Distributionally Robust Multilingual Machine Translation

Chunting Zhou[*,1], **Daniel Levy**[*,2], **Xian Li**[3], **Marjan Ghazvininejad**[3], **Graham Neubig**[1]
[1]Language Technologies Institute, Carnegie Mellon University
[2]Stanford University
[3]Facebook AI
{chuntinz, gneubig}@cs.cmu.edu danilevy@stanford.edu
{ghazvini,xianl}@fb.com

Abstract

Multilingual neural machine translation (MNMT) learns to translate multiple language pairs with a single model, potentially improving both the accuracy and the memory-efficiency of deployed models. However, the heavy data imbalance between languages hinders the model from performing uniformly across language pairs. In this paper, we propose a new learning objective for MNMT based on *distributionally robust optimization*, which minimizes the worst-case expected loss over the set of language pairs. We further show how to practically optimize this objective for large translation corpora using an iterated best response scheme, which is both effective and incurs negligible additional computational cost compared to standard empirical risk minimization. We perform extensive experiments on three sets of languages from two datasets and show that our method consistently outperforms strong baseline methods in terms of average and per-language performance under both many-to-one and one-to-many translation settings.[1]

1 Introduction

Multilingual methods that process multiple languages with one single model have gained favor across a variety of NLP tasks (Firat et al., 2016; Ha et al., 2016; Johnson et al., 2017; Devlin et al., 2019; Aharoni et al., 2019; Conneau et al., 2020) because (1) training and deployment of one multilingual model is more computationally efficient compared to maintaining one model for each language (Arivazhagan et al., 2019), (2) training multilingually can improve accuracy, particularly for low-resource languages (LRLs) (Zoph et al., 2016; Neubig and Hu, 2018; Pires et al., 2019).

However, in multilingual training, the amount and type of training data available varies greatly across languages. Because most models are trained using empirical risk minimization (ERM), which minimizes the average training loss on the training set, high-resource languages (HRLs) with large amounts of data contribute to the majority of the training objective. When model capacity is limited, this results in trade-offs or decreased performance on some languages, particularly LRLs (Arivazhagan et al., 2019; Wang et al., 2020b, 2021). To better control this trade-off, a common practice is to balance the training distribution by heuristic oversampling of LRLs (Johnson et al., 2017; Neubig and Hu, 2018; Arivazhagan et al., 2019).

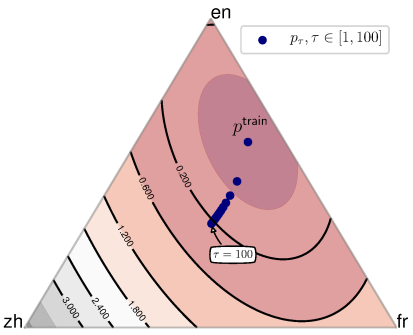

Figure 1: Illustration of different training distributions where the training distribution of the three languages `fr`, `zh` and `en` is $(0.1, 0.3, 0.6)$. Contours represent different radii of the χ^2-ball around p^{train}. The blue points are the tempered distributions described in §2.1.

Although simple data balancing can improve the performance on LRLs significantly, it is far from optimal. First, the sampling hyperparameters need to be adjusted for different datasets. Second, the use of simple heuristics does not consider the inherent level of difficulty in learning each language, the similarity between languages in the multilingual dataset, and other factors that affect cross-lingual transfer. Because of this, previous work has indicated the importance of learning strategies that are explicitly tailored for each multilingual learning scenario (Wang et al., 2020a).

In this paper, we propose a new learning proce-

[*] Equal contribution.
[1] Our code is available at https://github.com/violet-zct/fairseq-dro-mnmt

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 5664–5674
November 7–11, 2021.

dure for multilingual translation that automatically adjusts the training distribution of different languages using distributionally robust optimization (DRO) (Ben-Tal et al., 2013a; Duchi et al., 2016). In constrast to ERM, DRO casts learning as a game between the learner and an adversary, where the learner picks a model while the adversary picks the hardest data distribution for that model within an *uncertainty set* $\mathcal{Q}$ of potential distributions we wish to perform well on (which typically contains the training distribution P_0).

We first demonstrate how to apply DRO to multilingual training by letting the adversary choose the relative weights of individual language pairs in the training objective. However, we empirically find that naively applying existing methods to multilingual learning yields inferior results to ERM, mostly because (1) standard DRO objectives tend to be overly conservative and only take into account language pairs with very large losses and (2) existing optimization algorithms for DRO essentially reweigh the gradients of examples in a mini-batch, which implicitly changes the scale of the learning rates. This hurts modern NLP models like Transformers (Vaswani et al., 2017) that are highly sensitive to learning rate schedules.

To remedy this, we propose both a novel training objective and a corresponding optimization algorithm amenable to the multilingual setting. Our objective is a variation on Group DRO of Sagawa et al. with a more flexible uncertainty set, parameterized by the χ^2-divergence. To efficiently solve the min-max game, we propose an *iterated best response* scheme that, at each epoch, re-samples the training data according to the worst weighting for the current model parameters, and then runs ERM training on the re-sampled dataset. Our method—which we refer to as χ-IBR —incurs negligible additional computational cost compared to ERM.

While this method applies to essentially any multilingual task, we specifically demonstrate its benefit on three sets of language pairs from two multilingual machine translation datasets. We experimentally test these choices by comparing several objectives and optimization algorithms, and results show that our method consistently outperforms existing DRO procedures and various strong baselines.

2 Preliminaries

Notation. Throughout this paper, n denotes the training set size and d the number of parameters of the model. For $m \in \mathbb{N}$, Δ^m denotes the m-dimensional simplex, i.e. $\Delta^m := \{q \in \mathbb{R}^m, q_i \geq 0 \text{ and } \sum_i q_i = 1\}$. The data lies in $\mathcal{X} \times \mathcal{Y}$ where $(\mathbf{x}, \mathbf{y}) \in \mathcal{X} \times \mathcal{Y}$ consists of a source and target sentence pair with $\mathbf{x} = (\mathbf{x}_1, \ldots, \mathbf{x}_{L_\mathbf{x}})$ and $\mathbf{y} = (\mathbf{y}_1, \ldots, \mathbf{y}_{L_\mathbf{y}})$. The function $\ell : (\mathcal{X} \times \mathcal{Y}) \times \mathbb{R}^d \to \mathbb{R}$ refers to the loss. We consider maximum-likelihood estimation, i.e. for a target sentence $\mathbf{y}$ with $L_\mathbf{y}$ tokens, we define

$$\ell(\mathbf{x}, \mathbf{y}; \theta) = -\frac{1}{L_\mathbf{y}} \sum_{i=1}^{L_\mathbf{y}} \log p(\mathbf{y}_i | \mathbf{x}, \mathbf{y}_{<i}; \theta)$$

2.1 Multilingual Machine Translation

In contrast to bilingual machine translation, which translates from a single source language S to a target language T, multilingual neural MT (MNMT) learns a single model to translate between N language pairs $\{(S_1, T_1), \ldots (S_N, T_N)\}$. The training data D_{train} is the concatenation of the N parallel datasets, i.e. $D_{\text{train}} = [D_1; D_2, \cdots; D_N]$. We can then define the probability over each language pair $p^{\text{train}} \in \Delta^N$ as $p_i^{\text{train}} = \frac{|D_i|}{\sum_j |D_j|}$. We now describe two common training objectives for MNMT.

Empirical Risk Minimization (ERM). The simplest and most common approach for MNMT is to minimize the empirical loss over data points, which we will refer to as ERM. More precisely, we define the average loss on a parallel dataset D_i as

$$\mathcal{L}(\theta; D_i) := \frac{1}{|D_i|} \sum_{(\mathbf{x}, \mathbf{y}) \in D_i} \ell(\mathbf{x}, \mathbf{y}; \theta).$$

ERM for multilingual models then corresponds to simply minimizing the loss over D, i.e. over all the aggregated parallel sentence pairs. This yields

$$\widehat{\theta}_n^{\text{ERM}} \in \operatorname*{argmin}_\theta \mathcal{L}(\theta; D) = \sum_{i \leq N} p_i^{\text{train}} \mathcal{L}(\theta; D_i).$$

Classical results in learning theory guarantee that, under mild assumptions, as n goes to infinity, $\widehat{\theta}_n^{\text{ERM}}$ will show good performance on test sets with the same distribution as D. However, this does not guarantee that our model will perform adequately on individual parallel datasets. To remedy this issue, several works propose varying the sampling distribution—or equivalently the weighting of the parallel datasets in ERM—in order to encourage more uniform performance across language pairs.

Weighted Risk Minimization and Sampling Strategies. The amount of training data can vary significantly across language pairs. As a result,

in ERM training—i.e. optimizing for the average loss across sentence pairs—HRLs contribute most of the objective, resulting in poor performance on LRLs. Balancing the objective—or equivalently, the usage of training data—between HRLs and LRLs is important to maintain good performance across all languages (Devlin et al., 2019; Arivazhagan et al., 2019). A commonly adopted approach in multilingual training is *temperature*-based sampling (Arivazhagan et al., 2019; Conneau et al., 2020) where the probability of sampling data from D_i is proportional to its data size exponentiated by a temperature term τ, i.e. $p_{\tau,i} = \frac{|D_i|^{1/\tau}}{\sum_j |D_j|^{1/\tau}}$ (referred to as ERM with τ in §4). This is equivalent to optimizing the re-weighted objective

$$\mathcal{L}_\tau(\theta; D) = \sum_{i \leq N} p_{\tau,i} \mathcal{L}(\theta; D_i).$$

As a result, $\tau = 1$ corresponds to ERM where most of the contribution comes from the HRLs and $\tau = \infty$ corresponds to sampling language pairs uniformly at random, i.e. with data from LRLs being over-sampled. This approach comes with three major drawbacks (1) τ is an extra hyper-parameter that requires tuning for each MNMT instance to balance the performance across both HRLs and LRLs, (2) this heuristic sampling method does not consider the training dynamics of each language and how the optimal sampling distribution might evolve during the training process and (3) the parameterization of p_τ is not only very constrained (essentially one degree of freedom), it is also only a function of the quantity of training data, which is too rigid to achieve the desired performance.

To resolve some of the above issues, the recently proposed MultiDDS (Wang et al., 2020a) uses a gradient-based meta-learning approach to learn the sampling distribution over language pairs to maximize gradient similarity with a multilingual development set. However, due to the necessity to calculate and store extra gradients, their approach comes at an increased computational and memory cost. In contrast, χ-IBR enjoys the same computational complexity as ERM, and as we show in experiments it also largely outperforms MultiDDS.

2.2 Distributionally Robust Optimization

In contrast to ERM and related sampling strategies which optimize for a fixed training distribution, DRO aims to find a model θ that performs well on an entire collection of potential test distributions $\mathcal{Q}$ (the *uncertainty set*). Formally, we wish to

$$\underset{\theta}{\text{minimize}} \sup_{Q \in \mathcal{Q}} \mathbb{E}_{(\mathbf{x},\mathbf{y}) \sim Q}[\ell(\mathbf{x}, \mathbf{y}; \theta)]. \quad (1)$$

Originating from operations research (Delage and Ye, 2010; Ben-Tal et al., 2013a,b; Bertsimas et al., 2018), DRO is a promising way to tackle robustness in a variety of machine learning and NLP problems (Hashimoto et al., 2018; Oren et al., 2019; Levy et al., 2020).

We present here a recent variant, Group DRO, developed by Sagawa et al. (2020) which incorporates additional information about the data distribution to define more meaningful uncertainty sets. Abstractly, this method assumes a collection of distributions over subpopulations $\{P_g\}_{g \in \mathcal{G}}$ such that the training distribution is a mixture of these subpopulations. Importantly, it assumes that this *group structure* is observed. The Group DRO objective then minimizes the worst-case loss over these groups, which is equivalent to (1) with $\mathcal{Q} = \{\sum_{g \in \mathcal{G}} q_g P_g : q \in \Delta^{|\mathcal{G}|}\}$, or equivalently, all possible mixtures of the distributions over subpopulations. In MNMT, the N language pairs at our disposal naturally correspond to groups; thus the Group DRO objective can be defined as

$$\mathcal{L}^{\text{GDRO}}(\theta; D) = \max_{i \in [N]} \mathcal{L}(\theta; D_i). \quad (2)$$

In other words, Group DRO wishes to find a model θ that performs well for the *worst* language pair. Oren et al. (2019) propose a related but less conservative objective, CVaR-Group DRO at level $\alpha \in [0, 1]$ which, considers instead the average of the $\lceil \alpha N \rceil$ largest group losses.

3 Methods for Distributionally Robust Multilingual Learning

As we previewed in §2, Group DRO is a natural objective for the multilingual setting. However, in experiments we found that naively applying existing DRO objectives fails to achieve performance on par with strong baselines, often improving results on language pairs with high losses but sacrificing too much performance overall. Our main contribution is showing how to successfully apply DRO to the MNMT setting, and to the best of our knowledge, our work is the first to do so. To that end, our methodological contributions are two-fold: (i) we first describe the shortcomings of the Group DRO objective (2), then propose a related training criterion that addresses these issues, (ii) we describe an

optimization algorithm to solve the min-max optimization problem that is amenable to the MNMT setting.

3.1 χ^2-Group DRO

Shortcomings of Group DRO. A weakness of the objective (2) is that apart from the language pair with largest loss, the objective does not take into account the value of the loss on the other language pairs. To illustrate this, consider this example with $N = 3$ language pairs and suppose that there exists two parameters θ_1 and θ_2 with the following loss:

$$\mathcal{L}(\theta_1; D_1) = 0.1,\ \mathcal{L}(\theta_1; D_2) = 0.1,\ \mathcal{L}(\theta_1; D_3) = 1.1$$
$$\mathcal{L}(\theta_2; D_1) = 1.0,\ \mathcal{L}(\theta_2; D_2) = 1.0,\ \mathcal{L}(\theta_2; D_3) = 1.0$$

We have that $\mathcal{L}^{\mathsf{GDRO}}(\theta_1; D) = 1.1$ but $\mathcal{L}^{\mathsf{GDRO}}(\theta_2; D) = 1.0$. Consequently, the Group DRO objective will prefer θ_2 to θ_1 while clearly one would pick θ_1 over θ_2 in most practical cases. The aforementioned CVaR-Group DRO also exhibits this behavior and ignores the values of the language pairs not in the largest α-fraction.

To address this issue, let us rewrite the objective

$$\mathcal{L}^{\mathsf{GDRO}}(\theta; D) = \max_{q \in \Delta^N} \sum_{i \le N} q_i \mathcal{L}(\theta; D_i),$$

where the equality holds because the optimal weighting just puts all the mass on the language with largest loss. A natural way to make the objective less conservative is to instead take the maximum over a *subset* of the simplex $\mathcal{U} \subset \Delta^N$. This leads to the following objective

$$\mathcal{L}^{\mathcal{U}}(\theta; D) = \sup_{q \in \mathcal{U}} \sum_{i \le N} q_i \mathcal{L}(\theta; D_i). \quad (3)$$

Different choices of $\mathcal{U}$ will yield different objectives with different robustness properties. Note that this is a general formulation as $\mathcal{U} = \{p^{\mathsf{train}}\}$ reduces to the ERM objective, while $\mathcal{U}_\alpha = \{q : \|q/p^{\mathsf{train}}\|_\infty \le 1/\alpha\}$ corresponds to the CVaR-Group DRO of Oren et al. (2019). We would like to choose $\mathcal{U}$ such that optimizing this objective results in models with better performance on language pairs with large losses (typically LRLs) without significant degradation of average performance.

To this end, we turn to a common and flexible choice for $\mathcal{U}$: f-divergence balls (Csiszár, 1967) of radius $\rho > 0$ around p^{train}, namely

$$\mathcal{U}_\rho^f := \{q : D_f(q, p^{\mathsf{train}}) \le \rho\}, \quad (4)$$

where $D_f(q, p) := \sum_{i \le N} p_i f(q_i/p_i)$. In particular, we propose using the χ^2-divergence which corresponds to $f(t) = \frac{1}{2}(t-1)^2$ and define $\chi^2(q, p) = \frac{1}{2}\sum_i p_i (q_i/p_i - 1)^2$ with its corresponding uncertainty set $\mathcal{U}_\rho^{\chi^2}$. The χ^2-divergence has a long history (Ben-Tal et al., 2013b) and previous work shows that minimizing the robust loss with the χ^2-uncertainty set enjoys favorable statistical properties such as optimally trading-off bias and variance (Duchi and Namkoong, 2019) or guaranteeing robustness and fairness (Hashimoto et al., 2018; Duchi and Namkoong, 2020). We refer to the objective $\mathcal{L}^{\mathcal{U}_\rho^{\chi^2}}$ as the χ^2-Group DRO. Going back to the toy example, setting $\rho = 0.1$, yields that $\mathcal{L}^{\mathcal{U}_\rho^{\chi^2}}(\theta_1; D) = 0.64$ while $\mathcal{L}^{\mathcal{U}_\rho^{\chi^2}}(\theta_2; D) = 1.0$. With the χ^2-uncertainty set, the objective rightly prefers θ_1 to θ_2 and takes into account all the losses and not only the largest. We further confirm these intuitions and show in §4 and §5 that this is a suitable choice of uncertainty set for MNMT.

3.2 Optimization algorithm

Desiderata of the optimization algorithm. Minimizing the objective (3) effectively corresponds to a min-max optimization problem. Even in the relatively simple convex-concave setting, these are generally harder to solve than convex minimization problems. Recall that we want to

$$\underset{\theta}{\text{minimize}} \sup_{q : \chi^2(q, p^{\mathsf{train}}) \le \rho} \sum_{i \le N} q_i \mathcal{L}(\theta; D_i). \quad (5)$$

Due to the architectures we consider in MNMT, we wish for an algorithm that effectively changes the sampling distribution over mini-batches of data instead of importance-weighting the gradients. Indeed, standard MT architectures such as the Transformer (Vaswani et al., 2017) are extremely sensitive to learning rate schedules and we empirically observe that importance-weighting the gradients result in poor performance.

The canonical way to solve min-max problem is via primal-dual methods (PD) (Nemirovski et al., 2009) (see background in Appendix B), where at each step t, one keeps two vectors (θ_t, q_t) and alternates between a gradient descent step on θ_t and a gradient ascent step on q_t. One can perform these updates efficiently as they only require *unbiased stochastic gradient estimate* of the loss w.r.t. θ_t and q_t. To obtain unbiased gradient estimate of the loss w.r.t. θ_t, one either has to, at each step, sample a mini-batch of examples from

$\mathsf{Multinomial}(q_t)$ and return the gradient of the loss or sample a mini-batch from $\mathsf{Multinomial}(p^{\mathsf{train}})$ and importance-weight the gradient.

As previously mentioned, the latter is not suitable for Transformer-type architectures. The former option is not ideal as it is more convenient for an algorithm to decide the sequence of mini-batches every epoch rather than every optimizer step as this integrates much more smoothly with data loaders in deep learning frameworks, especially when doing distributed training. As a result, we posit that primal-dual algorithms are not an adequate choice for optimizing DRO-type objectives in our setting. We further discuss this point in §5.

To circumvent this issue, we consider a different optimization algorithm which we refer to as *iterated best response* (IBR), where, instead of doing a single gradient descent and ascent step, we iterate between (approximately) solving the maximization (resp. minimization) on q (resp. θ), while keeping θ_t (resp. q_t) fixed. This is similar in spirit to algorithms in the game theory literature where individuals play the optimal strategy (best response) assuming everyone else's strategies remain constant. Under some mild assumptions, this procedure converges to the equilibrium of the game (Roughgarden, 2016). Formally, we alternate between

$$\theta^{t+1} \leftarrow \operatorname*{argmin}_{\theta} \sum_i q_i^t \mathcal{L}(\theta; D_i) \tag{6}$$

$$q^{t+1} \leftarrow \operatorname*{argmax}_{q:\chi^2(q,p^{\mathsf{train}})\leq\rho} \sum_i q_i \mathcal{L}(\theta^{t+1}; D_i). \tag{7}$$

Practical implementation. As we show in Appendix A, given the values of the loss, the q-update (7) is computed to accuracy ϵ in $O(N\log(1/\epsilon))$ steps. Indeed, by taking the dual (Boyd and Vandenberghe, 2004) of (7), we transform the N-dimensional problem into a one-dimensional root-finding procedure over the dual variable which we efficiently solve with a bisection. We provide the details in Appendix A. Note that computation cost is negligible compared to computing the gradient of the loss. We implement the θ-update of (6) by running a training epoch on a *re-sampling* of the training set D according to q^{t+1}.

To compute the loss values $\{\mathcal{L}(\theta_t; D_i)\}_{i\leq N}$, necessary to perform the q-update, one needs to compute the loss $\ell(\mathbf{x}, \mathbf{y}; \theta_t)$ for every single example $(\mathbf{x}, \mathbf{y}) \in D$. This is prohibitively expensive to compute at every epoch. To avoid this, we keep track of the (approximate) historical values of the token-level loss $\hat{\mathcal{L}}_k$ on each language pair k with an exponential moving average (EMA) (see line 14 in Algorithm 1). We precisely describe our implementation in Algorithm 1. We see that it respects our desiderata and comes at no computational cost. In §5, we compare primal-dual and iterated best response for various uncertainty sets.

Algorithm 1 Iterated Best Response

1: **Input:** N parallel datasets $D_1, \ldots, D_N$, radius of the uncertainty set ρ, number of epochs T, learning rate schedule $\{\eta_{t,j}\}_{t\leq T, j\leq n}$, baseline loss $\{b_i\}_{i\leq N}$, EMA parameter $\lambda \in [0, 1]$.
2: Set $p_i^{\mathsf{train}} \leftarrow |D_i|/(\sum_j |D_j|)$.
3: Set $q^0 \leftarrow p^{\mathsf{train}}$ and $\widehat{\mathcal{L}}_i \leftarrow 0$ for $i \in [N]$.
4: Initialize θ^0.
5: **for** $t = 0, \ldots, T-1$ **do**
6: {Construction of $D(q^t)$}
7: Set $n(q^t)_i \leftarrow \lceil N \cdot q_i^t \rceil$.
8: Sample $n(q^t)_i$ data points from D_i and add them to $D(q^t)$ for $i \in [N]$.
9: $D(q^t) \leftarrow \mathsf{Shuffle}(D(q^t))$.
10: {θ- and $\widehat{\mathcal{L}}$-update}
11: **for** $(\mathbf{x}_j, \mathbf{y}_j) \in D(q^t)$ **do**
12: Let k be the language pair of $(\mathbf{x}_j, \mathbf{y}_j)$.
13: $\theta^{t,j} \leftarrow \theta^{t,j-1} - \eta_{t,j} \nabla \ell(\mathbf{x}_j, \mathbf{y}_j; \theta^{t,j-1})$.
14: $\widehat{\mathcal{L}}_k \leftarrow \lambda \cdot \ell(\mathbf{x}_j, \mathbf{y}_j; \theta^{t,j-1}) + (1-\lambda) \cdot \widehat{\mathcal{L}}_k$.
15: **end for**
16: {q-update}
17: $q^{t+1} \leftarrow \operatorname{argmax}_{q:\chi^2(q,p^{\mathsf{train}})} \sum_{i\leq N} q_i \left(\widehat{\mathcal{L}}_i - b_i\right)$
18: $\theta^{t+1,0} \leftarrow \theta^{t,|D(q^t)|}$.
19: **end for**
20: Return $\theta^{T,0}$.

Subtracting the baseline. Oren et al. propose subtracting a per-group scalar—which we refer to as a baseline—to each group loss before taking the maximum over q. They learn this baseline using a generative bi-gram model on each group. Recall that $\widehat{\theta}^{\mathsf{ERM}}$ is the parameter we obtain when we minimize the average loss and define $\widehat{\theta}^{\tau}$ when optimizing $\mathcal{L}_\tau$. In this work, we propose using $b_i = \mathcal{L}(\widehat{\theta}^{\mathsf{ERM}}; D_i)$ or $b_i = \mathcal{L}(\widehat{\theta}^{\tau}; D_i)$. Intuitively, the baseline corresponds to the minimum performance we wish for our model on the given group and as such the loss obtained with ERM and its temperature variants are natural candidates. We show in §5 that these yield significant improvement and conveniently make our method more robust to the choice of ρ. We leave different (potentially learned) choices of baseline to future work.

4 Experiments

4.1 Datasets

We evaluate the proposed method on two datasets: the 58-languages TED talk corpus (Qi et al., 2018)

	Method	aze 0.004	bel 0.006	glg 0.013	slk 0.081	tur 0.240	rus 0.274	por 0.243	ces 0.136	Avg
any→en	ERM (τ=1)	14.11	20.14	**31.94**	32.47	27.18	25.68	45.26	30.26	28.38
	MultiDDS	**14.97**	**20.60**	31.70	32.54	26.56	25.40	44.67	29.79	28.28
	χ-IBR	14.68	19.98	31.89	**33.16**	**27.76**	**26.08**	**45.33**	**30.76**	**28.71**
en→any	ERM (τ=1)	7.61	12.68	24.79	25.24	16.79	20.80	40.84	22.93	21.46
	MultiDDS	8.04	**15.04**	**26.60**	25.19	16.32	20.44	40.47	22.93	21.88
	χ-IBR	**8.49**	13.84	25.77	**26.09**	**16.78**	**21.59**	**41.38**	**23.70**	**22.21**
	Method	**bos** 0.007	**mar** 0.017	**hin** 0.033	**mkd** 0.045	**ell** 0.237	**bul** 0.308	**fra** 0.340	**kor** 0.363	**Avg**
any→en	ERM (τ=1)	24.79	12.12	23.76	34.32	39.17	40.17	**41.08**	20.19	29.45
	MultiDDS	**26.39**	**12.62**	**24.62**	**34.65**	38.46	39.71	40.60	19.46	29.56
	χ-IBR	25.12	12.52	24.42	34.47	**39.42**	**40.24**	40.98	**20.72**	**29.74**
en→any	ERM (τ=1)	16.29	5.59	16.83	26.42	33.22	35.79	39.68	9.09	22.86
	MultiDDS	**17.96**	**5.61**	**17.44**	25.98	32.72	35.28	39.57	8.96	22.94
	χ-IBR	17.33	5.59	16.90	**28.02**	**33.82**	**36.37**	**40.35**	**9.13**	**23.44**

Table 1: BLEU scores of the best ERM model (among τ=1/5/100, $\tau = 5/100$ are significantly worse than $\tau = 1$, thus we omit these results), MultiDDS (Wang et al., 2020a) and our approach on the test sets of the TED dataset. Bold (resp. underlined) values indicate the best (resp. second best) performance for each language pair. Values under the language codes are the proportion of the language in the training data.

Method	any→en					en→any				
	deu	fra	tam	tur	Avg	deu	fra	tam	tur	Avg
ERM (τ=1)	**29.98**	30.32	15.81	19.85	23.99	**23.82**	33.09	9.28	13.29	19.87
ERM (τ=5)	29.25	31.60	16.31	21.89	24.76	22.67	32.36	10.04	16.09	20.29
ERM (τ=100)	28.75	30.71	15.80	**22.44**	24.43	22.02	31.65	10.41	**16.44**	20.13
MultiDDS	29.31	31.41	16.12	21.43	24.57	22.99	31.55	10.09	14.51	19.79
χ-IBR	29.67	**31.75**	**16.48**	22.33	**25.06**	23.45	**33.16**	**10.73**	15.53	**20.72**

Table 2: BLEU scores of the ERM (τ=1/5/100), MultiDDS and our method on the test sets of the WMT dataset. The ratios of training data of de, fr, ta and tr are (0.499, 0.359, 0.102, 0.039).

and WMT datasets. For the TED corpus, we evaluate on two sets of languages with varying levels of language diversity following Wang et al. (2020a): (1) *related* includes 4 LRLs (aze, bel, glg, slk) and their corresponding related HRL (tur, rus, pos, ces). (2) *diverse* includes 8 languages with varying amount of data without considering linguistic similarities (bos, mar, hin, mkd, ell, bul, fra, kor)[2]. Both of the related and diverse sets have around 760K sentences of training data.

For WMT, we consider 2 HRLs (German:deu and French:fra) and 2 LRLs (Tamil:tam and Turkish:tur). We subsample around 5M training sentences from the parallel corpus provided by the WMT shared task. Specifically, the training data of deu-eng, fra-eng is from WMT14, tam-eng is from WMT20 and tur-eng is from WMT18. We use the corresponding test and dev sets from each shared task for evaluation and validation.

We evaluate both *en-to-any* (translate English to a target language) and *any-to-en* (translate a source language into English) directions for all language sets. We provide dataset statistics in Appendix C.

4.2 Experimental setup

For the translation models, we adopt the encoder-decoder Transformer (Vaswani et al., 2017) architecture with the implementation provided in fairseq (Ott et al., 2019). For both datasets, we use a Transformer-base architectures that also has 6 encoder and decoder layers with hidden dimension size being 512 and 8 attention heads.[3] The model is trained for 200K and 300K steps for TED and WMT respectively with the batch size of 65,536 tokens. For both datasets, we learn the sentencepiece (Kudo and Richardson, 2018) vocabulary for the English and the combined corpus of other languages respectively. We use beam search with beam size 5 for decoding and report the SacreBLEU score (Post, 2018; Papineni et al., 2002) on test sets for evaluation. For the TED and WMT datasets respectively, the constraint size ρ for the

[2]See Wang et al. (2020a) for the interpretation of the language codes.

[3]We train for more steps with a larger batch size, which yields much better results than reported in Wang et al. (2020a).

Method	any→en					en→any				
	deu	**fra**	**tam**	**tur**	**Avg**	**deu**	**fra**	**tam**	**tur**	**Avg**
FastDRO	25.14	27.58	12.71	15.54	20.24	21.39	28.21	8.88	12.74	17.81
GDRO with PD	26.72	29.13	15.78	21.89	23.38	20.81	29.43	10.29	15.52	19.01
CVaR-GDRO with PD	28.62	30.70	15.94	20.61	23.97	22.81	32.44	9.68	14.33	19.82
CVaR-GDRO with IBR	29.14	31.65	16.31	20.98	24.52	22.34	31.97	10.15	14.82	19.82
χ^2-GDRO with PD	29.49	31.47	16.07	21.24	24.57	23.10	32.30	9.87	14.70	19.99
ERM (τ=5)	29.25	31.60	16.31	21.89	24.76	22.67	32.36	10.04	**16.09**	20.29
χ-IBR	**29.67**	**31.75**	**16.48**	**22.33**	**25.06**	**23.45**	**33.16**	**10.73**	15.53	**20.72**

Table 3: BLEU scores of different DRO objectives and algorithms—primal-dual (PD) and iterated best response (IBR)—on the WMT test sets.

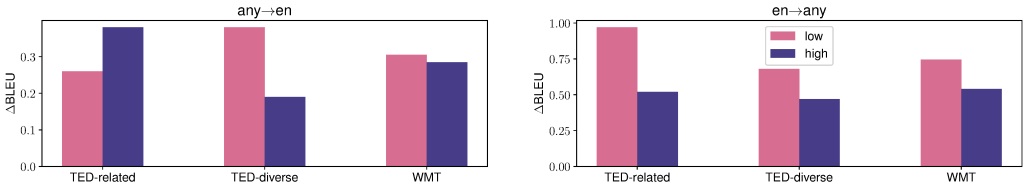

Figure 2: ΔBLEU of low- and high-resource language groups for the three language sets. ΔBLEU = difference of BLEU scores of χ-IBR and the best ERM model.

chi-square ball is set to be 0.05 and 0.3, and for the baseline losses we use the average token-level loss on each D_i computed from the ERM model with $\tau = 1$ and $\tau = 100$—see §5 for more analyses of these choices. We provide additional preprocessing and training details in Appendix D.

Baselines. We compare with (1) **temperature-based sampling** method described in §2.1 in three standard settings ($\tau = 1/5/100$), where $\tau = 100$ approximates uniform sampling over language pairs, and (2) **MultiDDS** described in §2.1. In addition, we also perform extensive empirical studies over different DRO uncertainty sets and optimization procedures in §5.

4.3 Main Results

We present the BLEU scores of en→any and any→en translation directions on TED and WMT data in Tab. 1 and 2 respectively. First, for both TED and WMT datasets, χ-IBR outperforms all the other baseline methods in terms of average BLEU score over all language pairs. By taking a closer look at the BLEU scores for each individual language pairs, χ-IBR improves over almost all the language pairs for both translation directions compared to ERM. Secondly, as expected from temperature-based sampling methods, different values of τ achieve different trade-offs between the performances on HRLs and LRLs. As we explained in §2.1, large values of τ favor LRLs as this results in data being sampled with equal probability from each language pair while small values of τ approach ERM and will benefit HRLs. As a result, τ needs to be carefully tuned to achieve adequate performance on both HRLs and LRLs.

Importantly, χ-IBR achieves a significantly better trade-off than the sampling method for any value of τ. We show in Fig. 2 the quantitative improvements of χ-IBR over the best τ for various datasets and in both translation directions. Surprisingly, while the improvements are larger on LRLs in most cases, we consistently observe improvements on HRLs. This indicates that finding the right sampling distribution over language pairs facilitates cross-lingual transfer. We further observe that χ-IBR achieves more significant improvements in the en→any direction than in the any→en direction. This further supports our hypothesis. Indeed, it is attested in previous work on MNMT (Arivazhagan et al., 2019) that it is harder to decode to multiple languages than encode from multiple languages and as such, the en→any direction is a significantly harder multi-task learning problem. As such, this is where optimal cross-lingual transfer would yield the larger gains, which is what we observe in practice. We also note that our method incurs negligible computational overhead compared to ERM.

5 Analysis

The importance of the sampling distribution. An advantage of our method over temperature-based sampling methods is that it dynamically adjusts the training distribution as the model evolves

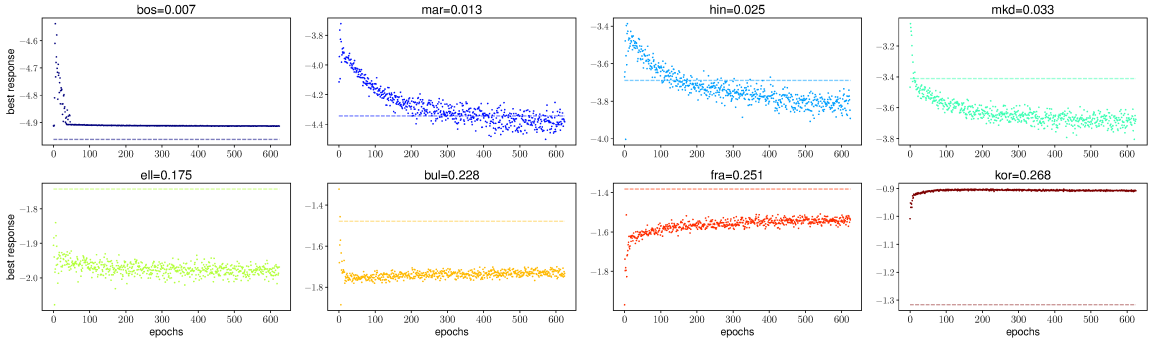

Figure 3: Best response q (in log-scale) across epochs on the TED diverse dataset for the any→en direction.

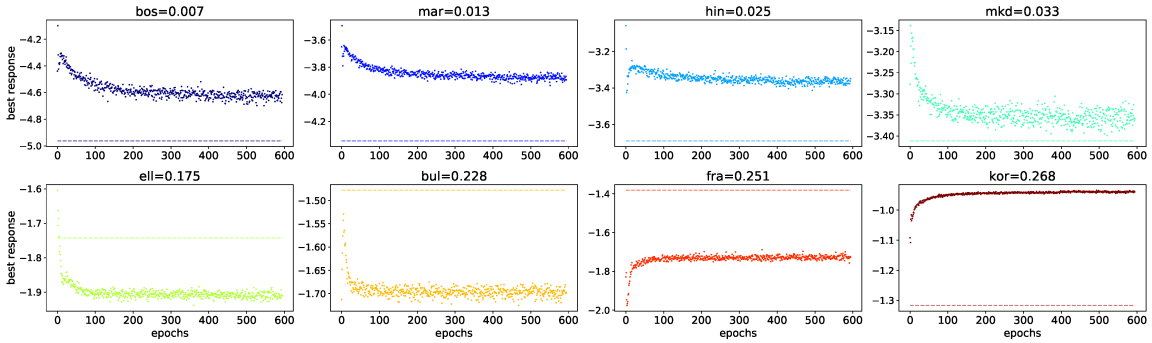

Figure 4: Best response q (in log-scale) across epochs on the TED diverse dataset for the en→any direction, the dashed line is the true data probability (in log-scale).

and does not compute it solely as a function of amount of training data. Our hypothesis is that this is important to achieve good performance across language pairs and that different sampling distributions will be adequate at different stages of training. We empirically check our hypothesis by studying how the training distribution q (the so-called best response) changes across training epochs. In Fig. 3 and 4, we plot the best response of χ-IBR across epochs on the TED-diverse dataset for both translation directions. In addition, we also plot the historical losses in Fig. 5 (Appendix). Our first observation is that the optimal q noticeably evolves across epochs which further showcases the need for dynamically adjusting the sampling distribution. We make the following observations (i) χ-IBR demonstrates the desired behavior and, at the early stages of training, always down-weights HRLs and up-weights LRLs; (ii) somewhat counter-intuitively, there is no direct correlation between $|D_i|$, the amount of data in language i and the final value $\mathcal{L}(\theta^{(t)}; D_i)$. The latter further evidences the limitations of sampling distributions only based on the amounts of training data $|D_i|$. Indeed, while `kor` is a HRL, it is typologically much farther from English so there is more inherent uncertainty in the task. Consequently, it has larger losses and is consistently up-weighted throughout training. On the other hand, while `hin` is a LRL, it achieves low loss after being up-weighted during the early stages of training and is consequently down-weighted after that.

Comparison to DRO variants. We demonstrate the benefits of χ-IBR over other DRO objectives by extensively evaluating a range of robust objectives and associated optimization algorithms. In terms of objective, we compare against (1) **Group DRO** (Sagawa et al., 2020), (2) **CVaR-Group DRO** (Oren et al., 2019) and (3) **FastDRO** (Levy et al., 2020). In terms of algorithms, we experiment with primal-dual methods and our proposed iterated best response procedure which we both described in §3.2. Note that in the case of Group DRO (i.e. $\mathcal{U} = \Delta^N$), iterated best response is not a sensible choice as it would result in each training epoch being spent on a single language pair. In the case of CVaR Group DRO, we follow the implementation of (Oren et al., 2019), which is a hybrid of the two optimization algorithms with a primal update on θ and a best response update on q. We compare the performance of these methods on the WMT dataset. For fairness, we baseline losses in the same way for all the DRO objectives. We first observe that, outside of χ-IBR, none of the DRO objectives are competitive with temperature-weighted ERM. We also observe that for both uncertainty sets, iterated

Dataset	Setting	Avg BLEU
TED	(a) ERM, $\tau = 1$	21.46
	(b) ERM, $\tau = 100$	20.41
	(c) Ours, $\rho = 0.05$, w/o BL	22.08
	(d) Ours, $\rho = 0.1$, w/o BL	21.75
	(e) Ours, $\rho = 0.05$, BL: $\tau = 1$	22.21
	(f) Ours, $\rho = 0.1$, BL: $\tau = 1$	22.13
	(g) Ours, $\rho = 0.05$, BL: $\tau = 100$	21.37
WMT	(h) Ours, $\rho = 0.1$, BL: $\tau = 1$	20.34
	(i) Ours, $\rho = 0.1$, BL: $\tau = 100$	20.62

Table 4: Average BLEU on the test sets of en→any direction, BL is short for baseline loss.

best response convincingly outperforms the same objective trained with primal-dual. We finally note that, for a fixed optimization algorithm, χ^2-Group DRO outperforms the CVaR objective on all but one language pairs. This validates both our choice of uncertainty set and of optimization procedure.

The effects of baselined losses. We study the effect of the choice of baseline on the performance across languages. In Tab. 4, we empirically evaluate different baseline choices and uncertainty sizes ρ. We observe that in the TED dataset, baselineing with $\mathcal{L}(\widehat{\theta}^{\mathsf{ERM}}; D_i)$ performs significantly better than baseline-ing with $\mathcal{L}(\widehat{\theta}^{\tau=100}; D_i)$ ((e) versus (g) while it is reversed for WMT. We explain this by observing that the LRLs in TED consist of very small amounts of data (on the order of a few thousands) and using $\tau = 100$ results in a severe oversampling of LRLs, which the model then fits perfectly. As a result, recall the intuition that the baseline sets a lower bound on the performance we wish to achieve but because of the small training data and overfitting, the model disproportionately up-weighs the LRLs, which harms overall performance. This does not occur in WMT and uniform sampling across language pairs sets a good target performance for DRO methods. Finally, we see that with the right baseline loss, our method is more robust to different choices of ρ (e.g., comparing (c) and (d) versus (e) and (f)). We consistently observe this for other translation directions and datasets.

6 Conclusion

We showed how to successfully apply DRO to the MNMT setting and automatically adjust the sampling distribution over language pairs resulting in sizeable improvements in performance. We posit that this approach would also be successful in other multilingual scenarios. Our work raises a few questions: (i) what are the right baseline losses? (ii) surprisingly, χ-IBR also improves performance on HRLs; under what circumstances does cross-lingual transfer happen and which languages does it benefit most? We hope our work could inspire better distributionally robust learning methods for multilingual training in the future.

Acknowledgements

This work was supported in part by a Facebook SRA Award and the NSF/Amazon Fairness in AI program under grant number 2040926.

References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. 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 3874–3884.

Naveen Arivazhagan, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, Yuan Cao, George Foster, Colin Cherry, et al. 2019. Massively multilingual neural machine translation in the wild: Findings and challenges. *arXiv preprint arXiv:1907.05019*.

Aharon Ben-Tal, Dick Den Hertog, Anja De Waegenaere, Bertrand Melenberg, and Gijs Rennen. 2013a. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357.

Aharon Ben-Tal, Dick den Hertog, Anja De Waegenaere, Bertrand Melenberg, and Gijs Rennen. 2013b. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357.

Dimitris Bertsimas, Vishal Gupta, and Nathan Kallus. 2018. Data-driven robust optimization. *Mathematical Programming, Series A*, 167(2):235–292.

Stephen Boyd and Lieven Vandenberghe. 2004. *Convex Optimization*. Cambridge University Press.

Sébastien Bubeck. 2015. Convex optimization: Algorithms and complexity. *Foundations and Trends in Machine Learning*, 8(3-4):231–357.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Édouard 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.

Imre Csiszár. 1967. Information-type measures of difference of probability distributions and indirect observation. *Studia Scientifica Mathematica Hungary*, 2:299–318.

Erick Delage and Yinyu Ye. 2010. Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research*, 58(3):595–612.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186.

John Duchi, Peter Glynn, and Hongseok Namkoong. 2016. Statistics of robust optimization: A generalized empirical likelihood approach. *arXiv preprint arXiv:1610.03425*.

John C. Duchi and Hongseok Namkoong. 2019. Variance-based regularization with convex objectives. *Journal of Machine Learning Research*, 20(68):1–55.

John C. Duchi and Hongseok Namkoong. 2020. Learning models with uniform performance via distributionally robust optimization. *Annals of Statistics*, to appear.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 866–875.

Thanh-Le Ha, Jan Niehues, and Alexander Waibel. 2016. Toward multilingual neural machine translation with universal encoder and decoder. *arXiv preprint arXiv:1611.04798*.

Tatsunori Hashimoto, Megha Srivastava, Hongseok Namkoong, and Percy Liang. 2018. Fairness without demographics in repeated loss minimization. In *Proceedings of the 35th International Conference on Machine Learning*.

Melvin Johnson, Mike Schuster, Quoc Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, et al. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71.

Daniel Levy, Yair Carmon, John C. Duchi, and Aaron Sidford. 2020. Large-scale methods for distributionally robust optimization. In *Advances in Neural Information Processing Systems 33*.

Hongseok Namkoong and John C. Duchi. 2016. Stochastic gradient methods for distributionally robust optimization with f-divergences. In *Advances in Neural Information Processing Systems 29*.

A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. 2009. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on Optimization*, 19(4):1574–1609.

Arkadi Nemirovski. 2004. Prox-method with rate of convergence $O(1/t)$ for variational inequalities with Lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15(1):229–251.

Graham Neubig and Junjie Hu. 2018. Rapid adaptation of neural machine translation to new languages. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 875–880.

Yonatan Oren, Shiori Sagawa, Tatsunori Hashimoto, and Percy Liang. 2019. Distributionally robust language modeling. 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 4218–4228.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL 2002*.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual bert? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

Ye Qi, Devendra Sachan, Matthieu Felix, Sarguna Padmanabhan, and Graham Neubig. 2018. When and why are pre-trained word embeddings useful for neural machine translation? In *Meeting of the North American Chapter of the Association for Computational Linguistics (NAACL)*, New Orleans, USA.

Tim Roughgarden. 2016. *Twenty Lectures on Algorithmic Game Theory*. Cambridge University Press.

Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. 2020. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations (ICLR)*, Addis Ababa, Ethiopia.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Xinyi Wang, Yulia Tsvetkov, and Graham Neubig. 2020a. Balancing training for multilingual neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8526–8537.

Zirui Wang, Zachary C Lipton, and Yulia Tsvetkov. 2020b. On negative interference in multilingual language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4438–4450.

Zirui Wang, Yulia Tsvetkov, Orhan Firat, and Yuan Cao. 2021. Gradient vaccine: Investigating and improving multi-task optimization in massively multilingual models. In *International Conference on Learning Representations*.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575.