Transfer Learning for Unsupervised NMT

Alexandra Chronopoulou

achron@cis.lmu.de

CIS, LMU Munich

10/06/2020

Presentation Outline

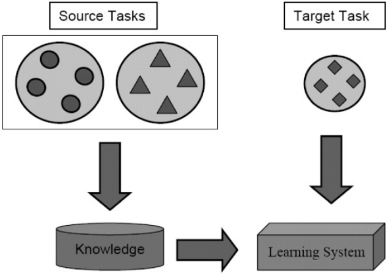
1. Motivation for Transfer Learning
2. Recap: What we have learned so far
3. Transfer Learning for NMT
4. Transfer Learning for Unsupervised NMT
   - Motivation for Unsupervised Language Model Pretraining
   - A state-of-the-art Transformer Language Model: BERT
   - Cross-Lingual Language Model Pretraining

Motivation for Transfer Learning

Machine learning

Problems (especially in deep learning):

- Scarcity of labeled data
- Models trained on small datasets often fail to generalize in test data $\rightarrow$ overfit



Transfer learning:

- Uses knowledge from a *learned* task to improve the performance on a *related* task
- Scarcity of labeled data $\rightarrow$ implicit data augmentation
- Helps a model generalize $\rightarrow$ avoid overfitting

Motivation for Transfer Learning

Natural language processing & Machine Translation

In Natural Language Processing tasks:

- **Out-of-context** pretrained word representations were used (*word2vec, fasttext) to initialize the* **embedding layer**

Motivation for Transfer Learning

Natural language processing & Machine Translation

In Natural Language Processing tasks:

- **Out-of-context** pretrained word representations were used (*word2vec, fasttext) to initialize the* **embedding layer**
- Recently: **contextual** representations from language models (*BERT*, *GPT OpenAI*) are used to initialize the **full model**

Presentation Outline

Recap: What we have learned so far

Supervised Learning methods in NMT work really well

... **if** a lot of parallel data available!

- We are provided the **ground truth**
- We use encoder-decoder models to
  - encode a sentence written in language x (hidden representation *s*)
  - provide *s* to decoder, it generates the sentence in language y → y'
  - compute training loss (by comparing translation y' to ground truth y)

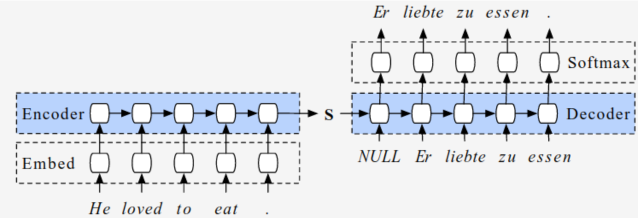


Figure: Seq2seq architecture for En-De NMT. Figure from https://smerity.com/articles/2016/google_nmt_arch.html

Recap: What we have learned so far

Why do we care about **Unsupervised Learning**?

- NMT models work very well, provided **a lot of** parallel data
- The size and domain of **parallel** data is limited

- **Monolingual** data is easier to acquire and abundant (for most lang.)

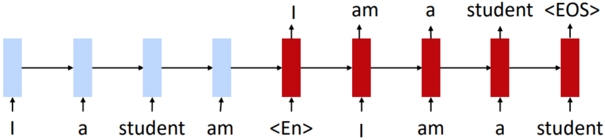
- **Goal**: uncover latent structure in unlabeled data
- **Unsupervised NMT** is not 100% realistic but...
- it serves as a very good baseline for extensions with parallel data (Semi-supervised Learning)

Recap: What we have learned so far

How does Unsupervised NMT work?

We use **two** new objectives:

1. Learn the structure of each language... How?
   Denoising auto-encoding
   (Language Model (LM) + noise + swap words)



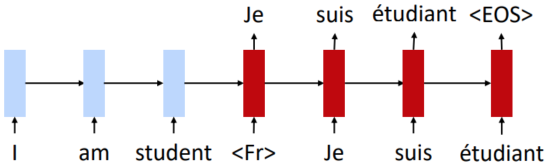
Recap: What we have learned so far

How does Unsupervised NMT work?

We use **two** new objectives:

2. Force the representation to be good at translating too...without parallel data. How?

Iterative backtranslation



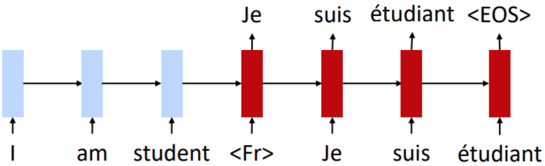
Recap: What we have learned so far

How does Unsupervised NMT work?

We use **two** new objectives:

2. Force the representation to be good at translating too...without parallel data. How?

Iterative backtranslation



- First translate fr $\rightarrow$ en

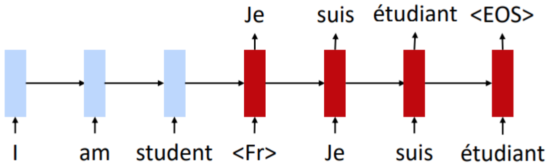
Recap: What we have learned so far

How does Unsupervised NMT work?

We use **two** new objectives:

2. Force the representation to be good at translating too...without parallel data. How?

Iterative backtranslation



- First translate fr $\to$ en
- Then use as a **pseudo-supervised** example to train en $\to$ fr

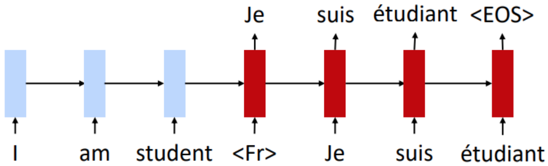
Recap: What we have learned so far

How does Unsupervised NMT work?

We use **two** new objectives:

2. Force the representation to be good at translating too...without parallel data. How?

Iterative backtranslation



- First translate fr $\rightarrow$ en
- Then use as a **pseudo-supervised** example to train en $\rightarrow$ fr
- Why does this work? We initialize the model with **word translations** from a dictionary created with `bilingual word embeddings` - guides first iteration

Presentation Outline

Transfer Learning for NMT

What happens when we **don't** have enough parallel data to train an NMT model?

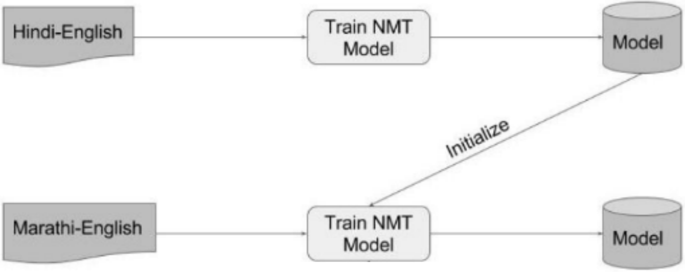
Transfer Learning for NMT

What happens when we **don't** have enough parallel data to train an NMT model?

How can we build systems that provide accurate translations between **low-resource** languages?

Transfer Learning for NMT

Transferring a model trained on a **lot** of parallel data to a model that has only **small** amounts of parallel data gives a large performance boost!
(e.g. `Hindi-English` $\rightarrow$ `Marathi-English`)

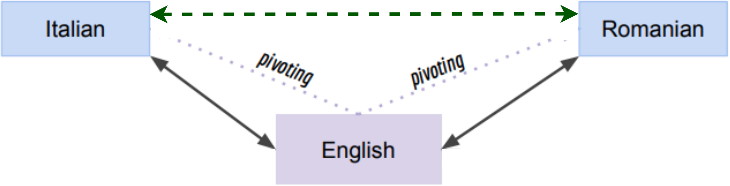


Transfer Learning for NMT

We can also use **pivot translation**!

We want to build an Italian-Romanian translation system
(low-resource - we don't have a lot of parallel corpora available)

We have **En-It** and **En-Ro** parallel corpora!



We can pretrain two NMT systems, that are then **transferred** to the final NMT system

Transfer Learning for NMT

- Transfer learning from an NMT system pretrained on **large parallel corpora** to an NMT system with **small parallel corpora** has **limitations**
- Parallel corpora are hard to find
- For some languages, there are no closely related high-resource languages
- How can we overcome this problem?

Transfer Learning for NMT

- Transfer learning from an NMT system pretrained on **large parallel corpora** to an NMT system with **small parallel corpora** has **limitations**
- Parallel corpora are hard to find
- For some languages, there are no closely related high-resource languages
- How can we overcome this problem?

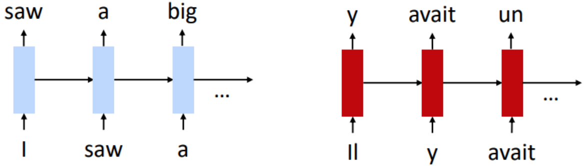
$\longrightarrow$ Unsupervised pretraining using monolingual data!

Transfer Learning for NMT

Can we use transfer learning (and specifically unsupervised pretraining) to initialize an NMT model in a better way?

Idea:

1. Separately **Pretrain** Encoder and Decoder as **Language Models**

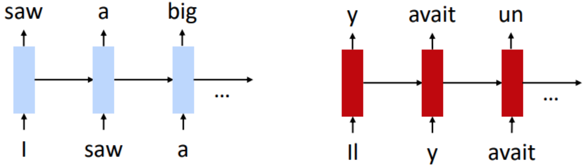


Transfer Learning for NMT

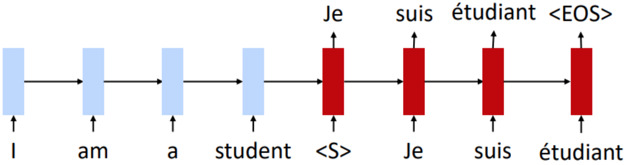
Can we use transfer learning (and specifically unsupervised pretraining) to initialize an NMT model in a better way?

Idea:

1. Separately **Pretrain** Encoder and Decoder as **Language Models**



2. Then **Train Jointly** on Bilingual Data (NMT)



(Figures from **Kevin Clark's talk**)

Presentation Outline

1. Motivation for Transfer Learning
2. Recap: What we have learned so far
3. Transfer Learning for NMT
4. Transfer Learning for Unsupervised NMT
   - Motivation for Unsupervised Language Model Pretraining
   - A state-of-the-art Transformer Language Model: BERT
   - Cross-Lingual Language Model Pretraining

Motivation for Unsupervised Language Model Pretraining

Remember that we use **word translations** obtained by bilingual word embeddings to initialize the unsupervised NMT model

How can we improve this?

Motivation for Unsupervised Language Model Pretraining

- Pretraining the encoder and decoder using two separate language models is not **directly** applicable to unsupervised NMT

Motivation for Unsupervised Language Model Pretraining

- Pretraining the encoder and decoder using two separate language models is not **directly** applicable to unsupervised NMT
- There is no "interaction" between the two languages during pretraining

Motivation for Unsupervised Language Model Pretraining

- Pretraining the encoder and decoder using two separate language models is not **directly** applicable to unsupervised NMT
- There is no "interaction" between the two languages during pretraining

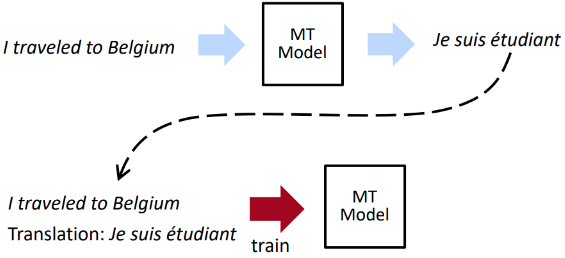
  1. The encoder LM learns how to produce proper `En` sentences

Motivation for Unsupervised Language Model Pretraining

- Pretraining the encoder and decoder using two separate language models is not **directly** applicable to unsupervised NMT
- There is no "interaction" between the two languages during pretraining
  1. The encoder LM learns how to produce proper `En` sentences
  2. The decoder LM learns how to produce proper `Fr` sentences

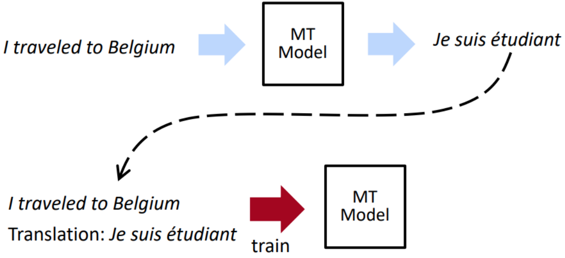
Motivation for Unsupervised Language Model Pretraining

- Pretraining the encoder and decoder using two separate language models is not **directly** applicable to unsupervised NMT
- There is no "interaction" between the two languages during pretraining
  1. The encoder LM learns how to produce proper `En` sentences
  2. The decoder LM learns how to produce proper `Fr` sentences
- If we directly applied it to unsupervised NMT...



Motivation for Unsupervised Language Model Pretraining

- Pretraining the encoder and decoder using two separate language models is not **directly** applicable to unsupervised NMT
- There is no "interaction" between the two languages during pretraining
  1. The encoder LM learns how to produce proper `En` sentences
  2. The decoder LM learns how to produce proper `Fr` sentences
- If we directly applied it to unsupervised NMT...



- The first sentence is in `En`, the second sentence is in `Fr`, **but** the `Fr` sentence is **not** a translation of the `En` sentence!

Motivation for Unsupervised Language Model Pretraining

Extension of idea, specifically for unsupervised NMT:

- Training two language models (LMs) separately does not permit "interaction" between the two languages

Motivation for Unsupervised Language Model Pretraining

Extension of idea, specifically for unsupervised NMT:

- Training two language models (LMs) separately does not permit "interaction" between the two languages
- Training **one** LM on **two** languages could be more helpful

Motivation for Unsupervised Language Model Pretraining

Extension of idea, specifically for unsupervised NMT:

- Training two language models (LMs) separately does not permit "interaction" between the two languages
- Training **one** LM on **two** languages could be more helpful

$\longrightarrow$ We want to pretrain a bilingual LM on **monolingual** data of 2 languages simultaneously to obtain better initial translations

Motivation for Unsupervised Language Model Pretraining

Extension of idea, specifically for unsupervised NMT:

- Training two language models (LMs) separately does not permit "interaction" between the two languages
- Training **one** LM on **two** languages could be more helpful

$\longrightarrow$ We want to pretrain a bilingual LM on **monolingual** data of 2 languages simultaneously to obtain better initial translations

And not just a "regular" LM...but the state-of-the-art LM nowadays, called **BERT**

Motivation for Unsupervised Language Model Pretraining

Extension of idea, specifically for unsupervised NMT:

- Training two language models (LMs) separately does not permit "interaction" between the two languages
- Training **one** LM on **two** languages could be more helpful

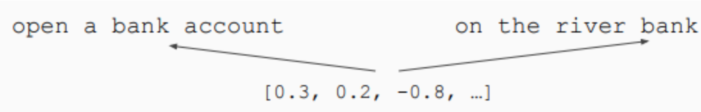
$\longrightarrow$ We want to pretrain a bilingual LM on **monolingual** data of 2 languages simultaneously to obtain better initial translations

And not just a "regular" LM...but the state-of-the-art LM nowadays, called **BERT**

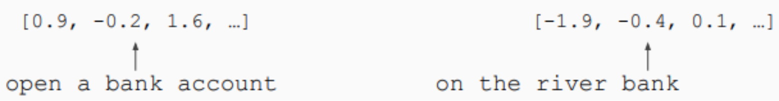
Not so fast... what is BERT?

A state-of-the-art Language Model: BERT

- **Problem**: Word embeddings (like word2vec) do not encode context (`bank` has the same embedding, but two different meanings)



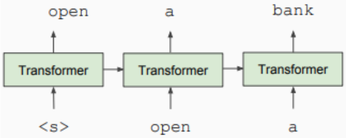
- **Solution**: Ideally, representations should be *contextual*



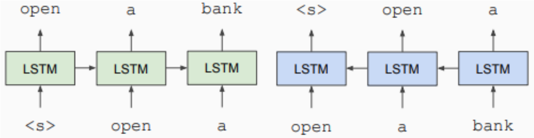
(Figures from **Jacob Devlin's talk**)

A state-of-the-art Transformer Language Model: BERT

- Previous approaches trained a **left-to-right** Transformer LM (OpenAI GPT)



- or a **bi-directional** LSTM LM



- Problem 1: Left-to-right Transformer LMs do not generate a well-formed probability distribution of words
- Problem 2: Bi-directional LSTM LMs "see themselves" in a bi-directional encoder

A state-of-the-art Transformer Language Model: BERT

Solution: Use a Transformer architecture (remember last lecture), randomly mask out 15% of the input words, and then predict only the masked words by attending to **all** unmasked words

```
                         store            gallon
                           ↑                ↑
the man went to the [MASK] to buy a [MASK] of milk
```

BERT is trained using the following 2 objectives:

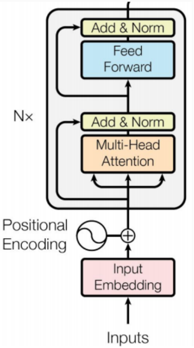
1. **LM**: At each time step, the LM predicts **only** the masked words
2. **Next Sentence Prediction**: Predict whether Sentence B is actual sentence that proceeds Sentence A, or a random sentence

Sentence A = `The man went to the store.`
Sentence B = `He bought a gallon of milk.`
Label = `IsNextSentence`

Sentence A = `The man went to the store.`
Sentence B = `Penguins are flightless.`
Label = `NotNextSentence`

A state-of-the-art Transformer Language Model: BERT

- The Masked LM is in fact an encoder Transformer

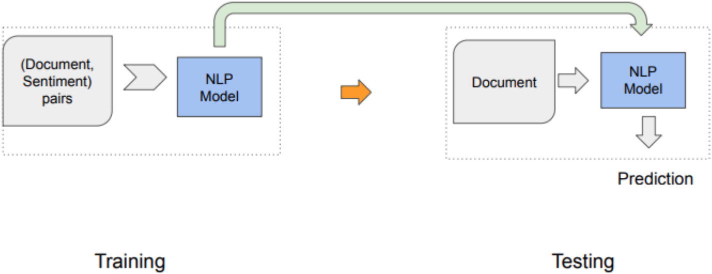


- Fine-tuning BERT to supervised tasks (NLI, sentiment analysis, question answering, and many others) gives **state-of-the-art** results

A state-of-the-art Transformer Language Model: BERT

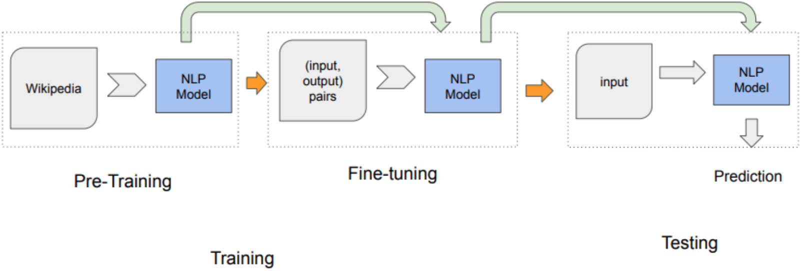
- How does that change the way we handle NLP tasks?

  Before, most models were trained **from scratch**, using pretrained embeddings (word2vec, fasttext) to initialize **only** the embedding layer:



A state-of-the-art Transformer Language Model: BERT

- **Now**, we fine-tune BERT to the supervised task and then we run the prediction:



A state-of-the-art Transformer Language Model: BERT

- Specifically for spam detection:

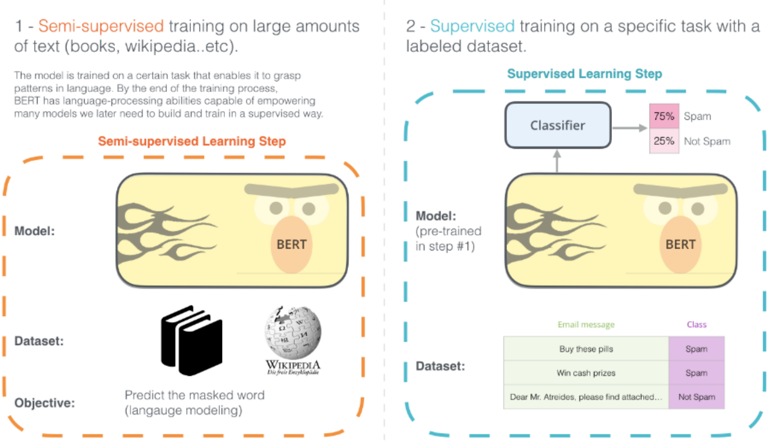


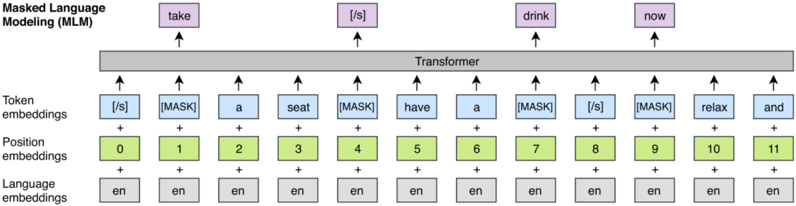
Figure: BERT fine-tuning example from `http://jalammar.github.io/illustrated-bert/`

Cross-Lingual Language Model Pretraining

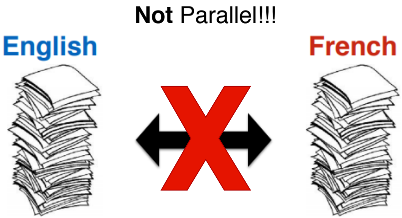
- Following the same line of thought, we want to use transfer learning for unsupervised NMT
- A LM that provides *contextual* word representations in both languages we care about gives far better initial translations than a simple dictionary obtained from bilingual word embeddings
- Then, we can initialize an **unsupervised** encoder-decoder NMT model with the pretrained bilingual LM!

Cross-Lingual Language Model Pretraining

- **Pretrain BERT simultaneously on 2 languages** (without the next sentence prediction task)



Large amounts of training data:

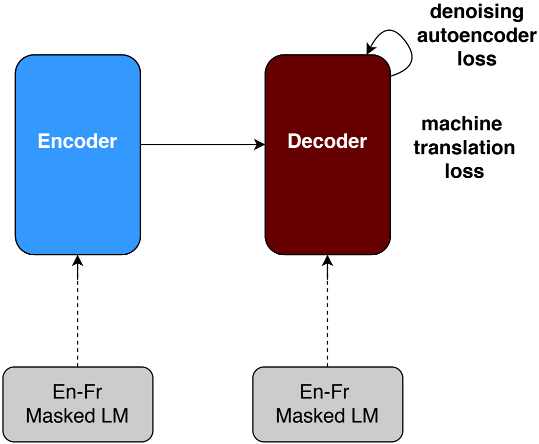


Cross-Lingual Language Model Pretraining

- We have a shared encoder and decoder (for both En→Fr and Fr→En)

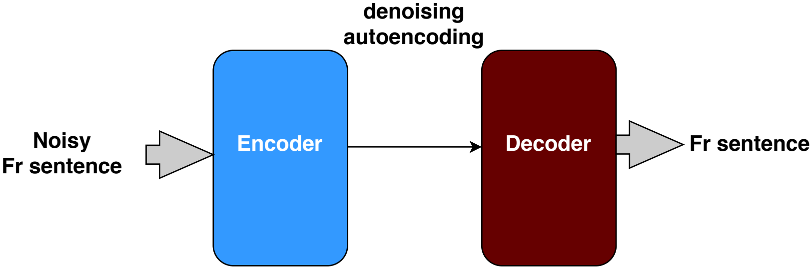
Cross-Lingual Language Model Pretraining

- We have a shared encoder and decoder (for both En→Fr and Fr→En)
- We initialize the encoder **and** the decoder with a **bilingual masked language model** (pretrained on a lot of monolingual data)!



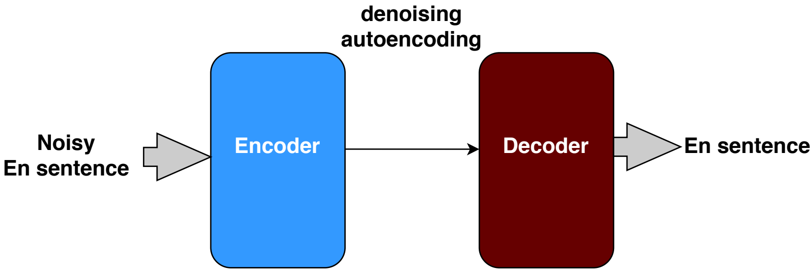
Cross-Lingual Language Model Pretraining

- We have a shared encoder and decoder (for both En→Fr and Fr→En)
- We train the NMT model using as training objectives (losses) **denoising auto-encoding** and **iterative backtranslation**



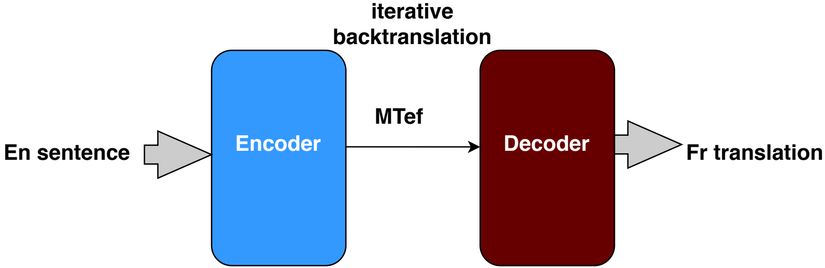
Cross-Lingual Language Model Pretraining

- We have a shared encoder and decoder (for both En→Fr and Fr→En)
- We train the NMT model using as training objectives (losses) **denoising auto-encoding** and **iterative backtranslation**



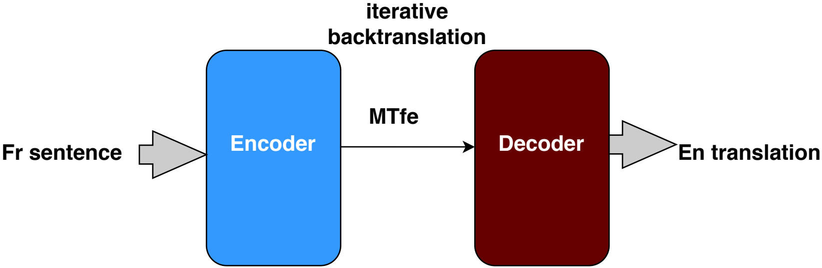
Cross-Lingual Language Model Pretraining

- We have a shared encoder and decoder (for both En→Fr and Fr→En)
- We train the NMT model using as training objectives (losses) **denoising auto-encoding** and **iterative backtranslation**



Cross-Lingual Language Model Pretraining

- We have a shared encoder and decoder (for both En→Fr and Fr→En)
- We train the NMT model using as training objectives (losses) **denoising auto-encoding** and **iterative backtranslation**



Cross-Lingual Language Model Pretraining

Unsupervised NMT Results

Model	En-Fr	En-De	En-Ro
UNMT	25.1	17.2	21.2
UNMT + Pre-Training	33.4	26.4	**33.3**
Current supervised State-of-the-art	**45.6**	**34.2**	29.9

Table from **Kevin Clark's talk**.

Cross-Lingual Language Model Pretraining

Why does training an LM jointly on 2 languages (and transferring it to an encoder-decoder NMT model) provide good initial translations?

- The underlying reason is that we encode text in a **subword** level
- Subword token improves the alignment of embedding spaces of two languages (especially if they share the `alphabet` or the `digits`)
- An example of phenomena for which subword information is useful:

Figure from Graham Neubig notes on MT class, Fall 2019.

Cross-Lingual Language Model Pretraining

Subword tokens provide useful cross-lingual information

- **Cognates**: words which share a common origin but have diverged at some point in the evolution of respective languages
- **Loan words**: words borrowed as-is from another language
- **Transliteration**: the process of converting words with identical or similar pronunciations from one script to another
- **Morphology**: systematic changing of word forms according to their grammatical properties such as tense, case, gender, part of speech

Cross-Lingual Language Model Pretraining

Limitations

- This pretraining method only works for **similar** languages, which have **comparable** corpora available (e.g. En Wikipedia and Fr News Corpus, not En Twitter and Fr Wikipedia)

Cross-Lingual Language Model Pretraining

Limitations

- There is only a **limited** number of languages that have **clean**, **comparable** monolingual data

Cross-Lingual Language Model Pretraining

Limitations

- There is only a **limited** number of languages that have **clean**, **comparable** monolingual data
- but there are more than 6000 languages in the world...

Some Stats

- 6000+ languages in the world
- 80% of the world population does not speak English
- Less than 5% of the people in the world are native English speakers.

Thank You for your Attention! Questions?

References I

T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems*, 2013, pp. 3111–3119. (visited on 03/22/2017).

P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017. DOI: 10.1162/tacl_a_00051. [Online]. Available: https://www.aclweb.org/anthology/Q17-1010.

References II

I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in Neural Information Processing Systems*, Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, Eds., 2014, pp. 3104–3112. [Online]. Available: `http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf`.

B. Zoph, D. Yuret, J. May, and K. Knight, "Transfer learning for low-resource neural machine translation," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, Austin, Texas: Association for Computational Linguistics, Nov. 2016, pp. 1568–1575. DOI: 10.18653/v1/D16-1163. [Online]. Available: `https://www.aclweb.org/anthology/D16-1163`.

References III

P. Ramachandran, P. Liu, and Q. Le, "Unsupervised pretraining for sequence to sequence learning," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, Copenhagen, Denmark: Association for Computational Linguistics, Sep. 2017, pp. 383–391. DOI: 10.18653/v1/D17-1039. [Online]. Available: https://www.aclweb.org/anthology/D17-1039.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019, pp. 4171–4186. [Online]. Available: https://www.aclweb.org/anthology/N19-1423.

References IV

G. Lample and A. Conneau, "Cross-lingual language model pretraining," in *Advances in Neural Information Processing Systems*, 2019, pp. 7057–7067. [Online]. Available: https://papers.nips.cc/paper/8928-cross-lingual-language-model-pretraining.