Investigating backtranslation for domain adaptation in machine translation

Seth Aycock s.aycock@uva.nl
Teaching Assistant, UvA

Abstract

Low-resource machine translation (Haddow et al., 2022) is the task of training models to translate languages for which limited translated data is available. Maximising the value of the available data is therefore an important problem. One intuitive method known as backtranslation involves taking monolingual data and generating synthetic parallel data using a trained reverse translation model; this synthetic data is then used to train or fine-tune a forward translation model. In this assignment you will focus on data-driven experiments to explore the impact of backtranslation on adapting to a domain with limited data. We provide several potential ideas for you to explore focused on data selection, backtranslation extensions, and evaluation of your results.

1 Introduction

Motivation Training neural machine translation (MT) models typically involves training an attentional sequence-to-sequence model to predict a target language sentence given a source language sentence, maximising the probability of the human-translated target sentence. This type of data is *parallel* data, and is the gold standard for MT training. We note that the encoder-decoder Transformer (Vaswani et al., 2017) has become the de-facto standard for neural MT, and we will use this architecture in this project.

Let us define the problem as follows. Given a source and target sequence X_p^n and Y_p^n, drawn from parallel corpus $\{\mathbf{X}_p, \mathbf{Y}_p\}$, we train a model $M_{X>Y}$ to predict target sequence $\tilde{Y}_p^n$ by modelling the conditional probability $p(Y_p^n|X_p^n)$:

$$\begin{aligned} X_p^n &= x_1, ..., x_N \\ Y_p^n &= y_1, ..., y_L \\ \tilde{Y}_p^n &= \tilde{y}_1, ..., \tilde{y}_L \end{aligned} \tag{1}$$

During standard autoregressive decoding we predict each output token one at a time given the source X and previously generated target tokens $y_{<i}$:

$$\tilde{y}_i = \text{argmax}\ p(y_i|y_1, ...y_{i-1}, X_p^n) \tag{2}$$

We calculate the cross-entropy loss based on the predicted probabilities for each output token j in vocabulary V, $\tilde{y}_{ij}$ at position i and given the supervised target probabilities y_{ij}.

$$CE = \frac{1}{L}\sum_{i=1}^{L}\sum_{j=1}^{|V|} y_{ij} \log p(\tilde{y}_{ij}) \tag{3}$$

However, in low-resource languages or domains, high-quality parallel data is not always available; instead we may only have access to monolingual (unaligned) data in the target, source or both. We therefore want to maximise the utility we can gain from the available data by integrating monolingual data into the process of training an MT model. One very successful and intuitive method was introduced by Sennrich et al. (2016), known as **backtranslation**: using a target→source MT model we translate monolingual target-side data to generate *synthetic* source side data, which is combined in some way with the available parallel data to train a source→target model.

To briefly define this, we now have an in-domain monolingual dataset in the target, $\mathbf{Y}_m$ (and/or an unaligned monolingual source dataset $\mathbf{X}_m$). We use Eq.s 1-3 and our parallel data to train a translation model $M_{Y>X}$ from language Y to X. We then use $M_{Y>X}$ to generate a synthetic source-side corpus $\tilde{\mathbf{X}}_m$ from $\mathbf{Y}_m$. Finally we can either combine $\{\mathbf{X}_p, \mathbf{Y}_p, \tilde{\mathbf{X}}_m, \mathbf{Y}_m\}$ for training a model $M_{X>Y}$, or use $\{\tilde{\mathbf{X}}_m, \mathbf{Y}_m\}$ and a parallel in-domain dataset $\{\mathbf{X}_{ft}, \mathbf{Y}_{ft}\}$ for fine-tuning a trained $M_{X>Y}$ model to a domain.

The key benefit of backtranslation (BT) is that it permits incorporation of monolingual data without modifying the standard NMT training objective. However, there are several important decisions in the BT pipeline that can have a significant impact on the final MT performance, such as the reverse translation model and decoding strategy, extensions like iterative BT (Hoang et al., 2018), and selection of target or synthetic source data.

Assignment In this project you will implement backtranslation for domain adaptation in a simulated low-resource setting, and investigate the effect of at least two modelling choices. For your experiments and implementations, you are recommended to use the `fairseq`[1] framework. You will be provided with raw and preprocessed monolingual and parallel src and tgt data, parallel dev and test sets for fine-tuning, plus various trained models in both directions, as well as skeleton scripts for data preprocessing. You will be given some template code to get started on training your models and some suggestions for potential modifications.

This assignment can be divided into these sub-tasks:

- Study past work introducing and investigating aspects of backtranslation, choose a part of the pipeline to focus your experiments, and justify your choice of experiments.
- Implement a standard backtranslation pipeline and compare against baselines.
- Compare results for BT with one data selection technique and one extension of BT against your initial results.
- Conduct a detailed error analysis of your results and compare findings with related work.

Although there are many methods you could implement and hyperparameters you could modify, we suggest the following directions. **Choose ONE** direction per category. Remember to always justify your choice of experiments and modifications.

- Data selection: **How can you best select, filter, and/or order target-side data or source-side synthetic data for downstream performance?**

[1] `https://github.com/facebookresearch/fairseq`

  - Consider **target-side selection** (pre-BT) (Moore and Lewis, 2010); **synthetic source selection** (post-BT) (Tsvetkov et al., 2016); or ordering the backtranslated data (**curriculum learning**) (Zhang et al., 2019).

- Optimising/extending backtranslation: **How can you improve or extend the backtranslation model or its hyperparameters, and what can you compare it to?**
  - Consider varying the BT **model size/quality** (Burlot and Yvon, 2018); varying **decoding** strategies (Imamura et al., 2018); or implementing **iterative backtranslation** (Hoang et al., 2018).
  - In implementing your modifications, ensure you run ablation tests to understand the impact of each change individually.

Research Questions In your work you are asked to address the following general research questions:

1. Does BT perform better for low-resource domain adaptation than fine-tuning, are they complementary, and if not, why?
2. How does data selection improve backtranslation and MT performance?
3. How can backtranslation be optimised or extended for this low-resource scenario?
4. A research question of your own, related to one of the other research questions. (You will have the chance to discuss your approach with the TA during the project).

For all these questions, you should analyse and evaluate how exactly BT can help and where BT fails, and compare results to the literature:

- Go beyond BLEU scores: this doesn't correlate well with human judgments (Mathur et al., 2020), so consider some more appropriate measures that balance fluency and adequacy; optionally compare other test sets to explore how fine-tuning degrades performance on other domains.
- Look at your outputs: What can you learn about your experimental modifications through a more detailed error analysis? This can be a quantitative analysis (e.g. of out-of-vocabulary word handling, output diversity) (Neubig et al., 2019) and/or qualitative analysis of a few test-set examples.

Deliverables

1. **Work plan (Due 18/4/24 23:59) - Mandatory**
   (a) Outline of proposed experiments and planned splits of workload between group.
2. **Mid-term report (Due 26/4/24 23:59) - 30%, 3-4 pages (excl. references)** - Detailing progress up to this point. A suggested (not mandatory) page distribution is as follows:
   (a) **Introduction**: Briefly describe the problem, motivate your research and specific research questions. (0.25pg)
   (b) **Related Work**: Summarise some research papers you've read so far, with a deeper look at the most relevant ones. (0.5pg)
   (c) **Research Questions**: List your central research questions that you *plan* to investigate. These might change or develop later on, but it is good to have ideas of what you want to test. (0.5pg)

(d) **Methodology**: Your methodology for planned/completed experiments. (0.5-1pg)

(e) **Experiments (& Results)**: You may not have any results yet, which you can mention in the report. You should still write up your experimental design and what results you want to collect. (0.5pg)

(f) **Next Steps**: What are you planning to do in the final 4 weeks of the project? (0.25pg)

3. **Final report (Due 31/5/24 23:59) - 70%, Max 8 pages (excl. references)**, using the ACL template.[2]. A suggested (not mandatory) distribution is as follows:

   - **Abstract**: Provide a (very) concise overview of your approach, and highlight your key findings. (0.25pg).
   - **Introduction**: Introduce the reader to your research area, provide research questions and an explicitly motivated problem statement, summarise your contributions, and highlight the relevance of your research. (1-1.5pg).
   - **Related Work**: Summarise research papers relevant for your work. Highlight the key differences between your work and related work. (1pg).
   - **Method**: Describe your approach, and highlight important research decisions you made along the way (e.g. selection metric choice, experimental focus). (1-1.5pg).
   - **Experiments**: Detail the precise experimental setup used and the numerical results your models achieved. Include results and baselines from related work. Make a comparison. (1-1.5pg).
   - **Results & Discussion**: Discuss your results in an honest and self-critical manner. (1-1.5pg).
   - **Conclusion**: Provide a future outlook, and highlight your paper's strengths and weaknesses. (0.5pg).
   - **Contributions - Mandatory**: Not included in 8 page limit. Describe individual members' contributions at each step of the project.

4. Jupyter Notebook (**Optional but recommended**): containing your BT pipeline and your implementation of a) a data selection method (can be a copy of a script or of the modified fairseq code) and b) experiments on a BT optimisation. We recommend training your models on GPUs using Colab.

2 Suggested Schedule

To stay on track, we recommend adhering to the following schedule.

2.1 Week 1

Important: Form your group as quickly as you can. If you do not have a group at the end of the first week let your TA know immediately.

- Reading:
  - Read introductory material on MT such as Philip Koehn's NMT guide (both the 2020 book and course materials from his MT course online) and other online resources/blog posts.

[2] `https://github.com/acl-org/acl-style-files`

  - Start reading the core backtranslation literature: Sennrich et al. (2016), Edunov et al. (2018), Burlot and Yvon (2018).

- Coding: For this project you can use any framework you prefer. However you will be provided models trained with `fairseq` Ott et al. (2019). Goals for the first week:
  - Familiarise yourself with the framework of your choice.
  - Understand the MT pipeline (data preprocessing, training, decoding, postprocessing), make sure you can train and evaluate a toy model.
  - Setup your environment (Colab (recommended) or GPU cluster environment if you have access).
  - Make sure you can both generate outputs with the trained models and can fine-tune these models with the FT dataset.

- Writing:
  - **Submit Work plan - 18/4 23:59**: Provide your preliminary work planning and expected workload distribution.

2.2 Week 2

Important: If you do not have a group at this point and you faily to notify your TA, you must complete the course on your own.

- Reading:
  - Continue reading the core literature; optionally read some investigations of BT including Poncelas et al. (2018); Fadaee and Monz (2018)
  - Begin reading more widely for the related work section.

- Coding:
  - Fine-tune models with with fine-tuning data. Compare initial results with prior work (dataset sizes and languages will differ but look at general trends).
  - Begin experimenting with different BT optimisations and versions.

- Writing:
  - Begin drafting a plan for your paper including all the required sections, study the rubric, and begin notes for related work based on your reading.

2.3 Week 3

- Reading:
  - Read more advanced literature e.g. Kumari et al. (2021), Caswell et al. (2019).

- Coding:
  - Complete baseline experiments with standard setups.
  - Implement backtranslation pipeline, and fine-tune models with backtranslated data.
  - Consider implementing a data selection method and an optimisation to the BT pipeline. Verify that your implementations work as expected.

- Writing:
  - Draft the *Method* section, explaining every decision you made for your implementation.
  - Write the *Related Work* and *Introduction* sections.
  - Submit Mid-term report including your planned experiments and progress so far.

2.4 Week 4

- Reading:
  - Research more advanced BT variants such as Iterative BT (Hoang et al., 2018) or data selection techniques like curriculum learning (Zhang et al., 2019), focusing on work related to your chosen experiments.
- Coding:
  - Aim to finalise implementations of data selection method and BT optimisation.
  - Run your planned experiments with the provided models, and tune parameters.
- Writing:
  - Write the *Method* section. Explain every decision you made for a) your data selection pipeline, and b) your chosen BT optimisations. Provide comparisons against standard fine-tuning (or other domain adaptation baselines).
  - **Submit mid-term report - 26/4 23:59**

2.5 Week 5

- Reading:
  - (Optional): 1 advanced/SOTA papers of your choice to include in related work.
- Coding:
  - Continue to run your planned experiments for your modifications of the pipeline.
  - Begin detailed evaluation and error analysis of your models' outputs.
- Writing:
  - Focus on the *Experiments* section. Describe all the details of your implemented data selection and BT optimisations.
  - Begin to compare your initial results with your own baselines and results from the literature.

2.6 Week 6

- Reading:
  - (Optional): 1 advanced/SOTA papers of your choice to include in related work.
- Coding:
  - Consider implementing a more complex implementation (such as iterative BT or dynamic curriculum learning), or expand the set of experiments and ablation tests for your chosen point of interest.

- Evaluate and perform a statistical and/or qualtitative error analysis of your models and results so far (e.g. using `compare-mt` tool).

- Writing:

  - Write *Analysis* section, making sure to include the necessary figures and examples to answer your RQs.

2.7 Week 7

- Reading:

  - (Optional) Feel free to explore recently published research works on this topic.

- Coding:

  - Aim to finish experiments and evaluations.
  - Clean and document the code.

- Writing:

  - Finalize *Method*, *Experiments* sections.
  - Expand related work if needed.
  - Write contributions section.
  - Write *Abstract* and *Discussion/Conclusion* sections.

2.8 Week 8

- Writing:

  - Make sure paper fits in to **8** pages, and ensure the story is coherent from beginning to end - do you answer your RQs sufficiently?
  - Make sure you include all the sections required in the rubric.

References

Burlot, F. and Yvon, F. (2018). Using Monolingual Data in Neural Machine Translation: a Systematic Study. In Bojar, O., Chatterjee, R., Federmann, C., Fishel, M., Graham, Y., Haddow, B., Huck, M., Yepes, A. J., Koehn, P., Monz, C., Negri, M., Névéol, A., Neves, M., Post, M., Specia, L., Turchi, M., and Verspoor, K., editors, *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 144–155, Brussels, Belgium. Association for Computational Linguistics.

Caswell, I., Chelba, C., and Grangier, D. (2019). Tagged Back-Translation. In Bojar, O., Chatterjee, R., Federmann, C., Fishel, M., Graham, Y., Haddow, B., Huck, M., Yepes, A. J., Koehn, P., Martins, A., Monz, C., Negri, M., Névéol, A., Neves, M., Post, M., Turchi, M., and Verspoor, K., editors, *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, pages 53–63, Florence, Italy. Association for Computational Linguistics.

Edunov, S., Ott, M., Auli, M., and Grangier, D. (2018). Understanding Back-Translation at Scale. In Riloff, E., Chiang, D., Hockenmaier, J., and Tsujii, J., editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 489–500, Brussels, Belgium. Association for Computational Linguistics.

Fadaee, M. and Monz, C. (2018). Back-Translation Sampling by Targeting Difficult Words in Neural Machine Translation. In Riloff, E., Chiang, D., Hockenmaier, J., and Tsujii, J., editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 436–446, Brussels, Belgium. Association for Computational Linguistics.

Haddow, B., Bawden, R., Miceli Barone, A. V., Helcl, J., and Birch, A. (2022). Survey of Low-Resource Machine Translation. *Computational Linguistics*, 48(3):673–732. Place: Cambridge, MA Publisher: MIT Press.

Hoang, V. C. D., Koehn, P., Haffari, G., and Cohn, T. (2018). Iterative Back-Translation for Neural Machine Translation. In Birch, A., Finch, A., Luong, T., Neubig, G., and Oda, Y., editors, *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 18–24, Melbourne, Australia. Association for Computational Linguistics.

Imamura, K., Fujita, A., and Sumita, E. (2018). Enhancement of Encoder and Attention Using Target Monolingual Corpora in Neural Machine Translation. In Birch, A., Finch, A., Luong, T., Neubig, G., and Oda, Y., editors, *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 55–63, Melbourne, Australia. Association for Computational Linguistics.

Kumari, S., Jaiswal, N., Patidar, M., Patwardhan, M., Karande, S., Agarwal, P., and Vig, L. (2021). Domain Adaptation for NMT via Filtered Iterative Back-Translation. In *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pages 263–271, Kyiv, Ukraine. Association for Computational Linguistics.

Mathur, N., Baldwin, T., and Cohn, T. (2020). Tangled up in BLEU: Reevaluating the Evaluation of Automatic Machine Translation Evaluation Metrics. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4984–4997, Online. Association for Computational Linguistics.

Moore, R. C. and Lewis, W. (2010). Intelligent Selection of Language Model Training Data. In Hajič, J., Carberry, S., Clark, S., and Nivre, J., editors, *Proceedings of the ACL 2010 Conference Short Papers*, pages 220–224, Uppsala, Sweden. Association for Computational Linguistics.

Neubig, G., Dou, Z.-Y., Hu, J., Michel, P., Pruthi, D., and Wang, X. (2019). compare-mt: A Tool for Holistic Comparison of Language Generation Systems. In Ammar, W., Louis, A., and Mostafazadeh, N., editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 35–41, Minneapolis, Minnesota. 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 Ammar, W., Louis, A., and Mostafazadeh, N., editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Poncelas, A., Shterionov, D., Way, A., Maillette de Buy Wenniger, G., and Passban, P. (2018). Investigating Backtranslation in Neural Machine Translation. In Pérez-Ortiz, J. A., Sánchez-Martínez, F., Esplà-Gomis, M., Popović, M., Rico, C., Martins, A., Van den Bogaert, J., and Forcada, M. L., editors, *Proceedings of the 21st Annual Conference of the European Association for Machine Translation*, pages 269–278, Alicante, Spain.

Sennrich, R., Haddow, B., and Birch, A. (2016). Improving Neural Machine Translation Models with Monolingual Data. In *Proceedings of the 54th Annual Meeting of the Association for Computational*

Linguistics (Volume 1: Long Papers), pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Tsvetkov, Y., Faruqui, M., Ling, W., MacWhinney, B., and Dyer, C. (2016). Learning the Curriculum with Bayesian Optimization for Task-Specific Word Representation Learning. In Erk, K. and Smith, N. A., editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 130–139, Berlin, Germany. Association for Computational Linguistics.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, , and Polosukhin, I. (2017). Attention is All you Need.

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 and Short Papers)*, pages 1903–1915, Minneapolis, Minnesota. Association for Computational Linguistics.