How can we build NMT systems for language pairs with very little parallel data?

Maxim Ryabinin, David Sharafyan, Anna Klepova, Viktoria Chekalina, Igor Novikov

October 25, 2018

Unsupervised MT

Low-resource parallel data

- No parallel data (only monolingual corpora)
- Very little parallel data

Monolingual corpora: Neural Unsupervised Machine Translation

3 principles of unsupervised MT

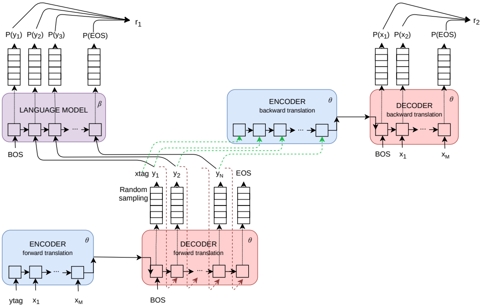
1. Initialization. The two distributions are roughly aligned, e.g. by performing word-by-word translation
2. Language modelling. Models train on monolingual data and express a data-driven prior about how sentences should read in each language.
   $L^{lm} = \mathbb{E}_{x \sim s}[-logP_{s \to s}(x|C(x)] + \mathbb{E}_{y \sim t}[-logP_{t \to t}(y|C(y)]$
   $C(x), C(y)$ - noised sentences
3. Back-translation
   $L^{bt} = \mathbb{E}_{x \sim s}[-logP_{s \to t}(y|u^*(y)] + \mathbb{E}_{y \sim t}[-logP_{t \to s}(x|v^*(x)]$
   $v^*(x), u^*(y)$ - sentences in target and sources languages respectively, obtained by Initialization

Very little data: Dual learning (reinforcement)

Corpus D_A, D_B; LM_A, LM_B - language models("environment"); θ_{BA}, θ_{AB} - translation models.
s - sentence on A, s_{mid} - middle translation

1. θ_{AB}, θ_{BA} - pretrained on little parallel data
2. $r_1 = LM_b(s_{mid}) = logP(s_{mid}|s, \theta_{AB})$ - how natural translated sentence is in B
3. $r_2 = logP(s|s_{mid}, \theta_{BA})$
4. $r = (1 - \alpha)r_1 + \alpha r_2$
5. use policy gradient methods

Zero-Shot Dual Machine Translation



	NMT-F	Dual-0	Dual-S
Aligned Data	en-fr (1M) en-es (1M) es-fr (1M)	en-fr (1M) en-es (1M)	en-fr (1M) en-es (1M) es-fr (10k)
Monol. Data		es (0.5M) fr (0.5M)	es (0.5M) fr (0.5M)
en→es	44.06	37.05	38.74
es→en	18.24	32.84	32.03
en→fr	34.75	29.58	30.89
fr→en	13.58	27.95	26.00
es→fr	37.67	35.54	35.63
fr→es	40.85	38.83	39.00

- Train NMT model on EN-FR and EN-ES parallel corpora
- Train EN and ES language models
- Fine-tune NMT model using language models
- Fine-tune NMT model on small ES-FR parallel corpus

Contextual parameter generation for Universal NMT

One encoder/decoder for language —
linear growth in number of parameters
Let's introduce language embeddings!

$$g^{enc}(l_s) \triangleq W^{enc} l_s, g^{dec}(l_t) \triangleq W^{dec} l_t,$$
$$W^{enc} \in \mathbb{R}^{P_{enc} \times M}, W^{dec} \in \mathbb{R}^{P_{dec} \times M},$$
$$l_s, l_t \in \mathbb{R}^M$$

	PNMT	GML	CPG*	CPG
En→De	25.99	15.92	26.41	**26.77**
De→En	30.93	29.60	31.24	**31.77**
En→Fr	38.25	34.40	38.10	**38.32**
Fr→En	37.40	35.14	37.11	**37.89**

Issues with setup: GML has fewer parameters, might be trained with auto-encoding as well

Main advantages

1. Number of NN parameters is constant wrt number of languages
2. Linguistic similarities are exploited
3. Possible to fine-tune embeddings only on low-resource data