

MetaXL: Meta Representation Transformation for Low-resource Cross-lingual Learning

Mengzhou Xia, Guoqing Zheng, Subhabrata Mukherjee, Milad Shokouhi,
Graham Neubig, Ahmed Hassan Awadallah
NAACL 2021

Motivation

- Extremely low resource languages are extremely challenging
 - No large-scale monolingual corpora for pretraining
 - No sufficient annotated data for finetuning
- Multilingual representations are disjoint across languages .

This paper proposes a meta-learning method that:

- Learn to transform representations from source language to the target language
- Bring their representations close to each other for effective transferring

Setting

Extremely low resource target language:

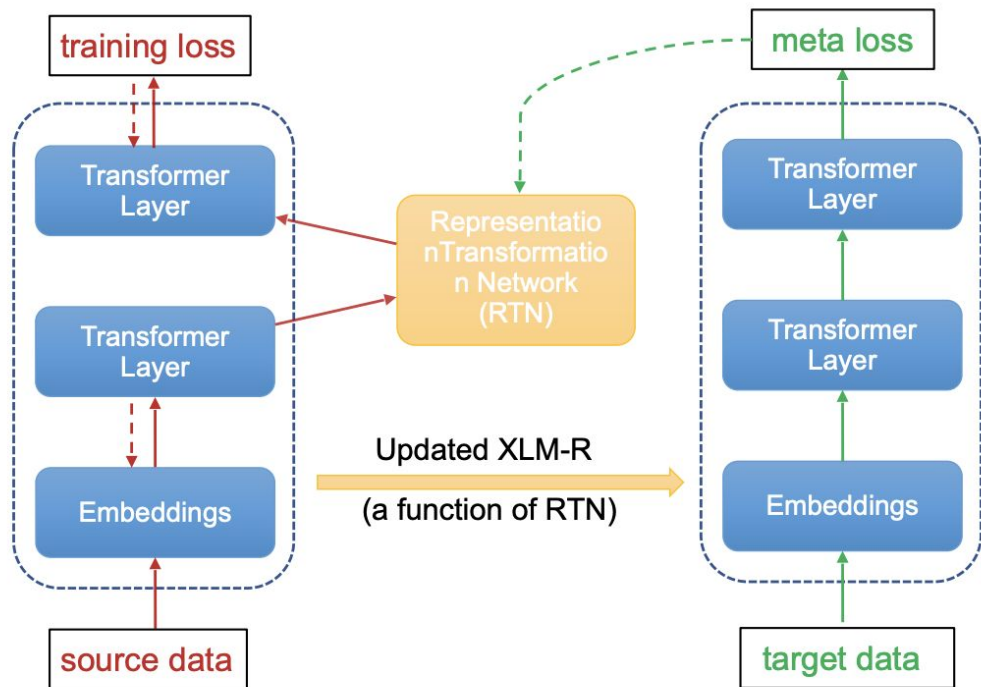
- Small annotated task-specific data
- The language is (under/not) covered by XLM-R

$$\mathcal{D}_t = \{(x_t^{(i)}, y_t^{(i)}); i \in [1, N]\}.$$





Source language

- Pretrained language model f_θ :
- Large annotated data $\mathcal{D}_s = \{(x_s^{(j)}, y_s^{(j)}); j \in [1, M]\}$ where $M \gg N$

Representation transformation network



Repeat following steps until convergence:

- ①  Forward pass for training loss
- ②  Backward pass from training loss and XLM-R update
(gradients dependency on RTN kept)
- ③  Forward pass for meta loss
- ④  Backward pass from meta loss and RTN update

Optimization

Quote:

“If the representation transformation network $g(\phi)$ effectively transforms the source language representations, such transformed representations $f(x_s; \phi, \theta)$ should be more beneficial to the target task than the original representations $f(x_s; \theta)$, such that the model achieves a smaller evaluation loss \mathcal{L}_{D_t} on the target language.”

$$\begin{aligned} & \min_{\phi} \mathcal{L}_{D_t} (f(x_t; \theta^*(\phi)), y_t) \\ \text{s.t. } & \theta^*(\phi) = \arg \min_{\theta} \mathcal{L}_{D_s} (f(x_s; \phi, \theta), y_s) \end{aligned}$$

Algorithm

Algorithm 1 Training procedure for MetaXL

Input: Input data from the target language D_t and the source language D_s

- 1: Initialize base model parameters θ with pretrained XLM-R weights, initialize parameters of the representation transformation network ϕ randomly
 - 2: **while** not converged **do**
 - 3: Sample a source batch (x_s, y_s) from D_s and a target batch (x_t, y_t) from D_t ;
 - 4: Update θ : $\theta^{(t+1)} = \theta^{(t)} - \alpha \nabla_{\theta} \mathcal{L}(x_s; \theta^{(t)}, \phi^{(t)})$
 - 5: Update ϕ : $\phi^{(t+1)} = \phi^{(t)} - \beta \nabla_{\phi} \mathcal{L}(x_t; \theta^{(t)} - \alpha \nabla_{\theta} \mathcal{L}(x_s; \theta^{(t)}, \phi^{(t)}))$
 - 6: **end while**
-

Named Entity Recognition

	Source	Method	qu	cdo	ilo	xmf	mhr	mi	tk	gn	average
(1)	-	target	57.14	37.72	61.32	59.07	55.17	76.27	55.56	48.89	56.39
(2)	English	JT	66.10	55.83	80.77	69.32	71.11	82.29	61.61	65.44	69.06
		MetaXL	68.67	55.97	77.57	73.73	68.16	88.56	66.99	69.37	71.13
(3)	Related	JT	79.65	53.91	78.87	79.67	66.96	87.86	64.49	70.54	72.74
		MetaXL	77.06	57.26	75.93	78.37	69.33	86.46	73.15	71.96	73.69

Table 2: F1 for NER across three settings where we, (1) only use the target language data; (2) use target language data along with 5k examples of English; (3) use the target language data along with 5k examples of a related language. JT stands for joint training and MetaXL stands for Meta Representation Transformation. We bold the numbers with a better average performance in each setting.

Language	Code	Language Family	Related Language
Quechua	qu	Quechua	Spanish
Min Dong	cdo	Sino-Tibetan	Chinese
Ilocano	ilo	Austronesian	Indonesian
Mingrelian	xmf	Kartvelian	Georgian
Meadow Mari	mhr	Uralic	Russian
Maori	mi	Austronesian	Indonesian
Turkmen	tk	Turkic	Turkish
Guarani	gn	Tupian	Spanish

Sentiment Analysis

	Method	tel	fa
(1)	target only	86.87	82.58
(2)	JT	88.68	85.51
	MetaXL	89.52	87.14

Table 3: F1 for sentiment analysis on two settings using (1) only the target language data; (2) target language data along with 1k examples of English.

Which layer to place RTN

Method	NER	SA	
	Average	tel	fa
JT	69.06	88.68	85.51
MetaXL L0	70.02	89.52	85.41
MetaXL L6	70.27	86.00	85.80
MetaXL L12	71.13	90.53	87.14
MetaXL L0,12	69.00	84.85	86.64

Table 5: F1 when placing the transfer component at different positions on XLM-R. Under this setting, we use 5k English data for NER and 1K English data for SA. L stands for layer.

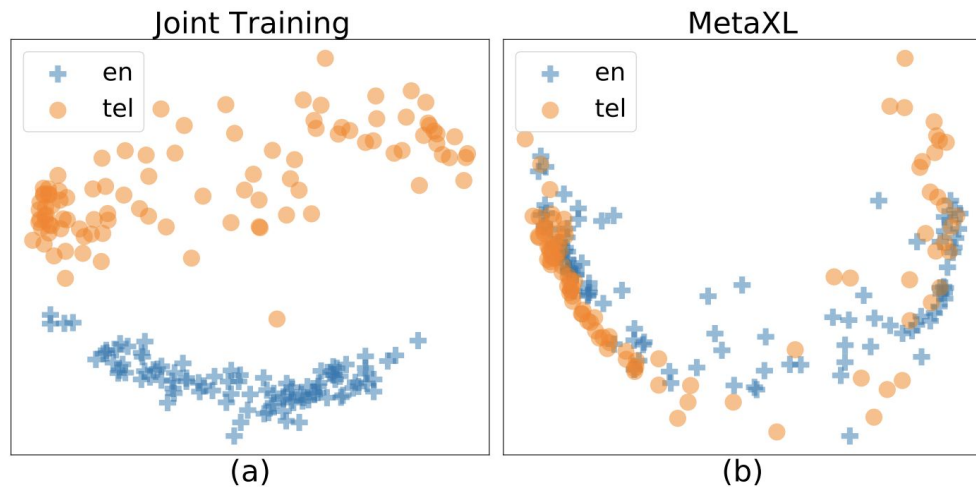


Figure 1: First two principal components of sequence representations (corresponding to [CLS] tokens) of Telugu and English examples from a jointly fine-tuned mBERT and a MetaXL model for the task of sentiment analysis. MetaXL pushes the source (EN) and target (TEL) representations closer to realize a more effective transfer. The Hausdorff distance between the source and target representations drops from 0.57 to 0.20 with F1 score improvement from 74.07 to 78.15.