Extreme Exploitation of Language Resources for Language Transfer and Pre-training in Neural Machine Translation
神经机器翻译语言迁移和预训练中语言资源的极致利用

LIU Qun 刘群

Huawei Noah’s Ark Lab 华为诺亚方舟实验室

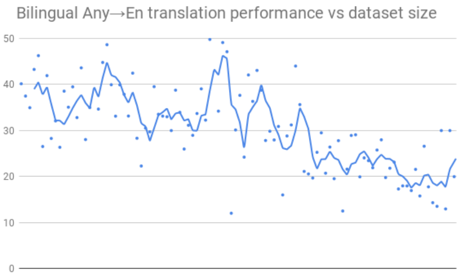
An invited talk at CCMT
2022-08-10

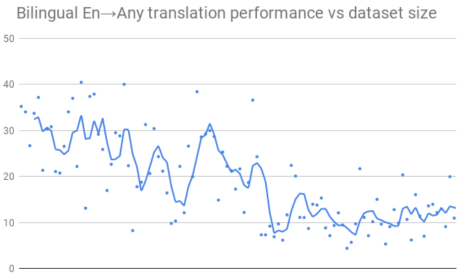
Content

Content

Low-resource MT: Performances

- Neural machine translation has been quite successful in high-resource conditions, but
- Still suffers in low-resource settings.

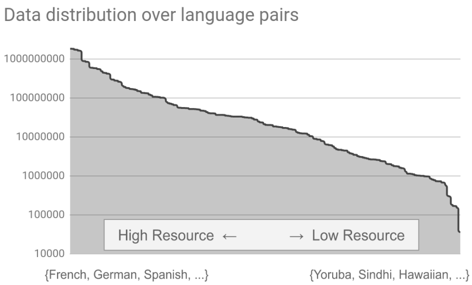




(Arivazhagan et al., 2019)

Low-resource MT: Scenarios

- Scenarios: Lack of parallel data
  - Low-resource Langauges: There are more than 7000 languages, of which most are low-resource langauges.
  - Low-resource Domains: Most of the parallel corpara exist in news domain, while the other domains are of low resources, including patent, law, medicine, etc.
- Values: increasing demands of the application of MT in low-resouce languages and domains.



HUAWEI NOAH'S ARK LAB

Previous work on low-resource MT

- Low-resource machine translation commonly uses auxiliary data
- Using parallel data of high-resource languages: transfer learning
  - (Zoph et al., 2016)
  - (Kim et al., 2019)
- Using monolingual data
  - Back-translation (BT) (Sennrich et al., 2016)
  - Pretrained language model (PLM) (Rothe et al., 2020)
- Multilingual machine translation
  - Multilingual machine translation commonly shares vocabulary, which makes it difficult to extend to new languages (Kocmi and Bojar, 2018)* (asterisk indicates such limitation)

Content

Dual Transfer: the Approach

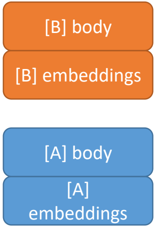
Consider transferring a high-resource A$\rightarrow$B MT model to a low-resource P$\rightarrow$Q model

1. Train PLM_A and PLM_B with monolingual data of A and B separately
2. Train PLM_P and PLM_Q based on PLM_A and PLM_B with monolingual data of P and Q:
   2.1 The model parameters of PLM_P and PLM_Q are inherented from PLM_A and PLM_B and frozed
   2.2 The word embeddings of PLM_P and PLM_Q are initialized randomly and trained
   2.3 The vocabularies of PLM_P and PLM_Q can be totally different from those of PLM_A and PLM_B
3. Train $\text{MT}_{\text{A}\rightarrow\text{B}}$ based on PLM_A and PLM_B with A$\rightarrow$B parallel data:
   3.1 Initialize $\text{MT}_{\text{A}\rightarrow\text{B}}$ encoder with PLM_A, and decoder with PLM_B
   3.2 Freeze word embeddings of the $\text{MT}_{\text{A}\rightarrow\text{B}}$ encoder and decoder during training
4. Train $\text{MT}_{\text{P}\rightarrow\text{Q}}$ based on $\text{MT}_{\text{A}\rightarrow\text{B}}$, PLM_P and PLM_Q with P$\rightarrow$Q parallel data:
   4.1 Initialize $\text{MT}_{\text{P}\rightarrow\text{Q}}$ encoder word embeddings with those in PLM_P
   4.2 Initialize $\text{MT}_{\text{P}\rightarrow\text{Q}}$ decoder word embeddings with those in PLM_Q
   4.3 Initialize $\text{MT}_{\text{P}\rightarrow\text{Q}}$ encoder and decoder model parameters with those in $\text{MT}_{\text{A}\rightarrow\text{B}}$
   4.4 Finetune $\text{MT}_{\text{P}\rightarrow\text{Q}}$ with P$\rightarrow$Q parallel data

Dual Transfer: the Approach

- Transfer:
  - from High-resouce Scenario A$\rightarrow$B
  - to Low-resource Scenario P$\rightarrow$Q

--->: initialization
Gray: frozen parameters
Color: trainable parameters



(1) Train PLM on mono-lingual data of A and B separately

Dual Transfer: the Approach

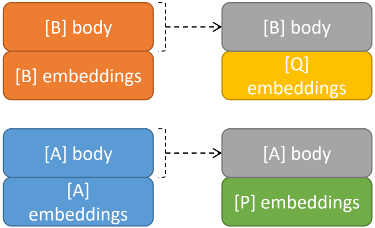
Consider transferring a high-resource A$\rightarrow$B MT model to a low-resource P$\rightarrow$Q model

1. Train PLM_A and PLM_B with monolingual data of A and B separately
2. Train PLM_P and PLM_Q based on PLM_A and PLM_B with monolingual data of P and Q:
   - 2.1 The model parameters of PLM_P and PLM_Q are inherented from PLM_A and PLM_B and frozed
   - 2.2 The word embeddings of PLM_P and PLM_Q are initialized randomly and trained
   - 2.3 The vocabularies of PLM_P and PLM_Q can be totally different from those of PLM_A and PLM_B
3. Train $\text{MT}_{\text{A}\rightarrow\text{B}}$ based on PLM_A and PLM_B with A$\rightarrow$B parallel data:
   - 3.1 Initialize $\text{MT}_{\text{A}\rightarrow\text{B}}$ encoder with PLM_A, and decoder with PLM_B
   - 3.2 Freeze word embeddings of the $\text{MT}_{\text{A}\rightarrow\text{B}}$ encoder and decoder during training
4. Train $\text{MT}_{\text{P}\rightarrow\text{Q}}$ based on $\text{MT}_{\text{A}\rightarrow\text{B}}$, PLM_P and PLM_Q with P$\rightarrow$Q parallel data:
   - 4.1 Initialize $\text{MT}_{\text{P}\rightarrow\text{Q}}$ encoder word embeddings with those in PLM_P
   - 4.2 Initialize $\text{MT}_{\text{P}\rightarrow\text{Q}}$ decoder word embeddings with those in PLM_Q
   - 4.3 Initialize $\text{MT}_{\text{P}\rightarrow\text{Q}}$ encoder and decoder model parameters with those in $\text{MT}_{\text{A}\rightarrow\text{B}}$
   - 4.4 Finetune $\text{MT}_{\text{P}\rightarrow\text{Q}}$ with P$\rightarrow$Q parallel data

Dual Transfer: the Approach

- Transfer:
  - from High-resouce Scenario A→B
  - to Low-resource Scenario P→Q

- - ->: initialization
Gray: frozen parameters
Color: trainable parameters



(1) Train PLM on mono-lingual data of A and B separately

(2) Train PLM on mono-lingual data of P and Q separately

Dual Transfer: the Approach

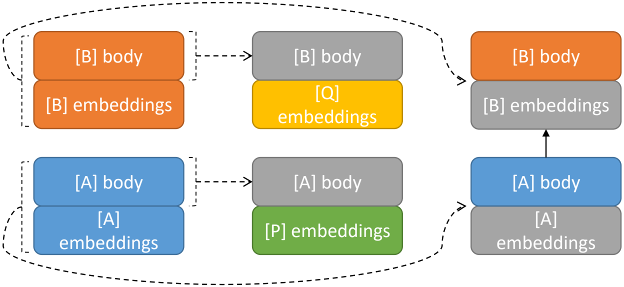
Consider transferring a high-resource A$\rightarrow$B MT model to a low-resource P$\rightarrow$Q model

1. Train PLM_A and PLM_B with monolingual data of A and B separately
2. Train PLM_P and PLM_Q based on PLM_A and PLM_B with monolingual data of P and Q:
   - 2.1 The model parameters of PLM_P and PLM_Q are inherented from PLM_A and PLM_B and frozed
   - 2.2 The word embeddings of PLM_P and PLM_Q are initialized randomly and trained
   - 2.3 The vocabularies of PLM_P and PLM_Q can be totally different from those of PLM_A and PLM_B
3. Train $\text{MT}_{\text{A}\rightarrow\text{B}}$ based on PLM_A and PLM_B with A$\rightarrow$B parallel data:
   - 3.1 Initialize $\text{MT}_{\text{A}\rightarrow\text{B}}$ encoder with PLM_A, and decoder with PLM_B
   - 3.2 Freeze word embeddings of the $\text{MT}_{\text{A}\rightarrow\text{B}}$ encoder and decoder during training
4. Train $\text{MT}_{\text{P}\rightarrow\text{Q}}$ based on $\text{MT}_{\text{A}\rightarrow\text{B}}$, PLM_P and PLM_Q with P$\rightarrow$Q parallel data:
   - 4.1 Initialize $\text{MT}_{\text{P}\rightarrow\text{Q}}$ encoder word embeddings with those in PLM_P
   - 4.2 Initialize $\text{MT}_{\text{P}\rightarrow\text{Q}}$ decoder word embeddings with those in PLM_Q
   - 4.3 Initialize $\text{MT}_{\text{P}\rightarrow\text{Q}}$ encoder and decoder model parameters with those in $\text{MT}_{\text{A}\rightarrow\text{B}}$
   - 4.4 Finetune $\text{MT}_{\text{P}\rightarrow\text{Q}}$ with P$\rightarrow$Q parallel data

Dual Transfer: the Approach

- Transfer:
  - from High-resouce Scenario A→B
  - to Low-resource Scenario P→Q

--->: initialization
Gray: frozen parameters
Color: trainable parameters



(1) Train PLM on mono-lingual data of A and B separately

(2) Train PLM on mono-lingual data of P and Q separately

(3) Train MT on A→B parallel data

Dual Transfer: the Approach

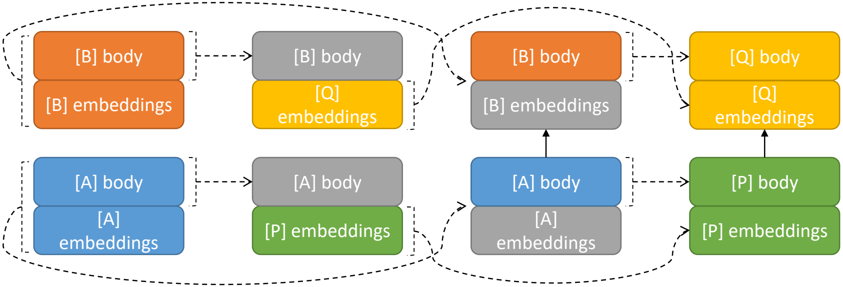
Consider transferring a high-resource A$\rightarrow$B MT model to a low-resource P$\rightarrow$Q model

1. Train PLM_A and PLM_B with monolingual data of A and B separately
2. Train PLM_P and PLM_Q based on PLM_A and PLM_B with monolingual data of P and Q:
   2.1 The model parameters of PLM_P and PLM_Q are inherented from PLM_A and PLM_B and frozed
   2.2 The word embeddings of PLM_P and PLM_Q are initialized randomly and trained
   2.3 The vocabularies of PLM_P and PLM_Q can be totally different from those of PLM_A and PLM_B
3. Train $\text{MT}_{\text{A}\rightarrow\text{B}}$ based on PLM_A and PLM_B with A$\rightarrow$B parallel data:
   3.1 Initialize $\text{MT}_{\text{A}\rightarrow\text{B}}$ encoder with PLM_A, and decoder with PLM_B
   3.2 Freeze word embeddings of the $\text{MT}_{\text{A}\rightarrow\text{B}}$ encoder and decoder during training
4. Train $\text{MT}_{\text{P}\rightarrow\text{Q}}$ based on $\text{MT}_{\text{A}\rightarrow\text{B}}$, PLM_P and PLM_Q with P$\rightarrow$Q parallel data:
   4.1 Initialize $\text{MT}_{\text{P}\rightarrow\text{Q}}$ encoder word embeddings with those in PLM_P
   4.2 Initialize $\text{MT}_{\text{P}\rightarrow\text{Q}}$ decoder word embeddings with those in PLM_Q
   4.3 Initialize $\text{MT}_{\text{P}\rightarrow\text{Q}}$ encoder and decoder model parameters with those in $\text{MT}_{\text{A}\rightarrow\text{B}}$
   4.4 Finetune $\text{MT}_{\text{P}\rightarrow\text{Q}}$ with P$\rightarrow$Q parallel data

Dual Transfer: the Approach

- Transfer:
  - from High-resouce Scenario A→B
  - to Low-resource Scenario P→Q

--->: initialization
Gray: frozen parameters
Color: trainable parameters



(1) Train PLM on mono-lingual data of A and B separately

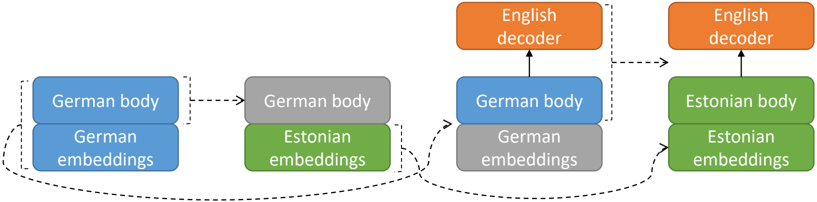
(2) Train PLM on mono-lingual data of P and Q separately

(3) Train MT on A→B parallel data

(4) Finetune on P→Q parallel data

Variation: Shared Target Transfer (B=Q=en)

--->: initialization
Gray: frozen parameters
Color: trainable parameters



(1) Train PLM on German monolingual data

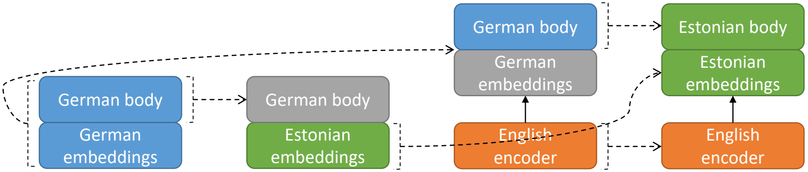
(2) Train PLM on Estonian monolingual data

(3) Train MT on German→ English parallel data

(4) Finetune on Estonian→ English parallel data

Variation: Shared Source Transfer (A=P=en)

- - ->: initialization
Gray: frozen parameters
Color: trainable parameters



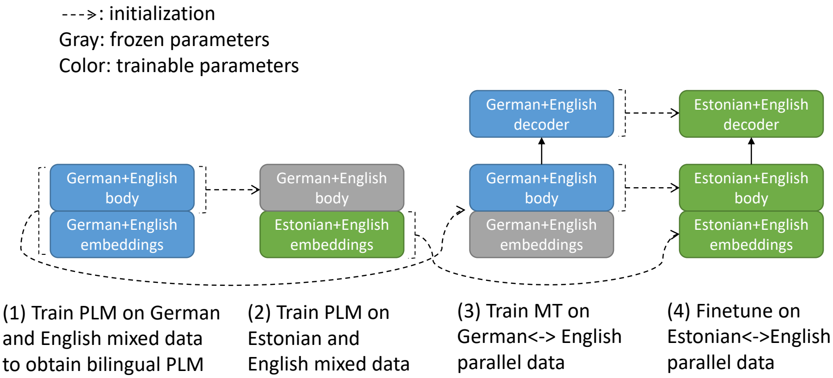
(1) Train PLM on German monolingual data

(2) Train PLM on Estonian monolingual data

(3) Train MT on English→ German parallel data

(4) Finetune on English→ Estonian parallel data

Variation: Bidirectional MT Transfer (A=B=de+en, P=Q=et+en)



Other Variations

- Domain Transfer:
  - A = Source Language in Source Domain
  - B = Target Language in Source Domain
  - P = Source Language in Target Domain
  - Q = Target Language in Target Domain
- Other neural network architectures
  - This Dual Transfer framework can also be applied to NMT of other neural network architectures
  - For example, if a low-resource RNN-based NMT is desired, then high-resource RNN-based PLMs and a high-resource RNN-based NMT can be prepared as the parent models

Experiments: Dataset

language code	# sentence pair
de-en	5.9m
et-en	1.9m
tr-en	207k
fr-es	10k
de-en medical	347k

language code	# sentence
en	94m
de	147m
et	139m
tr	100m
fr	4.1m
es	4.2m
en medical	4.0m
de medical	3.6m

Experiments: Usage of auxiliary data

	High-resource language		Low-resource language	
	monolingual	parallel	monolingual	parallel
no transfer				✔
(Zoph et al., 2016)		✔		✔
(Kim et al., 2019)		✔	✔	✔
BERT2RND			✔	✔
BERT2BERT			✔	✔
(Kocmi and Bojar, 2018)*		✔		✔
BBERT2BBERT*			✔	✔
BBERT transfer*	✔	✔	✔	✔
dual transfer	✔	✔	✔	✔

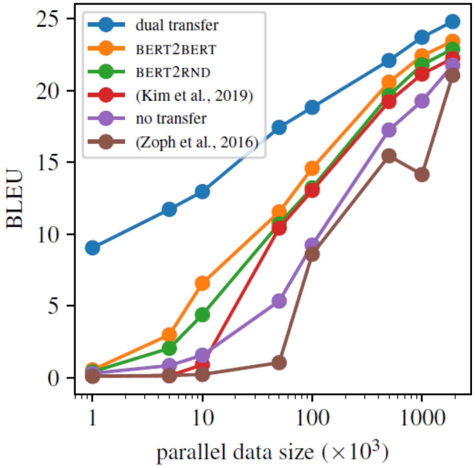
Results: Dual Transfer from de→en to et→en

Our approach significantly outperforms strong baselines:

	et-en BLEU
no transfer	21.76
(Zoph et al., 2016)	21.07
(Kim et al., 2019)	22.25
BERT2RND	22.89
BERT2BERT	23.44
(Kocmi and Bojar, 2018)*	23.58
BBERT2BBERT*	23.90
BBERT transfer*	24.08
dual transfer	**24.81**

Results: different parallel data size for low-resource languages

Our approach performs reasonably well even with a very small amount of parallel data, alleviating the data issue for low-resource language pairs



HUAWEI NOAH'S ARK LAB

Results: no parallel data size for low-resource languages

Freezing the entire encoder when training the parent NMT (in step 3) enables our approach to perform zero-shot translation

parallel data size ($\times 10^3$)	**0**	**1**	**5**	**10**	**50**	**100**	**500**	**1000**
dual transfer	0.43	9.06	11.74	12.97	17.44	18.84	22.10	23.72
+freezing parent NMT encoder	**6.20**	8.82	11.58	12.76	16.62	18.50	21.69	23.59

However, it does not have advantage when parallel data is available.

Results: transfer to other translation directions from de→en

	tr-en BLEU	en-et BLEU	en-tr BLEU	fr-es BLEU
no transfer	15.44	16.29	9.63	10.59
BERT2BERT	19.73	17.36	11.78	18.26
dual transfer	**21.12**	**19.41**	**13.18**	**22.28**

Results: our approach is complementary to back-translation

	en-et BLEU
no transfer	16.29
dual transfer	18.79
no transfer + 4m BT	19.78 (+3.49)
dual transfer + 4m BT	**22.34** (+3.55)
no transfer + 130m BT	20.52 (+4.23)
dual transfer + 130m BT	22.23 (+3.44)

Note: Numbers in parentheses indicate differences from the corresponding approach trained on authentic parallel data.

Results: domain adaptation

For domain adaptation (from news to medical), our approach can use either source domain (parent) vocabulary, or target domain (child) vocabulary

	BLEU
no transfer (child)	62.94
BERT2BERT (child)	64.33
finetuning (parent)	64.91
dual transfer (parent)	65.14
dual transfer (child)	**65.40**

Results: on in-house data

Myanmar-English

language code	# sentence (pair)
de→en	175m
en→de	127m
my-en	1.5m
my→en BT	86m
en→my BT	21m
en	175m
de	127m
my	24m

en-my	BLEU
Google	13.60
dual transfer	**15.81**

my-en	BLEU
Google	23.32
dual transfer	**24.91**

Results: on in-house data

Assamese-English: Transferring from the related Bengali is helpful

language code	# sentence (pair)
de→en	175m
en→de	127m
bn-en	6m
as-en	0.1m
en	175m
de	127m
bn	56m
as	13m

en-as	BLEU
transfer from de	20.98
transfer from bn	**21.52**

as-en	BLEU
transfer from de	25.88
transfer from bn	**26.57**
transfer from de BT	25.00
transfer from bn BT	25.91

Publication: Dual Transfer

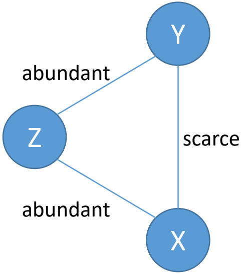
Two Parents, One Child:
Dual Transfer for Low-Resource Neural Machine Translation

Meng Zhang, Liangyou Li, Qun Liu
Huawei Noah's Ark Lab
{zhangmeng92, liliangyou, qun.liu}@huawei.com

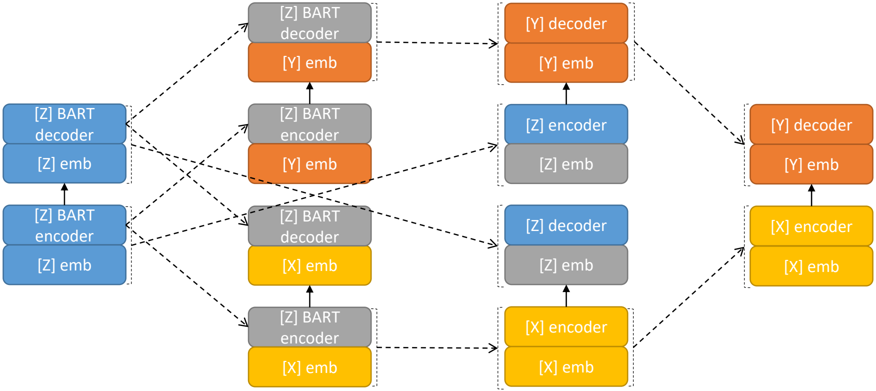
Published in: Findings of ACL2021

Content

Scenario: Triangle Translation



Trianglar Transfer: the Approach



Usage of auxiliary data

	X	Y	Z	X-Z	Z-Y	X-Y
Baseline						✔
Pivot translation				✔	✔	
Step-wise pre-training				✔	✔	✔
Shared target dual transfer	✔		✔		✔	✔
Shared source dual transfer		✔	✔	✔		✔
Triangular transfer	✔	✔	✔	✔	✔	✔

Results

	fr-de BLEU
Baseline	13.49
Pivot translation	18.99
Step-wise pre-training	18.49
Shared target dual transfer	18.88
Shared source dual transfer	18.89
Triangular transfer	**19.91**

Publication: Trianglar Transfer

Triangular Transfer: Freezing the Pivot for Triangular Machine Translation

Meng Zhang, Liangyou Li, Qun Liu
Huawei Noah's Ark Lab
{zhangmeng92, liliangyou, qun.liu}@huawei.com

Published in: Proceedings of ACL2022 (short paper)

References

- Arivazhagan, Naveen, Ankur Bapna, Orhan Firat, Dmitry Lepikhin, Melvin Johnson, Maxim Krikun, Mia Xu Chen, et al. 2019. "Massively Multilingual Neural Machine Translation in the Wild: Findings and Challenges," July. https://arxiv.org/abs/1907.05019v1.
- Conneau, Alexis, and Guillaume Lample. 2019. "Cross-Lingual Language Model Pretraining." In *Advances in Neural Information Processing Systems 32*, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, 7059–7069. Curran Associates, Inc. http://papers.nips.cc/paper/8928-cross-lingual-language-model-pretraining.pdf.
- Kim, Yunsu, Yingbo Gao, and Hermann Ney. 2019. "Effective Cross-Lingual Transfer of Neural Machine Translation Models without Shared Vocabularies." In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1246–1257. Florence, Italy: Association for Computational Linguistics. https://doi.org/10.18653/v1/P19-1120.
- Kocmi, Tom, and Ondřej Bojar. 2018. "Trivial Transfer Learning for Low-Resource Neural Machine Translation." In *Proceedings of the Third Conference on Machine Translation: Research Papers*, 244–252. Brussels, Belgium: Association for Computational Linguistics. https://doi.org/10.18653/v1/W18-6325.
- Rothe, Sascha, Shashi Narayan, and Aliaksei Severyn. 2020. "Leveraging Pre-Trained Checkpoints for Sequence Generation Tasks." *Transactions of the Association for Computational Linguistics* 8 (0): 264–80.
- Sennrich, Rico, Barry Haddow, and Alexandra Birch. 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)*, 86–96. Berlin, Germany: Association for Computational Linguistics. https://doi.org/10.18653/v1/P16-1009.
- Zoph, Barret, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. "Transfer Learning for Low-Resource Neural Machine Translation." In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1568–1575. Austin, Texas: Association for Computational Linguistics. https://doi.org/10.18653/v1/D16-1163.

Content

Content

CeMAT: Universal Conditional Masked Language Pre-training for NMT

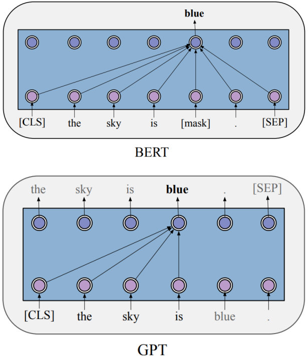
- Background: Pre-training for NMT
- Our method: CeMAT

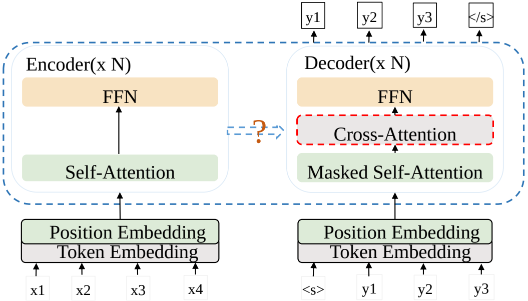
Transfer Learning vs. Pre-training for MT

- Transfer learning:
  - Focusing on specificing language pairs
  - Utilize a small set of language resources
  - Model size is flexible
- Pre-training:
  - Single model for multiple language pairs
  - Utilize a variaty of language resources
  - Model size is relatively large

Pre-training NMT: Using pretrained BERT and GPT

- The architecture of GPT is different from the decoder of a encoder-decoder model (no cross-attention layer)
- The cross-attention layer is not pre-trained

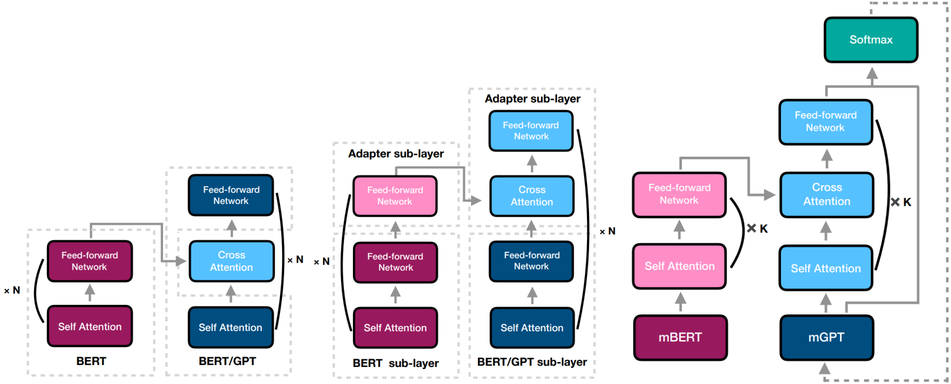




Guillaume Lample et al., Cross-lingual Language Model Pretraining. 2019 Radford et al., Improving Language Understanding by Generative Pre-Training. 2018
Jacob Devlin et al., BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 2018

Pre-training NMT: Insert adapter modules

- Insert adapter modules to extract knowledge from PLMs and fill the gap between PLMs and MT models
- Need additional parameters



Direct Adapter Graformer

Zewei Sun et al., Multilingual Translation via Grafting Pre-trained Language Models. 2021

HUAWEI NOAH'S ARK LAB

Pre-training NMT: MASS (Microsoft)

- Auto-encoder: masked sequence-to-sequence PLMs
- Data: monolingual corpus (source and target)
- Self-supervised task: reconstruct a consecutive sentence fragment given the remaining part of the sentence
- Fine-tuning: initialize directly with PLMs parameters, and then fine-tune on the translation datasets

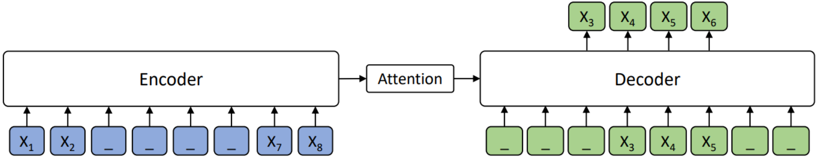


Figure 1. The encoder-decoder framework for our proposed MASS. The token "_" represents the mask symbol $[\mathbb{M}]$.

Kaitao Song et al., MASS: Masked Sequence to Sequence Pre-training for Language Generation. 2019

HUAWEI NOAH'S ARK LAB

Pre-training NMT: CSP (Tencent)

- Auto-encoder: code-switching (CS) on MASS
- CS: replace the source fragment with their translation words based on probabilistic translation lexicons
- Data: monolingual corpus (source and target)
- Self-supervised task

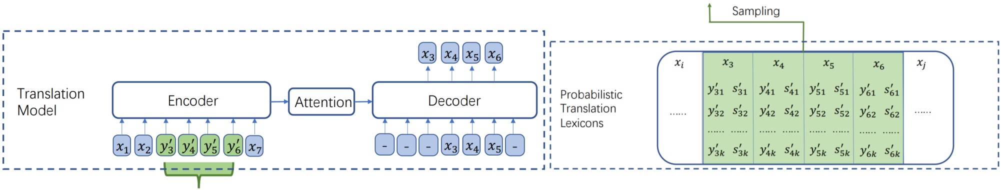
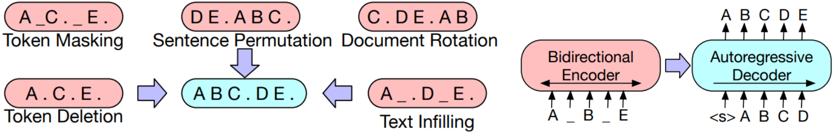


Figure 1: The training example of our proposed CSP which randomly replaces some words in the source input with their translation words based on the probabilistic translation lexicons. Identical to MAS, the token - represents the padding in the decoder. The attention module represents the attention between the encoder and decoder

Yang, Z et al., CSP: Code-Switching Pre-training for Neural Machine Translation. 2020

Pre-training NMT: BART and mBART (FaceBook/Meta)

- DAE: a denoising auto-encoding for pre-training sequence-to-sequence PLMs
- Noising: masking, deletion, ⋯
- Data: monolingual corpus in many (25/50) languages
- Self-supervised task: learn to reconstruct the original text from the corrupted text



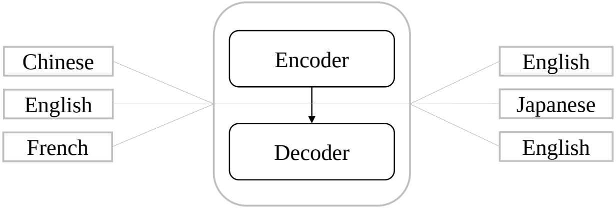
(c) BART: Inputs to the encoder need not be aligned with decoder outputs, allowing arbitary noise transformations. Here, a document has been corrupted by replacing spans of text with mask symbols. The corrupted document (left) is encoded with a bidirectional model, and then the likelihood of the original document (right) is calculated with an autoregressive decoder. For fine-tuning, an uncorrupted document is input to both the encoder and decoder, and we use representations from the final hidden state of the decoder.

Lewis, M et al., BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. 2020
Liu, Y et al., Multilingual Denoising Pre-training for Neural Machine Translation. 2020.

HUAWEI NOAH'S ARK LAB

Multilingual NMT

- Multilingual NMT task
- Data: bilingual in many languages
- Supervised task: transfer learning from high resource or similar language pairs
- Fine-tuning: initialize directly with PLMs parameters, and then fine-tune on the specific translation datasets



Pre-training NMT: mRASP (ByteDance)

- Code-switching on multilingual NMT task
- Data: bilingual in many languages
- Supervised task
- Fine-tuning: initialize directly with PLMs parameters, and then fine-tune on the specific translation datasets

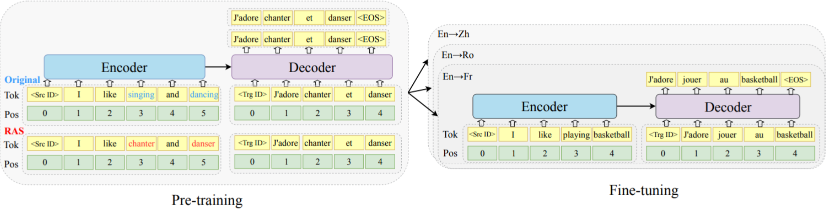
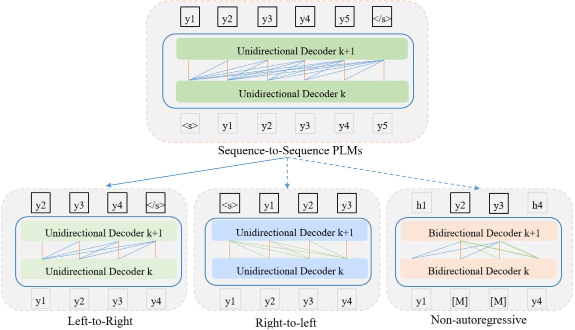


Figure 1: The proposed mRASP method. "Tok" denotes token embedding while "Pos" denotes position embedding. During the pre-training phase, parallel sentence pairs in many languages are trained using translation loss, together with their substituted ones. We randomly substitute words with the same meanings in the source and target sides. During the fine-tuning phase, we further train the model on the downstream language pairs to obtain specialized MT models.

Lin, Z et al., Pre-training Multilingual Neural Machine Translation by Leveraging Alignment Information. 2020.

Pre-training NMT: Limitations

- Pre-training a complete sequence-to-sequence model
- Restricted with specific downstream tasks may lead to higher costs of PLMs
  - √ Left-to-right autoregressive
  - ☐ Right-to-left autoregressive
  - ☐ Non-autoregressive



Pre-training NMT: Limitations

- Pre-training a complete sequence-to-sequence model
- Restricted with specific downstream tasks may lead to higher costs of PLMs
  - √ Left-to-right autoregressive
  - ☐ Right-to-left autoregressive
  - ☐ Non-autoregressive
- Self-supervised pre-training
  - e.g., MASS, CSP, mBART
  - Training with monolingual data
  - No improvement on extremely-high (>25M) resource translation tasks
- Supervised pre-training
  - e.g., mRASP
  - Training with bilingual data
  - No effective enhancements when using monolingual data
- Few pre-training models using both monolingual and bilingual data

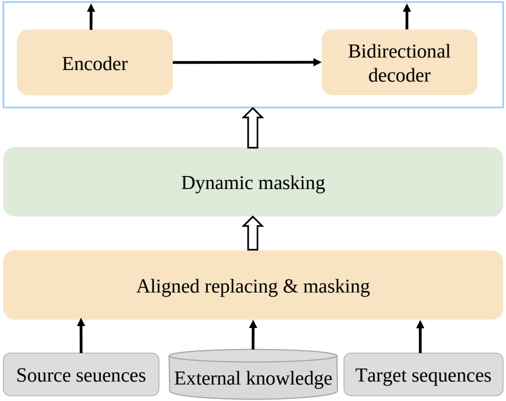
Content

CeMAT: Universal Conditional Masked Language Pre-training for NMT

Background: Pre-training for NMT

Our method: CeMAT

CeMAT: Overall structure of CeMAT



Model
- Bidirectional decoder
- Joint training

Policy
- Aligned replacing & masking
- Dynamic masking

Data
- Source and target sequences
- External knowledge
- Output: masked tokens

CeMAT: Data

Input

- **Source and target sequences**
  - Bilingual in many languages
  - Monolingual in many languages
  - Monolingual sentences are duplicated to have the same format of bilingual data
- **External knowledge**
  - Extracting aligned tokens or phrases from the source and target sequences

Output

- **Prediction task**
  - All the tokens that are masked in the source and target sequences

CeMAT: Policy for preparing bilingual and monolingual data

■ Aligned replacing & masking (ACM)

a. Getting the aligned set

- Extracting aligned pairs of source and target
- **Selecting a subset** with a certain probability

b. **Replacing** on the source sequence (subsets)

- Replacing the tokens with their translation words

c. **Masking** on the target sequence (subsets)

- Masking the tokens, then predicting at the output

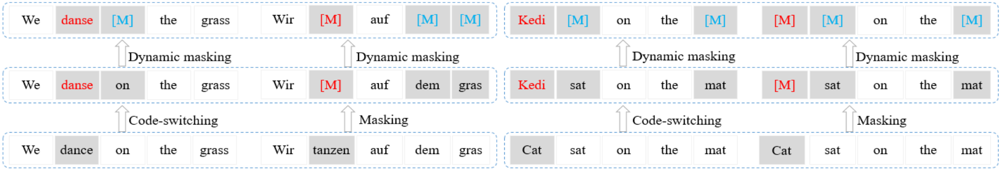
■ Dynamic masking (DM)

a. Bilingual

- **Separately masking** source and target sequences
- Maintain a **dynamic masking probability**
- Guaranteeing greater masking probability of target

b. Monolingual

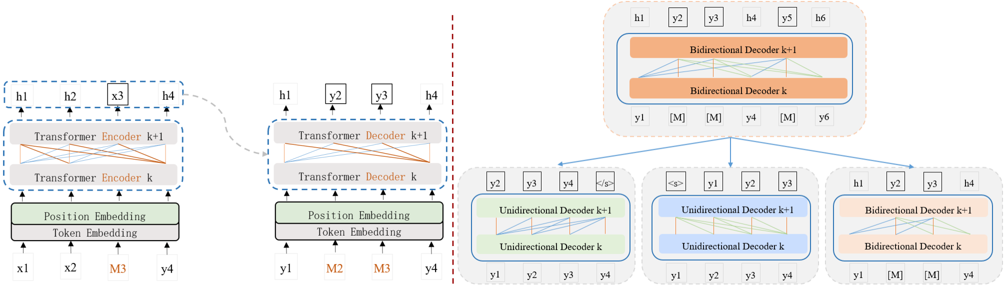
- Guaranteeing consistent mask tokens



(a) Bilingual (b) Monolingual

CeMAT: Model

- Sequence-to-sequence joint representation model
  - Encoder and bidirectional decoder
  - Joint training
- Matching all three downstream tasks



Sequence-to-Sequence joint representation model

Left-to-Right

Right-to-left

Non-autoregressive

Experiments: Autoregressive NMT

- 13 translation tasks
  - Improved 2.3∼14.4 BLEU
  - Average improvement 7.9 BLEU
  - SOTA:
    - 3.8 BLEU higher than mBART
    - 1.2 BLEU higher than mRASP

- Transformer big (6+6,16*1024*4096)
- WMT: open datasets for translation tasks
- BLEU (↑): metric for translation tasks

Significant improvements on low, medium, high and extremely-high resources

BLEU	WMT19		WMT17		WMT18		WMT17		WMT17		WMT19	WMT19	WMT14	Avg
	91k(low)		207k(low)		1.94M(mid)		2.66M(mid)		4.5M(mid)		11M(high)	38M(extre -high)	41M(extre -high)	
	En2Kk	Kk2En	En2Tr	Tr2en	En2Et	Et2En	En2Fi	Fi2En	En2Lv	Lv2En	En2Cs	En2De	En2Fr	
Direct	0.2	0.8	9.5	12.2	17.9	22.6	20.2	21.8	12.9	15.6	16.5	30.9	41.4	17.1
mBART	2.5	7.4	17.8	22.5	21.4	27.8	22.4	28.5	15.9	19.3	18.0	30.5	41.0	21.2
mRASP	8.3	12.3	20.0	23.4	20.9	26.8	24.0	28.0	21.6	**24.4**	19.9	35.2	**44.3**	23.8
Ours	**8.8**	**12.9**	**23.9**	**23.6**	**22.2**	**28.5**	**25.4**	**28.7**	**22.0**	24.3	**21.5**	**39.2**	43.7	25.0
Improve	+8.6	+12.1	+14.4	+11.4	+4.3	+5.9	+5.2	+6.9	+9.1	+8.7	+5.0	+8.3	+2.3	+7.9

Experiments: Autoregressive NMT

- Compatible with BT (Back-Translation)

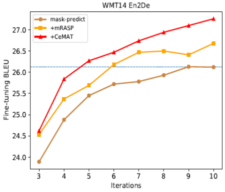
BLEU	WMT16		
	En2Ro	Ro2En	Ro2En(+BT)
Direct	34.3	34.0	36.8
mBART	37.7	37.8	38.8
mRASP	37.6	36.9	38.9
XLM	--	35.6	38.5
Ours	38.0	37.1	39.0

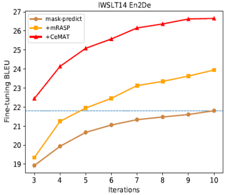
- Ablation experiments

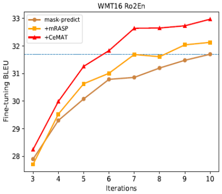
BLEU	WMT19		WMT18		Avg
	91k(low)		1.94M （mid）		
	Kk2En	En2Kk	En2Et	Et2En	
Direct	0.2	0.8	17.9	22.6	10.4
Bilingual	7.8	5.5	19.1	24.4	14.2
Monolingual	5.4	5.4	18.9	23.5	13.3
Bi- & Monolingual	9.0	5.6	19.0	25.2	14.7
w/o. ACM	8.4	5.1	18.2	24.3	14.0
w/o. DM	8.8	5.6	18.1	23.7	14.1
w/o. encoder loss	5.0	3.3	17.8	23.8	12.5
w/o. encoder mask	5.0	3.6	17.0	21.6	11.8

Experiments: Non-autoregressive NMT

- 6 translation tasks
  - Fine-tuning on Mask-predict
  - Average improvement of 2.5 BLEU
  - SOTA: 1.2 BLEU higher than mRASP
- Transformer big (6+6,16*1024*4096)
- WMT: open datasets for translation tasks
- BLEU (↑): metric for translation tasks







BLEU	IWSLT14		WMT16		WMT14		Avg
	16k(low)		770K(low)		4.5M(mid)		
	En2De	De2En	En2Ro	Ro2En	En2De	De2En	
Auto-regressive	23.9	32.8	34.1	34.5	28.0	32.7	31.0
Direct	22.0	28.4	31.5	31.7	26.1	29.0	28.1
mRASP	23.9	30.3	32.2	32.1	26.7	29.8	29.2
Ours	**26.7**	**33.7**	**33.3**	**33.0**	**27.2**	**29.9**	30.6

HUAWEI NOAH'S ARK LAB

Publication: CeMAT

Universal Conditional Masked Language Pre-training for Neural Machine Translation

Pengfei Li[1] **Liangyou Li**[1] **Meng Zhang**[1] **Minghao Wu**[2] **Qun Liu**[1]
[1]Huawei Noah's Ark Lab [2]Monash University
{lipengfei111, liliangyou, zhangmeng92, qun.liu}@huawei.com
minghao.wu@monash.edu

Published in: Proceedings of ACL2022 (long paper)

Content

Summary

Thank you!

把数字世界带入每个人、每个家庭、
每个组织，构建万物互联的智能世界。

Bring digital to every person, home and organization for a fully connected, intelligent world.

HUAWEI