Multilingual Transfer Learning for QA Using Translation as Data Augmentation

Mihaela Bornea, Lin Pan, Sara Rosenthal, Radu Florian, Avirup Sil
IBM Research AI, Thomas J. Watson Research Center, Yorktown Heights, NY 10598
{mabornea,panl,sjrosenthal,raduf,avi}@us.ibm.com

Abstract

Prior work on multilingual question answering has mostly focused on using large multilingual pre-trained language models (LM) to perform zero-shot language-wise learning: train a QA model on English and test on other languages. In this work, we explore strategies that improve cross-lingual transfer by bringing the multilingual embeddings closer in the semantic space. Our first strategy augments the original English training data with machine translation-generated data. This results in a corpus of multilingual silver-labeled QA pairs that is 14 times larger than the original training set. In addition, we propose two novel strategies, language adversarial training and language arbitration framework, which significantly improve the (zero-resource) cross-lingual transfer performance and result in LM embeddings that are less language-variant. Empirically, we show that the proposed models outperform the previous zero-shot baseline on the recently introduced multilingual MLQA and TyDi QA datasets.

Wikipedia Page: The Reader (2008 film)

English

Context: Hanna receives a **life** sentence for her admitted leadership role in the church deaths, while the other defendants are sentenced to four years and three months each.
Question: What was Hanna's prison sentence?
Prior work predictions: *four years and three months each*
This work predictions: **life**

Multilingual

Context: Hanna es declarada culpable y sentenciada **a cadena perpetua**, mientras que sus compañeras reciben sentencias de cuatro años de cárcel.
Question: Welche Gefängnisstrafe erhielt Hanna?
Prior work predictions: *cuatro años*
This work predictions: **a cadena perpetua**

Figure 1: Examples from the MLQA dataset. Prior zero-shot (ZS) learning models (Lewis (2020)) cannot answer these examples correctly whereas our proposed models (LAF and AT) can. The correct answer is underlined in the context.

Introduction

Recent advances in open domain question answering (QA) have mostly revolved around machine reading comprehension (MRC) where the task is to read and comprehend a given text and then answer questions based on it. However, most recent work in MRC has only been in English *e.g.* SQuAD (Rajpurkar et al. 2016; Rajpurkar, Jia, and Liang 2018), HotpotQA (Yang et al. 2018) and Natural Questions (Kwiatkowski et al. 2019). Significant performance gains and the state-of-the-art (SOTA) on these datasets are credited to large pre-trained language models (Devlin et al. 2019; Radford et al. 2019; Yang et al. 2019b).

Multilingual BERT (mBERT), which is trained on Wikipedia articles from 104 languages and equipped with a 120k shared wordpiece vocabulary, has encouraged a lot of progress on cross-lingual tasks *e.g.* XNLI (Conneau et al. 2018), NER (Keung, Lu, and Bhardwaj 2019; Wu and Dredze 2019) and QA (Artetxe, Ruder, and Yogatama 2019; Cui et al. 2019b; He et al. 2018) by performing *zero-shot* training: train on one language and test on unseen target languages.

In this work, we focus on multilingual QA and, in particular, on two recent large-scale datasets: MLQA (Lewis et al. 2020) and TyDi QA[1] (Clark et al. 2020). Both datasets contain English QA pairs but also examples from 13 other diverse languages.

Some examples are shown in Figure 1. MLQA evaluates two challenging scenarios: 1) *Cross-Lingual Transfer (XLT)* when the question and the context are in the same language, and 2) *Generalized Cross-lingual Transfer (G-XLT)* when the question is in one language (eg. En) and the context is in another language (eg. De). TyDi QA is designed for *XLT* only. The two datasets are challenging for multilingual QA due to the large number of languages and the variety of linguistic phenomena they encompass (e.g. word order, re-duplication, grammatical meanings).

Ideally, we want to build QA systems for all existing languages but it is impractical to collect manually labeled training data for all of them. In the absence of labeled data, (Clark et al. 2020) suggested several research directions for pushing the boundaries in multilingual QA, including zero-shot QA, exploring data augmentation with machine

[1]TyDi QA in our paper refers to the Gold Passage task.

translation, as well as effective transfer learning. These are avenues we explore in our work in addition to asking the following research questions:

1. Is a large pre-trained LM sufficient for zero-shot multilingual QA?

Prior work proposes zero-shot transfer learning from English SQuAD data (Rajpurkar et al. 2016) to other languages using *only* a pre-trained LM and competitive results are achieved on MLQA (Lewis et al. 2020) and TyDi QA (Clark et al. 2020). We venture beyond zero-shot training by first exploring data augmentation (Alberti et al. 2019) on top of their underlying model. We achieve this by using translation methodologies (Yarowsky, Ngai, and Wicentowski 2001) to augment the English training data. We use machine translation to obtain additional silver labeled data allowing us to improve cross-lingual transfer at a low cost. Our approach introduces several multilingual extensions to the SQuAD training data: translating just the questions but keeping the context in English, translating just the context but keeping the question in English, and translating the question *and* the context to other languages. This enables us to augment the original English human-labeled training examples with 14 times more multilingual silver-labeled QA pairs.

2. Can we bring language-specific embeddings in multilingual LMs closer for effective cross-lingual transfer?

Our hypothesis is that we can make the cross-lingual QA transfer more effective if we can bring the embeddings in a multilingual pre-trained LM closer to each other in the same semantic space. To answer a question in French it should suffice to train the system on Hindi and not be necessary to train a system on the target language: hence, French and Hindi should look as if they are the same language. We propose two approaches to explore cross-lingual transfer:

In our first approach, we propose a novel strategy based on adversarial training (AT) (Miyato, Dai, and Goodfellow 2017; Chen et al. 2018; Yang et al. 2019a). We investigate how the addition of a language-adversarial task during QA finetuning for a pre-trained LM can significantly improve the cross-lingual transfer performance while causing the embeddings in the LM to become less language-dependent.

In our second approach, we develop a novel Language Arbitration Framework (LAF) to consolidate the embedding representation across languages using properties of the translation. We train additional auxiliary tasks *e.g.* making sure an English question and its translation in Arabic produce the same answer when they see the same input context in Spanish. The intuition behind language arbitration is that while we are training the model on English and translated examples, the proposed multilingual objectives bring the language-specific embeddings closer to the English embeddings.

Overall, our main contributions in this paper are as follows:

- We create a new translation dataset which has *14 times* more multilingual silver-labeled QA pairs than SQuAD.
- We present an adversarial training approach and a language arbitration framework to bring the LM embeddings closer to each other to improve cross-lingual QA transfer.
- We achieve statistically significant improvements compared to prior work (Lewis et al. 2020; Clark et al. 2020) with all of our models.

Multilingual Question Answering

In this section, we briefly discuss the LM and QA models. These are the foundations applied to our approach.

Pre-trained Language Model

Given a token sequence $\mathbf{X} = [x_1, x_2, \ldots, x_T]$, we choose mBERT, a deep Transformer (Vaswani et al. 2017) network, which outputs a sequence of contextualized token representations $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_T]$.

$$\mathbf{h}_1, \ldots, \mathbf{h}_T = mBERT(x_1, \ldots, x_T) \quad (1)$$

mBERT has 12 layers each with 12 heads and $\mathbf{h}_t \in \mathbb{R}^{768}$. It is pre-trained on 104 languages and produces SOTA results on many cross-lingual tasks (Conneau et al. 2018; Keung, Lu, and Bhardwaj 2019).

Underlying QA model: mBERT$_{QA}$

We build mBERT$_{QA}$, our underlying QA model, as described in (Lewis et al. 2020; Devlin et al. 2019). To create the input sequence we concatenate the `[CLS]`, question, `[SEP]` and context tokens. mBERT$_{QA}$ adds two dense layers followed by a *softmax* on top of mBERT for answer extraction:

$$\boldsymbol{\alpha}_b = softmax(\mathbf{H}\mathbf{W}_1),$$
$$\boldsymbol{\alpha}_e = softmax(\mathbf{H}\mathbf{W}_2),$$

where $\mathbf{W}_1$, $\mathbf{W}_2 \in \mathbb{R}^{768\times1}$, and $\mathbf{H} \in \mathbb{R}^{T\times768}$. $\boldsymbol{\alpha}_b^t$ and $\boldsymbol{\alpha}_e^t$ denote the probability of the t^{th} token in the sequence being the answer start and end, respectively. These two layers are trained during the finetuning stage using the cross entropy loss:

$$\mathcal{L}_{QA} = -\frac{1}{2}(\sum_{t=1}^{T} \mathbb{1}(\mathbf{b}_t) \log \boldsymbol{\alpha}_b^t + \sum_{t=1}^{T} \mathbb{1}(\mathbf{e}_t) \log \boldsymbol{\alpha}_e^t) \quad (2)$$

where $\mathbb{1}(\mathbf{b})$ and $\mathbb{1}(\mathbf{e})$ are one-hot vectors from the ground truth offsets of the answer start and end.

Prior work - Zero-shot (ZS) Learning: Both (Lewis et al. 2020) and (Clark et al. 2020) propose zero-shot learning for multilingual QA by training on English QA data (SQuAD v1.1) and testing on all other languages. This is our basic model and the baseline setting. We train mBERT$_{QA}$ with examples of the form (Q_{En}, C_{En}, A_{En}) where $A_{En} \subset C_{En}$. During inference, we use the trained model to extract the answer span A_y from C_y where y is any language.

Models

In this section, we outline our improvements on top of the prior work on MLQA.

Dataset	Avg. # of Words		Question Type Frequency								# Q-A Pairs
	Ques.	Ans.	Why	How	What	When	Where	Who	Which	OTHER	
SQuAD v1.1	10.1	3.2	1,194	8,082	37,506	5,414	3,261	8,366	4,146	19,630	87,599
T (Q)	11.9	3.2	7,164	48,492	225,036	32,484	19,566	50,196	24,876	117,780	525,594
T (C)	10.1	4.4	5,742	39,668	190,309	26,975	16,225	42,951	21,522	98,298	441,690
T (Q+C)	12.1	4.4	5,742	39,668	190,309	26,975	16,225	42,951	21,522	98,298	441,690
T(All)	11.6	4.1	16,260	111,664	530,642	75,606	45,494	119,366	59,628	275,116	1,233,776

Table 1: Comparing our original traning data SQuAD v1.1 with our augmented training data using translation techniques. The Question Type is based on the first word in the question.

Algorithm 1 Pseudo-code for adversarial training on the multilingual QA task.

Require: $\langle Q^l_{En}, C_{En}\rangle, (b, e), L, D, \text{MBERT}_{QA}, \eta$.

$\{Q^l_{En}$ is the translated question (or English), C_{En} English context. b and e are the correct answer start and end positions. L is the language label for the question, D is discriminator, and $h(\text{MBERT}_{QA})$ is the question representation from the MBERT_{QA} model, and learning rate η.$\}$

for #epochs **do**
  for #steps **do**
    $\langle Q^l_{En}, C_{En}\rangle_i, (b, e)_i, L_i$ ▷ Sample in batch of data
    $(\boldsymbol{\alpha}_b, \boldsymbol{\alpha}_e)_i \leftarrow \text{MBERT}_{QA}(\langle Q^l_{En}, C_{En}\rangle_i)$ ▷ QA predictions from MBERT_{QA}
    $p_{l_i} \leftarrow D(h(\text{MBERT}_{QA}))$ ▷ discriminator language predictions
    $\eta \bigtriangledown_{\theta_{QA}} (\mathcal{L}_{QA} + \mathcal{L}_{adv})$ ▷ update QA model with Eq 2 and 4
    $p_{l_i} \leftarrow D(h(\text{MBERT}_{QA}))$ ▷ discriminator language predictions
    $\eta \bigtriangledown_{\theta_D} \mathcal{L}_D$ ▷ update Discriminator with Eq 3
  end for
end for

Data Augmentation with Translation

Our first approach beyond zero-shot QA is to introduce data-augmentation (Yu et al. 2018; Alberti et al. 2019) based models. Since we only have English examples to train our system on, we expand our training data and explore several translation-based data augmentation models for MLQA. Table 1 shows statistics for the different datasets. We use the IBM Watson Language Translator [2] to:

1. Translate (Q+C):

We pick a language $l \in L$ where $L = \{De, Es, Ar, Hi, Zh\}$[3] and translate (Q_{En}, C_{En}, A_{En}) to create examples $(Q^l_{En}, C^l_{En}, A^l_{En})$ in that language. We do this for each of the 5 languages. Note, Q^l_{En} and C^l_{En} are the translations of Q_{En} and C_{En} and $A^l_{En} \in C^l_{En}$ is the translated answer, all in language l. In order to obtain the alignment of the gold answer A_{En} in the translated context C^l_{En}, we place pseudo HTML tags around A_{En} and then translate C_{En}. Note that the main challenge of this strategy is the answer alignment step and we only keep the translated examples where this succeeds. The number of translated examples we obtained is $87,062$ for German, $77,759$ for Spanish, $84,185$ for Arabic, $20,981$ for Hindi and $84,104$ for Chinese. The final data set including English has $441,690$ examples. The percentage of reduced question type ranges from 14% (Which) to 20% (Why).

2. Translate(Q) : *Only* Q_{En} is translated to other languages leaving C_{En} intact to create examples $\cup_l(Q^l_{En}, C_{En}, A_{En})$. This data augmentation strategy produces a more accurate dataset since it does not require the answer alignment stage which can be error-prone. We translate every Q_{En} to 5 other languages and we obtain a dataset of $525,594$ examples, which is 6 times larger than SQuAD v1.1. T(Q) increases the average number of words in the question by 1.8.

3. Translate (C): We *only* translate C_{En} to other languages to create $\cup_l(Q_{En}, C^l_{En}, A^l_{En})$. We use the same answer alignment strategy as in *Translate (Q+C)* to generate the gold answer A^l_{En} for the translated examples in L. We obtain $441,690$ examples (same as Translate (Q+C)). T(C) increases the average number of words in the answer by 1.2.

4. Translate(ALL): We combine the data from all the 3 strategies together to create a meta-translation model with $1,233,776$ examples, 14 times larger than SQuAD.

Adversarial Training

Translation-based strategies provide ample scope for MBERT_{QA} to train on plenty of $\{Q, C, A\}$ examples where Q and C can be in different languages. However, it can still be challenging as new languages can continuously be added to the model requiring optimal MT systems in all languages. Therefore, it is important to explore bringing the embeddings of different languages in MBERT close to each other to achieve effective cross-lingual transfer. For this purpose, we introduce a novel multilingual adversarial training (AT) method inspired by (Goodfellow et al. 2014). The goal is to fine-tune MBERT so that its embeddings become as *language-invariant* as possible. Algorithm 1 provides an overview of this approach.

[2]https://www.ibm.com/watson/services/language-translator/

[3]Our translation api does not support Vietnamese and Swahili.

Algorithm 2 Pseudo-code for our language arbitration framework for the multilingual QA task.

Require: $\langle Q_{En}, C_{En}\rangle, \langle Q^l_{En}, C_{En}\rangle, (b, e)$, MBERT_{QA}, η $\{Q^l_{En}$ is the translated question, C_{En} English context. b and e are correct answer start and end, $h(\text{MBERT}_{QA})$ is the question representation from the MBERT_{QA} and η is the learning rate .$\}$

for #epochs **do**
  for #steps **do**
    $\langle Q_{EN}, C_{En}\rangle_i, \langle Q^l_{En}, C_{En}\rangle_i, (b, e)_i$ ▷ Sample in batch of data
    $(\boldsymbol{\alpha}^{En}_b, \boldsymbol{\alpha}^{En}_e)_i \leftarrow \text{MBERT}_{QA}(\langle Q_{En}, C_{En}\rangle_i)$ ▷ generate predictions from the QA model for En
    $(\boldsymbol{\alpha}^{l}_b, \boldsymbol{\alpha}^{l}_e)_i \leftarrow \text{MBERT}_{QA}(\langle Q^l_{En}, C_{En}\rangle_i)$ ▷ generate predictions from the QA model on language l
    $\eta \bigtriangledown_{\theta_{QA}} (\mathcal{L}^{En}_{QA} + \mathcal{L}^{l}_{QA})$ ▷ update QA model with Eq 2 for En and language l
    $(\boldsymbol{b^{En}}, \boldsymbol{e^{En}})_i \leftarrow argmax((\boldsymbol{\alpha}^{En}_b, \boldsymbol{\alpha}^{En}_e)_i)$ ▷ find the begin and end of answer when question in En
    $\eta \bigtriangledown_{\theta_{QA}} \mathcal{L}_{PSA}$ ▷ update QA model with with the PSA loss in Eq 5
    $(\overline{h}_{Q_{En}})_i \leftarrow h(\text{MBERT}_{QA}(\langle Q_{En}, C_{En}\rangle_i))$ ▷ get question representation for En
    $(\overline{h}_{Q^l_{En}})_i \leftarrow h(\text{MBERT}_{QA}(\langle Q^l_{En}, C_{En}\rangle_i))$ ▷ get question representation for language l
    $\eta \bigtriangledown_{\theta_{QA}} \mathcal{L}_{QS}$ ▷ update QA model with the QS loss in Eq 6
  end for
end for

Concretely, we use the **Translate(Q)** strategy outlined in the previous section, and for every $\{Q_{En}, C_{En}, A_{En}\}$, we derive examples of $\{Q^l_{En}, C_{En}, A_{En}\}$, where the question is translated. All the examples are added to the training data. The discriminator D of the AT model is trained to classify the question representation in different languages to the correct language label $L \in [En, De, Es, Ar, Hi, Zh]$. We use the [CLS] token to derive a single question representation as input for D and train with cross-entropy loss:

$$\mathcal{L}_D = -\sum_{l=1}^{L} \mathbb{1}(\mathbf{g}_l) \log \mathbf{p}_l, \tag{3}$$

where $\mathbb{1}(\mathbf{g})$ is a one-hot vector for the ground-truth language labels, and $\mathbf{p}$ are the language predictions from the model. D is implemented as a multilayer perceptron.

Under the AT objective, the underlying QA model, in addition to the QA objective, is trained to also minimize the KL-divergence between the uniform distribution, and the language labels predicted by the discriminator.

$$\mathcal{L}_{adv} = -\sum_{l=1}^{L} KL[U(\mathbf{g}_l)|| \log \mathbf{p}_l], \tag{4}$$

$\mathcal{L}_{adv}$ encourages the LM embeddings to appear uniform to the discriminator, across all languages. In contrast, $\mathcal{L}_D$ drives the discriminator to recognize the language. During training, in each step, we first update MBERT_{QA} with $\mathcal{L}_{QA} + \mathcal{L}_{adv}$ (See Eq 2 for $\mathcal{L}_{QA}$) while fixing the parameters of the discriminator (Alg. 1 line 6), and then update the discriminator with $\mathcal{L}_D$ fixing those of MBERT_{QA} (line 8).

In addition to performing AT using all 6 languages, **AT (en-all)**, we also conduct experiments picking just one random language (*e.g.* $l = Zh$) to perform **AT (en-zh)**.

Language Arbitration Framework

In this section, we explore an alternative approach for bringing the language-specific embeddings closer to each other using a novel Language Arbitration Framework (LAF) to train a multilingual QA model. Just like a regular human *arbitrator*, LAF's job at the end of training is to make sure the same question in different languages produce the same answer while maintaining that the underlying representation of the questions are the same. Similar to the AT method, **Translate(Q)** is used to generate our training examples. For every $\{Q_{En}, C_{En}, A_{En}\}$ in the original English dataset, we derive an augmented training set with example pairs $(\{Q_{En}, C_{En}, A_{En}\}, \{Q^l_{En}, C_{En}, A_{En}\})$ where the question is translated to language $l \in L$. Training of LAF proceeds with such example pairs and exploits properties of the translation to consolidate the LM embeddings. In addition to training the base MBERT_{QA} model on English and the translation, using the standard objective from Eq (2), LAF also performs the following objectives during training:

1. Produce the same answer (PSA): PSA encourages the translation $\langle Q^l_{En}, C_{En}\rangle$ to produce the same answer as the original example $\langle Q_{En}, C_{En}\rangle$, for all languages $l \in L$. We run MBERT_{QA} on English and the translation. Then, in addition to computing $\mathcal{L}_{QA}$ (Equation 2) we compute the additional loss:

$$\mathcal{L}_{PSA} = -\frac{1}{2}\left(\sum_{t=1}^{T} \mathbb{1}(\mathbf{b}^{En}_t) \log \boldsymbol{\alpha}^{l^t}_b + \sum_{t=1}^{T} \mathbb{1}(\mathbf{e}^{En}_t) \log \boldsymbol{\alpha}^{l^t}_e\right) \tag{5}$$

$\mathbb{1}(\mathbf{b}^{En}_t)$ and $\mathbb{1}(\mathbf{e}^{En}_t)$ are one-hot vectors indicating the answer start and end positions predicted by the MBERT_{QA} for $\langle Q_{En}, C_{En}\rangle$. $\boldsymbol{\alpha}^l_b$ and $\boldsymbol{\alpha}^l_e$ denote the answer begin and end probability predicted by MBERT_{QA} for $\langle Q^l_{En}, C_{En}\rangle$. While **Translate(Q)** optimizes the standard $\mathcal{L}_{QA}$ objective on translated data, $\mathcal{L}_{PSA}$ uses the English predictions for additional supervision and brings the LM embeddings closer by maintaining agreement between English and the translation. This is beneficial in cases where there is partial overlap between the English predicted answer and the gold label.

2. Produce the same answer and question similarity (PSA+QS): In this approach, in addition to the PSA loss, we also compute the *cosine-similarity* between Q_{En} and Q^l_{En} in all languages $l \in L$. The intuition is that the cosine similarity of translations should be high, encouraging the embeddings to move even closer to each other.

To obtain a single question representation, $\overline{h}_{Q_{En}}$ for En

and $\overline{h}_{Q^l_{En}}$ for language l, we perform average pooling over the hidden vectors for the question tokens from MBERT.

$$\mathcal{L}_{QS} = 1 - cosine(\overline{h}_{Q_{En}}, \overline{h}_{Q^l_{En}}) \quad (6)$$

In addition to performing PSA and PSA+QS in *all* the languages, we also apply them in a single language, $l = Zh$ as **PSA(en-zh)** and **PSA+QS(en-zh)**.

Experiments

Data and Evaluation Metric

MLQA: We first evaluate our techniques on MLQA (Lewis et al. 2020) which is a large multilingual QA dataset that covers 7 languages as listed in Table 2. The dataset is 4-ways language-parallel with parallel passages from Wikipedia articles on the same topic. Questions are originally asked in English and they are translated to other target languages.

The dataset provides a development set (1,148 parallel instances) that is significantly smaller than the blind test (11,590 parallel instances). Hence, we train our models on the SQuAD v1.1 dataset (details in Table 1). We also create a much larger multilingual training corpus with the help of machine translation. To provide a comprehensive evaluation of our techniques we run all experiments on the MLQA dataset since it was designed for both G-XLT and XLT task.

TyDi QA: We choose the best models based on our MLQA experiments and run them on the TyDi QA (Clark et al. 2020) GoldP dataset. The GoldP task was designed only for XLT evaluation and is similar to MLQA. There are 9 languages of which English (en) and Arabic (ar) are the only ones in common between TyDi QA and MLQA. Although TyDi QA has a multilingual training set, in this work we train our models on SQuAD v1.1 in order to test the cross-lingual transfer ability of our proposed models. We also create a separate training corpus by translating the questions to the TyDi QA languages, resulting in 700,792 examples. We use this augmented training corpus to implement AT and LAF. The evaluation (dev) set contains 5,077 instances.

Evaluation Metric: We use the official evaluation metric from both datasets and report the mean token F1 [4]. For MLQA, we report separate F1 scores on both the G-XLT and XLT tasks. For TyDi QA, we report the XLT F1 since the question is always in the language of the context.

Hyper-parameters

We perform hyper-parameter selection on the SQuAD and MLQA dev split. We use 3×10^{-5} as the learning rate, 384 as maximum sequence length, and a doc stride of 128. Everything except ZS was trained for 1 epoch. We use the same hyper-parameter values on the MLQA test set and TyDi QA experiments. The best question representation is achieved with the `[CLS]` token for AT and average pooling for LAF (PSA+QS). Other methods tried were the concatenation of `[CLS]` and `[SEP]`. The discriminator is implemented as a multilayer perceptron with 2 hidden layers and a hidden size of $768 * 4$. For both AT and LAF, in addition to (en-zh), which was chosen at random, we also experimented with German, the language closest to English. Both achieve similar performance.

MLQA Results

Table 2 shows the performance of various competing strategies for MLQA. For each language of the context we report the G-XLT performance averaged across questions in all the 7 languages. The final two columns show the *overall* G-XLT and the XLT performance across all the 7 languages.

Zero-shot: We report the results of our re-implementation of the ZS setting of MBERT$_{QA}$ (Lewis et al. 2020) which is the underlying QA model and show our improvements on top it.

Translation: T(Q) provides the biggest improvement out of all the competing translation techniques T(C), T(Q+C) with an overall gain (on average) of 6 points on G-XLT and 3.5 points on XLT. We believe that this degradation is due to answer alignment errors when translating the context. The alignment also causes a loss in training examples compared to the case when just the questions are translated. Note that the T(C) model is the weakest as it is the most affected by the alignment strategies and has the highest standard deviation among all the models. Combining all the strategies together provides a tiny improvement on G-XLT but at a cost to XLT performance: we believe that the T(C) data hurts this model and the parameters of MBERT alone are not sufficient to bring embeddings of different languages close to each other even with translation data. As we add more languages, the per-language capacity of the QA system decreases. This impacts the performance (known as the *curse of multilinguality* (Conneau et al. 2019)).

Adversarial Training: We first experiment with the AT (en-zh) model and noticed that adding a single language to the training data significantly improves performance over ZS. However AT (en-zh) is not strong {56.5 (G-XLT), 62.8 (XLT)} compared to T(Q), T(Q+C) and T(All). During training the discriminator is tasked to make a binary classification between En and Zh in this case. We hypothesize that this task may be too easy to balance the overall system training, since (Sønderby et al. 2017) showed that making the discriminator work harder is beneficial for training AT models. We leave training AT individually with each of the 6 other languages as part of our future work. When we extend the scope of the model to look at all languages together, we get the best performing MLQA system so far with {61.2 (G-XLT), 65.2 (XLT)}.

Language Arbitration Framework: Similar to AT, for LAF, we first start with an 'en-zh' model and then move on to an 'en-all' model. Our PSA+QS is weaker than just doing PSA on 'en-zh' suggesting again that choosing only one extra language in the LAF setting improves over the ZS baseline but is not as beneficial as adding all languages together. By choosing all the languages, we get the best performing overall model on the test split. PSA (en-all) does not lag behind but PSA+QS (en-all) provides an overall improvement of 10.2 and 4 points and 0.8 and 1.5 points improvement in G-XLT and XLT respectively over the ZS baseline and the best translation system 'T(All)'. It is more beneficial to bring

[4] We report token-level F1 as opposed to Exact Match (EM) as the latter severely penalizes a system if it adds functions words.

Model	Method	MLQA Languages (G-XLT) ar	de	en	es	hi	vi	zh	G-XLT	XLT
mBERT$_{QA}$	ZS	46.9	51.4	60.2	55.0	47.0	52.0	49.7	51.7 (±0.4)	61.7 (±0.3)
Trans	T(Q)	**53.8**	60.8	73.5	65.4	**53.2**	63.2	56.7	60.9 (±0.2)	**64.9** (±0.2)
	T(C)	44.8	51.7	62.0	57.6	42.7	55.8	50.8	52.2 (±1.0)	58.5 (±0.9)
	T(Q+C)	48.9	58.4	70.4	63.6	46.8	61.4	54.4	57.7 (±0.1)	64.3 (±0.0)
	T(ALL)	52.6	**61.1**	**73.8**	**66.3**	50.6	**64.8**	**58.2**	**61.1** (± 0.1)	64.2 (±0.2)
AT	(en-zh)	50.5	56.7	68.0	60.8	50.7	57.4	51.5	56.5 (± 0.1)	62.8 (± 0.1)
	(en-all)	**54.1**	**61.1**	**73.6**	**65.5**	**54.2**	**63.4**	**56.8**	**61.2** (± 0.1)	**65.2** (± 0.1)
LAF	PSA (en-zh)	50.8	56.9	68.6	61.3	51.0	57.8	51.6	56.9 (± 0.1)	62.8 (±0.1)
	PSA+QS (en-zh)	50.7	56.6	68.5	61.2	51.0	57.7	51.8	56.8 (± 0.4)	62.7 (±0.1)
	PSA (en-all)	54.5	61.4	74.2	66.0	54.6	64.2	57.5	61.8 (±0.1)	65.6 (±0.0)
	PSA+QS (en-all)	**54.8**	**61.5**	**74.3**	**66.1**	**54.9**	**64.3**	**57.6**	**61.9** (±0.1)	**65.7** (±0.0)

Table 2: Our results on MLQA test averaged over 3 runs. We compare our models against the previous baseline (Lewis et al. 2020): ZS setting with mBERT$_{QA}$. Best numbers within the method are in bold. The best LAF and AT models are statistically significantly better than the best Trans model.

q\c	ar	de	en	es	hi	vi	zh	**AVG**
ar	58.0	59.7	70.3	62.7	51.1	61.6	54.0	59.6
de	58.6	65.5	78.0	71.0	59.0	66.2	59.9	65.5
en	56.8	64.9	80.2	69.6	56.6	66.6	59.6	64.9
es	55.8	66.0	77.7	70.2	55.3	64.5	58.2	64.0
hi	50.5	57.1	70.3	61.5	58.9	60.7	54.0	59.0
vi	49.3	56.7	69.0	61.4	50.8	64.0	54.7	58.0
zh	54.3	60.9	74.3	66.0	52.4	66.3	63.1	62.5
AVG	54.8	61.5	74.3	66.1	54.9	64.3	57.6	61.9

Table 3: G-XLT F1 scores of the LAF:PSA+QS (en-all) model on the overall test set for individual cross languages performance. XLT F1 is 65.7 averaged across the diagonal, as shown with the G-XLT results in the last row of Table 2.

Model	ar	de	en	es	hi	vi	zh	XLT
ZS	51.7	60.6	80.4	66.8	50.5	61.4	60.1	61.7
LAF	**58.0**	**65.5**	**80.2**	**70.2**	**58.9**	**64.0**	**63.1**	**65.7**

Table 4: XLT F1 scores of ZS and LAF with mBERT.

Model	TyDi QA Languages en	bn	ko	in	te	sw	ar	ru	fi	XLT
ZS	**75.0**	62.8	55.3	61.6	49.9	57.8	61.6	65.0	58.5	60.8
T(Q)*	73.2	59.4	56.7	61.4	47.8	62.4	67.5	63.8	54.6	60.8
AT*	74.1	59.9	56.5	63.0	49.0	63.8	67.0	64.6	56.7	61.6
LAF*	74.1	59.9	55.3	64.1	49.1	63.8	**68.4**	65.9	57.3	61.9
T(Q)	73.7	63.8	59.7	70.8	49.5	60.6	65.5	65.7	69.3	64.3
AT	73.7	64.2	**62.1**	71.9	49.1	62.2	66.6	66.2	70.8	65.2
LAF	74.3	**67.6**	61.9	**72.0**	**50.6**	**62.4**	68.0	**67.0**	**71.2**	**66.1**

Table 5: Our results on TyDi QA dev. We compare our models against the previous baseline (Clark et al. 2020): ZS setting with mBERT$_{QA}$. T(Q)*, AT*, LAF* are the MLQA models. The LAF and AT models are statistically significantly better than the T(Q) model and ZS.

the multilingual embeddings closer to English for LAF than the global level as in the AT approach.

We observe that the best LAF model is consistently better than the competing strategies for *all* language pairs: 61.9 vs 61.1 (G-XLT) and 65.7 vs. 64.2 (XLT). Table 3 shows the detailed results of our best LAF model across all MLQA language combinations. In Table 4, we compare our best performance on XLT against ZS results introduced in prior work (Lewis et al. 2020) achieving a significant 4 point improvement[5].

Statistical Significance: We compute statistical significance via the Fisher randomization test. The best LAF model (PSA+QS(en-all)) is statistically significantly better than the best AT and Translation model ($p < 0.05$). The best model for all three methods (T(Q), AT (en-all) and PSA+QS (en-all)) is significantly better than the ZS baseline.

TyDi QA Results

Table 5 shows the results on TyDi QA. We first experiment with the same models that we trained for MLQA by translating SQuAD to the MLQA languages. In this setting, we evaluate cross-lingual transfer beyond translation, since *en* and *ar* are the only languages the two datasets have in common. Our best MLQA translation strategy T(Q), improves the F1 significantly on *ar* but it is slightly detrimental for the other target languages. On average the translation baseline shows no improvement over ZS. The best performing model is LAF with 1.5 F1 gains over ZS. LAF also has the best cross-lingual transfer performance, improving *in*, *sw*, *ru* as well as *ar* compared to the ZS baseline. We also

[5]Our ZS re-implementation results are higher than (Lewis et al. 2020).

tested our models trained by translating SQuAD to the TYDI QA languages. In this case, we notice consistent trends with the MLQA results. All techniques improve the cross-lingual transfer across all languages. Data augmentation with MT shows large improvement over ZS increasing the F1 by 3.4 points. AT is better compared to T(Q) and the best results are obtained with cross-lingual LAF with an average increase of 5.3 F1 points compared to ZS. Our improvements over ZS and T(Q) are statistically significant and we used the Fisher randomization test.

Error Analysis

We take a random sample of our dev data and perform error analysis on the output to provide insights into our contributions. The correct answer predicted by the better model is shown with **underline** and the incorrect answer predicted by the poorer model is shown with ***italics***.

Translation is better than ZS:
C(En): ***Stephen William Kuffler*** is known for his research on neuromuscular junctions in frogs, presynaptic inhibition, and the neurotransmitter **GABA**.
Q(Zh): 他以什么神经递质的名字而闻名
Explanation: Data augmentation helps.

AT is better than Translation:
C(De): Heftiger Regen verursachte auf **Hawai'i** geringere Schäden durch örtliche Überflutungen ..***auf der Nordhalbkugel*** die stärksten Winde und...
Q(En): Where were heavy rains?
Explanation: Adversarial training makes the MBERT embeddings more language-invariant.

LAF is better than AT:
C(Es): La película, que combina animación por computadora con acción en vivo, fue dirigida por ***Michael Bay***, con **Steven Spielberg** como productor ejecutivo.
Q(Vi): Ai là đạo diễn sản xuất bộ phim Transformers năm 2007?
Explanation: LAF makes the MBERT embeddings even more language-invariant than AT.

LAF & AT are better than Translation:
C(En): Berlin is a world city of **culture, politics, media and science**...serves as a continental hub...metropolis is a popular ***tourist destination***.
Q(De): Wofür war Berlin bekannt?
Explanation: See previous explanations.

Related Work

A large number of recent QA/ MRC datasets such as SQuAD (Rajpurkar et al. 2016; Rajpurkar, Jia, and Liang 2018), TriviaQA (Joshi et al. 2017), NewsQA (Trischler et al. 2017) and Natural Questions (Kwiatkowski et al. 2019) have focused on English and have not explored multilingual QA.

There are plenty of non-English QA datasets (Gao et al. 2016; He et al. 2018; Shao et al. 2018; Mozannar et al. 2019; Gupta et al. 2018; Lee et al. 2018; Li et al. 2018; Asai et al. 2018; Croce, Zelenanska, and Basili 2019) in Chinese, Arabic, Hindi, Korean, French, Japanese and Italian. These datasets are 2-3 way parallel or mono-lingual. XQuAD (Artetxe, Ruder, and Yogatama 2019) is a translated subset of SQuAD v1.1 into 10 languages. The most competitive multilingual datasets are MLQA and TYDI QA due to their scale and use of the original contexts as they appear in Wikipedia rather than manual translation from English.

Prior work has explored back-translation for data-augmentation (Yu et al. 2018), multi-task learning (McCann et al. 2018; Bonadiman, Uva, and Moschitti 2017; Chen et al. 2017), adversarial learning (Wallace et al. 2019; Yang et al. 2019a; Wang and Bansal 2018; Zhu et al. 2020; Keung, Lu, and Bhardwaj 2019; Chen et al. 2018) either for mono-lingual QA or for other NLP tasks. None of these have explored multilingual techniques similar to ours that make the embeddings in the LM become language-agnostic.

Contrary to our approach, (Yuan et al. 2020) present results on MLQA but assume access to a commercial search engine and web queries to create their specialized training data for their answer boundary detection task. They only report XLT results on 3/7 MLQA languages, whereas, we evaluate on *all* 7 languages and report *both* XLT and G-XLT performance. We also note that access to a search engine is not always feasible and since the authors do not provide the web queries it is unclear how to extend their technique to other languages.

Perhaps, the closest work to ours is (Cui et al. 2019a), their approach relies on back-translation and an ensemble of two QA systems one on source (context) and one on target (question) language. Our proposed methods 1. do not rely on back-translation, 2. we introduce more diverse translation models and 3. we introduce two novel strategies for multilingual QA based on language arbitration and adversarial learning. Most importantly their ensemble approach relies on training data in the target language whereas we do not.

Choosing which of the multilingual LMs (e.g. MBERT (Devlin et al. 2019), XLM-R (Conneau et al. 2019) and M4 (Arivazhagan et al. 2019)) to use for MLQA is a separate thread of work that involves comparing pre-training objectives and which large corpora to train on and is not the main focus of this paper. Due to the large number of experiments we ran we focus on one framework and we chose MBERT.

Conclusion

In this work, we highlight open challenges in the existing multilingual approach by (Lewis et al. 2020) and (Clark et al. 2020). Specifically, we show that large pre-trained multilingual LMs are not enough for this task. We produce several novel strategies for multilingual QA that go beyond zero-shot training and outshine the previous baseline built on top of MBERT. We present a translation model that has *14 times more* training data. Further, our AT and LAF strategies utilize translation as data augmentation to bring the language-specific embeddings of the LM closer to each other. These approaches help us significantly improve the cross-lingual transfer. Empirically, our models demonstrate strong results and all approaches improve over the previous ZS strategy. We hope these techniques spur further research in the field such as exploring other multilingual LMs and invoking additional networks on top of large LMs for multilingual NLP.

Acknowledgments

We thank Graeme Blackwood for his help with the machine translation api. We are grateful to Salim Roukos and the IBM MNLP team for the helpful discussions. We also thank the anonymous reviewers for their suggestions that helped us improve this paper.

References

Alberti, C.; Andor, D.; Pitler, E.; Devlin, J.; and Collins, M. 2019. Synthetic QA Corpora Generation with Roundtrip Consistency. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 6168–6173.

Arivazhagan, N.; Bapna, A.; Firat, O.; Lepikhin, D.; Johnson, M.; Krikun, M.; Chen, M. X.; Cao, Y.; Foster, G.; Cherry, C.; et al. 2019. Massively multilingual neural machine translation in the wild: Findings and challenges. *arXiv preprint arXiv:1907.05019* .

Artetxe, M.; Ruder, S.; and Yogatama, D. 2019. On the cross-lingual transferability of monolingual representations. *arXiv preprint arXiv:1910.11856* .

Asai, A.; Eriguchi, A.; Hashimoto, K.; and Tsuruoka, Y. 2018. Multilingual extractive reading comprehension by runtime machine translation. *arXiv preprint arXiv:1809.03275* .

Bonadiman, D.; Uva, A.; and Moschitti, A. 2017. Effective shared representations with Multitask Learning for Community Question Answering. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, 726–732.

Chen, D.; Fisch, A.; Weston, J.; and Bordes, A. 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)*, 1870–1879.

Chen, X.; Sun, Y.; Athiwaratkun, B.; Cardie, C.; and Weinberger, K. 2018. Adversarial deep averaging networks for cross-lingual sentiment classification. *Transactions of the Association for Computational Linguistics* 6: 557–570.

Clark, J.; Choi, E.; Collins, M.; Garrette, D.; Kwiatkowski, T.; Nikolaev, V.; and Palomaki, J. 2020. TyDi QA: A Benchmark for Information-Seeking Question Answering in Typologically Diverse Languages. *Transactions of the Association for Computational Linguistics* 8: 454–470.

Conneau, A.; Khandelwal, K.; Goyal, N.; Chaudhary, V.; Wenzek, G.; Guzmán, F.; Grave, E.; Ott, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116* .

Conneau, A.; Rinott, R.; Lample, G.; Williams, A.; Bowman, S.; Schwenk, H.; and Stoyanov, V. 2018. XNLI: Evaluating Cross-lingual Sentence Representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2475–2485.

Croce, D.; Zelenanska, A.; and Basili, R. 2019. Enabling deep learning for large scale question answering in Italian. *Intelligenza Artificiale* 13(1): 49–61.

Cui, Y.; Che, W.; Liu, T.; Qin, B.; Wang, S.; and Hu, G. 2019a. Cross-Lingual Machine Reading Comprehension. *EMNLP* .

Cui, Y.; Liu, T.; Che, W.; Xiao, L.; Chen, Z.; Ma, W.; Wang, S.; and Hu, G. 2019b. A Span-Extraction Dataset for Chinese Machine Reading Comprehension. In Inui, K.; Jiang, J.; Ng, V.; and Wan, X., eds., *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, 5882–5888.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*.

Gao, H.; Mao, J.; Zhou, J.; Huang, Z.; Wang, L.; and Xu, W. 2016. Multilingual image question answering. US Patent App. 15/137,179.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative Adversarial Nets. In Ghahramani, Z.; Welling, M.; Cortes, C.; Lawrence, N. D.; and Weinberger, K. Q., eds., *Advances in Neural Information Processing Systems 27*, 2672–2680.

Gupta, D.; Kumari, S.; Ekbal, A.; and Bhattacharyya, P. 2018. MMQA: A multi-domain multi-lingual question-answering framework for English and Hindi. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

He, W.; Liu, K.; Liu, J.; Lyu, Y.; Zhao, S.; Xiao, X.; Liu, Y.; Wang, Y.; Wu, H.; She, Q.; Liu, X.; Wu, T.; and Wang, H. 2018. DuReader: a Chinese Machine Reading Comprehension Dataset from Real-world Applications. In *Proceedings of the Workshop on Machine Reading for Question Answering*, 37–46.

Joshi, M.; Choi, E.; Weld, D.; and Zettlemoyer, L. 2017. 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)*, 1601–1611.

Keung, P.; Lu, Y.; and Bhardwaj, V. 2019. Adversarial Learning with Contextual Embeddings for Zero-resource Cross-lingual Classification and NER. 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)*, 1355–1360.

Kwiatkowski, T.; Palomaki, J.; Redfield, O.; Collins, M.; Parikh, A.; Alberti, C.; Epstein, D.; Polosukhin, I.; Kelcey, M.; Devlin, J.; Lee, K.; Toutanova, K. N.; Jones, L.; Chang, M.-W.; Dai, A.; Uszkoreit, J.; Le, Q.; and Petrov, S. 2019. Natural Questions: a Benchmark for Question Answering Research. *TACL* .

Lee, K.; Yoon, K.; Park, S.; and Hwang, S.-w. 2018. Semi-supervised training data generation for multilingual question answering. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.

Lewis, P.; Oğuz, B.; Rinott, R.; Riedel, S.; and Schwenk, H. 2020. MLQA: Evaluating cross-lingual extractive question answering. *ACL* .

Li, J.; Tu, Z.; Yang, B.; Lyu, M. R.; and Zhang, T. 2018. Multi-Head Attention with Disagreement Regularization. In *EMNLP*, 2897–2903.

McCann, B.; Keskar, N. S.; Xiong, C.; and Socher, R. 2018. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730* .

Miyato, T.; Dai, A. M.; and Goodfellow, I. J. 2017. Adversarial Training Methods for Semi-Supervised Text Classification. In *5th International Conference on Learning Representations, ICLR*.

Mozannar, H.; Maamary, E.; El Hajal, K.; and Hajj, H. 2019. Neural Arabic Question Answering. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, 108–118.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I. 2019. Language models are unsupervised multitask learners. *OpenAI Blog* 1(8): 9.

Rajpurkar, P.; Jia, R.; and Liang, P. 2018. 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)*, 784–789.

Rajpurkar, P.; Zhang, J.; Lopyrev, K.; and Liang, P. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. *EMNLP* .

Shao, C. C.; Liu, T.; Lai, Y.; Tseng, Y.; and Tsai, S. 2018. Drcd: a chinese machine reading comprehension dataset. *arXiv preprint arXiv:1806.00920* .

Sønderby, C. K.; Caballero, J.; Theis, L.; Shi, W.; and Huszár, F. 2017. Amortised MAP Inference for Image Super-resolution. In *5th International Conference on Learning Representations, ICLR*.

Trischler, A.; Wang, T.; Yuan, X.; Harris, J.; Sordoni, A.; Bachman, P.; and Suleman, K. 2017. NewsQA: A Machine Comprehension Dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, 191–200.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L. u.; and Polosukhin, I. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, 5998–6008.

Wallace, E.; Feng, S.; Kandpal, N.; Gardner, M.; and Singh, S. 2019. Universal Adversarial Triggers for Attacking and Analyzing NLP. 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)*, 2153–2162.

Wang, Y.; and Bansal, M. 2018. Robust Machine Comprehension Models via Adversarial Training. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 575–581.

Wu, S.; and Dredze, M. 2019. Beto, Bentz, Becas: The Surprising Cross-Lingual Effectiveness of BERT. 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)*, 833–844.

Yang, Z.; Cui, Y.; Che, W.; Liu, T.; Wang, S.; and Hu, G. 2019a. Improving Machine Reading Comprehension via Adversarial Training. *arXiv preprint arXiv:1911.03614* .

Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J.; Salakhutdinov, R. R.; and Le, Q. V. 2019b. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; dÁlché Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 32*, 5753–5763.

Yang, Z.; Qi, P.; Zhang, S.; Bengio, Y.; Cohen, W.; Salakhutdinov, R.; and Manning, C. D. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2369–2380.

Yarowsky, D.; Ngai, G.; and Wicentowski, R. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the first international conference on Human language technology research*, 1–8.

Yu, A. W.; Dohan, D.; Luong, M.; Zhao, R.; Chen, K.; Norouzi, M.; and Le, Q. V. 2018. QANet: Combining Local Convolution with Global Self-Attention for Reading Comprehension. In *6th International Conference on Learning Representations, ICLR*.

Yuan, F.; Shou, L.; Bai, X.; Gong, M.; Liang, Y.; Duan, N.; Fu, Y.; and Jiang, D. 2020. Enhancing Answer Boundary Detection for Multilingual Machine Reading Comprehension. *ACL* .

Zhu, C.; Cheng, Y.; Gan, Z.; Sun, S.; Goldstein, T.; and Liu, J. 2020. FreeLB: Enhanced Adversarial Training for Natural Language Understanding. In *8th International Conference on Learning Representations, ICLR*.