Sentence-Level Adaptation for Low-Resource Neural Machine Translation

Aaron Mueller* and **Yash Kumar Lal***
Center for Language & Speech Processing
Department of Computer Science
Johns Hopkins University
{amueller,ykumar}@jhu.edu

Abstract

Current neural machine translation (NMT) approaches achieve state-of-the-art accuracy in high-resource contexts. However, NMT requires a great deal of parallel data to deliver acceptable results; thus, it is currently unsuited for translating in low-resource contexts (especially when compared to phrase-based approaches). We propose a method that better leverages the limited data available in such low-resource settings by adapting the model for each sentence at inference time. A general NMT model is trained on the limited training data; then, for each test sentence, its parameters are fine-tuned over a subset of similar sentences extracted from the training set. We experiment with various similarity metrics to extract these similar sentences. It is observed that the sentence-adapted models achieve slightly increased BLEU scores compared to standard neural approaches on a Xhosa-English dataset.

1 Introduction

Neural machine translation (NMT) (Bahdanau et al., 2014) has become the primary paradigm in machine translation literature. NMT aims to learn an end-to-end neural model to optimize translation performance by generalizing machine translation as a sequence-to-sequence machine learning problem.

The first NMT systems (Sutskever et al., 2014; Kalchbrenner and Blunsom, 2013) were built with recurrent neural networks based on encoder-decoder architectures. Bahdanau et al. (2014) and Luong et al. (2015) proposed the use of attention mechanisms to translate better by considering the context in which particular target words occur with respect to the source contexts. Recently, transformers (Vaswani et al., 2017) have been shown to achieve state-of-the-art performance across various high-resource language pairs.

The strength of this approach lies in processing large amounts of parallel data and quickly learning from aligned translations without pre-defined linguistic rules. NMT directly models the probability of a target-language sentence given aligned source- and target-language sentences and does not need to train separate language models and alignment models like statistical machine translation (SMT) (Koehn et al., 2003; Chiang, 2005). The unavailability of large parallel corpora for most language pairs, however, is a ubiquitous problem. These are only available for a handful of resource-rich languages, and in limited domains such as news reports or parliamentary/congressional proceedings.

Neural approaches to MT in general are data-hungry and therefore tend to perform inadequately in low-resource contexts (Koehn and Knowles, 2017). Thus, improving NMT for low-resource languages has been a topic of recent interest. While unsupervised NMT (Artetxe et al., 2018) has been suggested to reduce NMT's need for aligned translations, it does not perform effectively for low-resource languages (Guzmán et al., 2019). Present practices in the domain leverage the strength of preliminary word-level translation models, which do not work well. However, transfer learning from high-resource parallel datasets (Zoph et al., 2016), as well as data augmentation through pivot corpora (Choi et al., 2018), trans-

* Equal contribution.

lating monolingual data (Zhang and Zong, 2016), and/or copying data from source to target side (Currey et al., 2017) have proven effective in such cases.

Our method attempts to better leverage limited data by adapting parameters for each sentence at inference time. This is carried out by fine-tuning (Sennrich et al., 2015; Luong and Manning, 2015b) the parameters of an NMT model over a subset of training sentences which are similar to a given test sentence. By contrast, existing NMT systems tend to employ parameters which are unchanged for any given test sentence after training or continued training (Luong and Manning, 2015a).

There exists evidence that customising an NMT model for each test sentence gives it a better chance of producing correct translations (Wuebker et al., 2018). In our model, for every test sentence, a unique subset of similar training sentences is retrieved. This training-sentence subset is used to fine-tune the base model at inference time. We experiment with string-based similarity and representation-based similarity to retrieve similar sentences; precision, recall, and Levenshtein distance are used for the former, and cosine similarity on word embeddings is used for the latter. A combination of these is used to create the final subset of similar sentences.

2 Related Work

In statistical machine translation, Liu et al. (2012) proposed a local training method to learn sentence-wise weights for different test sentences. Due to the relatively lower number of weights in SMT, fine-tuning them does not fully exploit similar sentences. Koehn and Senellart (2010; Ma et al. (2011; Bertoldi and Federico (2013; Wang et al. (2013) carefully designed features to generate similar sentences and use them in the translation memory. These methods worked when the similarity of the test sentence and the sentences in the similar subset was reasonably high. Moore and Lewis (2010) proposed selecting non-domain-specific language model (LM) training data by comparing its cross-entropy with as domain-specific LM, while Duh et al. (2013) used neural LMs for adaptation data selection.

Domain adaptation (Ben-David et al., 2010; Chu and Wang, 2018) can be applied in order to learn from a source-language distribution a well performing model on a different (but related) target data distribution. Continued training (Luong and Manning, 2015a) is a commonly applied technique in domain adaptation where a general NMT system is trained on a large amounts of out-of-domain parallel data; then, the general model is adapted for a particular domain. Sentence-level adaptation is analogous to the problem of domain adaptation if each sentence is considered its own domain, and we therefore consider the sentence adaptation task a subset of the domain adaptation task. Our approach is similar to the more fine-grained document-level adaptation of Kothur et al. (2018), though we adapt on multiple complete sentences rather than populating a dictionary of novel-word translations or adapting on the previous sentence. Farajian et al. (2017) work on translations in multiple domains by generating instance-based adaptation hyperparameters in an unsupervised fashion.

Li et al. (2016) present a dynamic NMT approach where the general NMT model is adapted per-sentence; however, they adapt on only a single similar sentence and employ their system in a high-resource context. We propose additional similarity metrics and adapt on multiple similar sentences obtained from each metric. The pipeline employed here is similar to that of Zhang et al. (2018), where ”translation pieces” are extracted to improve translations for particular sentences. However, their approach uses only lexical measures of similarity—edit distance and similar n-grams—and relies on these similar lexical features as opposed to entire sentences from the training corpus. Our system employs lexical, character-based, and embedding-based similarities to retrieve sentences, making it better suited for the task.

3 Model Architecture

We discuss the various components of our proposed approach in detail. An overview of the architecture can be found in Figure 1.

3.1 Transformer

Recently, transformers (Vaswani et al., 2017) have proven highly effective in machine translation; as they process each word, self-attention allows them to peek at other positions in the input sequence itself to create a better encoding. We employ transformers as the foundation for our model.

The transformer encoder is composed of 2 sublayers: self-attention and a feedforward network.

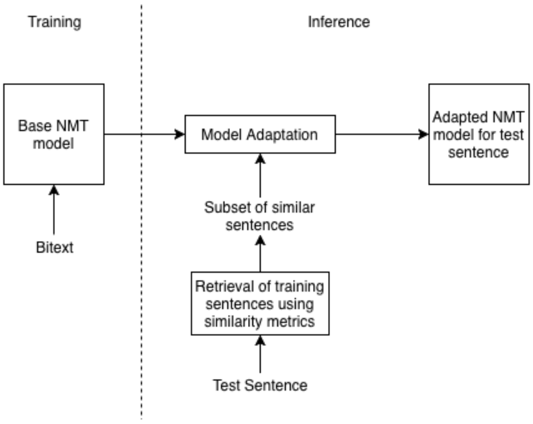
Training
Inference
Base NMT model
Bitext
Model Adaptation
Adapted NMT model for test sentence
Subset of similar sentences
Retrieval of training sentences using similarity metrics
Test Sentence

Figure 1: Architecture overview.

First, the input is used to create query, key, and value vectors. Vaswani et al. (2017) extend the dot product attention described in Luong et al. (2015) to consider these vectors. Self-attention is further refined into multi-head attention, allowing the model to focus on different parts of the input sequence at once.

Self-attention in the decoder is applied as it is in the encoder. However, the attention on future time steps is masked out to prevent from attending to future positions. The output embeddings in the decoder are offset by one position. Both of these modifications combined ensure that model predictions for any position can depend only on known outputs of previous positions. Such blocks can be stacked to form multi-layer encoders and decoders.

3.2 Similarity Metrics

Our method consists of adapting a base NMT model over a small set of relevant sentences for refinement of its parameters.

We employ four types of similarity metrics (eight total metrics) to retrieve sentences from the training set that are similar to a given test sentence. The first of these is character-based Levenshtein distance:

$$\text{distance} = \text{subs} + \text{dels} + \text{inserts} \tag{1}$$

The sentences which return the minimum distance from the test sentence are considered to be the most similar and are added to the fine-tuning subset. We expect that this metric may capture similar subwords.

Our second and third metric types employ lexical similarities between sentences. We take inspiration from BLEU (Papineni et al., 2002), which is a modified n-gram precision between a reference and generated translation. To capture lexical similarity, we count unigram, bigram, or trigram matches, then normalize over the number of n-grams in the test sentence (for recall) and the number of n-grams in the candidate sentence from the training set (for precision):

$$\text{precision} = \frac{\text{count}_{\text{match}}(\text{train}, \text{test})}{\text{count}(\text{ngrams} \in \text{train})} \tag{2}$$

$$\text{recall} = \frac{\text{count}_{\text{match}}(\text{train}, \text{test})}{\text{count}(\text{ngrams} \in \text{test})} \tag{3}$$

In Equations 2 and 3, $count_{match}$ refers to the number of matching n-grams between the sentence to be inferred ($test$) and a candidate sentence from the training set ($train$). Note that we employ three different n-gram orders (unigram, bigram, trigram) for both of these metric types, yielding six total precision- and recall-based similarity metrics.

Our fourth metric type attempts to capture semantic similarity between sentences. For this, we calculate the cosine similarity across two sentences as follows:

$$cos(\theta) = \frac{s_1 \cdot s_2}{||s_1|| \cdot ||s_2||} \tag{4}$$

Here, vectors s_1 and s_2 are the mean word embeddings (Mikolov et al., 2013) for sentence 1 and sentence 2, respectively:

$$s = \frac{\sum_{w \in W} w}{|s|} \tag{5}$$

where w is a word embedding, W is the list of all word embeddings in a given sentence s, and $|s|$ is the length of the sentence in tokens. The sum and division are element-wise operations which yield a vector of the same length as any given w in W. Although simple, it has been demonstrated that this is a strong method to generate sentence embeddings (Arora et al., 2017).

Each metric (precision and recall of different n-gram orders account for six out of the eight distinct metrics) contributes 11 sentences[1] each to the final adaptation subset. We present a list of similar sentences retrieved by each metric for a sample test sentence in Table 6.

Since each of the metrics is calculated against all training sentences, this approach is more suited for a low-resource setting rather than a high-resource one.

[1] This is an arbitrarily chosen number.

	Sentence	Eng Words	Xhosa Words
Train	20544	614441	388778
Test	1956	58323	36700
Dev	1956	59140	37353

Table 1: English-Xhosa Bible dataset at a glance.

α_A	BLEU
0.0001	22.51
0.0004	**22.83**
0.00045	22.82
0.0005	22.82
0.0006	22.59
0.00075	22.26
0.001	19.12

Table 2: Learning rate during adaptation (α_A) vs. BLEU scores in the Xhosa→English translation task. Note: $\alpha_T = 0.0005$.

3.3 Inference-Time Adaptation

Our pipeline is split into two stages. First, network parameters are calculated by training over the entire training corpus; this is denoted as θ. This corresponds to the training stage of Figure 1. Second, the parameters θ are modified slightly to increase the log-likelihood over the subset of sentences which are similar to the test sentence (that are extracted using the similarity metrics in Section 3.2). The modified parameters are denoted as $\hat{\theta}$. This is formalized as follows:

$$\hat{\theta} = \arg\max_{\theta} \left(\log \prod_{S^{(k)} \sim S} p(T^{(k)} | S^{(k)}; \theta) \right) \quad (6)$$

where S denotes the source-language corpus of similar sentences, T denotes the target-language corpus of similar sentences, $S^{(k)}$ and $T^{(k)}$ denote the k^{th} sentence in the aligned corpus, and $\hat{\theta}$ refers to the network parameters of the adapted model. These computations[2] occur in the inference stage of Figure 1.

4 Experiments

4.1 Data

We translate Xhosa—a true low-resource language—to English, employing translated Bibles as our dataset (Christodouloupoulos and Steedman, 2015).[3] Dataset statistics are available in Table 1. We work with word-level data for our experiments.

Xhosa is a Niger-Congo language spoken by approximately 8 million native speakers and 11 million L2 speakers (Lewis, 2015). Relative to English, it is a synthetic language with a rich morpheme inventory (Oosthuysen, 2016). Due to Xhosa's synthetic morphology, its English translations often demonstrate one-to-many relations; i.e., one Xhosa word will often translate as multiple English words, which explains the disparity between the number of Xhosa tokens and English tokens in our dataset.

4.2 Training Details

All neural models herein are trained with Sockeye (Hieber et al., 2017).

For each of the similarity metrics, we retrieve the most similar sentences and concatenate them into a single dataset, generating a total adaptation subset of 88 sentences for each test sentence (11 per metric). As the adaptation dataset is small compared to the training corpus, special care is needed to optimize strategic overfitting during inference; we therefore restrict adaptation to just one epoch.

4.2.1 Adaptation Learning Rate Experiments

The learning rate for adaptation α_A essentially dictates how much fine-tuning the NMT system receives during adaptation. Each language has a different ideal adaptation rate, so we perform a sweep and report our findings in Table 2.

It is clear that trying to learn very aggressively from the adaptation subset results in a decrease in performance. Trying to adjust the weights of the network too much with respect to the loss function might result in disregarding some local minima from consideration, resulting in an adverse effect. It is also found that setting α_A too low also results in a slight score decrease, so finding the optimal α_A is crucial. It is observed that, in this case, an α_A of 0.0004 best suits our objective. Note that this is similar to the training learning rate α_T of 0.0005, and that the other best-performing α_A values are similar to α_T as well.

[2] Note that we pre-compute similar sentences before running inference; this saves time when translating sentences at test time. We do not peek at or manually modify the similar sentences for any test sentence.

[3] Religious texts are often the first to be translated into a given language. Translated Bibles are therefore available for many low-resource language varieties.

Base Model	Unadapted	$\alpha_A = .0004$	$\alpha_A = .0005$
LSTM (Luong et al., 2015)	20.73	-	-
Transformer ($\alpha_T = .0001$) (Vaswani et al., 2017)	20.52	-	17.74
Transformer ($\alpha_T = .0005$) (Vaswani et al., 2017)	**22.76**	**22.83**	22.82

Table 3: Evaluation of Xhosa→English translation systems.

src	Wathi uThixo , Makubekho isibhakabhaka phakathi kwawo amanzi , sibe ngumahlulo wokwahlula amanzi kumanzi .
ref	And God said , Let there be a firmament in the midst of the waters , and let it divide the waters from the waters .
no adaptation	And God said , Let there be **clouds in the midst of them , let the water of the morning to the water** .
w/ adaptation	And God said , Let there be **clouds** *in the midst of the waters* to *divide the water from the waters* .
src	Wathi uYehova uThixo kumfazi , Yintoni na le nto uyenzileyo ? Wathi umfazi , Inyoka indilukuhlile , ndadla ke .
ref	And the LORD God said unto the woman , What is this that thou hast done ? And the woman said , The serpent beguiled me , and I did eat .
no adaptation	*And the LORD God said unto the woman* , What hast thou done this thing ? And she said , **I have eaten the wife** , and did eat .
w/ adaptation	*And the LORD God said unto the woman* , What hast thou done ? And the woman said , **I have eaten** , and did eat .

Table 4: Sample translations comparing unadapted and adapted output. Notably poor translations are highlighted in **red bold**, whereas notably good translations are highlighted in *blue italics*.

Metric	Unadapted	Adapted
Unigram Match %	53.9	54.1
Bigram Match %	28.4	28.5
Trigram Match %	16.7	16.7
4-gram Match %	10.5	10.6
Brevity Penalty	1.000	1.000

Table 5: Investigation of the constituent features of our BLEU scores for Xhosa→English translations.

4.3 Baselines

We focus on comparing the performance of neural models, as this work extends NMT for low-resource contexts.

The first neural model against which we evaluate our approach is the standard encoder-decoder architecture with recurrent units. The encoder units are bidirectional LSTMs (Schuster and Paliwal, 1997) while the decoder unit incorporates an LSTM (Hochreiter and Schmidhuber, 1997) with dot product attention (Luong et al., 2015). The model was trained with a word batch size of 1024, with source and target embedding layer size 256 and hidden layer size 512. The initial learning rate was set to 0.0001 with a decay factor of 0.9. We impose a dropout rate (Srivastava et al., 2014) of 0.1 and use the Adam optimizer (Kingma and Ba, 2015).

The second baseline is a Transformer architecture. Both the encoder and decoder have two sublayers employing multi-head attention. The number of heads in this mechanism is 4. Other parameters are kept constant from the LSTM model. As the transformer model outperforms the LSTM (see Table 3), we use it as the base of our adapted model.

5 Results and Evaluation

Table 3 contains all BLEU scores for our unadapted and adapted models. While it may seem beneficial in theory to have α_T be less than α_A, we find empirically that having similar α_T and α_A values results in better BLEU scores. The base transformer trained with a learning rate α_T of 0.0001 performs more poorly compared to that with an α_T of .0005. We therefore focus primarily on models where $\alpha_T = 0.0005$. Both of these trends could be because we are "adapting" on a subset of the data on which we train.

The percentage of n-gram matches (unigram to 4-gram) is higher for the adapted model than the other neural approaches; see Table 5. This suggests that we match more lexical content to the reference translations; this causes increased fluency and semantic similarity. Indeed, our model narrows the lexical matching gap between the baseline transformer and the phrase-based system. This leads to a slight increase in BLEU scores for the generated translations.

Sample translations may be found in Table 4; these were chosen randomly from the output translations. Note that the example translations from the adapted model tend to be more fluent than the translations from the unadapted model due to not including as many non-sequitur tokens.

The adequacy of the adapted translations also seems to be slightly better (or at least no worse): the only non-matching lexical translation in the first sample (*clouds*, as opposed to the reference *firmament*) is semantically close to the reference. Compare to the unadapted model's sentence, whose second clause is semantically unacceptable and bears little resemblance to the reference translation's intended semantic value. Similarly, in the second sentence, the adapted model has a similar non-sequitur translation for the highlighted clause, although the adapted model's translation omits more non-sequitur words to produce a more fluent translation without losing as much adequacy as the unadapted model's translation.

5.1 Qualitative Sentence Similarity Metric Evaluation

To investigate what types of sentences are retrieved by our similarity metrics from Section 3.2, we run a script which retrieves the most similar training sentences (per-metric) for a randomly chosen test sentence in English. The most similar sentences per-metric, as well as their similarity/distance scores, are shown in Table 6. Note that this sentence similarity process is run for only the source language, Xhosa, and that this set of similar sentences in English is retrieved solely to demonstrate what types of sentences these similarity metrics choose in general.

Notably, precision and recall sometimes result in different similar sentences for the same n-gram orders. Unigram precision and unigram recall retrieve largely distinct sentences with very different scores, though there is often overlap: unigram recall, bigram precision, and bigram recall return the same sentence as most similar. Trigram precision and recall return similar sentences that are distinct from the previous n-gram orders; the precision and recall sentences are the same in this case, but not always. Thus, using different n-gram orders—and precision as well as recall within each n-gram order—can feasibly return different similar sentences. We thus keep all of these similarity metrics in our similar-sentence subset.

Cosine similarity retrieves a sentence which has a similar general tone to the test sentence, as well as a similar topic (the story of creation), but otherwise the n-grams are quite different. This seems to be beneficial, for it demonstrates that we retrieve sentences which do not necessarily have the same words as the sentence on which we perform inference, but which have commonalities with respect to some supralinguistic or semantic feature(s). This trend also holds for other sentences in the test set for which we retrieved similar sentences, so it does generally seem to return related sentences.

Levenshtein distance, in contrast, does not seem to return a useful similar sentence in this example. There are few n-gram or morphemic matches in common between the test and similar sentences, and the meaning of the retrieved sentence bears little resemblance to that of the test sentence. In general, the Levenshtein distance seems useful in retrieving similar sentences with different inflections of the same words primarily when there exists another sentence with similar unigrams in the same order as the test sentence (i.e., it works primarily when two sentences exist that are already very lexically similar). In the future, it would perhaps it would be more beneficial to run Levenshtein distance on subwords after performing a BPE operation, rather than on characters. As this metric only comprises a small fraction of the similar-sentence subset on which we adapt, it should be inconsequential if some sentences are not particularly relevant from this metric. If they are relevant, however, it will be quite beneficial, so we keep these sentences in our similar-sentence adaptation set regardless.

We observe that sometimes, a sentence with zero or negligible score is also returned by one of the metrics. As an extension, thresholding the

test sentence	Behold , this is the joy of his way , and out of the earth shall others grow .	
levenshtein distance	And the evening and the morning were the third day .	54
unigram precision	And God said , Let the earth bring forth grass , the herb yielding seed , and the fruit tree yielding fruit after his kind , whose seed is in itself , upon the earth : and it was so .	0.8421
unigram recall	And the earth was without form , and void ; and darkness was upon the face of the deep . And the Spirit of God moved upon the face of the waters .	0.455
bigram precision	And the earth was without form , and void ; and darkness was upon the face of the deep . And the Spirit of God moved upon the face of the waters .	0.222
bigram recall	And the earth was without form , and void ; and darkness was upon the face of the deep . And the Spirit of God moved upon the face of the waters .	0.125
trigram precision	And he shewed me a pure river of water of life , clear as crystal , proceeding out of the throne of God and of the Lamb .	0.059
trigram recall	And he shewed me a pure river of water of life , clear as crystal , proceeding out of the throne of God and of the Lamb .	0.038
cosine similarity	And God said , Let there be light : and there was light .	0.397

Table 6: This table features the most similar sentence retrieved from the training set per similarity metric for an arbitrary test sentence. Note that Levenshtein distance is a distance metric and not a similarity metric, so we retrieve the minimum-distance sentence as opposed to the highest-similarity sentence.

score for each metric when retrieving similar sentences might boost performance since it will only return higher quality matches.

6 Conclusion

We propose an architecture-independent approach to give neural models a better chance of leveraging limited parallel data in low-resource contexts. The model produced by adapting the low-resource NMT model per-sentence generates translations with slightly higher adequacy and seemingly improved fluency; BLEU scores are similar, though in this case slightly higher after adaptation. We note in particular that tuning both the training-time and adaptation-time learning rates is crucial; extensions could therefore test different values in a grid search for linguistically diverse language pairs.

Future work could also refine the similar-sentence adaptation subset and threshold sentences according to some interpolated metric based on all similarity metrics. The flexibility of our approach means that it is easy to integrate other similar algorithms as new similarity metrics. In particular, bilateral multi-perspective matching (Wang et al., 2017) at the sentence level could be of interest. Another possible extension is to look at subword-level matching criteria for the retrieval component of our approach. One could also study the relative performance of this approach for synthetic vs. analytic languages with different neural model base architectures before adaptation.

References

Arora, Sanjeev, Yingyu Liang, and Tengyu Ma. 2017. A Simple but Tough-to-Beat Baseline for Sentence Embeddings. In *International Conference on Learning Representations*.

Artetxe, Mikel, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018. Unsupervised Neural Machine Translation. In *International Conference on Learning Representations*.

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv preprint arXiv:1409.0473*.

Ben-David, Shai, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Vaughan. 2010. A Theory of Learning from Different Domains. *Machine Learning*, 79:151–175.

Bertoldi, N., Cettolo M. and M. Federico. 2013. Cache-based online adaptation for machine translation enhanced computer assisted translation. In *Pro-*

ceedings of the XIV Machine Translation Summit, pages 35–42.

Chiang, David. 2005. A Hierarchical Phrase-based Model for Statistical Machine Translation. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 263–270.

Choi, Gyu Hyeon, Jong Hun Shin, and Young Kil Kim. 2018. Improving a Multi-Source Neural Machine Translation Model with Corpus Extension for Low-Resource Languages. In chair), Nicoletta Calzolari (Conference, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Koiti Hasida, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hlne Mazo, Asuncion Moreno, Jan Odijk, Stelios Piperidis, and Takenobu Tokunaga, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May 7-12, 2018. European Language Resources Association (ELRA).

Christodouloupoulos, Christos and Mark Steedman. 2015. A Massively Parallel Corpus: The Bible in 100 Languages. *Lang. Resour. Eval.*, 49(2):375–395, June.

Chu, Chenhui and Rui Wang. 2018. A Survey of Domain Adaptation for Neural Machine Translation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1304–1319. Association for Computational Linguistics.

Currey, Anna, Antonio Valerio Miceli Barone, and Kenneth Heafield. 2017. Copied Monolingual Data Improves Low-Resource Neural Machine Translation. In *Proceedings of the Second Conference on Machine Translation*, pages 148–156. Association for Computational Linguistics.

Duh, Kevin, Sudoh Katsuhito Neubig, Graham, and Hajime Tsukada. 2013. Adaptation Data Selection using Neural Language Models: Experiments in Machine Translation. In *Proceedings of the 51th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Sofia, Bulgaria, August. Association for Computational Linguistics.

Farajian, M. Amin, Marco Turchi, Matteo Negri, and Marcello Federico. 2017. Multi-Domain Neural Machine Translation through Unsupervised Adaptation. In *Proceedings of the Second Conference on Machine Translation*, pages 127–137. Association for Computational Linguistics.

Guzmán, Francisco, Peng-Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc'Aurelio Ranzato. 2019. Two New Evaluation Datasets for Low-Resource Machine Translation: Nepali-English and Sinhala-English. *arXiv preprint arXiv:1902.01382*.

Hieber, Felix, Tobias Domhan, Michael Denkowski, David Vilar, Artem Sokolov, Ann Clifton, and Matt Post. 2017. Sockeye: A Toolkit for Neural Machine Translation. *CoRR*, abs/1712.05690.

Hochreiter, Sepp and Jürgen Schmidhuber. 1997. Long Short-term Memory. *Neural computation*, 9(8):1735–1780.

Kalchbrenner, Nal and Phil Blunsom. 2013. Recurrent Continuous Translation Models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709. Association for Computational Linguistics.

Kingma, Diederik P. and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*.

Koehn, Philipp and Rebecca Knowles. 2017. Six Challenges for Neural Machine Translation. In *Proceedings of the First Workshop on Neural Machine Translation*, pages 28–39. Association for Computational Linguistics.

Koehn, Philipp and Jean Senellart. 2010. Convergence of Translation Memory and Statistical Machine Translation. In *Proceedings of AMTA Workshop on MT Research and the Translation Industry*, pages 21–31, Denver.

Koehn, Philipp, Franz Josef Och, and Daniel Marcu. 2003. Statistical Phrase-based Translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, pages 48–54.

Kothur, Sachith Sri Ram, Rebecca Knowles, and Philipp Koehn. 2018. Document-Level Adaptation for Neural Machine Translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 64–73. Association for Computational Linguistics.

Lewis, M. Paul, editor. 2015. *Ethnologue: Languages of the World*. SIL International, Dallas, TX, USA, eighteenth edition.

Li, Xiaoqing, Jiajun Zhang, and Chengqing Zong. 2016. One Sentence One Model for Neural Machine Translation. *CoRR*, abs/1609.06490.

Liu, Lemao, Hailong Cao, Taro Watanabe, Tiejun Zhao, Mo Yu, and CongHui Zhu. 2012. Locally Training the Log-linear Model for SMT. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 402–411.

Luong, Minh-Thang and Christopher D. Manning. 2015a. Neural Machine Translation Systems for Spoken Language Domains. In *International Workshop on Spoken Language Translation*.

Luong, Minh-Thang and Christopher D. Manning. 2015b. Stanford Neural Machine Translation Systems for Spoken Language Domain. In *International Workshop on Spoken Language Translation*, Da Nang, Vietnam.

Luong, Minh-Thang, Hieu Pham, and Christopher D Manning. 2015. Effective Approaches to Attention-based Neural Machine Translation. *arXiv preprint arXiv:1508.04025*.

Ma, Yanjun, Yifan He, Andy Way, and Josef van Genabith. 2011. Consistent Translation Using Discriminative Learning: A Translation Memory-inspired Approach. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 1239–1248.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems*.

Moore, Robert C. and William Lewis. 2010. Intelligent Selection of Language Model Training Data. In *Proceedings of the ACL 2010 Conference Short Papers*, ACLShort '10, pages 220–224, Stroudsburg, PA, USA. Association for Computational Linguistics.

Oosthuysen, JC. 2016. *The Grammar of isiXhosa*. African Sun Media, Stellenbosch, South Africa.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 311–318.

Schuster, Mike and Kuldip K Paliwal. 1997. Bidirectional Recurrent Neural Networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Sennrich, Rico, Barry Haddow, and Alexandra Birch. 2015. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*.

Srivastava, Nitish, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.*, pages 1929–1958.

Sutskever, Ilya, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All You Need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Wang, Kun, Chengqing Zong, and Keh-Yih Su. 2013. Integrating Translation Memory into Phrase-Based Machine Translation during Decoding. In *ACL*.

Wang, Zhiguo, Wael Hamza, and Radu Florian. 2017. Bilateral Multi-perspective Matching for Natural Language Sentences. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, pages 4144–4150. AAAI Press.

Wuebker, Joern, Patrick Simianer, and John DeNero. 2018. Compact Personalized Models for Neural Machine Translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 881–886. Association for Computational Linguistics.

Zhang, Jiajun and Chengqing Zong. 2016. Exploiting Source-side Monolingual Data in Neural Machine Translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545. Association for Computational Linguistics.

Zhang, Jingyi, Masao Utiyama, Eiichro Sumita, Graham Neubig, and Satoshi Nakamura. 2018. Guiding Neural Machine Translation with Retrieved Translation Pieces. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1325–1335. Association for Computational Linguistics.

Zoph, Barret, 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. Association for Computational Linguistics.