MAD-X: An Adapter-based Framework for Multi-task Cross-lingual Transfer

Jonas Pfeiffer, Ivan Vulic, Iryna Gurevych, Sebastian Ruder

Overview

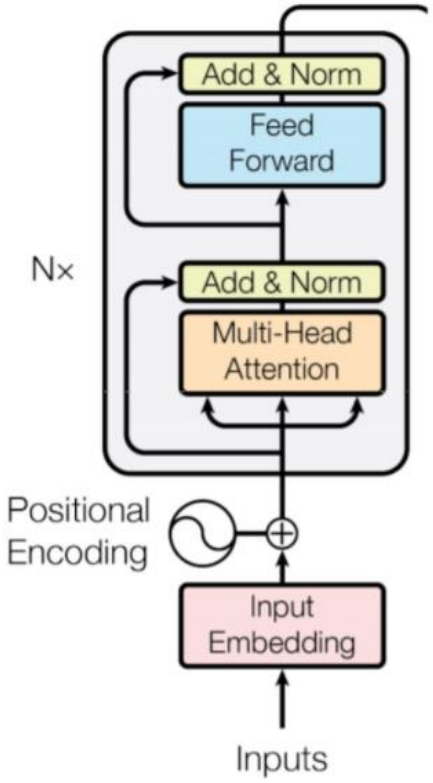
- What is an Adapter?
- MAD-X: Multiple ADapters for Cross-lingual transfer .

What is an Adapter?

- The usual practice for transfer learning is to fine-tune all weights of the pretrained model on the target task.
- Adapters (Houlsby et al., 2019) have been introduced as an alternative lightweight fine-tuning strategy that achieves on-par performance to full fine-tuning.
- Adapters consist of a small set of additional newly initialized weights at every layer of the transformer. Note that, different adapters are usually used for different layers.
- These weights are then trained during fine-tuning, while the pre-trained parameters of the large model are kept frozen/fixed.

What is an Adapter?

- Transformer encoder architecture:

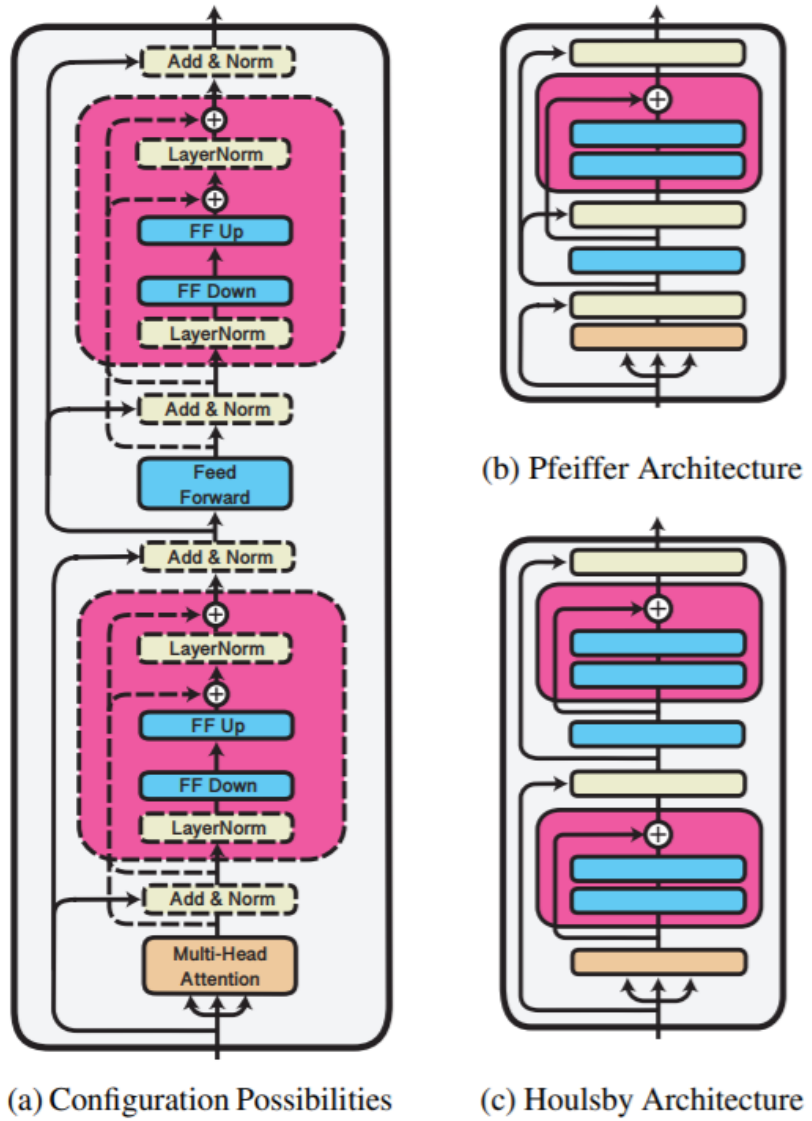



```
x_encoder = word_embeds + pos_embeds
for layer_i in range(N):
  q, k, v = x_encoder, x_encoder, x_encoder
  att = MultiHeadAttention(q, k, v)
  att = LayerNorm(x_encoder + Dropout(att))
  ffw = Feedfw(att)
  x_encoder = LayerNorm(att + dropout(ffw))
```

Figure: Pseudo codes of Transformer Encoder.

What is an Adapter?

- The placement and architecture of adapter parameters within a pre-trained model is non-trivial and may impact their efficacy.
- Different works on adapters agree on an architecture of a two-layer feed-forward (i.e., down and up projection) neural network with a bottleneck.



MAD-X: Multiple ADapters for Cross-lingual transfer

- Problem with current multilingual pretrained models (MPMs):

  + Cannot represent a shared vocabulary for ALL languages (~ 7,000 languages).

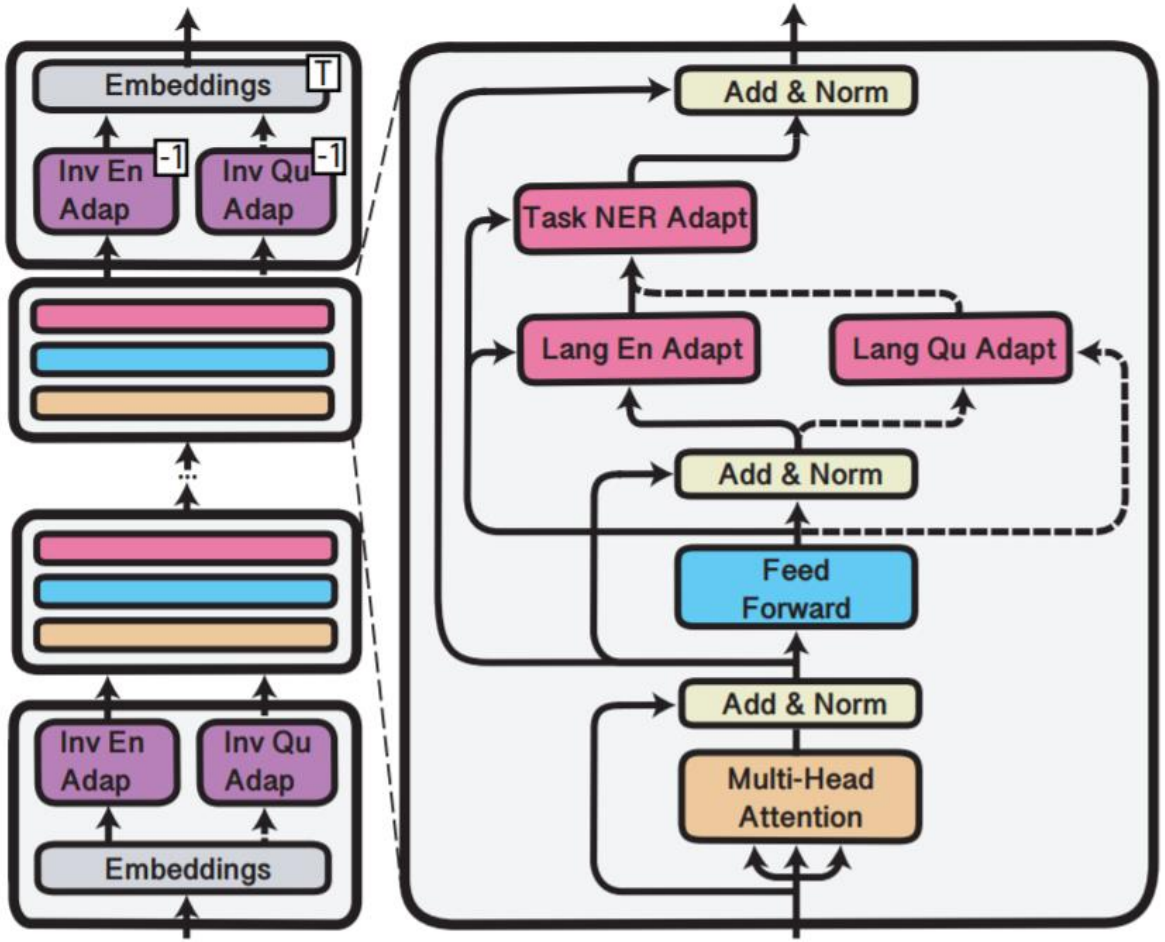
  + Low-resource languages are less focused by the model (model capacity is finite).

MAD-X: Multiple ADapters for Cross-lingual transfer

- Goal of MAD-X:
  + High portability to LOW-RESOURCE or UNSEEN languages.
    => Language Adapters and Invertible Adapters: is an alternate for finetuning a MPM toward a specific target language.
  + Efficient fine-tuning on downstream tasks.
    => Task Adapters: is an alternate for finetuning a MPM toward a specific task.

MAD-X: Multiple ADapters for Cross-lingual transfer

- Overall of MAD-X's architecture:



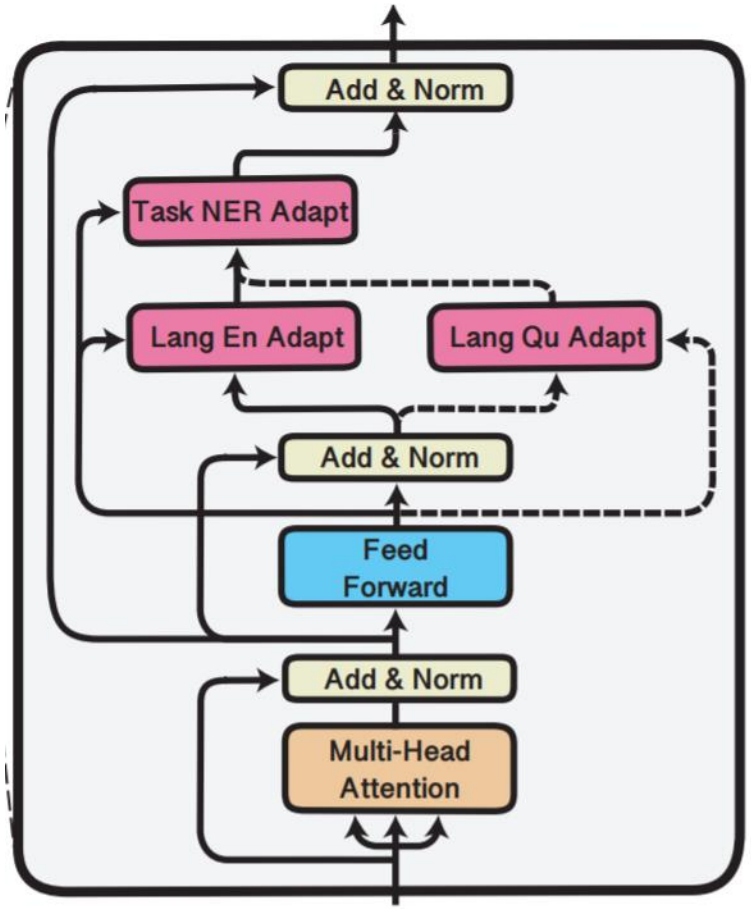
Language Adapters (LAs)

- Language Adapter at layer l:

$$\mathrm{LA}_l(\mathbf{h}_l, \mathbf{r}_l) = \mathbf{U}_l(\mathrm{ReLU}(\mathbf{D}_l(\mathbf{h}_l))) + \mathbf{r}_l$$

Where:

+ $\mathbf{h}_l$ is the Transformer hidden state at layer l.
+ $\mathbf{r}_l$ is the residual at layer l.



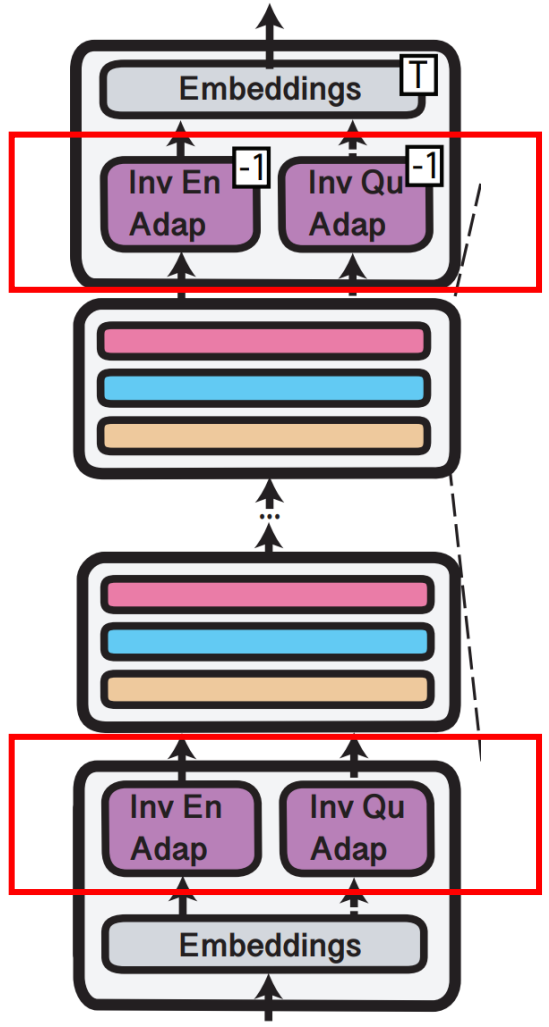
Invertible Adapters (IAs)

- Invertible Adapters' motivation:

  + Using language adapters only cannot affect the first embedding layer.

  + For unseen languages, embeddings taken from the first embedding layer are from other languages.

- Invertible Adapters are injected at the beginning and the end of the MPM.



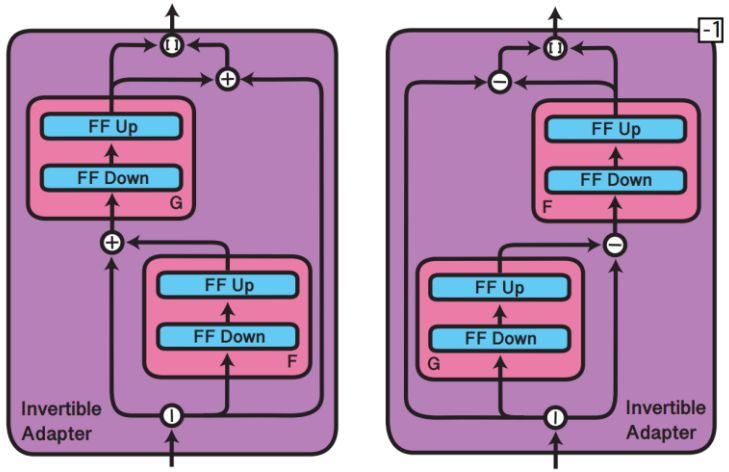
Invertible Adapters (IAs)

- Invertible Adapter is a non-linear transformation whose inverse can be trivially obtained via its special design. Its design is inherited from NICE (Dinh et al., 2015).
- Given an embedding vector $\mathbf{e}$, we first split it into two equal vectors: $\mathbf{e} = [\mathbf{e}_1, \mathbf{e}_2]$
- Forward Pass:

$$\mathbf{o}_1 = F(\mathbf{e}_2) + \mathbf{e}_1$$
$$\mathbf{o}_2 = G(\mathbf{o}_1) + \mathbf{e}_2$$
$$\mathbf{o} = [\mathbf{o}_1, \mathbf{o}_2]$$

- Inverted Pass:

$$\mathbf{e}_2 = \mathbf{o}_2 - G(\mathbf{o}_1)$$
$$\mathbf{e}_1 = \mathbf{o}_1 - F(\mathbf{e}_2)$$
$$\mathbf{e} = [\mathbf{e}_1, \mathbf{e}_2].$$



(a) The invertible adapter (b) The inversed adapter

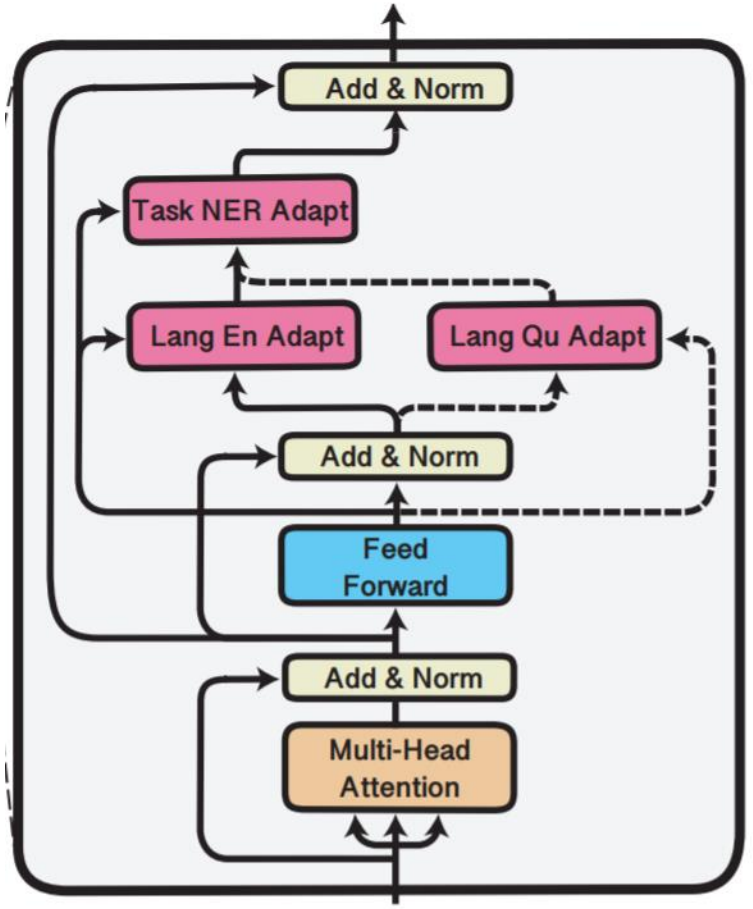
- They assume that $[\mathbf{o}_1, \mathbf{o}_2]$ don't change much as they are tied via the multilingual pretrained model.

Training of LAs and IAs

- Language Adapters and Invertible Adapters are updated during the training with the masked language modeling (MLM) task on unlabeled monolingual data of the language of interest.
- Two sets of LAs and IAs are trained separately for each language (source and target language).

Task Adapters

- Task adapters have the same architecture as language adapters.
- Task adapters are stacked on top of language adapters.
- Different task adapters are used for different layers.
- During the training on a target task, only task adapters are updated while all other components (MPM, LAs, IAs) are fixed.
- During the training on the target task, task adapters are stacked on top of trained LAs of the source language.
- During the testing on target task, trained LAs and IAs of the source language are replaced with trained LAs and IAs of the target language.



Results

- Experiments are conducted with 16 languages which are selected based on:
  + variance in data availability.
  + whether data in the particular language was included in the pretraining data.
  + typological diversity to ensure that different language types and families are covered.

Results

- MAD-X gains improvement on most of the languages for NER.

Model	en	ja	zh	ar	jv	sw	is	my	qu	cdo	ilo	xmf	mi	mhr	tk	gn	avg
XLM-R	44.2	38.2	40.4	36.4	37.4	42.8	47.1	**26.3**	27.4	18.1	28.8	**35.0**	16.7	**31.7**	20.6	**31.2**	32.6
XLM-R MLM-SRC	39.5	45.2	34.7	17.7	34.5	35.3	43.1	20.8	26.6	21.4	28.7	22.4	18.1	25.0	27.6	24.0	29.0
XLM-R MLM-TRG	54.8	**47.4**	**54.7**	51.1	38.7	48.1	53.0	20.0	29.3	16.6	27.4	24.7	15.9	26.4	26.5	28.5	35.2
MAD-X – LAD – INV	44.5	38.6	40.6	42.8	32.4	43.1	48.6	23.9	22.0	10.6	23.9	27.9	13.2	24.6	18.8	21.9	29.8
MAD-X – INV	52.3	46.0	46.2	56.3	**41.6**	48.6	52.4	23.2	32.4	**27.2**	30.8	33.0	**23.5**	29.3	30.4	28.4	37.6
MAD-X	**55.0**	46.7	47.3	**58.2**	39.2	**50.4**	**54.5**	24.9	**32.6**	24.2	**33.8**	34.3	16.8	**31.7**	**31.9**	30.4	**38.2**

Table 2: NER F1 scores averaged over all 16 source languages when transferring to each target language (i.e., the columns refer to target languages). The vertical dashed line distinguishes between languages seen in multilingual pretraining and the unseen ones (see also Table 1).

Results

- Unfortunately, the results on the most important setting are not that promising.

	en	ja	zh	ar	jv	sw	is	my	qu	cdo	ilo	xmf	mi	mhr	tk	gn	avg
mBERT	84.8	**26.7**	**38.5**	38.7	**57.8**	66.0	65.7	42.9	54.9	14.20	63.5	31.1	21.8	46.0	47.2	45.4	44.0
XLM-R	83.0	15.2	19.6	41.3	56.1	63.5	67.2	46.9	58.3	20.47	61.3	32.2	15.9	41.8	43.4	41.0	41.6
XLM-R MLM-SRC	84.2	8.45	11.0	27.3	44.8	57.9	59.0	35.6	52.5	21.4	60.3	22.7	22.7	38.1	44.0	41.7	36.5
XLM-R MLM-TRG	84.2	9.30	15.5	**44.5**	50.2	**77.7**	71.7	**55.5**	**68.7**	**47.6**	**84.7**	**60.3**	43.6	56.3	56.4	50.6	52.8
MAD-X – LAD – inv	82.0	15.6	20.3	41.0	54.4	66.4	67.8	48.8	57.8	16.9	59.9	36.9	14.3	44.3	41.9	42.9	41.9
MAD-X – INV	82.2	16.8	20.7	36.9	54.1	68.7	71.5	50.0	59.6	39.2	69.9	54.9	48.3	58.1	53.1	52.8	50.3
MAD-X	82.3	19.0	20.5	41.8	55.7	73.8	**74.5**	51.9	66.1	36.5	73.1	57.6	**51.0**	**62.1**	**59.7**	**55.1**	**53.2**

Table 5: NER F1 scores for zero-shot transfer from English.

References

- NICE: NON-LINEAR INDEPENDENT COMPONENTS ESTIMATION. (Dinh et al., 2015).
- Parameter-Efficient Transfer Learning for NLP. (Houlsby et al., 2019).
- AdapterHub: A Framework for Adapting Transformers. (Pfeiffer et al., 2020).
- MAD-X: An Adapter-based Framework for Multi-task Cross-lingual Transfer (Pfeiffer et al., 2020)