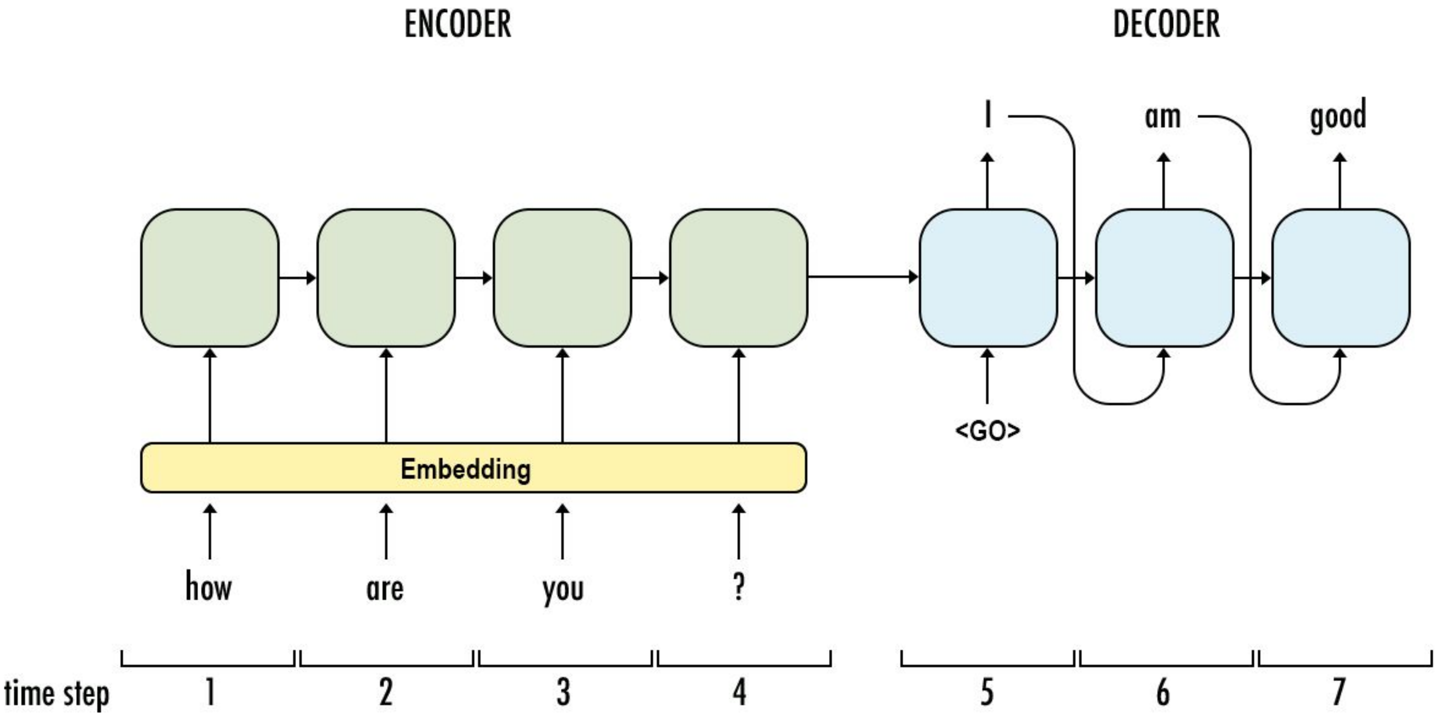
Unsupervised Machine Translation

Sequence to sequence



https://towardsdatascience.com/sequence-to-sequence-model-introduction-and-concepts-44d9b41cd42d

Common improvements

- Multi-layer RNNs (residual connections, bi-directional)
- Attention Mechanism
- Sub-word units
- Reinforcement Learning

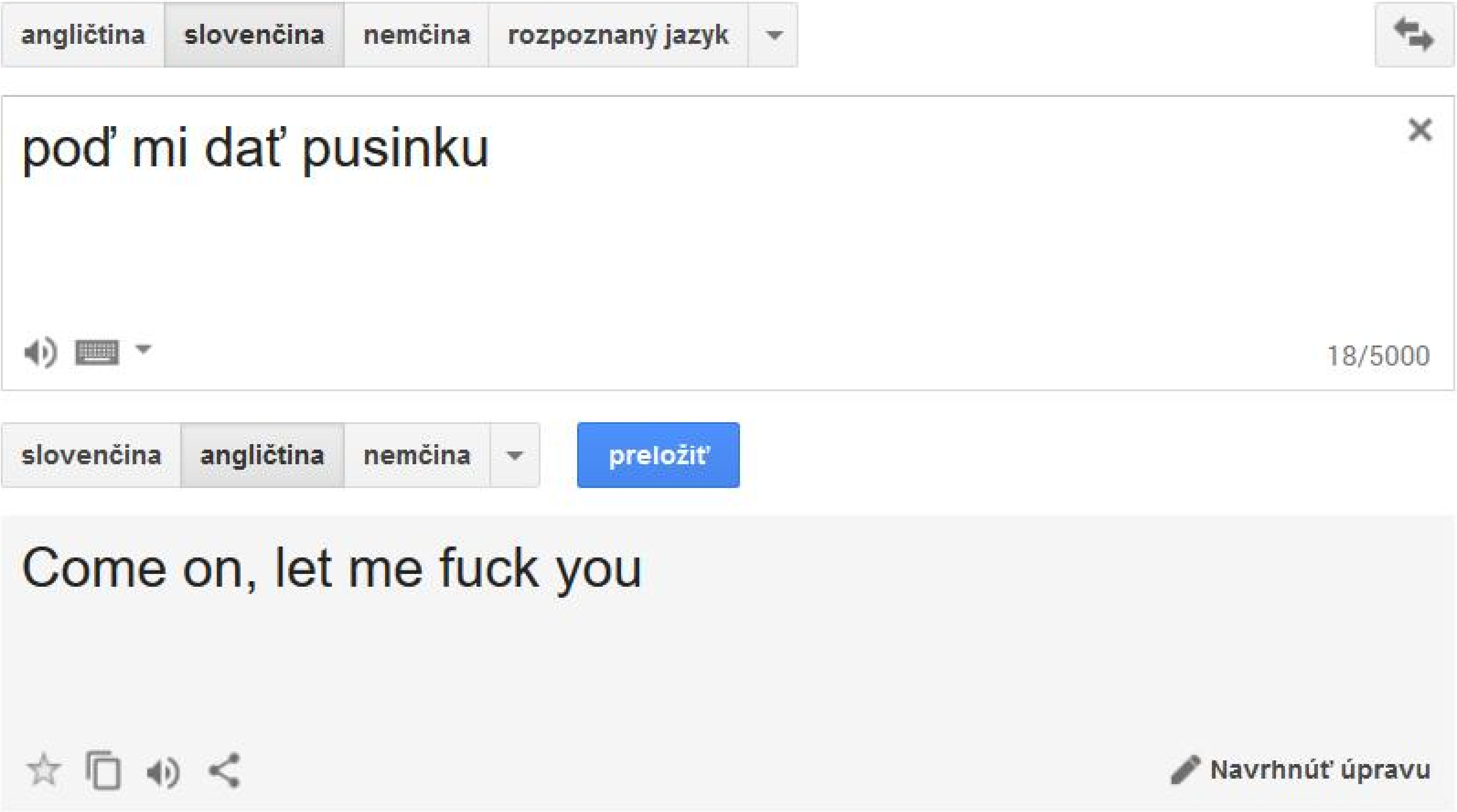
Data requirements are still huge, some solutions for low-resource languages:

Improving Neural Machine Translation Models with Monolingual Data [Sennrich 2016, ACL]

Unsupervised Neural Machine Translation [Artetxe 2018, ICLR]

Unsupervised Machine Translation Using Monolingual Corpora Only [Lample 2018, ICLR]

and many others


angličtina
slovenčina
nemčina
rozpoznaný jazyk
poď mi dať pusinku
18/5000
slovenčina
angličtina
nemčina
preložiť
Come on, let me fuck you
Navrhnúť úpravu

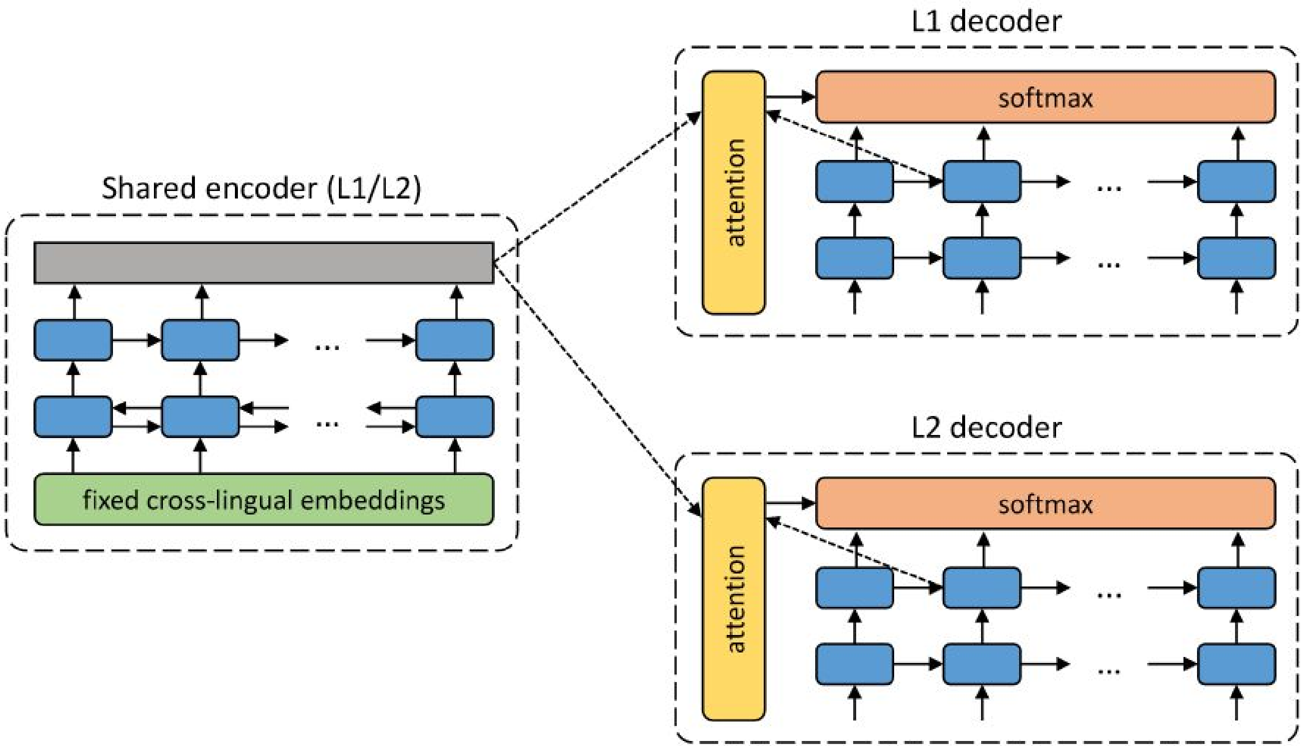
Improving Neural Machine Translation Models with Monolingual Data

We add fake data to training set to improve decoder’s performance:

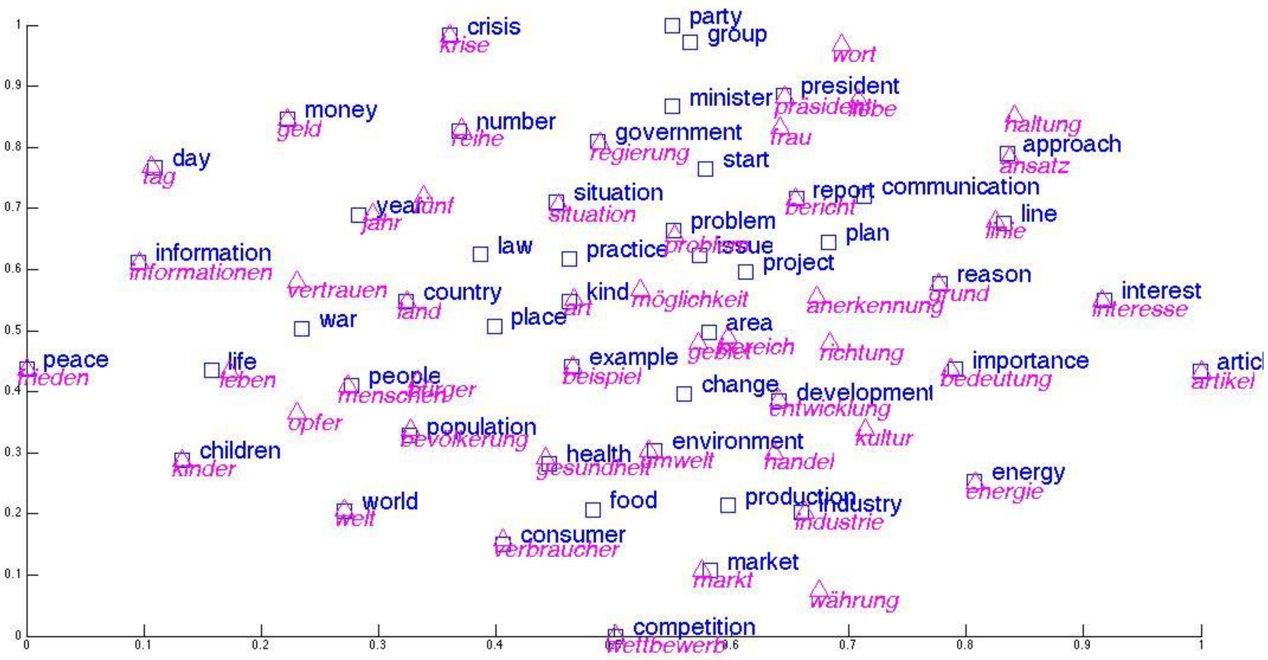
- empty source > target language sentence
- back-translated target language sentence > target language sentence

It’s **cheap** to generate these samples

Unsupervised Neural Machine Translation



Unsupervised Neural Machine Translation



Luong, M.-T., Pham, H., & Manning, C. D. (2015). *Bilingual Word Representations with Monolingual Quality in Mind.* Workshop on Vector Modeling for NLP, 151–159.

Unsupervised Neural Machine Translation

1) During training we use Encoder and L_X Decoder on L_X samples as denoising autoencoder. Noise is random word swaps simulating differences in word order of languages.
2) They use back-translation: First translate from L_X to L_Y with current system and then learn to translate back to L_X.

They alternate between (1) and (2) each batch of 50 samples. (1) is more important at the start of the learning, (2) at the end.

Unsupervised Machine Translation Using Monolingual Corpora Only

One encoder, one decoder, separate embeddings. Loss function:

1) denoising autoencoding on both languages (dropped words, swapped words)
2) back-translation (bootstrapped with bilingual dictionary)
3) adversarial training to force encoder to create universal representations

$$\begin{aligned}\mathcal{L}(\theta_{enc}, \theta_{dec}, \mathcal{Z}) =& \lambda_{auto}[\mathcal{L}_{auto}(\theta_{enc}, \theta_{dec}, \mathcal{Z}, src) + \mathcal{L}_{auto}(\theta_{enc}, \theta_{dec}, \mathcal{Z}, tgt)] + \\ & \lambda_{cd}[\mathcal{L}_{cd}(\theta_{enc}, \theta_{dec}, \mathcal{Z}, src, tgt) + \mathcal{L}_{cd}(\theta_{enc}, \theta_{dec}, \mathcal{Z}, tgt, src)] + \\ & \lambda_{adv}\mathcal{L}_{adv}(\theta_{enc}, \mathcal{Z}|\theta_D)\end{aligned}$$

Unsupervised Machine Translation Using Monolingual Corpora Only

Adversarial training: Adversity between

1) *discriminator*, which tries to predict sample language from its representation
2) *encoder*, which tries to create language independent representations

Unsupervised Machine Translation Using Monolingual Corpora Only

Key differences:

- shared decoder (it expects language id as parameter)
- word translation instead of multi-lingual word embeddings
- adversarial training
- joint training of all parameters