

---

# Learning Curricula for Multilingual Neural Machine Translation Training

**Gaurav Kumar**  
**Philipp Koehn**  
**Sanjeev Khudanpur**  
CLSP, Johns Hopkins University, Baltimore, 21218, USA

gkumar@cs.jhu.edu  
phi@jhu.edu  
khudanpur@jhu.edu

---

## Abstract

Low-resource Multilingual Neural Machine Translation (MNMT) is typically tasked with improving the translation performance on one or more language pairs with the aid of high-resource language pairs. In this paper, we propose two simple search based curricula – orderings of the multilingual training data – which help improve translation performance in conjunction with existing techniques such as fine-tuning. Additionally, we attempt to learn a curriculum for MNMT from scratch jointly with the training of the translation system using contextual multi-arm bandits. We show on the FLORES low-resource translation dataset that these learned curricula can provide better starting points for fine tuning and improve overall performance of the translation system.

## 1 Introduction

Curriculum learning (Bengio et al., 2009; Elman, 1993; Rohde and Plaut, 1994) hypothesizes that presenting training samples in a meaningful order to machine learners during training may help improve model quality and convergence speed. In the field of Neural Machine Translation (NMT) most curricula are hand designed e.g., fine-tuning (Luong and Manning, 2015; Freitag and Al-Onaizan, 2016) and data selection (Moore and Lewis, 2010; Axelrod et al., 2011; Duh et al., 2013; Durrani et al., 2016). Another common curriculum is one based on ordering samples from *easy to hard* using linguistic features and auxiliary model scores (Zhang et al., 2018, 2019) but these are hard to tune, relying to extensive trial and error to find the right hyperparameters. Attempts to learn a curriculum jointly with the NMT training setup (Kumar et al., 2019) can suffer from observation sparsity, where a single training run does not provide enough training samples for an external agent to learn a good curriculum policy.

Our NMT task of choice in this paper is *low-resource multi-lingual NMT* (MNMT). While standard NMT systems typically deal with a language pair, the source and the target, an MNMT model may have multiple languages as source and/or target. Most large-scale MNMT models are trained using some form of model parameter sharing (Johnson et al., 2017; Aharoni et al., 2019; Arivazhagan et al., 2019; Bapna and Firat, 2019). The notion of how input data should be presented to the MNMT system during training only finds prominence in the case of low-resource MNMT. A typical low-resource task will try to leverage a high-resource language pair to aid the training of an NMT system for a low-resource (very small or no parallel data available) and related language-pair of interest. Typical approaches for low resource MNMT involve pivoting and zero-shot training (Lakew et al., 2018; Johnson et al., 2017) and transfer learning via fine-tuning (Zoph et al., 2016; Dabre et al., 2019). Finn et al. (2017) attempt to meta-learn parameter initialization for child models using trained-high resource parent models for this task.

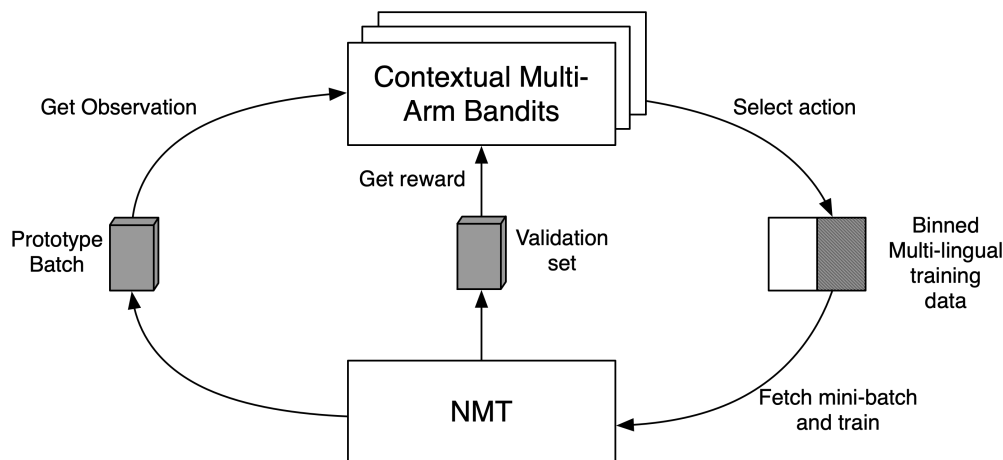


Figure 1: The multi-arm bandit agents’ (MAB) interface with the NMT system.

In this paper, we build upon the framework for learning curricula introduced in Kumar et al. (2019) and attempt to alleviate the problem of observation sparsity by learning more robust policies from multiple training runs. We use contextual multi-arm bandits for our agents which learn multilingual data sampling policies jointly with the training of the NMT system. Additionally, we explore some simple policy search methods to our list of baselines; specifically, we try and find the best policies using the expensive grid search and pruned-tree search methods. We use state-of-the-art hand-designed curricula as our baselines to beat. Building upon the task and datasets established by Guzmán et al. (2019), in this paper, we will attempt to learn a curriculum to train an NMT system for the Nepali-English language pair while leveraging the high resource Hindi-English pair. The agent will learn to choose between mini-batches containing either Hindi-English or Nepali-English data at each time step during NMT training to maximize the expected reward (improvement in validation set performance). The learned curriculum will hence condition on the state of the NMT system during training and determine whether to expose it to a batch of Nepali-English or Hindi-English data. We start by presenting our methods for obtaining search-based and learned curricula in section 2. We present our experiment setup in section 3 and results in section 4.

## 2 Methods

The procedure for learning a multi-lingual training curriculum uses multiple multi-arm bandits as agents which explore independent of each other in randomly initialized environments (NMT systems) and effectively learn their own policies. The stochastic nature of their exploration policy ensures that they explore different action-reward spaces (the agent executes an action on the NMT environment and receives a reward associated with this action). Figure 1 shows an overview of this interface. The training data for all agents is pooled at the end of the training of individual agents and one final agent is trained using this data which determines the final policy we use as our multi-lingual curriculum. We provide more details about this method of learning and the associated baselines below.

### 2.1 Data Binning

Instead of mixing together all the language pairs into one single dataset, we create separate *bins* for each language pair. Hence, with respect to the agent, this is a two bin problem, where its

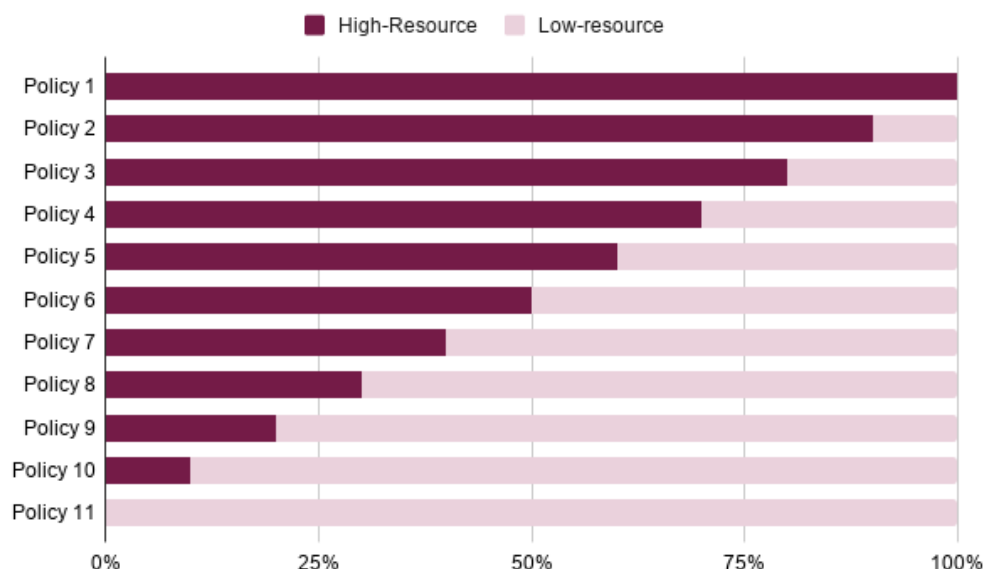


Figure 2: A line search for a fixed curriculum baseline which samples from one language pair (high resource, Hindi-English) with a fixed probability or else samples from the other (low resource, Nepali-English).

*action* is the choice of the bin to draw the next mini-batch for NMT training. As a result of this design decision, each batch will only contain a single language pair and will hence be relatively homogeneous (with respect to the feature of interest, language id). More generally, this can be extended to an arbitrary number of bins, one per language-pair being used to train the MNMT system.

## 2.2 Grid-search baselines

The simplest (albeit expensive to find) search-based learn-able curriculum to consider in this case is one where we sample batches from one language with a fixed probability or else sample from the other bin during training. Since there is only one degree of freedom (the probability of sampling from one language-pair) in this search problem, we perform a simple line-search over the range of possible values for this probability. Note that, although this curriculum is ‘learned’ it remains fixed during each training run and does not change based on the state of the NMT system. Figure 2 shows a visual representation of this search method.

## 2.3 Pruned Tree search

A variation of the previous search method involves one which uses a technique similar to beam search. We divide training into a finite number of *phases* and then starting from the beginning of training, we search for the best fixed sampling probability. At the end of this *phase*, we discard all but the best model and the policy (sampling probability) which led to it, and continue the search for the best policy in the next phase from this model checkpoint. The result is a tree-search which prunes all but the best node after each phase. The final policy is the culmination of all phase-wise best fixed sampling ratios. This procedure appears in Algorithm 1.

---

**Algorithm 1:** Pruned tree-search for multi-lingual curricula search

---

**Result:**  $P^*$ , the list of the best policies per phase  
 $\hat{p} = \{0.0, 0.1, \dots, 1.0\}$  // Policies to explore;  
Randomly initialize starting NMT model  $\Theta^*$ ;  
**while** NMT next training phase  $t$  exists **do**  
  **for**  $p$  in  $\hat{p}$  **do**  
    Bin sampling probability =  $p$ ;  
    Training start checkpoint =  $\Theta^*$ ;  
    Run training of NMT training for phase  $t$ ;  
    Store trained model checkpoint  $\theta$   
  **end**  
  Select model  $\theta^*$  with best score on validation set with policy  $p^*$ ;  
   $P^* = P^* + [(t, p^*)]$ ;  
   $\Theta^* = \theta^*$ ;  
**end**

---

## 2.4 Observation Engineering

The observations provided to the multi-arm bandit agents are identical in structure to the ones introduced in Kumar et al. (2019). A *prototype* batch – a finite number of sentences from each language pair – is randomly sampled per bin (language-pair) and concatenated together. At each time step, the observation is the vector containing sentence-level log-likelihoods produced by the NMT system for this prototype batch. We exclude observations from the initial portion of NMT interaction to counteract the naturally decaying property of log-likelihood scores during NMT training.

## 2.5 Contextual Multi-arm Bandits

Multi-arm bandit (MAB) based agents are typically trained to learn policies which maximize the expected reward received (minimize regret). Contextual multi-arm bandits (Pandey et al., 2007; Chih-Chun Wang et al., 2005; Langford and Zhang, 2008) allows the use of state based information to determine this policy. In our case the contextual MABs condition on the *observation* received from the NMT system to determine an *action*, the choice of bin to sample a mini-batch. The *reward* obtained for this action is the *delta-validation perplexity* post update, the improvement in perplexity on the validation set in a finite window. The exploration strategy is the linearly-decaying *epsilon-greedy* strategy (Kuleshov and Precup, 2014). The contextual MABs are implemented as simple feed-forward neural networks which take the *observation* vector as input and produce a distribution over two states representing the bins. If we choose to *exploit* this learned policy, the bin with maximum probability mass is selected for sampling.

## 3 Experiment Setup

We use Fairseq (Ott et al., 2019) for all our NMT experiments and the our NMT systems are configured to replicate the setup described in Guzmán et al. (2019). The grid search experiments search over the the range  $[0, 1]$  for sampling in increments of 0.1. The pruned tree-search uses a beam width of 1. The phase duration for tree-search is set to one epoch of NMT training. We use either 5 or 10 concurrent contextual MABs which are implemented as two 256-dimensional feed forward neural networks trained using RMSProp with a learning rate of 0.00025 and a decay of 0.95. Rewards for the agent (validation delta-perplexity) are provided every ten training steps. To create the observations, we sample 32 prototype sentences from each bin to create a *prototype*

Dataset	Sentences	Tokens
Nepali-English	563K	6.8M
Hindi-English	1.6M	16.7M

Table 1: Statistics of the training data for the Nepali-Hindi-English multilingual NMT system.

	valid	test
Baselines		
ne-en: Random Baseline	6.35	7.71
hi-en: Random baseline (with ne valid)	2.71	3.9
ne-hi-en: Random Baseline	<b>12.24</b>	<b>14.88</b>
ne-hi-en: Multi-lingual Transformer	12.01	14.78
ne-hi-en: Continued training from hi-en	12.2	14.3
Searched Curricula		
Grid Search (best = 50/50)	12.01	14.78
Grid Search (best = 50/50) + Continued training	<b>12.33</b>	<b>15.1</b>
Pruned Tree-search	12.3	14.8
Pruned Tree-search + Continued training	12.41	14.92
Agent Learned Curricula		
MAB2 (best = 10 concurrent, 500 updates)	12.21	14.87
MAB4 (best = 5 concurrent, 2 epochs)	12.18	14.67
MAB2 + Continued Training	<b>12.4</b>	<b>15.45</b>
MAB4 + Continued Training	12.27	15.2

Table 2: BLEU scores for the Nepali-English test set using the fixed, searched and learned multilingual curricula. The values in bold are the best results per section. Continued training from the models learned using multi-arm bandits provides the best results overall.

*batch* of 64 sentences and measure sentence level log-likelihood after each update. We use an NMT warmup of 5000 steps (no training data for the agent from this period is recorded). For the exploration strategy we use a linearly decaying epsilon function with decay period set to 25k steps. The decay floor was set to 0.01. The window for the delta-perplexity reward was 1.

We use the datasets provided as part of the FLORES task (Guzmán et al., 2019) for our experiments. The statistics of the training dataset for the multi-lingual task appear in table 1. The Hindi-English dataset comes from the IIT Bombay corpus<sup>1</sup>. The validation and test sets for Nepali-English (the low resource language-pair of interest) contain 2500 and 3000 sentences respectively.

## 4 Results

Our results are presented in Table 2. Our baselines consist of:

- ne-en random baseline: This is the NMT setup which is only trained on the Nepali-English corpus. The data is randomly shuffled to form mini-batches.
- hi-en random baseline: The NMT system trained on the high-resource Hindi-English dataset with the Nepali-English validation and test sets.

<sup>1</sup>[http://www.cfilt.iitb.ac.in/iitb\\_parallel](http://www.cfilt.iitb.ac.in/iitb_parallel)

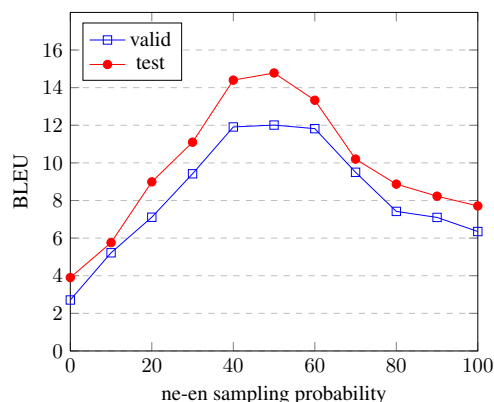


Figure 3: BLEU scores for the Nepali-English validation and test set at various values of the ne-en sampling probability.

- ne-hi-en random baseline: The Hindi-English and Nepali-English data is mixed together to train the NMT system. The Nepali-English data is upsampled to match the size of the the Hindi-English corpus.
- Multilingual transformer: Replicates the setup from Guzmán et al. (2019).
- Continued training baseline: Uses the hi-en random baseline as a starting point to fine tune using the Nepali-English validation and test sets.

Our non-MAB search-based curriculum baselines are:

- Grid search: A static curriculum is learned by searching over the space of sampling probabilities for the bins.
- Grid Search + Continued training: The previous model is fine tuned using the Nepali-English validation and test sets.
- Pruned tree-search: Epoch-dependent curriculum searched using the pruned tree-search method.
- Pruned tree-search + Continued training: The previous model is fine tuned using the Nepali-English validation and test sets.

From Table 2, we see that the ne-en and hi-en baselines are very weak, with the latter lagging behind despite having access to more data. This indicates that with these language pairs, even though adding the high-resource dataset may help, in isolation it is not a good proxy for the low-resource pair. The random baseline with the combination of the two datasets (upsampled low-resource) is the strongest amongst the fixed baselines marginally beating the multi-lingual transformer and (surprisingly) the continued training baselines. While the grid search and pruned-tree search baselines are close in performance to the best fixed baselines, continued training with them provides much stronger results where the 50/50 configuration for the grid search<sup>2</sup> provides the best result at 15.1 BLEU and the tree search slightly behind at 14.92 BLEU. Figure 3 shows the BLEU scores for the grid search experiments over the chosen search points in the probability space (the probability of sampling from the low resource language pair).

<sup>2</sup>Note that the grid search method has access to the binned data and can only ever select data from one language

	valid	test
MAB1 (5 conc., 500 updates)	12.2	14.11
MAB2 (10 conc., 500 updates)	<b>12.21</b>	<b>14.87</b>
MAB3 (5 conc., 1 epoch)	11.44	13.98
MAB4 (5 conc., 2 epoch)	12.18	14.67
With continued training		
MAB2 + Continued Training	<b>12.4</b>	<b>15.45</b>
MAB4 + Continued Training	12.27	15.2

Table 3: BLEU scores for the Nepali-English test set using various configurations (number of concurrent agents, policy update interval) of the contextual MABs to learn the multilingual sampling curriculum.

For the contextual MABs, we use either 5 or 10 concurrent agents; training data is gathered from all concurrent bandits to train the final curriculum. In addition, we choose to update the bandit policy only once every 500 updates, 1 epoch or 2 epochs of NMT training. The results of all our experiments appear in table 3 and the best configurations are in table 2. While the curricula learned using the contextual MABs are able to match the performance of the strongest fixed policy (ne-hi-en random baseline), it performs slightly worse than the curriculum obtained using the (expensive) grid search combined with continued training, by about 0.2 BLEU points. Interestingly, continuing training from the models trained using the curricula learned by the MABs leads to the strongest results. Specifically, using the model trained using the curriculum learned by the strongest MAB (MAB2 in Table 2) results in a BLEU score of 15.45 on this task, a gain of 0.6 on the strongest baseline.

## 5 Conclusion

In this paper, we present techniques which learn curricula for multilingual NMT training from multiple training runs and agents. On the task of low-resource multilingual NMT training, we use conditional multi-arm bandits which condition on the state of the NMT system and learn policies which determine whether to train on a batch of a high-resource (Hindi-English) or the low-resource (Nepali-Hindi) language pair per step in training. In addition, we introduce some simple search-based methods for policy search (grid search and pruned tree search) for this task. We show that both these simple *learned* curricula and the ones derived from the MABs can match the state-of-the-art hand-designed multilingual baselines. However, continued training on models trained using these *learned* curricula yields better results, indicating that they may serve as good starting models for fine-tuning.

## References

- Aharoni, R., Johnson, M., and Firat, O. (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, Minneapolis, Minnesota. Association for Computational Linguistics.
- Arivazhagan, N., Bapna, A., Firat, O., Lepikhin, D., Johnson, M., Krikun, M., Chen, M. X., Cao, Y., Foster, G. F., Cherry, C., Macherey, W., Chen, Z., and Wu, Y. (2019). Massively multilingual neural machine translation in the wild: Findings and challenges. *CoRR*, abs/1907.05019.

---

pair or the other, resulting in relatively homogeneous training batches. This is in contrast to the truly random baseline, (ne-hi-en) random, which can have mixed data in mini-batches.

- Axelrod, A., He, X., and Gao, J. (2011). Domain adaptation via pseudo in-domain data selection. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 355–362. Association for Computational Linguistics.
- Bapna, A. and Firat, O. (2019). Simple, scalable adaptation for neural machine translation. 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 1538–1548, Hong Kong, China. Association for Computational Linguistics.
- Bengio, Y., Louradour, J., Collobert, R., and Weston, J. (2009). Curriculum Learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pages 41–48, Montreal, Quebec, Canada. ACM.
- Chih-Chun Wang, Kulkarni, S. R., and Poor, H. V. (2005). Bandit problems with side observations. *IEEE Transactions on Automatic Control*, 50(3):338–355.
- Dabre, R., Fujita, A., and Chu, C. (2019). Exploiting multilingualism through multistage fine-tuning for low-resource neural machine translation. 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 1410–1416, Hong Kong, China. Association for Computational Linguistics.
- Duh, K., Neubig, G., Sudoh, K., and Tsukada, H. (2013). Adaptation data selection using neural language models: Experiments in machine translation. In *The 51st Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 678–683, Sofia, Bulgaria.
- Durrani, N., Sajjad, H., Joty, S. R., and Abdelali, A. (2016). A deep fusion model for domain adaptation in phrase-based MT. In *COLING*, pages 3177–3187. ACL.
- Elman, J. L. (1993). Learning and development in neural networks: the importance of starting small. *Cognition*, 48(1):71 – 99.
- Finn, C., Abbeel, P., and Levine, S. (2017). Model-agnostic meta-learning for fast adaptation of deep networks. *CoRR*, abs/1703.03400.
- Freitag, M. and Al-Onaizan, Y. (2016). Fast domain adaptation for neural machine translation. *CoRR*, abs/1612.06897.
- Guzmán, F., Chen, P.-J., Ott, M., Pino, J., Lample, G., Koehn, P., Chaudhary, V., and Ranzato, M. (2019). The FLORES evaluation datasets for low-resource machine translation: Nepali–English and Sinhala–English. 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 6098–6111, Hong Kong, China. Association for Computational Linguistics.
- Johnson, M., Schuster, M., Le, Q. V., Krikun, M., Wu, Y., Chen, Z., Thorat, N., Viegas, F., Wattenberg, M., Corrado, G., Hughes, M., and Dean, J. (2017). Google’s multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.
- Kuleshov, V. and Precup, D. (2014). Algorithms for multi-armed bandit problems. *CoRR*, abs/1402.6028.
- Kumar, G., Foster, G., Cherry, C., and Krikun, M. (2019). Reinforcement learning based curriculum optimization for 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 2054–2061, Minneapolis, Minnesota. Association for Computational Linguistics.

Lakew, S. M., Lotito, Q. F., Negri, M., Turchi, M., and Federico, M. (2018). Improving zero-shot translation of low-resource languages. *CoRR*, abs/1811.01389.

Langford, J. and Zhang, T. (2008). The epoch-greedy algorithm for multi-armed bandits with side information. In Platt, J., Koller, D., Singer, Y., and Roweis, S., editors, *Advances in Neural Information Processing Systems*, volume 20. Curran Associates, Inc.

Luong, M.-T. and Manning, C. D. (2015). Stanford neural machine translation systems for spoken language domain. In *International Workshop on Spoken Language Translation*, Da Nang, Vietnam.

Moore, R. C. and Lewis, W. (2010). Intelligent selection of language model training data. In *Proceedings of the ACL 2010 Conference Short Papers*, ACLShort '10, pages 220–224. Association for Computational Linguistics.

Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., and Auli, M. (2019). fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Pandey, S., Agarwal, D., Chakrabarti, D., and Josifovski, V. (2007). Bandits for taxonomies: a model-based approach. In *SIAM Data Mining Conference*.

Rohde, D. L. and Plaut, D. C. (1994). Language acquisition in the absence of explicit negative evidence: how important is starting small? In *Cognition*, volume 72, pages 67–109.

Zhang, X., Kumar, G., Khayrallah, H., Murray, K., Gwinnup, J., Martindale, M. J., McNamee, P., Duh, K., and Carpuat, M. (2018). An empirical exploration of curriculum learning for neural machine translation. *CoRR*, abs/1811.00739.

Zhang, X., Shapiro, P., Kumar, G., McNamee, P., Carpuat, M., and Duh, K. (2019). Curriculum learning for domain adaptation in 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 Papers)*. Association for Computational Linguistics.

Zoph, B., Yuret, D., May, J., and Knight, K. (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, Austin, Texas. Association for Computational Linguistics.