Regularization techniques for fine-tuning in neural machine translation

Antonio Valerio Miceli Barone **Barry Haddow**
Ulrich Germann **Rico Sennrich**
School of Informatics, The University of Edinburgh
{amiceli, bhaddow, ugermann}@inf.ed.ac.uk
rico.sennrich@ed.ac.uk

Abstract

We investigate techniques for supervised domain adaptation for neural machine translation where an existing model trained on a large out-of-domain dataset is adapted to a small in-domain dataset.

In this scenario, overfitting is a major challenge. We investigate a number of techniques to reduce overfitting and improve transfer learning, including regularization techniques such as dropout and L2-regularization towards an out-of-domain prior. In addition, we introduce *tuneout*, a novel regularization technique inspired by dropout. We apply these techniques, alone and in combination, to neural machine translation, obtaining improvements on IWSLT datasets for English→German and English→Russian. We also investigate the amounts of in-domain training data needed for domain adaptation in NMT, and find a logarithmic relationship between the amount of training data and gain in BLEU score.

1 Introduction

Neural machine translation (Bahdanau et al., 2015; Sutskever et al., 2014) has established itself as the new state of the art at recent shared translation tasks (Bojar et al., 2016; Cettolo et al., 2016). In order to achieve good generalization accuracy, neural machine translation, like most other large machine learning systems, requires large amounts of training examples sampled from a distribution as close as possible to the distribution of the inputs seen during execution. However, in many applications, only a small amount of parallel text is available for the specific application domain, and it is therefore desirable to leverage larger out-domain datasets.

Owing to the incremental nature of stochastic gradient-based training algorithms, a simple yet effective approach to transfer learning for neural networks is *fine-tuning* (Hinton and Salakhutdinov, 2006; Mesnil et al., 2012; Yosinski et al., 2014): to continue training an existing model which was trained on out-of-domain data with in-domain training data. This strategy was also found to be very effective for neural machine translation (Luong and Manning, 2015; Sennrich et al., 2016b).

Since the amount of in-domain data is typically small, overfitting is a concern. A common solution is early stopping on a small held-out in-domain validation dataset, but this reduces the amount of in-domain data available for training.

In this paper, we show that we can make fine-tuning strategies for neural machine translation more robust by using several regularization techniques. We consider fine-tuning with varying amounts of in-domain training data, showing that improvements are logarithmic in the amount of in-domain data.

We investigate techniques where domain adaptation starts from a pre-trained out-domain model, and only needs to process the in-domain corpus. Since we do not need to process the large out-domain corpus during adaptation, this is suitable for scenarios where adaptation must be performed quickly or where the original out-domain corpus is not available. Other works consider techniques that jointly train on the out-domain and in-domain corpora, distinguishing them using specific input features (Daume III, 2007; Finkel and Manning, 2009; Wuebker et al., 2015). These techniques are largely orthogonal to

Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 1489–1494
Copenhagen, Denmark, September 7–11, 2017.

ours[1] and can be used in combination. In fact, Chu et al. (2017) successfully apply fine-tuning in combination with joint training.

2 Regularization Techniques for Transfer Learning

Overfitting to the small amount of in-domain training data that may be available is a major challenge in transfer learning for domain adaptation. We investigate the effect of different regularization techniques to reduce overfitting, and improve the quality of transfer learning.

2.1 Dropout

The first variant that we consider is fine-tuning with dropout. Dropout (Srivastava et al., 2014) is a stochastic regularization technique for neural networks. In particular, we consider "Bayesian" dropout for recurrent neural networks (Gal and Ghahramani, 2016).

In this technique, during training, the columns of the weight matrices of the neural network are randomly set to zero, independently for each example and each epoch, but with the caveat that when the same weight matrix appears multiple times in the unrolled computational graph of a given example, the same columns are zeroed.

For an arbitrary layer that takes an input vector h and computes the pre-activation vector v (ignoring the bias parameter),

$$v_{i,j} = W \cdot M_{W,i,j} \cdot h_{i,j} \qquad (1)$$

where $M_{W,i,j} = \frac{1}{p}\text{diag}(\text{Bernoulli}^{\otimes n}(p))$ is the dropout mask for matrix W and training example i seen in epoch j. This mask is a diagonal matrix whose entries are drawn from independent Bernoulli random variables with probability p and then scaled by $1/p$. Gal and Ghahramani (2016) have shown that this corresponds to approximate variational Bayesian inference over the weight matrices considered as model-wise random variables, where the individual weights have a Gaussian prior with zero mean and small diagonal covariance. During execution we simply set the dropout masks to identity matrices, as in the standard approximation scheme.

Since dropout is not a specific transfer learning technique per se, we can apply it during fine-tuning, irrespective of whether or not the original out-of-domain model was also trained with dropout.

2.2 MAP-L2

L2-norm regularization is widely used for machine learning and statistical models. For linear models, it corresponds to imposing a diagonal Gaussian prior with zero mean on the weights. Chelba and Acero (2006) extended this technique to transfer learning by penalizing the weights of the in-domain model by their L2-distance from the weights of the previously trained out-of-domain model.

For each parameter matrix W, the penalty term is

$$L_W = \lambda \cdot \left\| W - \hat{W} \right\|_2^2 \qquad (2)$$

where W is the in-domain parameter matrix to be learned and $\hat{W}$ is the corresponding fixed out-of-domain parameter matrix. Bias parameters may be regularized as well. For linear models, this corresponds to maximum a posteriori inference w.r.t. a diagonal Gaussian prior with mean equal to the out-of-domain parameters and $1/\lambda$ variance.

To our knowledge this method has not been applied to neural networks, except for a recent work by Kirkpatrick et al. (2017) which investigates a variant of it for *continual learning* (learning a new task while preserving performance on previously learned task) rather than domain adaptation. In this work we investigate L2-distance from out-of-domain penalization (MAP-L2) as a domain adaptation technique for neural machine translation.

2.3 Tuneout

We also propose a novel transfer learning technique which we call *tuneout*. Like Bayesian dropout, we randomly drop columns of the weight matrices during training, but instead of setting them to zero, we set them to the corresponding columns of the out-of-domain parameter matrices.

This can be alternatively seen as learning matrices of parameter differences between in-domain and out-of-domain models with standard dropout, starting from a zero initialization at the beginning of fine-tuning. Therefore, equation 2 becomes

$$v_{i,j} = (\hat{W} + \Delta W \cdot M_{\Delta W,i,j}) \cdot h_{i,j} \qquad (3)$$

where $\hat{W}$ is the fixed out-of-domain parameter matrix and ΔW is the parameter difference matrix to be learned and $M_{\Delta W,i,j}$ is a Bayesian dropout mask.

[1] although in the special case of linear models, they are related to MAP-L2 fine-tuning.

3 Evaluation

We evaluate transfer learning on test sets from the IWSLT shared translation task (Cettolo et al., 2012).

3.1 Data and Methods

Test sets consist of transcripts of TED talks and their translations; small amounts of in-domain training data are also provided. For English-to-German we use IWSLT 2015 training data, while for English-to-Russian we use IWSLT 2014 training data. For the out-of-domain systems, we use training data from the WMT shared translation task,[2] which is considered permissible for IWSLT tasks, including back-translations of monolingual training data (Sennrich et al., 2016b), i.e., automatic translations of data available only in target language "back" into the source language.[3].

We train out-of-domain systems following tools and hyperparameters reported by Sennrich et al. (2016a), using Nematus (Sennrich et al., 2017) as the neural machine translation toolkit. We differ from their setup only in that we use Adam (Kingma and Ba, 2015) for optimization. Our baseline fine-tuning models use the same hyperparameters, except that the learning rate is 4 times smaller and the validation frequency for early stopping 4 times higher. Early stopping serves an important function as the only form of regularization in the baseline fine-tuning model. We also use this configuration for the in-domain only baselines.

After some exploratory experiments for English-to-German, we set dropout retention probabilities to 0.9 for word-dropout and 0.8 for all the other parameter matrices. Tuneout retention probabilities are set to 0.6 (word-dropout) and 0.2 (other parameters). For MAP-L2 regularization, we found that a penalty of 10^{-3} per mini-batch performs best. For English-to-Russian, retention probabilities of 0.95 (word-dropout) 0.89 (other parameters) for both dropout and tuneout performed best.

The out-of-domain training data consists of about $7.92M$ sentence pairs for English-to-German and $4.06M$ sentence pairs for English-to-Russian. In-domain training data is about $206k$ sentence pairs for English-to-German and $181k$ sentence pairs for English-to-Russian. Training data is tokenized, truecased and segmented into subword units using byte-pair encoding (BPE) (Sennrich et al., 2016c).

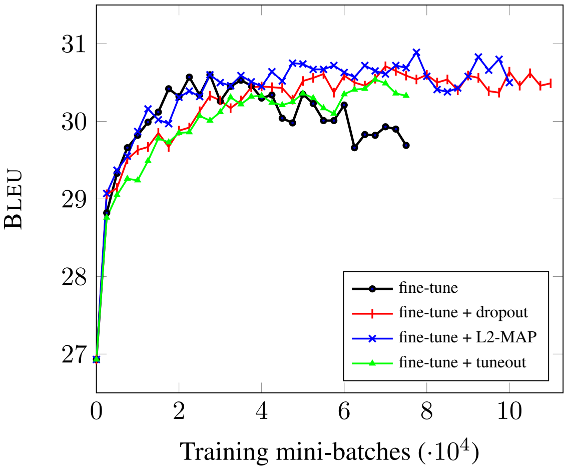

Figure 1: English→German validation BLEU over training mini-batches.

For replicability and ease of adoption, we include our implementation of dropout and MAP-L2 in the master branch of Nematus. Tuneout regularization is available in a separate code branch of Nematus.[4]

3.2 Results

We report the translation quality in terms of NIST-BLEU scores of our models in Table 1 for English-to-German and Table 2 for English-to-Russian. Statistical significance on the concatenated test sets scores is determined via bootstrap resampling (Koehn, 2004).

Dropout and MAP-L2 improve translation quality when fine-tuning both separately and in combination. When the two methods are used in combination, the improvements are significant at 5% for both language pairs, while in isolation dropout is non-significant and MAP-L2 is only significant for English-to-Russian. Tuneout does not yield improvements for English-to-German, in fact it is significantly worse, but yields a small, non-significant improvement for English-to-Russian.

In order to obtain a better picture of the training dynamics, we plot training curves[5] for several of our English-to-German models in Figure 1.

[2] http://www.statmt.org/wmt16/

[3] http://data.statmt.org/rsennrich/wmt16_backtranslations/

[4] https://github.com/EdinburghNLP/nematus/tree/tuneout-branch

[5] These BLEU scores are computed using Moses `multi-bleu.perl` which gives slightly different results than NIST `mteval-v13a.pl` that is used for Table 1.

Table 1: English-to-German translation BLEU scores

	valid	test			
System	tst2010	tst2011	tst2012	tst2013	avg
Out-of-domain only	27.19	29.65	25.78	27.85	27.76
In-domain only	25.95	27.84	23.68	25.83	25.78
Fine-tuning	30.53	32.62	28.86	32.11	31.20
Fine-tuning + dropout	30.63	33.06	28.90	32.02	31.33
Fine-tuning + MAP-L2	30.81	32.87	28.99	31.88	31.25
Fine-tuning + tuneout	30.49	32.07	28.66	31.60	30.78†
Fine-tuning + dropout + MAP-L2	30.80	**33.19**	**29.13**	**32.13**	**31.48**†

†: different from the fine-tuning baseline at 5% significance.

Table 2: English-to-Russian translation BLEU scores

	valid	test			
System	dev2010	tst2011	tst2012	tst2013	avg
Out-of-domain only	15.74	17.48	15.15	17.81	16.81
Fine-tuning	17.47	19.67	17.17	19.18	18.67
Fine-tuning + dropout	17.68	**19.96**	17.11	19.32	18.80
Fine-tuning + MAP-L2	**17.77**	19.91	17.34	19.49	18.91†
Fine-tuning + tuneout	17.51	19.72	17.27	19.35	18.78
Fine-tuning + dropout + MAP-L2	17.74	19.68	**17.83**	**19.78**	**19.10**†

†: different from the fine-tuning baseline at 5% significance.

Baseline fine-tuning starts to noticeably overfit between the second and third epoch (1 epoch ≈ 10^4 mini-batches), while dropout, MAP-L2 and tuneout seem to converge without displaying noticeable overfitting.

In our experiments, all forms of regularization, including early stopping, have shown to be successful at mitigating the effect of overfitting. Still, our results suggest that there is value in not relying only on early stopping:

- our results suggest that multiple regularizers outperform a single one.
- if the amount of in-domain data is very small, we may want to use all of it for fine-tuning, and not hold out any for early stopping.

To evaluate different fine-tuning streategies on varying amounts of in-domain data, we tested fine-tuning with random samples of in-domain data, ranging from 10 sentence pairs to the full data set of 206k sentence pairs. Fine-tuning with low amounts of training data is of special interest for online adaptation scenarios where a system is fed back post-edited translation.[6] Results are shown

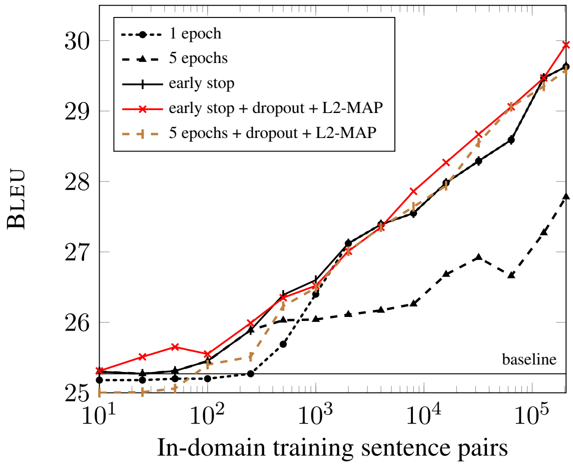

Figure 2: English→German test BLEU with fine-tuning on different in-domain data set size. Baseline trained on WMT data.

[6] We expect even bigger gains in that scenario because we would not train on a random sample, but on translations that are conceivably from the same document.

in Figure 2.

The results show an approximately logarithmic relation between the size of the in-domain training set and BLEU. We consider three baseline approaches: fine-tuning for a fixed number of epochs (1 or 5), or early stopping. All three baseline approaches have their disadvantages. Fine-tuning for 1 epoch shows underfitting on small amounts of data (less than 1,000 sentence pairs); fine-tuning for 5 epochs overfits on 500-200,000 sentence pairs. Early stopping is generally a good strategy, but it requires an in-domain held-out dataset.

On the same amount of data, regularization (dropout+MAP-L2) leads to performance that is better (or no worse) than the baseline with only early stopping. Fine-tuning with regularization is also more stable, and if we have no access to a in-domain valdiation set for early stopping, can be run for a fixed number of epochs with little or no accuracy loss.

4 Conclusion

We investigated fine-tuning for domain adaptation in neural machine translation with different amounts of in-domain training data, and strategies to avoid overfitting. We found that our baseline that relies only on early stopping has a strong performance, but fine-tuning with recurrent dropout and with MAP-L2 regularization yield additional small improvements of the order of 0.3 BLEU points for both English-to-German and English-to-Russian, while the improvements in terms of final translation accuracy of tuneout appear to be less consistent.

Furthermore, we found that regularization techniques that we considered make training more robust to overfitting, which is particularly helpful in scenarios where only small amounts of in-domain data is available, making early-stopping impractical as it relies on a sufficiently large in-domain validation set. Given the results of our experiments, we recommend using both dropout and MAP-L2 regularization for fine-tuning tasks, since they are easy to implement, efficient, and yield improvements while stabilizing training. We also present a learning curve that shows a logarithmic relationship between the amount of in-domain training data and the quality of the adapted system.

Our techniques are not specific to neural machine translation, and we propose that they could be also tried for other neural network architectures and other tasks.

Acknowledgments

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreements 644333 (TraMOOC) and 645487 (ModernMT). We also thank Booking.com for their support.

References

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio. 2015. "Neural Machine Translation by Jointly Learning to Align and Translate." *Proceedings of the International Conference on Learning Representations (ICLR)*.

Bojar, Ondřej, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. "Findings of the 2016 Conference on Machine Translation (WMT16)." *Proceedings of the First Conference on Machine Translation, Volume 2: Shared Task Papers*, 131–198. Berlin, Germany.

Cettolo, Mauro, Christian Girardi, and Marcello Federico. 2012. "WIT^3: Web Inventory of Transcribed and Translated Talks." *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT)*, 261–268. Trento, Italy.

Cettolo, Mauro, Jan Niehues, Sebastian Stüker, Luisa Bentivogli, and Marcello Federico. 2016. "Report on the 13th IWSLT Evaluation Campaign." *IWSLT 2016*. Seattle, USA.

Chelba, Ciprian and Alex Acero. 2006. "Adaptation of maximum entropy capitalizer: Little data can help a lot." *Computer Speech & Language*, 20(4):382–399.

Chu, Chenhui, Raj Dabre, and Sadao Kurohashi. 2017. "An Empirical Comparison of Simple Domain Adaptation Methods for Neural Machine Translation." *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*. Vancouver, Canada.

Daume III, Hal. 2007. "Frustratingly Easy Domain Adaptation." *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, 256–263. Prague, Czech Republic.

Finkel, Jenny Rose and Christopher D. Manning. 2009. "Hierarchical Bayesian Domain Adaptation." *Proceedings of Human Language Technologies: The*

2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, NAACL '09, 602–610. Stroudsburg, PA, USA.

Gal, Yarin and Zoubin Ghahramani. 2016. "A Theoretically Grounded Application of Dropout in Recurrent Neural Networks." *Advances in Neural Information Processing Systems 29 (NIPS)*.

Hinton, Geoffrey E and Ruslan R Salakhutdinov. 2006. "Reducing the dimensionality of data with neural networks." *Science*, 313(5786):504–507.

Kingma, Diederik P. and Jimmy Ba. 2015. "Adam: A Method for Stochastic Optimization." *The International Conference on Learning Representations*. San Diego, California, USA.

Kirkpatrick, James, Razvan Pascanu, Neil C. Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. 2017. "Overcoming catastrophic forgetting in neural networks." *Proceedings of the National Academy of Sciences*, 114(13):3521–3526.

Koehn, Philipp. 2004. "Statistical Significance Tests for Machine Translation Evaluation." *Proceedings of EMNLP 2004*, 388–395. Barcelona, Spain.

Luong, Minh-Thang and Christopher D. Manning. 2015. "Stanford Neural Machine Translation Systems for Spoken Language Domains." *Proceedings of the International Workshop on Spoken Language Translation 2015*. Da Nang, Vietnam.

Mesnil, Grégoire, Yann Dauphin, Xavier Glorot, Salah Rifai, Yoshua Bengio, Ian J Goodfellow, Erick Lavoie, Xavier Muller, Guillaume Desjardins, David Warde-Farley, et al. 2012. "Unsupervised and Transfer Learning Challenge: a Deep Learning Approach." *ICML Unsupervised and Transfer Learning*, 27:97–110.

Sennrich, Rico, Orhan Firat, Kyunghyun Cho, Alexandra Birch, Barry Haddow, Julian Hitschler, Marcin Junczys-Dowmunt, Samuel Läubli, Antonio Valerio Miceli Barone, Jozef Mokry, and Maria Nadejde. 2017. "Nematus: a Toolkit for Neural Machine Translation." *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, 65–68. Valencia, Spain.

Sennrich, Rico, Barry Haddow, and Alexandra Birch. 2016a. "Edinburgh Neural Machine Translation Systems for WMT 16." *Proceedings of the First Conference on Machine Translation*, 371–376. Berlin, Germany.

Sennrich, Rico, Barry Haddow, and Alexandra Birch. 2016b. "Improving Neural Machine Translation Models with Monolingual Data." *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 86–96. Berlin, Germany.

Sennrich, Rico, Barry Haddow, and Alexandra Birch. 2016c. "Neural Machine Translation of Rare Words with Subword Units." *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1715–1725. Berlin, Germany.

Srivastava, Nitish, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. "Dropout: a simple way to prevent neural networks from overfitting." *Journal of Machine Learning Research*, 15(1):1929–1958.

Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le. 2014. "Sequence to Sequence Learning with Neural Networks." *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014*, 3104–3112. Montreal, Quebec, Canada.

Wuebker, Joern, Spence Green, and John DeNero. 2015. "Hierarchical Incremental Adaptation for Statistical Machine Translation." *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 1059–1065. Lisbon, Portugal.

Yosinski, Jason, Jeff Clune, Yoshua Bengio, and Hod Lipson. 2014. "How transferable are features in deep neural networks?" *Advances in neural information processing systems*, 3320–3328.