A Brief Introduction to Machine Translation

Excerpt from **CS224N**, Natural Language Processing with Deep Learning, Stanford & **CMSC 723**, Computational Linguistics I, UMIACS

Historical Background

Rule-based & Statistical Machine Translation

Machine Translation

- **Machine Translation (MT)** is the task of translating a sentence *x* from one language (the source language) to a sentence *y* in another language (the target language).

x: L'homme est né libre, et partout il est dans les fers

y: Man is born free, but everywhere he is in chains

Early Machine Translation

- Early 1950s
  - **Rule-based Machine Translation**: Build dictionaries to map words in one language into their counterparts in another language
- Approach:
  - Build dictionaries
  - Write transformation rules
  - Refine, refine, refine

Statistical Machine Translation (SMT)

- 1990s – 2010s
  - **Statistical Machine Translation** (SMT): Learn a **probabilistic model** from data
  - We want to find best English sentence *y*, given French sentence *x*

$$argmax_y P(y|x)$$

  - Use Bayes Rule to break this down into two components to be learnt separately:

$$= argmax_y \underbrace{P(x|y)}\underbrace{P(y)}$$

<u>Translation Model*</u>

Models how words and phrases should be translated (fidelity).
Learnt from parallel data.

<u>Language Model</u>

Models how to write good English (fluency).
Learnt from monolingual data.

$argmax_y P(y|x)$

* Translation Model does not consider order of words.

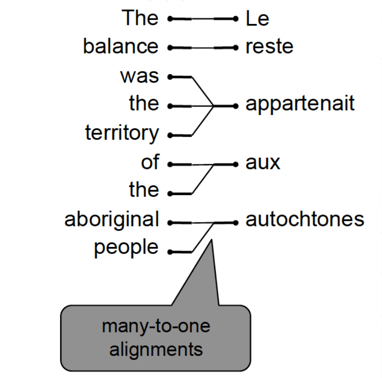
$= argmax_y P(x|y)P(y)$

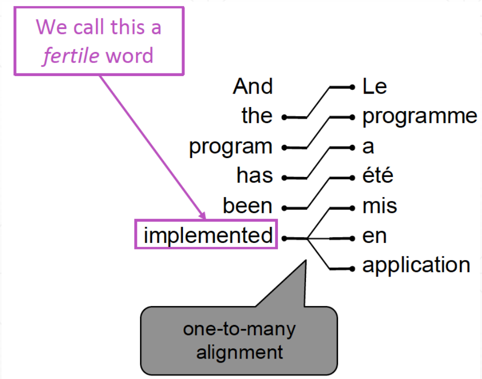
Learning Alignment for SMT

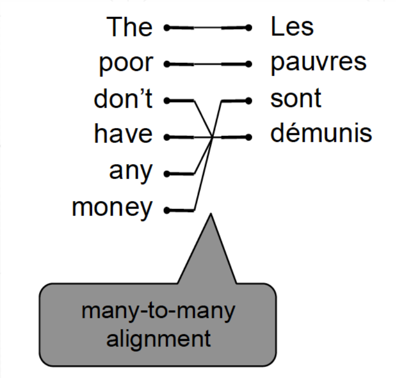
- Question: How to learn translation model $P(x|y)$ from the parallel corpus?
- Break it down further: we actually want to consider

$$P(x, a|y)$$

- where *a* is the alignment, i.e. word-level correspondence between French sentence x and English sentence y
- alignment can be one-to-one, one-to-many or many-to-many







Statistical Machine Translation (SMT)

- SMT was a huge research field
- The best systems were extremely complex
  - Hundreds of important details we haven't mentioned here
  - Systems had many separately-designed subcomponents
  - Lots of feature engineering
    - Need to design features to capture particular language phenomena
  - Require compiling and maintaining extra resources
    - Like tables of equivalent phrases
  - Lots of human effort to maintain
    - Repeated effort for each language pair!

Neural Machine Translation

Sequence-to-sequence model

Neural Machine Translation (NMT)

- Sutskever, I., O. Vinyals, and Q. V. Le. "*Sequence to sequence learning with neural networks.*" *Advances in NIPS* (2014).

- **Neural machine translation** (NMT) is an approach to **machine translation** that uses an **artificial neural network** to predict the likelihood of a sequence of words, typically modeling entire sentences in a single integrated model.

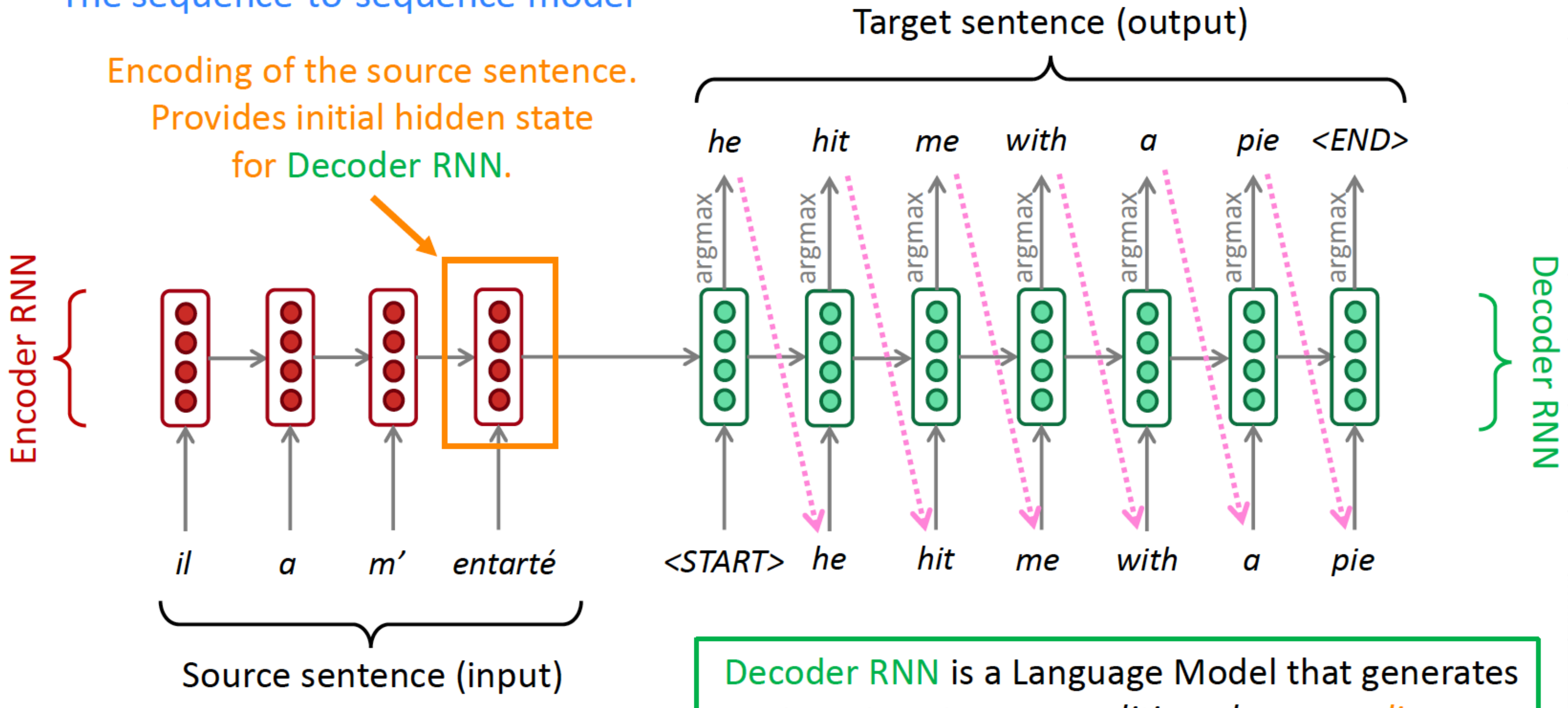
- The neural network architecture is called **sequence-to-sequence** (*aka* seq2seq) and it involves **two RNNs**.

  Encoder RNN **Decoder RNN**

- Sometimes called encoder-decoder network

Sequence to Sequence Model

The sequence-to-sequence model



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

Encoder RNN produces an encoding of the source sentence.

Note: This diagram shows **test time** behavior: decoder output is fed in ······> as next step's input

Sequence to Sequence Model

- Ideally we want to find a (length T) translation y that maximizes

$$P(y|x) = P(y_1|x)P(y_2|\, y_1, x) \ldots P(y_T|y_1, \ldots, y_{T-1}, x)$$
$$= \prod_{t=1}^{T} P(y_t|y_1, \ldots, y_{t-1}, x)$$

Probability of next target word, given target words so far and source sentence *x*

- We could try computing all possible sequences y
  - Far too expensive!
- Beam search decoding*
  - On each step of decoder, keep track of the k most probable partial translations (which we call hypotheses)
  - k is the beam size (in practice around 5 to 10)

* Check CS224N Course for more details of beam search

Neural Machine Translation (NMT)

- **Advantages** of NMT

- Better performance
  - More fluent
  - Better use of context
  - Better use of phrase similarities
- A single neural network to be optimized end-to-end
  - No subcomponents to be individually optimized
- Requires much less human engineering effort
  - No feature engineering
  - Same method for all language pairs

Neural Machine Translation (NMT)

- **Disadvantages** of NMT?

- NMT is less interpretable
  - Hard to debug
- NMT is difficult to control
  - For example, can't easily specify rules or guidelines for translation

NMT: success story of NLP Deep Learning

- Neural Machine Translation went from a fringe research activity in **2014** to the leading standard method in **2016**
  - 2014: First seq2seq paper published
  - 2016: Google Translate switches from SMT to NMT
- SMT systems, built by hundreds of engineers over many years, were outperformed by NMT systems trained by a handful of engineers in a few months
- However, many difficulties still remain
  - Out-of-vocabulary words
  - Domain mismatch between training and test data
  - Maintaining context over longer text
  - Low-resource language pairs

Evaluation

How good is a translation?

Precision & Recall of Words

SYSTEM A: Israeli officials ~~responsibility~~ ~~of~~ airport ~~safety~~

REFERENCE: Israeli officials are responsible for airport security

Precision

$$\frac{\textit{correct}}{\textit{output-length}} = \frac{3}{6} = 50\%$$

Recall

$$\frac{\textit{correct}}{\textit{reference-length}} = \frac{3}{7} = 43\%$$

F-measure

$$\frac{\textit{precision} \times \textit{recall}}{(\textit{precision} + \textit{recall})/2} = \frac{.5 \times .43}{(.5 + .43)/2} = 46\%$$

Precision & Recall of Words

Metric	System A	System B
precision	50%	100%
recall	43%	100%
f-measure	46%	100%

Flaw: no penalty for **re-ordering**

How do we evaluate Machine Translation?

- **BLEU** (**Bi**lingual **E**valuation **U**nderstudy) Metric
  - Papineni, Kishore, et al. "*BLEU: a method for automatic evaluation of machine translation.*" Proceedings of the 40th annual meeting on association for computational linguistics. Association for Computational Linguistics, 2002.

- BLEU compares the machine-written translation to one or several human-written translation(s), and computes a similarity score based on:
  - n-gram precision (usually for 1, 2, 3 and 4-grams)
  - Plus a penalty for too-short system translations

Bilingual Evaluation Understudy (BLEU)

- BLEU compares the machine-written translation to one or several human-written translation(s), and computes a similarity score based on:
  - n-gram precision (usually for 1, 2, 3 and 4-grams)
  - Plus a penalty for too-short system translations

$$BLEU = \underbrace{\min\left(1, \frac{len(output)}{len(reference)}\right)}_{\textbf{Penalty}} \underbrace{\left(\prod_{i=1}^{4} precision_i\right)^{1/4}}_{\textbf{Precision}}$$

Bilingual Evaluation Understudy (BLEU)

SYSTEM A: Israeli officials responsibility of airport safety

2-GRAM MATCH (Israeli officials) · 1-GRAM MATCH (airport)

REFERENCE: Israeli officials are responsible for airport security

SYSTEM B: airport security Israeli officials are responsible

2-GRAM MATCH (airport security) · 4-GRAM MATCH (Israeli officials are responsible)

*One 4-gram match also contains three 2-gram matches & two 3-gram matches

Metric	System A	System B
precision (1gram)	3/6	6/6
precision (2gram)	1/5	4/5
precision (3gram)	0/4	2/4
precision (4gram)	0/3	1/3
brevity penalty	6/7	6/7
BLEU	0%	52%

How do we evaluate Machine Translation?

- BLEU is **useful** but **imperfect**
  - There are many valid ways to translate a sentence
  - So a good translation can get a poor BLEU score because it has low n-gram overlap with the human translation

- Many other metrics
  - GLEU
  - NIST
  - CHRF
  - METEOR
  - ...

Thanks

Q & A