Improving Language Model Integration for Neural Machine Translation

Christian Herold    Yingbo Gao    Mohammad Zeineldeen    Hermann Ney
Human Language Technology and Pattern Recognition Group
Computer Science Department
RWTH Aachen University
D-52056 Aachen, Germany
{herold|ygao|zeineldeen|ney}@cs.rwth-aachen.de

Abstract

The integration of language models for neural machine translation has been extensively studied in the past. It has been shown that an external language model, trained on additional target-side monolingual data, can help improve translation quality. However, there has always been the assumption that the translation model also learns an implicit target-side language model during training, which interferes with the external language model at decoding time. Recently, some works on automatic speech recognition have demonstrated that, if the implicit language model is neutralized in decoding, further improvements can be gained when integrating an external language model. In this work, we transfer this concept to the task of machine translation and compare with the most prominent way of including additional monolingual data - namely back-translation. We find that accounting for the implicit language model significantly boosts the performance of language model fusion, although this approach is still outperformed by back-translation.

1 Introduction

Machine translation (MT) is the task of automatically translating text from one language to another. Nowadays, the dominant approach is neural machine translation (NMT), where a neural network is used to predict the probability of a sentence in the target language, given a sentence in the source language (Bahdanau et al., 2014; Vaswani et al., 2017). For this approach to be effective, a large number of bilingual training samples - consisting of sentences and their corresponding translations - is needed. This poses a challenge, especially when we want to build a system for a specific domain, where zero or only limited amounts of in-domain bilingual data are available.

In these situations, people turn towards monolingual text data, which is simply text in the source or target language and of which plenty exists for most languages and domains. Before NMT became feasible, the preferred way of incorporating additional monolingual data in the MT system was the usage of an external target-side language model (LM), which is trained on monolingual data to predict the probability of a sentence (Brown et al., 1990; Della Pietra, 1994; Zens et al., 2002).

However, with the rise of NMT, it was found that a technique called back-translation outperforms the LM incorporation by a large margin (Sennrich et al., 2016a). Back-translation is a two step process, where we first create synthetic parallel data by automatically translating target side monolingual data into the source language. Then, the final NMT system is trained on the combination of the real and synthetic parallel data. It was argued that the back-translation approach better suits the NMT framework because the NMT system implicitly learns an internal language model (ILM) as part of the training, which might interfere with an additional external LM (Sennrich et al., 2016a).

More recently, for automatic speech recognition (ASR), there have been works focusing on neutralizing this ILM before combination with an external LM and significant improvements were reported (McDermott et al., 2019; Variani et al., 2020; Meng et al., 2021; Zeyer et al., 2021; Zeineldeen et al., 2021). In this work, we adapt the methods for ILM compensation, developed for ASR, and test them for NMT. We compare against back-translation in different settings and find that ILM compensation significantly boosts the performance of LM fusion, although back-translation is still outperforming this approach for NMT. Also, applying ILM compensation on top of back-translation does not result in significant performance improvements.

2 Related Work

Several approaches to combine an LM and NMT model have been proposed in the past. Shallow
fusion (SF) is the most straightforward way, using a weighted log-linear combination of the model output probabilities (Gulcehre et al., 2015, 2017). Deep fusion denotes the concatenation of the hidden states of NMT model and LM and requires joint fine-tuning of both models (Gulcehre et al., 2015, 2017). Simple fusion is similar to shallow fusion, but the NMT model is trained using information from a pre-trained LM (Stahlberg et al., 2018).

For the task of ASR, people recently have started to remove the ILM that is implicitly learned. The biggest question there is, how to best approximate the ILM. Approaches include: (1) training an additional LM on the target side of the parallel data (McDermott et al., 2019), (2) removing/averaging encoder information (Variani et al., 2020; Meng et al., 2021; Zeyer et al., 2021) and (3) training a small sub-network while freezing all other parameters (Zeineldeen et al., 2021).

As an alternative to LM fusion, back-translation (Schwenk, 2008; Bertoldi and Federico, 2009; Sennrich et al., 2016a) has become the standard method for incorporating additional monolingual data for NMT. Some work has been done to improve this approach, including sampling (Edunov et al., 2018; Graça et al., 2019), tagging (Caswell et al., 2019) and block-BT (Popel et al., 2020). For sake of simplicity, we focus on the standard back-translation approach using beam search in this work.

Apart from using an external LM and back-translation, additional monolingual data can also be utilized by pre-training (Ramachandran et al., 2017; Zhu et al., 2019), multi-task-learning (Zhang and Zong, 2016; Domhan and Hieber, 2017) or post-editing (Junczys-Dowmunt and Grundkiewicz, 2016; Freitag et al., 2019). In principle, all these approaches can also be combined with LM fusion, potentially further improving the performance of the resulting system.

3 Internal LM Estimation

During decoding, given a source sentence \( f_1 \) and a model \( P(e_1 | f_1) \), we want to find the translation \( \hat{e}_1 \) that maximizes

\[
\hat{e}_1 = \arg\max_{e_1} \{ P(e_1 | f_1) \}.
\]

In our framework, \( P \) is the combination of three models:

\[
P(e_1 | f_1) \propto P_{MT}(e_1 | f_1) \cdot \tilde{P}_{LM}(e_1) \cdot \tilde{P}^{-\lambda_2}_{ILM}(e_1),
\]

where \( P_{MT} \), \( P_{LM} \) and \( P_{ILM} \) are the probabilities of the NMT model, external LM (trained on additional monolingual data) and ILM respectively, and \( \lambda_1, \lambda_2 \geq 0 \). Note that the ILM gets a negative weight, because we want to neutralize its impact in this model combination. If \( \lambda_2 = 0 \), we fall back to standard shallow fusion.

In principle, the ILM can be exactly calculated from the NMT model by marginalizing over all source sentences \( f_1 \). However, this summation would be intractable. Instead, different ILM approximations have been proposed in the recent past for ASR, which we will briefly recall here. For a more in-depth discussion of the different approximation methods we refer the reader to Zeineldeen et al. (2021).

**separate LM**: The ILM is approximated by training a separate LM on the target side of the parallel training data.

\[
h = 0 \quad \text{The ILM is approximated by taking the fully trained NMT model } P_{MT}(e_1 | f_1) \text{ and setting the encoder outputs } h_1 \text{ to 0.}
\]

\[
h = h_{avg} \quad \text{Instead of setting all encoder outputs } h_1 \text{ to 0, we replace the vector } h_j \text{ for each position } j \text{ with the average } h_{avg,j}, \text{ extracted over the whole parallel training data.}
\]

\[
c = c_{avg} \quad \text{Instead of } h, \text{ we replace all context vectors } c \text{ (the output of the encoder-decoder attention module) with the position-wise average over the whole parallel training data.}
\]

**mini-self-attn**: We replace the encoder-decoder attention of the fully trained NMT model with an additional self-attention module (with causal masking), which is then trained on the target side of the parallel training data while the rest of the NMT network is frozen. This is different from the separate LM approach because most of the parameters are still shared between NMT model and ILM, which might result in a better overall ILM approximation.\footnote{In their work, Zeineldeen et al. (2021) used a mini-LSTM network with the same dependencies as our mini-self-attention.}

4 Experiments

We perform experiments on four machine translation tasks, representing different data conditions.
The exact data conditions and statistics are provided in the Appendix A. For all tasks, the additional monolingual data, as well as the test sets, are in the news domain. The monolingual data comes from Newscrawl\(^2\) where we sample ca. 10M sentences for LM training and back-translation. For IWSLT En→De and IWSLT En→It, the parallel training data consists of around 200k sentence pairs and is in the scientific-talks-domain, coming from the IWSLT17 Multilingual Task (Cettolo et al., 2017). For this setting, we expect the biggest improvements from the additional monolingual data, since the parallel data is out-of-domain. For NEWS En→De, the parallel training data (around 300k sentence pairs) is in the news domain, coming from the News Commentary V14 corpus.\(^3\) Finally, WMT14 En→De is a standard NMT benchmark used by Vaswani et al. (2017) where the parallel training data consists of around 3.9M sentence pairs and is of mixed domain.

We tokenize the data using byte-pair-encoding (Sennrich et al., 2016b; Kudo, 2018) with 15k joint merge operations (40k for WMT14). The models are implemented using the fairseq toolkit (Ott et al., 2019) following the transformer base architecture (Vaswani et al., 2017). The details of the training setups can be found in Appendix A. All systems are trained until the validation perplexity no longer improves and the best checkpoint is selected using validation perplexity as well. We use beam-search with beam-size 12 and utilize SacreBLEU (Post, 2018) to calculate BLEU (Papineni et al., 2002) and TER (Snover et al., 2006). We report BLEU and TER since we are most familiar with these metrics and to be comparable with previous works. However, we acknowledge that these metrics might have some biases and in future work it might be worth utilizing additional metrics like COMET (Rei et al., 2020) and BLEURT (Sellam et al., 2020). Additionally, in future work we should separate our test sets for original source and target text to better understand the effect of translationese in both training and test data, as this might very much influence the improvements we see, especially in the case of back-translation (Freitag et al., 2020).

### 4.1 Comparison of ILM Approximations

We start by analyzing the ILM neutralization approaches on the IWSLT En→De task and then verify the results on the other tasks.

We implement and re-train (if applicable) all the different ILM approximation methods discussed in Section 3. The resulting perplexities on the validation set are listed in Table 1. The variants separate LM and mini-self-attention have been trained directly using the language model objective, so it is no surprise that they exhibit a much lower perplexity than the other approaches. However, it can be argued that a lower perplexity of the ILM does not necessarily correspond to a better approximation

---

\(^2\)https://data.statmt.org/news-crawl/

\(^3\)https://data.statmt.org/news-commentary/v14/
of the implicit language model.

In order to effectively use the external LM and the ILM during decoding, we need to optimize the weights $\lambda_1$ and $\lambda_2$ (see Section 3). We do this via a grid search over the validation set by optimizing for the highest BLEU score. The resulting grid for the $\text{mini-self-attention}$ ILM variant on the IWSLT En-De task is shown in Figure 1.

The NMT system by itself has a BLEU\[^{10M}\] score of 21.2. By log-linear combination with just the external LM ($\lambda_2 = 0$, vanilla shallow fusion) we can gain around 1% absolute improvement on the validation set with the best choice of $\lambda_1 = 0.15$. By including the ILM with a negative weight, we can get further improvements, up to a final score of 23.8 BLEU\[^{10M}\]. 4 Interestingly, the best performance is reached when $\lambda_1 \approx \lambda_2$ and with the ILM neutralization, the external LM can be assigned a much bigger weight compared to the case $\lambda_2 = 0$. We find that for all ILM approximation variants, the optimal weights are similar, and that the TER scores on the validation set follow an almost identical pattern. The final performance of each variant on the test set is shown in Table 2.

We want to point out, that the improvements we see on the validation set transfer nicely to the test set with the same tuned weights $\lambda_1$ and $\lambda_2$. This is because, in our experiments, the validation and test sets are of the exact same domain. In some additional experiments we found that the optimal values for these weights are indeed domain specific and have to be re-tuned if the system were to be optimized for a different domain. All ILM approximation variants lead to a significant performance improvement over simple shallow fusion. Out of all ILM approximations, the $\text{mini-self-attention}$ approach performs best, which is the same observation that Zeineldeen et al. (2021) made for ASR.

4.2 Comparison to Back-Translation

For the back-translation experiments, we train NMT systems on the same parallel training data in the reverse direction and then translate a total of 10M sentences from the monolingual target data (the same data used for training the external LM). Afterwards, the final systems are trained on the combination of real and synthetic data. The final results for all four MT tasks are shown in Table 3.

We observe the same trend for all four MT tasks. In general, the improvements from the additional monolingual data are getting smaller, when the amount of parallel training data increases. In almost all cases, shallow fusion gives a small improvement over just using the NMT system. ILM neutralization again improves consistently over simple shallow fusion, with the $\text{mini-self-attention}$ approximation variant always performing the best. Back-translation out-performs language model integration on all four tasks, although the gap is getting smaller the more parallel training data is available.

We also combine back-translation with the best ILM approximation approach ($\text{mini-self-attention}$). This does not further increase translation quality, with the exception of the WMT14 task, where we see a small improvement. In general, the ILM approach performs the closest to back-translation on the WMT14 task, so it might be worthwhile to apply this concept to an even bigger MT task.

5 Conclusion

We re-visit the method of language model integration for neural machine translation. We implement and experiment with a new approach of neutraliz-

---

Table 3: Comparison of LM fusion and back-translation on the four MT tasks. BLEU and TER are reported in percentage. External baselines are from 1 Kim et al. (2019) and 2 Vaswani et al. (2017).
ing the implicit language model, which has already shown promising result for the task of automatic speech recognition. We find that ILM neutralization significantly improves the translation quality compared to standard shallow fusion. However, back-translation as an alternative way to incorporate additional monolingual data, still outperforms the approaches using an external language model. Therefore, for future work we will focus on scenarios where back-translation can not be applied effectively, e.g. when the quality of the initial NMT system is too bad to create helpful synthetic data.

Acknowledgements

This work was partially supported by the project HYKIST funded by the German Federal Ministry of Health on the basis of a decision of the German Federal Parliament (Bundestag) under funding ID ZMVII-2520DAT04A, and by NeuroSys which, as part of the initiative “Clusters4Future”, is funded by the Federal Ministry of Education and Research BMBF (03ZU1106DA).

Limitations

The approach of language model integration for neural machine translation is analyzed and compared to the de-facto standard method of back-translation. Due to constrained resources, this work has several limitations. We focus on translation of text in a single domain, namely news-articles. Different domains might exhibit different behaviour. For the back-translation experiments, we use beam search to create the synthetic data, other methods like sampling were not considered. When combining the synthetic and real parallel data, there are additional methods like tagging and block-wise batching, which we did not utilize in this work. Finally, we compare against the most commonly used LM fusion approach, i.e. shallow fusion. There exist other LM fusion techniques which might exhibit different behaviour when used in combination with ILM neutralization.

References


A Appendix

All validation and test sets are from the WMT news translation tasks (Farhad et al., 2021). The validation/test sets are WMT newstest2015/newstest2018 for IWSLT En→De and NEWS En→De, newssyscomb2009/newstest2009 for IWSLT En→It and newstest2013/newstest2014 for WMT14 En→De. Data statistics can be found in Table 4.

<table>
<thead>
<tr>
<th>task</th>
<th>dataset</th>
<th>domain</th>
<th># sent.</th>
</tr>
</thead>
<tbody>
<tr>
<td>IWSLT</td>
<td>train</td>
<td>scientific-talks</td>
<td>210k</td>
</tr>
<tr>
<td>En→De</td>
<td>valid</td>
<td>news</td>
<td>2.2k</td>
</tr>
<tr>
<td></td>
<td>test</td>
<td>news</td>
<td>3k</td>
</tr>
<tr>
<td></td>
<td>mono.</td>
<td>news</td>
<td>9.7M</td>
</tr>
<tr>
<td>IWSLT</td>
<td>train</td>
<td>scientific-talks</td>
<td>232k</td>
</tr>
<tr>
<td>En→It</td>
<td>valid</td>
<td>news</td>
<td>500</td>
</tr>
<tr>
<td></td>
<td>test</td>
<td>news</td>
<td>2.5k</td>
</tr>
<tr>
<td></td>
<td>mono.</td>
<td>news</td>
<td>10.0M</td>
</tr>
<tr>
<td>NEWS</td>
<td>train</td>
<td>news</td>
<td>330k</td>
</tr>
<tr>
<td>En→De</td>
<td>valid</td>
<td>news</td>
<td>2.2k</td>
</tr>
<tr>
<td></td>
<td>test</td>
<td>news</td>
<td>3k</td>
</tr>
<tr>
<td></td>
<td>mono.</td>
<td>news</td>
<td>9.7M</td>
</tr>
<tr>
<td>WMT14</td>
<td>train</td>
<td>mixed</td>
<td>3.9M</td>
</tr>
<tr>
<td>En→De</td>
<td>valid</td>
<td>news</td>
<td>3k</td>
</tr>
<tr>
<td></td>
<td>test</td>
<td>news</td>
<td>3k</td>
</tr>
<tr>
<td></td>
<td>mono.</td>
<td>news</td>
<td>10.0M</td>
</tr>
</tbody>
</table>

Table 4: Data statistics for all tasks.

We use dropout 0.3 and label-smoothing 0.2 for IWSLT En→De, IWSLT En→It and NEWS En→De and dropout 0.3 and label-smoothing 0.1 for WMT14 En→De. The resulting NMT models had ca. 51M parameters for IWSLT En→De, IWSLT En→It and NEWS En→De and ca. 67M parameters for WMT14 En→De. The NMT training took around 24h for IWSLT En→De, IWSLT En→It and NEWS En→De and around 150h for WMT14 En→De on a single NVIDIA GeForce RTX 2080 Ti graphics card. The language models had ca. 26M parameters for IWSLT En→De, IWSLT En→It and NEWS En→De and ca. 41M parameters for WMT14 En→De. All language model trainings took around 150h on a single NVIDIA GeForce RTX 2080 Ti graphics card. Due to computational limitations, we report results only for a single run.
ACL 2023 Responsible NLP Checklist

A  For every submission:

☑️ A1. Did you describe the limitations of your work?
   Section Limitations

☒ A2. Did you discuss any potential risks of your work?
   The authors do not foresee potential risks of this work.

☑️ A3. Do the abstract and introduction summarize the paper’s main claims?
   Abstract and Introduction

☒ A4. Have you used AI writing assistants when working on this paper?
   Left blank.

B  ☑️ Did you use or create scientific artifacts?
   Section 4 Experiments

☑️ B1. Did you cite the creators of artifacts you used?
   Section 4 Experiments

☒ B2. Did you discuss the license or terms for use and/or distribution of any artifacts?
   All artifacts that were used allow such usage for research purposes.

☒ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
   All artifacts that were used allow such usage for research purposes.

☒ B4. Did you discuss the steps taken to check whether the data that was collected/used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect/anonymize it?
   We only use standard datasets which allow usage for research purposes.

☑️ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
   Section 4 Experiments

☑️ B6. Did you report relevant statistics like the number of examples, details of train/test/dev splits, etc. for the data that you used/created? Even for commonly-used benchmark datasets, include the number of examples in train/validation/test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
   Section Appendix

C  ☑️ Did you run computational experiments?
   Section 4 Experiments

☑️ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
   Section Appendix

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.
C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?

Section 4 Experiments

C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?

Section Appendix

C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?

Section 4 Experiments

D Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?

No response.

D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants’ demographic (e.g., country of residence)?

No response.

D3. Did you discuss whether and how consent was obtained from people whose data you’re using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?

No response.

D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?

No response.

D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?

No response.