MH6812:
Advanced Natural Language Processing with Deep Learning

Lecture 8: Seq2Seq Models, Attentions, and Subword

Instructor: Luu Anh Tuan
Email: anhtuan.luu@ntu.edu.sg
Office: #N4-02b-66

Where we are

Models/Algorithms

- Linear models
- Feed-forward Neural Nets (FNN)
- Window-based methods
- Convolutional Nets
- Recurrent Neural Nets
- Seq2Seq

NLP tasks/applications

- Word meaning
- Language modelling
- Sequence tagging
- Sequence encoding
- Machine Translation

Before Recess

Today's Plan

Today (lecture 8)

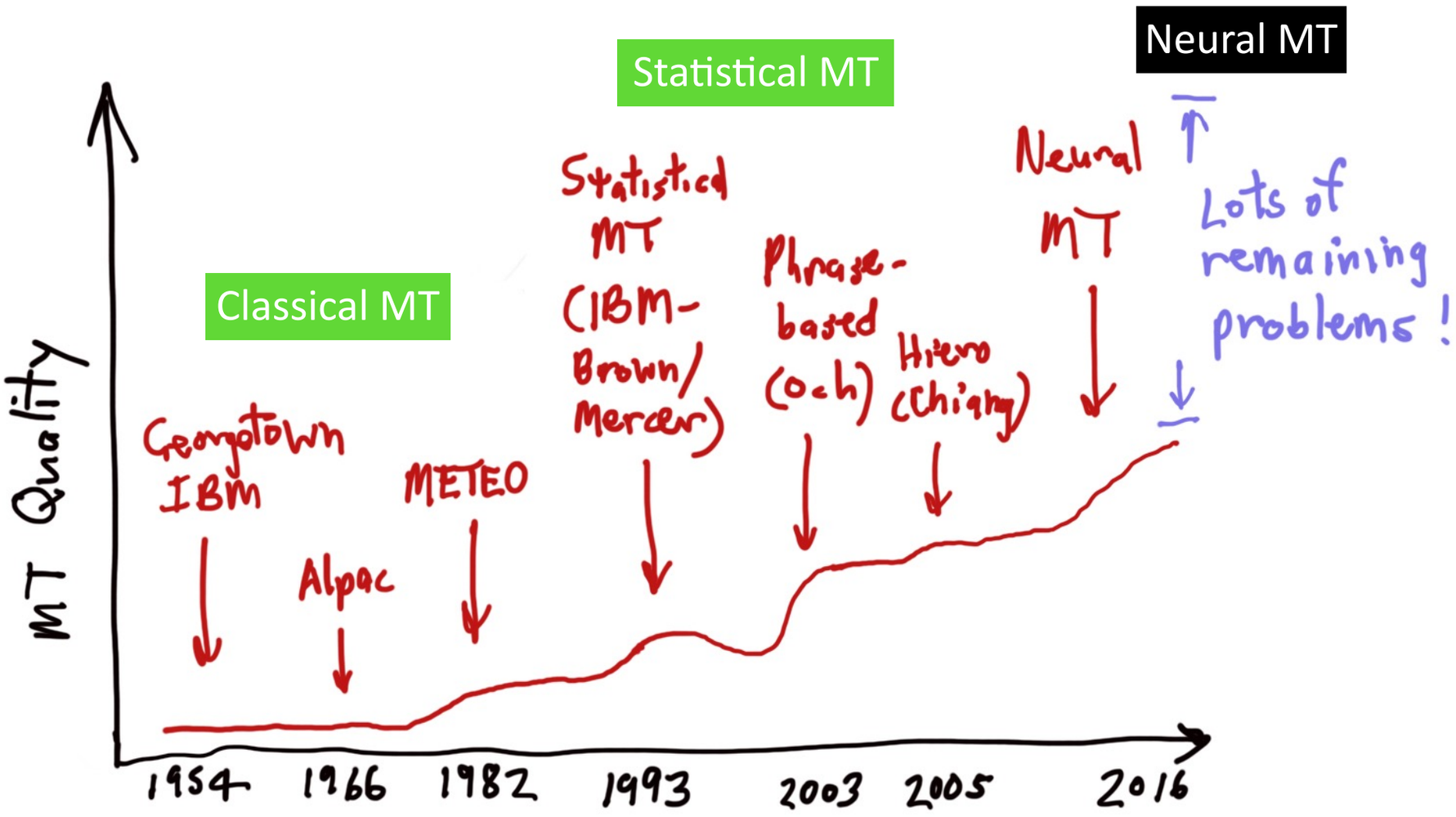
+ Attention

+ Subword

NLP tasks/applications

- **Machine Translation**
- Summarization
- Parsing
- Dialogue generation

Progress in MT



- Microsoft, Google, Yandex claimed **human parity** in MT in 2018 with NMT.

Only true in a controlled setup

Source: Luong, Cho, Manning (ACL tutorial 2016)

Neural Machine Translation (2014 -)

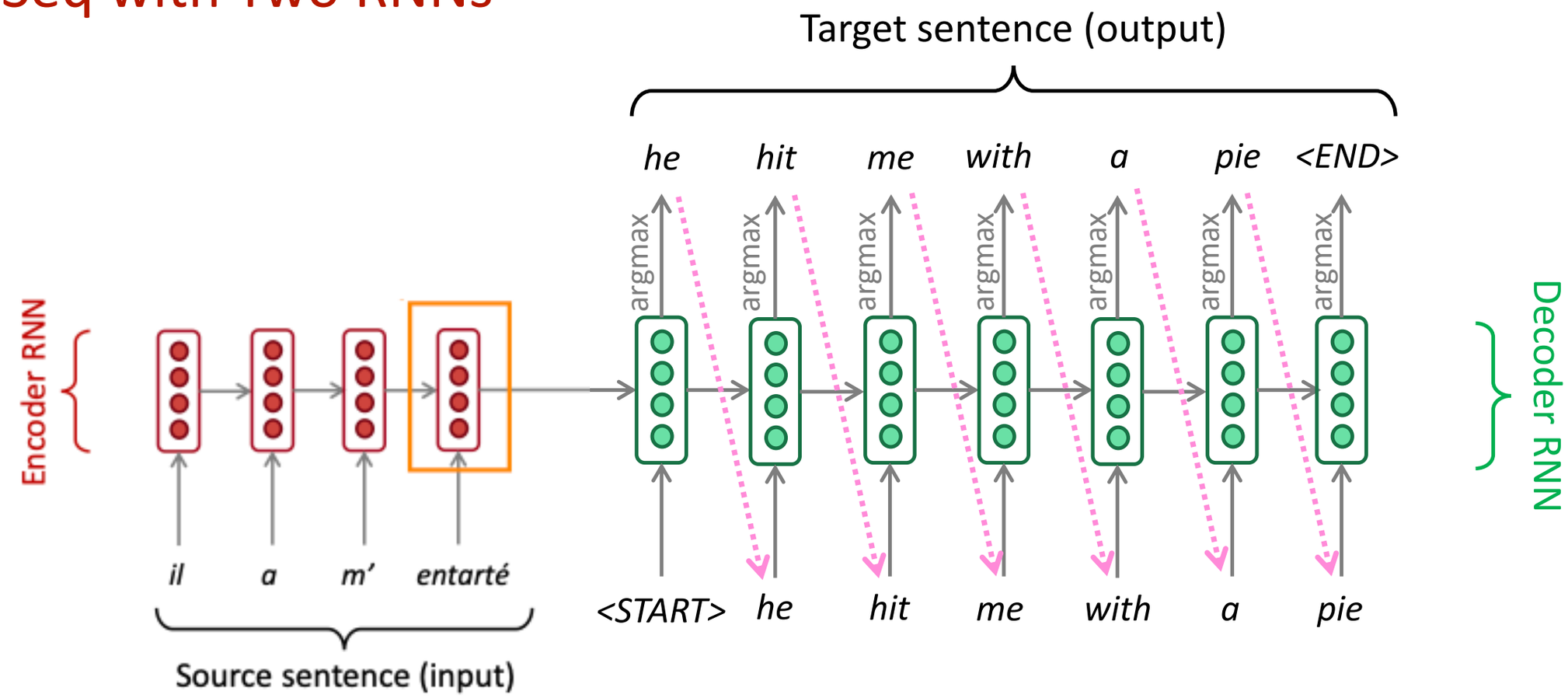
- New paradigm

- One Network for everything!

Photo credit: Internet

Sequence-to-sequence NMT Model

- Seq2Seq with Two RNNs

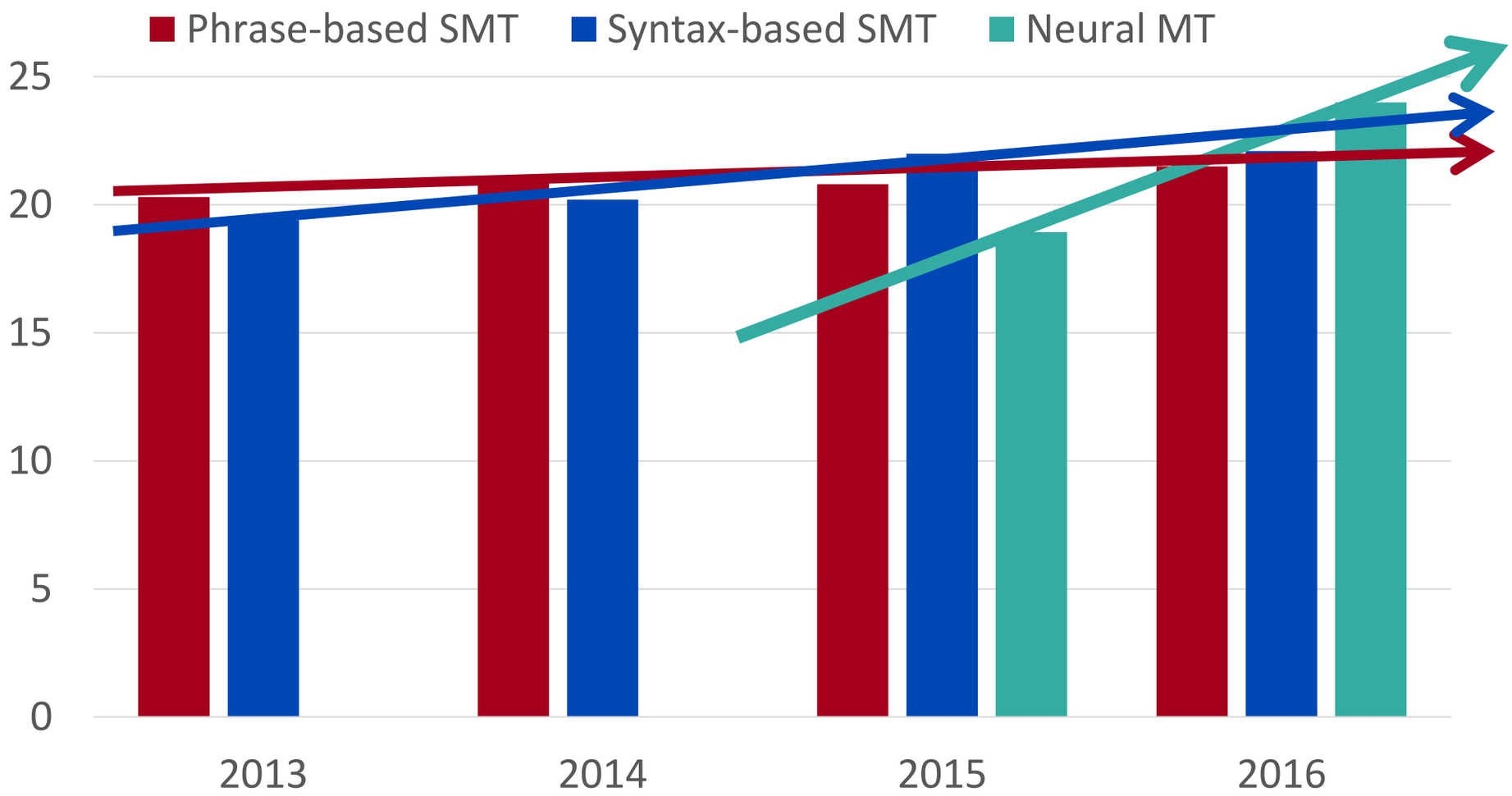


Encoder produces an encoding of the source sentence.

Provides Initial hidden state for Decoder

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

NMT Success



Source: http://www.meta-net.eu/events/meta-forum-2016/slides/09_sennrich.pdf

NMT Success

NMT: the biggest 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 [1]
- **2016**: Google Translate switches from SMT to NMT
- This is amazing! **SMT** systems, built by hundreds of engineers over many years, outperformed by NMT systems trained by a handful of engineers in a few months

[1] Sequence to Sequence Learning with Neural Networks. Ilya Sutskever, Oriol Vinyals, Quoc V. Le. NIPS 2014

Is MT Solved?

- Is MT solved?
  - Nope!
  - Although there are claims from Google, Microsoft, Yandex that their MT systems have achieved human parity, this is true for only constrained setups [1,2].

[1] Samuel Läubli, Rico Sennrich, and Martin Volk. Has machine translation achieved human parity? a case for document-level evaluation. In EMNLP, 2018.

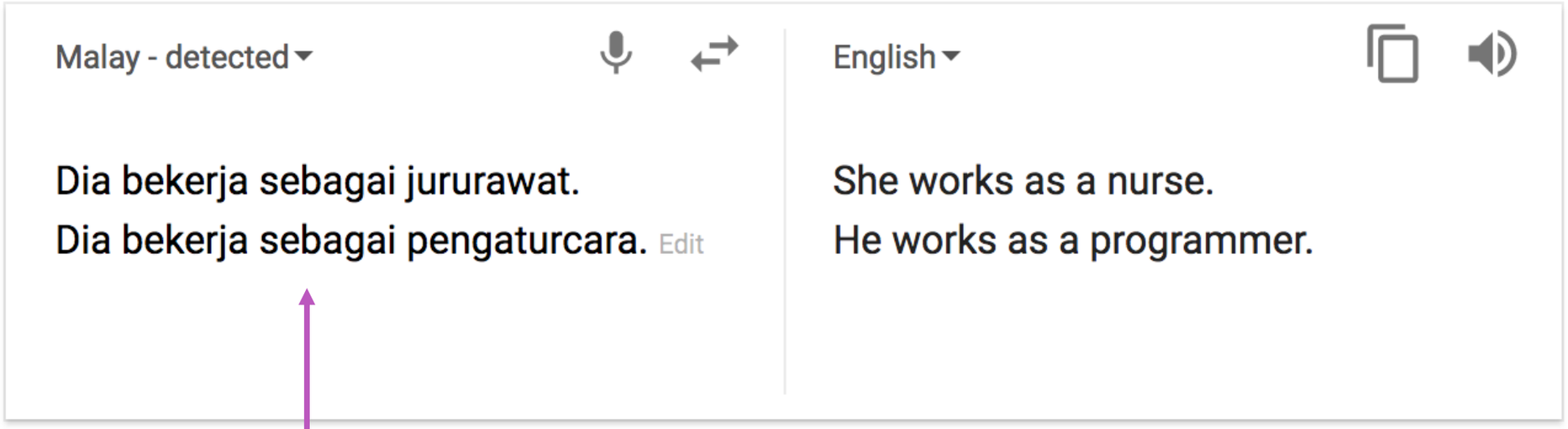
[2] M. Popel, M. Tomková, J. Tomek, Lukasz Kaiser, Jakob Uszkoreit, Ondrej Bojar, and Z. Žabokrt-ský. Transforming machine translation: a deep learning system reaches news translation qualitycomparable to human professionals. Nature Communications, 11, 2020.

Is MT Solved?

- Still many difficulties:
  - Data hungry! Has been successful for high-resource languages
    - How to deal with low-resource languages (or domains)?
    - Most languages are low-resource
  - Dealing with out-of-vocabulary words
    - Input and Output dictionaries need to be fixed and limited
  - Maintaining longer context
    - Discourse-level aspects (anaphora, connectives)
  - Domain mismatch between train and test data
  - How to interpret
  - Translation speed

Is MT Solved?

- Nope!
- **Dataset biases!**



Didn’t specify gender

Source: Stanford 224n

Is MT Solved?

- Nope!
- **Dataset biases (low-resource)!**

Is MT Solved?

- Nope!
- **OOV word!**

BBC Sign in News Sport Reel Worklife Travel Future More

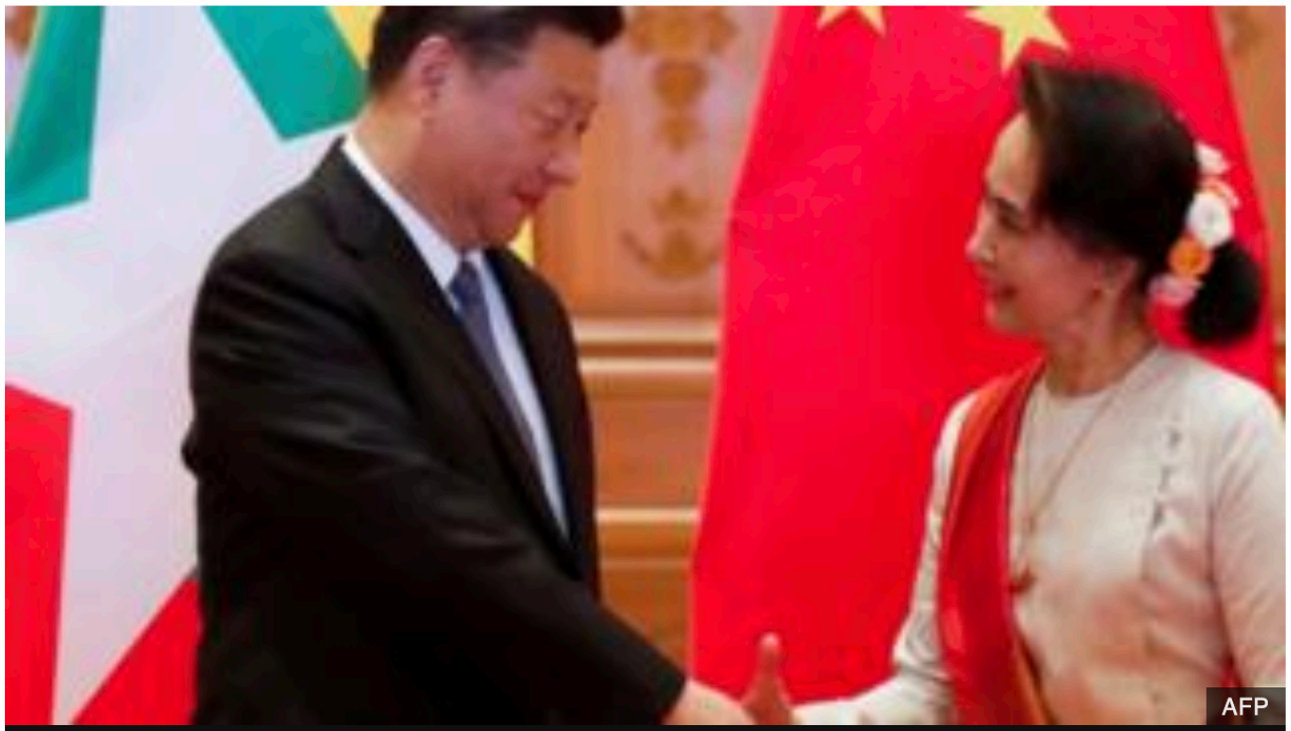
NEWS

Home Video World Asia UK Business Tech Science Stories Entertainment & Arts

Asia China India

Facebook blames 'technical issue' for offensive Xi Jinping translation

19 January 2020 Share

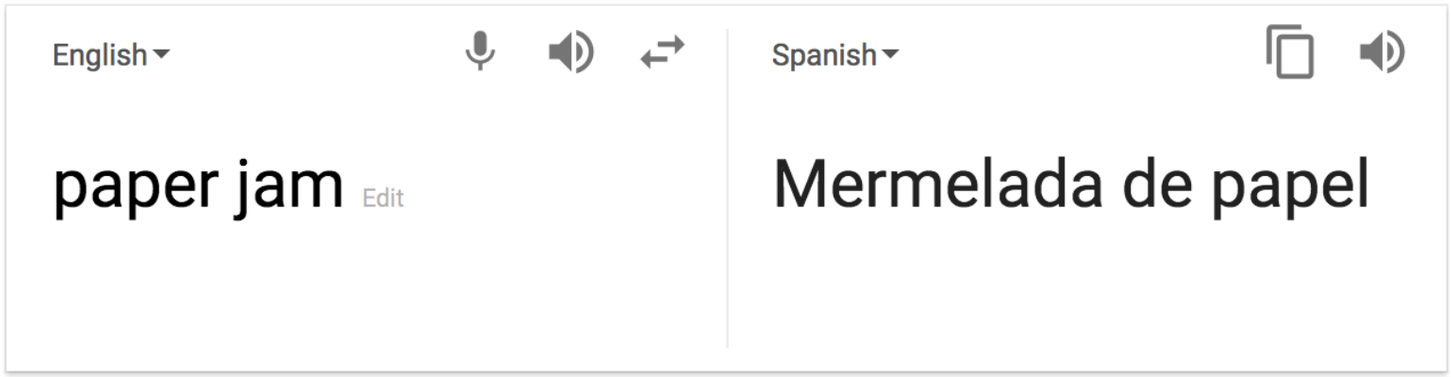


Facebook blamed a "technical issue" for the mistranslation of Xi Jinping's name

Facebook has apologised for translating Chinese President Xi Jinping's name from Burmese to English into an obscenity on its platform.

Is MT Solved?

- Nope!
- Incorporating **common sense knowledge is still hard**

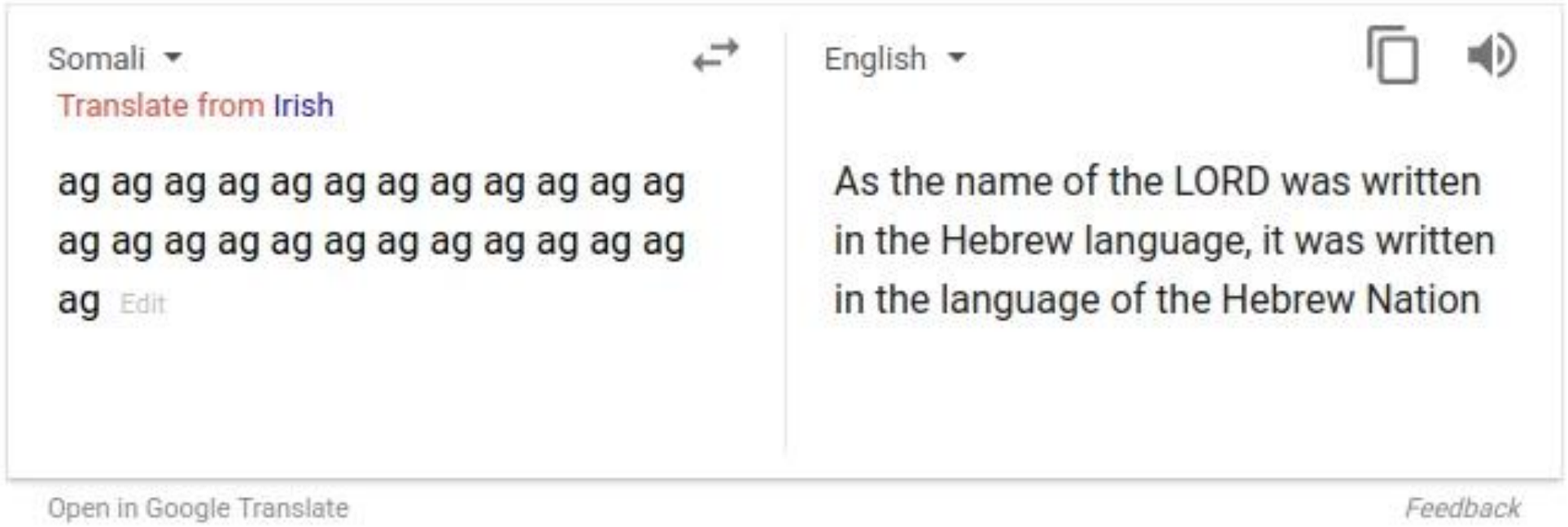


?

Source: Stanford 224n

Is MT Solved?

- Nope!
- **Uninterpretable systems do strange things**



Picture source: https://www.vice.com/en_uk/article/j5npeg/why-is-google-translate-spitting-out-sinister-religious-prophecies

Explanation: https://www.skynettoday.com/briefs/google-nmt-prophecies \

NMT Research Continues!

- NMT has been the **flagship task** for NLP Deep Learning
- NMT research has pioneered many of the recent innovations of Deep Learning (not just in NLP)
- In **2017-2021:** NMT research continues to thrive
  - Researchers have found *many, many* improvements to the "vanilla" seq2seq NMT system we've presented
- But two improvements are integral:

1. Attention
2. Subword

Attention Mechanism

Breakthrough

BLOG POST
RESEARCH

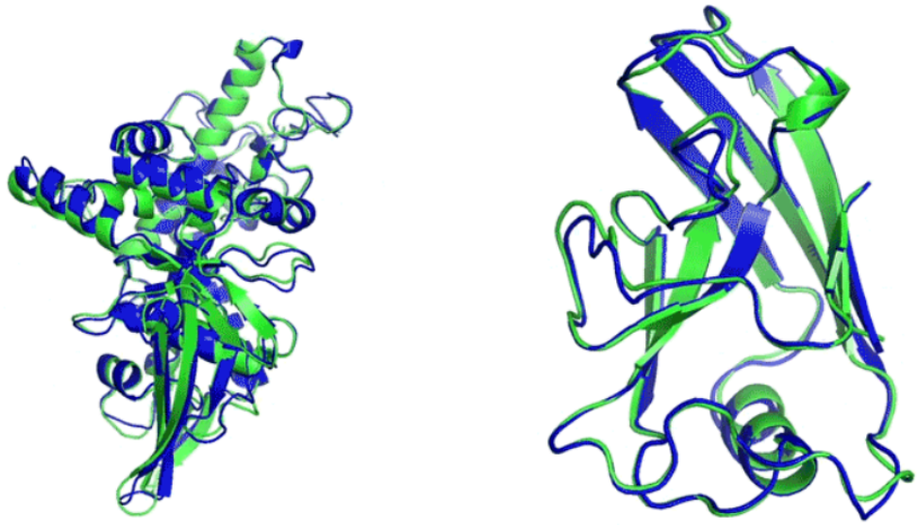
30 NOV 2020

AlphaFold: a solution to a 50-year-old grand challenge in biology

We have been stuck on this one problem – how do proteins fold up – for nearly 50 years. To see DeepMind produce a solution for this, having worked personally on this problem for so long and after so many stops and starts, wondering if we'd ever get there, is a very special moment.

PROFESSOR JOHN MOULT
CO-FOUNDER AND CHAIR OF CASP, UNIVERSITY OF MARYLAND

3D structure



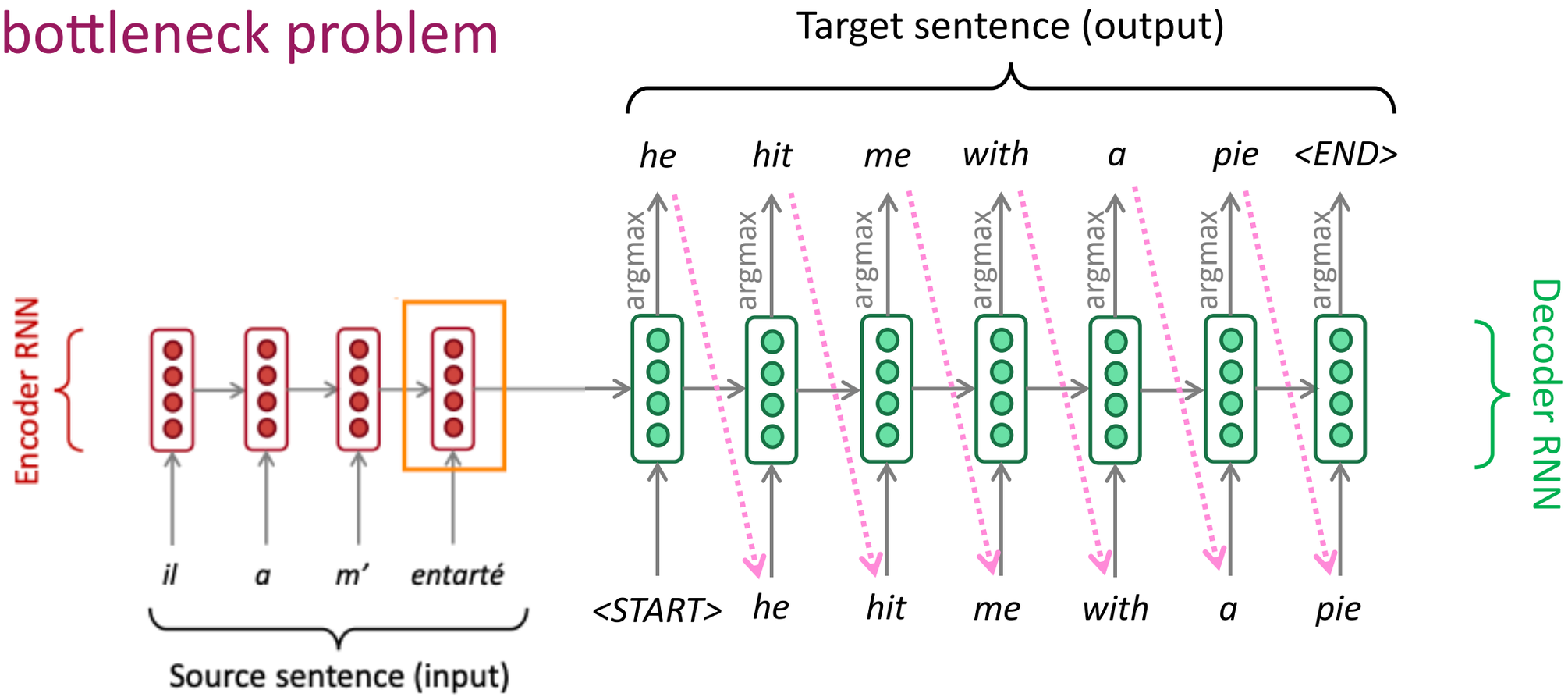
*"Improvement over the first version of AlphaFold is mostly usage of **transformer/attention mechanisms** applied to residue space and combining it with the working ideas from the first version"*

Attention/Transformer was proposed first for NMT!

https://deepmind.com/blog/article/alphafold-a-solution-to-a-50-year-old-grand-challenge-in-biology

Sequence-to-sequence Model

- The bottleneck problem



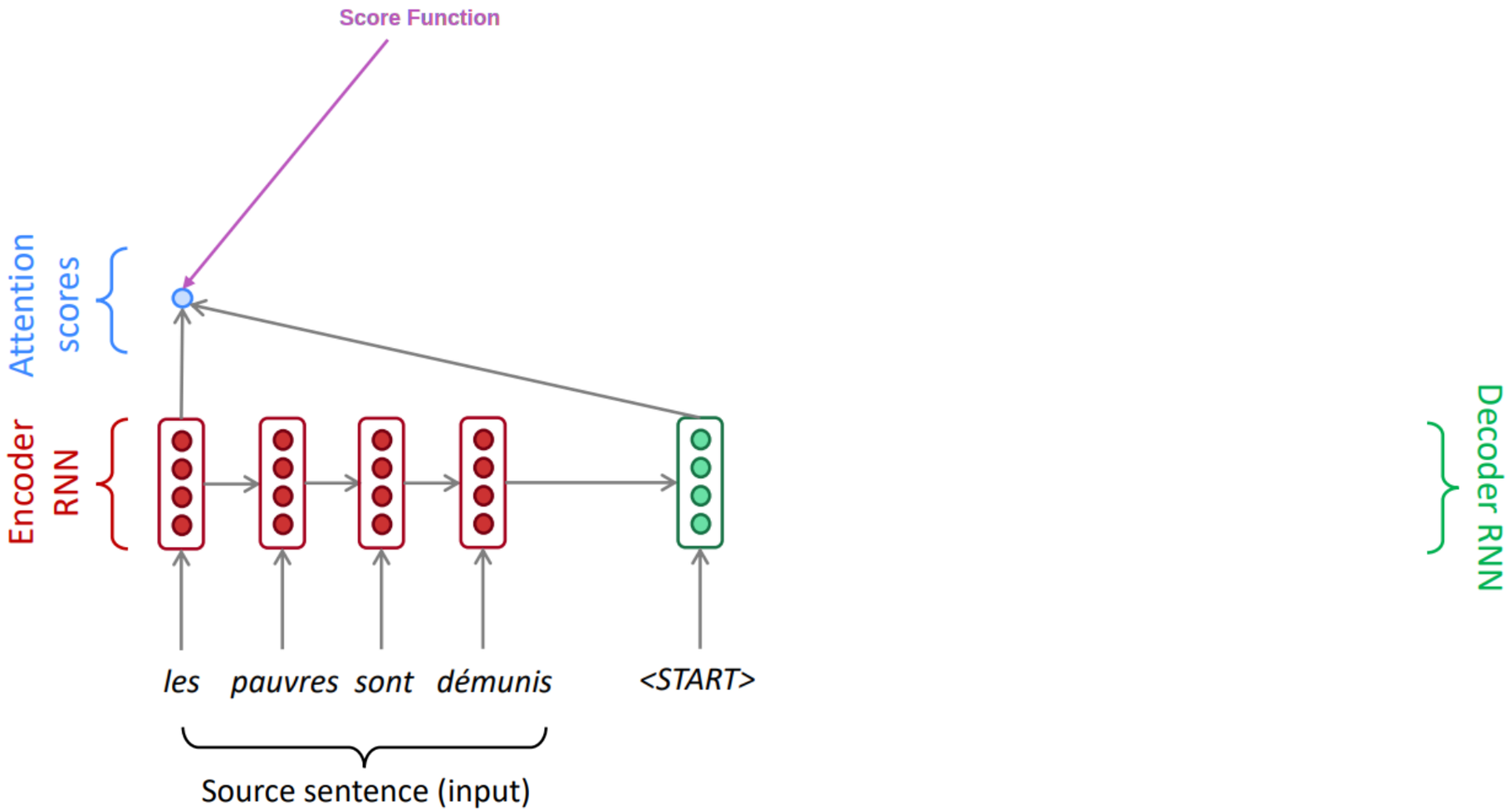
Encoding of the source sentence. This needs to capture *all information* about the source sentence. Information bottleneck!

Source: Stanford 224n

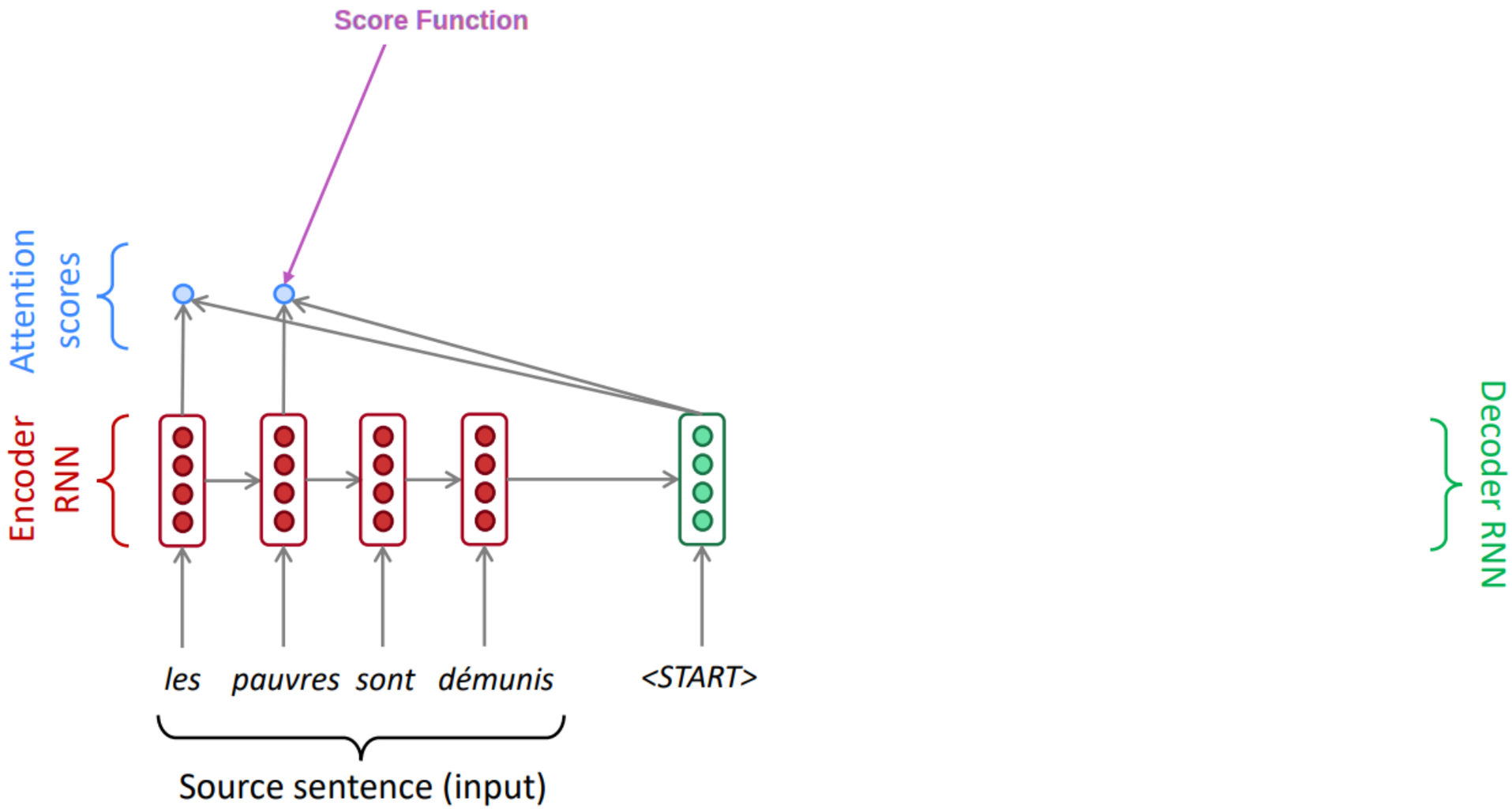
Attention Mechanism

- **Attention** provides a solution to the bottleneck problem.
- **Core idea:** on each step of the decoder, use *direct connection to the encoder* to *focus on the relevant part* of the source sequence

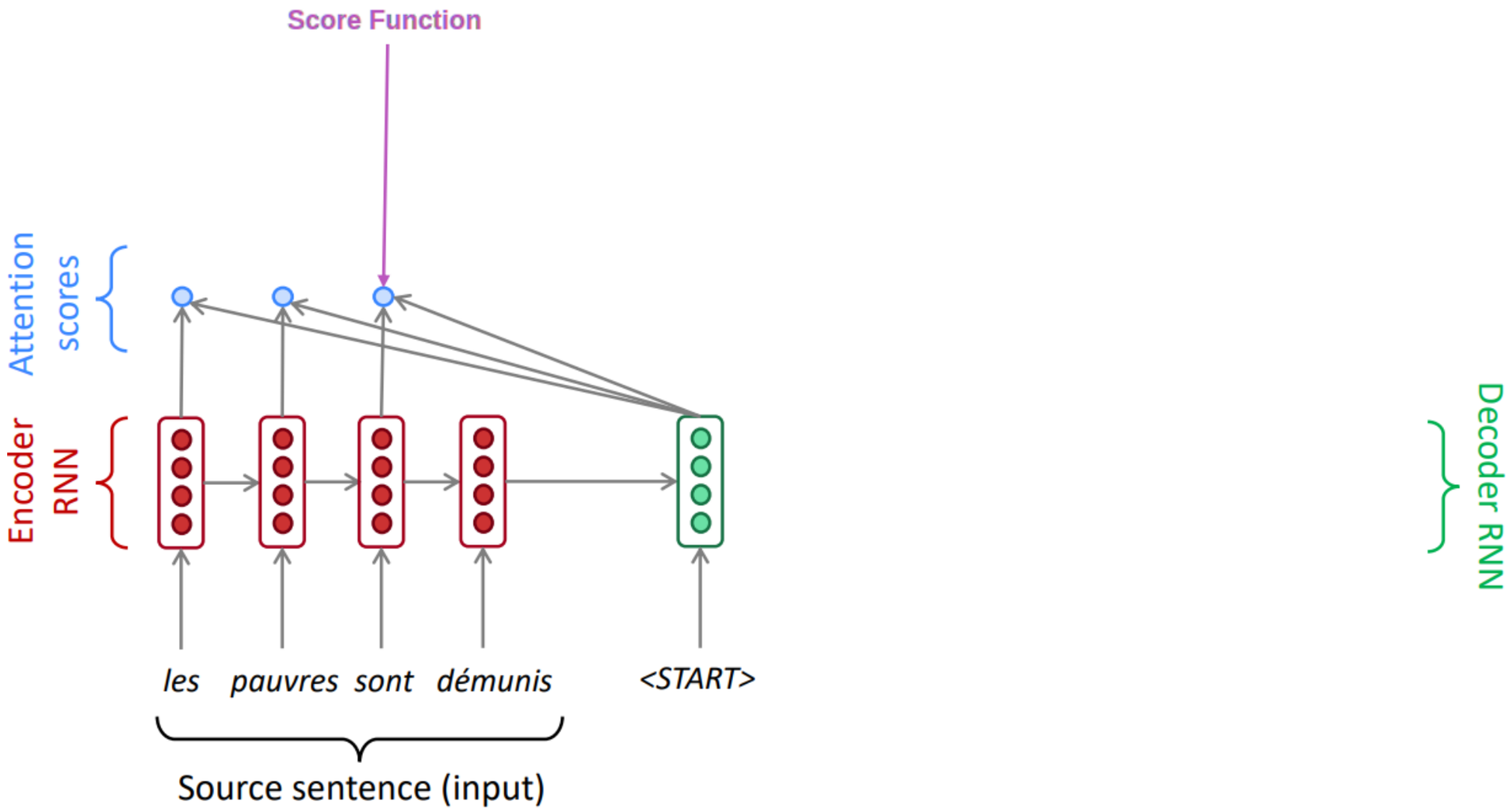
Attention Mechanism



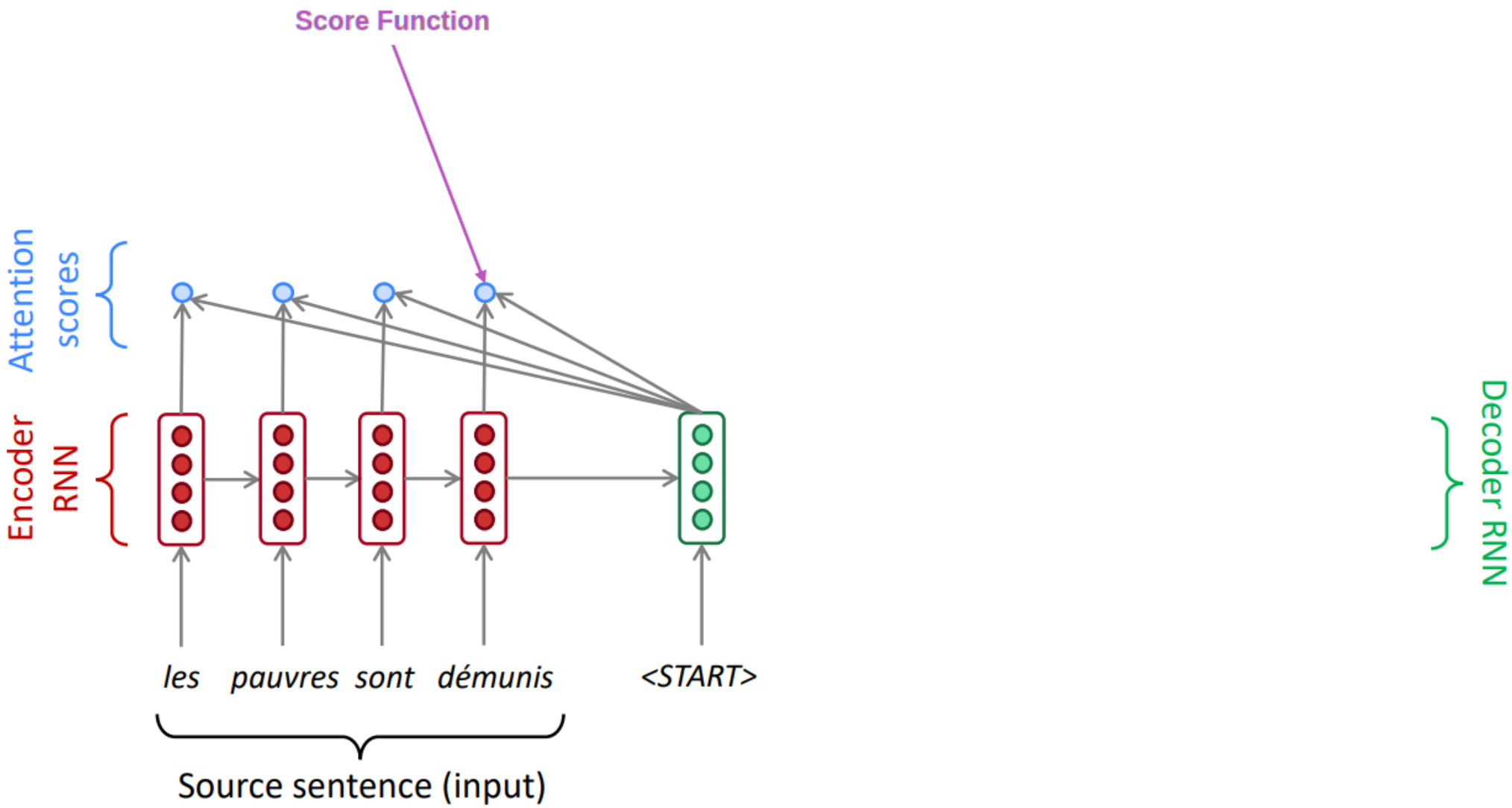
Attention Mechanism



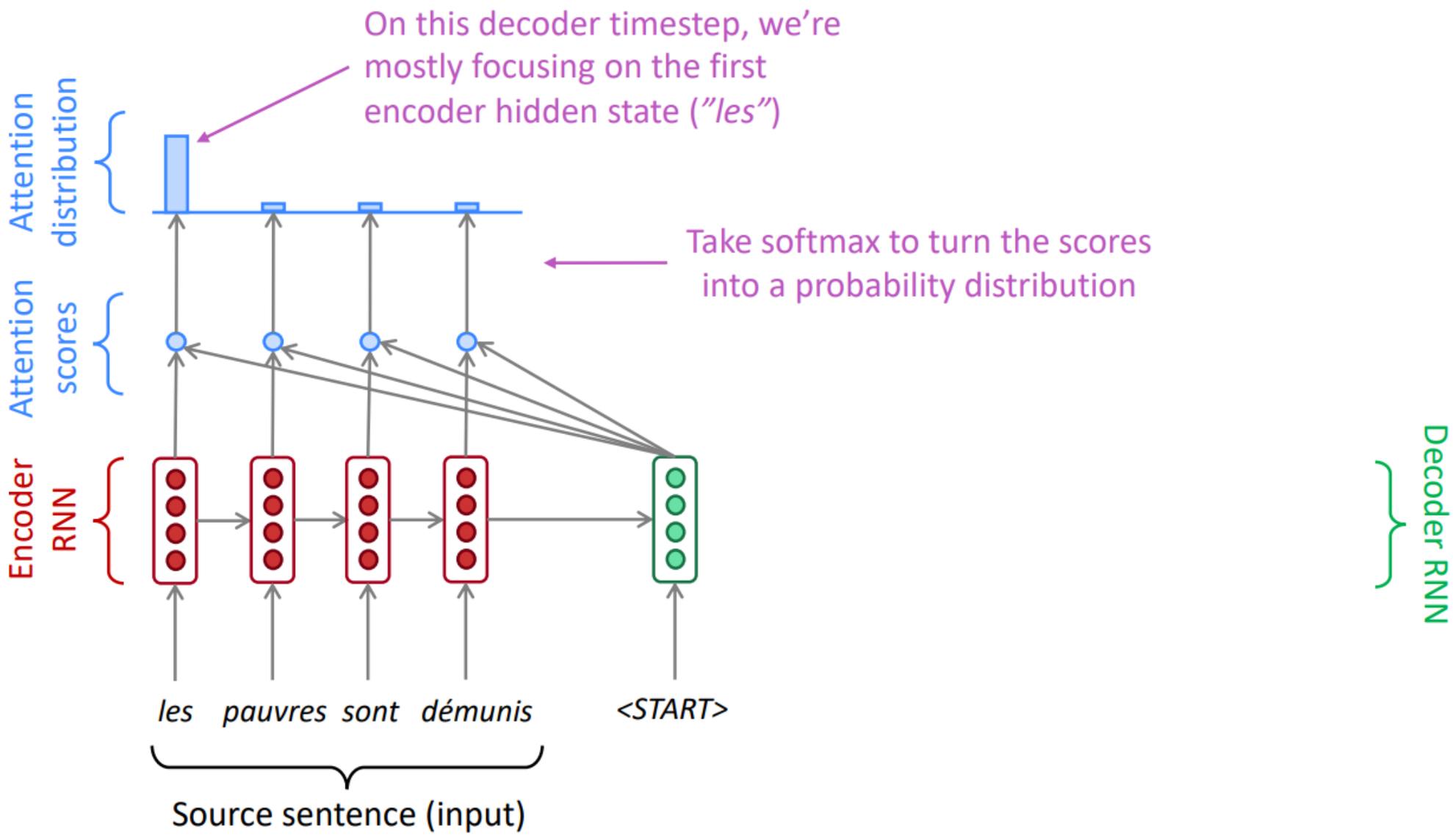
Attention Mechanism



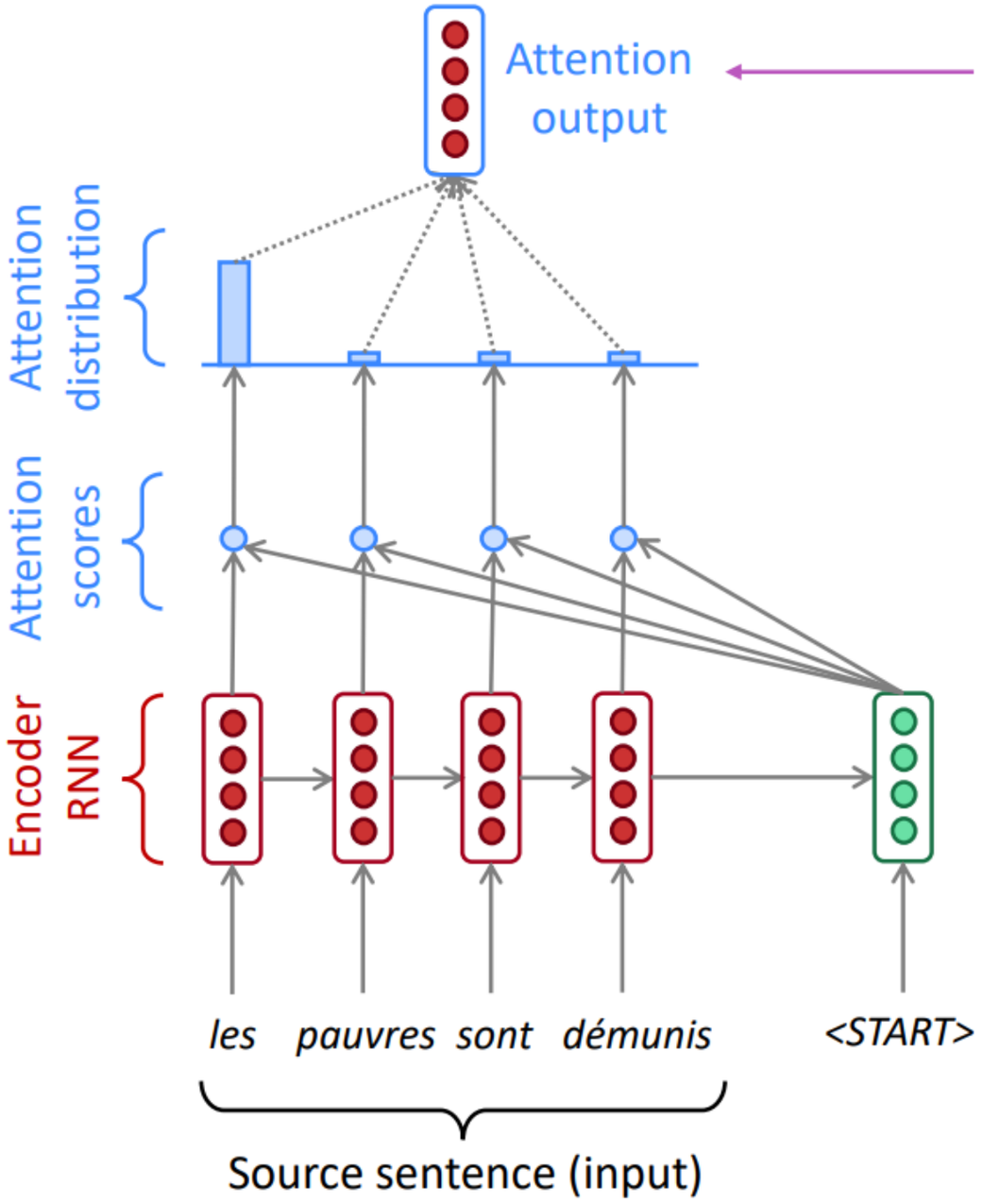
Attention Mechanism



Attention Mechanism



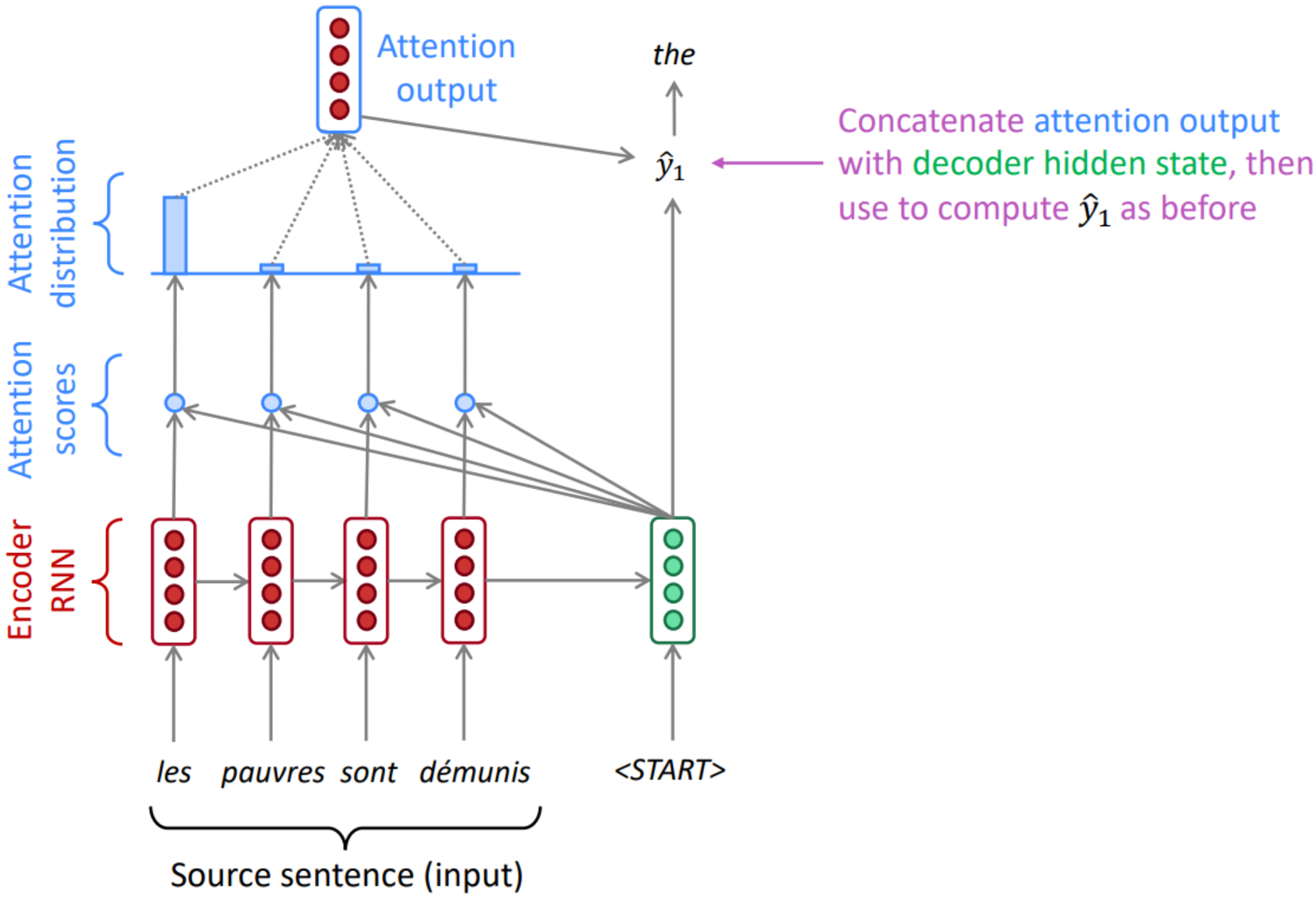
Attention Mechanism



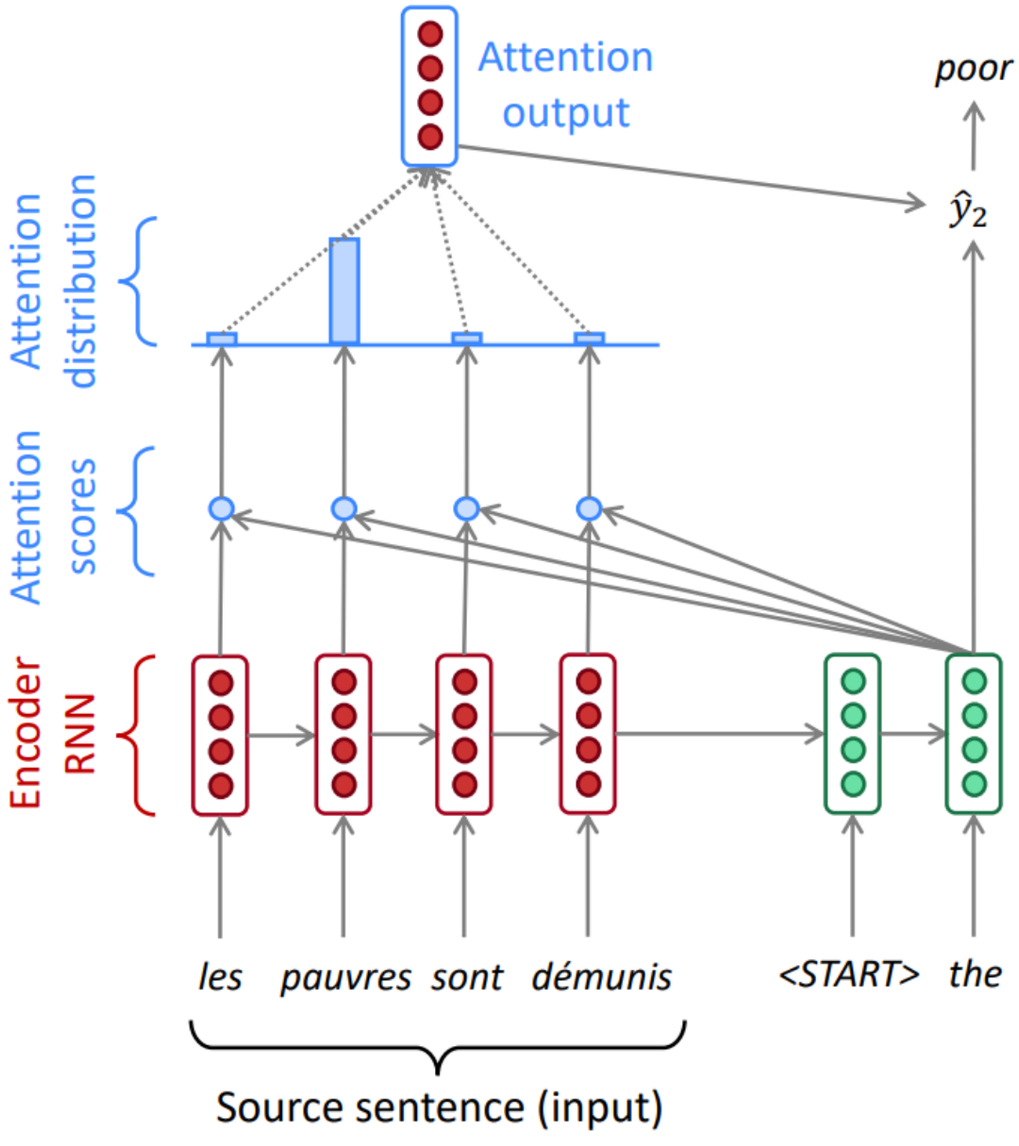
Use the attention distribution to take a **weighted sum** of the encoder hidden states.

The attention output mostly contains information the hidden states that received high attention.

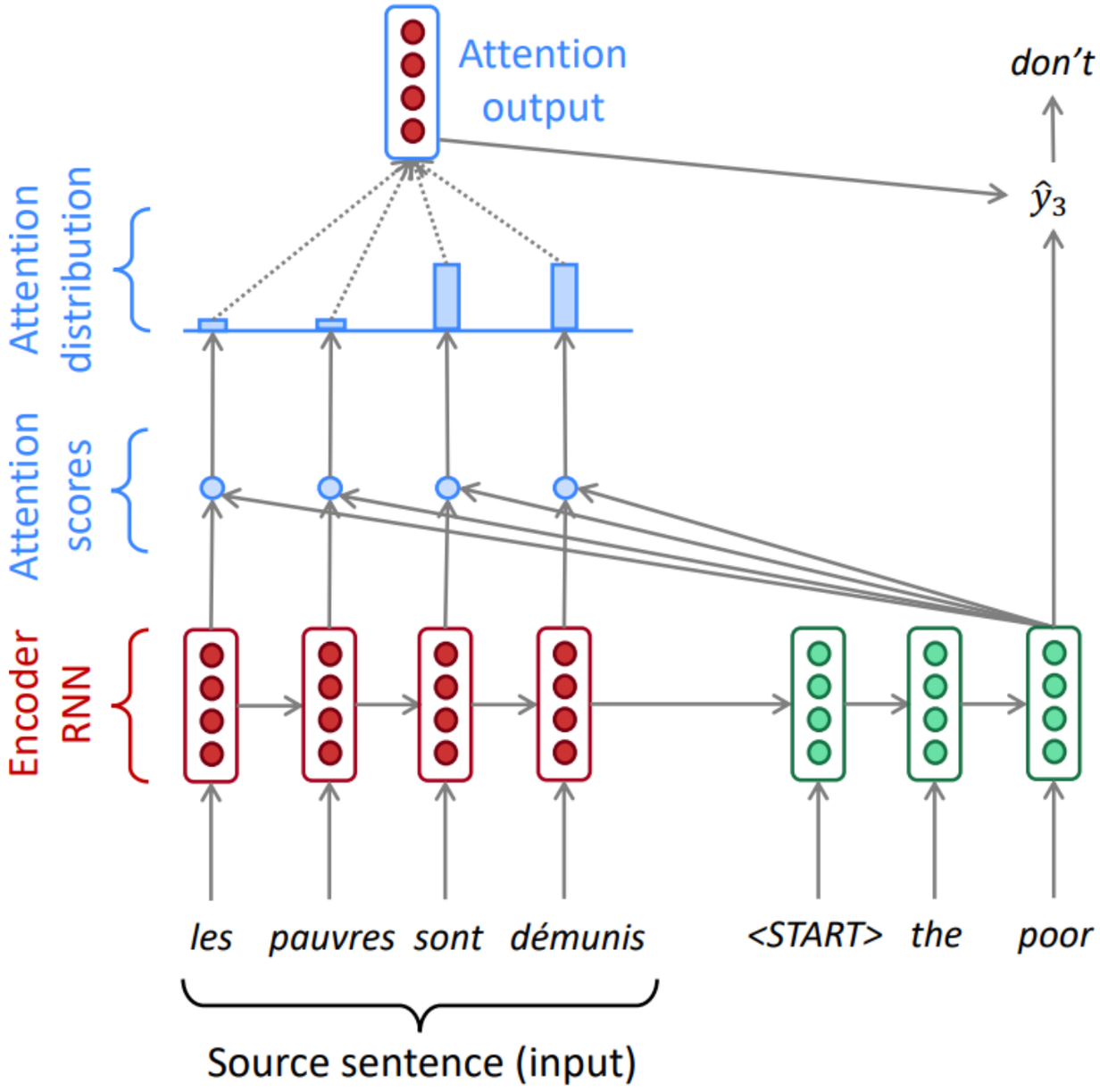
Attention Mechanism



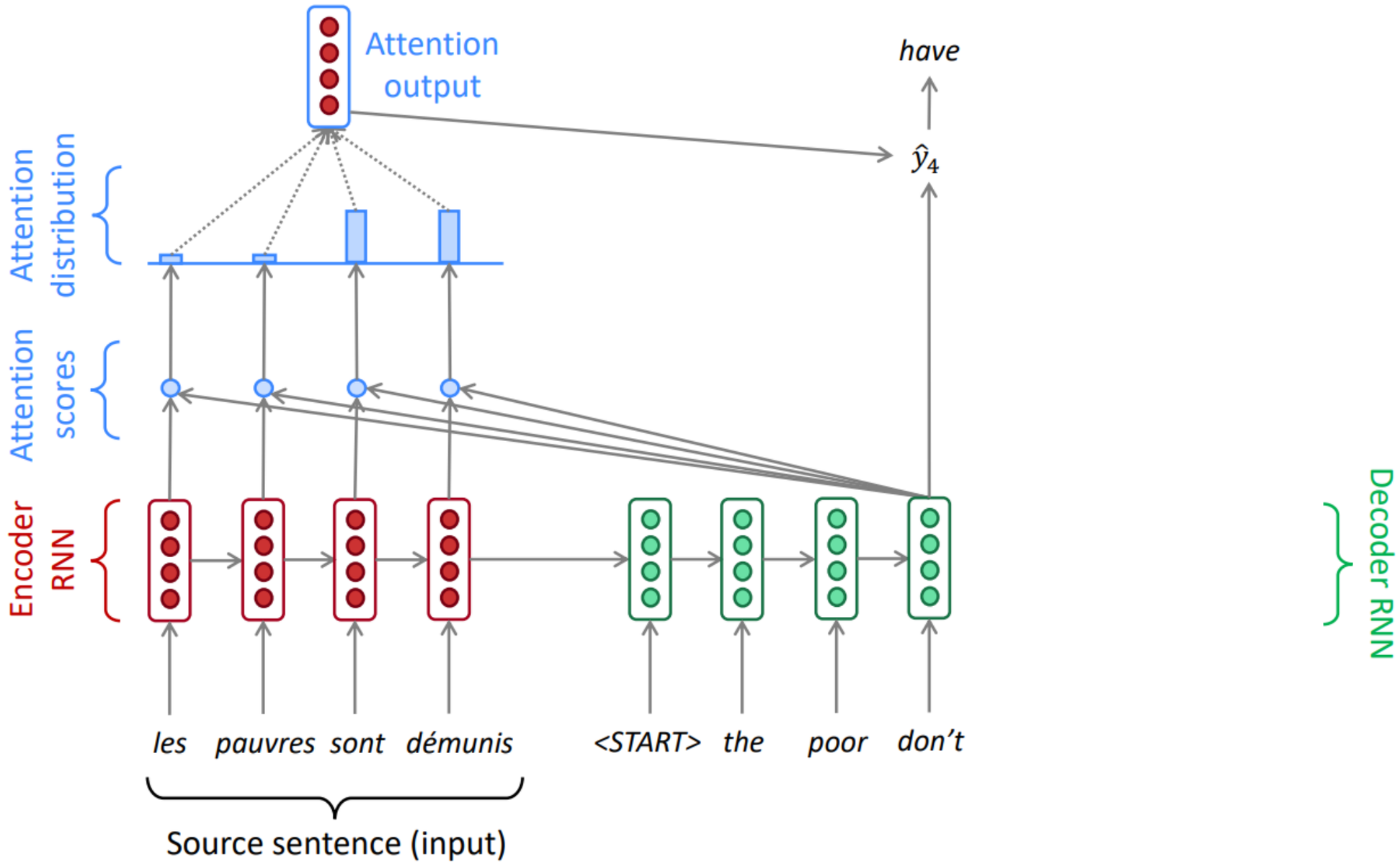
Attention Mechanism



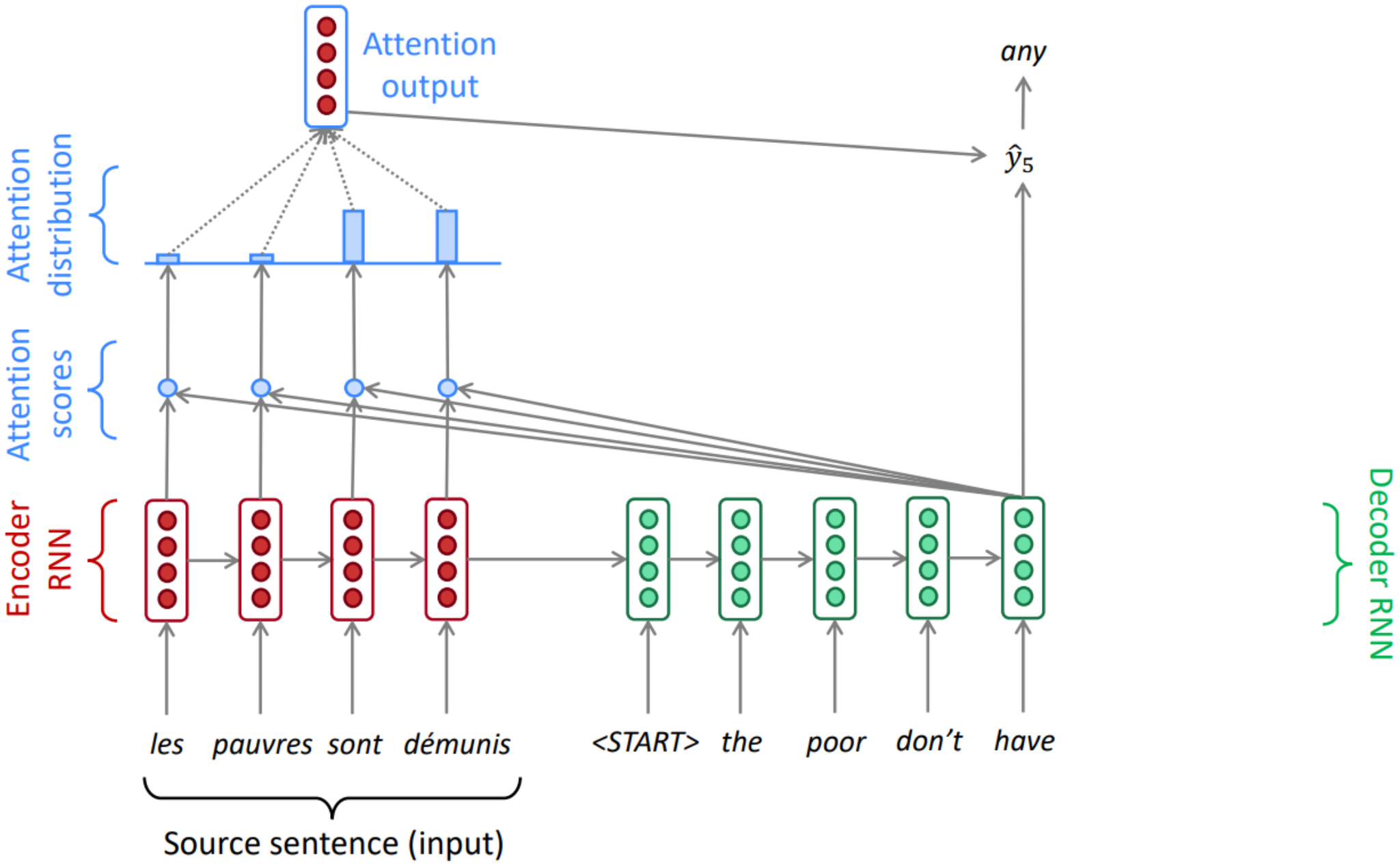
Attention Mechanism



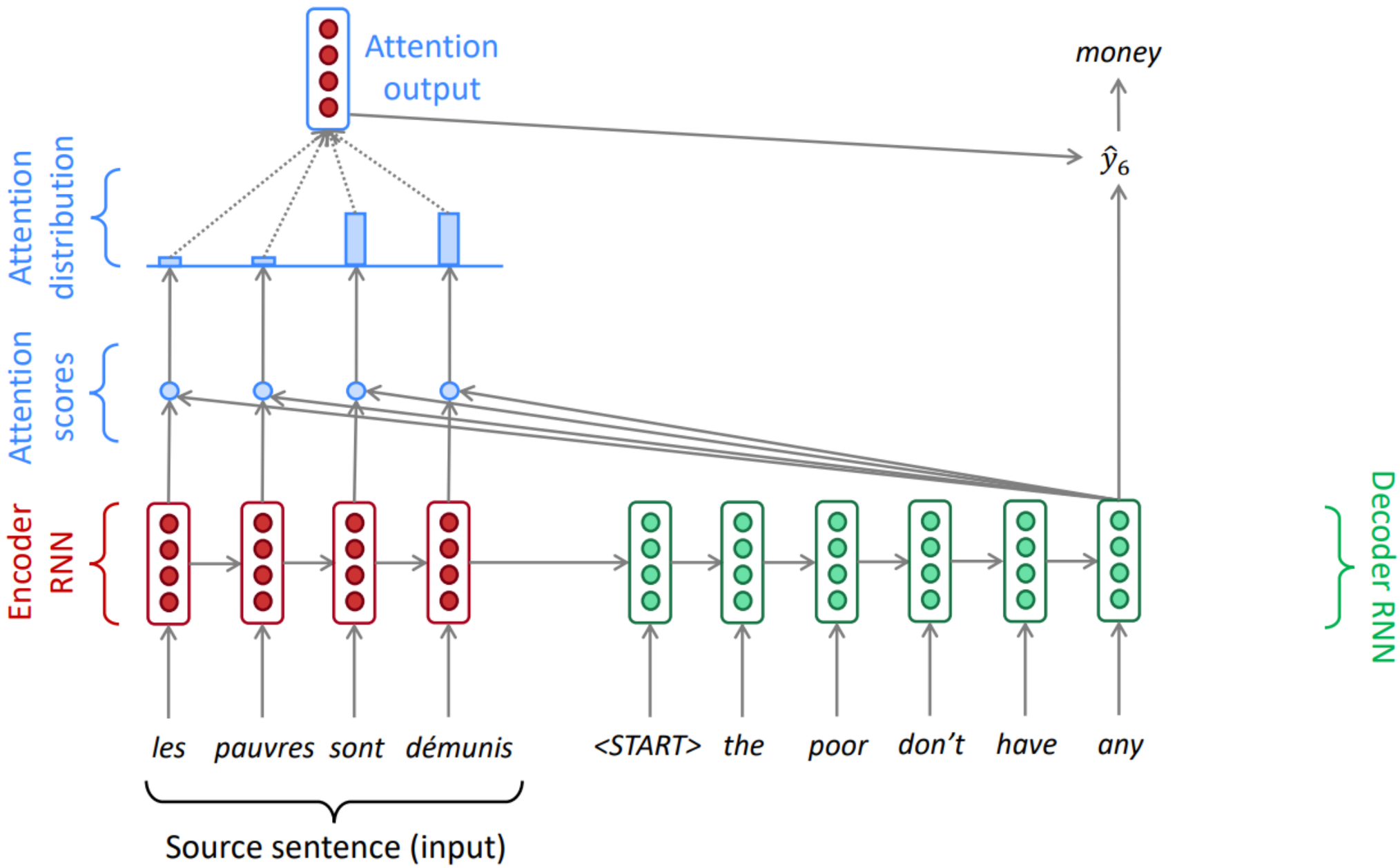
Attention Mechanism



Attention Mechanism



Attention Mechanism



Attention Mechanism (Formally)

- We have encoder hidden states $h_1, \ldots, h_N \in \mathbb{R}^h$
- On timestep *t*, we have decoder hidden state $s_t \in \mathbb{R}^h$
- We get the attention scores $\boldsymbol{e}^t$ for this step:

$$\boldsymbol{e}^t = [\boldsymbol{s}_t^T \boldsymbol{h}_1, \ldots, \boldsymbol{s}_t^T \boldsymbol{h}_N] \in \mathbb{R}^N$$

- We take softmax to get the attention distribution α^t for this step (this is a probability distribution and sums to 1)

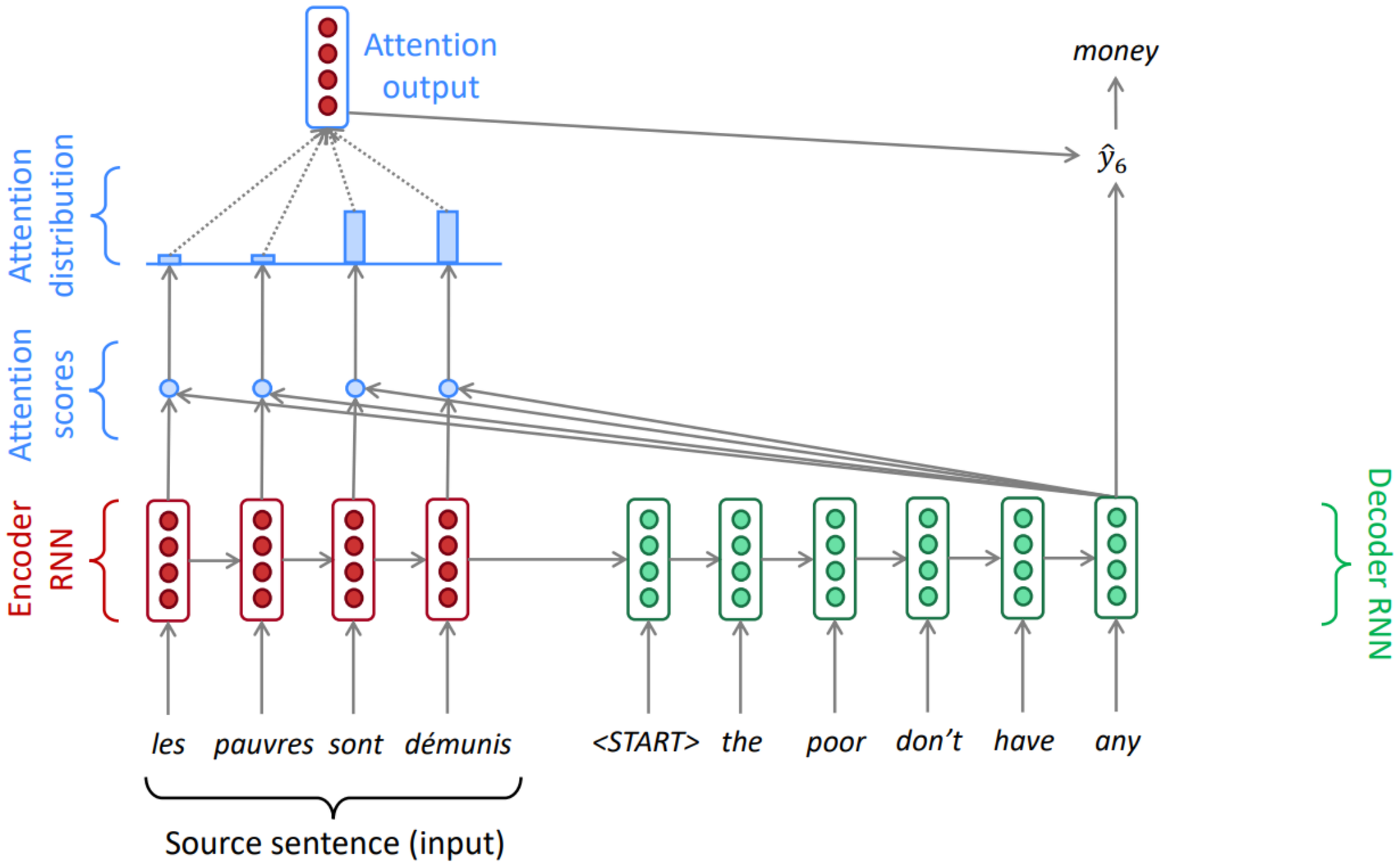
$$\alpha^t = \text{softmax}(\boldsymbol{e}^t) \in \mathbb{R}^N$$

- We use α^t to take a weighted sum of the encoder hidden states to get the attention output $\boldsymbol{a}_t$

$$\boldsymbol{a}_t = \sum_{i=1}^{N} \alpha_i^t \boldsymbol{h}_i \in \mathbb{R}^h$$

Attention Mechanism

Questions?

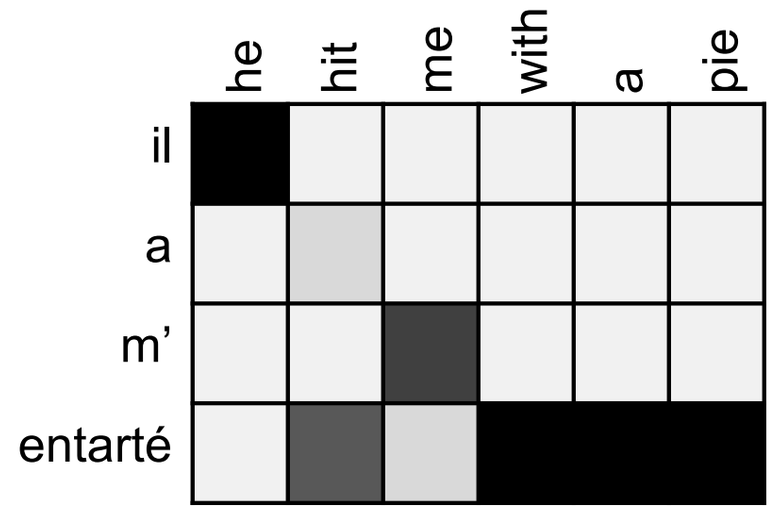


Attention turns out to be very important!

- Attention improves NMT performance significantly
  - It's necessary to allow decoder to focus on the relevant parts of the source
  - Also this is how humans translate
- Attention solves the bottleneck problem
  - allows decoder to look directly at source; bypass bottleneck
- Attention helps with vanishing gradient problem
  - Provides shortcut to faraway states

Attention turns out to be very important!

- Attention provides interpretability
  - By inspecting attention distribution, we can see what the decoder was focusing on
  - (soft) alignment for free
  - Although, it has been questioned recently [1,2,3]

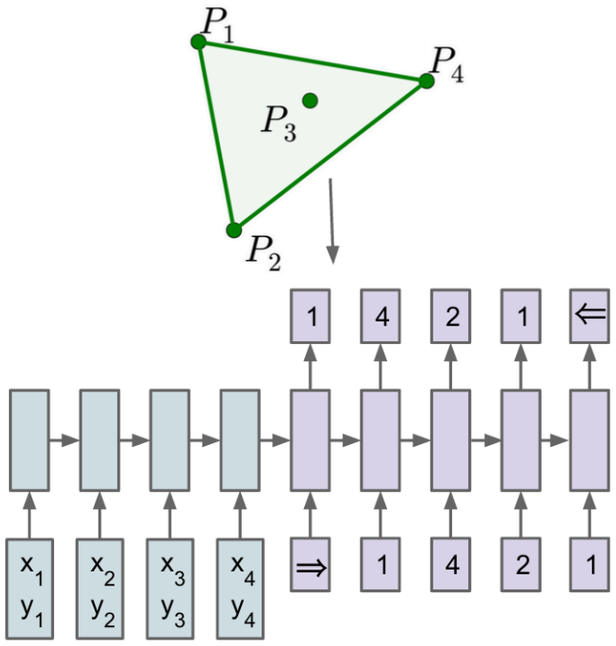


[1] Attention is not Explanation. Sarthak Jain, Byron C. Wallace. In NAACL-2019
[2] Attention is not not Explanation. Sarah Wiegreffe, Yuval Pinter. In EMNLP-2019
[3] Learning to Deceive with Attention-Based Explanations. Danish Pruthi, Mansi Gupta, Bhuwan Dhingra, Graham Neubig, Zachary C. Lipton. In ACL-2020

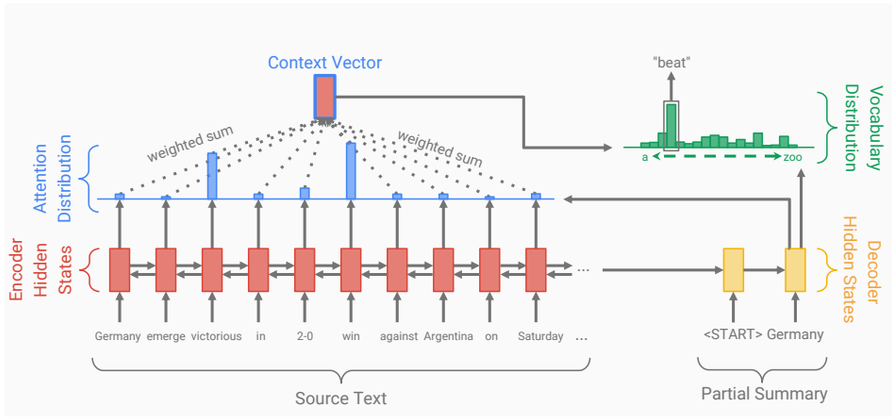
Attention turns out to be very important!

- Attention is a *general* Deep Learning technique
  - Not just in a seq2seq model
  - Not just in NMT; now almost everywhere in DL
  - Can be repurposed to point, to copy, or as a representation layer.

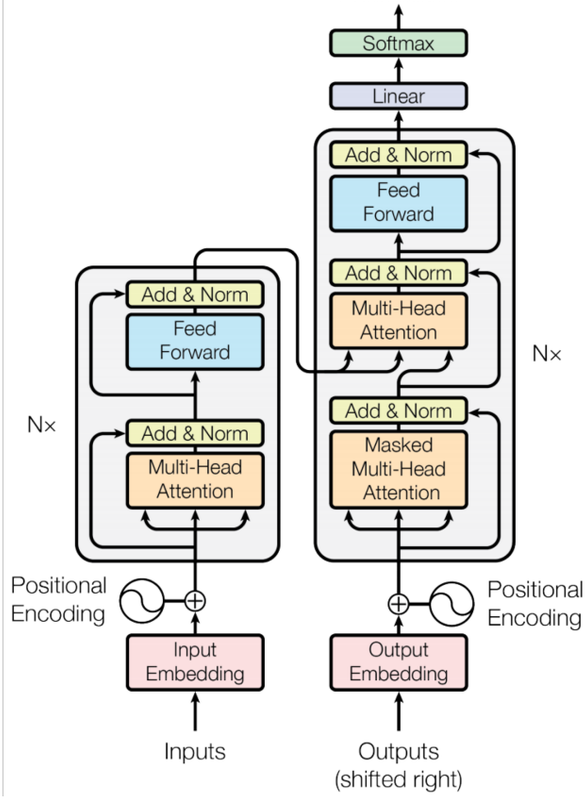
=> Lectures 10 - 11



Pointer Net



Pointer-Generator Net



Transformers

Attention is a *general* Deep Learning technique

- We have encoder hidden states $h_1, \ldots, h_N \in \mathbb{R}^h$ **Keys/Values**
- On timestep *t*, we have decoder hidden state $s_t \in \mathbb{R}^h$ **Query**

- Given a set of key-value vectors, and a *query vector*, **attention** is a technique to compute a weighted sum of the value vectors, dependent on the query-key relevance.

Intuition:

- The weighted sum is a *selective summary* of the information contained in the values, where the query-key determines which values to focus on.

- Attention is a way to obtain a *fixed-size representation of an arbitrary set of representations* (values), dependent on some other representation (query-key).

Attention (Formally)

- We have encoder hidden states $h_1, \ldots, h_N \in \mathbb{R}^h$ **Keys/Values**
- On timestep *t*, we have decoder hidden state $s_t \in \mathbb{R}^h$ **Query**
- We get the attention scores $\boldsymbol{e}^t$ for this step:

$$\boldsymbol{e}^t = [\boldsymbol{s}_t^T \boldsymbol{h}_1, \ldots, \boldsymbol{s}_t^T \boldsymbol{h}_N] \in \mathbb{R}^N$$

There are several ways to do this

- We take softmax to get the attention distribution α^t for this step (this is a probability distribution and sums to 1)

$$\alpha^t = \text{softmax}(\boldsymbol{e}^t) \in \mathbb{R}^N$$

- We use α^t to take a weighted sum of the encoder hidden states to get the attention output $\boldsymbol{a}_t$

$$\boldsymbol{a}_t = \sum_{i=1}^{N} \alpha_i^t \boldsymbol{h}_i \in \mathbb{R}^h$$

Attention Variants

- We have encoder hidden states $h_1, \ldots, h_N \in \mathbb{R}^h$
- On timestep *t*, we have decoder hidden state $s_t \in \mathbb{R}^h$

- Basic dot-product attention: $\boldsymbol{e}_i = \boldsymbol{s}^T \boldsymbol{h}_i \in \mathbb{R}$
  - Note: this assumes $d_1 = d_2$
  - This is the version we saw earlier

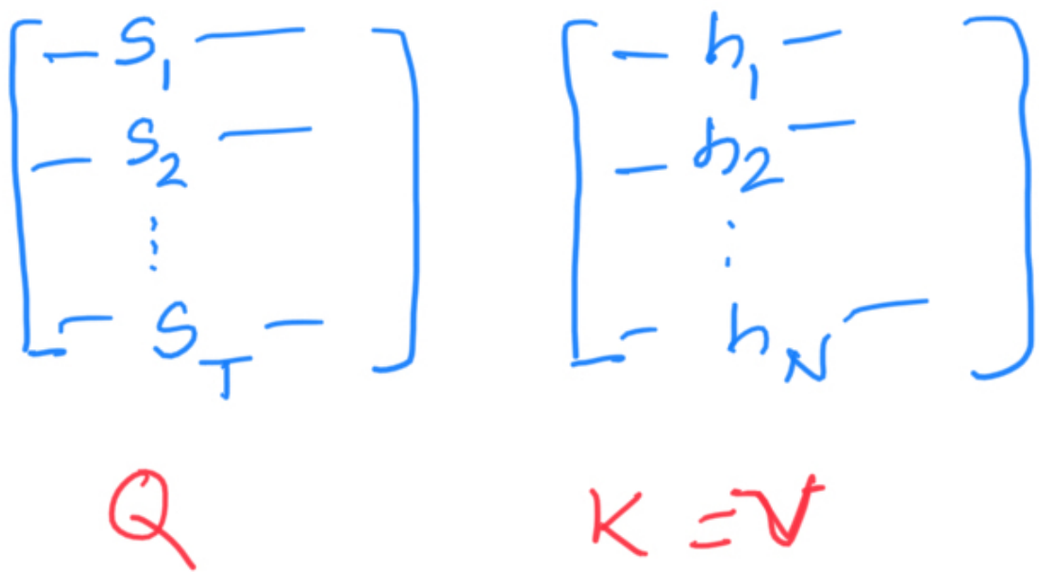
- Multiplicative attention: $\boldsymbol{e}_i = \boldsymbol{s}^T \boldsymbol{W} \boldsymbol{h}_i \in \mathbb{R}$
  - Where $\boldsymbol{W} \in \mathbb{R}^{d_2 \times d_1}$ is a weight matrix

- Additive attention: $\boldsymbol{e}_i = \boldsymbol{v}^T \tanh(\boldsymbol{W}_1 \boldsymbol{h}_i + \boldsymbol{W}_2 \boldsymbol{s}) \in \mathbb{R}$
  - Where $\boldsymbol{W}_1 \in \mathbb{R}^{d_3 \times d_1}, \boldsymbol{W}_2 \in \mathbb{R}^{d_3 \times d_2}$ are weight matrices and $\boldsymbol{v} \in \mathbb{R}^{d_3}$ is a weight vector.
  - d_3 (the attention dimensionality) is a hyperparameter

Attention in Matrix Notation

- We have encoder hidden states $h_1, \ldots, h_N \in \mathbb{R}^h$
- On timestep *t*, we have decoder hidden state $s_t \in \mathbb{R}^h$

- Dot-product attention in matrix notation

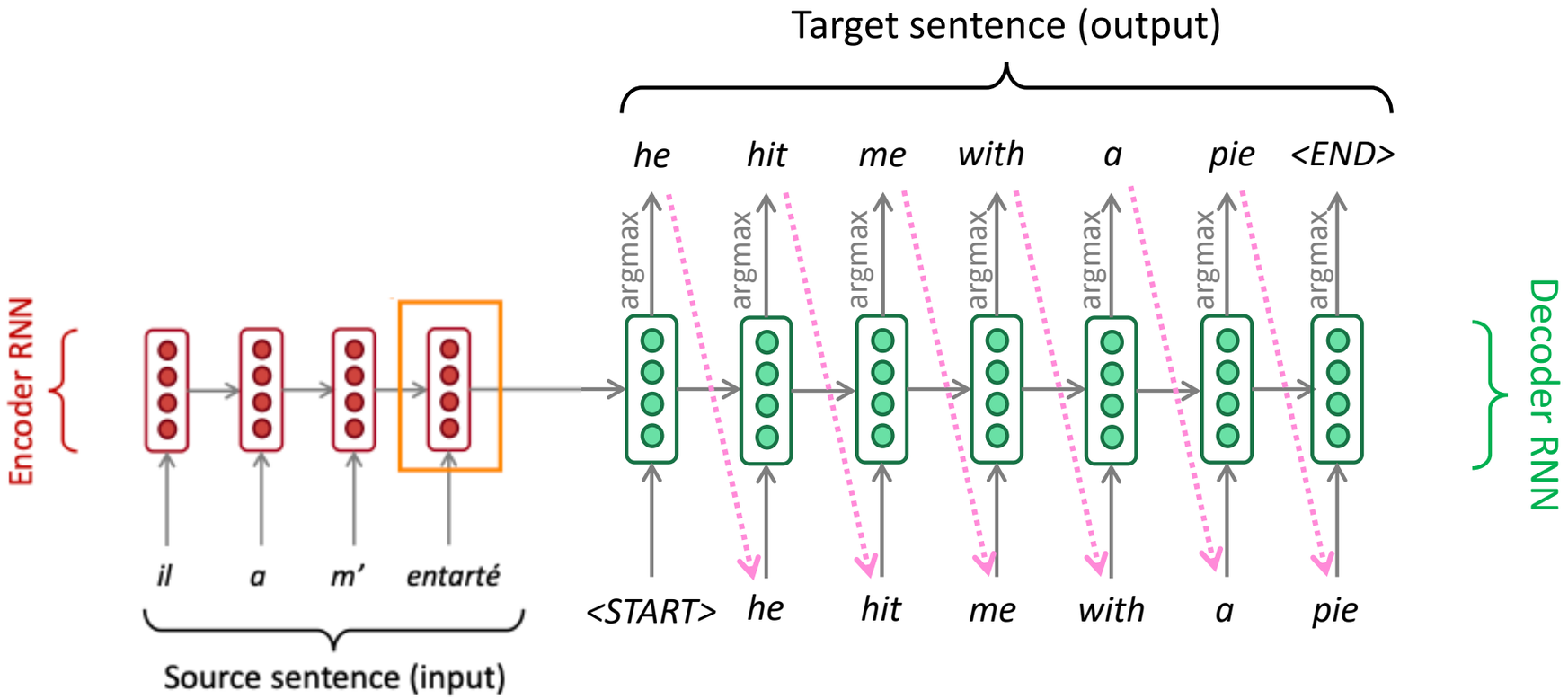


$$A = S(QK^T)V$$

Subword Level Models

Solution to OOV

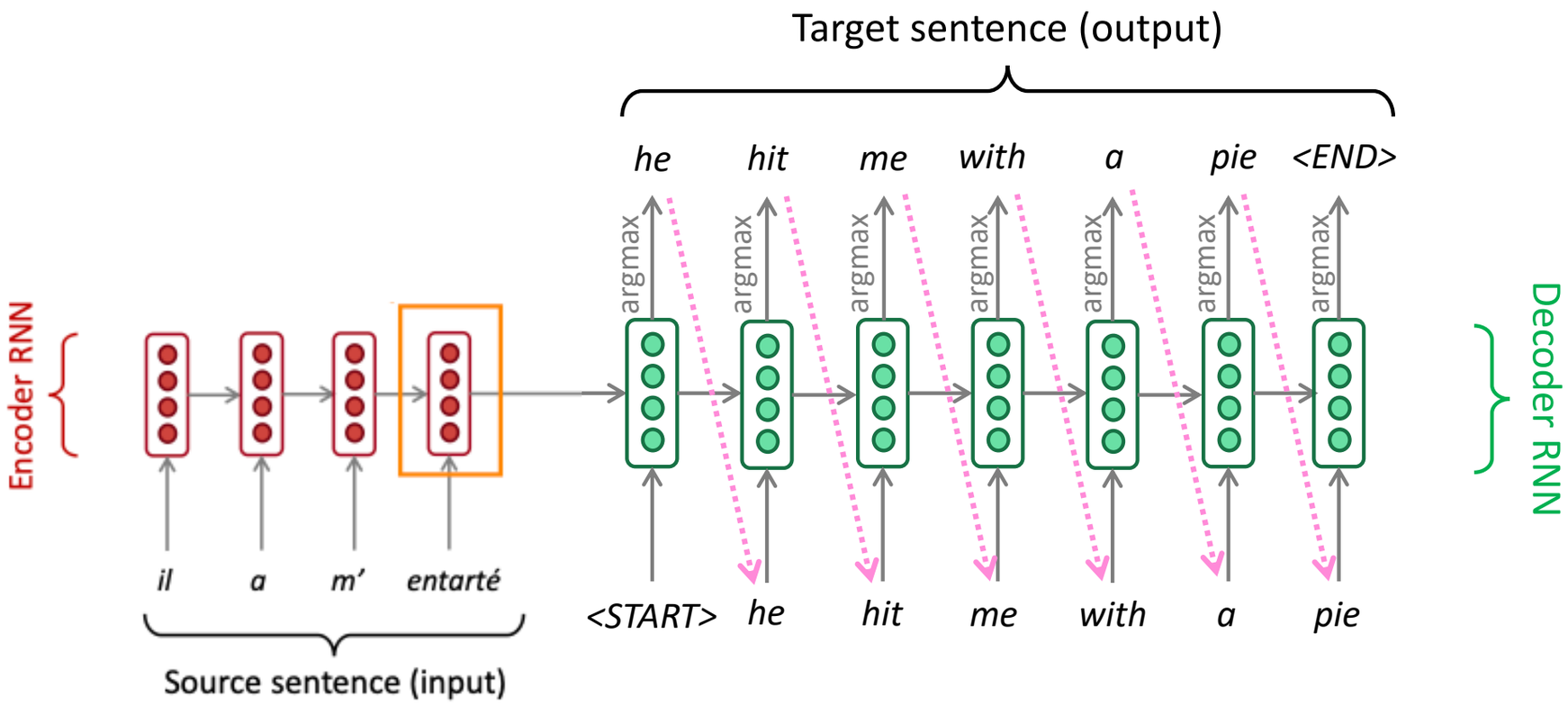
So Far



Words as input/output

- lookup of word embedding for input
- probability distribution over vocabulary for output

Large Vocabularies



- increase network size
- decrease training and decoding speed
- typical vocabulary size: 10,000–100,000 symbols

But

Translation is an open-vocabulary problem

- Many training corpora contain millions of word types
- Word formation processes (e.g., compounding, derivation) allow formation and understanding of unseen words

  - they charge a carry-on bag fee.
  - sie erheben eine Hand|gepäck|gebühr.

- Names, numbers are morphologically simple, but open word classes

  - Obama(English; German)
  - Обама (Russian)
  - オバマ (o-ba-ma) (Japanese)

Credit: Rico Sennrich

Open-vocabulary Models

Dealing with the open-vocabulary problem

- Non-Solution: Ignore Rare Words
- Solution 1: Back-off Models

Modeling Morphology:

- Solution 2: Character-level Models
- Solution 3: Hybrid Models
- Solution 4: Subword-level Models

Non-Solution

- Replace rare words with UNK
- A vocabulary of 50,000 words covers 95% of text

Gets you 95% of the way… if you only care about automatic metrics ….

Credit: Rico Sennrich

Non-Solution

- Replace rare words with UNK
- A vocabulary of 50,000 words covers 95% of text

  Gets you 95% of the way… if you only care about automatic metrics ….

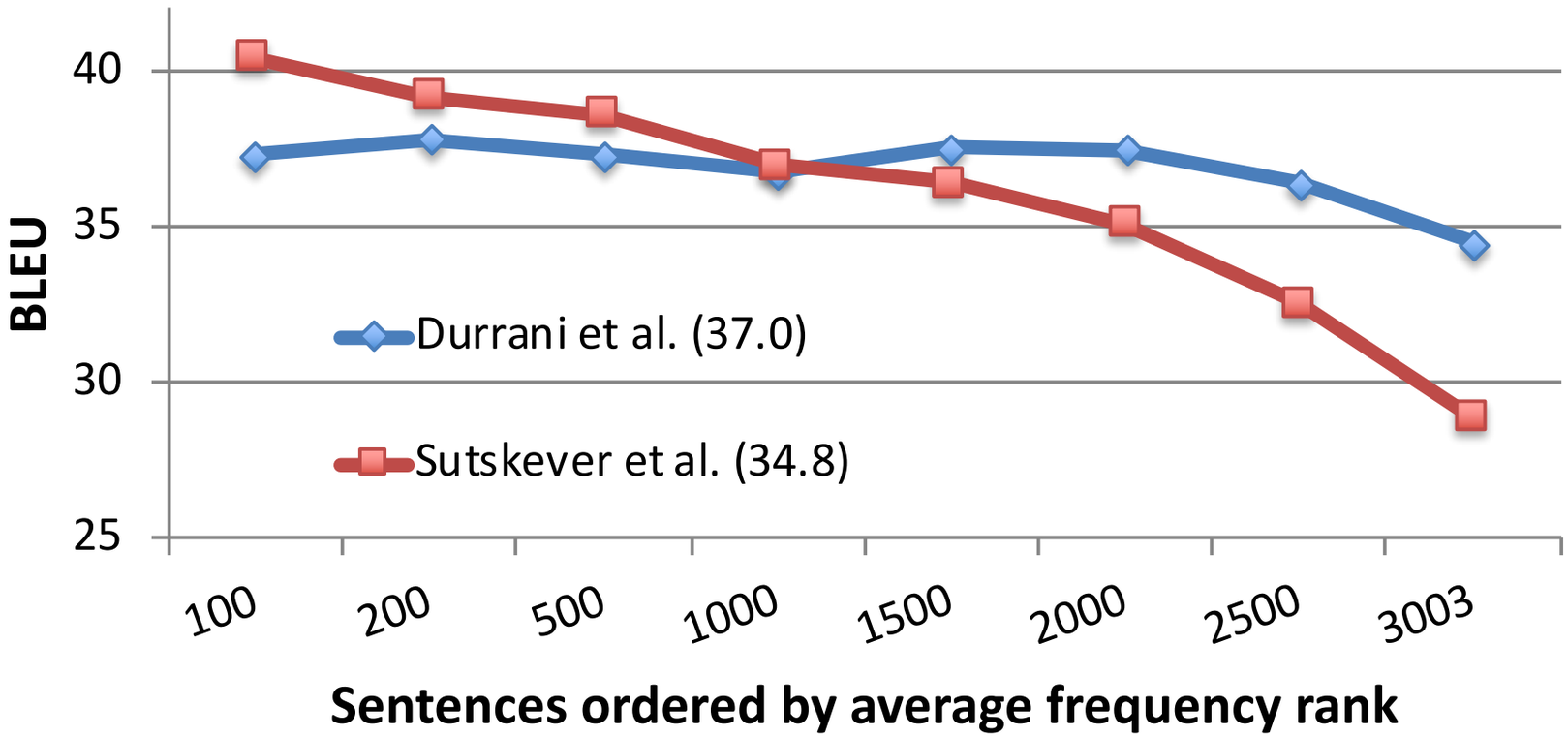
- Why 95% is not enough

source	The **indoor temperature** is very pleasant.
reference	Das **Raumklima** ist sehr angenehm.
[Bahdanau et al., 2015]	Die **UNK** ist sehr angenehm. ✗

Credit: Rico Sennrich

Non-Solution

- Why 95% is not enough
- NMTs translate poorly for sentences with more rare words.



Credit: Luong 2015

Solution 1: Back-off Models

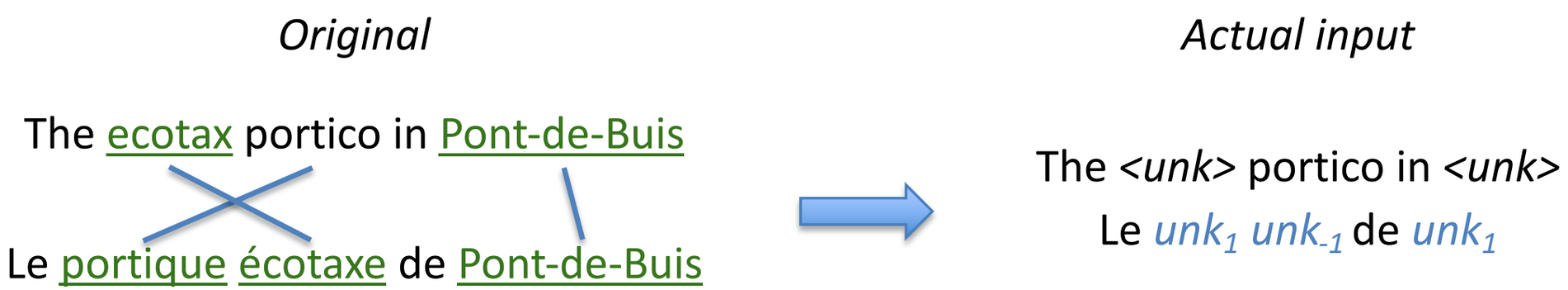
- Replace rare words with UNK at training time
- When system produces UNK, align it to source word, and translate this with back-off method (word/identity translate)

Luong, T., Sutskever, I., Le, Q., Vinyals, O., and Zaremba, W. (2015). Addressing the Rare Word Problem in Neural Machine Translation. In ACL-2015

Credit: Luong 2015

Solution 1: Back-off Models

- Replace rare words with UNK at training time
- When system produces UNK, align it to source word, and translate this with back-off method (word/identity translate)

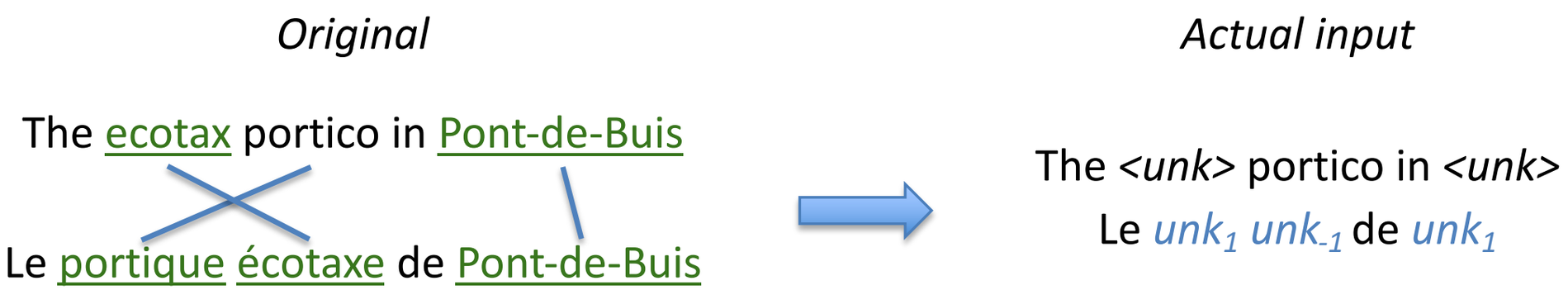


Annotate **train** data: unsupervised alignments & relative indices.
Post-process **test** translations: word/identity translations.

Luong, T., Sutskever, I., Le, Q., Vinyals, O., and Zaremba, W. (2015). Addressing the Rare Word Problem in Neural Machine Translation. In ACL-2015

Credit: Luong 2015

Solution 1: Back-off Models



- Limitations
  - Compounds: hard to model 1-to-many relationships
  - Morphology: hard to predict inflection with back-off dictionary
  - Names: if alphabets differ, we need transliteration
  - Alignment: attention model unreliable (both train and test time)

Credit: Rico Sennrich

Open-vocabulary Models

- Non-Solution: Ignore Rare Words
- Solution 1: Back-off Models

Modeling Morphology:

- Solution 2: Character-level Models
- Solution 3: Hybrid Models
- Solution 4: Subword-level Models

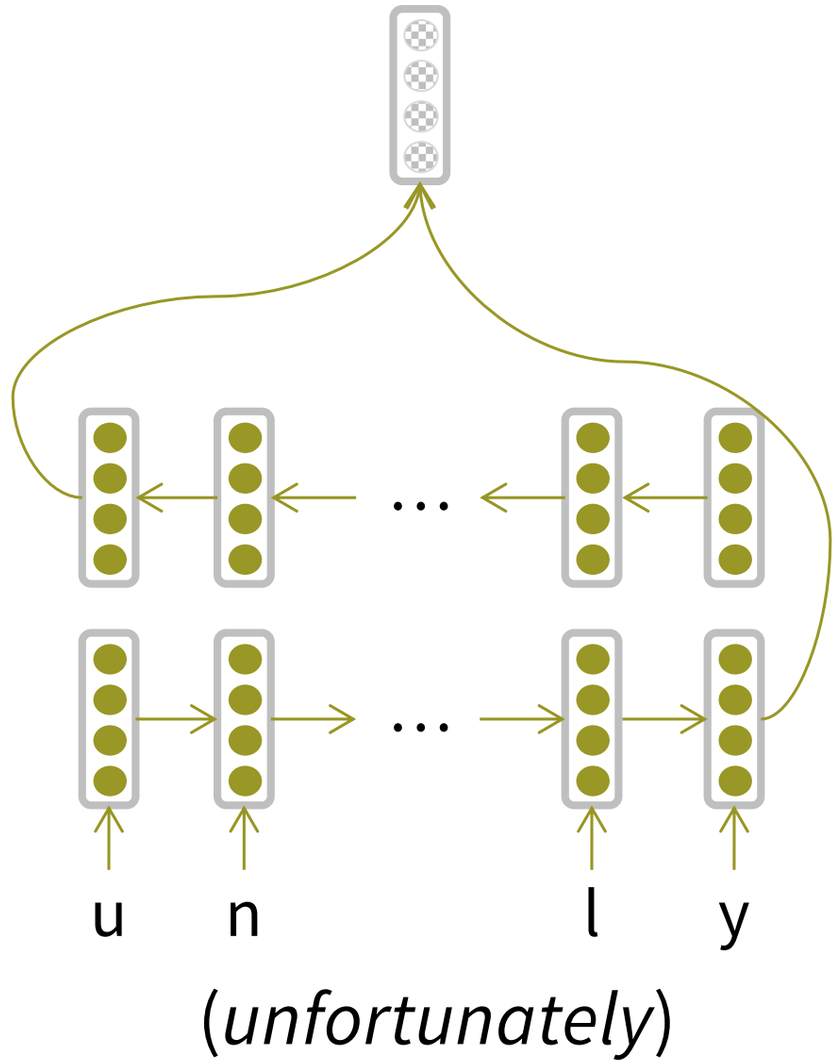
Credit: Rico Sennrich

Solution 2: Character-level Models

- Easier alternative is to just work with character *n*-grams

- Character-level models: RNNs, ConvNets.

- Good results!



Challenges

- Word boundaries:

  Boundaries between words not marked in some languages

  - Chinese, Japanese, Thai

- Clitics:

  فقلناها = ها +نا +قال +ف = so+said+we+it

  Arabic

  I'm, We've English

- Compound nouns:

  German

  - Separated life insurance company employee
  - Joined Lebensversicherungsgesellschaftsangestellter

Challenges

- Rich morphology:

"uygarlaştıramadıklarımızdanmışsınızcasına"

"uygar+laş+tır+ama+dık+lar+ımız+dan+mış+sınız+casına"

Behaving as if you are among those whom we could not cause to become civilized

Turkish

- Transliteration:

Christopher ↦ Kryštof

- Social media:

Character-Level Models (Two approaches)

- Word embeddings can be composed from character embeddings
  - Generates embeddings for unknown words
  - Similar spellings share similar embeddings
  - Solves OOV problem
- Build entirely character-level models
  - No notion of separate words

Both methods have proven to work successfully!

Entirely Character-Level Models

- Advantages
  - open-vocabulary
  - no heuristic or language-specific segmentation
  - neural network can conceivably learn from raw character sequences

Entirely Character-Level Models

- English-Czech WMT 2015 Results

System	BLEU
Word-level model (single; large vocab; UNK replace)	15.7
Character-level model (single; 600-step backprop)	15.9

- Character-level seq2seq (LSTM) NMT system
- Worked well against word-level baseline

(Thang Luong, Christopher Manning, ACL 2016)

Entirely Character-Level Models

- English-Czech Example

source	Her ***11-year-old*** daughter , ***Shani Bart*** , said it felt a little bit ***weird***
human	Její **jedenáctiletá** dcera **Shani Bartová** prozradila , že je to trochu **zvláštní**
char	Její **jedenáctiletá** dcera , **Shani Bartová** , říkala , že cítí trochu ***divně***
word	Její <unk> dcera <unk> <unk> řekla , že je to trochu divné
	Její **11-year-old** dcera **Shani ,** řekla , že je to trochu ***divné***

- Disadvantage
  - Increasing sequence length slows training/decoding (reported x2–x4 increase in training time)
  - Can hurt modelling of long-range dependencies

(Thang Luong, Christopher Manning, ACL 2016)

Open-vocabulary Models

- Non-Solution: Ignore Rare Words
- Solution 1: Back-off Models

Modeling Morphology:

- Solution 2: Character-level Models
- Solution 3: Hybrid Models
- Solution 4: Subword-level Models

Credit: Rico Sennrich

Word-Character Hybrid Models

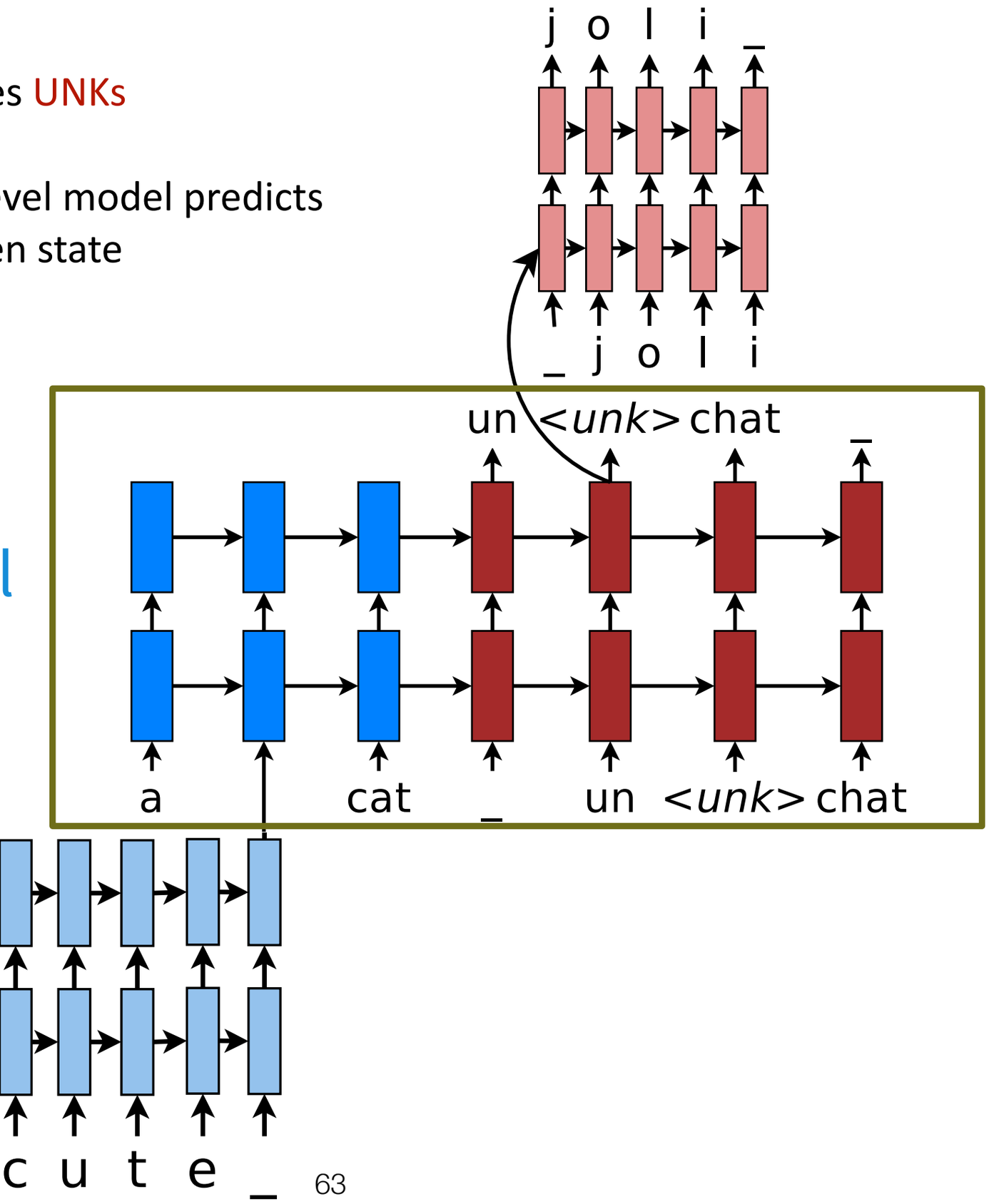
A *best-of-both-worlds* architecture:

- Translate mostly at the word level
- Only go to the character level when needed

Thang Luong and Chris Manning. Achieving Open Vocabulary Neural Machine Translation with Hybrid Word-Character Models. ACL 2016.

Word-Character Hybrid Models

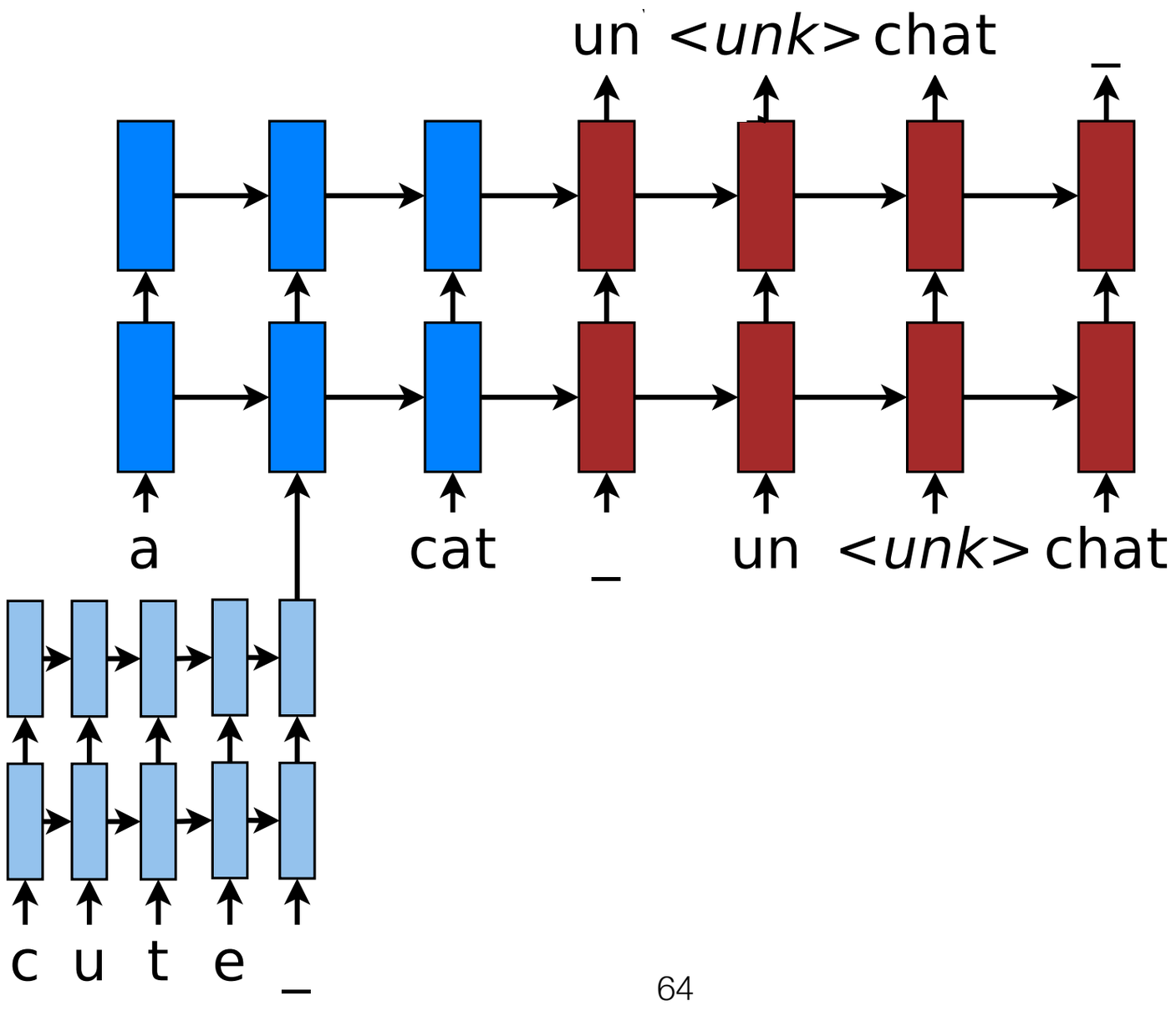
- Word-level model produces UNKs
- For each UNK, character-level model predicts word based on word hidden state



Word-Character Hybrid Models

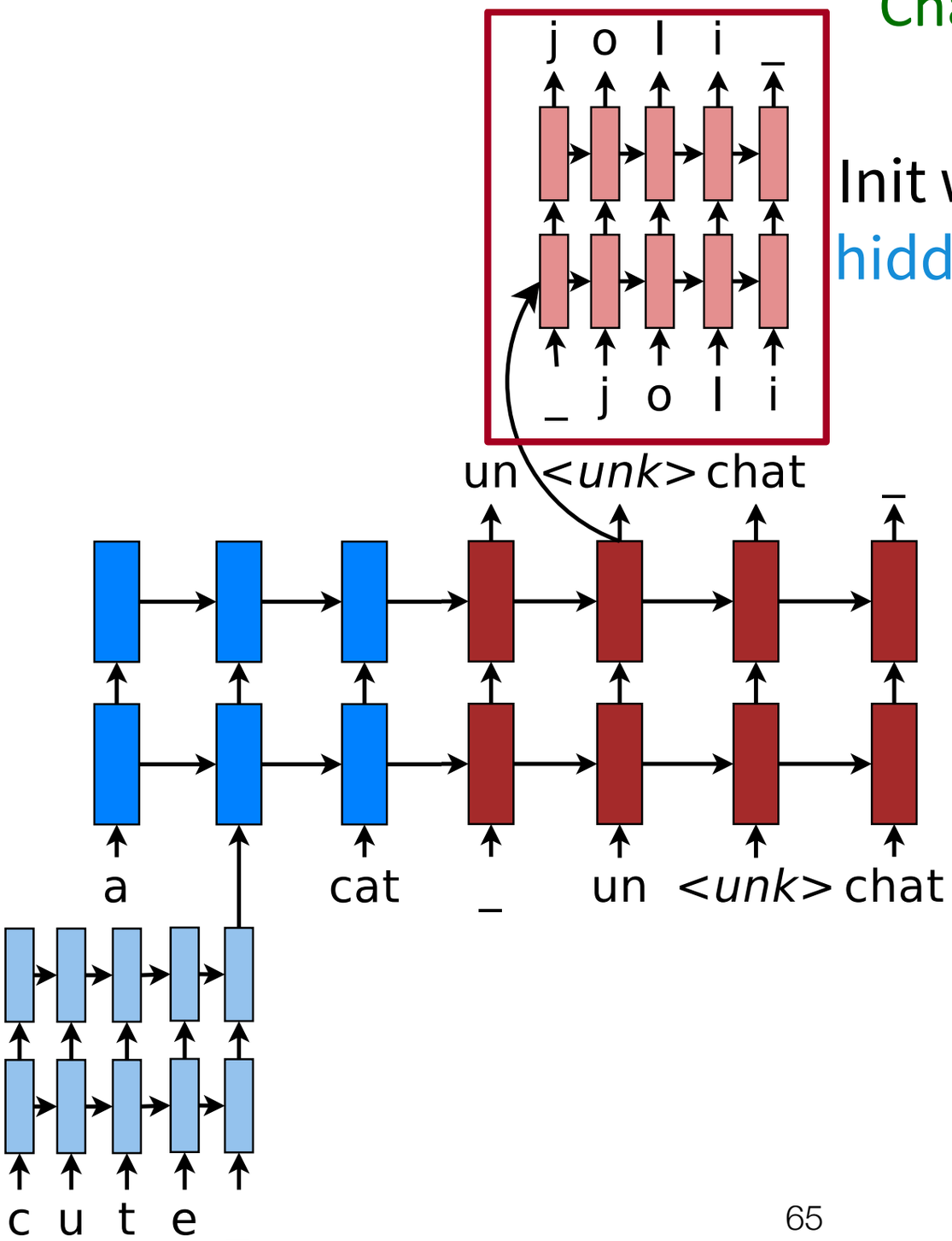
2-stage Decoding

Word-level beam search



Word-Character Hybrid Models

2-stage Decoding



Char-level beam search for <unk>

Init with word hidden states.

Word-Character Hybrid Models

- English-Czech WMT'15
  - Train on WMT'15 data (12M sentence pairs)
    - newstest2015

Systems	BLEU
Winning WMT'15 (Bojar & Tamchyna, 2015)	18.8
Word-level NMT (Jean et al., 2015)	18.3
Hybrid NMT (Luong & Manning, 2016)*	**20.7**

30x data
3 systems

Large vocab
+ copy mechanism

Then SOTA!

But cf. Cherry et al. 2018: ~26 BLEU

Thang Luong and Chris Manning. Achieving Open Vocabulary Neural Machine Translation with Hybrid Word-Character Models. ACL 2016.

Word-Character Hybrid Models

- English-Czech WMT'15

source	The author ***Stephen Jay Gould*** died 20 years after ***diagnosis*** .
human	Autor **Stephen Jay Gould** zemřel 20 let po **diagnóze** .
char	Autor **Stepher Stepher** zemřel 20 let po **diagnóze** .
word	Autor Stephen Jay <unk> zemřel 20 let po <unk> .
	Autor **Stephen Jay Gould** zemřel 20 let po **po** .
hybrid	Autor Stephen Jay <unk> zemřel 20 let po <unk> .
	Autor **Stephen Jay Gould** zemřel 20 let po **diagnóze** .

Perfect translation!

Thang Luong and Chris Manning. Achieving Open Vocabulary Neural Machine Translation with Hybrid Word-Character Models. ACL 2016.

Open-vocabulary Models

- Non-Solution: Ignore Rare Words
- Solution 1: Back-off Models

Modeling Morphology:

- Solution 2: Character-level Models
- Solution 3: Hybrid Models
- Solution 4: Subword-level Models

Credit: Rico Sennrich

Subword Models

- Same architecture as for word-level model

  But use smaller units: word pieces or subwords

Subword Units

Segmentation algorithms wishlist:

- **Open-vocabulary NMT**: encode all words through small vocabulary
- Encoding generalizes to unseen words
- Small sequence size for faster processing
- Good translation quality

Byte Pair Encoding

Originally a compression algorithm [Gage, 1994]:

- Most frequent byte pair ↦ a new byte
- Replace bytes with character ngrams

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural Machine Translation of Rare Words with Subword Units. ACL 2016

Byte Pair Encoding

An unsupervised word segmentation algorithm:

- Bottom-up character merging
- Start with a unigram vocabulary of all (Unicode) characters in data
- Most frequent ngram pairs ↦ a new ngram
- Hyperparameter: when to stop
  - Controls vocabulary size

Byte Pair Encoding

- Start with a unigram vocabulary of all (Unicode) characters in data

Vocabulary

l, o, w, e, r, n, w, s, t, i, d

Dictionary

5	l o w
2	l o w e r
6	n e w e s t
3	w i d e s t

Dictionary of word frequency

Byte Pair Encoding

- Most frequent ngram pairs ↦ a new ngram

Vocabulary

l, o, w, e, r, n, w, s, t, i, d, **es**

Dictionary

5 l o w
2 l o w e r
6 n e w **es** t
3 w i d **es** t

- Add **es** with freq 9
- Add **est** with freq 9

Dictionary of word frequency

Byte Pair Encoding

- Most frequent ngram pairs ↦ a new ngram

Vocabulary

l, o, w, e, r, n, w, s, t, i, d, es, est, **lo**

Dictionary

5 **lo** w
2 **lo** w e r
6 n e w est
3 w i d est

Dictionary of word frequency

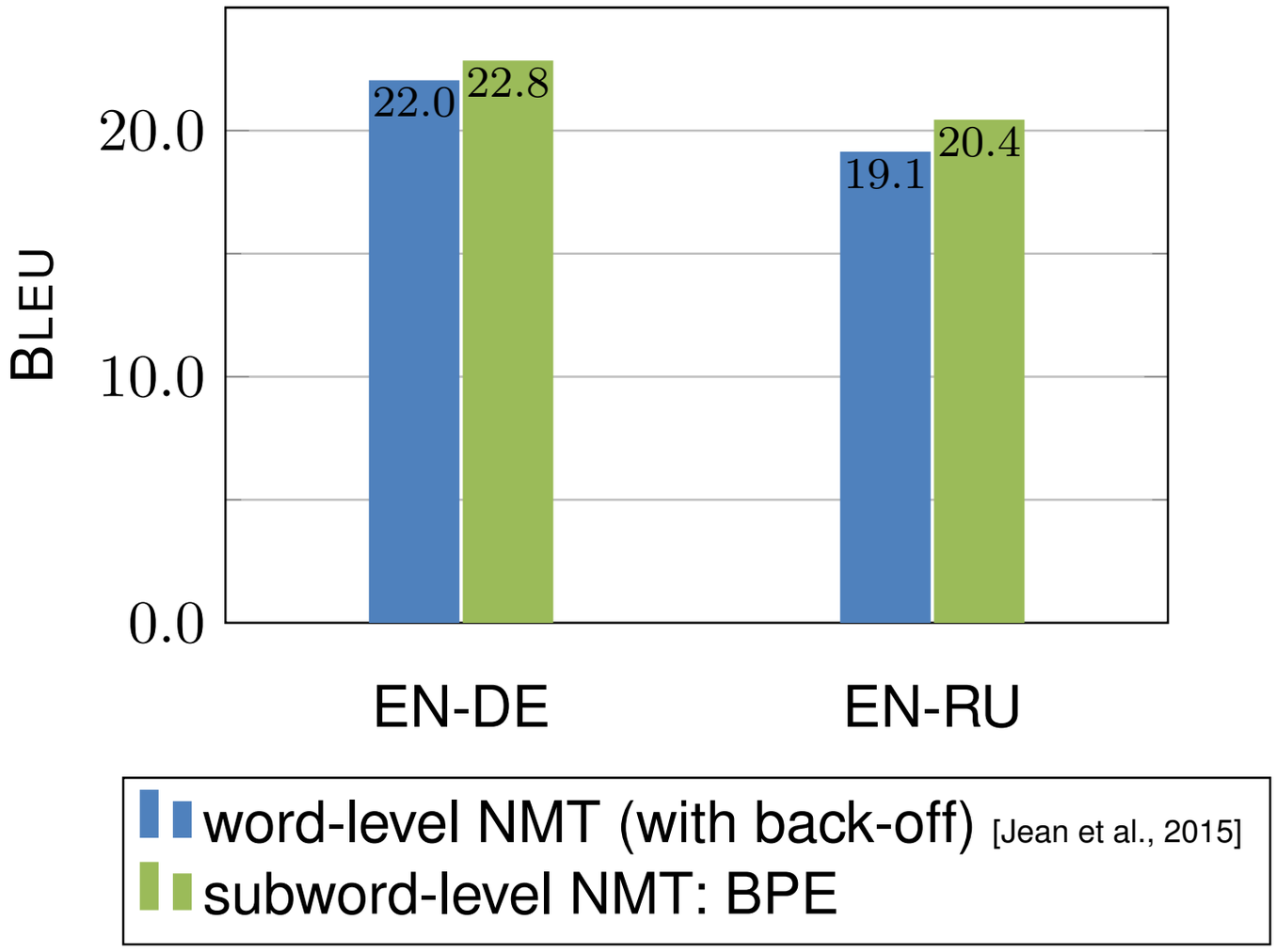
- Add **lo** with freq 7

- Add **low** with freq 7

Byte Pair Encoding

- Have a target vocabulary size and stop when you reach it
- Do deterministic longest piece segmentation of words
- Segmentation is only within words identified by some prior tokenizer (commonly Moses tokenizer for MT)
- Automatically decides vocab for system
  - No longer strongly "word" based in conventional way

Byte Pair Encoding



Top places in WMT 2016!

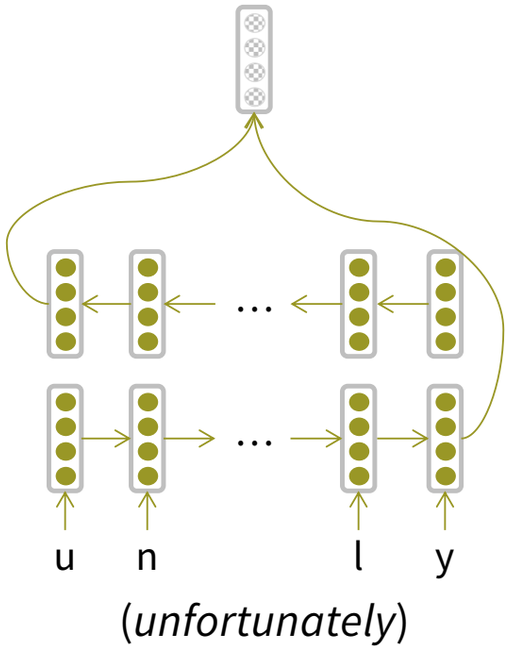
Credit: Rico Sennrich

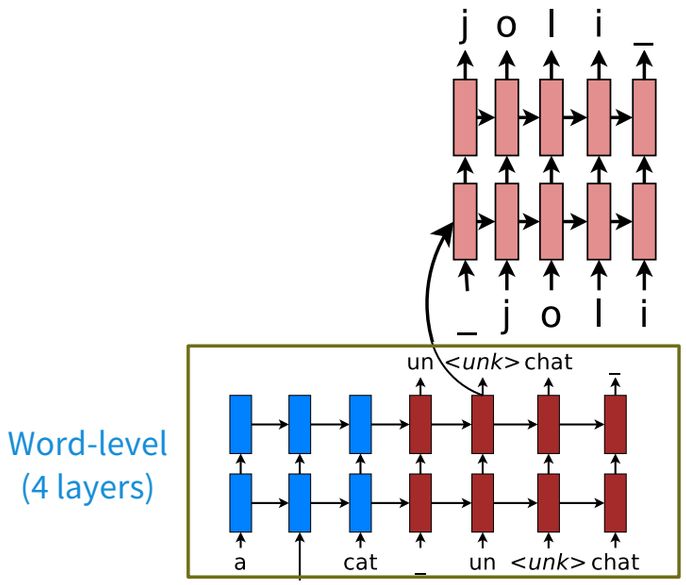
Byte Pair Encoding

system	sentence
source	health research institutes
reference	Gesundheitsforschungsinstitute
word-level (with back-off)	Forschungsinstitute
character bigrams	Fo|rs|ch|un|gs|in|st|it|ut|io|ne|n
BPE	Gesundheits|forsch|ungsin|stitute
source	rakfisk
reference	ракфиска (rakfiska)
word-level (with back-off)	rakfisk → UNK → rakfisk
character bigrams	ra|kf|is|k → ра|кф|ис|к (ra|kf|is|k)
BPE	rak|f|isk → рак|ф|иска (rak|f|iska)

Credit: Rico Sennrich

Subword Models (Summary)

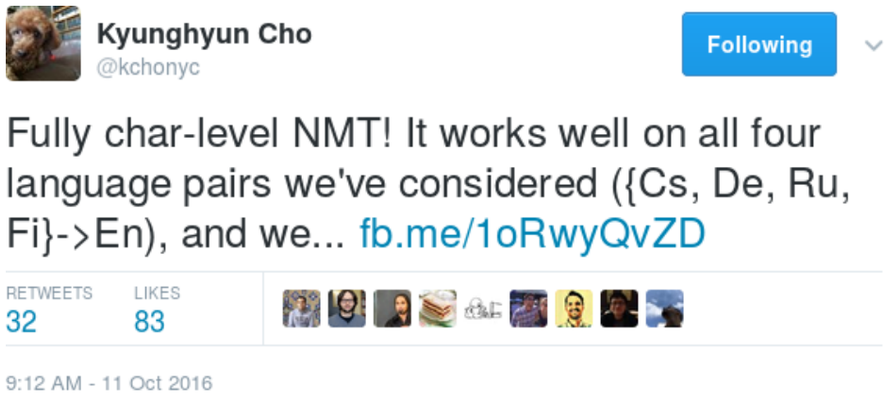


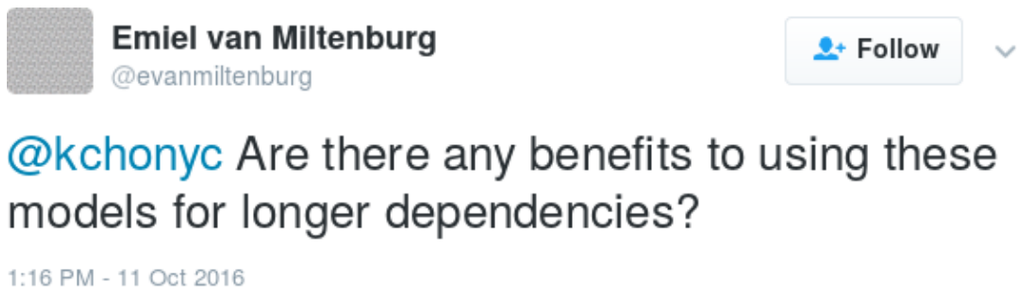


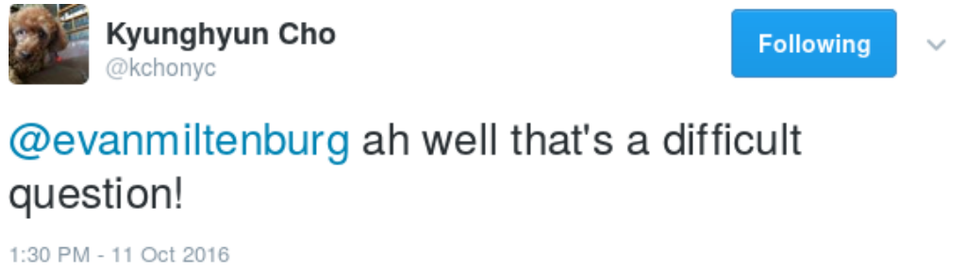
Dictionary

5 **lo** w
2 **lo** w e r
6 n e w est
3 w i d est

Subword Models (Remarks)



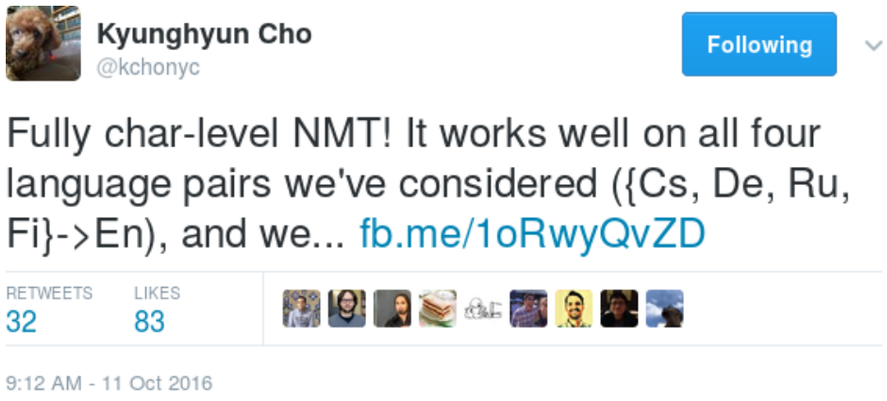


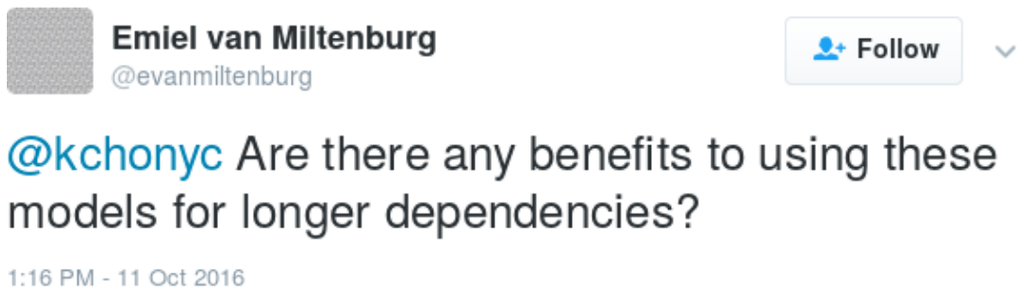


word-level	but as the **example** of Mobilking in Poland **shows** |———— 5 steps ————|
subword-level (byte-pair encoding)	but as the **example** of Mobil+ king in Poland **shows** |———— 6 steps ————-|
character-level	b u t _ a s _ t h e _ **e x a m p l e** _ o f _ M o b i l k i n g _ i n _ P o l a n d _ **s h o w s** | ———— 29 steps ————-|

Credit: Rico Sennrich

Subword Models (Remarks)

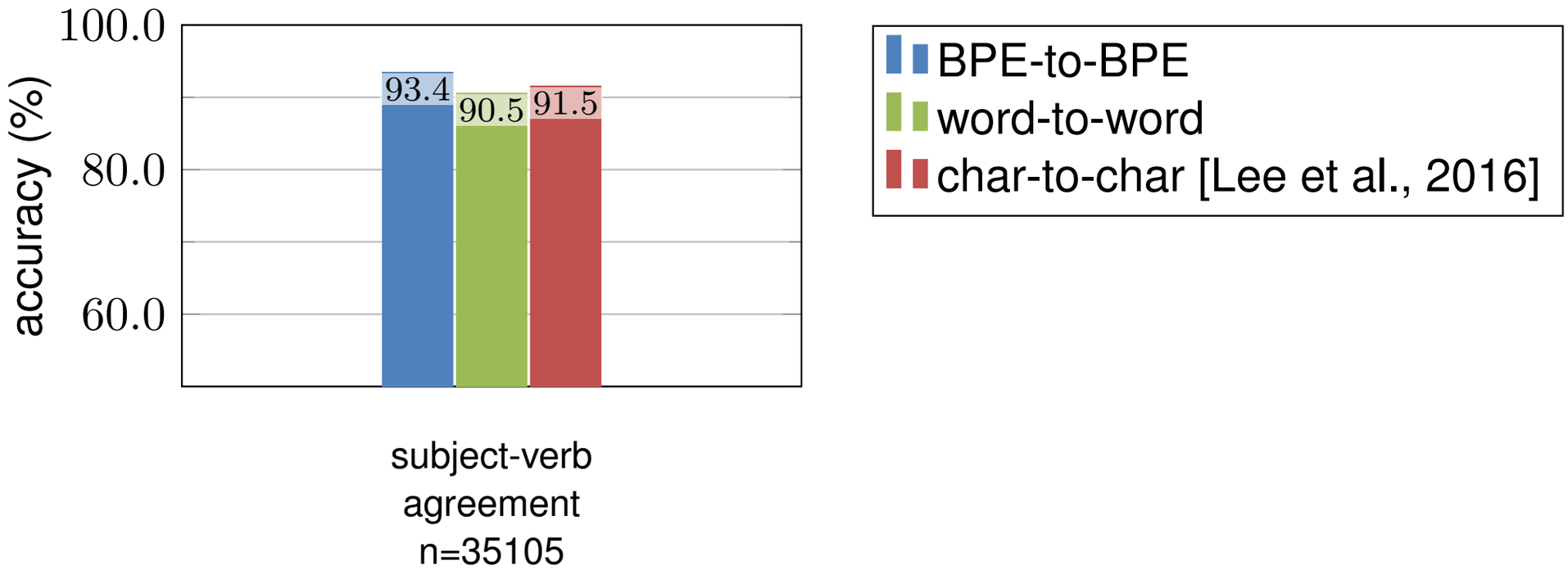




word-level	but as the **example** of UNK in Poland **shows** |———— 5 steps ————|
subword-level (byte-pair encoding)	but as the **example** of Mobil+ king in Poland **shows** |———— 6 steps ————-|
character-level	b u t _ a s _ t h e _ **e x a m p l e** _ o f _ M o b i l k i n g _ i n _ P o l a n d _ **s h o w s** | ———— 29 steps ————-|

Credit: Rico Sennrich

Subword Models (Remarks)



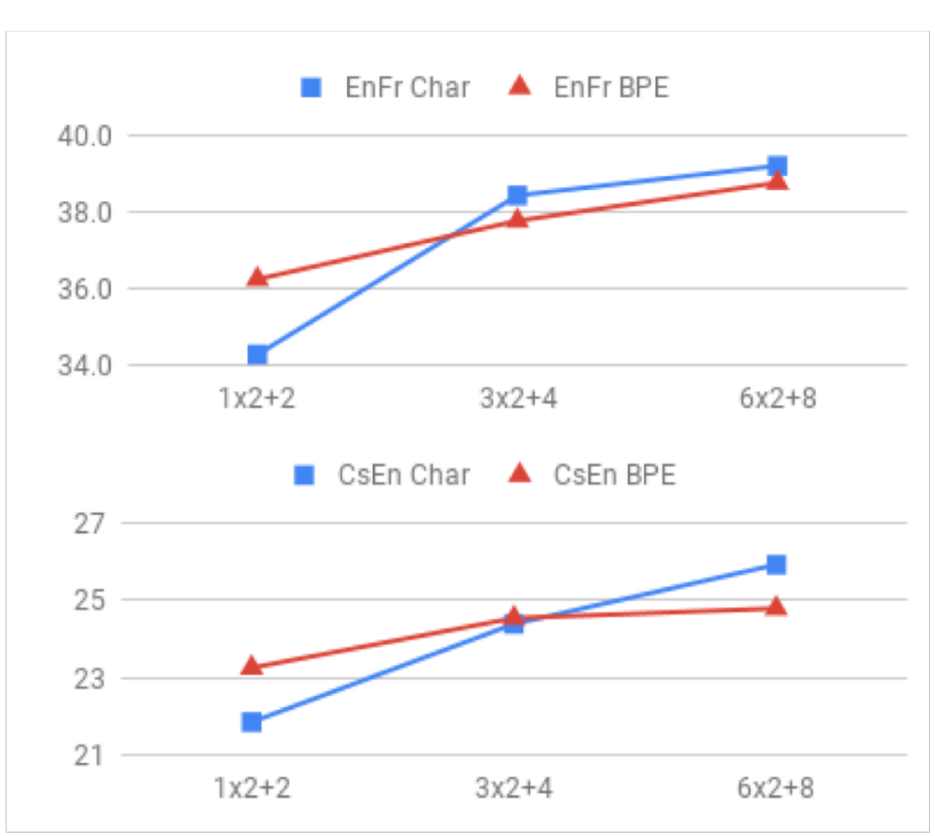
- Word-level model is poor for rare words
- Character-level model is poor for long distances
- BPE subword segmentation is a good compromise

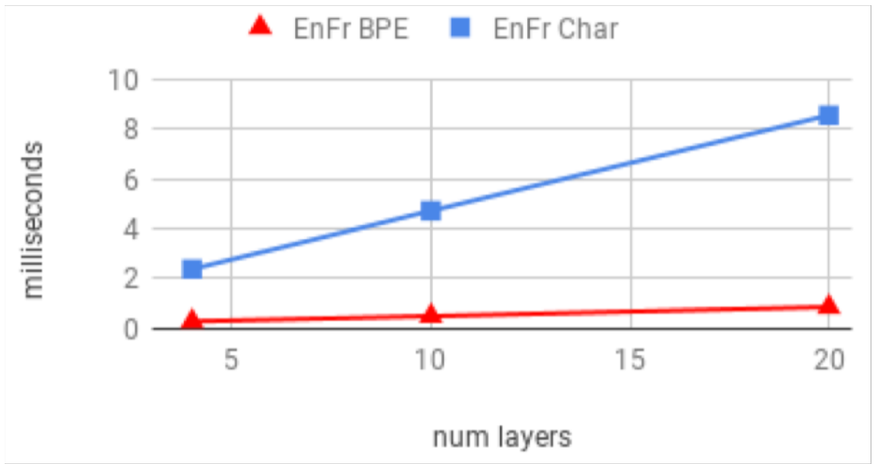
Credit: Rico Sennrich

Entirely Character-Level Models

- Revisiting Character-Based Neural Machine Translation with Capacity and Compression. 2018. Cherry, Foster, Bapna, Firat, Macherey, Google AI

- With sufficient depth (of layers) the modelling problem can be solved!





MT Research Continues!

- Still many difficulties:
  - Data hungry!
    - Low-resource languages (or domain)
  - Maintaining longer context
    - Discourse-level aspects
  - Translation speed
  - MT Robustness