SYNTHETIC PRE-TRAINING FOR ROBUSTNESS IN INFORMATION RETRIEVAL

BY

REVANTH GANGI REDDY

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Computer Science
in the Graduate College of the
University of Illinois Urbana-Champaign, 2022

Urbana, Illinois

Adviser:

Professor Heng Ji

ABSTRACT

Research on neural information retrieval has so far been focused primarily on standard supervised learning settings, where it outperforms traditional term matching baselines. Many practical use cases of such models, however, may involve previously unseen target domains. In this thesis, we first improve the out-of-domain generalization of Dense Passage Retrieval (DPR)—a popular choice for neural information retrieval (IR)—through synthetic data augmentation *only in the source domain*. We empirically show that pre-training DPR with additional synthetic data in its source domain (Wikipedia), which we generate using a fine-tuned sequence-to-sequence generator, can be a low-cost yet effective first step towards its generalization. Across five different test sets, our augmented model shows more robust performance than DPR in both in-domain and zero-shot out-of-domain evaluation.

We then show that supervised neural IR models are prone to learning sparse attention patterns over passage tokens, which can result in key phrases including named entities receiving low attention weights, eventually leading to model under-performance. Using a novel targeted synthetic data generation method that identifies poorly attended entities and conditions the generation episodes on those, we teach neural IR to attend more uniformly and robustly to all entities in a given passage. On two public IR benchmarks, we empirically show that the proposed method helps improve both the model's attention patterns and retrieval performance, including in zero-shot settings.

To my parents, for their love and support.

ACKNOWLEDGMENTS

Firstly, I would like to thank my advisor, Professor Heng Ji, who has been an endless source of support and inspiration for me. I admire her desire to push for excellence and am extremely grateful for how she has shaped me as a researcher. I am really looking forward to learning more from her as I continue my PhD at UIUC.

I cannot express how much indebted I feel to the Multi-lingual Natural Language Processing (NLP) group at IBM Research New York, for giving me a chance to get back into research as part of the residency program. Specifically, I would like to thank Dr. Arafat Sultan, Dr. Avirup Sil, Dr. Vittorio Castelli, Dr. Ramon Astudillo, Dr. Radu Florian, Dr. Salim Roukos and other members of the group for their invaluable advice throughout. This work would not have been possible without Arafat and Avi's belief in me and I am grateful to them for continuing to collaborate with me even after the residency. Finally, I cannot thank Arafat enough for his patience and I thoroughly enjoyed all of our brainstorming sessions.

I would also like to thank members of the Blender lab for their constant guidance and help throughout my masters program. I am grateful to have gotten the opportunity to work with such a talented set of peers.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

Information Retrieval (IR) typically involves finding documents relevant to a query from a large corpus. Retrieval systems are a crucial component of many important natural language processing applications such as open-domain question answering, web search, claim verification etc. Broadly, information retrieval can be seen as way in which artificial intelligence can help meet human information needs. The retrieved information can be at different levels of granularity, which can be either document-level or passage-level.

Traditional IR methods involve using TF-IDF matching or BM25 term weighting [1] to retrieve evidence from a corpus. However, since these methods cater specifically to keyword search queries, more recent approaches [2, 3] in information retrieval have leveraged the advent of large pre-trained language models [4, 5] to use dense semantic representations for retrieval. Further, this has been made possible by the availability of tools like FAISS [6], which use specialized indexing schemes to provide highly efficient search in a dense vector space.

Retrieval models can be evaluated along various dimensions. While the relevance of retrieved results is of outmost importance, retrieval models are also expected to demonstrate efficiency and robustness. Efficiency of retrieval can be easily measured in terms of number of queries processed per second, while also accounting for the size of the index which needs to be stored. However, it is not straightforward to understand and measure the robustness of IR models. As suggested by [7], robustness in retrieval models can have various aspects. Some of these include: 1) *Robustness to rare inputs:* Queries and passages can contain terms rarely seen in training, which might require exact matching more than semantic matching; 2) *Robustness to corpus variance:* Corpus distributions at test time can be different from those that the model was trained on, meaning retrieval models cannot be too dependent on corpus specific patterns.

In this regard, one way of measuring robustness is via testing the generalization capabilities of retrieval models across different domains in zero-shot settings. Recent work [8, 9] has shown that neural retrieval models can be limited in their out-of-distribution generalization capabilities and underperform unsupervised lexical matching methods when used in new domains. This is potentially due to a domain shift as documents in such domains can have specific terminologies. Further, given little or no labelled data in such domains, it is difficult to adapt the model to these targeted domains. Hence, it becomes crucial to build general-domain robust retrieval models that can handle human queries out-of-the-box in various domains such as Wikipedia, news, scientific text, etc.

1.1 RESEARCH QUESTIONS

In view of the challenges faced by neural models in IR, this dissertation focuses on the development of neural retrieval models that are robust and can work out-of-the-box in different domains. In this regard, we have explored data augmentation through synthetic pre-training, with additional tailoring built into the data generation strategy to handle specific model deficiencies. We leverage pre-trained transformer models [10] to generate high quality synthetic data, both in terms of diversity [11] and roundtrip consistency [12]. Following are the research questions we aim to answer in this thesis:

- **Can training with more data in the general domain help retrieval models work well across different domains?**: [8] has shown that training with target domain synthetic data can help improve retrieval performance in that domain. Instead, we investigate whether the need to generate such synthetic data for each individual domain can be overcome by leveraging Wikipedia, which contains data from various domains. We aim to explore whether pre-training using large scale synthetic data generated from Wikipedia results in a single robust retrieval system that can work out-of-the-box in multiple domains. More details in Chapter 3.
- **What kind of attention patterns do retrieval models learn and how can we improve them for better robustness?**: Robust retrieval models should be able to encode all important details in a given passage as information relevant to the question could be present anywhere within it. We aim to understand whether neural models are able to capture all the key-phrases in the passages. We investigate how we can add more control into the generation of synthetic data to handle any shortcomings in the attention patterns of such neural retrieval models. More details in Chapter 4.

1.2 THESIS OUTLINE

This thesis is organized as follows:

- Chapter 2 gives a brief background of traditional term-matching and neural information retrieval approaches and the corresponding evaluation metrics for the retrieved results.
- In Chapter 3, we propose an unsupervised data-augmentation strategy that generates synthetic questions from corpus passages. We show that pre-training with such synthetic data makes the neural model more robust, with improvements in zero-shot performance in both near and far domains.

- In Chapter 4, we show that neural models can learn sparse attention patterns, potentially due to biases in training data. We mitigate this issue with an entity-conditioned question generation system, that augments the pre-training data with questions that are specifically about entities in the passages that receive low attentions from the neural model. We show that this strategy helps the model learn attention patterns that are more robust and spread out, which leads to improvement in retrieval performance.
- Chapters 5 and 6 contain the conclusions, along with some directions for future work which can improve robustness of neural models, without the need for generating additional synthetic data.

CHAPTER 2: BACKGROUND

In this section, we briefly introduce some information retrieval approaches, including traditional term-matching approaches, such as TF-IDF (§2.1) and BM25 (§2.2), and neural information retrieval (§2.3). We'll further briefly expand on the different kinds of architectures used for neural IR (§2.3.1) along with some details for the training (§2.3.2) and inference (§2.3.3) phases. Finally, we describe the various metrics used to evaluate retrieval performance (§2.4).

2.1 TF-IDF

TF-IDF is a statistical technique that assesses how pertinent a word is to a document within a collection of documents. It is commonly used in search engines, along with being useful for identifying keyphrases for applications such as text summarization and classification. TF-IDF for a word in a document is computed using the following metrics:

- **Term frequency:** This is usually the number of times a word appears in a document, with more advanced versions adjusting it by the length of the document or by the count of the most frequent word.
- **Inverse document frequency:** This is a measure of how rare a word is in the entire document collection. It is usually calculated by dividing the collection count with the number of documents that contain a word.

Mathematically, the TF-IDF score for a word w in a document d from a document collection D is given as:

$$tf - idf(w, d, D) = tf(w, d) \cdot idf(w, D) \tag{2.1}$$

where the term frequency $tf(w, d)$ and the inverse document frequency $idf(w, D)$ are calculated as follows:

$$tf(w, d) = log(1 + freq(w, d)) \tag{2.2}$$

$$idf(w, D) = log(\frac{N}{count(d \in D : w \in d)}) \tag{2.3}$$

Here, N is the number of documents in the collection and $count(d \in D : w \in d)$ corresponds to the number of documents which contain the word w.

2.2 BM25

BM25, or popularly known as Okapi BM25, is a ranking function used to estimate the relevance of a text document given the search query. BM25 can be considered as a bag-of-words retrieval function, since terms are considered based on whether they are present in the document and where they are present is not of relevance. BM25 is one of the most commonly used first phase ranking function for ranking text documents.

BM25 builds on top of the TF-IDF formulation, with refinements for how the term frequencies and inverse document frequencies are combined into a single score. Mathematically, given a query Q comprising terms $q_1, q_2, ..., q_n$ and a document D, the relevance score of the document is given as:

$$Score(Q, D) = \sum_{i}^{n} IDF(q_i) \frac{f(q_i, D) * (k_1 + 1)}{f(q_i, D) + k_1 \cdot (1 - b + b \cdot \frac{field\_len}{avg\_field\_len})} \tag{2.4}$$

The individual components of this scoring function are:

- $f(q_i, D)$ is the number of times query term q_i occurs in document D.
- $IDF(q_i)$ is the inverse document frequency of the query term q_i. Here this is computed differently compared to the standard TF-IDF formulation:

  $$IDF(q_i) = log(1 + \frac{N - n(q_i) + 0.5}{n(q_i) + 0.5}) \tag{2.5}$$

  where N is the number of documents in the collection and $n(q_i)$ is the number of documents that contain query term q_i.
- $\frac{field\_len}{avg\_field\_len}$ accounts for the length of the document relative to the average document length. Intuitively, since longer documents can have a higher chance of containing query terms, this term lowers the score for documents that have more terms that don't match the query.
- b is a hyper-parameter which controls for document length normalization. Usually, this is set to 0.75 in elastic search.
- k_1 is a hyper-parameter that controls for term-frequency saturation. This is used to limit how much a single query term can influence the final score.

2.3 NEURAL INFORMATION RETRIEVAL

Neural IR involves the use of deep neural networks for ranking search results corresponding to a query. This involves learning a semantic representation of the query and the document in order to

compute the relevance between them. Neural IR involves learning embeddings for terms in both queries and documents, to enable *semantic* matching in the embedding space, as opposed to lexical matching employed by traditional IR methods such as TF-IDF and BM25. In this section, we first introduce two different types of architectures for neural information retrieval (§2.3.1) along with the corresponding training (§2.3.2) and inference (§2.3.3) procedures.

2.3.1 Architecture

Most recent approaches to neural information retrieval use transformer-based [13] architectures for computing the representations of queries and documents. [2, 14, 15] use encoders based on BERT [5] and obtain state-of-the-art performance for retrieval. In this section, we briefly introduce the cross-encoder and dual-encoder based architectures for neural information retrieval, which are typically used for re-ranking and first-step retrieval respectively.

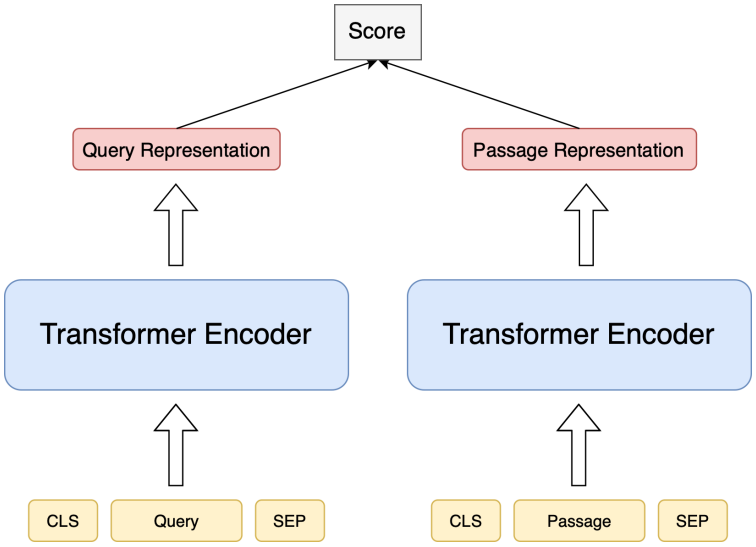


Figure 2.1: Architecture of a dual-encoder model

Dual-Encoder: Given a collection of passages, the dual-encoder model creates an index in a continuous space to retrieve relevant passages, given an input question. The model uses a Siamese neural network [16] with separate encoders $E_Q(.)$ and $E_P(.)$ for the question and passage respectively, as shown in Figure 2.1. The individual encoders are based on BERT [5] and use the final hidden representation of the [CLS] token as the output. In a dual-encoder model, the interaction between the query and the passage occurs only at the final scoring phase, thereby enabling the representations to be computed independently of each other. As we describe later in Section 2.3.3,

this is crucial to ensuring first-stage retrieval using dual-encoder models is efficient. The question-passage similarity is defined as the dot product of their corresponding output representations:

$$sim(q, p) = E_Q(q)^T E_P(p) \tag{2.6}$$

Cross-Encoder: The cross-encoder architecture involves feeding the query and passage as input together into a single transformer-based encoder $E(.,.)$, as shown in Figure 2.2. The model then outputs a score using the final hidden representation corresponding to the [CLS] token. The question-passage similarity is computed as follows:

$$sim(q, p) = W E(q, p) \tag{2.7}$$

Cross-encoders are more powerful than dual-encoders on account of cross-attention between query and passage tokens before computing the final similarity score. However, the score needs to be computed individually for each (query, passage) pair at inference, making it too expensive for full collection retrieval. Thereby, cross-encoders are typically used as re-rankers to improve the performance over first-stage retrieval models (which are usually term-matching methods i.e TF-IDF/BM25 or dual-encoder based models).

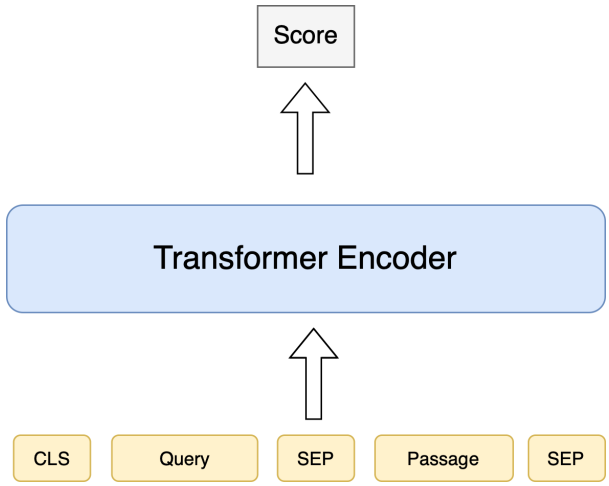


Figure 2.2: Architecture of a cross-encoder model.

2.3.2 Training

In this section, we describe the procedure for training a dual-encoder model. [2] demonstrated that training examples for such a dual-encoder model can be obtained from existing machine reading comprehension datasets. Each training instance $(q_i, p_i^+, p_{i,1}^-, ..., p_{i,n}^-)$ contains a question q_i, one

positive passage p_i^+ and n negative passages $p_{i,j}^-$. The training loss is the negative log-likelihood of the positive passage:

$$L = -\log \frac{e^{sim(q_i, p_i^+)}}{e^{sim(q_i, p_i^+)} + \sum_{j=1}^{n} e^{sim(q_i, p_{i,j}^-)}} \tag{2.8}$$

While negative passages for a given question can be simply sampled from the collection, [2] show that having a top passage returned by BM25 among the negatives helps improve performance. To make the training process more efficient, the trick of in-batch negatives [17, 18] is also used.

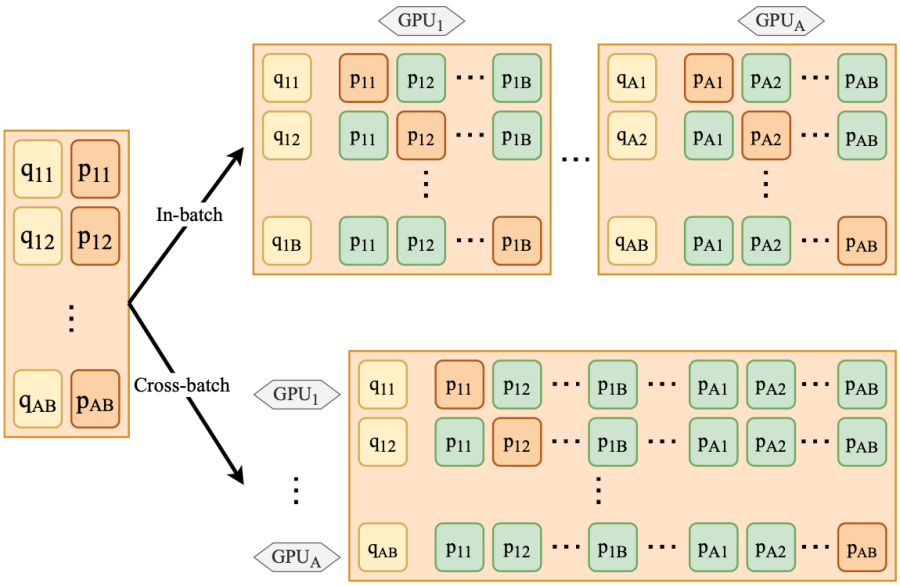


Figure 2.3: Cross-batch negatives used in [19] along with the traditional in-batch negatives. Here, A is the number of GPUs and B is the number of questions in each batch.

For each question in a training mini-batch, the following passages are used as negatives: (1) a passage returned by BM25 that is not labeled positive, (2) positive passages as well as BM25-retrieved negatives for other questions in the mini-batch. Further, [19] show that, when training on multiple GPUs, cross-batch negatives can be used to further optimize the training with more negatives. This is done by sharing the passage embeddings across multiple GPUs, as shown in Figure 2.3.

2.3.3 Inference

In this section, we'll briefly describe the inference procedure for a neural retrieval model that uses a dual-encoder architecture. Recollect that the question-passage similarity in such a model is computed as:

$$sim(q, p) = E_Q(q)^T E_P(p) \tag{2.9}$$

We can see that the above equation is decomposable, on account of the scoring using the representations of the query and the passage from separate encoders. The overall inference procedure is shown in Figure 2.4.

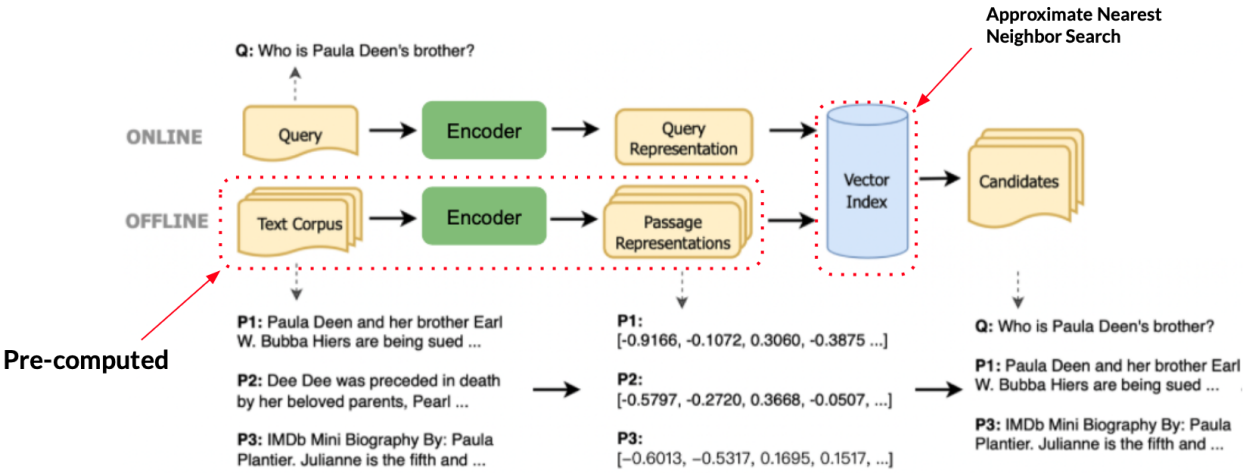


Figure 2.4: Adapted version of flowchart from [19] showing the overall pipeline during inference.

Inference with dual-encoder models can be efficiently done with the following steps:

1. *[Offline]* **Pre-computing passage representations**: Since we assume prior access to the inference collection (but not the queries), the passage representations can be pre-computed, independent of the queries, by using the passage encoder. This can be further parallelized across GPUs and is a one-time operation (assuming the corpus is fixed).

2. *[Offline]* **Indexing passage representations**: A dense vector index of the passage representations is created using FAISS [6].

3. *[Online]* **Computing query representation**: Given access to the query at inference, the query representation is obtained from the query encoder.

4. *[Online]* **Retrieving from index**: Approximate nearest neighbour search is used to retrieve the top-k most relevant passages, given the query. This process involves a matrix product between the query and individual passage representations, as shown in Figure 2.5.

2.4 METRICS

In this section, we briefly describe some of the evaluation metrics to measure to how relevant the retrieved results are to a given query. Most metrics assume the availability of binary (relevant/ir-

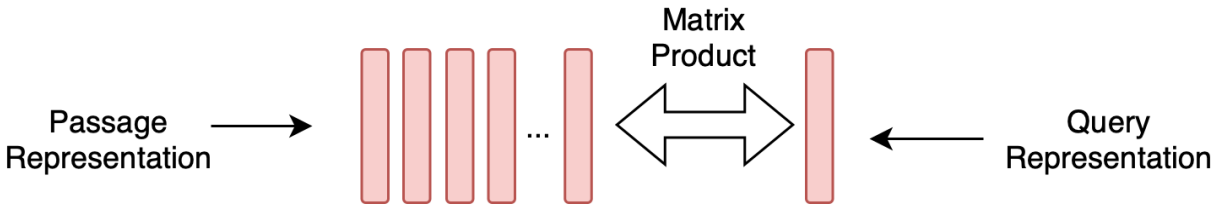


Figure 2.5: Figure showing the matrix product between the query and passage representations, during the approximate nearest neighbour search at inference.

relevant) or multi-level (relevance from 0 to 4) judgements for documents, with those for which judgements are unavailable being deemed potentially irrelevant.

2.4.1 Precision and Recall

These metrics are computed with respect to the fraction of relevant documents retrieved. For a given a query q, consider R_q as the documents retrieved for that query from a large collection of documents D. Given a binary relevance function $rel_q(d)$ that provides the relevance of a document d for a query q, the precision and recall are defined as follows:

$$Precision = \frac{sum_{d \in R_q} rel_q(d)}{|R_q|} \tag{2.10}$$

$$Recall = \frac{sum_{d \in R_q} rel_q(d)}{\sum_{d \in D} rel_q(d)} \tag{2.11}$$

2.4.2 Mean Reciprocal Rank (MRR)

Mean reciprocal rank is simply defined as the reciprocal rank of the first relevant document averaged over all the queries. This metric is also computed for binary relevance judgements. If $rank(d)$ gives the rank of a retrieved document, the reciprocal rank for a query is defined as:

$$RR = max_{d \in R_q} \frac{rel_q(d)}{rank(d)} \tag{2.12}$$

2.4.3 Mean Average Precision (MAP)

The mean average precision is computed as the mean of the average precision over all the queries. Given a binary relevance function $rel_q(d)$ and $Precision_i$ as the precision computed at rank i for a

query q, the average precision for the query is given as:

$$AvgP = \frac{\sum_{d \in R_q} Precision_i \cdot rel_q(d)}{\sum_{d \in D} rel_q(d)} \tag{2.13}$$

2.4.4 Normalized Discounted Cumulative Gain (nDCG)

This metric is defined for when multi-level judgements are available, for e.g. on a five-point scale from zero to four. The discounted cumulative gain for a query q is given as:

$$DCG = \sum_{d \in R_q} \frac{2^{rel_q(d)} - 1}{log_2(rank(d) + 1)} \tag{2.14}$$

The ideal DCG (iDCG) is computed using the same formula as above, but by assuming that the documents up to rank k as present in the ideal rank order. Given the iDCG, the normalized DCG (nDCG) for the query is computed as:

$$nDCG = \frac{DCG}{iDCG} \tag{2.15}$$

CHAPTER 3: ROBUSTNESS FROM PRE-TRAINING WITH SYNTHETIC QUESTIONS

3.1 INTRODUCTION

Traditional approaches to information retrieval (IR) such as TF-IDF [20] and BM25 [1] rely on lexical matching for query-passage alignment. In contrast, neural IR encodes passages and questions into continuous vector representations, enabling deeper semantic matching. Modern neural IR systems [3, 21] based on pre-trained masked language models (MLM) [5] typically employ a dual encoder architecture [22], where two separate MLMs encode the question and the passage. [2] show that useful weak supervision for such systems can be derived from the related task of machine reading comprehension (MRC) [23, 24]. Their Dense Passage Retrieval (DPR) model demonstrates state-of-the-art (SOTA) in-domain performance on multiple Wikipedia-based datasets [23, 24, 25, 26], outperforming both term matching baselines like BM25 and prior neural approaches, e.g., the Inverse Cloze Task [3] and latent learning of the retriever during MLM pre-training [27].

Despite its high in-domain utility, however, [8] show that DPR performance drops significantly in a novel test domain. They propose target domain synthetic data augmentation as a solution to this problem, which augments DPR with additional synthetic training data generated from target domain text. While this approach does indeed improve DPR scores in the new test domain, it has a key practical limitation: for every new target domain, it requires generating a new synthetic training corpus and re-training the model. Here we ask if an augmentation approach that only operates once in the source domain, and does not require re-training every time a new test domain is encountered, can also help improve domain generalization.

To better understand DPR's zero-shot out-of-domain (OOD) utility, we first run an empirical evaluation where both BM25 and DPR are applied to several out-of-domain test datasets. We observe that (i) DPR still holds an advantage over BM25 in near domain evaluation on Wikipedia-based datasets, but the difference is considerably lower than in the in-domain case, and (ii) In the far domain of biomedical text, DPR actually underperforms BM25. Our OOD evaluation is more comprehensive than [8], demonstrating the zero-shot utility of DPR in a more detailed and fine-grained manner.

Next we investigate if a one-off pre-training of DPR with large amounts of ***source domain*** synthetic IR data can help improve its robustness to domain shift. Utilization of synthetic training data is common in related tasks such as machine reading comprehension (MRC) [11, 28, 29]. Nevertheless, a close examination of synthetic pre-training as an augmentation technique is key for zero-shot neural IR due to the presence of highly effective and domain-agnostic term matching

baselines like BM25.

We fine-tune a sequence-to-sequence generator on labeled MRC data and use it to generate synthetic IR examples from source domain passages (§3.2). Our experiments show that pre-training DPR with these generated examples does indeed improve its accuracy on both in-domain and out-of-domain test sets. Crucially, the gap with BM25 in far domains is significantly reduced.

Our main contributions are as follows:

- We conduct an empirical evaluation of SOTA neural IR on multiple in-domain and out-of-domain test sets, showing how its utility varies in different test conditions.
- We show that a one-off ***source domain*** synthetic pre-training step can significantly improve the robustness of neural IR, with improvements on five different test sets, including in the practical zero-shot setting.

3.2 SOURCE DOMAIN SYNTHETIC PRE-TRAINING

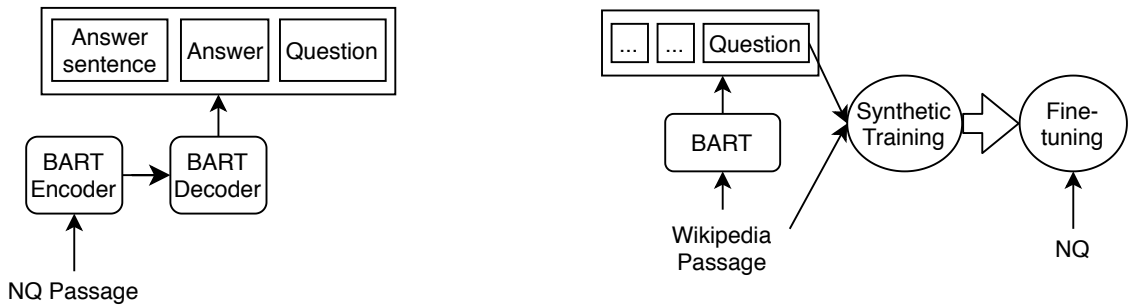
In this section, we describe the procedure for synthetic pre-training of the DPR model. We first detail how we train the sequence-to-sequence generator and generate source domain syntheic data from it. Next, we describe how this data is used for training the DPR model.

Let c be a text corpus and $d \in c$ be a document. An IR example, more specifically a passage retrieval example, consists of a question q and a passage p in d such that p contains an answer a to q. Let s be the sentence in p that contains a.

We first train an example generator by fine-tuning BART [30]—a pre-trained encoder-decoder language model—to generate an ordered triple (s, a, q) from an input passage p. This procedure in essence uses generation to first identify a candidate sentence s in p, then extract a candidate answer a from s, and finally generate a corresponding question q. In practice, we approximate the generation of s by generating only its first and last words. Finally, (q, p) is retained as a synthetic IR example. Labeled (p, s, a, q) tuples needed for the supervision of this model are taken from Natural Questions (NQ) [23], an existing MRC dataset over Wikipedia articles.

With the generator, we produce positive synthetic pre-training examples for DPR from Wikipedia passages. Following [11], we use top-p top-k sampling [31] to promote diversity in the generated examples. Training and inference of the synthetic example generator are depicted in Figures 3.1a and 3.1b, respectively. Figure 3.1c shows two example questions output by the generator from a Wikipedia passage.

To obtain a negative sample for each generated question q, we retrieve passages from Wikipedia using BM25 and randomly sample one that does not contain the generated answer a. Following [2], we also use in-batch negative samples for training. After pre-training with synthetic examples, we



(a) (b)

History of Tanzania The African Great Lakes nation of Tanzania dates formally from ***1964***, when it was formed out of the union of the much larger mainland territory of Tanganyika and the coastal archipelago of Zanzibar. The former was a colony and part of German East Africa from the 1880s to 1919, when, under the League of Nations, it became a ***British mandate***. It served as a military outpost ...

when did tanzania became a country in africa?
who owned zanzibar and tanganyika before they were independent?

(c)

Figure 3.1: The proposed IR training pipeline and a synthetic example. (a) A BART encoder-decoder LM is fine-tuned on NQ for QA example generation; (b) Synthetic examples generated from Wikipedia passages are used to pre-train the neural IR model before fine-tuning on NQ; (c) Two synthetic questions output by our synthetic generator using the depicted Wikipedia passage.

fine-tune the model with IR examples derived from NQ. We name this synthetically augmented DPR model *AugDPR*. We refer the reader to [2] for a more detailed description of the DPR training process.

3.3 EXPERIMENTAL SETUP

3.3.1 Datasets

We briefly describe our datasets in this section. Statistics for each dataset are shown in Table 3.1 with some sample questions from each shown in Table 3.2.

Training and In-Domain Evaluation: We train all systems on Natural Questions (NQ) [23], a dataset with questions derived from Google's search log and their human-annotated answers from Wikipedia articles. [32] report that 30% of the NQ test set questions have near-duplicate paraphrases in the training set and 60–70% of the test answers are also present in the training set. For this reason, in addition to the entire NQ test set, we also use the non-overlapping subsets released by [32] for in-domain evaluation.

Dataset	Domain	Passages	Questions
NQ	Wikipedia	21.0M	3,610
TriviaQA	Wikipedia	21.0M	11,313
WebQuestions	Wikipedia	21.0M	2,032
WikiMovies	Wikipedia	21.0M	9,952
BioASQ	Biomedical	37.4M	1092

Table 3.1: Statistics of the retrieval corpora and the test sets we use to evaluate all IR models.

Dataset	**Question**	**Answers**
NQ	what does hp mean in war and order	[‘hit points or health points’]
	who was named african footballer of the year 2014	[‘Yaya Touré’]
TriviaQA	Who was the man behind The Chipmunks?	[‘David Seville’]
	On a standard dartboard, which number lies between 12 and 20?	[‘five’, ‘5’]
WebQuestions	who was richard nixon married to?	[‘Pat Nixon’]
	what highschool did harper lee go to?	[‘Monroe County High School’]
WikiMovies	what does Tobe Hooper appear in?	[‘Body Bags’]
	Mick Davis directed which movies?	[‘The Match’]
BioASQ	Which receptor is inhibited by bimagrumab?	[‘activin type II receptors’]
	Which antiepileptic drug is most strongly associated with spina bifida?	[‘Valproate’]

Table 3.2: Sample questions from each of the datasets used in evaluation.

Near Domain Evaluation: For zero-shot near domain evaluation, where Wikipedia articles constitute the retrieval corpus, we use the test sets of three existing datasets.

TriviaQA [24] contains questions collected from trivia and quiz league websites, which are created by Trivia enthusiasts.

WebQuestions (WQ) [25] consists of questions obtained using the Google Suggest API, and answers selected from entities in Freebase by AMT workers.

WikiMovies [33] contains question-answer pairs on movies, built using the OMDb and MovieLens databases. We use the test split adopted in [34].

Far Domain Evaluation. For zero-shot far domain evaluation, we use a biomedical dataset.

BioASQ [35] is a competition[1] on large-scale biomedical semantic indexing and QA. We evaluate on all factoid question-answer pairs from the training and test sets of task 8B.

[1] http://bioasq.org/participate/challenges

3.3.2 Setup

Training: We train the synthetic example generator using the *(question, passage, answer)* triples from NQ. We then randomly sample 2M passages from the 21M-passage Wikipedia corpus and generate around four synthetic questions per passage. For top-p top-k sampling, we use $p = 0.95$ and $k = 10$. During synthetic pre-training of DPR, for each of the 2M passages, we randomly select one of its synthetic questions at each epoch to create a synthetic training example. After six epochs of synthetic pre-training, we fine-tune DPR on NQ for twenty epochs to get the AugDPR model. Table 3.3 gives the hyperparameters for training the generator and Table 3.4 lists the hyperparameters for pre-training and finetuning the neural IR model.

Hyperparameter	Value
Learning rate	3e-5
Epochs	3
Batch size	24
Max Sequence length	1024

Table 3.3: Hyperparameter settings during training the synthetic example generator (BART) using data from NQ.

Hyperparameter	Pre-training	Finetuning
Learning rate	1e-5	1e-5
Epochs	6	20
Batch size	1024	128
Gradient accumulation steps	8	1
Max Sequence length	256	256

Table 3.4: Hyperparameter settings for the neural IR model during pre-training on synthetic data and finetuning on NQ.

Baselines and Metrics: We evaluate BM25 as a term matching baseline. Our BM25 baseline is based on Lucene[2] implementation. BM25 parameters $b = 0.75$ (document length normalization) and $k_1 = 1.2$ (term frequency saturation) worked best. As our neural baseline, we use the DPR-single model trained on NQ and made public[3] by [2]. Both DPR and AugDPR use BERT-base-uncased for question and passage encoding. As in [2], our evaluation metric is top-k retrieval accuracy, which is the percentage of questions with at least one answer in the top k retrieved passages.

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

[3]https://github.com/facebookresearch/DPR

3.4 RESULTS

Table 3.5 shows NQ results on the entire test set as well as on the two subsets released by [32]. Synthetic pre-training yields larger gains on the non-overlapping splits, with up to a 4-point improvement in top-1 retrieval accuracy.

Model	Total			No answer overlap			No question overlap		
	Top-1	Top-10	Top-20	Top-1	Top-10	Top-20	Top-1	Top-10	Top-20
BM25	30.5	54.5	62.5	26.4	47.1	54.7	31.0	52.1	59.8
DPR	46.3	74.9	80.1	32.2	62.2	68.7	37.4	68.5	75.3
AugDPR	**46.8**	**76.0**	**80.8**	**36.0**	**65.0**	**70.8**	**41.4**	**70.8**	**76.6**

Table 3.5: NQ top-k retrieval results. Performance improves across the board with synthetic pre-training (AugDPR), but more on the non-overlapping subsets of [32].

To assess the cross-domain utility of AugDPR, we evaluate it zero shot on both near and far domain test sets. Table 3.6 shows the results. For comparison, we also show results for supervised models reported by [2] on TriviaQA and WebQuestions where the DPR model was trained directly on the training splits of these datasets. For the near domain datasets, both DPR and AugDPR outperform BM25 by a sizable margin; additionally, AugDPR consistently outperforms DPR. Furthermore, performance of AugDPR on WebQuestions is comparable to that of the supervised model. On the far domain, however, we observe that BM25 is a rather strong baseline, with clearly better scores than DPR. The synthetic pre-training of AugDPR reduces this gap considerably, resulting in a slightly lower top-20 score but a 2-point gain in top-100 score over BM25.

Model	Near Domains						Far Domain	
	TriviaQA		WebQuestions		WikiMovies		BioASQ	
	Top-20	Top-100	Top-20	Top-100	Top-20	Top-100	Top-20	Top-100
BM25	66.9	76.7	55.0	71.1	54.0	69.3	**42.1**	50.5
DPR	69.0	78.7	63.0	78.3	69.8	78.1	34.7	46.9
AugDPR	**72.2**	**81.1**	**71.1**	**80.8**	**72.5**	**80.7**	41.4	**52.4**
Supervised	79.4	85.0	73.2	81.4	-	-	-	-

Table 3.6: Zero-shot neural retrieval accuracy improves with synthetic pre-training (AugDPR) in all out-of-domain test settings. However, BM25 remains a strong baseline on the far domain dataset of BioASQ. The numbers for the supervised models are taken from [2].

To investigate the relative under-performance of neural IR on BioASQ, we take a closer look at the vocabularies of the two domains of Wikipedia articles and biomedical literature. Following [36], we compute the overlap between the 10k most frequent tokens (excluding stop words) in the two domains, represented by 3M randomly sampled passages from each. We observe a vocabulary overlap of only 17%, which shows that the two domains are considerably different in terminology,

explaining in part the performance drop in our neural models. Based on these results, we also believe that performance of neural IR in distant target domains can be significantly improved via pre-training on synthetic examples that are generated from raw text in the target domain. We plan to explore this idea in future work.

We also examine the lexical overlap between the questions and their passages, since a high overlap would favor term matching methods like BM25. We find that the coverage of the question tokens in the respective gold passages is indeed higher in BioASQ: 72.1%, compared to 58.6% and 63.0% in NQ and TriviaQA, respectively.

Model	Top-10	Top-20	Top-100
DPR	73.6	78.1	85.0
AugDPR-1M	74.4	79.2	85.5
AugDPR-2M	74.8	79.7	85.9
AugDPR-4M	74.6	79.1	85.9

Table 3.7: Retrieval accuracy on the Natural Questions development set with varying number of synthetic examples (1M vs 2M vs 4M) during pre-training.

To analyze how much synthetic data is required, we experiment with pre-training using 1M and 4M synthetic examples while keeping the number of training updates fixed. As Table 3.7 shows, we do not see any improvements from using more examples beyond 2M.

[2] report that DPR fine-tuning takes around a day on eight 32GB GPUs, which is a notable improvement over more computationally intensive pre-training approaches like [3, 27]. Our synthetic pre-training takes around two days on four 32GB GPUs, which is comparable with fine-tuning in terms of computational overhead.

CHAPTER 4: ROBUST ATTENTION DISTRIBUTIONS FROM PRE-TRAINING WITH ENTITY-CONDITIONED QUESTIONS

4.1 INTRODUCTION

Neural information retrieval (IR) performs query-passage matching at a semantic level, often using a dual-encoder architecture that encodes the queries and the passages separately. Examples of such models include the Dense Passage Retriever (DPR) [2] and ANCE [37], which fine-tune transformer-based [13] pre-trained language models [5] to compute contextualized representations of queries and passages.

In this work, we first uncover a shortcoming in the passage encoder of such a dual-encoder IR model, namely DPR, which stems from its sparse attention pattern. To illustrate, in Figure 4.1 we show a heatmap of the attention weights of DPR's passage encoder over different tokens of an example passage (taken from the Natural Questions (NQ) dataset [23]). We can see that the attention given to many potentially important words and phrases, e.g, *academy of management* and *twentieth century*, are rather low.

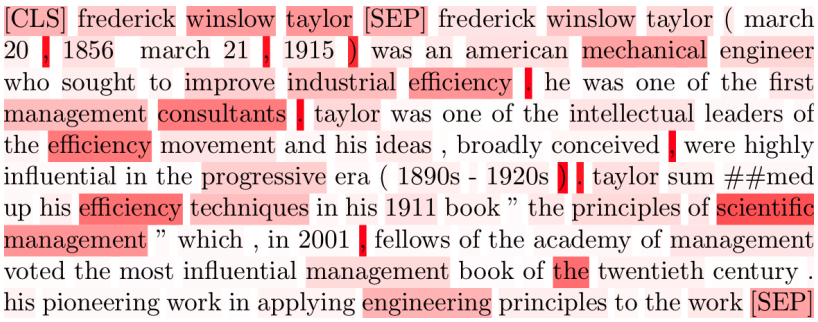


Figure 4.1: Heatmap of attention given to each token in DPR's passage representation. Darker shading indicates more attention.

What is the effect of such attention, or lack thereof, on retrieval performance? Table 4.1 shows DPR's retrieval scores for a gold-standard question (from the NQ dataset) and three automatically generated synthetic questions (details in Section 4.3) when paired with the passage of Figure 4.1. The gold-standard question, which overlaps highly with the well-attended first sentence of the passage, receives a relatively high retrieval score. Among the synthetic questions, the one that refers to the highest-attended entity (*principles of scientific management*) gets the highest score, whereas the ones about less attended entities (*twentieth century, progressive era*) receive considerably lower scores.

To further quantify this, we randomly sampled 20k passages from Wikipedia and identified

Question	Type	Score
the *american mechanical engineer* who sought to improve *industrial efficiency*	Gold	85.9
who wrote the *most influential management book* of the *twentieth century*	Synthetic	78.0
who was considered the father of management during the *progressive* era	Synthetic	82.2
who wrote the *principles of scientific management*	Synthetic	86.8

Table 4.1: Retrieval scores from DPR for the passage in Figure 4.1, against both a gold-standard question from NQ and three synthetic questions. The important terms in the question, that are also in the passage, are shown in *italic*.

named entities that received the highest and lowest attentions from the DPR passage encoder (using the process described in Section 4.3.1). We then generated synthetic questions corresponding to those entities (using the process of Section 4.3.2). We observe that on an average, the DPR score for questions corresponding to the highest attended entities was greater than that for questions corresponding to the lowest attended entities (73.7 vs. 72.1) in this sample. Further, we see the following pattern in the distribution of these entities: in a majority of cases (65%), the highest attended entity in a given passage is present in the first half of the passage, whereas the lowest attended entity can be found more often in the second half of the passage (60% of the cases). These observations are indicative of certain biases present in DPR's passage encoder that prevent it from attending uniformly over the different named entities in an input passage. We hypothesize that this bias in passage attentions is likely due to the training data having more questions about the beginnings of passages, which may limit the effectiveness of neural models.

As models trained on limited amounts of human-labeled data are prone to biases such as these, here we propose to augment the training data for neural IR with synthetic questions that are conditioned on the sparsely-attended parts of the passage. Concretely, we generate questions specifically about entities that receive low attentions from the passage encoder of the neural IR model. Our experiments show that augmenting the training with such questions does indeed enable neural IR models to attend more uniformly over passage tokens, resulting in performance improvements on multiple benchmark datasets.

Recently, there has been interest in understanding the robustness of neural IR models [38] and analyzing their behavior [39]. Our approach follows this line of work by leveraging the attentions of an IR model over given passages as a signal for better synthetic data augmentation. Prior work has also explored synthetic question generation for both question answering [11, 12, 40] and neural information retrieval [8, 41, 42]; different approaches to generating questions from passages include:

(a) unconditioned generation [8, 42], (b) generation conditioned on the candidate answer phrases within the passage [12, 40], and (c) conditioned on the summary of the passage [43, 44]. In contrast, our approach generates questions that are targeted towards the deficiencies of a given neural IR model, by conditioning the generation on sparsely attended entities in the passage. As we explain later in Section 4.3, our proposed methods for both identifying low-attention entities and generating questions that correspond to them can be leveraged for any dual-encoder IR model, including DPR [2], ANCE [37] or TAS-B [45].

Our main contributions are as follows:

- We show that a SOTA neural IR model is prone to learning sparse attention patterns over input passage tokens where key phrases (such as named entities) can receive low attention, leading to poor retrieval performance.
- We present an analysis of the bias in the Natural Questions[23] dataset and how this affects neural IR models in terms of retrieval and attention over the passages.
- We propose an entity-conditioned data augmentation strategy that generates questions about less attended entities in the passage.
- We demonstrate that incorporating these conditionally generated questions into the synthetic pre-training helps improve both model attention patterns and retrieval performance, including zero-shot settings.

4.2 BIASES IN THE NATURAL QUESTIONS DATASET

In this section, we give a brief analysis of Natural Questions (NQ) [23] to get an understanding of the bias in the dataset. The NQ dataset was originally introduced for machine reading comprehension with both answerable and unanswerable examples (meaning, queries were paired with passages containing the answer and not containing the answer), but was later adapted by [3] to train IR systems. In this version of the dataset, each example has a question, answer and a annotator-selected (*gold*) passage corresponding to it. The gold passages in the train and dev sets have 5.3 sentences on average.

We analyze the bias in the NQ training set as follows. We take the first 5 sentences in the gold passage and check whether each sentence contains the answer or not. Next, for each sentence, we also compute the extent of overlap of the question with the passage sentence. We only consider the non-stop words in the questions when computing this overlap. These numbers are shown in Table 4.2 according to the index of the sentence in the gold passage.

Sentence Index	Question Overlap	Answer Location
1	26.2%	30.0%
2	22.5%	29.4%
3	19.8%	26.0%
4	16.9%	21.4%
5	13.7%	16.5%

Table 4.2: Measure of presence of the answers and extent of overlap of questions in training set of NQ, according to sentence index in the gold passage. The *Question Overlap* column gives the % of overlap of question words with the words in the n^{th} sentence. *Answer Location* gives the % of cases when the n^{th} sentence has the answer within it.

Sentence Index	R@20 = 1	R@20 = 0
1	27.9%	16.1%
2	21.9%	17.1%
3	19.7%	16.3%
4	16.6%	14.9%
5	13.1%	10.4%

Table 4.3: Measure of extent of overlap of questions in dev set of NQ, with each sentence in the gold passage, according to DPR[2] retrieval performance. The *R@20 = 1* column gives the % of question overlap with sentence when gold passage is within the top-20 DPR results and *R@20 = 0* column gives the overlap when top-20 results don't have the gold passage.

Next, we also analyze how the extent of question overlap affects the performance on the NQ dev set. For this analysis, we consider the Dense Passage Retriever (DPR) [2] that is trained on the NQ training set. On the NQ dev set, we measure the extent of overlap of the questions with each gold passage sentence separately according to whether the passage in the top-20 DPR retrieval results. From Table 4.3, it can be seen that when the gold passage is not in the top-20 results (i.e $R@20 = 0$), the question overlap with the first two sentences is considerably lower than when the gold passage is in the top-20 results (i.e $R@20 = 1$). Thus, retrieval performance seems to directly related to the extent of question overlap with the first two sentences. On the other hand, the question overlap with the latter sentences hasn't dropped a lot in the $R@20 = 0$ case.

4.3 AUGMENTING WITH ENTITY-CONDITIONED SYNTHETIC QUESTIONS

To help neural retrievers capture all entities in the passage, we propose to augment the training data with synthetic questions that are conditioned on the less attended entities in the passage. Our synthetic data generation process, shown in Figure 4.2, involves the following steps: (a) Identifying entities with low attention, (b) Generating questions that are conditioned on these entities, and (c)

Filtering out low-quality synthetic questions. We describe each step in detail.

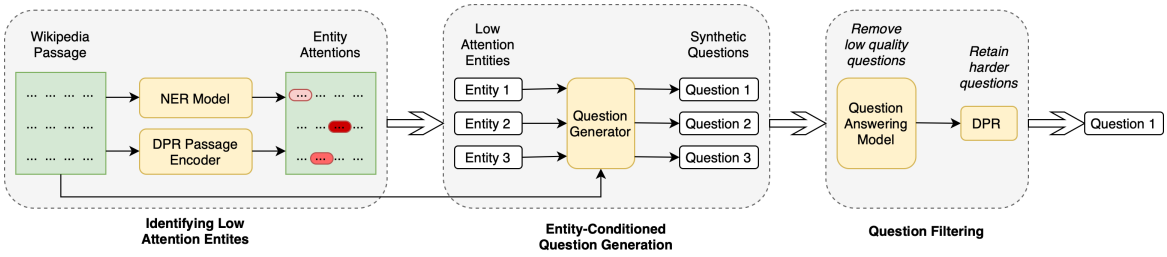


Figure 4.2: Overall framework of our synthetic data generation process to generate questions about named entities that receive low attentions from the DPR model.

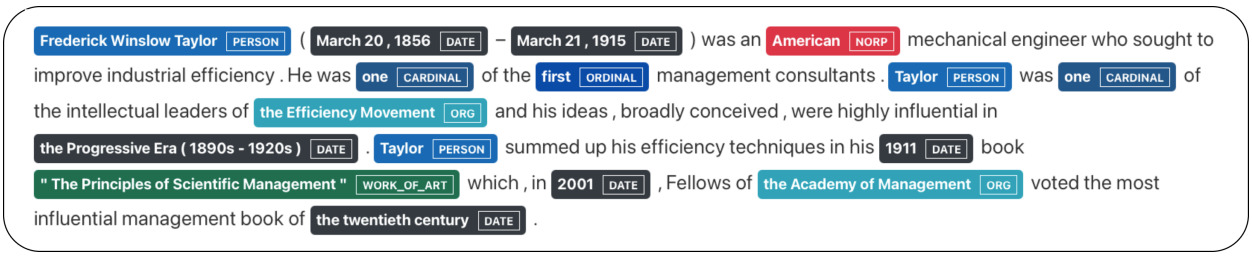


Figure 4.3: Entities automatically extracted from the passage of Figure 4.1.

4.3.1 Identifying entities with low attention

We use a named entity recognition system to first identify all the entities in a given passage (see Figure 4.3). Then we compute attentions of the neural IR model over the passage and aggregate the attentions over the corresponding word-pieces to get the attention for each of the entities in the passage. Finally, we identify the entities with the lowest attentions. Since DPR's passage encoder returns the CLS representation of the final layer as output, we use the attentions from that CLS token (i.e., where it serves as the query) as our passage attentions.

4.3.2 Entity-conditioned question generation

Given a passage and an entity in that passage, we aim to generate a synthetic question about that entity using the passage. Specifically, we train a synthetic example generator to take a passage p, an entity e and generate a question q and its corresponding answer a. To achieve this, we fine-tune an encoder-decoder language model [10] using examples from existing machine reading comprehension (MRC) datasets, which take the form of (q, p, a) triples. Given such a triple, we first identify entities in q that also appear in p. One such entity e is passed as input along with p to condition the question generation. Following [11], we promote diversity in the generated questions

Question	Conditioned Entity
who was considered the father of management during the progressive era	Progressive Era
who wrote the principles of scientific management	Principles of Scientific Management
who is known as the father of efficiency movement	Efficiency Movement

Table 4.4: Questions output by the synthetic generation system for the passage in Figure 4.3, based on the entity used for conditioning.

by using top-p top-k sampling [46] during generation. Table 4.4 shows some generated questions conditioned on entities in the passage of Figure 4.3.

4.3.3 Question filtering

We employ a two-stage filtering process to promote high quality in the synthetic data. In the first stage, a generated question q is considered to be consistent with the input passage p if a separately trained MRC model can find an answer to q in p with high confidence. All other questions are filtered out. Among the remaining questions and their corresponding passages, we expect those to provide the best complementary signal (relative to existing gold-standard data) for which the baseline neural IR model has a low retrieval score. Hence, we only include such low scoring (harder) pairs in the synthetic pre-training set.

4.4 EXPERIMENTAL SETUP

4.4.1 Datasets

We use the 21M Wikipedia passages from [2] as the retrieval corpus for all our experiments. These passages come from the December 2018 Wikipedia dump, with each article split into text blocks of 100 words as passages, serving as our basic retrieval units. We use two public IR datasets in our experiments.

Natural Questions: We train all systems on Natural Questions (NQ) [23], a dataset with questions derived from Google's search log and their human-annotated answers coming from Wikipedia articles. [32] report that 30% of the NQ test set questions have near-duplicate paraphrases in the training set and 60–70% of the test answers are also present in the training set. For this reason,

in addition to the original 3,610 test questions, we also report evaluation on the non-overlapping subsets (1,313 no-answer overlap and 672 no-question overlap) released by [32].

WebQuestions: The dataset consists of questions obtained using the Google Suggest API, with answers selected from entities in Freebase by AMT workers [47]. We use the 2,032 test questions in this dataset for zero-shot evaluation.

4.4.2 Baselines

As traditional term matching baselines, we evaluate the TF-IDF system[1] from [34] and the BM25 implementation provided by Pyserini[2]. We evaluate DPR[3] as our neural IR baseline[4]. [2] report that the performance of DPR is affected by the number of in-batch negatives used in training, which in turn is dependent on the number of GPUs available. They use 128 in-batch negatives with eight 32GB V100s. Since we only had access to four 32GB V100s, we use 64 in-batch negatives. We call this implementation *DPR (ours)*.

To compare our approach with a generation strategy that does not use any conditioning, we also train an unconditioned generation system, similar to [42], that generates question-answer pairs using just the passage as input. We call this the *unconditioned* question generator, since the questions are not conditioned to be about any specific entities. This serves as a baseline question generation approach and is comparable with prior work [8, 41, 42] in synthetic data generation for IR, which do not enforce such specific conditioning into the question generation process.

4.4.3 Synthetic Data Generation

To create our synthetic pre-training corpus, first we derive a random sample of passages from the retrieval collection. We identify the named entities in these passages using a publicly available NER system[5] trained on the OntoNotes corpus [48]. When selecting the entities that are used for conditioning, the following entity types are considered: Person, NORP, Facility, Organization, GPE, Location, Product, Event, Work of art, Law and Language. The MRC model used in the first stage of question filtering is trained sequentially on SQuAD2.0 [49] and Natural Questions [23], with hyper-parameters shown in Table 4.5.

[1] https://github.com/efficientqa/retrieval-based-baselines#tfidf-retrieval
[2] https://github.com/castorini/pyserini/blob/master/docs/experiments-dpr.md
[3] https://github.com/facebookresearch/DPR
[4] We note that our approach can be similarly applied to other dual-encoder IR models such as ANCE [37].
[5] https://demo.deeppavlov.ai

Hyperparameter	SQuAD2.0	NQ
Learning rate	3e-5	2e-5
Epochs	3	1
Batch size	8	48
Max sequence length	384	512
Max question length	64	18
Document stride	128	192

Table 4.5: Hyperparameter settings for training the MRC model used for question filtering.

We train the *unconditioned* question generator by fine-tuning BART [10] with the question-passage-answer triples present in NQ. Table 4.6 gives the hyperparameters for training the synthetic example generator. We fine-tune a separate BART model for *conditioned* question generation, which takes a passage-entity pair as input and generates an entity-conditioned question and its answer as output. We repurpose the NQ dataset for training a conditioned question generation system, by converting the question-passage-answer triples into (question, conditioned entity, passage, answer) quadruples. To obtain the conditioning entities used in training, we identify entities from noun chunks (obtained using spaCy [50]) in the question that also occur in the corresponding passage.

Hyperparameter	Value
Learning rate	3e-5
Epochs	3
Batch size	24
Max Sequence length	1024

Table 4.6: Hyperparameter settings during training the synthetic example generator (BART) using data from NQ.

We use the unconditioned generation system to first generate 1M synthetic training examples. We then use the conditioned generation system to obtain 500k examples after filtering, and mix them with 500k unconditioned examples to obtain our final dataset of size 1M, which we call *mixed* synthetic data. Since the conditioned data contains questions primarily about less attended entities, this combination with unconditioned examples helps maintain adequate diversity in the final mixed dataset. We follow the same process as in [2] and use term matching to sample hard negatives for the questions.

4.4.4 Training

The DPR baseline is trained only on data fron the Natural Questions dataset [23]. We name the model pre-trained on the 1M unconditioned synthetic data as *UnCon-DPR* and the one pre-trained

Hyperparameter	Pre-training	Finetuning
Learning rate	1e-5	2e-5
Epochs	10	40
Batch size	512	64
Gradient accumulation steps	8	1
Max Sequence length	256	256

Table 4.7: Hyperparameter settings for the neural IR model during pre-training on synthetic data and fine-tuning on NQ.

on the 1M mixed synthetic data as *Mixed-DPR*. Table 4.7 lists the hyperparameters for pre-training and fine-tuning the neural IR models.

4.5 RESULTS

Model	Natural Questions (NQ)						WebQuestions	
	Full test		No ans. overlap		No ques. overlap		Test	
	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
TF-IDF	14.2	32.0	13.6	28.6	14.6	31.8	14.5	32.1
BM25	22.7	44.6	20.1	39.6	24.0	43.4	18.9	41.8
DPR (ours)	44.3	67.1	32.2	53.2	37.2	60.1	29.4	51.6
UnCon-DPR	45.8	68.4	32.7	54.4	36.9	60.6	31.5	53.2
Mixed-DPR	**45.9**	**69.0**	**33.8**	**55.7**	**37.9**	**62.0**	**32.2**	**53.9**

Table 4.8: Top-k retrieval results (in %) on test sets of Natural Questions (including the non-overlapping subsets of [32]) and WebQuestions. Numbers on WebQuestions are in zero-shot settings, since models have been trained on NQ.

Similar to [2], we evaluate all systems using top-k retrieval accuracy, which is the percentage of questions with at least one answer in the top k retrieved passages. Table 4.8 shows the results for the term matching and neural models.

Firstly, we can see that the two DPR models with synthetic pre-training improve over the baseline DPR system. Our Mixed-DPR model, which employs entity-conditioned synthetic questions for pre-training, consistently outperforms all other models including UnCon-DPR, which is pre-trained only on unconditioned questions. Crucially, on NQ, we observe greater improvements with Mixed-DPR on the non-overlapping and thus harder subsets of NQ, which indicates that the robustness of DPR improves with our proposed data augmentation strategy. Further, we see improvements for Mixed-DPR in a zero-shot evaluation on WebQuestions.

4.5.1 Analysis

To investigate the effect of the entity-conditioned questions used in synthetic pre-training, we examine how their application affects both the passage-level and token-level attentions of the DPR model.

Passage-level attention distribution. First, we observe that the baseline DPR model (which is trained only on NQ) tends to attend more to the earlier sentences of a given passage. We therefore compare attention on the first sentence (computed as the average attention over its tokens) with average attention on the rest of the sentences in the passage. We sample 10k passages from the retrieval corpus and compute attentions for the baseline DPR, UnCon-DPR and Mixed-DPR models. We observe that Mixed-DPR pays 1.8% higher attention to the later sentences of the passage compared to the baseline DPR model. When compared to UnCon-DPR, this difference is 1.1%. These results show that Mixed-DPR learns to attend more to the latter sentences of the passage which, as shown in Figure 4.1, is typically where most of the weakly attended entities of the baseline model occur.

Token-level attentions. Here, we look at the entropy of token-level attentions in a given passage for the above models. Entropy here is a measure of the uniformity of a model's attention over the tokens in the passage, with a higher entropy indicating a more uniform distribution. For the 10k passages previously sampled, we see that the baseline DPR, UnCon-DPR and Mixed-DPR models have attention entropies of 3.97, 3.80 and 4.10 respectively, with Mixed-DPR being the highest. This suggests that the improvements in top-k retrieval accuracy stem (at least partly) from a more scattered and potentially more robust attention pattern learned by Mixed-DPR.

CHAPTER 5: CONCLUSION

We have shown that pre-training a SOTA neural IR model using large amounts of *source domain* synthetic data improves its robustness in zero-shot settings. Our experiments show consistent performance gains on five in-domain and out-domain test sets, including a far target domain that has significant vocabulary mismatch with the training domain.

We then discovered a specific issue in neural IR systems that stems from sparse attention patterns learned over input passage tokens, which can lead to sub-optimal performance on queries about less attended areas of the passage. With targeted synthetic data augmentation, we address this issue for DPR—a state-of-the-art neural IR model—and enable it to attend more uniformly over passage tokens. Our proposed method improves performance on two different datasets, and in in-domain as well as zero-shot evaluation. While our work is an important first step towards solving this problem, one of the primary goals of this thesis is to draw attention of the community to this important limitation of supervised neural IR and inspire future research on the topic.

CHAPTER 6: FUTURE DIRECTIONS

In this thesis, we leveraged synthetic pre-training as a means to help neural retrieval models overcome shortcomings from training on limited gold-standard data. Here, we propose some directions for future work that can push towards improving robustness of neural retrieval models, without the need for an additional synthetic pre-training step. These include: Incorporating multi-task learning for identifying important parts of the passage (*Direction 1*); Learning a separate module for token importance scores, which can be made specific to the domain at hand (*Direction 4*). One line of work along interpretability would to be understand which tokens in the passage are pre-dominantly captured in the passage representation (*Direction 3*). Finally, another interesting direction would be to improve the diversity of the retrieved passages, to ensure passages with similar information are not repeated in the top-k results (*Direction 2*). More details about each below.

Direction 1: Multi-task learning One potential direction is to incorporate additional objectives, e.g. multitask learning, to help models learn more robust attention patterns without requiring synthetic data. For example, named entity recognition as an auxiliary task may help the model identify key phrases in the passages, which in principle can help it to pay more attention to those during encoding.

Direction 2: Improving diversity of retrieval Another direction is in improving the diversity of the retrieved passages, in order to maximize the amount of relevant information available for subsequently answering the question. This can provide better coverage for questions with multiple correct answers and would also be useful in multilingual settings, wherein different answers can be present exclusively in different languages.

Direction 3: Understanding token capture Since dual-encoder retrieval models output a single vector representation, it is difficult to understand which tokens in the passage are captured in the passage representation. Hence, it would be useful to design a probe that can measure how sensitive the output representation is to the individual tokens in the passage.

Direction 4: Disentangling importance scores from token representations Neural encoders currently jointly learn the individual token representations and corresponding importance scores. One direction would be to disentangle this into two separate modules. In the target domain, one could then leverage domain-specific language models for obtaining the token representations and inject domain knowledge while computing the importance scores. This overcomes the need for labelled domain-specific IR data in order to adapt neural retrieval models to the target domain.

REFERENCES

[1] S. Robertson and H. Zaragoza, "The probabilistic relevance framework: Bm25 and beyond," *Foundations and Trends in Information Retrieval*, vol. 3, no. 4, pp. 333–389, 2009.

[2] V. Karpukhin, B. Oğuz, S. Min, L. Wu, S. Edunov, D. Chen, and W.-t. Yih, "Dense passage retrieval for open-domain question answering," *arXiv preprint arXiv:2004.04906*, 2020.

[3] K. Lee, M.-W. Chang, and K. Toutanova, "Latent retrieval for weakly supervised open domain question answering," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 6086–6096.

[4] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever et al., "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "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)*, 2019, pp. 4171–4186.

[6] J. Johnson, M. Douze, and H. Jégou, "Billion-Scale Similarity Search with GPUs," *IEEE Transactions on Big Data*, 2019.

[7] B. Mitra, N. Craswell et al., "An introduction to neural information retrieval," *Foundations and Trends® in Information Retrieval*, vol. 13, no. 1, pp. 1–126, 2018.

[8] R. Reddy, B. Iyer, M. A. Sultan, R. Zhang, A. Sil, V. Castelli, R. Florian, and S. Roukos, "Synthetic Target Domain Supervision for Open Retrieval QA," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 1793–1797.

[9] N. Thakur, N. Reimers, A. Rücklé, A. Srivastava, and I. Gurevych, "Beir: A heterogeneous benchmark for zero-shot evaluation of information retrieval models," in *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

[10] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 7871–7880.

[11] M. A. Sultan, S. Chandel, R. F. Astudillo, and V. Castelli, "On the importance of diversity in question generation for QA," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 5651–5656.

[12] C. Alberti, D. Andor, E. Pitler, J. Devlin, and M. Collins, "Synthetic qa corpora generation with roundtrip consistency," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 6168–6173.

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, 2017, pp. 5998–6008.

[14] R. Nogueira and K. Cho, "Passage re-ranking with bert," *arXiv preprint arXiv:1901.04085*, 2019.

[15] Z. Dai and J. Callan, "Deeper text understanding for ir with contextual neural language modeling," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2019, pp. 985–988.

[16] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese Neural Networks for One-Shot Image Recognition," in *ICML deep learning workshop*, vol. 2. Lille, 2015.

[17] W.-t. Yih, K. Toutanova, J. C. Platt, and C. Meek, "Learning discriminative projections for text similarity measures," *CoNLL-2011*, p. 247, 2011.

[18] D. Gillick, S. Kulkarni, L. Lansing, A. Presta, J. Baldridge, E. Ie, and D. Garcia-Olano, "Learning dense representations for entity retrieval," in *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, 2019, pp. 528–537.

[19] Y. Qu, Y. Ding, J. Liu, K. Liu, R. Ren, W. X. Zhao, D. Dong, H. Wu, and H. Wang, "RocketQA: An optimized training approach to dense passage retrieval for open-domain question answering," in *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, June 2021. [Online]. Available: https://aclanthology.org/2021.naacl-main.466 pp. 5835–5847.

[20] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. USA: McGraw-Hill, Inc., 1986.

[21] W.-C. Chang, X. Y. Felix, Y.-W. Chang, Y. Yang, and S. Kumar, "Pre-training tasks for embedding-based large-scale retrieval," in *International Conference on Learning Representations*, 2019.

[22] J. Bromley, J. W. Bentz, L. Bottou, I. Guyon, Y. LeCun, C. Moore, E. Säckinger, and R. Shah, "Signature verification using a "siamese" time delay neural network," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 7, no. 04, pp. 669–688, 1993.

[23] T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. Parikh, C. Alberti, D. Epstein, I. Polosukhin, J. Devlin, K. Lee et al., "Natural questions: a benchmark for question answering research," *Transactions of the Association for Computational Linguistics*, vol. 7, pp. 453–466, 2019.

[24] M. Joshi, E. Choi, D. S. Weld, and L. Zettlemoyer, "Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, pp. 1601–1611.

[25] J. Berant, A. Chou, R. Frostig, and P. Liang, "Semantic parsing on freebase from question-answer pairs," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 2013, pp. 1533–1544.

[26] P. Baudiš and J. Šedivỳ, "Modeling of the question answering task in the yodaqa system," in *International Conference of the Cross-Language Evaluation Forum for European Languages*. Springer, 2015, pp. 222–228.

[27] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M.-W. Chang, "Realm: Retrieval-augmented language model pre-training," *arXiv preprint arXiv:2002.08909*, 2020.

[28] S. Shakeri, C. Nogueira dos Santos, H. Zhu, P. Ng, F. Nan, Z. Wang, R. Nallapati, and B. Xiang, "End-to-end synthetic data generation for domain adaptation of question answering systems," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, Nov. 2020. [Online]. Available: https://www.aclweb.org/anthology/2020.emnlp-main.439 pp. 5445–5460.

[29] R. Zhang, R. Gangi Reddy, M. A. Sultan, V. Castelli, A. Ferritto, R. Florian, E. Sarioglu Kayi, S. Roukos, A. Sil, and T. Ward, "Multi-stage pre-training for low-resource domain adaptation," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, Nov. 2020. [Online]. Available: https://www.aclweb.org/anthology/2020.emnlp-main.440 pp. 5461–5468.

[30] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, July 2020, pp. 7871–7880.

[31] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi, "The curious case of neural text degeneration," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. [Online]. Available: https://openreview.net/forum?id=rygGQyrFvH

[32] P. Lewis, P. Stenetorp, and S. Riedel, "Question and answer test-train overlap in open-domain question answering datasets," *arXiv preprint arXiv:2008.02637*, 2020.

[33] A. Miller, A. Fisch, J. Dodge, A.-H. Karimi, A. Bordes, and J. Weston, "Key-value memory networks for directly reading documents," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 1400–1409.

[34] D. Chen, A. Fisch, J. Weston, and A. Bordes, "Reading wikipedia to answer open-domain questions," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, pp. 1870–1879.

[35] G. Tsatsaronis, G. Balikas, P. Malakasiotis, I. Partalas, M. Zschunke, M. R. Alvers, D. Weissenborn, A. Krithara, S. Petridis, D. Polychronopoulos et al., "An overview of the bioasq large-scale biomedical semantic indexing and question answering competition," *BMC bioinformatics*, vol. 16, no. 1, p. 138, 2015.

[36] S. Gururangan, A. Marasović, S. Swayamdipta, K. Lo, I. Beltagy, D. Downey, and N. A. Smith, "Don't stop pretraining: Adapt language models to domains and tasks," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, July 2020. [Online]. Available: https://www.aclweb.org/anthology/2020.acl-main.740 pp. 8342–8360.

[37] L. Xiong, C. Xiong, Y. Li, K.-F. Tang, J. Liu, P. N. Bennett, J. Ahmed, and A. Overwijk, "Approximate Nearest Neighbor Negative Contrastive Learning for Dense Text Retrieval," in *International Conference on Learning Representations*, 2020.

[38] C. Wu, R. Zhang, J. Guo, Y. Fan, and X. Cheng, "Are Neural Ranking Models Robust?" *arXiv preprint arXiv:2108.05018*, 2021.

[39] S. MacAvaney, S. Feldman, N. Goharian, D. Downey, and A. Cohan, "ABNIRML: Analyzing the Behavior of Neural IR Models," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 224–239, 2022.

[40] S. Shakeri, C. N. d. Santos, H. Zhu, P. Ng, F. Nan, Z. Wang, R. Nallapati, and B. Xiang, "End-to-end synthetic data generation for domain adaptation of question answering systems," *arXiv preprint arXiv:2010.06028*, 2020.

[41] J. Ma, I. Korotkov, Y. Yang, K. Hall, and R. McDonald, "Zero-Shot Neural Passage Retrieval via Domain-targeted Synthetic Question Generation," in *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, 2021, pp. 1075–1088.

[42] R. G. Reddy, V. Yadav, M. A. Sultan, M. Franz, V. Castelli, H. Ji, and A. Sil, "Towards Robust Neural Retrieval Models with Synthetic Pre-Training," *arXiv preprint arXiv:2104.07800*, 2021.

[43] C. Lyu, L. Shang, Y. Graham, J. Foster, X. Jiang, and Q. Liu, "Improving Unsupervised Question Answering via Summarization-Informed Question Generation," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 4134–4148.

[44] L. Zhou, K. Small, Y. Zhang, and S. Atluri, "Generating Self-Contained and Summary-Centric Question Answer Pairs via Differentiable Reward Imitation Learning," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2021, pp. 5103–5135.

[45] S. Hofstätter, S.-C. Lin, J.-H. Yang, J. Lin, and A. Hanbury, "Efficiently Teaching an Effective Dense Retriever with Balanced Topic Aware Sampling," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2021, pp. 113–122.

[46] A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi, "The Curious Case of Neural Text Degeneration," in *International Conference on Learning Representations*, 2020.

[47] J. Berant, A. Chou, R. Frostig, and P. Liang, "Semantic Parsing on Freebase from Question-Answer Pairs," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Seattle, Washington, USA: Association for Computational Linguistics, Oct. 2013. [Online]. Available: https://aclanthology.org/D13-1160 pp. 1533–1544.

[48] R. Weischedel, S. Pradhan, L. Ramshaw, M. Palmer, N. Xue, M. Marcus, A. Taylor, C. Greenberg, E. Hovy, R. Belvin et al., "Ontonotes Release 4.0," *LDC2011T03, Philadelphia, Penn.: Linguistic Data Consortium*, 2011.

[49] P. Rajpurkar, R. Jia, and P. Liang, "Know What You Don't Know: Unanswerable Questions for SQuAD," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2018, pp. 784–789.

[50] M. Honnibal, I. Montani, S. Van Landeghem, and A. Boyd, "spaCy: Industrial-Strength Natural Language Processing in Python," *Zenodo*, 2020.