Bauman Moscow State Technical University www.bmstu.ru
Huawei Company www.huawei.com

French-Portuguese Neural Machine Translation Based on Transformers

V. Lobanova[1]

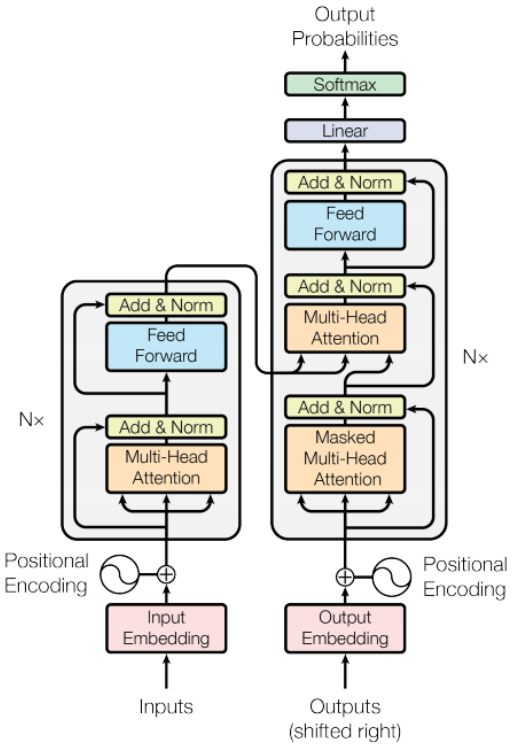
[1]Fundamental Sciences Department, Bauman Moscow State Technical University, Moscow, Russia

lobanova_vs@mail.ru

Huawei Natural Language Processing Course
19 January 2022
Moscow - Russia

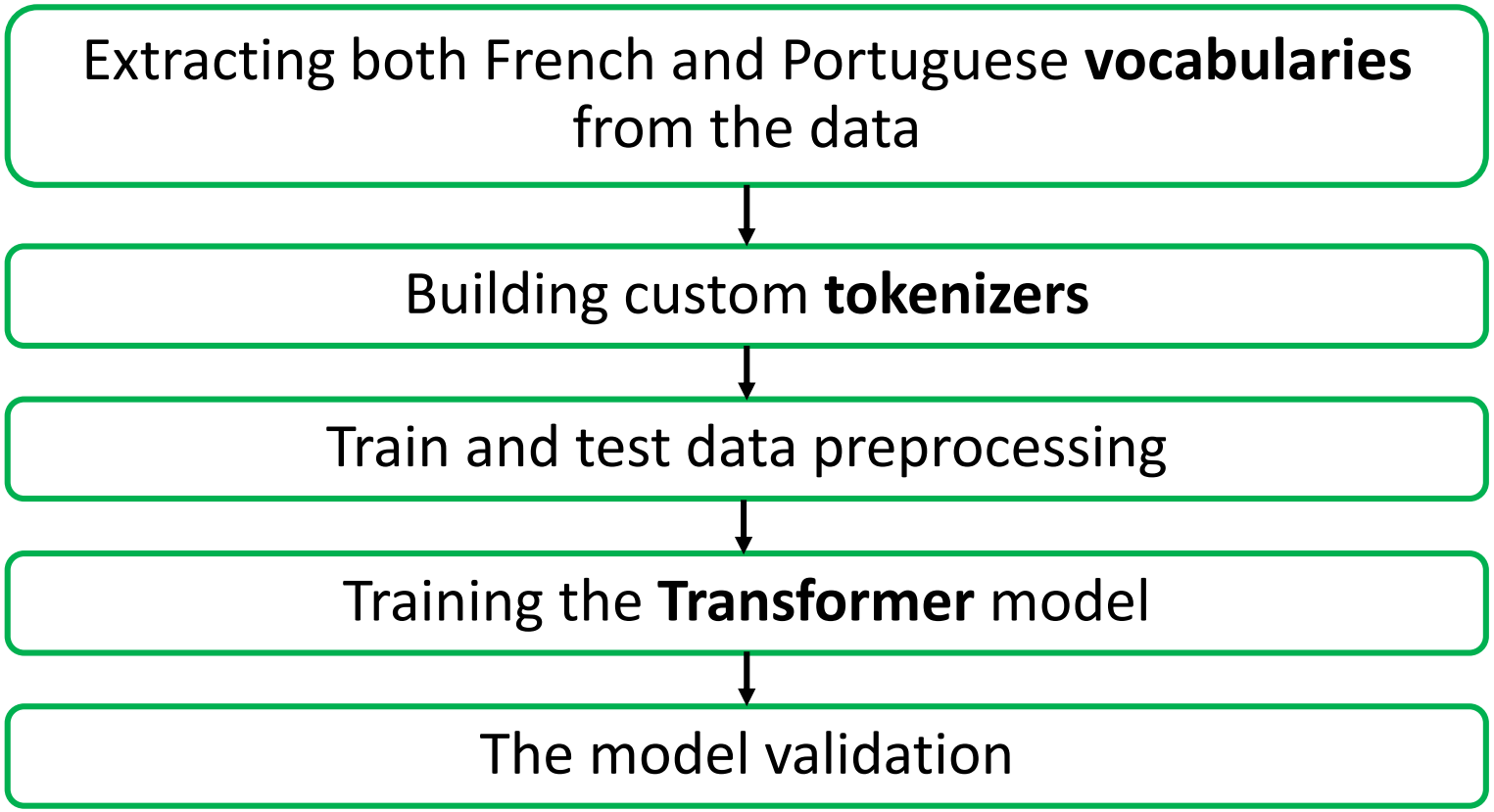
The Task

- **Neural machine translation** (NMT) is a general approach in machine translation
- The aim of the project was to train a French-Portuguese **NMT** model
- The resource and target languages are related
- **Data**: a relatively small French-Portuguese parallel corpus (TED Talks)
- **Architecture**: the Transformer model [Vaswani et al., 2017]



The figure on the slide is from Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need.

How to Solve The Problem?



Building Vocabularies

- We used French to Portuguese TED Talks dataset

Split	Examples
'test'	1,494
'train'	43,873
'validation'	1,131

```
French:  mais cela trahit aussi la panique , la terreur , que la grossophobie peut évoquer .
Portuguese:    mas também faz notar o pânico , o terror literal , que o medo da gordura evoca .
```

- Examples are lower case
- There are spaces around the punctuation

```
from tensorflow_text.tools.wordpiece_vocab import bert_vocab_from_dataset as bert_vocab
```

Custom tokenizers

```
pt_tokenizer = text.BertTokenizer('pt_vocab.txt', **bert_tokenizer_params)
fr_tokenizer = text.BertTokenizer('fr_vocab.txt', **bert_tokenizer_params)
```

```
[103, 42, 3784, 59, 340, 3996, 377, 40, 1068, 94, 247, 178, 55, 1794, 247, 1124, 14, 178, 146, 1037, 5126, 14, 84, 178, 46, 1290, 700, 731, 1758, 5426,
1746, 194, 377, 44, 1608, 1124, 149, 16]
```

```
array([b'mais c ##ela t ##ra ##hi ##t a ##us ##s ##i la p ##an ##i ##que , la ter ##re ##ur , que la g ##ros ##so ##p ##ho ##bie pe ##u ##t e ##vo ##que
##r .',
```

- The tokenized text have to include [START] and [END] tokens
- We reserved space at the beginning of the vocabulary, so [START] and [END] have the same indexes for both languages

```
array([b'[START] mais cela trahit aussi la panique , la terreur , que la grossophobie peut evoquer . [END]',
```

- The tokenize method converts a batch of strings to a padded-batch of token IDs

Model setup

- We aimed to keep our model small and relatively fast
- Model hyperparameters were

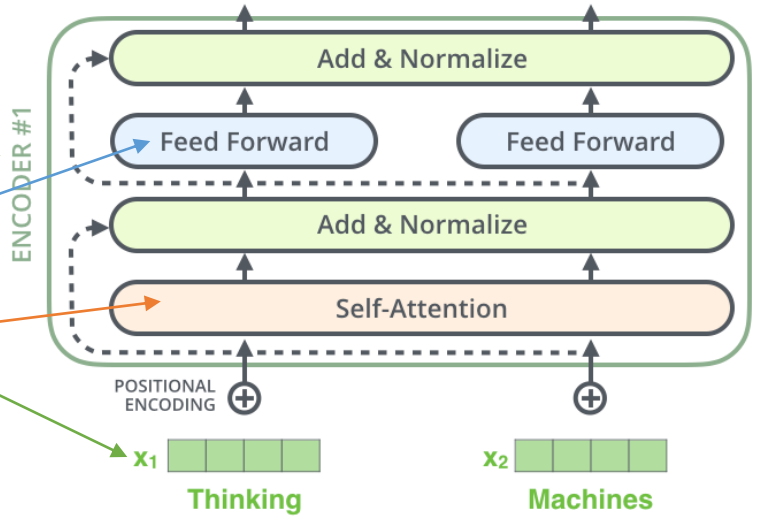
number of layers

dimension of outputs

inner-layers dimensionality

number of heads in multi-head attention

```
num_layers = 4
d_model = 128
dff = 512
num_heads = 8
dropout_rate = 0.1
```

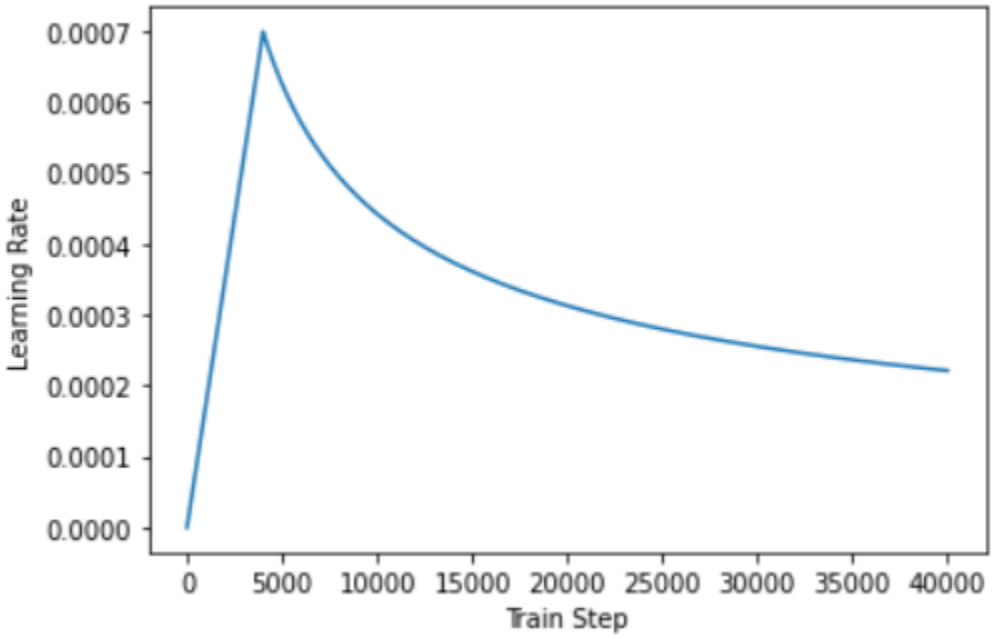


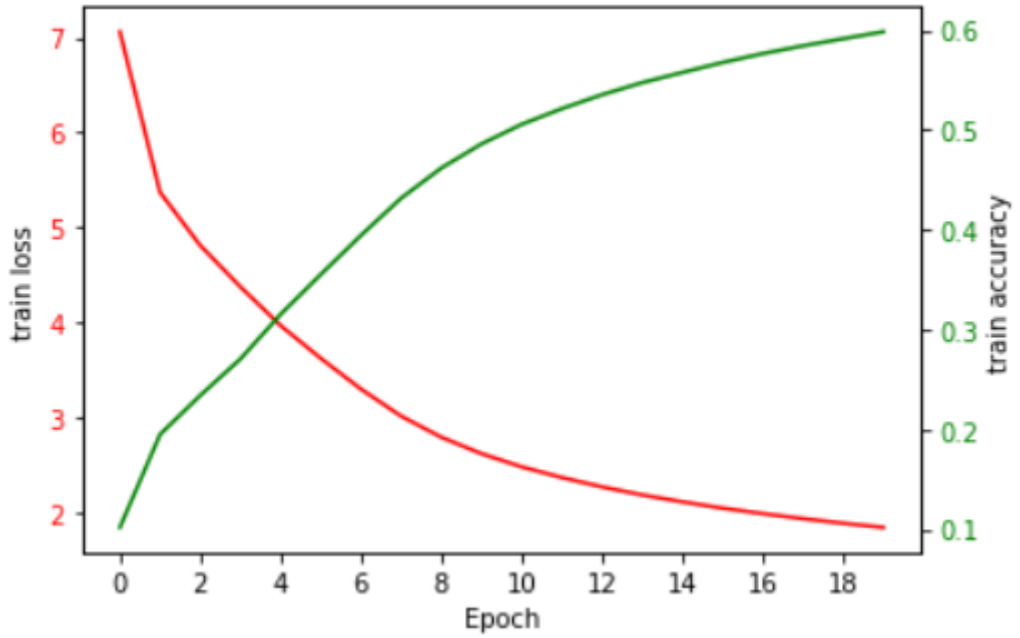
The figure on the right of the slide is : Fang Sun and Jing Zhang 2021 J. Phys.: Conf. Ser. 1952 042097, I. (2017). Research on Grammar Checking System Using Computer Big Data and Convolutional Neural Network Constructing Classification Model.

Model training

- We used the Adam optimizer with a custom learning rate

$$lrate = d_{model}^{-0.5} * \min(step\_num^{-0.5}, step\_num \cdot warmup\_steps^{-1.5})$$





Model validation

Input#1 (French): Masha marchait sur l'autoroute et aspirait le séchage.
Output#1 (Portuguese): O meu desejo estava a atravessar a auto - estrada e aspiratoria.
Reference#1 (Portuguese): A Masha estava a andar na auto-estrada e a sugar a secagem.

Input#2 (French): Dans la forêt , un arbre de Noël est né, dans la forêt , elle a grandi.
Output#2 (Portuguese): Na floresta , uma arvore de Natal, na floresta , cresceu na floresta , ela cresceu.
Reference#2 (Portuguese): Na floresta nasceu uma árvore de Natal, na floresta ela cresceu.

Input#3 (French): Qui est à blâmer et que faire ?
Output#3 (Portuguese): Quem e aprovisionar e o que fazer?
Reference#3 (Portuguese): Quem é o culpado e o que fazer?

```
[ ]  import nltk
     nltk.translate.bleu_score.corpus_bleu(references, predictions,
                                           weights=(0, 0, 0, 1.0))
```

Weights for 1-, 2-, 3-, and 4-grams, respectively	BLEU	BLEU + Smoothing function
(0.25, 0.25, 0.25, 0.25)	0.02	0.18
(1, 0, 0, 0)	0.19	0.46
(0, 1, 0, 0)	0.01	0.16
(0, 0, 1, 0)	0.00	0.12
(0, 0, 1, 0)	1.00	0.11

Smoothing method 4:
Shorter translations may have inflated precision values due to having smaller denominators; therefore, we give them proportionally smaller smoothed counts. Instead of scaling to 1/(2^k), Chen and Cherry suggests dividing by 1/ln(len(T)), where T is the length of the translation.

Smoothing method 5:
The matched counts for similar values of n should be similar. To a calculate the n-gram matched count, it averages the n−1, n and n+1 gram matched counts.

Boxing Chen and Collin Cherry (2014) A Systematic Comparison of Smoothing Techniques for Sentence-Level BLEU. In WMT14.