

ZusammenQA: Data Augmentation with Specialized Models for Cross-lingual Open-retrieval Question Answering System

Chia-Chien Hung¹, Tommaso Green¹, Robert Litschko¹,
Tornike Tsereteli¹, Sotaro Takeshita¹, Marco Bombieri²,
Goran Glavaš³ and Simone Paolo Ponzetto¹

¹ Data and Web Science Group, University of Mannheim, Germany

² ALTAIR Robotics Lab, University of Verona, Italy

³ CAIDAS, University of Würzburg, Germany

{chia-chien.hung, tommaso.green, robert.litschko,
tornike.tsereteli, sotaro.takeshita, ponzetto}@uni-mannheim.de
marco.bombieri_01@univr.it, goran.glavas@uni-wuerzburg.de

Abstract

This paper introduces our proposed system for the MIA Shared Task on Cross-lingual Open-retrieval Question Answering (COQA). In this challenging scenario, given an input question the system has to gather evidence documents from a multilingual pool and generate from them an answer in the language of the question. We devised several approaches combining different model variants for three main components: *Data Augmentation*, *Passage Retrieval*, and *Answer Generation*. For passage retrieval, we evaluated the monolingual BM25 ranker against the ensemble of *re-rankers based on multilingual pretrained language models* (PLMs) and also variants of the shared task baseline, re-training it from scratch using a recently introduced contrastive loss that maintains a strong gradient signal throughout training by means of mixed negative samples. For answer generation, we focused on language- and domain-specialization by means of continued language model (LM) pretraining of existing multilingual encoders. Additionally, for both passage retrieval and answer generation, we augmented the training data provided by the task organizers with automatically generated question-answer pairs created from Wikipedia passages to mitigate the issue of data scarcity, particularly for the low-resource languages for which no training data were provided. Our results show that language- and domain-specialization as well as data augmentation help, especially for low-resource languages.

1 Introduction

Open-retrieval Question Answering (OQA), where the agent helps users to retrieve answers from large-scale document collections with given *open* ques-

tions, has arguably been one of the most challenging natural language processing (NLP) applications in recent years (e.g., Lewis et al., 2020; Karpukhin et al., 2020; Izacard and Grave, 2021). As is the case with the vast majority of NLP tasks, much of the OQA focused on English, relying on a pipeline that crucially depends on a neural passage retriever, i.e., a (re-)ranking model – trained on large-scale English QA datasets – to find evidence passages in English (Lewis et al., 2020) for answer generation. Unlike in many other retrieval-based tasks, such as ad-hoc document retrieval (Craswell et al., 2021), parallel sentence mining (Zweigenbaum et al., 2018), or Entity Linking (Wu et al., 2020), the progress toward *Cross-lingual Open-retrieval Question Answering (COQA)* has been hindered by the lack of efficient integration and consolidation of knowledge expressed in different languages (Loginova et al., 2021). COQA is especially relevant for opinionated information, such as news, blogs, and social media. In the era of fake news and deliberate misinformation, training on only (or predominantly) English texts is more likely to lead to more biased and less reliable NLP models. Further, an Anglo- and Indo-European-centric NLP (Joshi et al., 2020) is unrepresentative of the needs of the majority of the world’s population (e.g., Mandarin and Spanish have more native speakers than English, and Hindi and Arabic come close) and contributes to the widening of the digital language divide.¹ Developing solutions for cross-lingual open QA (COQA) thus contributes towards the goal of global equity of information access.

COQA is the task of automatic question answering, where the answer is to be found in a large

¹<http://labs.theguardian.com/digital-language-divide/>

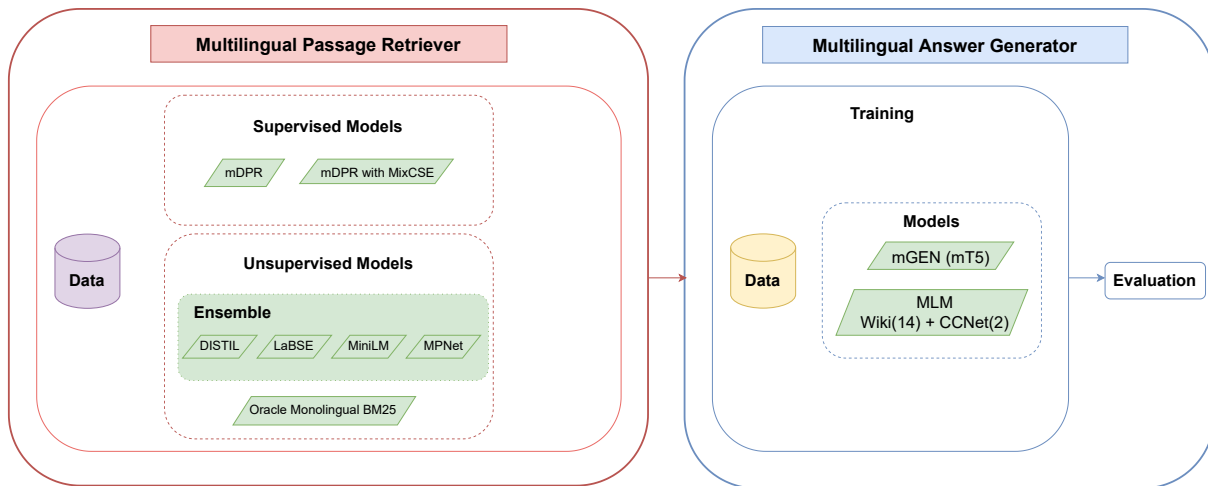


Figure 1: The proposed pipeline for the cross-lingual QA problem. The pipeline is composed of two stages: (i) the retrieval of documents containing a possible answer (red box) and the generation of an answer (blue box). For the retrieval part, we exploited different methods, based both on training mDPR variants and on the ensembling of *blackbox* models. For training the mDPR variants, we enlarge the original training dataset with samples from our data augmentation pipeline. For the generation part, we enrich the existing baseline method with data augmentation and masked language modeling.

multilingual document collection. It is a challenging NLP task, with questions written in a user’s preferred language, where the system needs to find evidence in a large-scale document collection written in different languages; the answer then needs to be returned to the user in their preferred language (i.e., the language of the question).

More formally, the goal of a COQA system is to find an answer a to the query q in the collection of documents $\{D\}_i^N$. In the cross-lingual setting, q and $\{D\}_i^N$ are, in general, in different languages. For example, the users can ask a question in Japanese, and the system can search an English document for an answer that then needs to be returned to the user in Japanese.

In this work, we propose data augmentation for specialized models that correspond to the two main components of a standard COQA system – *passage retrieval* and *answer generation*: (1) we first extract the passages from all documents of all languages, exploiting both *unsupervised* and *supervised* (e.g., mDPR variants) passage retrieval methods; (2) we then use the retrieved passages from the previous step and further conduct intermediate training of a pretrained language model (e.g., mT5 (Xue et al., 2021)) on the extracted augmented data in order to inject language-specific knowledge into the model, and then generate the answers for each question from different language portions. As a result, we obtain a specialized model trained on

the augmented data for the COQA system. The overall process is illustrated in Figure 1.

2 Data Augmentation

We use a language model to generate question-answer (QA) pairs from English texts, which we then filter according to a number of heuristics and translate into the other 15 languages.² An example can be seen in Figure 2 in the Appendix.

2.1 Question-Answer Generation

For generating the question-answer pairs, we use the provided Wikipedia passages as the input to a language model, which then generates questions and answers based on the input text. We based our choice of the model on the findings by Dong et al. (2019) and Bao et al. (2020), who showed that language models that are fine-tuned jointly on Question Answering and Question Generation, outperform individual models fine-tuned independently on those tasks. More specifically, we use the model by Dugan et al. (2022) and make slight modifications. Dugan et al. (2022) used a T5 model fine-tuned on SQuAD and further fine-tuned it on three tasks simultaneously: Question Generation (GQ), Question Answering (QA), and Answer Extraction (AE). They also included a summarization

²Arabic(AR), Bengali(BN), Finish(FI), Japanese(JA), Korean(KO), Russian(RU), Telugu(TE), Spanish(ES), Khmer(KM), Malay(MS), Swedish(SV), Turkish(TR), Chinese(ZH-CN), Tagalog(TL) and Tamil(TA).

module to create lexically diverse question-answer pairs. We found that using this module sometimes leads to factually incorrect passages, and leave this to future work. Similar to Dugan et al. (2022), we split the original passages into whole sentences that are shorter than 512 tokens.³ We then generate the pairs using the first three sub-passages.

2.2 Filtering

Before translating question-answer pairs, to ensure better translations, we enforce each pair to satisfy at least one of a number of heuristics, which we determined through manual evaluation of the generated pairs. Each pair is evaluated on whether one of the following is true (in the respective order): the answer is a number, the question starts with the word *who*, the question starts with the words *how many*, or the answer contains a number or a date. After filtering, we are left with roughly 339,000 question-answer pairs.

2.3 Translation

We use the Google Translate API provided by *translatepy*⁴ to translate the filtered question-answer pairs from English into the relevant 15 languages. Each language has an equal number of question-answer pairs. In total, we generate about 5.4 million pairs for all languages combined.

3 Methodology

Following the approach described in Asai et al. (2021b), we consider the COQA problem as two sub-components: the *retrieval* of documents containing a possible answer and the *generation* of an answer. Figure 1 summarizes the proposed methods. This section is organized as follows: we present the proposed retrieval methods in §3.1 and demonstrate the language-specialized methods for answer generation in §3.2.

3.1 Passage Retrieval

For the passage retrieval phase, we explored the approaches described in the following sections, which fall into three main categories: the enhancement of the training procedure of the mDPR baseline, the ensembling of *blackbox* models (i.e. retrieval using multilingual sentence encoders trained for semantic similarity) and lexical retrieval using BM25. While

³We use a different sentence splitting method, namely pySBD.

⁴<https://github.com/Animenosekai/translate>

the first category is a *supervised* approach, which uses QA datasets to inject task knowledge into pre-trained models, the others use general linguistic knowledge for retrieving (i.e., *unsupervised*).

Baseline: mDPR We take as a baseline the method proposed in Asai et al. (2021b). They propose mDPR (Multilingual Dense Passage Retriever), a model that extends the Dense Passage Retriever (DPR) (Qu et al., 2021) to a multilingual setting. It is made of two mBERT-based encoders (Devlin et al., 2019), one for the question and one for the passages. The training approach proceeds over two subsequent stages: (i) *parameter updates* and (ii) *cross-lingual data mining*.

In the first phase, both mDPR and mGEN (§3.2) are trained one after the other. For mDPR, the model processes a dataset $\mathcal{D} = \{(q_i, p_i^+, p_{i,1}^-, p_{i,2}^-, \dots, p_{i,n}^-)\}_{i=1}^m$ made of tuples containing a question q_i , the passage p_i^+ containing an answer (called positive or gold), and a set $\{p_{i,j}^-\}_{j=1}^n$ of negative passages. For every question, negatives are made up of the positive passages from the other questions or passages either extracted at random or produced by the subsequent data mining phase. To do this, they use a contrastive loss ($\mathcal{L}_{\text{mdpr}}$) that moves the embedding of the question close to its positive passage, while at the same time repelling the representations of negative passages:

$$\mathcal{L}_{\text{mdpr}} = -\log \frac{\langle \mathbf{e}_{q_i}, \mathbf{e}_{p_i^+} \rangle}{\langle \mathbf{e}_{q_i}, \mathbf{e}_{p_i^+} \rangle + \sum_{j=1}^n \langle \mathbf{e}_{q_i}, \mathbf{e}_{p_{i,j}^-} \rangle}$$

In the second stage, the training set is expanded by finding new positive and negative passages using Wikipedia language links and mGEN (§3.2) to automatically label passages. This two-staged training pipeline is repeated T times.

mDPR variants One of our approaches is to simply substitute the loss function presented above with a contrastive loss described in Zhang et al. (2022), named MixCSE. In this work, the authors tackle a common problem of out-of-the-box BERT sentence embeddings, called anisotropy (Li et al., 2020), which makes all the sentence representations to be distributed in a narrow cone. Contrastive learning has already proven effective in alleviating this issue by distributing embeddings in a larger space (Gao et al., 2021). Zhang et al. (2022) prove that hard negatives, i.e. data points hard to distinguish from the selected anchor, are key for keeping

a strong gradient signal; however, as learning proceeds, they become orthogonal to the anchor and make the gradient signal close to zero. For this reason, the key idea of MixCSE is to continually generate hard negatives via mixing positive and negative examples, which maintains a strong gradient signal throughout training. Adapting this concept to our retrieval scenario, we construct mixed negative passages as follows:

$$\tilde{\mathbf{e}}_i = \frac{\lambda \mathbf{e}_{p_i^+} + (1 - \lambda) \mathbf{e}_{p_{i,j}^-}}{\|\lambda \mathbf{e}_{p_i^+} + (1 - \lambda) \mathbf{e}_{p_{i,j}^-}\|_2},$$

where $p_{i,j}^-$ is a negative passage chosen at random. We provide the equation of the loss in Appendix B. The main difference with $\mathcal{L}_{\text{mdpr}}$ is the addition of a mixed negative in the denominator and the similarity used (exponential of cosine similarity instead of dot product).

We train mDPR with the original loss and with the MixCSE loss on the concatenation of the provided training set for mDPR and the augmented data obtained via the methods described in §2. We refer to these two variants as *mDPR(AUG)* and *mDPR(AUG) with MixCSE*, respectively.

Ensembling “blackbox” models Following the approaches presented in Litschko et al. (2022), we also ensemble the ranking of some *blackbox* models that directly produce a semantic embedding of the input text. We provide a brief overview of the models included in our ensemble below.

- **DISTIL** (Reimers and Gurevych, 2020) is a teacher-student framework for injecting the knowledge obtained through specialization for semantic similarity from a specialized monolingual transformer (e.g., BERT) into a non-specialized multilingual transformer (e.g., mBERT). For semantic similarity, it first specializes a monolingual (English) teacher encoder using the available semantic sentence-matching datasets for supervision. In the second knowledge distillation step, a pre-trained multilingual student encoder is trained to mimic the output of the teacher model. We benchmark different DISTIL models:
 - $\text{DISTIL}_{\text{use}}$: instantiates the student as the pretrained m-USE (Yang et al., 2020) instance;
 - $\text{DISTIL}_{\text{xlmr}}$: initializes the student model with the pretrained XLM-R (Conneau et al., 2020) transformer;

- $\text{DISTIL}_{\text{dmbert}}$: distills the knowledge from the Sentence-BERT (Reimers and Gurevych, 2019) teacher into a multilingual version of DistilBERT (Sanh et al., 2019), a 6-layer transformer pre-distilled from mBERT.

- **LaBSE** (Language-agnostic BERT Sentence Embeddings Feng et al. (2022)) is a neural dual-encoder framework, trained with parallel data. LaBSE training starts from a pretrained mBERT instance. LaBSE additionally uses standard self-supervised objectives used in the pretraining of mBERT and XLM (Conneau and Lample, 2019): masked and translation language modelling (MLM and TLM).
- **MiniLM** (Wang et al., 2020) is a student model trained by deeply mimicking the self-attention behavior of the last Transformer layer of the teacher, which allows a flexible number of layers for the students and alleviates the effort of finding the best layer mapping.
- **MPNet** (Song et al., 2020) is based on a pre-training method that leverages the dependency among the predicted tokens through permuted language modeling and makes the model see auxiliary position information to reduce the discrepancy between pre-training and fine-tuning.

We produce an ensembling of the *blackbox* models by simply taking an average of the ranks for each of the documents retrieved, which is denoted as *EnsembleRank*.

Oracle Monolingual BM25 (Sparck Jones et al., 2000; Jonesa et al., 2000) This approach is made of two phases: first, we automatically detect the language of the question, then we query the index in the detected language. As a weighting scheme in the vector space model, we choose BM25. It is based on a probabilistic interpretation of how terms contribute to the document’s relevance. It uses exact term matching and the score is derived from a sum of contributions from each query term that appears in the document. We use an *oracle BM25* approach: this naming derives from the fact that we query the index with the answer rather than the question. This was done at training time to increase the probability of the answer to be in the passages consumed by mGEN, so that the generation model

would hopefully learn to extract the answer from its input, rather than generating it from the question only. At inference time, we query the index using the question.

3.2 Answer Generation

Our answer generation modules take a concatenation of the question and the related documents retrieved by the retrieval module as an input and generate an answer. In this section, we first explain the baseline system which is the basis of our proposed approaches and then present our specialization method.

Baseline: mGEN We use mGEN (Multilingual Answer Generator; Asai et al. (2021b)) as the baseline for the answer generation phase. They propose to take mT5 (Xue et al., 2021), a multilingual version of a pretrained transformer-based encoder-decoder model (Raffel et al., 2020), and fine-tune it for multilingual answer generation. The pre-training process of mT5 is based on a variant of masked language modeling named span-corruption, in which the objective is to reconstruct continuously masked tokens in an input sentence (Xue et al., 2021). For fine-tuning, the model is trained on a sequence-to-sequence (seq2seq) task as follows:

$$P(a^L | q^L, P^N) = \prod_i^T p(a_i^L | a_{<i}^L, q^L, P^N)$$

The model predicts a probability distribution over its vocabulary at each time step (i). It is conditioned on the previously generated answer tokens ($a_{<i}^L$), the input question (q^L) and N retrieved passages (P^N). Because of a possible language mismatch between the answer and the passages, it is not possible to extract answers as in existing work in monolingual QA tasks (Karpukhin et al., 2020): for this reason, mGEN opts for directly generating answers instead.

Masked Language Modeling (MLM) Following successful work on language-specialized pre-training via language modeling (Glavaš et al., 2020; Hung et al., 2022), we investigate the effect of running MLM on the language-specific portions of Wikipedia passages (Asai et al., 2021b) and CCNet (Wenzek et al., 2020) with mT5 (Xue et al., 2021). For the extracted texts of all 16 languages, 14 languages are from the released Wikipedia passages and the missing two *surprise* languages

(Tamil, Tagalog) are from CCNet. We additionally clean all language portions by removing email addresses, URLs, extra emojis and punctuations, and selected 7K for training and 0.7K for validation for each language. In this way, we inject both the domain-specific (i.e., Wikipedia knowledge) and language-specific (i.e., 16 languages) knowledge into the multilingual pretrained language model via MLMing as an intermediate specialization step.

Augmentation Data Variants To further investigate model capability on (1) extracting answers from English passages or (2) extracting answers from translated passages, while keeping the Question-Answer pairs in other non-English languages, we conduct experiments on two augmentation data variants: **AUG-QA** and **AUG-QAP**. **AUG-QA** keeps the English passage with the translated Question-Answer pairs, while **AUG-QAP** translates the English passage to the same language as the translated Question-Answer pairs. Detailed examples are shown in Table 1.

4 Experimental Setup

We demonstrate the effectiveness of our proposed COQA systems by comparing them to the baseline models and thoroughly comparing different specialization methods from §1.

Evaluation Task and Measures Our proposed approaches are evaluated in 16 languages, 8 of which are not covered in the training data.⁵ The training and evaluation data are originally from Natural Questions (Kwiatkowski et al., 2019), XOR-TyDi QA (Asai et al., 2021a), and MKQA (Longpre et al., 2020). Data size statistics for each resource and language are shown in Table 2 and 3.

The evaluation results are measured on the competition platform hosted at eval.ai.⁶ The systems are evaluated on two COQA datasets: XOR-TyDi QA (Asai et al., 2021a), and MKQA (Longpre et al., 2020), using token-level F1 (**F1**), as common evaluation practice of open QA systems (Lee et al., 2019). For *non-spacing* languages, we follow the token-level tokenizers⁷ for both predictions and

⁵Languages with training data: English(EN), Arabic(AR), Bengali(BN), Finish(FI), Japanese(JA), Korean(KO), Russian(RU), Telugu(TE). Without training data: Spanish(ES), Khmer(KM), Malay(MS), Swedish(SV), Turkish(TR), Chinese(ZH-CN). Tagalog(TL) and Tamil(TA) are considered as *surprise* languages.

⁶<https://eval.ai/>

⁷Tokenizers for non-spacing languages: Mecab (JA); khmermltk (KM); jieba (ZH-CN).

	AUG-QA	AUG-QAP
QA pair	Q: レゴグループを設立したのは誰ですか？ A: オレ・カーク・クリスチャンセン	
Passage	The Lego Group began manufacturing the interlocking toy bricks in 1949. Movies, games, competitions, and six Legoland amusement parks have been developed under the brand. As of July 2015, 600 billion Lego parts had been produced. In February 2015, Lego replaced Ferrari as Brand Finance’s “world’s most powerful brand”. History. The Lego Group began in the workshop of Ole Kirk Christiansen	レゴグループは1949年に連動おもちゃのレンガの製造を開始しました。映画、ゲーム、競技会、および6つのレゴランド遊園地がこのブランドで開発されました。2015年7月現在、6,000億個のレゴパーツが生産されています。2015年2月、レゴはブランドファイナンスの「世界で最も強力なブランド」としてフェラーリに取って代わりました。歴史。レゴグループは、OleKirkChristiansenのワークショップで始まりました

Table 1: Examples of augmented training instances for **AUG-QA** and **AUG-QAP**. Top row: translated question-answer pair in Japanese. Below are different training examples: (1) **AUG-QA**: the English Wikipedia passage is kept with the translated question-answer pair. (2) **AUG-QAP**: the English Wikipedia passage is translated to the same language as the question-answer pair.

Dataset	Lang	Train size
Natural Questions	en	76635
XOR-TyDi QA	ar	18402
	bn	5007
	fi	9768
	ja	7815
	ko	4319
	ru	9290
	te	6759

Table 2: The training data size for 8 languages.

Dataset	Lang	Dev size	Test size
MKQA (parallel)	12	1758	5000
XOR-TyDi QA	ar	590	1387
	bn	203	490
	fi	1368	974
	ja	1056	693
	ko	1048	473
	ru	910	1018
	te	873	564
Surprise	ta	-	350
	tl	-	350

Table 3: The development and test data size for each language. The data size for MKQA is equal for all 12 languages. Two *surprise* languages are provided without development data.

ground-truth answers. The overall score is calculated by using macro-average scores on XOR-TyDi QA and MKQA datasets, and then taking the average F1 scores of both datasets.

Data We explicitly state that we did not train on the development data or the subsets of the Natural Questions and TyDi QA, which are used to create MKQA or XOR-TyDi QA datasets. This makes all of our proposed approaches fall into the *constrained* setup proposed by the organizers.

For training the mDPR variants, we exploit the organizer’s dataset that was obtained from DPR Natural Questions (Qu et al., 2021) and XOR-TyDiQA gold paragraph data. More specifically, for training and validation, we always use the version of the dataset containing augmented positive

and negative passages obtained from the top 50 retrieval results of the organizer’s mDPR. We merge this dataset with the augmented data, filtering the latter to get 100k samples for each of the 16 languages.

We base our training data for answer generation models on the organizer’s datasets with the top 15 retrieved documents from the coupled retriever. To use automatically generated question-answer pairs for each language from §4 for fine-tuning, we align the format with retrieved results by randomly sampling passages from English Wikipedia as negative contexts,⁸ while we keep the seed documents as positive ones. We explore two ways of merging the positive and negative passages: in the “shuffle” style, the positive passage appears in one of the top 3 documents; in the “non-shuffle” method, the positive passage always appears on the top. However, since these two configurations did not show large differences, we only report the former one in this paper. We also investigated if translating passages into the different 16 languages⁹ may be beneficial with respect to keeping all the passages in English (**AUG-QA**). Due to computational limitations, in our data augmented setting for generation model fine-tuning, we use 2K question-answer pairs with positive/negative passages for each language for our final results.

Hyperparameters and Optimization For multi-lingual dense passage retrieval, we mostly follow the setup provided by the organizers: learning rate $1e-5$ with AdamW (Loshchilov and Hutter, 2019), linear scheduling with warm-up for 300 steps and

⁸For the negative contexts, we use the passages that were used for generating the question-answer pairs (i.e., the first three sub-passages). These were then trimmed down to 100 tokens. We ensure that the answer is not contained in the negative contexts through lowercase string-matching.

⁹Using the same Google Translate API adopted for the QA translation in §4.

dropout rate 0.1. For **mDPR(AUG) with MixCSE** (Zhang et al., 2022), we use $\lambda = 0.2$ and $\tau = 0.05$ for the loss (see Appendix B). We train with a batch size of 16 on 1 GPU for at most 40 epochs, using average rank on the validation data to pick the checkpoints. The training is done independently of mGEN, in a non-iterative fashion.

For retrieving the passages, we use cosine similarity between question and passage across all proposed retrieval models, returning the top 100 passages for each of the questions.

For language-specialized pretraining via MLM, we use AdaFactor (Shazeer and Stern, 2018) with the learning rate $1e - 5$ and linear scheduling with warm-up for 2000 steps up to 20 epochs. For multilingual answer generation fine-tuning, we also mostly keep the setup from the organizers: learning rate $3e - 5$ with AdamW (Loshchilov and Hutter, 2019), linear scheduling with warm-up for 500 steps, and dropout rate as 0.1. We take the top 15 documents from the retrieved results as our input and truncate the input sequence after 16,000 tokens to fit the model into the memory constraints of our available infrastructure.

5 Results and Discussion

Results Overview Results in Table 4 show the comparison between the baseline and our proposed methods on XOR-TyDi QA while Table 5 shows the results on MKQA. While we can see that the additional pretraining on the answer generation model (*mDPR+MLM-14*) helps to outperform the baseline in XOR-TyDi QA, the same approach leads to a degradation in MKQA. None of the proposed methods for the retrieval module improved over the baseline mDPR in both datasets, as shown in Table 6.

Unsupervised vs Supervised Retrieval In all evaluation settings, unsupervised retrieval methods underperform supervised methods by a large margin (see Table 4 and 5). This might be due to the nature of the task, which is to find a document containing an answer, rather than simply finding a document similar to the input question. For this reason, such an objective might not align well with models specialized in semantic similarity (Litschko et al., 2022). Fine-tuning mBERT, however, makes the model learn to focus on retrieving an answer-containing document and not simply retrieving documents similar to the question.

Language Specialization We compare the evaluation results for the fine-tuned answer generation model with and without language specialization (i.e., MLMing): for *XORQA-ar* and *XORQA-te* we have +2.0 and +1.1 percentage points improvement compared to the baseline model (with mT5 trained on 100+ languages). We further distinguish MLM-14 and MLM-16, where the former is trained on the released Wikipedia passages for 14 languages and the latter is trained on the concatenation of Wikipedia passages and CCNet (Wenzek et al., 2020), to which we resort for the two *surprise* languages (Tamil and Tagalog), which were missing in the Wikipedia data. Overall, MLM-14 performs better than MLM-16: we hypothesize that this might be due to the domain difference between text coming from Wikipedia and CCNet: the latter is not strictly aligned with the structured text (i.e., clean) version of Wikipedia passages, and causes a slight drop in performance as we train for 2 additional languages.

Data Augmentation Data augmentation is considered a way to mitigate the performance of low-resource languages while reaching performance on par with high-resource languages (Kumar et al., 2019; Riabi et al., 2021; Shakeri et al., 2021). Two variations are considered: AUG-QA and AUG-QAP, while the former concatenates the XOR-TyDi QA training set with the additional augmented data with translated Question-Answer pairs, and the latter is made from the concatenation of both XOR-TyDi QA training set and the translated Question-Answer-Passage.¹⁰ We assume that by also translating passages, the setting should be closer to test time when the retrieval module can retrieve passages in any of 14 languages (without the two surprise languages). In contrast, in AUG-QA setting, the input passages to the answer generation are always in English. Models trained with additional AUG-QA data could increase the capacity of *seeing* more data for unseen languages, while AUG-QAP may further enhance the ability of the model to generate answers from the translated passages. As expected, models trained with additional augmented data have better performance compared to the ones without. The encouraging finding states that, especially for two *surprise* languages, the language specialized models fine-tuned with both XOR-TyDi

¹⁰XORQA-TyDi QA training set is with 8 languages (see Table 2) and augmented data are with *all* 16 languages included in the test set.

Models	XOR-TyDi QA							Avg.
	ar	bn	fi	ja	ko	ru	te	
mDPR + mGEN (baseline 1)	49.66	33.99	39.54	39.72	25.59	40.98	36.16	37.949
<i>Unsupervised Retrieval</i>								
OracleBM25 + MLM-14	0.34	0.49	0.52	2.56	0.19	0.57	5.16	1.404
EnsembleRank + MLM-14	0.34	0.49	1.33	2.56	0.38	6.27	16.21	3.161
<i>Supervised Retrieval</i>								
mDPR(AUG) with MixCSE + MLM-14	20.94	7.18	15.27	23.16	10.25	19.23	10.53	15.223
mDPR(AUG) + MLM-14	24.99	15.19	20.33	22.31	10.68	18.82	11.97	17.754
mDPR + MLM-14	51.66	31.96	38.68	40.89	25.35	39.87	37.26	37.951
mDPR + MLM-14(XORQA & AUG-QA)	49.41	32.90	37.95	40.97	24.22	39.29	35.76	37.213
mDPR + MLM-14(XORQA & AUG-QAP)	48.79	33.73	38.33	39.87	25.26	39.11	37.94	37.577
mDPR + MLM-16	49.92	31.16	37.20	39.92	24.63	38.78	34.30	36.558
mDPR + MLM-16(XORQA & AUG-QA)	49.45	31.59	38.33	40.44	23.83	38.67	35.92	36.889
mDPR + MLM-16(XORQA & AUG-QAP)	48.21	34.20	38.78	40.76	24.81	39.49	34.37	37.231

Table 4: Evaluation results on XOR-TyDi QA test data with F1 and macro-average F1 scores.

Models	MKQA											Surprise		Avg.	
	ar	en	es	fi	ja	km	ko	ms	ru	sv	tr	zh-cn	ta		tl
mDPR + mGEN (baseline1)	9.52	36.34	27.23	22.70	15.89	6.00	7.68	25.11	14.60	26.69	21.66	13.78	0.00	12.78	17.141
<i>Unsupervised Retrieval</i>															
OracleBM25 + MLM-14	2.80	10.81	3.70	3.29	5.89	1.53	1.51	5.49	1.85	7.42	2.94	1.81	0.00	8.23	4.090
EnsembleRank + MLM-14	6.43	31.66	20.02	17.38	10.68	6.24	4.38	21.03	6.27	21.09	17.13	7.22	0.00	8.39	12.709
<i>Supervised Retrieval</i>															
mDPR(AUG) with MixCSE + MLM-14	4.71	28.06	12.78	8.22	7.92	5.44	2.74	12.90	4.65	13.86	8.38	3.99	0.00	6.72	8.599
mDPR(AUG) + MLM-14	5.64	29.23	17.27	15.51	7.81	5.83	3.38	16.57	6.80	17.21	13.10	4.53	0.00	8.09	10.785
mDPR + MLM-14	8.73	35.32	25.54	20.42	14.27	6.06	6.78	24.10	12.01	25.97	20.27	13.95	0.00	11.14	16.040
mDPR + MLM-14(XORQA & AUG-QA)	8.46	35.12	24.74	19.50	14.38	5.62	7.22	23.24	11.46	24.49	19.67	15.79	0.86	12.18	15.909
mDPR + MLM-14(XORQA & AUG-QAP)	8.48	34.73	25.46	20.09	14.61	5.00	7.42	24.16	12.04	25.61	19.62	15.60	0.00	12.41	16.089
mDPR + MLM-16	8.15	34.14	24.85	19.38	13.73	5.93	6.51	22.21	11.46	24.91	18.82	13.62	0.00	12.59	15.451
mDPR + MLM-16(XORQA & AUG-QA)	8.21	34.06	25.65	20.14	14.22	5.80	6.70	24.40	11.82	25.71	19.92	15.42	0.40	12.36	16.057
mDPR + MLM-16(XORQA & AUG-QAP)	8.08	33.89	24.94	20.50	14.11	5.15	7.15	22.95	12.95	24.93	19.68	15.27	0.14	13.07	15.915

Table 5: Evaluation results on MKQA test dataset and two *surprise* languages with F1 and macro-average F1 scores.

Models	Avg.
mDPR + mGEN (baseline1)	27.55
<i>Unsupervised Retrieval</i>	
OracleBM25 + MLM-14	2.75
EnsembleRank + MLM-wiki14	7.94
<i>Supervised Retrieval</i>	
mDPR(AUG) with MixCSE + MLM-14	11.91
mDPR(AUG) + MLM-14	14.27
mDPR + MLM-14	27.00
mDPR + MLM-14(XORQA & AUG-QA)	26.56
mDPR + MLM-14(XORQA & AUG-QAP)	26.83
mDPR + MLM-16	26.00
mDPR + MLM-16(XORQA & AUG-QA)	26.47
mDPR + MLM-16(XORQA & AUG-QAP)	26.57

Table 6: Results of macro-average F1 for two QA datasets: XOR-TyDi QA, MKQA, and two *surprise* languages.

QA and AUG-QAP drastically improve the performance of these *unseen, low-resource* languages.

mDPR variants results As shown in Table 4 and 5, we can see that the mDPR variants we trained are considerably worse than the baseline. We think this is mainly caused by the limited batch size used (16) which is a constraint due to our infrastructure. The number of samples in a batch is critical for contrastive training, as larger batches

provide a stronger signal due to a higher number of negatives. For this reason, we think that the mDPR variants have not been thoroughly investigated and might still prove beneficial when trained with larger batches.

6 Related Work

Passage Retrieval and Answer Generation To improve the information accessibility, open-retrieval question answering systems are attracting much attention in NLP applications (Chen et al., 2017; Karpukhin et al., 2020). Rajpurkar et al. (2016) were one of the early works to present a benchmark that requires systems to understand a passage to produce an answer to a given question. (Kwiatkowski et al., 2019) presented a more challenging and realistic dataset with questions collected from a search engine. To tackle these complex and knowledge-demanding QA tasks, Lewis et al. (2020) proposed to first retrieve related documents from a given question and use them as additional aids to predict an answer. In particular, they explored a general-purpose fine-tuning recipe for retrieval-augmented generation models, which

combine pretrained parametric and non-parametric memory for language generation. Izacard and Grave (2021), they solved the problem in two steps, first retrieving support passages before processing them with a seq2seq model, and Sun et al. (2021) further extended to the cross-lingual conversational domain. Some works are explored with *translate-then-answer* approach, in which texts are translated into English, making the task monolingual (Ture and Boschee, 2016; Asai et al., 2021a). While this approach is conceptually simple, it is known to cause the *error propagation* problem in which errors of the translation get amplified in the answer generation stage (Zhu et al., 2019). To mitigate this problem, Asai et al. (2021b) proposed to extend Lewis et al. (2020) by using multilingual models for both the passage retrieval (Devlin et al., 2019) and answer generation (Xue et al., 2021).

Data Augmentation Data augmentation is a common approach to reduce the data sparsity for deep learning models in NLP (Feng et al., 2021). For Question Answering (QA), data augmentation has been used to generate paraphrases via *back-translation* (Longpre et al., 2019), to replace parts of the input text with translations (Singh et al., 2019), and to generate novel questions or answers (Riabi et al., 2021; Shakeri et al., 2021; Dugan et al., 2022). In the cross-lingual setting, available data have been translated into different languages (Singh et al., 2019; Kumar et al., 2019; Riabi et al., 2021; Shakeri et al., 2021) and language models have been used to train question and answer generation models (Kumar et al., 2019; Chi et al., 2020; Riabi et al., 2021; Shakeri et al., 2021).

Our approach is different from previous work in Cross-lingual Question Answering task in that it only requires English passages to augment the training data, as answers are generated automatically from the trained model by Dugan et al. (2022). In addition, our filtering heuristics remove incorrectly generated question-answer pairs, which allows us to keep only question-answer pairs with answers that are more likely to be translated correctly, thus limiting the problem of error propagation.

7 Reproducibility

To ensure full reproducibility of our results and further fuel research on COQA systems, we release the model within the Huggingface repository as the publicly available multilingual pretrained language model specialized in 14 and 16

languages.¹¹ We also release our code and data, which make our approach completely transparent and fully reproducible. All resources developed as part of this work are publicly available at: <https://github.com/umanlp/ZusammenQA>.

8 Conclusion

We introduced a framework for a cross-lingual open-retrieval question-answering system, using data augmentation with specialized models in a *constrained* setup. Given a question, we first retrieved top relevant documents and further generated the answer with the specialized models (i.e., MLM-ing on Wikipedia passages) along with the augmented data variants. We demonstrated the effectiveness of data augmentation techniques with language- and domain-specialized additional training, especially for resource-lean languages. However, there are still remaining challenges, especially in the retrieval model training with limited computational resources. Our future efforts will be to focus on more efficient approaches of both multilingual passage retrieval and multilingual answer generation (Abdaoui et al., 2020) with the investigation of different data augmentation techniques (Zhu et al., 2019). We hope that our generated QA language resources with the released models can catalyze the research focus on resource-lean languages for COQA systems.

References

- Amine Abdaoui, Camille Pradel, and Grégoire Sigel. 2020. Load what you need: Smaller versions of multilingual BERT. In *Proceedings of SustaiNLP: Workshop on Simple and Efficient Natural Language Processing*, pages 119–123, Online. Association for Computational Linguistics.
- Akari Asai, Jungo Kasai, Jonathan H. Clark, Kenton Lee, Eunsol Choi, and Hannaneh Hajishirzi. 2021a. XOR QA: cross-lingual open-retrieval question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 547–564. Association for Computational Linguistics.

- Akari Asai, Xinyan Yu, Jungo Kasai, and Hanna Hajishirzi. 2021b. One question answering model for many languages with cross-lingual dense passage

¹¹MLM-14: <https://huggingface.co/umanlp/mt5-mlm-wiki14>; MLM-16: <https://huggingface.co/umanlp/mt5-mlm-16>

- retrieval. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 7547–7560.
- Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Xiaodong Liu, Yu Wang, Jianfeng Gao, Songhao Piao, Ming Zhou, et al. 2020. **Unilmv2: Pseudo-masked language models for unified language model pre-training**. In *International Conference on Machine Learning*, pages 642–652. PMLR.
- 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 (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada. Association for Computational Linguistics.
- Zewen Chi, Li Dong, Furu Wei, Wenhui Wang, Xianling Mao, and Heyan Huang. 2020. **Cross-lingual natural language generation via pre-training**. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):7570–7577.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. **Unsupervised cross-lingual representation learning at scale**. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.
- Alexis Conneau and Guillaume Lample. 2019. **Cross-lingual language model pretraining**. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.
- Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Jimmy Lin. 2021. **Ms marco: Benchmarking ranking models in the large-data regime**. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1566–1576.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. **BERT: Pre-training of deep bidirectional transformers for language understanding**. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. **Unified language model pre-training for natural language understanding and generation**. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.
- Liam Dugan, Eleni Miltsakaki, Shriyash Upadhyay, Etan Ginsberg, Hannah Gonzalez, Dayheon Choi, Chuning Yuan, and Chris Callison-Burch. 2022. **A feasibility study of answer-agnostic question generation for education**. *arXiv preprint arXiv:2203.08685*.
- Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Ariavzhagan, and Wei Wang. 2022. **Language-agnostic BERT sentence embedding**. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.
- Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Edouard Hovy. 2021. **A survey of data augmentation approaches for NLP**. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 968–988, Online. Association for Computational Linguistics.
- Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. **SimCSE: Simple contrastive learning of sentence embeddings**. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Goran Glavaš, Mladen Karan, and Ivan Vulić. 2020. **XHate-999: Analyzing and detecting abusive language across domains and languages**. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6350–6365, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Chia-Chien Hung, Anne Lauscher, Ivan Vulić, Simone Paolo Ponzetto, and Goran Glavaš. 2022. **Multi²WOZ: A robust multilingual dataset and conversational pretraining for task-oriented dialog**. Accepted for publication in *the North American Chapter of the Association for Computational Linguistics: NAACL 2022*.
- 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*, pages 874–880, Online. Association for Computational Linguistics.
- K Sparck Jonesa, S Walkerb, and SE Robertsonb. 2000. **A probabilistic model of information retrieval: development and comparative experiments part 2**. *Information Processing and Management*, 36(809):840.
- Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. **The state and fate of linguistic diversity and inclusion in the NLP world**. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

- Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick 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)*, pages 6769–6781, Online. Association for Computational Linguistics.
- Vishwajeet Kumar, Nitish Joshi, Arijit Mukherjee, Ganesh Ramakrishnan, and Preethi Jyothi. 2019. [Cross-lingual training for automatic question generation](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4863–4872, Florence, Italy. Association for Computational Linguistics.
- Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. [Natural questions: A benchmark for question answering research](#). *Transactions of the Association for Computational Linguistics*, 7:452–466.
- Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. [Latent retrieval for weakly supervised open domain question answering](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.
- 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, December 6-12, 2020, virtual*.
- Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. [On the sentence embeddings from pre-trained language models](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9119–9130, Online. Association for Computational Linguistics.
- Robert Litschko, Ivan Vulić, Simone Paolo Ponzetto, and Goran Glavaš. 2022. [On cross-lingual retrieval with multilingual text encoders](#). *Information Retrieval Journal*, 25(2):149–183.
- Ekaterina Logina, Stalin Varanasi, and Günter Neumann. 2021. [Towards end-to-end multilingual question answering](#). *Inf. Syst. Frontiers*, 23(1):227–241.
- Shayne Longpre, Yi Lu, and Joachim Daiber. 2020. [MKQA: A linguistically diverse benchmark for multilingual open domain question answering](#). *CoRR*, abs/2007.15207.
- Shayne Longpre, Yi Lu, Zhucheng Tu, and Chris DuBois. 2019. [An exploration of data augmentation and sampling techniques for domain-agnostic question answering](#). In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 220–227, Hong Kong, China. Association for Computational Linguistics.
- Ilya Loshchilov and Frank Hutter. 2019. [Decoupled weight decay regularization](#). In *International Conference on Learning Representations*.
- Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. 2021. [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, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 5835–5847. Association for Computational Linguistics.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. [Exploring the limits of transfer learning with a unified text-to-text transformer](#). *Journal of Machine Learning Research*, 21(140):1–67.
- Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. [SQuAD: 100,000+ questions for machine comprehension of text](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.
- Nils Reimers and Iryna Gurevych. 2019. [Sentence-BERT: Sentence embeddings using Siamese BERT-networks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.
- Nils Reimers and Iryna Gurevych. 2020. [Making monolingual sentence embeddings multilingual using knowledge distillation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online. Association for Computational Linguistics.
- Arij Riabi, Thomas Scialom, Rachel Keraron, Benoît Sagot, Djamel Seddah, and Jacopo Staiano. 2021. [Synthetic data augmentation for zero-shot cross-lingual question answering](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7016–7030, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. [Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter](#). *CoRR*, abs/1910.01108.

- Siamak Shakeri, Noah Constant, Mihir Kale, and Linting Xue. 2021. [Towards zero-shot multilingual synthetic question and answer generation for cross-lingual reading comprehension](#). In *Proceedings of the 14th International Conference on Natural Language Generation*, pages 35–45, Aberdeen, Scotland, UK. Association for Computational Linguistics.
- Noam Shazeer and Mitchell Stern. 2018. [Adafactor: Adaptive learning rates with sublinear memory cost](#). In *International Conference on Machine Learning*, pages 4596–4604. PMLR.
- Jasdeep Singh, Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2019. [XLDA: cross-lingual data augmentation for natural language inference and question answering](#). *CoRR*, abs/1905.11471.
- Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. [Mpnet: Masked and permuted pre-training for language understanding](#). In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.
- K. Sparck Jones, S. Walker, and S.E. Robertson. 2000. [A probabilistic model of information retrieval: development and comparative experiments: Part 1](#). *Information Processing & Management*, 36(6):779–808.
- Weiwei Sun, Chuan Meng, Qi Meng, Zhaochun Ren, Pengjie Ren, Zhumin Chen, and Maarten de Rijke. 2021. [Conversations powered by cross-lingual knowledge](#). In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1442–1451.
- Ferhan Ture and Elizabeth Boschee. 2016. [Learning to translate for multilingual question answering](#). In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 573–584, Austin, Texas. Association for Computational Linguistics.
- Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. [Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers](#). In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.
- Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. [CCNet: Extracting high quality monolingual datasets from web crawl data](#). In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.
- Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2020. [Scalable zero-shot entity linking with dense entity retrieval](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6397–6407, Online. Association for Computational Linguistics.
- Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. [mT5: A massively multilingual pre-trained text-to-text transformer](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.
- Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. 2020. [Multilingual universal sentence encoder for semantic retrieval](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 87–94, Online. Association for Computational Linguistics.
- Yanzhao Zhang, Richong Zhang, Samuel Mensah, Xudong Liu, and Yongyi Mao. 2022. [Unsupervised sentence representation via contrastive learning with mixing negatives](#).
- Junnan Zhu, Qian Wang, Yining Wang, Yu Zhou, Jijun Zhang, Shaonan Wang, and Chengqing Zong. 2019. [NCLS: Neural cross-lingual summarization](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3054–3064, Hong Kong, China. Association for Computational Linguistics.
- Pierre Zweigenbaum, Serge Sharoff, and Reinhard Rapp. 2018. [Overview of the third bucc shared task: Spotting parallel sentences in comparable corpora](#). In *Proceedings of 11th Workshop on Building and Using Comparable Corpora*, pages 39–42.

A Data Augmentation Example

Figure 2 shows an example of a Wikipedia passage about *An American in Paris*. The passage in orange is the set of sentences whose length does not exceed 512 tokens, which is the first of three sub-passages used for generating question-answer pairs. The generated pairs can be seen at the bottom of the figure. Questions and answers highlighted in red are those that satisfy the filtering heuristics detailed in §2.2. These are then translated into other languages.

<p>Title: An American in Paris</p> <p>URL: https://en.wikipedia.org/wiki?curid=309</p> <p>Text: An American in Paris is a jazz-influenced orchestral piece by American composer George Gershwin written in 1928. It was inspired by the time that Gershwin had spent in Paris and evokes the sights and energy of the French capital in the 1920s. Gershwin composed “An American in Paris” on commission from conductor Walter Damrosch. He scored the piece for the standard instruments of the symphony orchestra plus celesta, saxophones, and automobile horns. He brought back some Parisian taxi horns for the New York premiere of the composition, which took place on December 13, 1928 in Carnegie Hall, with Damrosch conducting the New York Philharmonic. He completed the orchestration on November 18, less than four weeks before the work’s premiere. He collaborated on the original program notes with critic and composer Deems Taylor. Gershwin was attracted by Maurice Ravel’s unusual chords, and Gershwin went on his first trip to Paris in 1926 ready to study with Ravel. After his initial student audition with Ravel turned into a sharing of musical theories, Ravel said he could not teach him, saying, “Why be a second-rate Ravel when you can be a first-rate Gershwin?” While the studies were cut short, that 1926 trip resulted in a piece entitled “Very Parisienne”, the initial version of “An American in Paris”, written as a ‘thank you note’ to Gershwin’s hosts, Robert and Mabel Shirmer. Gershwin called it “a rhapsodic ballet”; it is written freely and in a much more modern idiom than his prior works. Gershwin strongly encouraged Ravel to come to the United States for a tour. To this end, upon his return to New York, Gershwin joined the efforts of Ravel’s friend Robert Schmitz, a pianist Ravel had met during the war, to urge Ravel to tour the U.S. Schmitz was the head of Pro Musica, promoting Franco-American musical relations, and was able to offer Ravel a \$10,000 fee for the tour, an enticement Gershwin knew would be important to Ravel. ...</p>
<p>Question: When was an American in Paris written? Answer: 1928 Type: Number</p> <p>Question: When did George Gershwin write an American in Paris? Answer: the 1920s Type: Contains number</p> <p>Question: Who was the conductor of “An American in Paris”? Answer: Walter Damrosch Type: Who</p> <p>Question: What was the name of the instrument that Gershwin scored for? Answer: automobile horns</p> <p>Question: What was the name of the orchestral piece Gershwin composed in 1928? Answer: New York Philharmonic</p> <p>Question: When did Gershwin complete the orchestration of “An American in Paris”? Answer: November 18 Type: Date</p> <p>Question: Who did Gershwin collaborate on the original program notes with? Answer: Deems Taylor Type: Who</p>

Figure 2: Example English question-answer pairs (on the bottom) generated from the highlighted text (in yellow) in the passage. The highlighted question-answer pairs (in red) are those that were kept after filtering.

B MixCSE Loss

The MixCSE loss described in 3.1 is given by:

$$\mathcal{L}_{\text{mixcse}} = -\log \frac{\exp(\cos(\mathbf{e}_{q_i}, \mathbf{e}_{p_i^+})/\tau)}{\exp(\cos(\mathbf{e}_{q_i}, \mathbf{e}_{p_i^+})/\tau) + \sum_{j=1}^n \exp(\cos(\mathbf{e}_{q_i}, \mathbf{e}_{p_{i,j}^-})/\tau) + \exp(\cos(\mathbf{e}_{q_i}, \text{SG}(\tilde{\mathbf{e}}_i))/\tau)}$$

where τ is a fixed temperature and SG is the stop-gradient operator, which prevents backpropagation from flowing into the mixed negative ($\tilde{\mathbf{e}}_i$).