# Parameter Sharing Methods for Multilingual Self-Attentional Translation Models

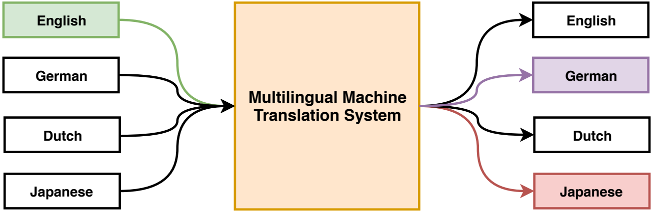
Devendra Sachan[1] **Graham Neubig**[2]

[1]Data Solutions Team,
Petuum Inc, USA

[2]Language Technologies Institute,
Carnegie Mellon University, USA

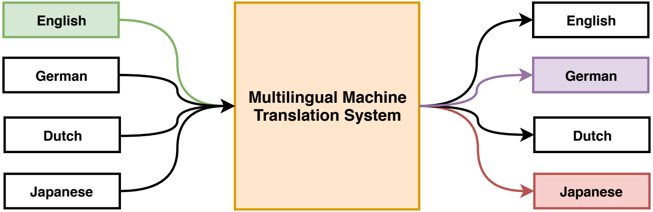
**Conference on Machine Translation**, Nov 2018

# Multilingual Machine Translation



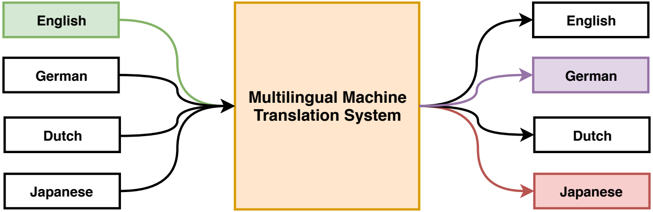

- Goal: Train a machine learning system to translate from multiple source languages to multiple target languages.

# Multilingual Machine Translation



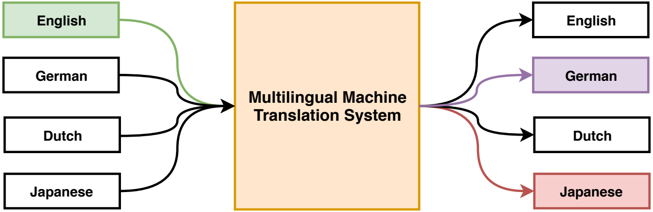

- Goal: Train a machine learning system to translate from multiple source languages to multiple target languages.
- Multilingual models follow the *multi-task learning* (MTL) paradigm

# Multilingual Machine Translation



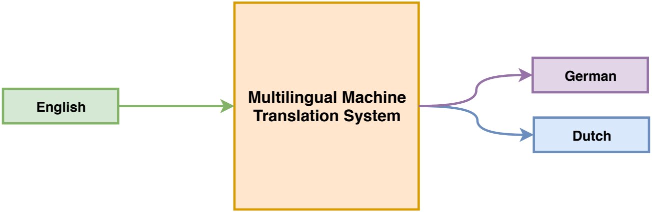

- Goal: Train a machine learning system to translate from multiple source languages to multiple target languages.
- Multilingual models follow the *multi-task learning* (MTL) paradigm
  1. Models are jointly trained on data from several language pairs.

# Multilingual Machine Translation



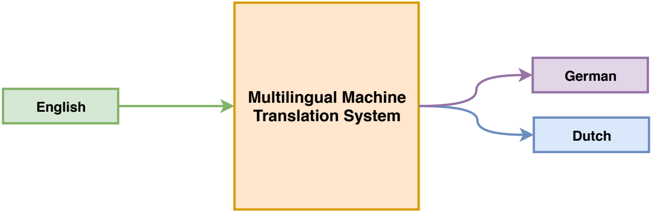

- Goal: Train a machine learning system to translate from multiple source languages to multiple target languages.
- Multilingual models follow the *multi-task learning* (MTL) paradigm
  1. Models are jointly trained on data from several language pairs.
  2. Incorporate some degree of parameter sharing.

# *One-to-Many* Multilingual Translation



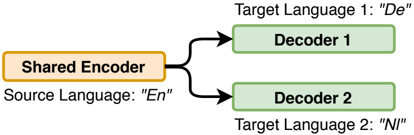

- Translation from a common source language ("En") to multiple target languages ("De" and "Nl")

# *One-to-Many* Multilingual Translation




- Translation from a common source language ("En") to multiple target languages ("De" and "Nl")
- Difficult task as we need to translate to (or generate) multiple target languages.

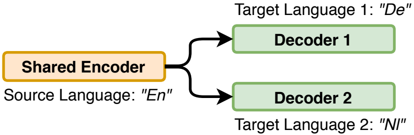
# Previous Approach: Separate Decoders




- One shared encoder and one decoder per target language.[1]

[1] Multi-Task Learning for Multiple Language Translation, ACL 2015

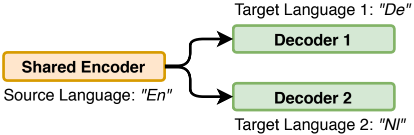
# Previous Approach: Separate Decoders




- One shared encoder and one decoder per target language.[1]
- Advantage: ability to model each target language separately.

[1] Multi-Task Learning for Multiple Language Translation, ACL 2015

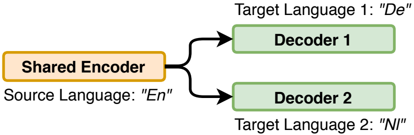
# Previous Approach: Separate Decoders




- One shared encoder and one decoder per target language.[1]
- Advantage: ability to model each target language separately.
- Disadvantages:
  1. Slower Training

[1] Multi-Task Learning for Multiple Language Translation, ACL 2015

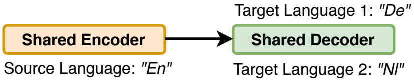
# Previous Approach: Separate Decoders




- One shared encoder and one decoder per target language.[1]
- Advantage: ability to model each target language separately.
- Disadvantages:
  1. Slower Training
  2. Increased memory requirements

[1] Multi-Task Learning for Multiple Language Translation, ACL 2015

# Previous Approach: Shared Decoder

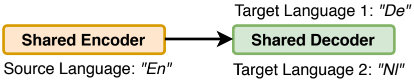



- Single *unified* model: shared encoder and shared decoder for all language pairs.[2]

---

[2]Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation, ACL 2017

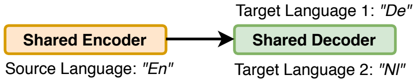
# Previous Approach: Shared Decoder




- Single *unified* model: shared encoder and shared decoder for all language pairs.[2]
- Advantages:
  - Trivially implementable: using a standard bilingual translation model.

[2] Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation, ACL 2017

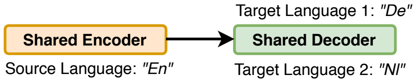
# Previous Approach: Shared Decoder




- Single *unified* model: shared encoder and shared decoder for all language pairs.[2]
- Advantages:
  - Trivially implementable: using a standard bilingual translation model.
  - Constant number of trainable parameters.

[2]Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation, ACL 2017

# Previous Approach: Shared Decoder

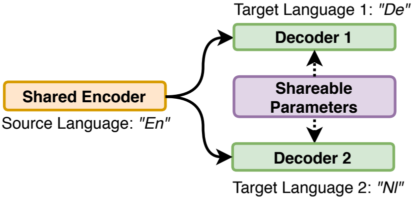



- Single *unified* model: shared encoder and shared decoder for all language pairs.[2]
- Advantages:
  - Trivially implementable: using a standard bilingual translation model.
  - Constant number of trainable parameters.
- Disadvantage: decoder's ability to model multiple languages can be significantly reduced.

---

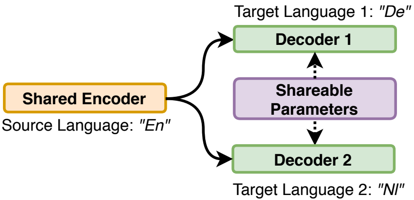
[2] Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation, ACL 2017

# Our Proposed Approach: **Partial Sharing**



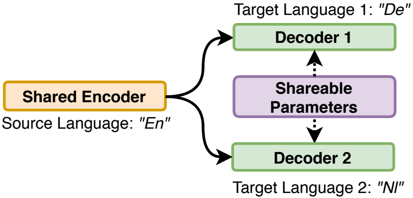

- Share **some but not all** parameters.

# Our Proposed Approach: **Partial Sharing**



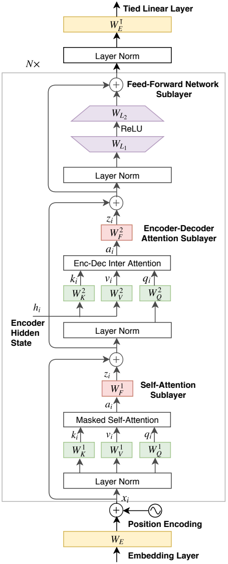

- Share **some but not all** parameters.
- Generalizes previous approaches.

# Our Proposed Approach: **Partial Sharing**




- Share **some but not all** parameters.
- Generalizes previous approaches.
- We focus on the self-attentional Transformer model.

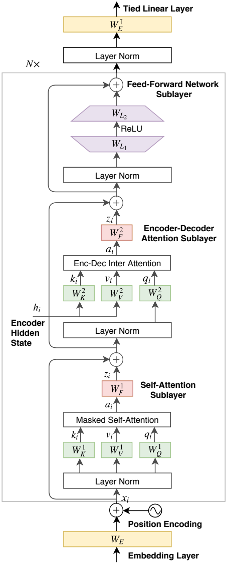
# Transformer Model[3]




[3] Attention is all you need, NIPS 2017

# Transformer Model[3]

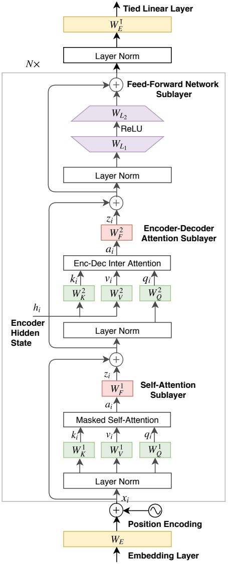
- Embedding Layer




[3] Attention is all you need, NIPS 2017

# Transformer Model[3]

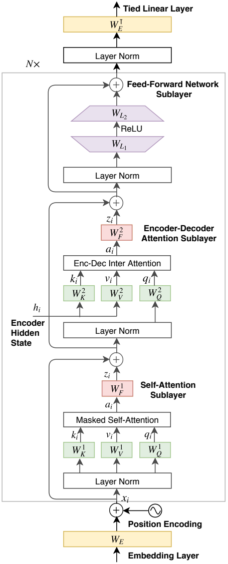
- Embedding Layer
- Encoder Layer (2 sublayers)




[3] Attention is all you need, NIPS 2017

# Transformer Model[3]

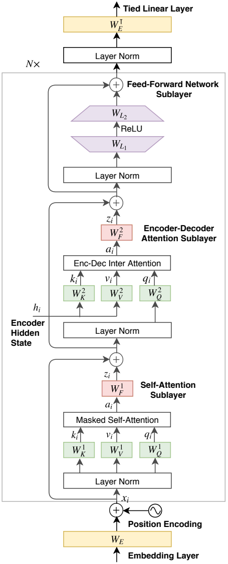
- Embedding Layer
- Encoder Layer (2 sublayers)
    1. Self-attention




[3] Attention is all you need, NIPS 2017

# Transformer Model[3]

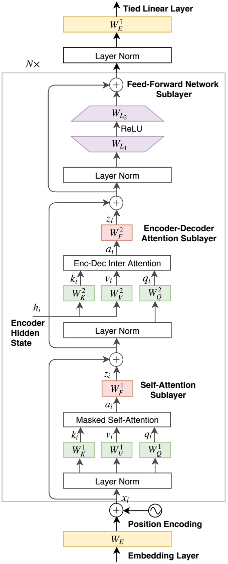
- Embedding Layer
- Encoder Layer (2 sublayers)
  1. Self-attention
  2. Feed-forward network




[3] Attention is all you need, NIPS 2017

# Transformer Model[3]

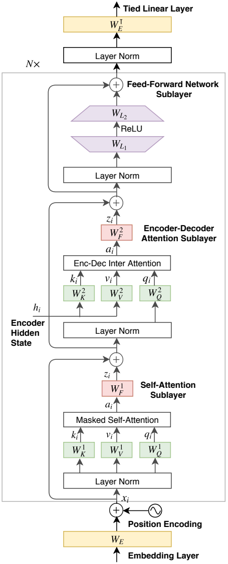
- Embedding Layer
- Encoder Layer (2 sublayers)
  1. Self-attention
  2. Feed-forward network
- Decoder Layer (3 sublayers)




[3] Attention is all you need, NIPS 2017

# Transformer Model[3]

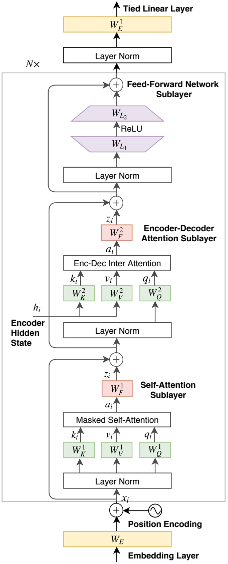
- Embedding Layer
- Encoder Layer (2 sublayers)
  1. Self-attention
  2. Feed-forward network
- Decoder Layer (3 sublayers)
  1. Masked self-attention




[3] Attention is all you need, NIPS 2017

# Transformer Model[3]

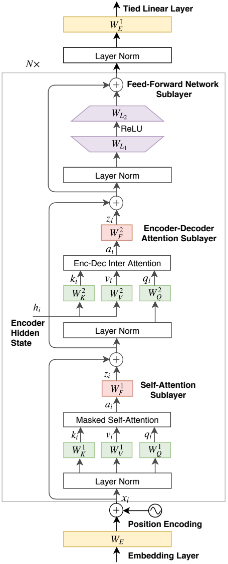
- Embedding Layer
- Encoder Layer (2 sublayers)
  1. Self-attention
  2. Feed-forward network
- Decoder Layer (3 sublayers)
  1. Masked self-attention
  2. Encoder-decoder attention




[3] Attention is all you need, NIPS 2017

# Transformer Model[3]

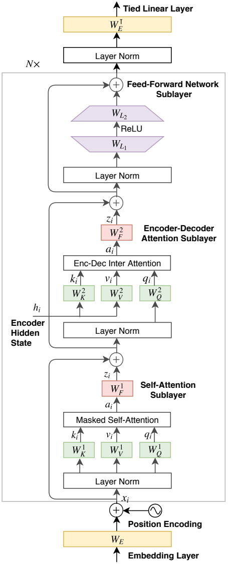
- Embedding Layer
- Encoder Layer (2 sublayers)
  1. Self-attention
  2. Feed-forward network
- Decoder Layer (3 sublayers)
  1. Masked self-attention
  2. Encoder-decoder attention
  3. Feed-forward network




[3] Attention is all you need, NIPS 2017

# Transformer Model[3]

- Embedding Layer
- Encoder Layer (2 sublayers)
  1. Self-attention
  2. Feed-forward network
- Decoder Layer (3 sublayers)
  1. Masked self-attention
  2. Encoder-decoder attention
  3. Feed-forward network
- Output generation layer

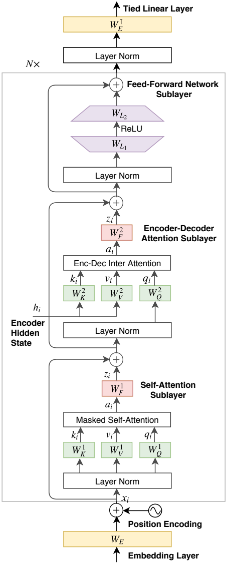



[3] Attention is all you need, NIPS 2017

# Transformer Decoder's Parameters

## Embedding Layer

- $\boldsymbol{W}_E \in \mathbb{R}^{d_m \times V}$



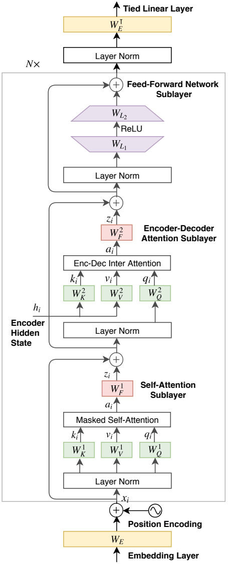
# Transformer Decoder's Parameters

## Embedding Layer

- $\boldsymbol{W}_E \in \mathbb{R}^{d_m \times V}$

## Masked Self-Attention

- $\boldsymbol{W}_K^1, \boldsymbol{W}_V^1, \boldsymbol{W}_Q^1, \boldsymbol{W}_F^1 \in \mathbb{R}^{d_m \times d_m}$



# Transformer Decoder's Parameters

## Embedding Layer

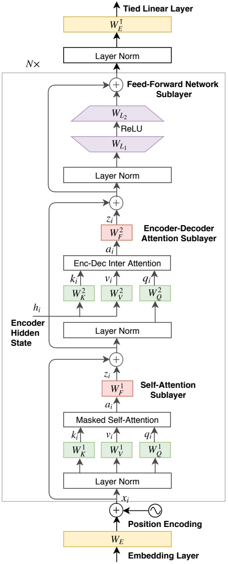
- $\boldsymbol{W}_E \in \mathbb{R}^{d_m \times V}$

## Masked Self-Attention

- $\boldsymbol{W}^1_K, \boldsymbol{W}^1_V, \boldsymbol{W}^1_Q, \boldsymbol{W}^1_F \in \mathbb{R}^{d_m \times d_m}$

## Encoder-Decoder Attention

- $\boldsymbol{W}^2_K, \boldsymbol{W}^2_V, \boldsymbol{W}^2_Q, \boldsymbol{W}^2_F \in \mathbb{R}^{d_m \times d_m}$



# Transformer Decoder's Parameters

## Embedding Layer

- $\boldsymbol{W}_E \in \mathbb{R}^{d_m \times V}$

## Masked Self-Attention

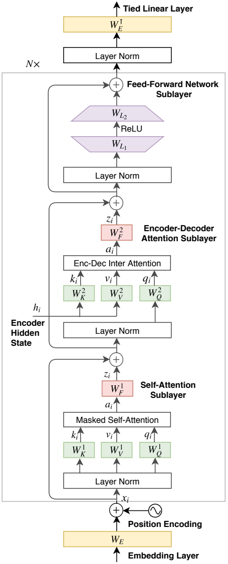
- $\boldsymbol{W}_K^1, \boldsymbol{W}_V^1, \boldsymbol{W}_Q^1, \boldsymbol{W}_F^1 \in \mathbb{R}^{d_m \times d_m}$

## Encoder-Decoder Attention

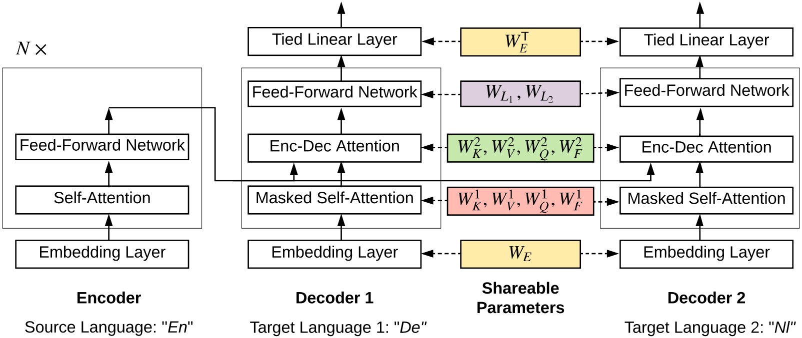
- $\boldsymbol{W}_K^2, \boldsymbol{W}_V^2, \boldsymbol{W}_Q^2, \boldsymbol{W}_F^2 \in \mathbb{R}^{d_m \times d_m}$

## Feed-Forward Network

- $\boldsymbol{W}_{L_1} \in \mathbb{R}^{d_m \times d_h}$
- $\boldsymbol{W}_{L_2} \in \mathbb{R}^{d_h \times d_m}$



# Parameter Sharing Strategies

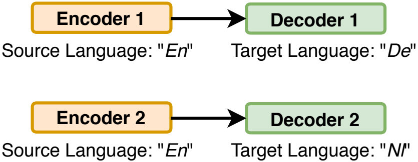



- Shareable parameters: embeddings, attention, embedding, linear layer weights.

# Parameter Sharing Strategies

- $\boldsymbol{\Theta}$ = set of shared parameters

# No Parameter Sharing

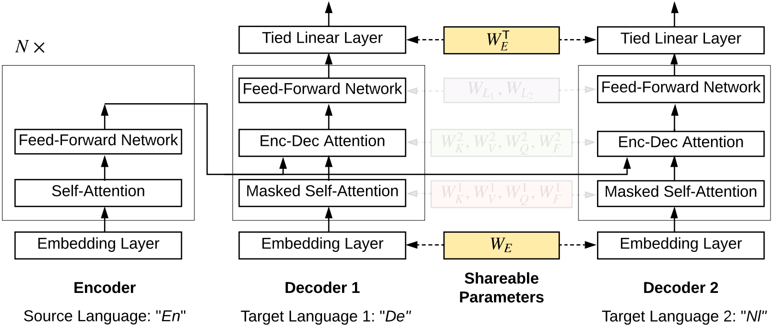



- Separate bilingual translation models
  $\boldsymbol{\Theta} = \emptyset$

# Embedding Sharing

- Common embedding layer
  $\boldsymbol{\Theta} = \{\boldsymbol{W}_E\}$

# +Encoder Sharing




- Common encoder and separate decoder for each target language
  $\Theta = \{W_E, \theta_{ENC}\}$

# +Decoder Sharing

- Next, include decoder parameters among the set of shared parameters.

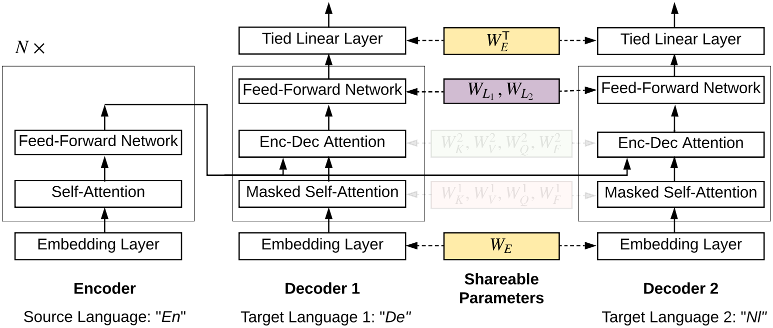
# +Decoder Sharing

- Next, include decoder parameters among the set of shared parameters.
- Exponentially many combinations possible: only select a subset.

# +Decoder Sharing

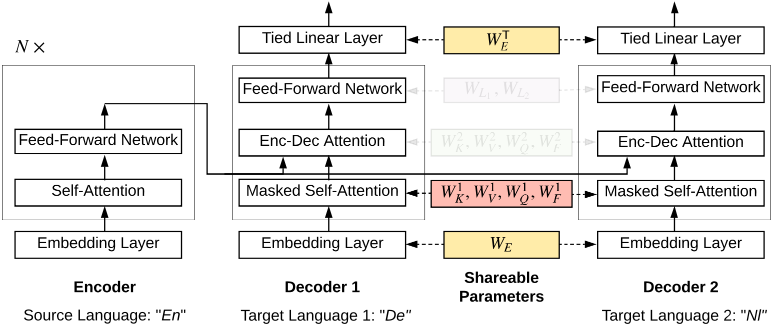
- Next, include decoder parameters among the set of shared parameters.
- Exponentially many combinations possible: only select a subset.
- The selected weights are shared in all layers.

# Parameter Sharing Strategies



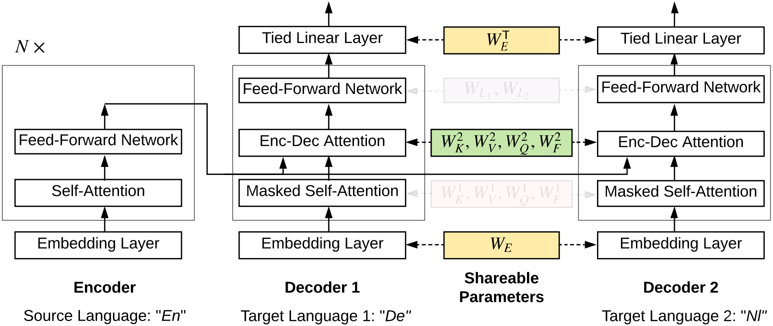

- FFN sublayer parameters are shared
  $\boldsymbol{\Theta} = \{\boldsymbol{W}_E, \boldsymbol{\theta}_{ENC}, \boldsymbol{W}_{L_1}, \boldsymbol{W}_{L_2}\}$

# Parameter Sharing Strategies



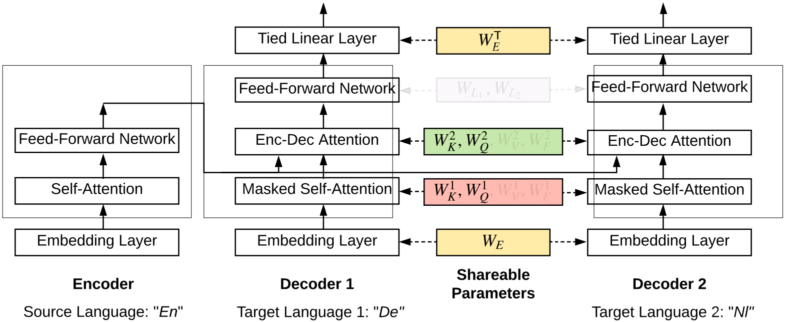

- Sharing the weights of the self-attention sublayer
  $\boldsymbol{\Theta} = \{\boldsymbol{W}_E, \boldsymbol{\theta}_{ENC}, \boldsymbol{W}_K^1, \boldsymbol{W}_Q^1, \boldsymbol{W}_V^1, \boldsymbol{W}_F^1\}$

# Parameter Sharing Strategies



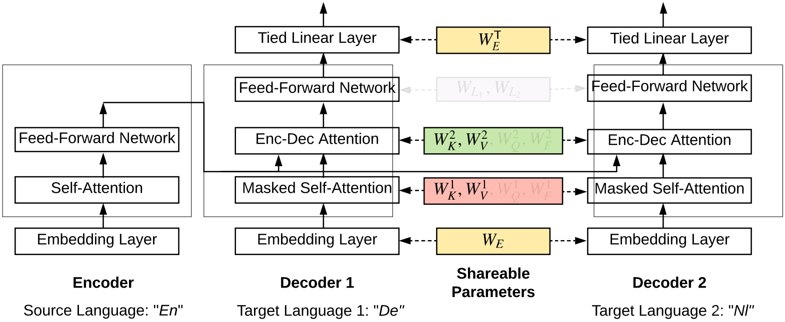

- Sharing the weights of the encoder-decoder attention sublayer
  $\boldsymbol{\Theta} = \{\boldsymbol{W}_E, \boldsymbol{\theta}_{ENC}, \boldsymbol{W}_K^{\boldsymbol{2}}, \boldsymbol{W}_Q^{\boldsymbol{2}}, \boldsymbol{W}_V^{\boldsymbol{2}}, \boldsymbol{W}_F^{\boldsymbol{2}}\}$

# Parameter Sharing Strategies



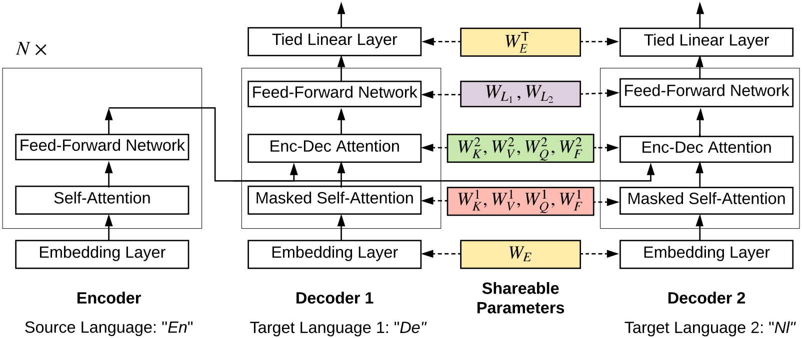

- Limit the attention weights to the key and query weights $\boldsymbol{\Theta} = \{\boldsymbol{W}_E, \boldsymbol{\theta}_{ENC}, \boldsymbol{W}_K^1, \boldsymbol{W}_Q^1, \boldsymbol{W}_K^2, \boldsymbol{W}_Q^2\}$

# Parameter Sharing Strategies




- Limit the attention weights to the key and value weights
  $\boldsymbol{\Theta} = \{\boldsymbol{W}_E, \boldsymbol{\theta}_{ENC}, \boldsymbol{W}_K^1, \boldsymbol{W}_V^1, \boldsymbol{W}_K^2, \boldsymbol{W}_V^2\}$

# Parameter Sharing Strategies




- Sharing all the decoder parameters to have a single unified model ($\Theta = \{\boldsymbol{W}_E, \boldsymbol{\theta}_{ENC}, \boldsymbol{\theta}_{DEC}\}$)

# Dataset

- Six language pairs from the TED talks dataset.[4]
  https://github.com/neulab/word-embeddings-for-nmt

[4] When and Why are Pre-trained Word Embeddings Useful for Neural Machine Translation?, NAACL 2018

# Dataset

- Six language pairs from the TED talks dataset.[4]
  https://github.com/neulab/word-embeddings-for-nmt
- Languages belong to different linguistic families

[4] When and Why are Pre-trained Word Embeddings Useful for Neural Machine Translation?, NAACL 2018

# Dataset

- Six language pairs from the TED talks dataset.[4]
  https://github.com/neulab/word-embeddings-for-nmt
- Languages belong to different linguistic families
  - Romanian (Ro) and French (Fr) are *Romance* languages

[4]When and Why are Pre-trained Word Embeddings Useful for Neural Machine Translation?, NAACL 2018

# Dataset

- Six language pairs from the TED talks dataset.[4]
  https://github.com/neulab/word-embeddings-for-nmt
- Languages belong to different linguistic families
  - Romanian (Ro) and French (Fr) are *Romance* languages
  - German (De) and Dutch (Nl) are *Germanic* languages

[4] When and Why are Pre-trained Word Embeddings Useful for Neural Machine Translation?, NAACL 2018

# Dataset

- Six language pairs from the TED talks dataset.[4]
  https://github.com/neulab/word-embeddings-for-nmt
- Languages belong to different linguistic families
  - Romanian (Ro) and French (Fr) are *Romance* languages
  - German (De) and Dutch (Nl) are *Germanic* languages
  - Turkish (Tr) and Japanese (Ja) are *unrelated languages*
    - Turkish: Turkic family
    - Japanese: Japonic family

[4]When and Why are Pre-trained Word Embeddings Useful for Neural Machine Translation?, NAACL 2018

# Multilingual Model Training Details

- Extra target language token at the start of source sentence.

# Multilingual Model Training Details

- Extra target language token at the start of source sentence.
- Trained using balanced mini-batches for every target language.

# Multilingual Model Training Details

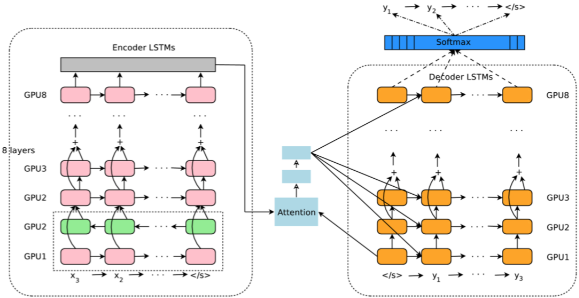
- Extra target language token at the start of source sentence.
- Trained using balanced mini-batches for every target language.
- Minimize weighted average cross-entropy loss.

# Multilingual Model Training Details

- Extra target language token at the start of source sentence.
- Trained using balanced mini-batches for every target language.
- Minimize weighted average cross-entropy loss.
  - Weighting term is proportional to word count in target languages.

# Results

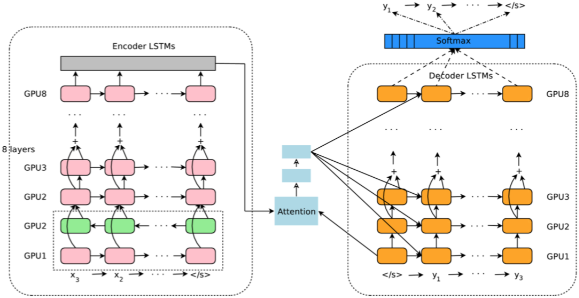
## Baselines




- **GNMT Model**: Based on recurrent LSTMs, residual connections, attention

# Results

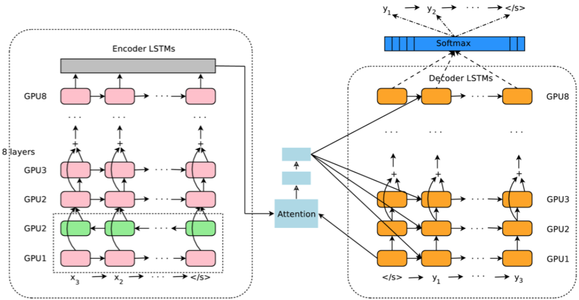
## Baselines




- **GNMT Model**: Based on recurrent LSTMs, residual connections, attention
    1. **GNMT NS**: No Sharing

# Results

## Baselines




- **GNMT Model**: Based on recurrent LSTMs, residual connections, attention
  1. **GNMT NS**: No Sharing
  2. **GNMT FS**: Full Sharing

# Results

## Baselines

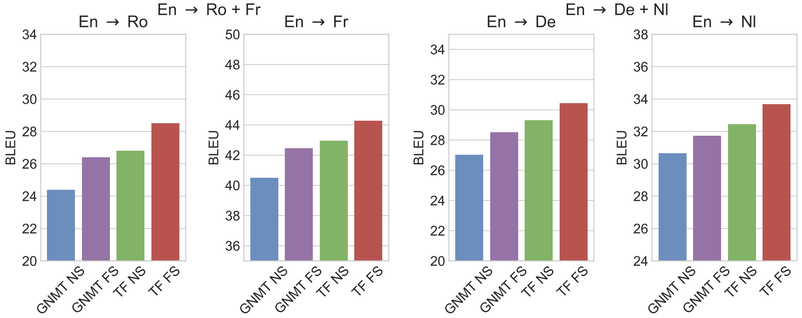
- **Transformer NS**: Separate models for each language pair

# Results

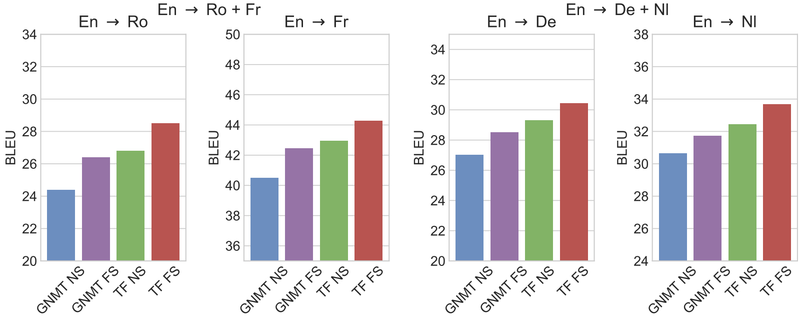
### Baselines

- **Transformer NS**: Separate models for each language pair
- **Transformer FS**: One model for all language pairs

# Results: Target languages are from the same family



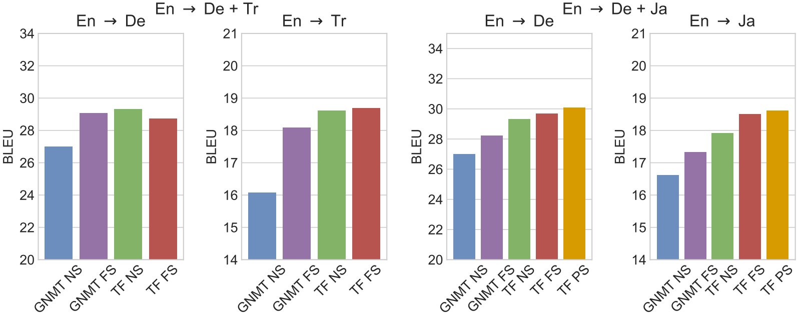
# Results: Target languages are from the same family



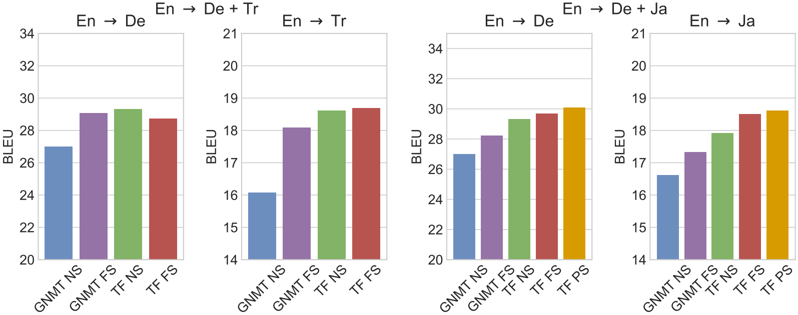

BLEU Scores

- **GNMT NS** $\ll$ **GNMT FS** $<$ **TF NS** $\ll$ **TF FS**

# Results: Target languages are from different families



# Results: Target languages are from different families

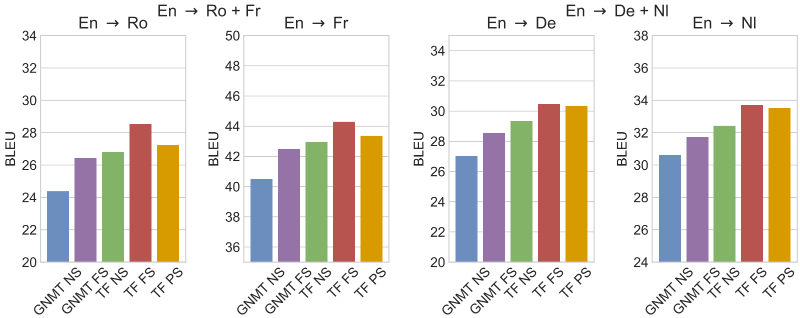



BLEU Scores

- **GNMT NS** $\ll$ **GNMT FS** $<\approx$ **TF NS**
- TF NS $\geq$ TF FS for En $\rightarrow$ De $+$ Tr
- TF NS $\approx$ TF FS for En $\rightarrow$ De $+$ Ja

# Results: Target languages are from the same family

## Transformer Partial Sharing: $\Theta = \{W_E\}$

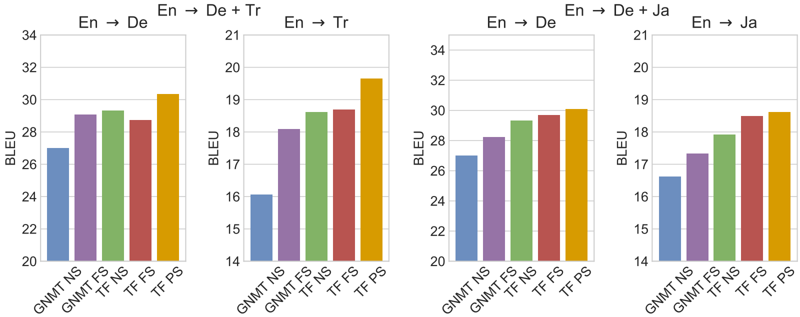



BLEU Scores:

- **TF FS $>$ TF PS for En $\rightarrow$ Ro + Fr**
- **TF FS $\approx$ TF PS for En $\rightarrow$ De + Nl**

# Results: Target languages are from different families

## Transformer Partial Sharing: $\boldsymbol{\Theta} = \{\boldsymbol{W}_E\}$

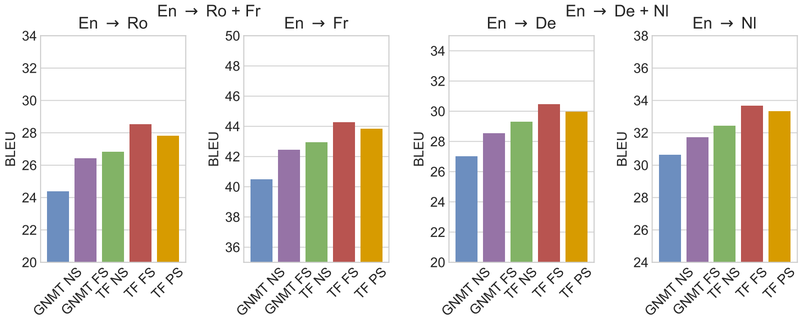



BLEU Scores

- **TF FS $<$ TF PS for En $\rightarrow$ De + Tr**
- **TF FS $\approx$ TF PS for En $\rightarrow$ De + Ja**

# Results: Target languages are from the same family

## Transformer Partial Sharing: $\boldsymbol{\Theta} = \{\boldsymbol{W}_E\} + \{\boldsymbol{\theta}_{ENC}\}$

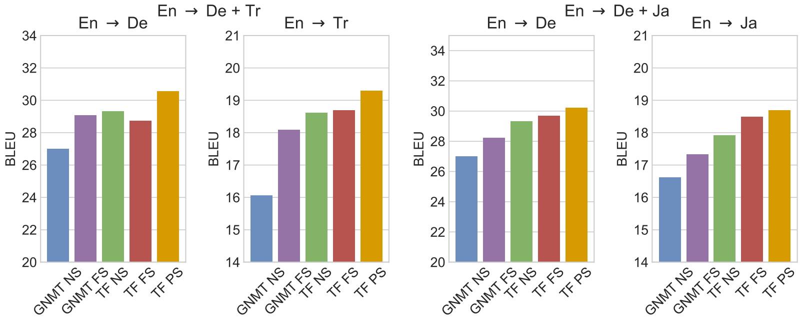



BLEU Scores:

- **TF FS $>$ TF PS for En $\rightarrow$ Ro + Fr and En $\rightarrow$ De + Nl**

# Results: Target languages are from different families

## Transformer Partial Sharing: $\boldsymbol{\Theta} = \{\boldsymbol{W}_E\} + \{\boldsymbol{\theta}_{ENC}\}$

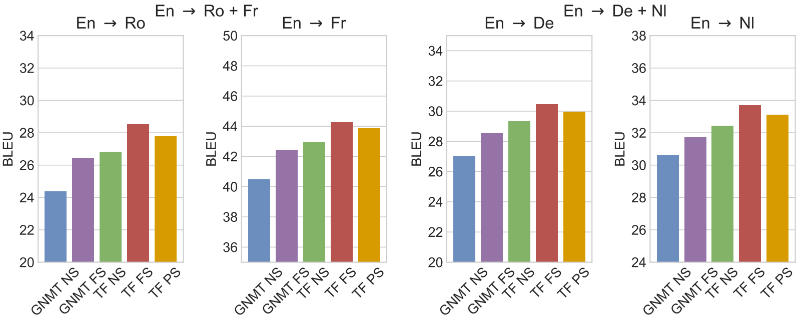



BLEU Scores:

- **TF FS $<$ TF PS for En $\rightarrow$ De $+$ Tr**
- **TF FS $\approx$ TF PS for En $\rightarrow$ De $+$ Ja**

# Results: Target languages are from the same family

Transformer Partial Sharing:
$\boldsymbol{\Theta} = \{\boldsymbol{W}_E, \boldsymbol{\theta}_{ENC}\} + \{\boldsymbol{W}_{L_1}, \boldsymbol{W}_{L_2}\}$



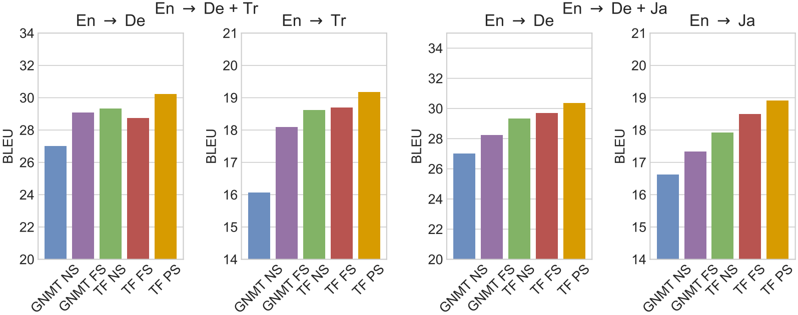

BLEU Scores:

- **TF FS $>$ TF PS for En $\rightarrow$ Ro $+$ Fr and En $\rightarrow$ De $+$ Nl**

# Results: Target languages are from different families

## Transformer Partial Sharing:

$\boldsymbol{\Theta} = \{\boldsymbol{W}_E, \boldsymbol{\theta}_{ENC}\} + \{\boldsymbol{W}_{L_1}, \boldsymbol{W}_{L_2}\}$



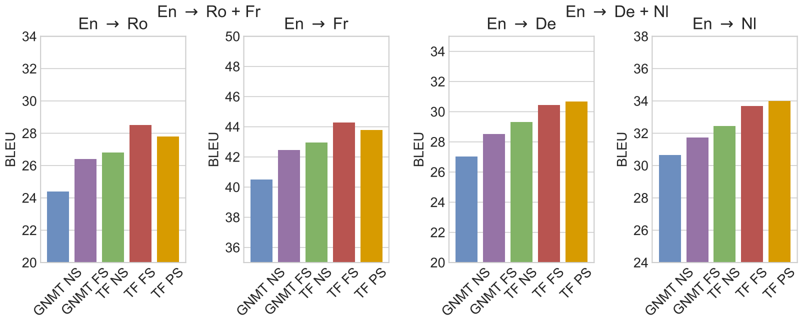

BLEU Scores:

- **TF FS $<$ TF PS for En $\rightarrow$ De + Tr and En $\rightarrow$ De + Ja**

# Results: Target languages are from the same family

## Transformer Partial Sharing:

$$\Theta = \{\boldsymbol{W}_E, \boldsymbol{\theta}_{ENC}\} + \{\boldsymbol{W}_K^1, \boldsymbol{W}_Q^1, \boldsymbol{W}_V^1, \boldsymbol{W}_F^1\}$$



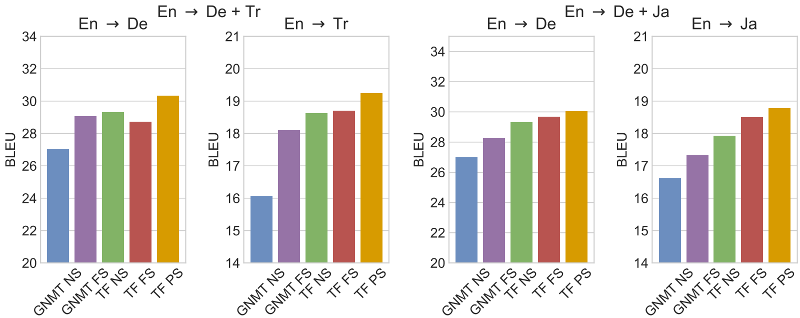

BLEU Scores:

- **TF FS $>$ TF PS for En $\rightarrow$ Ro $+$ Fr**
- **TF FS $\approx$ TF PS for En $\rightarrow$ De $+$ Nl**

# Results: Target languages are from different families

Transformer Partial Sharing:

$$\Theta = \{\boldsymbol{W}_E, \boldsymbol{\theta}_{ENC}\} + \{\boldsymbol{W}_K^1, \boldsymbol{W}_Q^1, \boldsymbol{W}_V^1, \boldsymbol{W}_F^1\}$$

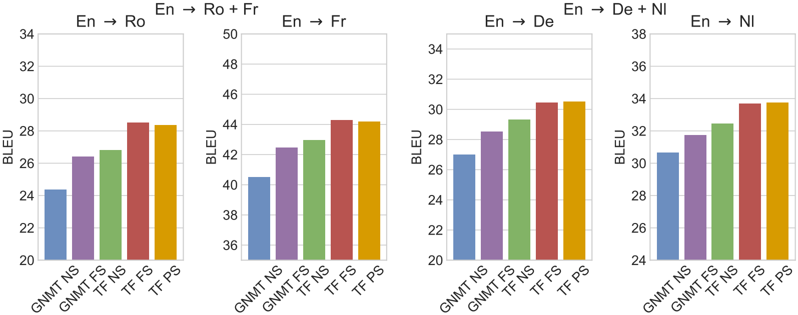



BLEU Scores:

- **TF FS $<$ TF PS for En $\rightarrow$ De + Tr**
- **TF FS $\approx$ TF PS for En $\rightarrow$ De + Ja**

# Results: Target languages are from the same family

Transformer Partial Sharing:
$$\boldsymbol{\Theta} = \{\boldsymbol{W}_E, \boldsymbol{\theta}_{ENC}\} + \{\boldsymbol{W}_K^{\mathbf{2}}, \boldsymbol{W}_Q^{\mathbf{2}}, \boldsymbol{W}_V^{\mathbf{2}}, \boldsymbol{W}_F^{\mathbf{2}}\}$$

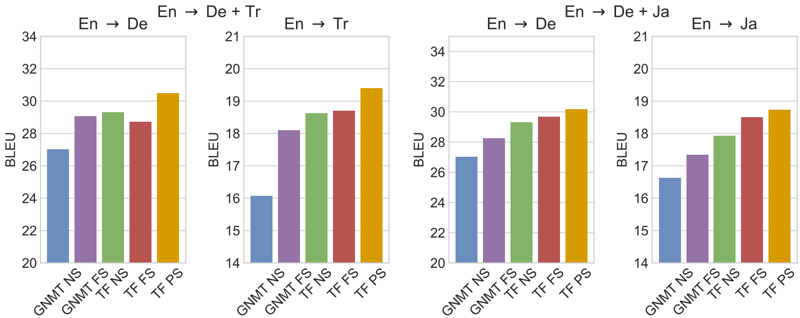



BLEU Scores:

- **TF FS $\approx$ TF PS for En $\rightarrow$ Ro + Fr and En $\rightarrow$ De + Nl**

# Results: Target languages are from different families

Transformer Partial Sharing:
$\boldsymbol{\Theta} = \{\boldsymbol{W}_E, \boldsymbol{\theta}_{ENC}\} + \{\boldsymbol{W}^{\mathbf{2}}_K, \boldsymbol{W}^{\mathbf{2}}_Q, \boldsymbol{W}^{\mathbf{2}}_V, \boldsymbol{W}^{\mathbf{2}}_F\}$



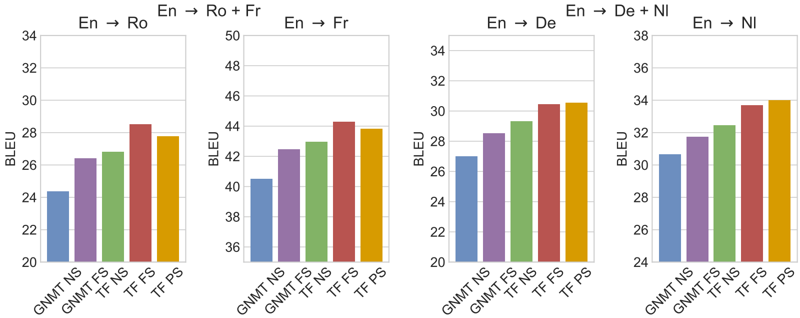

BLEU Scores:

- **TF FS $<$ TF PS for En $\rightarrow$ De + Tr**
- **TF FS $\approx$ TF PS for En $\rightarrow$ De + Ja**

# Results: Target languages are from the same family

## Transformer Partial Sharing:

$$\boldsymbol{\Theta} = \{\boldsymbol{W}_E,\, \boldsymbol{\theta}_{ENC}\} + \{\boldsymbol{W}_K^{\boldsymbol{1}},\, \boldsymbol{W}_V^{\boldsymbol{1}},\, \boldsymbol{W}_K^{\boldsymbol{2}},\, \boldsymbol{W}_V^{\boldsymbol{2}}\}$$



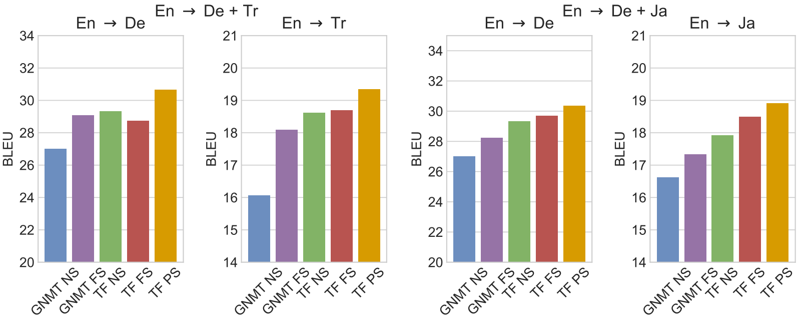

BLEU Scores:

- **TF FS $>$ TF PS for En $\rightarrow$ Ro + Fr**
- **TF FS $\approx$ TF PS for En $\rightarrow$ De + Nl**

# Results: Target languages are from different families

## Transformer Partial Sharing:

$$\boldsymbol{\Theta} = \{\boldsymbol{W}_E, \boldsymbol{\theta}_{ENC}\} + \{\boldsymbol{W}_K^1, \boldsymbol{W}_V^1, \boldsymbol{W}_K^2, \boldsymbol{W}_V^2\}$$

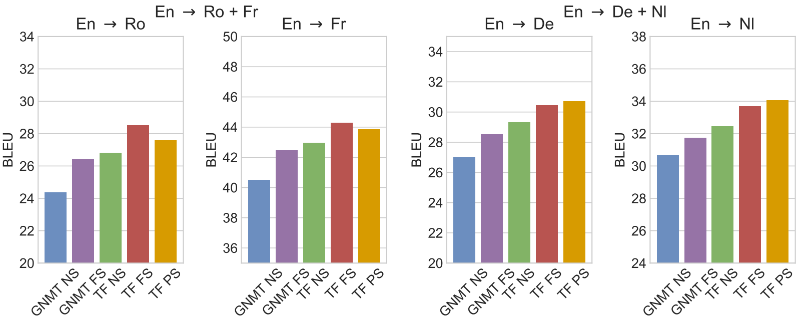



BLEU Scores:

- **TF FS $<$ TF PS for En $\rightarrow$ De + Tr and En $\rightarrow$ De + Ja**

# Results: Target languages are from the same family

Transformer Partial Sharing:
$\boldsymbol{\Theta} = \{\boldsymbol{W}_E, \boldsymbol{\theta}_{ENC}\} + \{\boldsymbol{W}^{\mathbf{1}}_K, \boldsymbol{W}^{\mathbf{1}}_Q, \boldsymbol{W}^{\mathbf{2}}_K, \boldsymbol{W}^{\mathbf{2}}_Q\}$

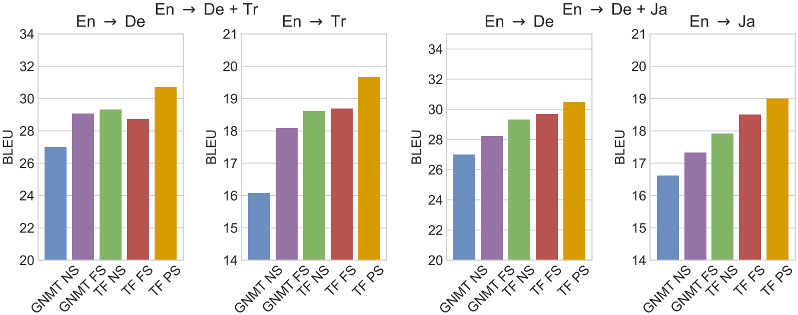



BLEU Scores:

- **TF FS ≈ TF PS for En → Ro + Fr and En → De + Nl**

# Results: Target languages are from different families

Transformer Partial Sharing:
$\boldsymbol{\Theta} = \{\boldsymbol{W}_E, \boldsymbol{\theta}_{ENC}\} + \{\boldsymbol{W}_K^1, \boldsymbol{W}_Q^1, \boldsymbol{W}_K^2, \boldsymbol{W}_Q^2\}$




BLEU Scores:

- **TF FS $\ll$ TF PS for En $\rightarrow$ De + Tr and En $\rightarrow$ De + Ja**

# Results: Target languages are from the same family

- Sharing all parameters leads to the best BLEU scores for En→Ro+Fr

# Results: Target languages are from the same family

- Sharing all parameters leads to the best BLEU scores for $\textsc{En}\rightarrow\textsc{Ro}+\textsc{Fr}$
- Sharing only the key, query from both the decoder attention layers leads to the best BLEU scores for $\textsc{En}\rightarrow\textsc{De}+\textsc{Nl}$

# Results: Target languages are from distant families

- Sharing all the parameters leads to a noticeable drop in the BLEU scores for both the considered language pairs.

# Results: Target languages are from distant families

- Sharing all the parameters leads to a noticeable drop in the BLEU scores for both the considered language pairs.
- Sharing the key, query parameters results in a large increase in the BLEU scores.

# Conclusions

- We explore parameter sharing strategies for multilingual translation using self-attentional models.

# Conclusions

- We explore parameter sharing strategies for multilingual translation using self-attentional models.
- We examine the case when the target languages come from the same or distant language families.

# Conclusions

- We explore parameter sharing strategies for multilingual translation using self-attentional models.
- We examine the case when the target languages come from the same or distant language families.
- The popular approach of full parameter sharing may perform well only when the target languages belong to the same family.

# Conclusions

- We explore parameter sharing strategies for multilingual translation using self-attentional models.
- We examine the case when the target languages come from the same or distant language families.
- The popular approach of full parameter sharing may perform well only when the target languages belong to the same family.
- Partial parameter sharing of embedding, encoder, decoder's key, query weights is applicable to all kinds of language pairs.

# Conclusions

- We explore parameter sharing strategies for multilingual translation using self-attentional models.
- We examine the case when the target languages come from the same or distant language families.
- The popular approach of full parameter sharing may perform well only when the target languages belong to the same family.
- Partial parameter sharing of embedding, encoder, decoder's key, query weights is applicable to all kinds of language pairs.
- Partial parameter sharing achieves the best BLEU scores when the target languages are from distant families.

# Conclusions

- We explore parameter sharing strategies for multilingual translation using self-attentional models.
- We examine the case when the target languages come from the same or distant language families.
- The popular approach of full parameter sharing may perform well only when the target languages belong to the same family.
- Partial parameter sharing of embedding, encoder, decoder's key, query weights is applicable to all kinds of language pairs.
- Partial parameter sharing achieves the best BLEU scores when the target languages are from distant families.

Code: https://github.com/DevSinghSachan/multilingual_nmt

# Conclusions

- We explore parameter sharing strategies for multilingual translation using self-attentional models.
- We examine the case when the target languages come from the same or distant language families.
- The popular approach of full parameter sharing may perform well only when the target languages belong to the same family.
- Partial parameter sharing of embedding, encoder, decoder's key, query weights is applicable to all kinds of language pairs.
- Partial parameter sharing achieves the best BLEU scores when the target languages are from distant families.

Code: https://github.com/DevSinghSachan/multilingual_nmt

**Thank you! Questions?**