

# Breaking language barriers in Wikipedia with Neural Machine Translation

Marine Carpuat & Eleftheria Briakou



REU-CAAR

June 9, 2023

7,111 languages  
spoken today

23 account for  
50% of population

43% endangered




Google Translate (today)



这座中国首都拥有速度高得惊人的互联网，有人脸识别软件等尖端技术，在人工智能方面投入了巨资并且拥有无可匹敌的国际化能量，它对富于探索精神的外国人而言是最激动人心的城市之一。

With its blazingly fast internet, cutting-edge technology like facial recognition software, huge investments in artificial intelligence and an unrivaled cosmopolitan energy, the Chinese capital is an ideal destination for the inquisitive foreigner. One of the most exciting cities.

UMD Machine Translation  
@WMT 2018



这座中国首都拥有速度高得惊人的互联网，有人脸识别软件等尖端技术，在人工智能方面投入了巨资并且拥有无可匹敌的国际化能量，它对富于探索精神的外国人而言是最激动人心的城市之一。

The Chinese capital, with its surprisingly high-speed Internet, sophisticated technology such as face-recognition software, has invested heavily in artificial intelligence and has unrivaled international energy, and is one of the most exciting cities for exploration-minded foreigners.

# Google Translate (today)

## Alima Mahama

Atweresem **Nkɔtahode**

Kenkan Sesa Hwe abakosɛm Nneema

Woakyerɛw nɛɛm yi wo **Akuapem** kɛsa mu

**Hajia Alima Mahama** (wɔwoo no Cibubuo da a eto 7, afe 1957 wo Walewale, Atifi-Apuei manatam mu) yɛ mmaraɛnifo. Na oye mmaraɛyɛbaguamuni a ogyina ho ma Nalerigu/Gambaga mpesua. Oye amanyokuo New Patriotic Party muni.<sup>[1][2]</sup>

### Nhomasua | sesa |

Alima Mahama nyaa koo ntoaso sukuu wo Wesley Girls Senior High School a ewo Cape Coast. Otooa so koo suapɔn a ene University of Ghana.<sup>[3]</sup> Eho no onyaa abodin krataa wo mmaraɛnim mu. Osan nso koo suapɔn a wofir no Institute of Social Studies a ewo Netherlands na eho nso onyaa abodin krataa foforo kaa nea ewo dedaw no ho.

### Amanyosɛm | sesa |

Alima Mahama gyinaa amanyokuw New Patriotic Party (NPP) din mu sil akan wo Ghana 2016 Mpesua abtaow mu. Wo saa abataw no mu no ode aba no mu oha mu nkyekyɛmu 53 dii nkonim wo Nalerigu/Gambaga mpesua so.<sup>[2]</sup>

Osan somee sɛ Ghana aban soafo a ohwɛ mmea ne mmofra yiyedi asoɛt(Minister for Women and Children's Affairs) so fi Opepon 2005 kosi Opepon afe 2009. Saa bere no na omanpanyin a na odi Ghana so ye John Kofi Agyekum Kufour. Osan nso somee sɛ aban soafo a ohwɛ mpesua ne nkurase amanbuo(Minister for Local Government and Rural Development) so. Ghana manpanyin Nana Akufo-Addo na spaw no wo Opepon da a eto so 10, afe 2017. Osan nso somee sɛ osoafo abadiakyiri a ohwɛ aguadi so.<sup>[4]</sup>

Mahama somee sɛ Ghana aban ananmusini ma United States, Amerika. One Ghanaɛni bea a odi kan a wɔpaw no sɛ Ghana aban ananmusini ma Amerika.<sup>[1][2]</sup> Wɔpaw no afe 2021 wo omanpanyin Nana Akufo-Addo nniso mu.<sup>[1][4]</sup>



Hon.  
**Hajia Alima Mahama**

**Minister for Women and Childrens Affairs (Ghana)**

**Omanpanyin** John Kufuor

**Member of Parliament for Nalerigu/Gambaga**

**Bere**  
7 January 2017 – 6 January 2020

**Omanpanyin** Nana Akufo-Addo

**Minister of Local Government and Rural Development**

**Mprenpren**

**Bere**  
28 January 2017

Alima Mahama contested the Ghana 2016 Mpesua abtaow on behalf of the New Partioitic Party (NPP). In that election he won the Nalerigu/Gambaga riding with 53 percent of the vote.

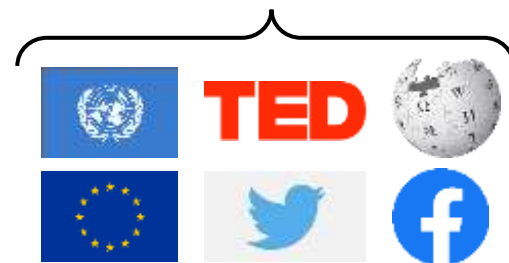
# Translation as Machine Learning

An English sentence  $e$  is translated  
into the French sentence

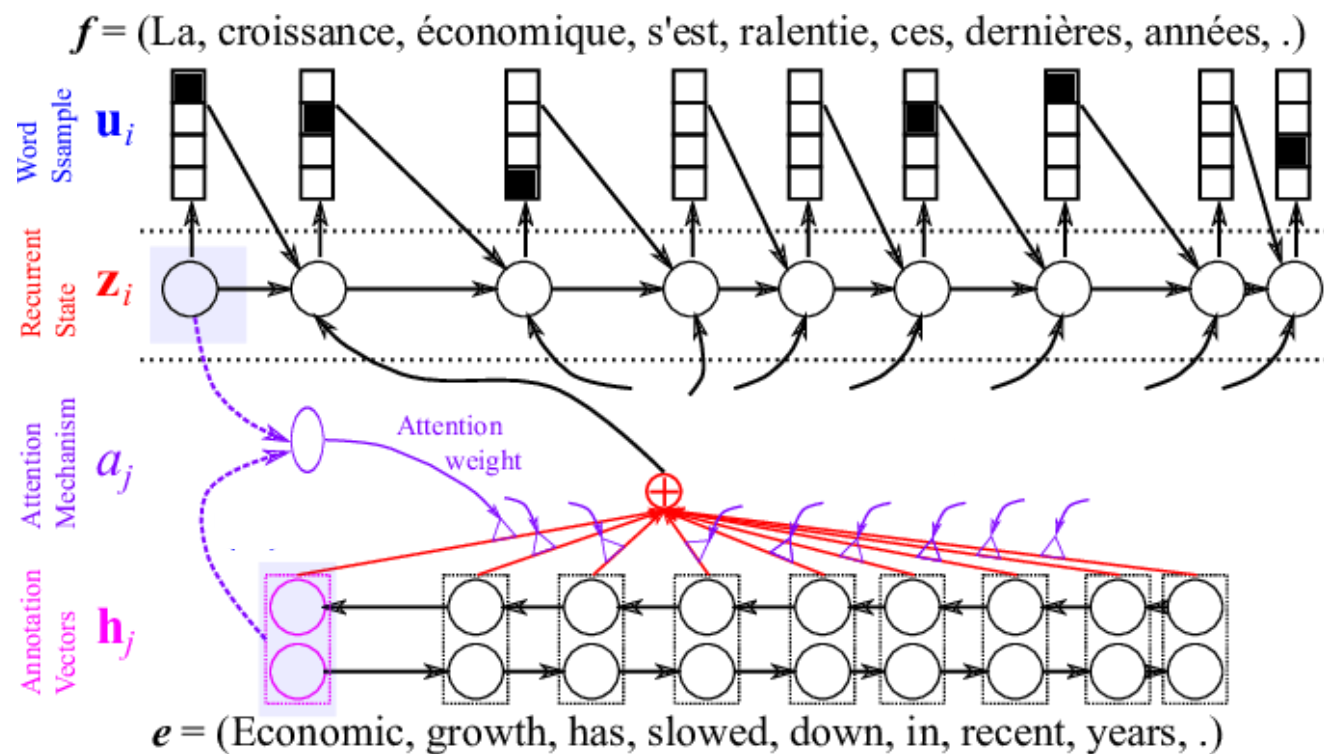
$$f^* = \operatorname{argmax}_f p(f|e; \theta)$$

“Learning” refers to finding good  
values for the model parameters

$$\theta^* = \operatorname{argmax}_\theta \sum_i \log p(f_i | e_i; \theta)$$



# Translation as Deep Learning



$$p(f | e; \theta) = \prod_{t=1}^T p(f_t | f_{<t}, e; \theta)$$

# Beyond Translation

**Transformer models** developed for Machine Translation are now general purpose tools to generate language

Share architecture design with language models (e.g., chatGPT) and word representation models (aka word embeddings)

Sequence-to-sequence models are used for dialog, text summarization, question answering, style transfer, etc.



# Challenges of Translation as Deep Learning

requires millions of translation examples  
not available for many languages!

raises fundamental machine learning challenges

intractably large output space, infinitely many correct outputs...

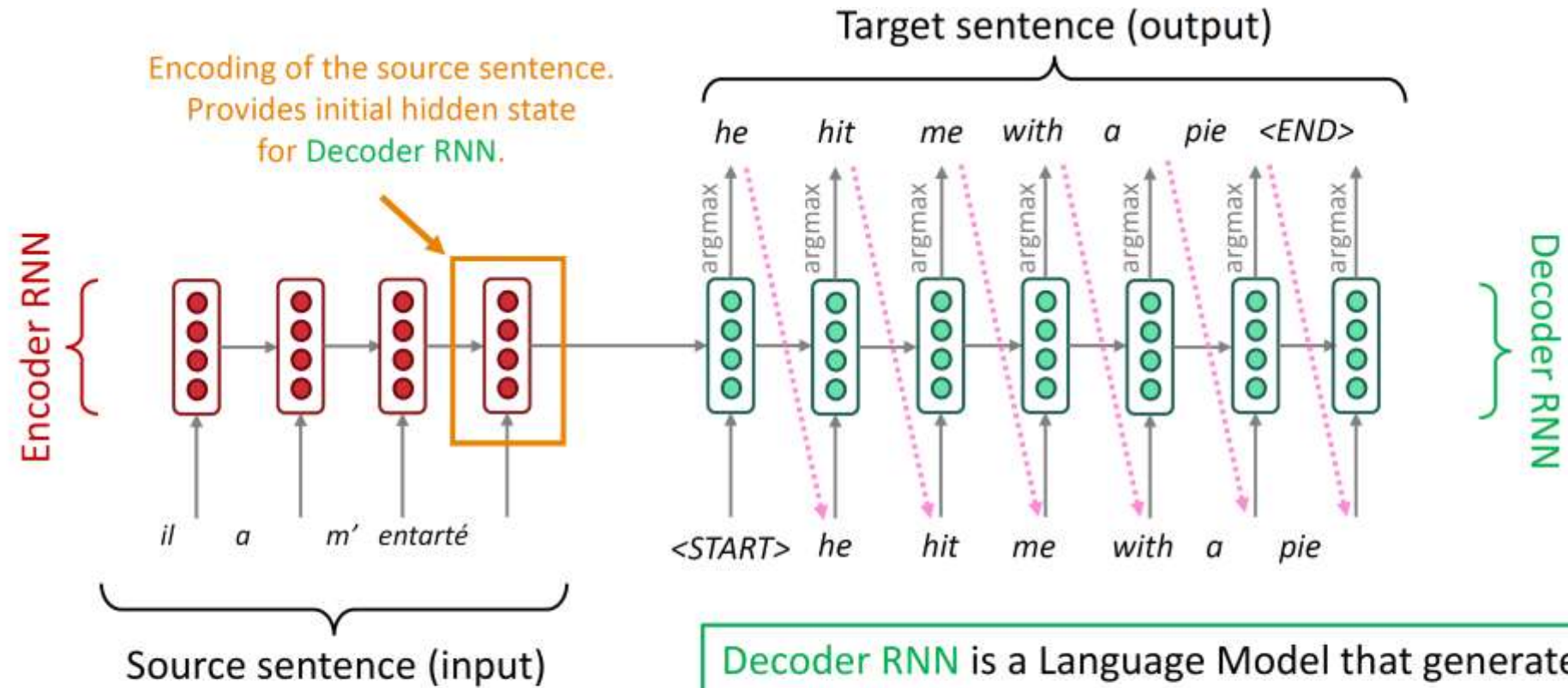
makes errors that have real world impact  
yet models are opaque, and developed independently from use cases

This summer project:  
How can we improve machine  
translation into low-resource languages  
for Wikipedia editors?

# A very brief introduction to Neural Machine Translation

A very brief introduction  
to Neural Machine Translation:  
Transformer models

# Neural MT with an encoder-decoder (initial c. 2015 models)



Encoder RNN produces an **encoding** of the source sentence.

Decoder RNN is a Language Model that generates target sentence, conditioned on *encoding*.

Can we build better building blocks within the encoder-decoder framework?

# Transformers: is attention all we need?

---

## Attention Is All You Need

---

**Ashish Vaswani\***  
Google Brain  
avaswani@google.com

**Noam Shazeer\***  
Google Brain  
noam@google.com

**Niki Parmar\***  
Google Research  
nikip@google.com

**Jakob Uszkoreit\***  
Google Research  
usz@google.com

**Llion Jones\***  
Google Research  
llion@google.com

**Aidan N. Gomez\* †**  
University of Toronto  
aidan@cs.toronto.edu

**Lukasz Kaiser\***  
Google Brain  
lukaszkaizer@google.com

**Illia Polosukhin\* ‡**  
illia.polosukhin@gmail.com

This 2017 paper introduces **Transformer models** that rely on **attention mechanisms** to aggregate context information across words within and across languages.

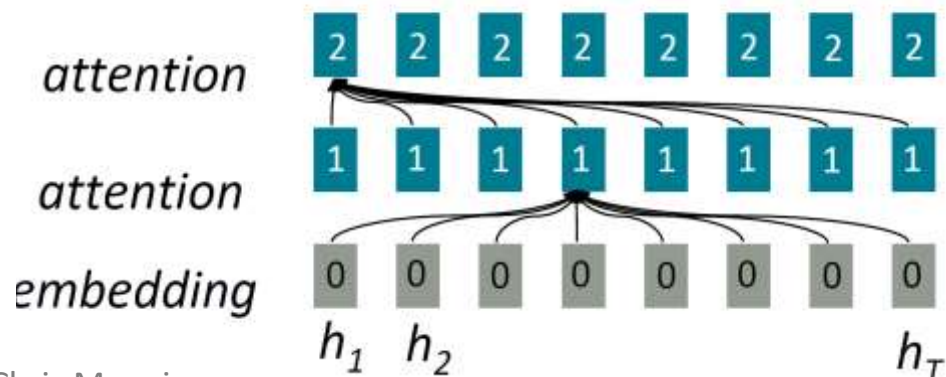
# Attention as an alternative to recurrence

## General definition of attention:

Given a set of vector **values** and a vector **query**, **attention** is a technique to compute a **weighted sum of the values, dependent on the query**

Attention can be applied

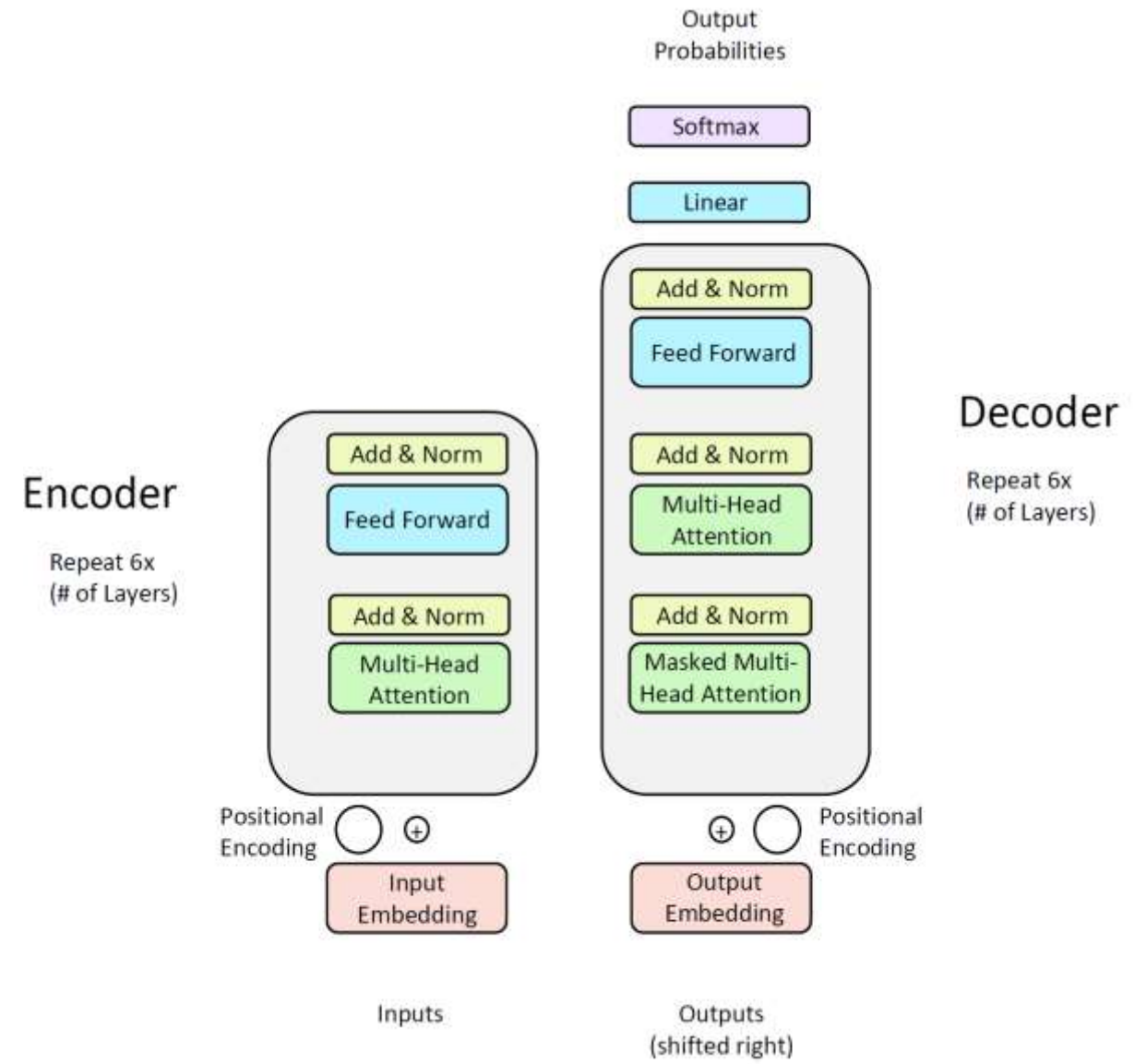
- Between the **decoder** and the **encoder**
- **Within the encoder-encoder** or **decoder-decoder** attention where query and values are within the **input** (or **output**)



All words attend to all words in previous layer; most arrows here are omitted

# Transformer

[Vaswani et al. 2017]





# Recipe for (vectorized) self-attention in the transformer encoder

Step 1: With embeddings stacked in  $X$ , calculate **queries**, **keys**, and **values**.

$$Q = XW^Q \quad K = XW^K \quad V = XW^V$$

Step 2: Calculate attention scores between **query** and **keys**.

$$E = QK^T$$

Step 3: Take the softmax to normalize attention scores.

$$A = \text{softmax}(E)$$

Step 4: Take a weighted sum of **values**.

$$\text{Output} = AV$$

Main take-away:

Lots of matrix-matrix operations that are efficient to compute on GPUs!

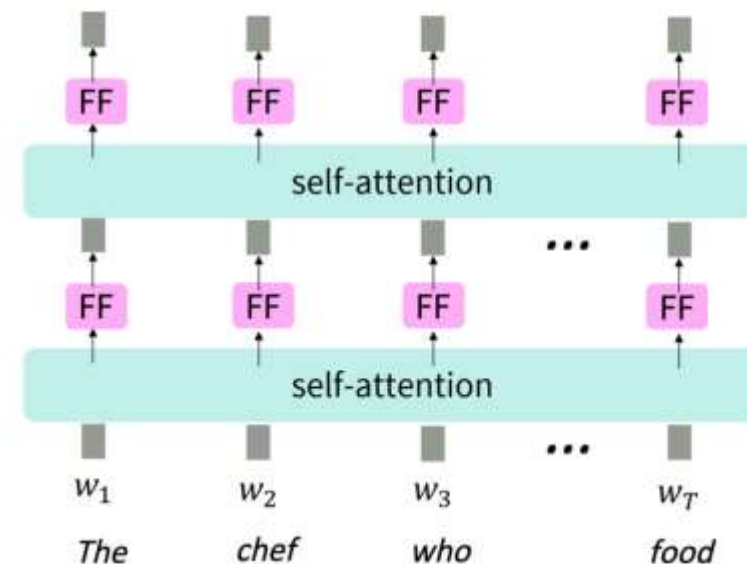
$$\text{Output} = \text{softmax}(QK^T)V$$

# But attention isn't quite all you need!

- Problem: self-attention is simply re-averaging the value vectors
- Fix: apply a **feedforward layer** to the output of attention providing non-linear activation (and thus additional expressive power)

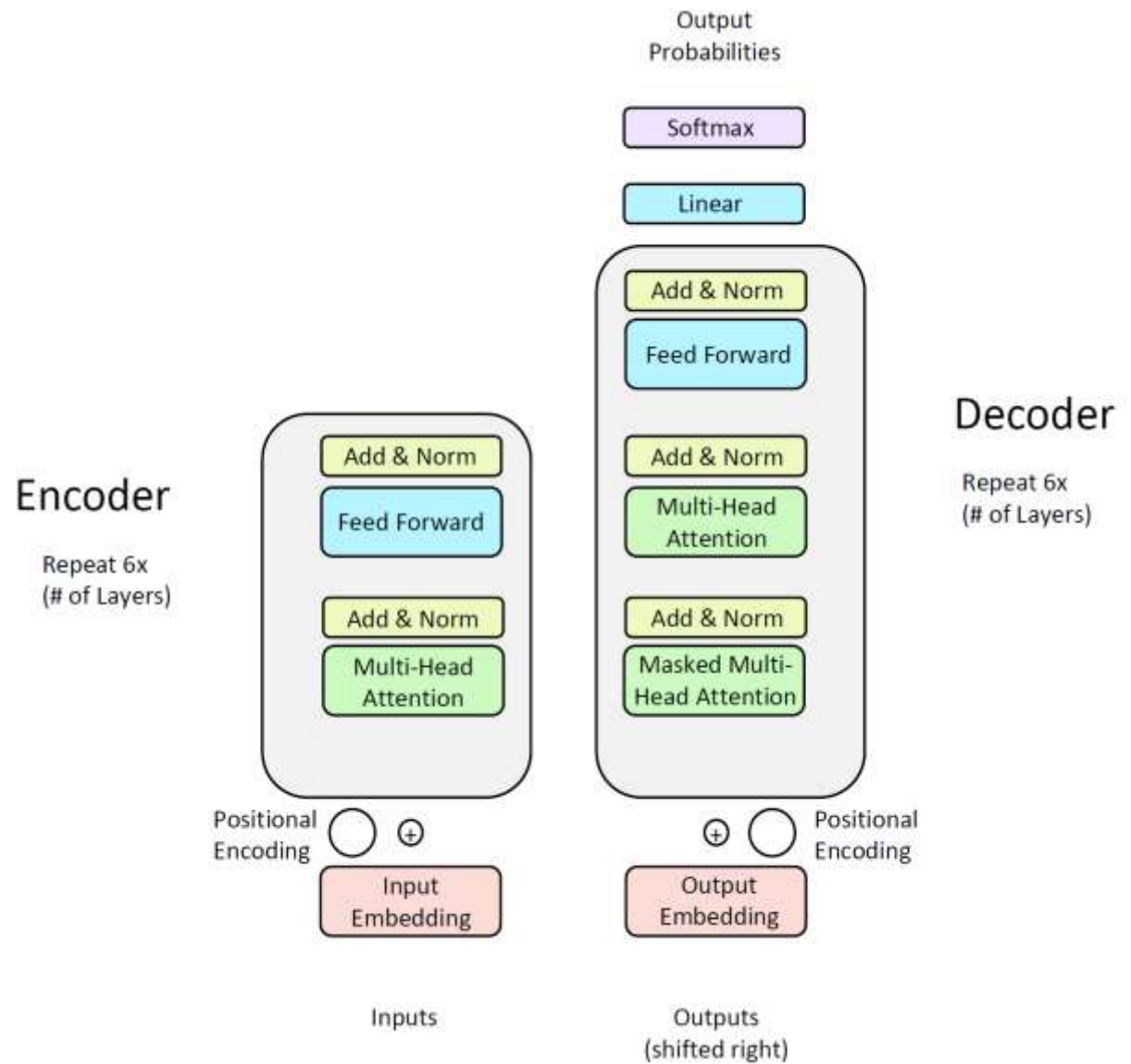
Equation for Feed Forward Layer

$$\begin{aligned} m_i &= MLP(\text{output}_i) \\ &= W_2 * \text{ReLU}(W_1 \times \text{output}_i + b_1) + b_2 \end{aligned}$$



# Transformer

[Vaswani et al. 2017]

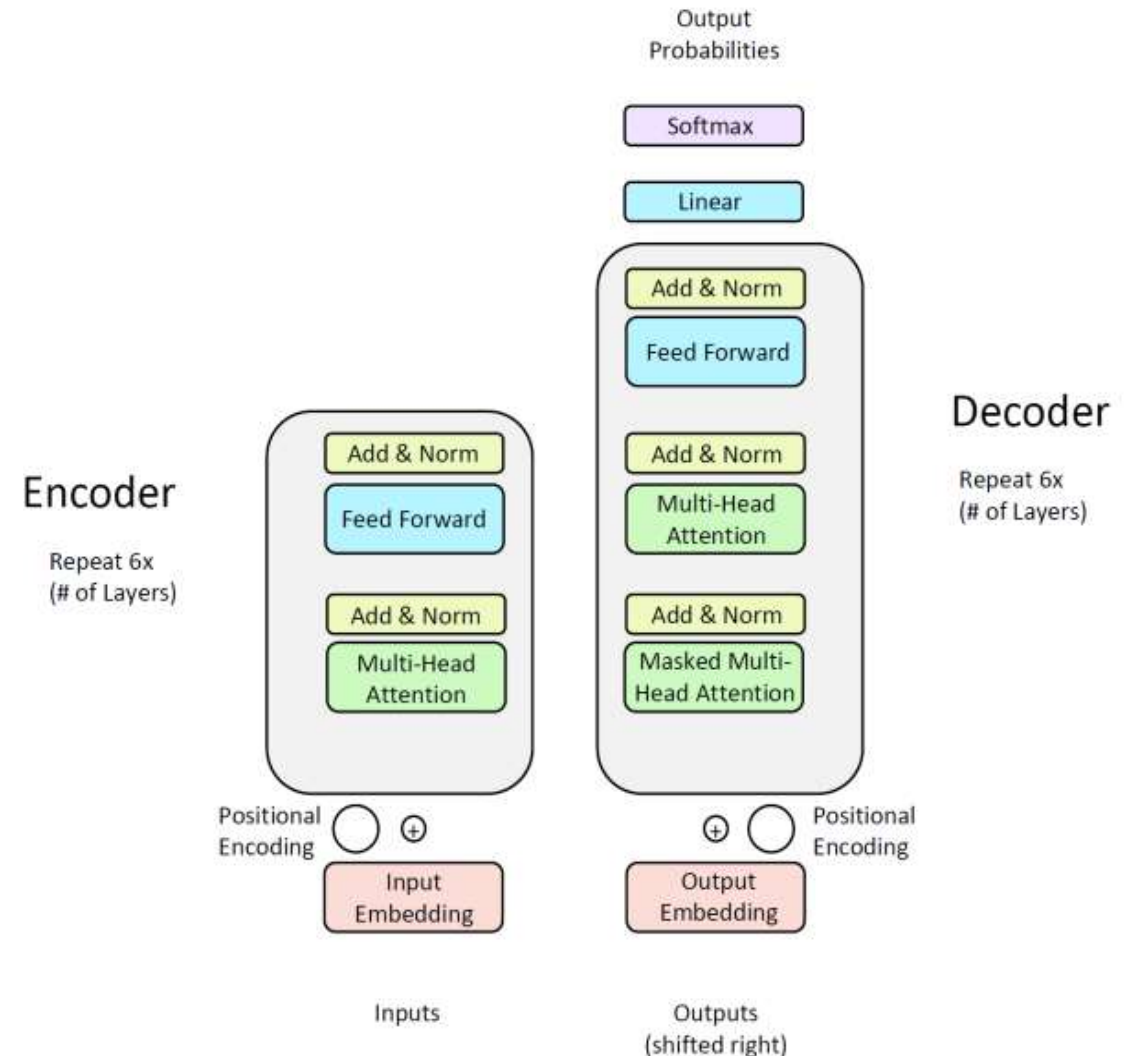


# We need a few more tricks to make this work

Goal: make it easier to train the model

## 3 tricks

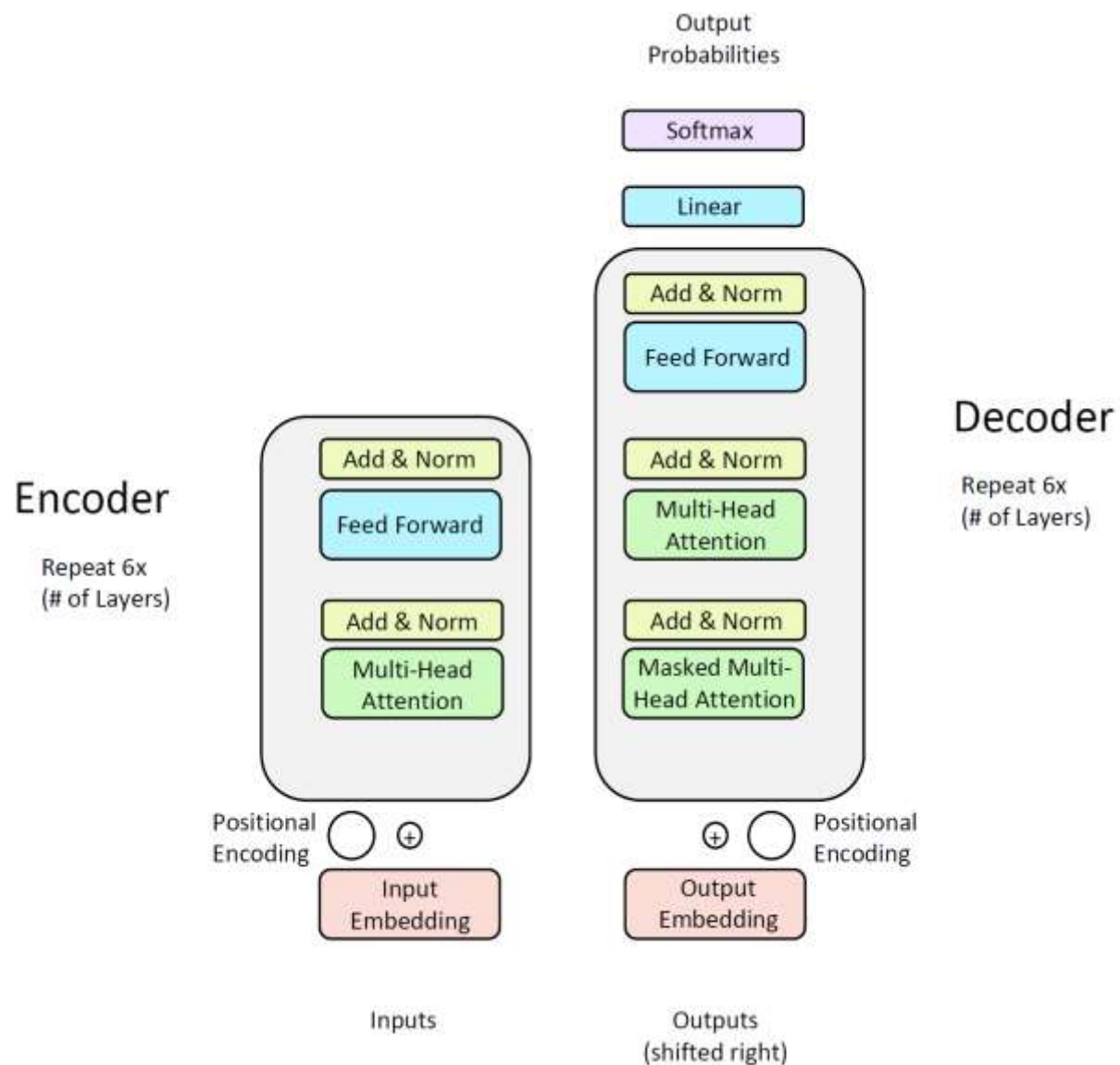
- Residual connections --- addresses gradient problems in deep networks
- LayerNorm and Scaled Dot Product Attention --- stabilize the values that parameters can take



# Transformer Decoder

Core components:

- masked multi-head attention
- encoder-decoder attention



# Encoder-Decoder Attention

- Self-attention is when keys, queries and values come from the same sequence
  - Source sequence for encoder self-attention
  - Target sequence for decoder self-attention
- In the decoder, we also need attention that captures source-target dependencies
  - **Keys** and **values** are drawn from the **encoder**  
 $k_i = Kh_i$ ,  $v_i = Xh_i$  where  $h_1, \dots, h_T$  are output vectors from the encoder
  - **Queries** are drawn from the **decoder**  
 $q_i = Qz_i$  where  $z_1, \dots, z_T$  are output vectors from the decoder

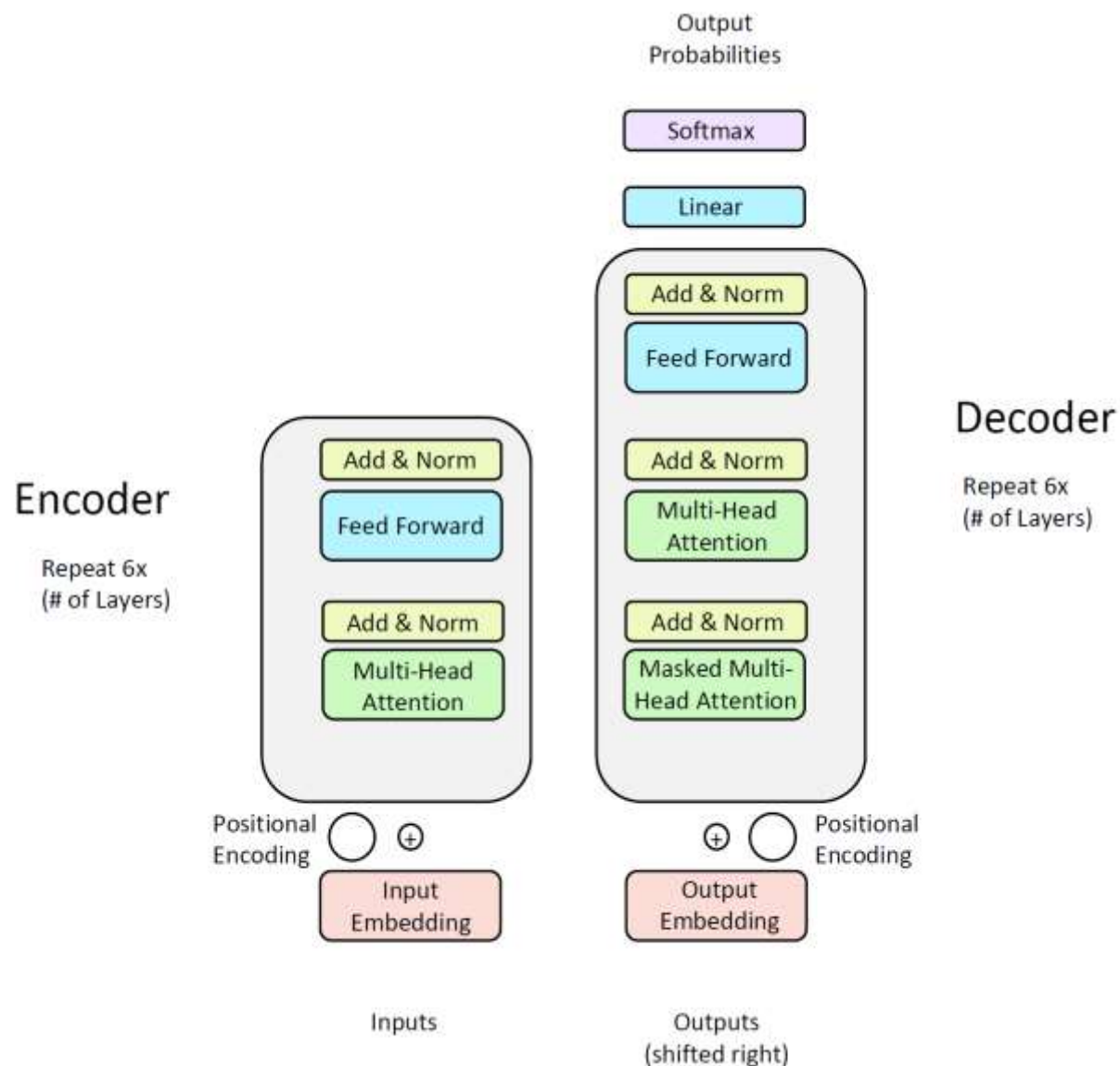
# Transformer Decoder

Core components:

- masked multi-head attention
- encoder-decoder attention

Finishing touches:

- Add a **feedforward layer** (with residual connection + layer norm)
- Add a **final layer** to project the embeddings into a vector of length vocab size (logits)
- Add a **final softmax** to generate a probability distribution of possible next words

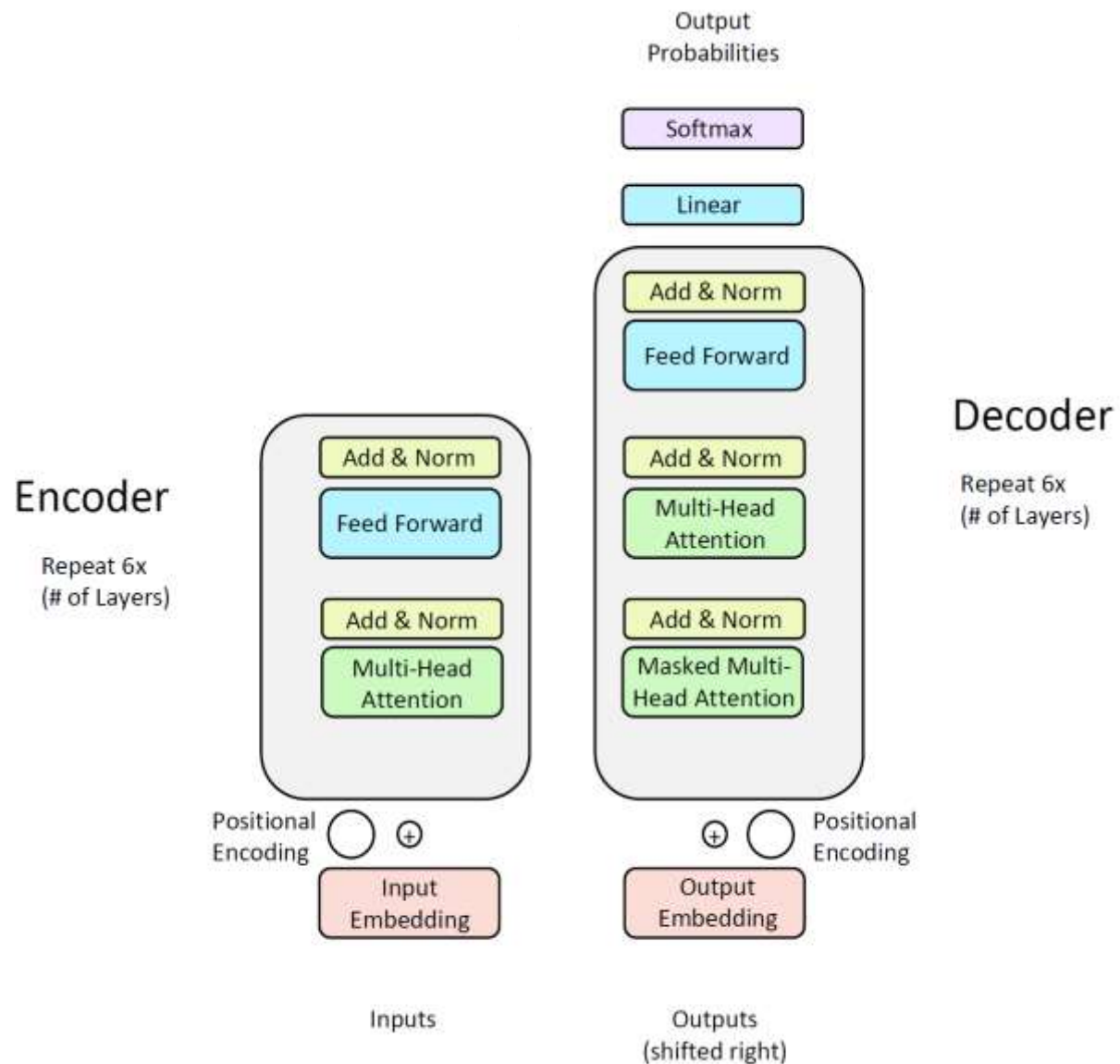




# Transformer Architecture

For more details and a nice implementation example in Pytorch, see [the annotated transformer](#)

Main take-away: Lots of matrix-matrix operations that are efficient to compute on GPUs!





# A very brief introduction to Neural Machine Translation: Training

# Learning Paradigms to Exploit Diverse Data Sources

**Supervised learning** from  
parallel samples (translations)

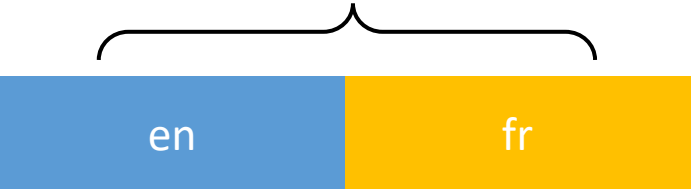


**Unsupervised learning** from  
unpaired monolingual samples



# Supervised Learning

requires parallel samples: input sequence paired with a correct output sequence

$$\theta^* = \operatorname{argmax}_{\theta} \sum_i \log p(f_i | e_i; \theta)$$


The diagram illustrates the relationship between the input and output sequences in the supervised learning equation. A blue box labeled 'en' (English) and a yellow box labeled 'fr' (French) are positioned below the summation index 'i'. A bracket above the summation symbol connects the two boxes, indicating that the summation is over pairs of input and output sequences.

# Unsupervised Learning

Learn a **language model** per language which predicts the next word given previous words in a sentence

fr

$$\theta_f^* = \operatorname{argmax}_{\theta} \sum_f \log p(f_i | f_{<i}; \theta)$$

en

$$\theta_e^* = \operatorname{argmax}_{\theta} \sum_e \log p(e_i | e_{<i}; \theta)$$

# Semi-Supervised Learning

learn from (few) paired + (many) unpaired samples

common approach: **back translation**

use auxiliary Machine Translation (MT) to translate target into source

en	fr
$en^* = MT(fr)$	fr

simple and effective [Sennrich et al. 2016]

# Multilingual Training

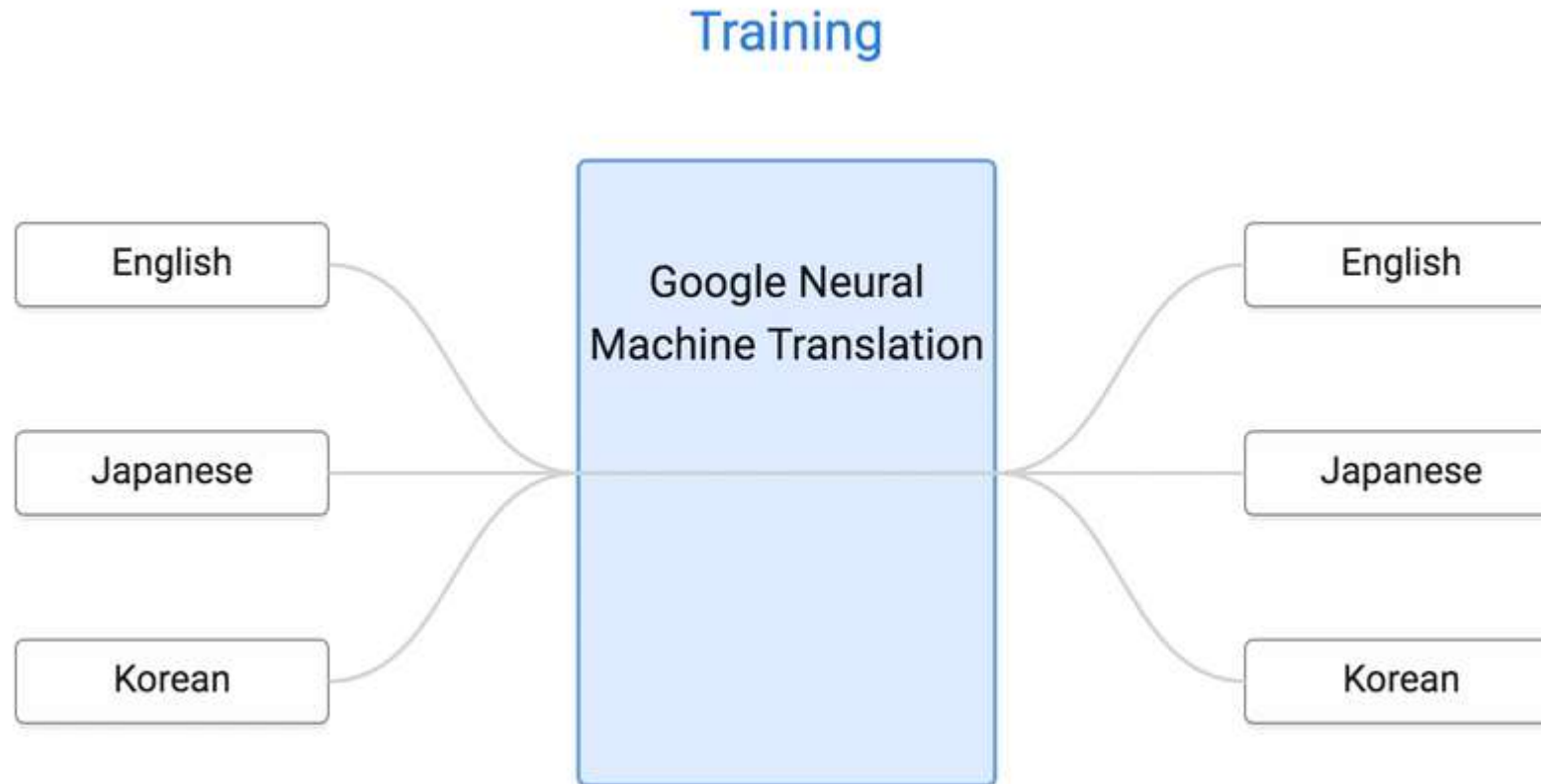


# Multilingual Translation

- Goal: support translation between any N languages
- Naïve approach: build on translation system for each language pair and translation direction
  - Results in  $N^2$  models
  - Impractical computation time
  - Some language pairs have more training data than others
- Can we train a single model instead?

# The Google Multilingual NMT System

[\[Johnson et al. 2017\]](#)





# The Google Multilingual NMT System

[Johnson et al. 2017]

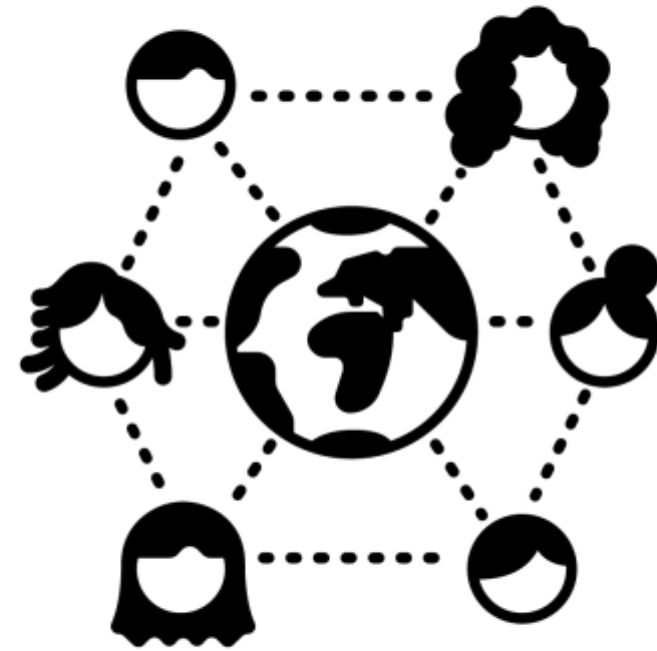
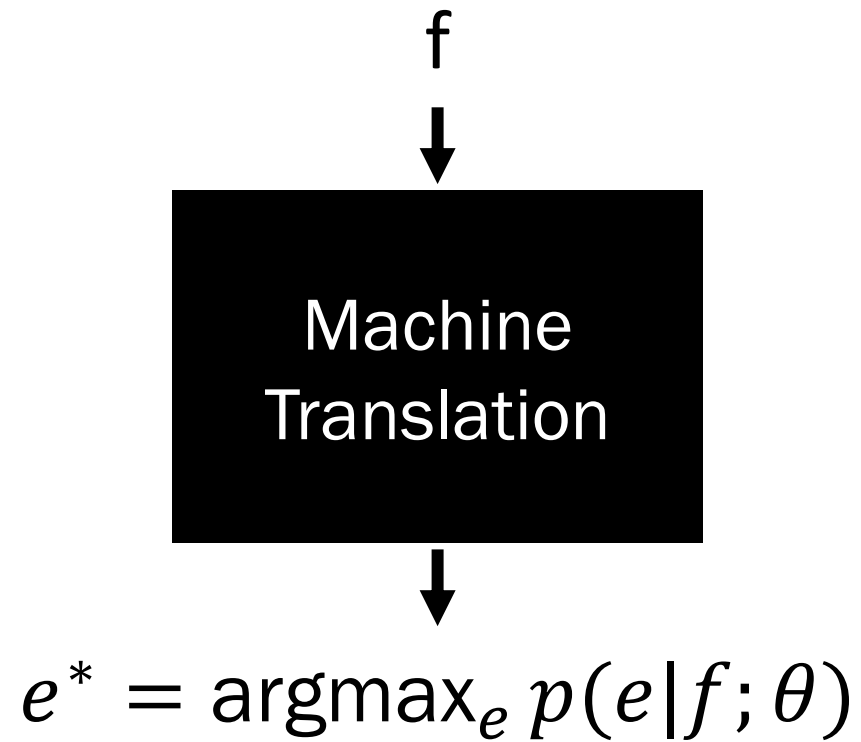
- Shared encoder, shared decoder for all languages
- Train on sentence pairs in all languages
- Add token to the input to mark target language

<2es> Hello, how are you? -> Hola, ¿cómo estás?

Reduces costs: only 1 system to maintain instead of  $N^2$ !

Information sharing across languages can improve translation quality, but also sometimes increases noise

# Human-Centered Machine Translation



from sequence transduction to communication across language barriers

This summer project:  
How can we improve machine  
translation into low-resource languages  
for Wikipedia editors?

# Why Machine Translation for Wikipedia



Wikipedia is the largest multilingual encyclopedia, but its coverage is very unbalanced

- 321 languages as of March 2023
- 94% of editions have fewer than 1M articles
- A Content Translation tool has been available to editors since 2015 to help them port content to their language
  - Yet, the consensus from Wikimedia communities is that “unedited machine translation is worse than nothing”.
- Since 2022, the Content Translation tool is powered by a large open multilingual model from meta that covers many previously underrepresented languages, with varying quality.

# Why Machine Translation for Wikipedia

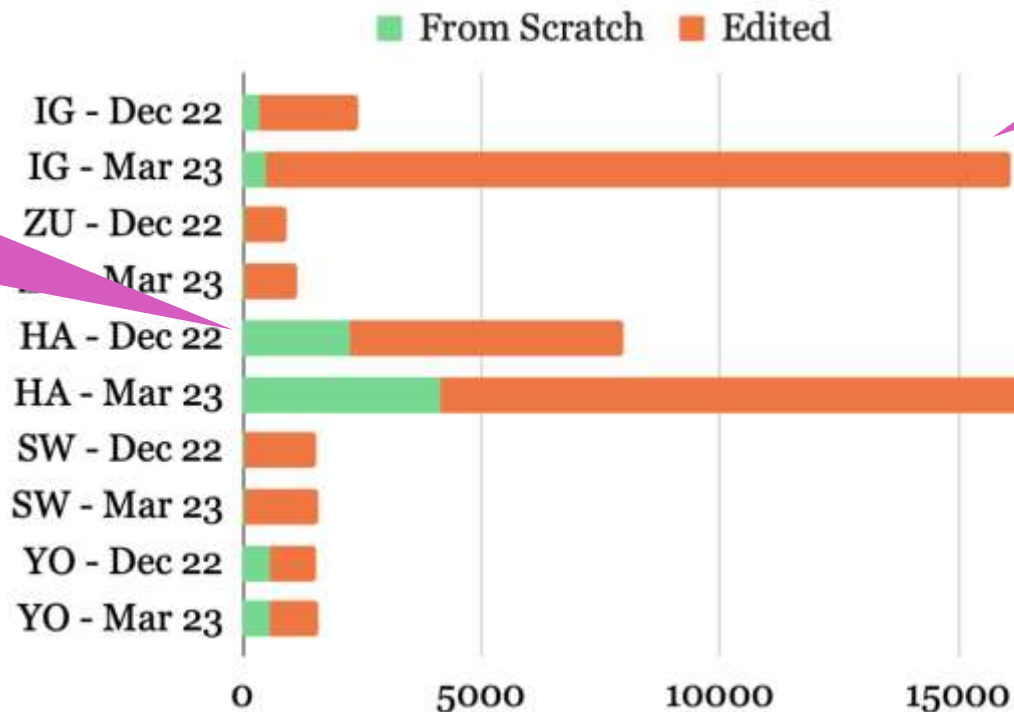


How good are the Content Translation tool outputs for underrepresented languages in Wikipedia?



How can we support editors by making the Content Translation tool as useful as possible? Can we automatically detect errors in outputs to help them decide what to do with them?

# Preliminary Data: How is the Content Translation tool used by editors?

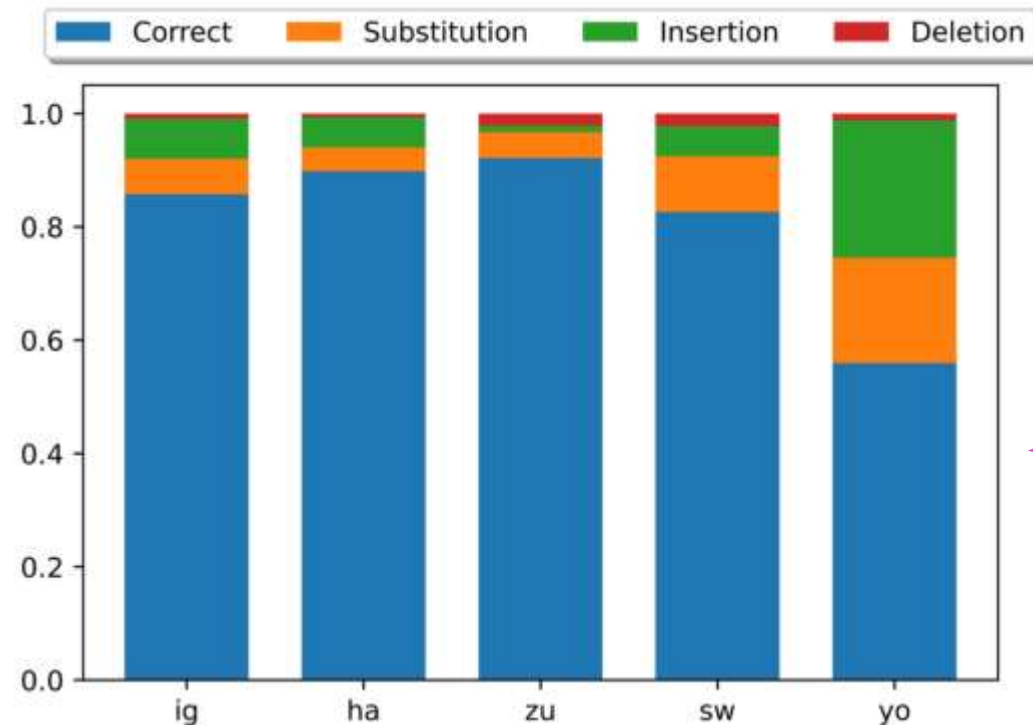


How do they edit translations?

What kinds of inputs do editors translate from scratch?

Number of published translations (roughly at the sentence level) for five underrepresented African languages supported by the ContentTranslation tool based on dumps of December 2022 and March 2023.

# Preliminary Data: How is the Content Translation tool used by editors?



The amount of editing is small. Yet, we know translation quality is low for these languages. So are editors letting errors go through?

Percentage of edit operations when editing machine translation texts for 5 African languages (computed using TER, on Content Translation dumps of March 2023)

# Improving Machine Translation for Wikipedia



Semi-automatically analyze Content Translation tool outputs at scale for diverse language pairs



Benchmark existing automatic methods to assess translation quality and detect errors



Design new Wikipedia-specific automatic methods to assess translation quality and detect errors



# Breaking language barriers in Wikipedia with Neural Machine Translation

Marine Carpuat & Eleftheria Briakou

REU-CAAR

June 9, 2023

