}_{QP}$ on the standard NQ training set.

- **DPR with random augmented questions**. This model is trained only with $\mathcal{L}_{QP}$, but we add M random synthetic questions from PAQ to the training set. This is to rule out the effect of simply adding more synthetic data.

- **DPR with augmented MEQs**. This model uses the same set of M synthetic MEQs retrieved from PAQ as data augmentation, but is trained only with $\mathcal{L}_{QP}$. We use this variant to test if $\mathcal{L}_{QQ}$ is necessary in model training.

Additionally, we test the performance of BM25 on retrieval as a reference. Recent research has shown that larger retrievers may exhibit better generalization (Ni et al., 2022). Therefore, in addition to the standard DPR which is built on BERT-Base, we use BERT-Large as the backbone

model to see: (1) whether MEQ contrast sets are still challenging for larger models and (2) whether our training method is still effective for larger models. We name the smaller model and larger model DPR$_{BASE}$ and DPR$_{LARGE}$, respectively.

We use the same set of basic hyper-parameters for each DPR model: a learning rate of 10^{-5}, a batch size of 64 (32 for DPR$_{LARGE}$), 40 training epochs with 5% warmup steps. On ranking evaluation, our best setting uses the InfoNCE loss with $\lambda = 0.5$. On retrieval and QA evaluation, our best setting uses the dot product loss with $\lambda = 0.03$. Since we do not have a dev set for MEQs,[6] we conduct ranking evaluation on MEQ contrast sets in a dev setting, where we select the highest score among all checkpoints. Then we use the checkpoint with the best ranking score to test its retrieval and QA performance. The scores on NQ-test is reported using the best checkpoint on NQ-dev.

6.2 Results

Experimental results on three datasets (NQ-test, MEQ-AmbigQA, MEQ-GPT) are presented from Table 3 to Table 6. We have the following findings:

(1) Our proposed method improves DPR's ability to distinguish MEQs. As shown in Tables 3

---

[6]We empirically found that model performance on NQ-dev is inconsistent with MEQ contrast sets.

Model	$\mathcal{L}_{QQ}$	AmbigQA		GPT	
		MR↓	MRR↑	MR↓	MRR↑
DPR$_{BASE}$	InfoNCE	4.85	0.569	**4.88**	**0.547**
DPR$_{BASE}$	Dot Product	**4.79**	**0.574**	4.98	0.539
DPR$_{BASE}$	Triplet	4.80	0.568	4.91	0.542
DPR$_{LARGE}$	InfoNCE	4.76	0.572	4.63	0.570
DPR$_{LARGE}$	Dot Product	**4.59**	**0.592**	4.61	0.565
DPR$_{LARGE}$	Triplet	4.61	0.582	**4.59**	**0.573**

Table 4: Ranking evaluation with different $\mathcal{L}_{QQ}$ functions on two MEQ contrast sets. All loss functions outperform the baselines in Table 3.

and 5, on passage ranking and passage retrieval, the DPR trained with query-side contrastive loss outperforms the vanilla DPR on both contrast sets, showing improved contrast consistency on MEQs. This improvement is consistent across models of different sizes. For example, on MEQ-GPT, our model improves the vanilla DPR by 8% and 10% on ranking MRR for base and large versions, respectively. On the choice of $\mathcal{L}_{QQ}$, Table 4 demonstrates that all three loss functions improve performance over baselines, while the optimal setting may require tuning on the specific dataset.

(2) The query-side contrastive loss contributes the most to the improved contrast consistency. Although synthetic MEQs themselves bring more training signals, the model cannot consistently outperform the vanilla DPR without $\mathcal{L}_{QQ}$. Actually, its performance is sometimes even lower than DPR. In contrast, after including the query-side contrastive loss, we observe consistent improvements across all datasets, as shown in Tables 3 and 5. For example, on MEQ-AmbigQA, simply adding synthetic MEQs into the training set gives 12% lower recall@1 than the vanilla DPR, while training with $\mathcal{L}_{QQ}$ outperforms the naive augmentation method by 18%.

(3) The improvement does not simply come from the increased number of training data. There is no significant difference on the performance between DPR augmented with random synthetic questions (''Random'' in ''Augmentation'' column) and the original DPR (''None'' in the column) in Tables 3, 5, and 6. The average improvement of inserting random synthetic ques-

tions on all metrics is only 0.2% for DPR$_{BASE}$ and 1.6% for DPR$_{LARGE}$, which indicates simply adding more synthetic data is not an effective solution.

(4) Improved retrieval performance leads to higher end-to-end QA accuracy. As shown in Table 6, our improved DPR provides more relevant information for answer prediction on MEQs. Even using only 1 retrieved passage, our improved DPR-Large outperforms its vanilla version by 12% and 11% on two contrast sets, respectively.

(5) Our method does not sacrifice performance on standard test questions. After joint training with the query-side contrastive loss and augmented with synthetic MEQs, our model still maintains its competitive performance on the standard NQ test set. Specifically, It outperform all baselines in ranking evaluation (see Table 3), while performing on par with the best baseline in retrieval and QA scores (see Tables 5 and 6).

Summary: The results are consistent across ranking, retrieval, and end-to-end QA experiments, which demonstrates the solidity of the above findings. Nevertheless, the performance of DPR still has a long way to improve, and such a gap is observed in both base and large versions of the model. Notably, DPR models perform significantly worse on MEQ contrast sets than the standard test set, even though it is trained under a development setting. This suggests that further research is still necessary to improve the contrast consistency of retrieval models on MEQs.

6.3 Analysis

Passage Overlap One of the indications that DPR lacks the ability to distinguish the original question and its MEQ is the high overlap between the passages retrieved for each. Figure 4 illustrates that both synthetic data augmentation and the query-side contrastive loss can reduce passage overlap. The synthetic MEQ augmentation helps to train the question embeddings of MEQs closer to their positive passages. Moreover, the query-side contrastive loss explicitly trains the model to distinguish the original question and its MEQ apart. Nevertheless, a lower passage overlap does not always indicate better performance. For instance, our model with the dot product loss

Model	Augmentation	NQ			MEQ-AmbigQA			MEQ-GPT		
		R@1	R@5	R@20	R@1	R@5	R@20	R@1	R@5	R@20
BM25	None	23.2	45.3	64.5	16.8	34.2	48.8	21.1	42.7	61.7
DPR_{BASE}	None	46.6	70.0	81.2	28.5	50.0	65.6	31.5	57.3	73.2
DPR_{BASE}	Random	**48.2**	**71.2**	81.6	27.5	49.2	65.8	31.8	58.0	73.8
DPR_{BASE}	MEQs	46.4	69.9	81.3	25.2	46.2	62.7	31.5	55.9	72.3
DPR_{BASE}	MEQs + $\mathcal{L}_{QQ}$	48.1	70.8	**81.9**	**29.5**	**52.3**	**66.4**	**32.8**	**58.7**	**74.4**
DPR_{LARGE}	None	46.0	67.6	80.3	26.8	49.2	64.0	29.2	54.9	70.7
DPR_{LARGE}	Random	49.0	70.9	81.5	26.2	48.4	64.1	31.3	56.7	72.3
DPR_{LARGE}	MEQs	48.0	70.5	81.4	27.7	47.2	61.5	31.2	57.0	71.8
DPR_{LARGE}	MEQs + $\mathcal{L}_{QQ}$	**51.0**	**71.2**	81.6	30.1	52.3	65.4	32.5	58.4	73.1

Table 5: Retrieval evaluation results. R@k stands for Recall@k.

Model	Augmentation	NQ			MEQ-AmbigQA			MEQ-GPT		
		1P	5P	20P	1P	5P	20P	1P	5P	20P
BM25	None	16.4	28.4	37.3	10.9	15.2	18.1	13.3	20.5	25.8
DPR_{BASE}	None	32.6	43.2	49.1	14.0	19.7	21.9	17.6	25.8	29.3
DPR_{BASE}	Random	33.7	**44.8**	**49.4**	14.7	19.1	22.4	16.8	25.4	29.5
DPR_{BASE}	MEQs	32.0	43.4	48.7	13.5	19.3	**23.1**	17.1	25.5	29.5
DPR_{BASE}	MEQs + $\mathcal{L}_{QQ}$	**34.4**	44.7	49.2	**16.6**	**21.8**	22.8	**19.5**	**26.7**	**31.1**
DPR_{LARGE}	None	31.4	42.2	47.9	14.3	19.2	21.4	16.1	24.6	29.1
DPR_{LARGE}	Random	**33.7**	44.6	**49.3**	13.4	20.4	21.5	17.3	25.5	29.4
DPR_{LARGE}	MEQs	33.0	**44.7**	48.7	15.7	19.3	21.7	17.4	25.0	29.1
DPR_{LARGE}	MEQs + $\mathcal{L}_{QQ}$	**33.7**	44.6	**49.3**	16.1	22.1	23.0	19.4	27.6	31.6

Table 6: End-to-end QA results (Exact Match). 1P, 5P, and 20P are the number of passages read by the FiD reader.
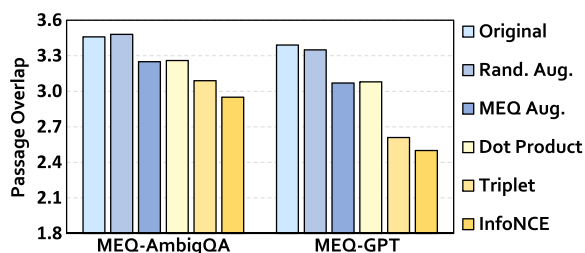
Figure 4: Overlap in top-5 retrieved passages between the original training question and its MEQ.

Figure 5: The ratio of successful MEQ identifications of different models on contrast sets, with paraphrase questions as distractors.

does not have the lowest passage overlap, but performs the best in retrieval evaluation.

Identification of Inter-question relationships
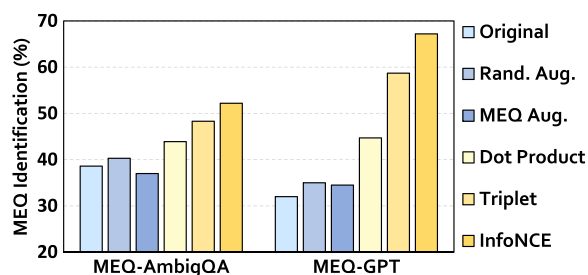To further analyze model behavior after the query-side contrastive training, we test the models' ability to distinguish inter-question relationships. A model is considered successful in identifying the MEQ if the generated embedding of the original

question is closer to its paraphrase question rather than its MEQ. The paraphrase questions are separately generated using InstructGPT to avoid conflict with those used in data augmentation. As shown in Figure 5, training with the query-side contrastive loss leads to an improved ability to distinguish between paraphrase questions and different questions, which indicates our models are better at identifying inter-question relationships. The model trained with InfoNCE loss has the highest success rate in identifying inter-question relationships, because it received more training signals from a positive example and a set of negative examples than those with other types of loss.

7 Conclusion

In this study, we addressed the gap in research on contrast consistency in OpenQA by collecting MEQs as challenging contrast sets to the popular NQ benchmark. Our findings reveal that DPR lacks contrast consistency on our contrast sets. To address this limitation, we introduced a query-side contrastive loss with the aid of data augmentation, which improved its ability to recognize inter-question relationships. Overall, our findings and data can pave the way for further exploring the role of contrast consistency in developing robust and effective OpenQA systems.

Acknowledgments

References

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020.*

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017.* https://doi.org/10.18653/v1/P17-1171

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, NAACL-HLT 2019.* https://doi.org/10.18653/v1/N19-1423

Martin Fajcik, Martin Docekal, Karel Ondrej, and Pavel Smrz. 2021. R2-D2: A modular baseline for open-domain question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 854–870, Punta Cana, Dominican Republic. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.findings-emnlp.73

Luyu Gao and Jamie Callan. 2021. Condenser: A pre-training architecture for dense retrieval. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021*, pages 981–993, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.emnlp-main.75

Matt Gardner, Yoav Artzi, Victoria Basmova, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hannaneh Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, Sameer Singh, Noah A. Smith, Sanjay Subramanian, Reut

Tsarfaty, Eric Wallace, Ally Zhang, and Ben Zhou. 2020. Evaluating models' local decision boundaries via contrast sets. In *Findings of the Association for Computational Linguistics: EMNLP*, pages 1307–1323, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.findings-emnlp.117

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021. Towards unsupervised dense information retrieval with contrastive learning. *ArXiv preprint*, 2112.09118. https://doi.org/10.48550/arXiv.2112.09118

Gautier Izacard and Edouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021*. https://doi.org/10.18653/v1/2021.eacl-main.74

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick S. H. Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020*, pages 6769–6781, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.550

Divyansh Kaushik, Eduard H. Hovy, and Zachary Chase Lipton. 2020. Learning the difference that makes a difference with counterfactually-augmented data. In *8th International Conference on Learning Representations, ICLR 2020*.

Akhil Kedia, Mohd Abbas Zaidi, and Haejun Lee. 2022. Fie: Building a global probability space by leveraging early fusion in encoder for open-domain question answering. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022*.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2020*, pages 39–48. Association for Computational Linguistics. https://doi.org/10.1145/3397271.3401075

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. 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. https://doi.org/10.1162/tacl_a_00276

Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*.

Patrick S. H. Lewis, Yuxiang Wu, Linqing Liu, Pasquale Minervini, Heinrich Küttler, Aleksandra Piktus, Pontus Stenetorp, and Sebastian Riedel. 2021. PAQ: 65 million probably-asked questions and what you can do with them. *Transactions of the Association for Computational Linguistics*, 9:1098–1115. https://doi.org/10.1162/tacl_a_00415

Daliang Li, Ankit Singh Rawat, Manzil Zaheer, Xin Wang, Michal Lukasik, Andreas Veit, Felix X. Yu, and Sanjiv Kumar. 2022. Large language models with controllable working memory. *ArXiv preprint*, 2211.05110. https://doi.org/10.48550/arXiv.2211.05110

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*, 1907.11692. https://doi.org/10.48550/arXiv.1907.11692

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021. Entity-based knowledge conflicts in question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021*, pages 7052–7063, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.emnlp-main.565

Sewon Min, Julian Michael, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2020. Ambigqa: Answering ambiguous open-domain questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020*, pages 5783–5797, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.emnlp-main.466

Nathan Ng, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. 2019. Facebook fair's WMT19 news translation task submission. In *Proceedings of the Fourth Conference on Machine Translation, WMT 2019*, pages 314–319, Florence, Italy. Association for Computational Linguistics. https://doi.org/10.18653/v1/W19-5333

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernández Abrego, Ji Ma, Vincent Y. Zhao, Yi Luan, Keith B. Hall, Ming-Wei Chang, and Yinfei Yang. 2022. Large dual encoders are generalizable retrievers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022*, pages 9844–9855. Association for Computational Linguistics.

Aäron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *Arxiv preprint*, 1807.03748. https://doi.org/10.48550/arXiv.1807.03748

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *ArXiv preprint*, 2203.02155. https://doi.org/10.48550/arXiv.2203.02155

Bhargavi Paranjape, Matthew Lamm, and Ian Tenney. 2022. Retrieval-guided counterfactual generation for QA. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022*, pages 1670–1686, Dublin, Ireland. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.acl-long.117.

Jae Sung Park, Sheng Shen, Ali Farhadi, Trevor Darrell, Yejin Choi, and Anna Rohrbach. 2022. Exposing the limits of video-text models through contrast sets. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022*, pages 3574–3586, Seattle, United States. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.naacl-main.261

Stephen E. Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389. https://doi.org/10.1561/1500000019

Alexis Ross, Tongshuang Wu, Hao Peng, Matthew E. Peters, and Matt Gardner. 2022. Tailor: Generating and perturbing text with semantic controls. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022*, pages 3194–3213, Dublin, Ireland. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.acl-long.228

Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015*, pages 815–823, Boston, MA. https://doi.org/10.1109/CVPR.2015.7298682

Jörg Tiedemann and Santhosh Thottingal. 2020. OPUS-MT - Building open translation services for the world. In *Proceedings of the 22nd*

Annual Conference of the European Association for Machine Translation, EAMT 2020, pages 479–480, Lisboa, Portugal. European Association for Machine Translation.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *7th International Conference on Learning Representations, ICLR 2019*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics. OpenReview.net. `https://doi.org/10.18653/v1/W18-5446`

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *ArXiv preprint*, 2203.11171. `https://doi.org/10.48550/arXiv.2203.11171`

Tongshuang Wu, Marco Túlio Ribeiro, Jeffrey Heer, and Daniel S. Weld. 2021. Polyjuice: Generating counterfactuals for explaining, evaluating, and improving models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021*, pages 6707–6723, Online. Association for Computational Linguistics. `https://doi.org/10.18653/v1/2021.acl-long.523`

Xi Ye, Rohan Nair, and Greg Durrett. 2021. Connecting attributions and QA model behavior on realistic counterfactuals. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021*, pages 5496–5512, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics. `https://doi.org/10.18653/v1/2021.emnlp-main.447`

Wenhao Yu, Dan Iter, Shuohang Wang, Yichong Xu, Mingxuan Ju, Soumya Sanyal, Chenguang Zhu, Michael Zeng, and Meng Jiang. 2023. Generate rather than retrieve: Large language models are strong context generators. In *11th International Conference on Learning Representations, ICLR 2023*.

Fengbin Zhu, Wenqiang Lei, Chao Wang, Jianming Zheng, Soujanya Poria, and Tat-Seng Chua. 2021. Retrieving and reading: A comprehensive survey on open-domain question answering. *ArXiv preprint*, 2101.00774. `https://doi.org/10.48550/arXiv.2101.00774`