Transfer Learning for Low-Resource Neural Machine Translation

Barret Zoph*
Information Sciences Institute
University of Southern California
barretzoph@gmail.com

Deniz Yuret
Computer Engineering
Koç University
dyuret@ku.edu.tr

Jonathan May and Kevin Knight
Information Sciences Institute
Computer Science Department
University of Southern California
{jonmay,knight}@isi.edu

Abstract

The encoder-decoder framework for neural machine translation (NMT) has been shown effective in large data scenarios, but is much less effective for low-resource languages. We present a transfer learning method that significantly improves BLEU scores across a range of low-resource languages. Our key idea is to first train a high-resource language pair (the *parent model*), then transfer some of the learned parameters to the low-resource pair (the *child model*) to initialize and constrain training. Using our transfer learning method we improve baseline NMT models by an average of 5.6 BLEU on four low-resource language pairs. Ensembling and unknown word replacement add another 2 BLEU which brings the NMT performance on low-resource machine translation close to a strong syntax based machine translation (SBMT) system, exceeding its performance on one language pair. Additionally, using the transfer learning model for re-scoring, we can improve the SBMT system by an average of 1.3 BLEU, improving the state-of-the-art on low-resource machine translation.

1 Introduction

Neural machine translation (NMT) (Sutskever et al., 2014) is a promising paradigm for extracting translation knowledge from parallel text. NMT systems have achieved competitive accuracy rates under large-data training conditions for language pairs such as English–French. However, neural methods are data-hungry and learn poorly from low-count events. This behavior makes vanilla NMT a poor choice for low-resource languages, where parallel data is scarce. Table 1 shows that for 4 low-resource languages, a standard string-to-tree statistical MT system (SBMT) (Galley et al., 2004; Galley et al., 2006) strongly outperforms NMT, even when NMT uses the state-of-the-art local attention plus feed-input techniques from Luong et al. (2015a).

Language	Train Size	Test Size	SBMT BLEU	NMT BLEU
Hausa	1.0m	11.3K	23.7	16.8
Turkish	1.4m	11.6K	20.4	11.4
Uzbek	1.8m	11.5K	17.9	10.7
Urdu	0.2m	11.4K	17.9	5.2

Table 1: NMT models with attention are outperformed by standard string-to-tree statistical MT (SBMT) when translating low-resource languages into English. Train/test bitext corpus sizes are English token counts. Single-reference, case-insensitive BLEU scores are given for held-out test corpora.

In this paper, we describe a method for substantially improving NMT results on these languages. Our key idea is to first train a high-resource language pair, then use the resulting trained network (the *parent model*) to initialize and constrain training for our low-resource language pair (the *child model*). We find that we can optimize our results by fixing certain parameters of the parent model and letting the rest be fine-tuned by the child model. We report NMT improvements from transfer learning of 5.6 BLEU on average, and we provide an analysis of why the method works. The final NMT system

This work was carried out while all authors were at USC's Information Sciences Institute.

*This author is currently at Google Brain.

Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1568–1575,
Austin, Texas, November 1-5, 2016.

approaches strong SBMT baselines in all four language pairs, and exceeds SBMT performance in one of them. Furthermore, we show that NMT is an exceptional re-scorer of 'traditional' MT output; even NMT that on its own is worse than SBMT is consistently able to improve upon SBMT system output when incorporated as a re-scoring model.

We provide a brief description of our NMT model in Section 2. Section 3 gives some background on transfer learning and explains how we use it to improve machine translation performance. Our main experiments translating Hausa, Turkish, Uzbek, and Urdu into English with the help of a French–English parent model are presented in Section 4. Section 5 explores alternatives to our model to enhance understanding. We find that the choice of parent language pair affects performance, and provide an empirical upper bound on transfer performance using an artificial language. We experiment with English-only language models, copy models, and word-sorting models to show that what we transfer goes beyond monolingual information and that using a translation model trained on bilingual corpora as a parent is essential. We show the effects of freezing, fine-tuning, and smarter initialization of different components of the attention-based NMT system during transfer. We compare the learning curves of transfer and no-transfer models, showing that transfer solves an overfitting problem, not a search problem. We summarize our contributions in Section 6.

2 NMT Background

In the neural encoder-decoder framework for MT (Neco and Forcada, 1997; Castaño and Casacuberta, 1997; Sutskever et al., 2014; Bahdanau et al., 2014; Luong et al., 2015a), we use a recurrent neural network (encoder) to convert a source sentence into a dense, fixed-length vector. We then use another recurrent network (decoder) to convert that vector to a target sentence. In this paper, we use a two-layer encoder-decoder system (Figure 1) with long short-term memory (LSTM) units (Hochreiter and Schmidhuber, 1997). The models were trained to optimize maximum likelihood (via a softmax layer) with back-propagation through time (Werbos, 1990). Additionally, we use an attention mechanism that allows the target decoder to look back at

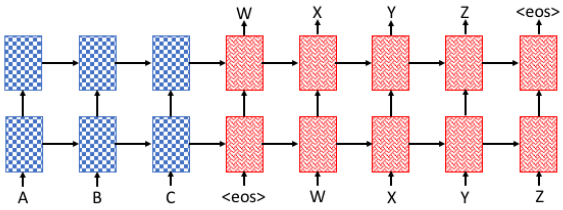

Figure 1: The encoder-decoder framework for neural machine translation (NMT) (Sutskever et al., 2014). Here, a source sentence C B A (presented in reverse order as A B C) is translated into a target sentence W X Y Z. At each step, an evolving real-valued vector summarizes the state of the encoder (blue, checkerboard) and decoder (red, lattice). Not shown here are the attention connections present in our model used by the decoder to access encoder states.

the source encoder, specifically the local attention model from Luong et al. (2015a). In our model we also use the feed-input input connection from Luong et al. (2015a) where at each timestep on the decoder we feed in the top layer's hidden state into the lowest layer of the next timestep.

3 Transfer Learning

Transfer learning uses knowledge from a learned task to improve the performance on a related task, typically reducing the amount of required training data (Torrey and Shavlik, 2009; Pan and Yang, 2010). In natural language processing, transfer learning methods have been successfully applied to speech recognition, document classification and sentiment analysis (Wang and Zheng, 2015). Deep learning models discover multiple levels of representation, some of which may be useful across tasks, which makes them particularly suited to transfer learning (Bengio, 2012). For example, Cireşan et al. (2012) use a convolutional neural network to recognize handwritten characters and show positive effects of transfer between models for Latin and Chinese characters. Ours is the first study to apply transfer learning to neural machine translation.

There has also been work on using data from multiple language pairs in NMT to improve performance. Recently, Dong et al. (2015) showed that sharing a source encoder for one language helps performance when using different target decoders

Decoder	Hausa	Turkish	Uzbek	Urdu
NMT	16.8	11.4	10.7	5.2
Xfer	21.3	17.0	14.4	13.8
Final	24.0	18.7	16.8	14.5
SBMT	23.7	20.4	17.9	17.9

Table 2: Our method significantly improves NMT results for the translation of low-resource languages into English. Results show test-set BLEU scores. The 'NMT' row shows results without transfer, and the 'Xfer' row shows results with transfer. The 'Final' row shows BLEU after we ensemble 8 models and use unknown word replacement.

Re-scorer	SBMT Decoder			
	Hausa	Turkish	Uzbek	Urdu
None	23.7	20.4	17.9	17.9
NMT	24.5	21.4	19.5	18.2
Xfer	**24.8**	**21.8**	**19.5**	**19.1**
LM	23.6	21.1	17.9	18.2

Table 3: Our transfer method applied to re-scoring output n-best lists from the SBMT system. The first row shows the SBMT performance with no re-scoring and the other 3 rows show the performance after re-scoring with the selected model. Note: the 'LM' row shows the results when an RNN LM trained on the large English corpus was used to re-score.

for different languages. In that paper the authors showed that using this framework improves performance for low-resource languages by incorporating a mix of low-resource and high-resource languages. Firat et al. (2016) used a similar approach, employing a separate encoder for each source language, a separate decoder for each target language, and a shared attention mechanism across all languages. They then trained these components jointly across multiple different language pairs to show improvements in a lower-resource setting.

There are a few key differences between our work and theirs. One is that we are working with truly small amounts of training data. Dong et al. (2015) used a training corpus of about 8m English words for the low-resource experiments, and Firat et al. (2016) used from 2m to 4m words, while we have at most 1.8m words, and as few as 0.2m. Additionally, the aforementioned previous work used the same domain for both low-resource and high-resource languages, while in our case the datasets come from vastly different domains, which makes the task much harder and more realistic. Our approach only requires using one additional high-resource language, while the other papers used many. Our approach also allows for easy training of new low-resource languages, while Dong et al. (2015) and Firat et al. (2016) do not specify how a new language should be added to their pipeline once the models are trained. Finally, Dong et al. (2015) observe an average BLEU gain on their low-resource experiments of +1.16, and Firat et al. (2016) obtain BLEU gains of +1.8, while we see a +5.6 BLEU gain.

The transfer learning approach we use is simple and effective. We first train an NMT model on a large corpus of parallel data (e.g., French–English). We call this the *parent model*. Next, we initialize an NMT model with the already-trained parent model. This new model is then trained on a very small parallel corpus (e.g., Uzbek–English). We call this the *child model*. Rather than starting from a random position, the child model is initialized with the weights from the parent model.

A justification for this approach is that in scenarios where we have limited training data, we need a strong prior distribution over models. The parent model trained on a large amount of bilingual data can be considered an anchor point, the peak of our prior distribution in model space. When we train the child model initialized with the parent model, we fix parameters likely to be useful across tasks so that they will not be changed during child model training. In the French–English to Uzbek–English example, as a result of the initialization, the English word embeddings from the parent model are copied, but the Uzbek words are initially mapped to random French embeddings. The parameters of the English embeddings are then frozen, while the Uzbek embeddings' parameters are allowed to be modified, i.e. *fine-tuned*, during training of the child model. Freezing certain transferred parameters and fine tuning others can be considered a hard approximation to a tight prior or strong regularization applied to some of the parameter space. We also experiment with ordinary L2 regularization, but find it does not significantly improve over the parameter freezing described above.

Our method results in large BLEU increases for a variety of low resource languages. In one of the

Language Pair	Role	Train Size	Dev Size	Test Size
Spanish–English	child	2.5m	58k	59k
French–English	parent	53m	58k	59k
German–English	parent	53m	58k	59k

Table 4: Data used for a low-resource Spanish–English task. Sizes are English-side token counts.

Parent	BLEU ↑	PPL ↓
none	16.4	15.9
French–English	**31.0**	**5.8**
German–English	29.8	6.2

Table 5: For a low-resource Spanish–English task, we experiment with several choices of parent model: none, French–English, and German–English. We hypothesize that French–English is best because French and Spanish are similar.

four language pairs our NMT system using transfer beats a strong SBMT baseline. Not only do these transfer models do well on their own, they also give large gains when used for re-scoring n-best lists ($n = 1000$) from the SBMT system. Section 4 details these results.

4 Experiments

To evaluate how well our transfer method works we apply it to a variety of low-resource languages, both stand-alone and for re-scoring a strong SBMT baseline. We report large BLEU increases across the board with our transfer method.

For all of our experiments with low-resource languages we use French as the parent source language and for child source languages we use Hausa, Turkish, Uzbek, and Urdu. The target language is always English. Table 1 shows parallel training data set sizes for the child languages, where the language with the most data has only 1.8m English tokens. For comparison, our parent French–English model uses a training set with 300 million English tokens and achieves 26 BLEU on the development set. Table 1 also shows the SBMT system scores along with the NMT baselines that do not use transfer. There is a large gap between the SBMT and NMT systems when our transfer method is not used.

The SBMT system used in this paper is a string-to-tree statistical machine translation system (Galley et al., 2006; Galley et al., 2004). In this system there are two count-based 5-gram language models. One is trained on the English side of the WMT 2015 English–French dataset and the other is trained on the English side of the low-resource bitext. Additionally, the SBMT models use thousands of sparsely-occurring, lexicalized syntactic features (Chiang et al., 2009).

For our NMT system, we use development sets for Hausa, Turkish, Uzbek, and Urdu to tune the learning rate, parameter initialization range, dropout rate, and hidden state size for all the experiments. For training we use a minibatch size of 128, hidden state size of 1000, a target vocabulary size of 15K, and a source vocabulary size of 30K. The child models are trained with a dropout probability of 0.5, as in Zaremba et al. (2014). The common parent model is trained with a dropout probability of 0.2. The learning rate used for both child and parent models is 0.5 with a decay rate of 0.9 when the development perplexity does not improve. The child models are all trained for 100 epochs. We re-scale the gradient when the gradient norm of all parameters is greater than 5. The initial parameter range is [-0.08, +0.08]. We also initialize our forget-gate biases to 1 as specified by Józefowicz et al. (2015) and Gers et al. (2000). For decoding we use a beam search of width 12.

4.1 Transfer Results

The results for our transfer learning method applied to the four languages above are in Table 2. The parent models were trained on the WMT 2015 (Bojar et al., 2015) French–English corpus for 5 epochs. Our baseline NMT systems ('NMT' row) all receive a large BLEU improvement when using the transfer method (the 'Xfer' row) with an average BLEU improvement of 5.6. Additionally, when we use unknown word replacement from Luong et al. (2015b) and ensemble together 8 models (the 'Final' row) we further improve upon our BLEU scores, bringing the average BLEU improvement to 7.5. Overall our method allows the NMT system to reach competitive scores and outperform the SBMT system in one of the four language pairs.

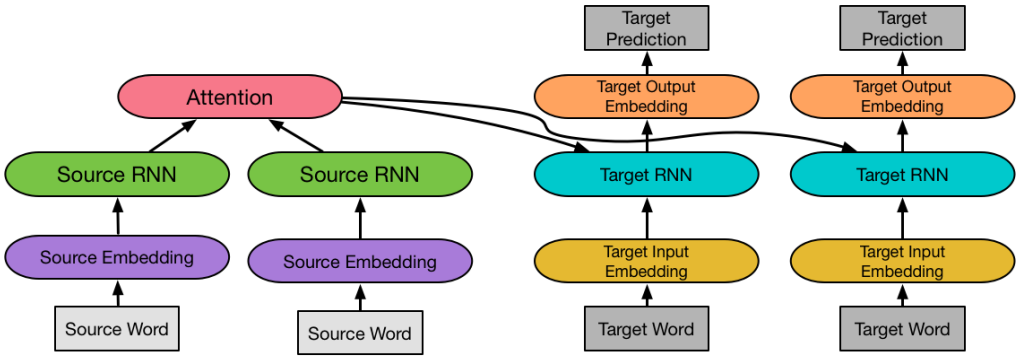

Figure 2: Our NMT model architecture, showing six blocks of parameters, in addition to source/target words and predictions. During transfer learning, we expect the source-language related blocks to change more than the target-language related blocks.

Language Pair	Parent	Train Size	BLEU ↑	PPL ↓
Uzbek–English	None	1.8m	10.7	22.4
	French–English	1.8m	**15.0 (+4.3)**	**13.9**
French′–English	None	1.8m	13.3	28.2
	French–English	1.8m	**20.0 (+6.7)**	**10.9**

Table 6: A better match between parent and child languages should improve transfer results. We devised a child language called French′, identical to French except for word spellings. We observe that French transfer learning helps French′ (13.3→20.0) more than it helps Uzbek (10.7→15.0).

4.2 Re-scoring Results

We also use the NMT model with transfer learning as a feature when re-scoring output n-best lists ($n = 1000$) from the SBMT system. Table 3 shows the results of re-scoring. We compare re-scoring with transfer NMT to re-scoring with baseline (i.e. non-transfer) NMT and to re-scoring with a neural language model. The neural language model is an LSTM RNN with 2 layers and 1000 hidden states. It has a target vocabulary of 100K and is trained using noise-contrastive estimation (Mnih and Teh, 2012; Vaswani et al., 2013; Baltescu and Blunsom, 2015; Williams et al., 2015). Additionally, it is trained using dropout with a dropout probability of 0.2 as suggested by Zaremba et al. (2014). Re-scoring with the transfer NMT model yields an improvement of 1.1–1.6 BLEU points above the strong SBMT system; we find that transfer NMT is a better re-scoring feature than baseline NMT or neural language models.

In the next section, we describe a number of additional experiments designed to help us understand the contribution of the various components of our transfer model.

5 Analysis

We analyze the effects of using different parent models, regularizing different parts of the child model, and trying different regularization techniques.

5.1 Different Parent Languages

In the above experiments we use French–English as the parent language pair. Here, we experiment with different parent languages. In this set of experiments we use Spanish–English as the child language pair. A description of the data used in this section is presented in Table 4.

Our experimental results are shown in Table 5, where we use French and German as parent languages. If we just train a model with no transfer on a small Spanish–English training set we get a BLEU score of 16.4. When using our transfer method we get Spanish–English BLEU scores of 31.0 and 29.8 via French and German parent languages, respectively. As expected, French is a better parent than German for Spanish, which could be the result of the parent language being more similar to the child language. We suspect using closely-related parent language pairs would improve overall quality.

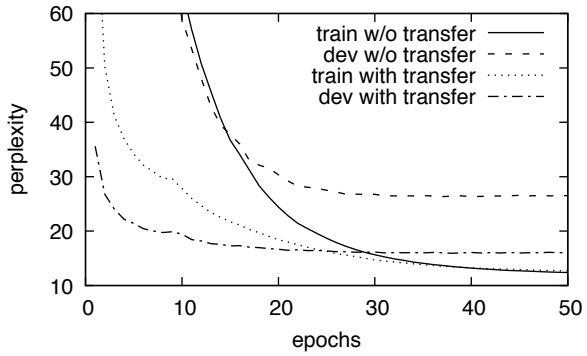

Figure 3: Uzbek–English learning curves for the NMT attention model with and without transfer learning. The training perplexity converges to a similar value in both cases. However, the development perplexity for the transfer model is significantly better.

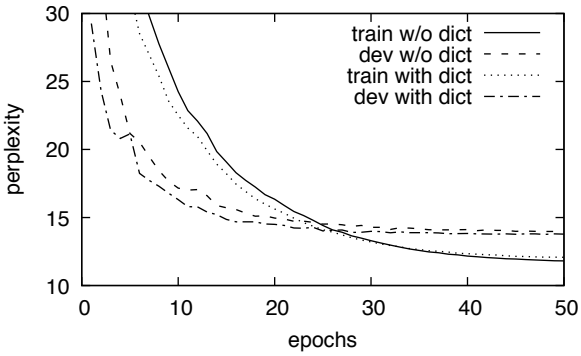

Figure 4: Uzbek–English learning curves for the transfer model with and without dictionary-based assignment of Uzbek word types to French word embeddings (from the parent model). Dictionary-based assignment enables faster improvement in early epochs. The model variants converge, showing that the unaided model is able to untangle the initial random Uzbek/French word-type mapping without help.

5.2 Effects of having Similar Parent Language

Next, we look at a best-case scenario in which the parent language is as similar as possible to the child language.

Here we devise a synthetic child language (called French′) which is exactly like French, except its vocabulary is shuffled randomly. (e.g., "internationale" is now "pomme," etc). This language, which looks unintelligible to human eyes, nevertheless has the same distributional and relational properties as actual French, i.e. the word that, prior to vocabulary reassignment, was 'roi' (king) is likely to share distributional characteristics, and hence embedding similarity, to the word that, prior to reassignment, was 'reine' (queen). French should be the ideal parent model for French′.

The results of this experiment are shown in Table 6. We get a 4.3 BLEU improvement with an unrelated parent (i.e. French–parent and Uzbek–child), but we get a 6.7 BLEU improvement with a 'closely related' parent (i.e. French–parent and French′–child). We conclude that the choice of parent model can have a strong impact on transfer models, and choosing better parents for our low-resource languages (if data for such parents can be obtained) could improve the final results.

5.3 Ablation Analysis

In all the above experiments, only the target input and output embeddings are fixed during training. In this section we analyze what happens when different parts of the model are fixed, in order to determine the scenario that yields optimal performance. Figure 2 shows a diagram of the components of a sequence-to-sequence model. Table 7 shows the effects of allowing various components of the child NMT model to be trained. We find that the optimal setting for transferring from French–English to Uzbek–English in terms of BLEU performance is to allow all of the components of the child model to be trained except for the input and output target embeddings.

Even though we use this setting for our main experiments, the optimum setting is likely to be language- and corpus-dependent. For Turkish, experiments show that freezing attention parameters as well gives slightly better results. For parent-child models with closely related languages we expect freezing, or strongly regularizing, more components of the model to give better results.

5.4 Learning Curve

In Figure 3 we plot learning curves for both a transfer and a non-transfer model on training and development sets. We see that the final training set perplexities for both the transfer and non-transfer model are very similar, but the development set perplexity for the transfer model is much better.

The fact that the two models start from and converge to very different points, yet have similar training set performances, indicates that our architecture

Source Embeddings	Source RNN	Target RNN	Attention	Target Input Embeddings	Target Output Embeddings	Dev BLEU ↑	Dev PPL ↓
🔒	🔒	🔒	🔒	🔒	🔒	0.0	112.6
🔓	🔒	🔒	🔒	🔒	🔒	7.7	24.7
🔓	🔓	🔒	🔒	🔒	🔒	11.8	17.0
🔓	🔓	🔓	🔒	🔒	🔒	14.2	14.5
🔓	🔓	🔓	🔓	🔒	🔒	**15.0**	13.9
🔓	🔓	🔓	🔓	🔓	🔒	14.7	**13.8**
🔓	🔓	🔓	🔓	🔓	🔓	13.7	14.4

Table 7: Starting with the parent French–English model (BLEU =24.4, PPL=6.2), we randomly assign Uzbek word types to French word embeddings, freeze various parameters of the neural network model (🔒), and allow Uzbek–English (child model) training to modify other parts (🔓). The table shows how Uzbek–English BLEU and perplexity vary as we allow more parameters to be re-trained.

and training algorithm are able to reach a good minimum of the training objective regardless of the initialization. However, the training objective seems to have a large basin of models with similar performance and not all of them generalize well to the development set. The transfer model, starting with and staying close to a point known to perform well on a related task, is guided to a final point in the weight space that generalizes to the development set much better.

5.5 Dictionary Initialization

Using the transfer method, we always initialize input language embeddings for the child model with randomly-assigned embeddings from the parent (which has a different input language). A smarter method might be to initialize child embeddings with similar parent embeddings, where similarity is measured by word-to-word t-table probabilities. To get these probabilities we compose Uzbek–English and English–French t-tables obtained from the Berkeley Aligner (Liang et al., 2006). We see from Figure 4 that this dictionary-based assignment results in faster improvement in the early part of the training. However the final performance is similar to our standard model, indicating that the training is able to untangle the dictionary permutation introduced by randomly-assigned embeddings.

5.6 Different Parent Models

In the above experiments, we use a parent model trained on a large French–English corpus. One might hypothesize that our gains come from exploiting the English half of the corpus as an additional language model resource. Therefore, we explore transfer learning for the child model with parent models that only use the English side of the French–English corpus. We consider the following parent models in our ablative transfer learning scenarios:

- A true translation model (French–English Parent)
- A word-for-word English copying model (English–English Parent)
- A model that unpermutes scrambled English (EngPerm–English Parent)
- (The parameters of) an RNN language model (LM Xfer)

Transfer Model	BLEU ↑	PPL ↓
None	10.7	22.4
French–English Parent	**14.4**	**14.3**
English–English Parent	5.3	55.8
EngPerm–English Parent	10.8	20.4
LM Xfer	12.9	16.3

Table 8: Transfer for Uzbek–English NMT with parent models trained only on English data. The English–English parent learns to copy English sentences, and the EngPerm–English learns to un-permute scrambled English sentences. The LM is a 2-layer LSTM RNN language model.

The results, in Table 8, show that transfer learning does not simply import an English language model, but makes use of translation parameters learned from the parent's large bilingual text.

6 Conclusion

Overall, our transfer method improves NMT scores on low-resource languages by a large margin and allows our transfer NMT system to come close to the performance of a very strong SBMT system, even exceeding its performance on Hausa–English. In addition, we consistently and significantly improve state-of-the-art SBMT systems on low-resource languages when the transfer NMT system is used for rescoring. Our experiments suggest that there is still room for improvement in selecting parent languages that are more similar to child languages, provided data for such parents can be found.

Acknowledgments

This work was supported by ARL/ARO (W911NF-10-1-0533), DARPA (HR0011-15-C-0115), and the Scientific and Technological Research Council of Turkey (TÜBİTAK) (grants 114E628 and 215E201).

References

D. Bahdanau, K. Cho, and Y. Bengio. 2014. Neural machine translation by jointly learning to align and translate. In *Proc. ICLR*.

P. Baltescu and P. Blunsom. 2015. Pragmatic neural language modelling in machine translation. In *Proc. HLT-NAACL*.

Y. Bengio. 2012. Deep learning of representations for unsupervised and transfer learning. *JMLR*, 27.

O. Bojar, R. Chatterjee, C. Federmann, B. Haddow, M. Huck, C. Hokamp, P. Koehn, V. Logacheva, C. Monz, M. Negri, M. Post, C. Scarton, L. Specia, and M. Turchi. 2015. Findings of the 2015 workshop on statistical machine translation. In *Proc. WMT*.

M. A. Castaño and F. Casacuberta. 1997. A connectionist approach to machine translation. In *Proc. Eurospeech*.

D. Chiang, K. Knight, and W. Wang. 2009. 11,001 new features for statistical machine translation. In *Proc. HLT-NAACL*.

D. C. Cireşan, U. Meier, and J. Schmidhuber. 2012. Transfer learning for Latin and Chinese characters with deep neural networks. In *Proc. IJCNN*.

D. Dong, H. Wu, W. He, D. Yu, and H. Wang. 2015. Multi-task learning for multiple language translation. In *Proc. ACL-IJCNLP*.

O. Firat, K. Cho, and Y. Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *Proc. NAACL-HLT*.

M. Galley, M. Hopkins, K. Knight, and D. Marcu. 2004. What's in a translation rule? In *Proc. HLT-NAACL*.

M. Galley, J. Graehl, K. Knight, D. Marcu, S. DeNeefe, W. Wang, and I. Thayer. 2006. Scalable inference and training of context-rich syntactic translation models. In *Proc. ACL-COLING*.

F. A. Gers, J. Schmidhuber, and F. Cummins. 2000. Learning to forget: Continual prediction with LSTM. *Neural computation*, 12(10).

S. Hochreiter and J. Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8).

R. Józefowicz, W. Zaremba, and I. Sutskever. 2015. An empirical exploration of recurrent network architectures. In *Proc. ICML*.

P. Liang, B. Taskar, and D. Klein. 2006. Alignment by agreement. In *Proc. HLT-NAACL*.

M. Luong, H. Pham, and C. Manning. 2015a. Effective approaches to attention-based neural machine translation. In *Proc. EMNLP*.

T. Luong, I. Sutskever, Q. Le, O. Vinyals, and W. Zaremba. 2015b. Addressing the rare word problem in neural machine translation. In *Proc. ACL*.

A. Mnih and Y. W. Teh. 2012. A fast and simple algorithm for training neural probabilistic language models. In *Proc. ICML*.

R. Neco and M. Forcada. 1997. Asynchronous translations with recurrent neural nets. In *Proc. International Conference on Neural Networks*.

S. J. Pan and Q. Yang. 2010. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10).

I. Sutskever, O. Vinyals, and Q. V. Le. 2014. Sequence to sequence learning with neural networks. In *Proc. NIPS*.

L. Torrey and J. Shavlik. 2009. Transfer learning. In E. Soria, J. Martin, R. Magdalena, M. Martinez, and A. Serrano, editors, *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*. IGI Global.

A. Vaswani, Y. Zhao, V. Fossum, and D. Chiang. 2013. Decoding with large-scale neural language models improves translation. In *Proc. EMNLP*.

D. Wang and T. Fang Zheng. 2015. Transfer learning for speech and language processing. *CoRR*, abs/1511.06066.

P. J. Werbos. 1990. Backpropagation through time: what it does and how to do it. *Proc. IEEE*, 78(10).

W. Williams, N. Prasad, D. Mrva, T. Ash, and T. Robinson. 2015. Scaling recurrent neural network language models. In *Proc. ICASSP*.

W. Zaremba, I. Sutskever, and O. Vinyals. 2014. Recurrent neural network regularization. *CoRR*, abs/1409.2329.