Neural Machine Translation

Thang Luong
Kyunghyun Cho
Christopher Manning

@lmthang · @kchonyc · @chrmanning

ACL 2016 tutorial · https://sites.google.com/site/acl16nmt/
1a. Intro to (Neural) Machine Translation

Ideas connecting Phrase-Based Statistical MT and NMT

Neural Language Models
Machine Translation

The classic test of language understanding!

Both language analysis & generation

Big MT needs … for humanity … and commerce

Translation is a US$40 billion a year industry

Huge in Europe, growing in Asia
Large social/government/military as well as commercial needs
The need for machine translation

Huge commercial use

Google translates over 100 billion words a day

Facebook has just rolled out new homegrown MT

“When we turned [MT] off for some people, they went nuts!”

eBay uses MT to enable cross-border trade

https://googleblog.blogspot.com/2016/04/ten-years-of-google-translate.html
https://techcrunch.com/2016/05/23/facebook-translation/
Scenarios for machine translation

1. The dream of fully automatic high-quality MT (FAHQMFT)
   This still seems a distant goal

2. User- or platform-initiated low quality translation
   The current mainstay of MT
   Google Translate
   Bing Translator
3. Author-initiated high quality translation

MT with human post-editing or MT as a translation aid is clearly growing ... but remains painful

Great opportunities for a much brighter future where MT assists humans: e.g., MateCat or LiLT

https://lilt.com/
Talk in Sess 1C!
Progress in MT

- 1954: Georgetown
- 1966: IBM
- 1982: METEO
- 1993: Statistical MT (IBM-Brown/Mercer)
- 2003: Phrase-based (Och)
- 2005: Hiero (Chiang)
- 2016: Neural MT

Lots of remaining problems!
Wow, @stanfordnlp 's neural MT system for the IWSLT en-de task outperforms 2nd place by a massive 4.7 BLEU points: workshop2015.iwslt.org/downloads/IWSL...
IWSLT 2015, TED talk MT, English-German

**BLEU (CASED)**

- Stanford: 30.85
- Karlsruhe: 26.18
- Edinburgh: 26.02
- Heidelberg: 24.96
- PJAIt: 22.51
- Baseline: 20.08

**HUMAN EVALUATION (HTER)**

- Stanford: 16.16%
- Edinburgh: 21.84%
- Karlsruhe: 22.67%
- Heidelberg: 23.42%
- PJAIt: 28.18%
Progress in Machine Translation
[Edinburgh En-De WMT newstest2013 Cased BLEU; NMT 2015 from U. Montréal]

Phrase-based Statistical Machine Translation

A marvelous use of big data but … it’s mined out?!?

1519年600名西班牙人在墨西哥登陆，去征服几百万人口的阿兹特克帝国，初次交锋他们损兵三分之二。

In 1519, six hundred Spaniards landed in Mexico to conquer the Aztec Empire with a population of a few million. They lost two thirds of their soldiers in the first clash.

translate.google.com (2009): 1519 600 Spaniards landed in Mexico, millions of people to conquer the Aztec empire, the first two-thirds of soldiers against their loss.

translate.google.com (2013): 1519 600 Spaniards landed in Mexico to conquer the Aztec empire, hundreds of millions of people, the initial confrontation loss of soldiers two-thirds.

translate.google.com (2014): 1519 600 Spaniards landed in Mexico, millions of people to conquer the Aztec empire, the first two-thirds of the loss of soldiers they clash.

translate.google.com (2015): 1519 600 Spaniards landed in Mexico, millions of people to conquer the Aztec empire, the first two-thirds of the loss of soldiers they clash.

translate.google.com (2016): 1519 600 Spaniards landed in Mexico, millions of people to conquer the Aztec empire, the first two-thirds of the loss of soldiers they clash.
Neural MT is good!
Neural MT went from a fringe research activity in 2014 to the widely-adopted leading way to do MT in 2016.

Amazing!
What is Neural MT (NMT)?

Neural Machine Translation is the approach of modeling the entire MT process via one big artificial neural network.*

*But sometimes we compromise this goal a little
Neural encoder-decoder architectures
NMT system for translating a single word

\[ V_{s \times 1} \quad d \times V_{s} \quad d \times 1 \]

\[ w \quad L \quad x = Lw \]

\[ \begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
\end{bmatrix}
\begin{bmatrix}
- & - & 0.2 & - & - \\
- & - & 0.4 & - & - \\
- & - & 0.3 & - & - \\
- & - & 0.1 & - & - \\
- & - & 0.1 & - & - \\
- & - & 0.5 & - & - \\
\end{bmatrix}
\begin{bmatrix}
0.2 \\
0.4 \\
0.3 \\
0.1 \\
0.1 \\
0.5 \\
\end{bmatrix} \]

one hot word symbol

word embedding matrix

Looks up column of word embedding matrix
NMT system for translating a single word

\[ V_{s \times 1} \cdot d \times V_s \cdot d \times 1 \cdot d \times d \cdot d \times 1 = f(z) \]

\[ w \cdot L \cdot x = Lw \quad A \quad z = A \cdot x + b \quad d \times 1 \]

\[
\begin{bmatrix}
0.7
0.3
0.1
0.5
\end{bmatrix}
\begin{bmatrix}
0.2 & 0.1 & 0.3 & -0.1 & 0.2
0.2 & 0.3 & 0.4 & -0.5 & 0.7
0.1 & 0.1 & 0.7 & 1.0 & 0.3 & 0.1
0.5 & 0.3 & 0.7 & -0.2 & 0.2
-0.2 & 0.2 & 0.1 & 0.2 & 0.3 & 0.2
0.4 & -0.1 & -0.1 & 0.5 & 0.9
\end{bmatrix}
\begin{bmatrix}
1.1
1.1
0.3
0.3
-0.1
-0.1
0.1
-0.1
0.5
0.5
0.7
0.7
\end{bmatrix}
\]

↑ one-hot word symbol
↑ word
↑ looks up column of word embedding matrix
↑ Transformation Matrix maps words in vector space
↑ Bias b
NMT system for translating a single word

Non-linearity: $f = \_\_\_\_

$$V_{v\times 1} \quad d \times V_{d} \quad d \times 1 \quad d \times d \quad d \times 1 \quad V_{t \times d}$$

$w \quad L \quad x = Lw$

$A \quad z = A \times b \quad d \times 1$

$U \quad u = Uz$

$P \quad p = \frac{e^{u_i}}{\sum e^{u_i}}$

$w$ word

Up arrow: one hot word symbol

Up arrow: word decoding matrix

Up arrow: weight on each word

Up arrow: word prob

Up arrow: bias b

Up arrow: transformation matrix maps words in vector space

Up arrow: looks up column of word embedding matrix
Softmax function: Standard map from $\mathbb{R}^V$ to a probability distribution

\[ p_i = \frac{e^{x_i}}{\sum_j e^{x_j}} \]

Exponentiate to make positive

Normalize to give probability
Neural MT: The Bronze Age

[Allen 1987 IEEE 1st ICNN]

3310 En-Es pairs constructed on 31 En, 40 Es words, max 10/11 word sentence; 33 used as test set

The grandfather offered the little girl a book ➔
El abuelo le ofreció un libro a la niña pequeña

Binary encoding of words – 50 inputs, 66 outputs; 1 or 3 hidden 150-unit layers. Ave WER: 1.3 words
Neural MT: The Bronze Age

[Chrisman 1992 *Connection Science*]

Dual-ported RAAM architecture
[Pollack 1990 *Artificial Intelligence*] applied to corpus of 216 parallel pairs of simple En-Es sentences:

You are not angry ⇛ Usted no esta furioso

Split 50/50 as train/test, 75% of sentences correctly translated!
The protests escalated over the weekend <EOS>
The three big wins of Neural MT

1. **End-to-end training**
   
   All parameters are simultaneously optimized to minimize a loss function on the network’s output.

2. **Distributed representations share strength**
   
   Better exploitation of word and phrase similarities.

3. **Better exploitation of context**
   
   NMT can use a much bigger context – both source and partial target text – to translate more accurately.
What wasn't on that list?

1. Explicit use of syntactic or semantic structures

2. Explicit use of discourse structure, anaphora, etc.

3. Black box component models for reordering, transliteration, etc.
The current baseline and its enduring ideas

1b. Ideas connecting Phrase-Based Statistical MT and NMT
**Word alignments**

Phrase-based SMT aligned words in a preprocessing-step, usually using EM

The balance was the territory of the aboriginal people

Le reste appartenait aux autochtones

➡️ Models of attention

[Bahdanau et al. 2014; ICLR 2015]

Part 3b later
Constraints on “distortion” (displacement) and fertility

SMT: Alignment probability depends on positions of the words, and position relative to neighbors.

The likelihood of an alignment depends on how many words align to a certain position.

|                  | \( f \) | \( t(f | e) \) | \( \phi \) | \( n(\phi | e) \) |
|------------------|--------|----------------|---------|----------------|
| *agriculteurs*   | 0.442  | 2              | 0.731   |
| *les*            | 0.418  | 1              | 0.228   |
| *cultivateurs*   | 0.046  | 0              | 0.039   |
| *producteurs*    | 0.021  |                |         |

| \( \phi \) | \( n(\phi | e) \) |
|-----------|----------------|
| 1         | 0.746          |
| 0         | 0.254          |
Constraints on “distortion” (displacement) and fertility

The U.S. island of Guam is maintaining a high state of alert after the Guam airport and its offices both received an e-mail from someone calling himself the Saudi Arabian Osama bin Laden and threatening a biological/chemical attack against public places such as the airport.

Reference translation 1:
The U.S. island of Guam is maintaining a high state of alert after the Guam airport and its offices both received an e-mail from someone calling himself the Saudi Arabian Osama bin Laden and threatening a biological/chemical attack against public places such as the airport.

Reference translation 3:
The US International Airport of Guam and its office has received an email from a self-claimed Arabian millionaire named Laden, which threatens to launch a biochemical attack on such public places as airport. Guam authority has been on alert.

Reference translation 4:
US Guam International Airport and its office received an email from Mr. Bin Laden and other rich businessman from Saudi Arabia. They said there would be biochemistry air raid to Guam Airport and other public places. Guam needs to be in high precaution about this matter.

Machine translation:
The American international airport and its office all receives one calls self the sand Arab rich business and so on electronic mail, which sends out; The threat will be able after public place and so on the airport to start the biochemistry attack, highly alerts after the maintenance.

[Reference translation 2: Guam International Airport and its offices are maintaining a high state of alert after receiving an e-mail that was from a person claiming to be the wealthy Saudi Arabian businessman Bin Laden and that threatened to launch a biological and chemical attack on the airport and other public places.

Reference translation 3: The US International Airport of Guam and its office has received an email from a self-claimed Arabian millionaire named Laden, which threatens to launch a biochemical attack on such public places as airport. Guam authority has been on alert.

Reference translation 4: US Guam International Airport and its office received an email from Mr. Bin Laden and other rich businessman from Saudi Arabia. They said there would be biochemistry air raid to Guam Airport and other public places. Guam needs to be in high precaution about this matter.

[Papineni et al. 2002]
Phrase-Based Statistical MT: Pharaoh/Moses

Source input segmented into phrases
- “phrase” is a subsequence of words – not linguistic phrase

Do we need phrases in NMT?
Or not, as have in-context word translation?
Cf. [Kalchbrenner & Blunsom 2013] source CNN and [Eriguchi, Hashimoto & Tsuruoka 2016] source tree
SMT phrase table weights gave a context-independent translation score

Each phrase is probabilistically translated

- \( P(\text{in spite} | \text{尽管}) \)
- \( P(\text{in spite of the fact} | \text{尽管}) \)

| 开发 || development || (0) || (0) || -2.97 -2.72 -0.86 -0.95 |
| 开发 || development of || (0) || (0) () || -3.41 -2.72 -3.22 -3.50 |
| 进行 监督 || that carries out a supervisory || (1,2,3) (4) || () (0) (0) (1) || 0.0 -3.68 -7.27 -21.24 |
| 进行 监督 || carries out a supervisory || (0,1,2) (3) || (0) (0) (0) (1) || 0.0 -3.68 -7.27 -17.17 |
| 监督 || supervisory || (0) || (0) || -1.03 -0.80 -3.68 -3.24 |
| 监督 检查 || supervisory inspection || (0) (1) || (0) (1) || 0.0 -2.33 -6.07 -4. |
| 检查 || inspection || (0) || (0) || -1.54 -1.53 -2.05 -1.60 |
| 尽管 || in spite || (1) || () (0) || -0.90 -0.50 -3.56 -6.14 |
| 尽管 || in spite of || (1) || () (0) () || -1.11 -0.50 -3.93 -8.68 |
| 尽管 || in spite of the || (1) || () (0) () () || -1.06 -0.50 -4.77 -10.50 |
| 尽管 || in spite of the fact || (1) || () (0) () () () || -1.18 -0.50 -6.54 -18.19 |
Phrase-based SMT: 
Log-linear feature-based MT models

\[
\hat{e} = \arg\max_e 1.9 \times \log P(e) + 1.0 \times \log P(f | e) + 1.1 \times \log \text{length}(e) + \ldots
\]

\[
= \arg\max_e \sum_i w_i \phi_i
\]

We have two things:

- “Features” \( \phi \), such as log translation model score
- Weights \( w \) for each feature for how good it is

The weights were learned

Feature scores from separately trained models
Language Models (LM)

A language model - $P(e)$ - gives the probability of a sequence of words.

Most important feature! Why not just do more with language models?

E.g., generate a translation with LM also conditioned on source $\rightarrow$ Use NLM
MT Decoder: Beam Search

[MT Decoder: Beam Search diagram]

Coverage set: What has been translated

Start

Decoder explores multiple source positions

Should one sample attention?

1st target word

2nd target word

3rd target word

4th target word

All source words covered


→ NMT uses a similar beam decoder. It can be simpler, because contextual conditioning is much better: A beam of ~8 is sufficient.

Work modeling coverage: [Tu, Lu, Liu, Liu, Li, ACL 2016]
An NMT system is an NLM with extra conditioning!

1c. Neural Language Models
There are way too many histories once you’re into a sentence a few words! Exponentially many.

\[ p(x_1, x_2, \ldots, x_T) = \prod_{t=1}^{T} p(x_t|x_1, \ldots, x_{t-1}) \]

[Chain rule]
Traditional Fix: Markov Assumption

An \(n\)th order Markov assumption assumes each word depends only on a short linear history

\[
p(x_1, x_2, \ldots, x_T) = \prod_{t=1}^{T} p(x_t | x_1, \ldots, x_{t-1})
\]

\[
\approx \prod_{t=1}^{T} p(x_t | x_{t-n}, \ldots, x_{t-1})
\]
Problems of Traditional Markov Model Assumptions (1): Sparsity

**Issue:** Very small window gives bad prediction; statistics for even a modest window are sparse

**Example:**

\[
P(w_0|w_{-3}, w_{-2}, w_{-1}) \quad |V| = 100,000 \Rightarrow 10^{15} \text{ contexts}
\]

Most have not been seen

The traditional answer is to use various backoff and smoothing techniques, but no good solution
Neural Language Models

The neural approach [Bengio, Ducharme, Vincent & Jauvin JMLR 2003] represents words as dense distributed vectors so there can be sharing of statistical weight between similar words. Doing just this solves the sparseness problem of conventional n-gram models.
Neural (Probabilistic) Language Model
[Bengio, Ducharme, Vincent & Jauvin JMLR 2003]
Neural (Probabilistic) Language Model
[Bengio, Ducharme, Vincent & Jauvin JMLR 2003]
Neural (Probabilistic) Language Model
[Bengio, Ducharme, Vincent & Jauvinin JMLR 2003]
Problems of Traditional Markov Model Assumptions (2): Context

Issue: Dependency beyond the window is ignored

Example:

the same stump which had impaled the car of many a guest in the past thirty years and which he refused to have removed
A Non-Markovian Language Model

Can we directly model the true conditional probability?

\[
p(x_1, x_2, \ldots, x_T) = \prod_{t=1}^{T} p(x_t | x_1, \ldots, x_{t-1})
\]

Can we build a neural language model for this?

1. Feature extraction: \( h_t = f(x_1, x_2, \ldots, x_t) \)

2. Prediction: \( p(x_{t+1} | x_1, \ldots, x_{t-1}) = g(h_t) \)

How can \( f \) take a variable-length input?
A Non-Markovian Language Model

Can we directly model the **true conditional probability**?

\[
p(x_1, x_2, \ldots, x_T) = \prod_{t=1}^{T} p(x_t | x_1, \ldots, x_{t-1})
\]

Recursive construction of \( f \)

1. Initialization \( h_0 = 0 \)
2. Recursion \( h_t = f(x_t, h_{t-1}) \)

We call \( h_t \) a hidden state or memory

\( h_t \) summarizes the history \((x_1, \ldots, x_t)\)
A Non-Markovian Language Model

Example: \( p(\text{the, cat, is, eating}) \)

1. Initialization: \( h_0 = 0 \)

2. Recursion with Prediction:
   
   \[
   h_1 = f(h_0, \langle \text{bos} \rangle) \rightarrow p(\text{the}) = g(h_1) \\
   h_2 = f(h_1, \text{cat}) \rightarrow p(\text{cat}|\text{the}) = g(h_2) \\
   h_3 = f(h_2, \text{is}) \rightarrow p(\text{is}|\text{the, cat}) = g(h_3) \\
   h_4 = f(h_3, \text{eating}) \rightarrow p(\text{eating}|\text{the, cat, is}) = g(h_4)
   \]

3. Combination:
   
   \[
   p(\text{the, cat, is, eating}) = g(h_1)g(h_2)g(h_3)g(h_4)
   \]

Read, Update and Predict
A Recurrent Neural Network Language Model solves the second problem!

Example: $p(\text{the, cat, is, eating})$

Read, Update and Predict
Building a Recurrent Language Model

Transition Function \( h_t = f(h_{t-1}, x_t) \)

Inputs
i. Current word \( x_t \in \{1, 2, \ldots, |V|\} \)
ii. Previous state \( h_{t-1} \in \mathbb{R}^d \)

Parameters
i. Input weight matrix \( W \in \mathbb{R}^{|V| \times d} \)
ii. Transition weight matrix \( U \in \mathbb{R}^{d \times d} \)
iii. Bias vector \( b \in \mathbb{R}^d \)

\[ h_t = f(h_{t-1}, x_t) \]

\( x_t \) to \( h_0 \) to \( h_1 \) to \( h_2 \) to \( h_3 \) to \( h_4 \)

\( \langle \text{bos} \rangle \) to \( \text{the} \) to \( \text{cat} \) to \( \text{is} \)

\( p(\text{the}) \) to \( p(\text{cat} | \ldots) \) to \( p(\text{is} | \ldots) \) to \( p(\text{eating} | \ldots) \)
Building a Recurrent Language Model

Transition Function \( h_t = f(h_{t-1}, x_t) \)

Naïve Transition Function

\[
f(h_{t-1}, x_t) = \tanh(W \cdot x_t + U h_{t-1} + b)
\]

Element-wise nonlinear transformation

Trainable word vector

Linear transformation of previous state

\[ h_0 \rightarrow h_1 \rightarrow h_2 \rightarrow h_3 \rightarrow h_4 \]

\( p(\text{the}) \quad p(\text{cat|...}) \quad p(\text{is|...}) \quad p(\text{eating|...}) \)

\( \langle \text{bos} \rangle \quad \text{the} \quad \text{cat} \quad \text{is} \)
Building a Recurrent Language Model

Prediction Function $p(x_{t+1} = w | x_{\leq t}) = g_w(h_t)$

Inputs
i. Current state $h_t \in \mathbb{R}^d$

Parameters
i. Softmax matrix $R \in \mathbb{R}^{|V| \times d}$
ii. Bias vector $c \in \mathbb{R}^{|V|}$
Building a Recurrent Language Model

Prediction Function \( p(x_{t+1} = w | x_{\leq t}) = g_w(h_t) \)

\[
p(x_{t+1} = w | x_{\leq t}) = g_w(h_t) = \frac{\exp(R[w] \mathbf{h}_t + c_w)}{\sum_{i=1}^{V} \exp(R[i] \mathbf{h}_t + c_i)}
\]

Compatibility between trainable word vector and hidden state

Normalization

Exponentiate
Training a recurrent language model

Having determined the model form, we:

1. Initialize all parameters of the models, including the word representations with small random numbers

2. Define a loss function: how badly we predict actual next words [log loss or cross-entropy loss]

3. Repeatedly attempt to predict each next word

4. Backpropagate our loss to update all parameters

5. Just doing this learns good word representations and good prediction functions – it’s almost magic
Neural Language Models as MT components

You can just replace the target-side language model of a conventional phrase-based SMT system with an NLM

NLM / Continuous space language models

[Schwenk, Costa-Jussà & Fonollosa 2006; Schwenk 2007; Auli & Gao 2013; Vaswani, Zhao, Fossum & Chiang 2013]

You can use the source as well as target words to predict next target word, usually using phrase alignment

Neural Joint Language Models

[Auli, Galley, Quirk & Zweig 2013; Devlin, Zbib, Huang, Lamar, Schwartz & Makhool 2014]
However, we want to move on to the goal of an end-to-end trained neural translation model!
Recurrent Language Model

*Example* \( p(\text{the, cat, is, eating}) \)

\[
p(\text{the}) \quad p(\text{cat} | \ldots) \quad p(\text{is} | \ldots) \quad p(\text{eating} | \ldots)
\]

*Read, Update and Predict*
2a. Training a Recurrent Language Model

Maximum likelihood estimation with stochastic gradient descent and backpropagation through time
Training a Recurrent Language Model

- Log-probability of one training sentence
  \[
  \log p(x^n_1, x^n_2, \ldots, x^n_{T_n}) = \sum_{t=1}^{T_n} \log p(x^n_t | x^n_1, \ldots, x^n_{t-1})
  \]

- Training set \( D = \{X^1, X^2, \ldots, X^N\} \)

- Log-likelihood Functional
  \[
  \mathcal{L}(\theta, D) = \frac{1}{N} \sum_{n=1}^{N} \sum_{t=1}^{T_n} \log p(x^n_t | x^n_1, \ldots, x^n_{t-1})
  \]

Minimize \(-\mathcal{L}(\theta, D)\) !!
Gradient Descent

• Move **slowly** in the **steepest descent direction**

\[ \theta \leftarrow \theta - \eta \nabla \mathcal{L}(\theta, D) \]

• Computational cost of a single update: \( O(N) \)
• Not suitable for a large corpus
**Stochastic Gradient Descent**

- Estimate the steepest direction with a minibatch
  \[ \nabla \mathcal{L}(\theta, D) \approx \nabla \mathcal{L}(\theta, \{X^1, \ldots, X^n\}) \]

- Until the convergence (w.r.t. a validation set)
  \[ |\mathcal{L}(\theta, D_{\text{val}}) - \mathcal{L}(\theta - \eta \mathcal{L}(\theta, D), D_{\text{val}})| \leq \epsilon \]
Not trivial to build a minibatch

1. Padding and Masking: *suitable for GPU’s, but wasteful*
   - *Wasted computation*
Stochastic Gradient Descent

1. Padding and Masking: suitable for GPU’s, but wasteful
   - Wasted computation

<table>
<thead>
<tr>
<th>Sentence 1</th>
<th>0’s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentence 2</td>
<td>0’s</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Sentence 3</td>
<td></td>
</tr>
<tr>
<td>Sentence 4</td>
<td>0’s</td>
</tr>
</tbody>
</table>

2. Smarter Padding and Masking: minimize the waste
   - Ensure that the length differences are minimal.
   - Sort the sentences and sequentially build a minibatch

<table>
<thead>
<tr>
<th>Sentence 1</th>
<th>0’s</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sentence 2</td>
<td>0’s</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Sentence 3</td>
<td>0’s</td>
</tr>
<tr>
<td>Sentence 4</td>
<td></td>
</tr>
</tbody>
</table>
How do we compute $\nabla \mathcal{L}(\theta, D)$?

- **Cost as a sum of per-sample cost function**

  $$\nabla \mathcal{L}(\theta, D) = \sum_{X \in D} \nabla \mathcal{L}(\theta, X)$$

- **Per-sample cost as a sum of per-step cost functions**

  $$\nabla \mathcal{L}(\theta, X) = \sum_{t=1}^{T} \nabla \log p(x_t| x_{<t}, \theta)$$
How do we compute $\nabla \log p(x_t | x_{<t}, \theta)$?

- Compute per-step cost function from time $t = T$

1. Cost derivative $\partial \log p(x_t | x_{<t}) / \partial g$
2. Gradient w.r.t. $R : \times \partial g / \partial R$
3. Gradient w.r.t. $h_t : \times \partial g / \partial h_t + \partial h_{t+1} / \partial h_t$
4. Gradient w.r.t. $U : \times \partial h_t / \partial U$
5. Gradient w.r.t. $b$ and $W : \times \partial h_t / \partial b$ and $\times \partial h_t / \partial W$
6. Accumulate the gradient and $t \leftarrow t - 1$

Note: I’m abusing math a lot here!!
Intuitively, what’s happening here?

1. Measure the influence of the past on the future

\[
\frac{\partial \log p(x_{t+n} | x_{<t+n})}{\partial h_t} = \frac{\partial \log p(x_{t+n} | x_{<t+n})}{\partial g} \frac{\partial g}{\partial h_{t+n}} \frac{\partial h_{t+n}}{\partial h_{t+n-1}} \ldots \frac{\partial h_{t+1}}{\partial h_t}
\]

2. How does the perturbation at \( t \) affect \( p(x_{t+n} | x_{<t+n}) \)?
Backpropagation through Time

Intuitively, what’s happening here?

1. Measure the influence of the past on the future

\[
\frac{\partial \log p(x_{t+n} | x_{<t+n})}{\partial h_t} = \frac{\partial \log p(x_{t+n} | x_{<t+n})}{\partial g} \frac{\partial g}{\partial h_{t+n}} \frac{\partial h_{t+n}}{\partial h_{t+n-1}} \cdots \frac{\partial h_{t+1}}{\partial h_t}
\]

2. How does the perturbation at \( t \) affect \( p(x_{t+n} | x_{<t+n}) \)?

3. Change the parameters to maximize \( p(x_{t+n} | x_{<t+n}) \)
**Backpropagation through Time**

*Intuitively, what’s happening here?*

1. Measure the influence of the past on the future

\[
\frac{\partial \log p(x_{t+n} | x_{<t+n})}{\partial h_t} = \frac{\partial \log p(x_{t+n} | x_{<t+n})}{\partial g} \frac{\partial g}{\partial h_{t+n}} \frac{\partial h_{t+n}}{\partial h_{t+n-1}} \ldots \frac{\partial h_{t+1}}{\partial h_t}
\]

2. With a naïve transition function

\[
f(h_{t-1}, x_{t-1}) = \tanh(W [x_{t-1}] + Uh_{t-1} + b)
\]

We get

\[
\frac{\partial J_{t+n}}{\partial h_t} = \frac{\partial J_{t+n}}{\partial g} \frac{\partial g}{\partial h_{t+N}} \prod_{n=1}^{N} U^\top \text{diag} \left( \frac{\partial \tanh(a_{t+n})}{\partial a_{t+n}} \right)
\]

**Problematic!**

[Bengio, IEEE 1994]
Gradient either vanishes or explodes

• What happens?

$$\frac{\partial J_{t+n}}{\partial h_t} = \frac{\partial J_{t+n}}{\partial g} \frac{\partial g}{\partial h_{t+N}} \prod_{n=1}^{N} U^\top \text{diag} \left( \frac{\partial \tanh(a_{t+n})}{\partial a_{t+n}} \right)$$

1. The gradient likely explodes if

$$e_{\text{max}} \geq \frac{1}{\max \tanh'(x)} = 1$$

2. The gradient likely vanishes if

$$e_{\text{max}} < \frac{1}{\max \tanh'(x)} = 1,$$ where $e_{\text{max}}$ : largest eigenvalue of $U$

[Backpropagation through Time, Bengio, Simard, Frasconi, TNN1994; Hochreiter, Bengio, Frasconi, Schmidhuber, 2001]
Backpropagation through Time

Addressing Exploding Gradient

• "when gradients explode so does the curvature along v, leading to a wall in the error surface"

• Gradient Clipping
  1. Norm clipping
     \[
     \tilde{\nabla} \left\{ \begin{array}{ll}
     \frac{c}{\| \nabla \|} \nabla & \text{if } \| \nabla \| \geq c \\
     \nabla & \text{otherwise}
     \end{array} \right.
     \]

  2. Element-wise clipping
     \[
     \nabla_i \left\{ \begin{array}{ll}
     \min(c, \| \nabla_i \|) \text{sgn}(\nabla_i) & \text{for all } i \in \{1, \ldots, \dim \nabla\}
     \end{array} \right.
     \]

[Pascanu, Mikolov, Bengio, ICML 2013]
Vanishing gradient is super-problematic

- When we only observe
  \[
  \left\| \frac{\partial h_{t+N}}{\partial h_t} \right\| = \left\| \prod_{n=1}^{N} U^\top \text{diag} \left( \frac{\partial \tanh(a_{t+n})}{\partial a_{t+n}} \right) \right\| \to 0 ,
  \]

- We cannot tell whether
  1. No dependency between \( t \) and \( t+n \) in data, or
  2. Wrong configuration of parameters:

\[
e_{\text{max}}(U) < \frac{1}{\max \tanh'(x)}
\]
2b. Gated Recurrent Units
Vanishing gradient, gated recurrent units and long short-term memory units
Gated Recurrent Unit

- Is the problem with the naïve transition function?
  \[ f(h_{t-1}, x_t) = \tanh(W [x_t] + Uh_{t-1} + b) \]

- With it, the temporal derivative is
  \[ \frac{\partial h_{t+1}}{\partial h_t} = U^\top \frac{\partial \tanh(a)}{\partial a} \]

- It implies that the error must be backpropagated through all the intermediate nodes:
Gated Recurrent Unit

- It implies that the error must backpropagate through all the intermediate nodes:

\[
\begin{array}{cccc}
  & h_t & & h_{t+N} \\
  & U^T & U & U^T & U & U^T & U & U^T & h_{t+N} \\
\end{array}
\]

- Perhaps we can create shortcut connections.
Gated Recurrent Unit

- Perhaps we can create *adaptive* shortcut connections.

\[ f(h_{t-1}, x_t) = u_t \odot \tilde{h}_t + (1 + u_t) \odot h_{t-1} \]

- Candidate Update \( \tilde{h}_t = \text{tanh}(W [x_t] + Uh_{t-1} + b) \)
- Update gate \( u_t = \sigma(W_u [x_t] + U_u h_{t-1} + b_u) \)

\( \odot \): element-wise multiplication
Gated Recurrent Unit

- Let the net prune unnecessary connections adaptively.

\[ f(h_{t-1}, x_t) = u_t \odot \tilde{h}_t + (1 + u_t) \odot h_{t-1} \]

- Candidate Update \( \tilde{h}_t = \tanh(W [x_t] + U (r_t \odot h_{t-1}) + b) \)

- Reset gate \( r_t = \sigma(W_r [x_t] + U_r h_{t-1} + b_r) \)

- Update gate \( u_t = \sigma(W_u [x_t] + U_u h_{t-1} + b_u) \)
Gated Recurrent Unit

tanh-RNN ....

1. Read the whole register $h$
2. Update the whole register

$$h \leftarrow \tanh(W \,[\,x\,] + Uh + b)$$
Gated Recurrent Unit

GRU ...

Registers $h$

Execution
1. Select a readable subset $r$
2. Read the subset $r \odot h$
3. Select a writable subset $u$
4. Update the subset
   $$h \leftarrow u \odot \tilde{h} + (1 - u_t) \odot h$$

Clearly gated recurrent units are much more realistic.
Gated Recurrent Unit

Two most widely used gated recurrent units

Gated Recurrent Unit
[Cho et al., EMNLP2014; Chung, Gulcehre, Cho, Bengio, DLUFL2014]

\[ h_t = u_t \odot \tilde{h}_t + (1 - u_t) \odot h_{t-1} \]
\[ \tilde{h} = \tanh(W [x_t] + U (r_t \odot h_{t-1}) + b) \]
\[ u_t = \sigma(W_u [x_t] + U_u h_{t-1} + b_u) \]
\[ r_t = \sigma(W_r [x_t] + U_r h_{t-1} + b_r) \]

Long Short-Term Memory
[Hochreiter&Schmidhuber, NC1999; Gers, Thesis2001]

\[ h_t = o_t \odot \tanh(c_t) \]
\[ c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \]
\[ \tilde{c}_t = \tanh(W_c [x_t] + U_c h_{t-1} + b_c) \]
\[ o_t = \sigma(W_o [x_t] + U_o h_{t-1} + b_o) \]
\[ i_t = \sigma(W_i [x_t] + U_i h_{t-1} + b_i) \]
\[ f_t = \sigma(W_f [x_t] + U_f h_{t-1} + b_f) \]
Training an RNN

A few well-established + my personal wisdoms

1. Use LSTM or GRU: *makes your life so much simpler*
2. Initialize recurrent matrices to be orthogonal
3. Initialize other matrices with a sensible scale
4. Use adaptive learning rate algorithms: Adam, Adadelta, ...
5. Clip the norm of the gradient: “1” *seems to be a reasonable threshold when used together with adam or adadelta.*
6. *Be patient!*

[Saxe et al., ICLR2014; Ba, Kingma, ICLR2015; Zeiler, arXiv2012; Pascanu et al., ICML2013]
Now, go build and train a recurrent language model!

Any questions?
2c. Conditional Recurrent Language Model
Encoder-Decoder Network for Machine Translation
Recurrent Language Model can

1. Score a given sentence very well

\[ \log p(\text{the, cat, is, sitting, on, a, couch, .}) \]

- Mere reranking significantly improves machine translation and speech recognition quality [Schwenk, 2007; Schwenk, 2012]
- Very good at sentence completion without much task-specific engineering [Tran, ..., Monz, NAACL 2016]

2. Generate a long, coherent text

- Observed earlier by Mikolov [2010, in his thesis] and Sutskever et al. [2011]
Le chat assis sur le tapis.

The cat sat on the mat.
Read a source sentence one symbol at a time.
The last hidden state \( Y \) summarizes the entire source sentence.
Any recurrent activation function can be used:
- Hyperbolic tangent \( \tanh \)
- Gated recurrent unit [Cho et al., 2014]
- Long short-term memory [Sutskever et al., 2014]
- Convolutional network [Kalchbrenner & Blunsom, 2013]
• Usual recurrent language model, except
  1. Transition $z_t = f(z_{t-1}, x_t, Y)$
  2. Backpropagation $\sum_t \partial z_t / \partial Y$

• Same learning strategy as usual: MLE with SGD

$$\mathcal{L}(\theta, D) = \frac{1}{N} \sum_{n=1}^{N} \sum_{t=1}^{T^n} \log p(x_t^n | x_1^n, \ldots, x_{t-1}^n, Y)$$
With conditional recurrent language model,

1. Score a translation

\[ \log p(\text{the, cat, is, sitting, on, a, couch, .} | \text{le, chat, est, assis, sur, un, canapé, .}) =? \]

2. Directly generate a translation

\[ \text{le, chat, est, assis, sur, un, canapé, .} \]

\[ \rightarrow \text{the, cat, is, sitting, on, a, couch, .} \]
2d. Decoding Strategies
Ancestral sampling, greedy decoding and beam search
Decoding (0) – Exhaustive Search

- Simple and exact decoding algorithm
- Score each and every possible translation
- Pick the best one

DO NOT EVEN THINK of TRYING IT OUT!*

* Perhaps with quantum computer and quantum annealing?
Decoding (1) – Ancestral Sampling

- Efficient, unbiased sampling
- One symbol at a time from $\tilde{x}_t \sim x_t | x_{t-1}, \ldots, x_1, Y$
- Until $\tilde{x}_t = \langle \text{eos} \rangle$

The cat sat

$Y = h_7$
Decoding (1) – Ancestral Sampling

- **Pros:**
  1. Unbiased (asymptotically exact)

- **Cons:**
  1. High variance
  2. Pretty inefficient

\[
Y = h_7
\]
Decoding (2) – Greedy Search

- Efficient, but heavily suboptimal search
- Pick the most likely symbol each time

\[ \tilde{x}_t = \arg \max_x \log p(x | x_{<t}, Y) \]

- Until \( \tilde{x}_t = \langle \text{eos} \rangle \)
- Pros:
  1. Super-efficient
     - Both computation and memory
- Cons:
  1. Heavily suboptimal
Decoding (3) – Beam Search

- Pretty, but not quite efficient
- Maintain K hypotheses at a time
  \[ \mathcal{H}_{t-1} = \{ (\tilde{x}_1^1, \tilde{x}_2^1, \ldots, \tilde{x}_{t-1}^1), (\tilde{x}_1^2, \tilde{x}_2^2, \ldots, \tilde{x}_{t-1}^2), \ldots, (\tilde{x}_1^K, \tilde{x}_2^K, \ldots, \tilde{x}_{t-1}^K) \} \]
- Expand each hypothesis
  \[ \mathcal{H}_t^k = \{ (\tilde{x}_1^k, \tilde{x}_2^k, \ldots, \tilde{x}_{t-1}^k, v_1), (\tilde{x}_1^k, \tilde{x}_2^k, \ldots, \tilde{x}_{t-1}^k, v_2), \ldots, (\tilde{x}_1^k, \tilde{x}_2^k, \ldots, \tilde{x}_{t-1}^k, v_{|V|}) \} \]
- Pick top-K hypotheses from the union \( \mathcal{H}_t = \bigcup_{k=1}^{K} \mathcal{B}_k \), where
  \[
  \mathcal{B}_k = \arg \max_{\tilde{X} \in \mathcal{A}_k} \log p(\tilde{X}|Y), \quad \mathcal{A}_k = \mathcal{A}_{k-1} - \mathcal{B}_{k-1}, \quad \text{and} \quad \mathcal{A}_1 = \bigcup_{k'=1}^{K} \mathcal{H}_t^{k'}.
  \]
Decoding (3) – Beam Search

• Asymptotically exact, as \( K \to \infty \)
• But, not necessarily monotonic improvement w.r.t. \( K \)
• \( K \) should be selected to maximize the translation quality on a validation set.
### Decoding

- **En-Cz**: 12m training sentence pairs

<table>
<thead>
<tr>
<th>Strategy</th>
<th># Chains</th>
<th>Valid Set</th>
<th>Test Set</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>NLL</td>
<td>BLEU</td>
</tr>
<tr>
<td>Ancestral Sampling</td>
<td>50</td>
<td>22.98</td>
<td>15.64</td>
</tr>
<tr>
<td>Greedy Decoding</td>
<td>-</td>
<td>27.88</td>
<td>15.50</td>
</tr>
<tr>
<td>Beamsearch</td>
<td>5</td>
<td>20.18</td>
<td>17.03</td>
</tr>
<tr>
<td>Beamsearch</td>
<td>10</td>
<td>19.92</td>
<td>17.13</td>
</tr>
</tbody>
</table>

[Cho, arXiv 2016]
Decoding

- Greedy Search
  - Computationally efficient
  - Not great quality

- Beam Search
  - Computationally expensive
  - Not easy to parallelize
  - Much better quality

*Is there anything in-between?*
2d. Ensemble of Neural MT
Decoding from an ensemble of encoder-decoder’s.
Ensemble of Conditional Recurrent LM

$\text{Le chat assis sur le tapis.}$
Ensemble of Conditional Recurrent LM

- Step-wise Ensemble: $p(x_t^{ens}|x_{<t}^{ens}, Y) = \bigoplus_{m=1}^{M} p(x_t^{m}|x_{<t}^{m}, Y)$

- Ensemble operator $\bigoplus$ implementations
  1. Majority voting scheme (OR):
     $$\bigoplus_{m=1}^{M} p^{ens} = \frac{1}{M} \sum_{m=1}^{M} p^{m}$$
  2. Consensus building scheme (AND):
     $$\bigoplus_{m=1}^{M} p^{ens} = \left( \prod_{m=1}^{M} p^{m} \right)^{1/M}$$

![Bar chart comparing En-De, En-Cs, En-Ru, En-Fi performance](Jung, Cho & Bengio, ACL2016)
Wrap up

1. Training a recurrent language model efficiently
2. Building a better model with gated recurrent units
3. Building a conditional recurrent language model
4. Generating a translation from a trained conditional recurrent language model
Do I smell coffee..?
Have we convinced you about NMT?
3. Advancing NMT

a. The **vocabulary** aspect
   • *Goal*: extend the vocabulary coverage.

b. The **memory** aspect
   • *Goal*: translate long sentences better.

c. The **language complexity** aspect
   • *Goal*: handle more language variations.

d. The **data** aspect
   • *Goal*: utilize more data sources.
3. Advancing NMT

a. The **vocabulary** aspect
   - *Goal*: extend the vocabulary coverage.

b. The **memory** aspect
   - *Goal*: translate long sentences better.

c. The **language complexity** aspect
   - *Goal*: handle more language variations.

d. The **data** aspect
   - *Goal*: utilize more data sources.
The word generation problem

Softmax parameters

Hidden state

\[ P(Je | ...) \]

\[ p_i = \frac{e^{u_i}}{\sum_j e^{u_j}} \]
The word generation problem

- **Word generation problem**

Softmax parameters → Hidden state

\[
P(J_e \mid \ldots) = \frac{e^{u_i}}{\sum_{j} e^{u_j}}
\]

Softmax computation is expensive.
The word generation problem

- Word generation problem
  - Vocabs are modest: 50K.

The ecotax portico in Pont-de-Buis
Le portique écotaxe de Pont-de-Buis

The <unk> portico in <unk>
Le <unk> <unk> de <unk>
**First thought: scale the softmax**

- Lots of ideas from the neural LM literature!

- **Hierarchical models**: tree-structured vocabulary
  - [Morin & Bengio, AISTATS’05], [Mnih & Hinton, NIPS’09].
  - Complex, sensitive to tree structures.

- **Noise-contrastive estimation**: binary classification
  - [Mnih & Teh, ICML’12], [Vaswani et al., EMNLP’13].
  - Different noise samples per training example.*

*We’ll mention a simple fix for this!
Large-vocab NMT

• GPU-friendly.

• *Training*: a subset of the vocabulary at a time.

• *Testing*: smart on the set of possible translations.

Training

- Each time train on a smaller vocab $V' \ll V$

$$|V'|$$

softmax

$$p_i = \frac{e^{u_i}}{\sum_j e^{u_j}}$$

How do we select $V'$?
Training

• Each time train on a smaller vocab $V' \ll V$

![Diagram showing softmax function and equation $p_i = \frac{e^{u_i}}{\sum_j e^{u_j}}$]

• Partition training data in subsets:
  • Each subset has $\tau$ distinct target words, $|V'| = \tau$. 
Training – Segment data

• **Sequentially** select examples: $|V'| = 5$.

$$V' = \{\text{she, loves, cats, he, likes}\}$$
Training – Segment data

- **Sequentially** select examples: $|V'| = 5$.

$V' = \{\text{cats, have, tails, dogs, chase}\}$
Training – Segment data

• **Sequentially** select examples: $|V'| = 5$.

  she loves cats  
  he likes dogs  
  cats have tails  
  dogs have tails  
  dogs chase cats  
  she loves dogs  
  cats hate dogs  

  $V' = \{\text{she, loves, dogs, cats, hate}\}$

• **Practice:** $|V| = 500K$, $|V'| = 30K$ or $50K$. 
Testing – Select candidate words

• $K$ most frequent words: unigram prob.

de, , la . et des les ...
Testing – Select candidate words

- **K** most frequent words: unigram prob.

- Candidate target words
  - **K’** choices per source word. **K’ = 3.**

```
She loves cats
def, la, . et des les...
elle celle ceci aime amour aimer chats chat félin
```
Testing – Select candidate words

- Produce translations within the candidate list
- **Practice**: $K' = 10$ or $20$, $K = 15k$, $30k$, or $50k$. 

**Candidate list**

- elle, celle, ceci
- aime, amour
- chats, chat, félin

$K'$
+ de, la, et des, les...

=
More on large-vocab techniques

• “BlackOut: Speeding up Recurrent Neural Network Language Models with very Large Vocabularies” – [Ji, Vishwanathan, Satish, Anderson, Dubey, ICLR’16].
  • Good survey over many techniques.

• “Simple, Fast Noise Contrastive Estimation for Large RNN Vocabularies” – [Zoph, Vaswani, May, Knight, NAACL’16].
  • Use the same samples per minibatch. GPU efficient.
2nd thought on word generation

- Scaling softmax is insufficient:
  - New names, new numbers, etc., at test time.

- But previous MT models can copy words.

Why can’t NMT?
Copy Mechanism

- Simple way to track target `<unk>`.  
- Treat any NMT as a black box.  
  - Annotate training data.  
  - Post-process translations.

Complementary to softmax scaling!

Training annotation

- Learn **alignments**

  The ecotax portico in Pont-de-Buis
  
  Le portique écotaxe de Pont-de-Buis

- Add **relative positions**

  The <unk> portico in <unk>
  
  Le <unk-1> <unk-1> de <unk-0>
Training annotation

• Learn alignments

The ecotax portico in Pont-de-Buis

Le portique écotaxe de Pont-de-Buis

• Add relative positions

The <unk> portico in <unk>

Le unk_1 unk_{-1} de unk_0
Training annotation

- Learn alignments
  
  The ecotax portico in Pont-de-Buis
  
  Le portique écotaxe de Pont-de-Buis

- Add relative positions
  
  The <unk> portico in <unk>
  
  Le unk₁ unk₁ de unk₀
Post-processing

Test sentence: The <unk> portico in <unk>

Translation: Le portique $\text{unk}_1$ de $\text{unk}_0$
Post-processing

Test sentence: The <unk> portico in <unk>

Translation: Le portique <unk> de <unk>

Dictionary translation: Le portique <écotaxe> de Pont-de-Buis

Post-edit Translation: Le portique <écotaxe> de Pont-de-Buis
Post-processing

Test sentence: The <unk> portico in <unk>

Translation: Le portique <unk> de <unk>

Post-edit Translation: Le portique écotaxé de Pont-de-Buis

Identity copy: Pont-de-Buis

Test sentence: Le portique écotaxé de Pont-de-Buis
Effects of Translating Rare Words

First SOTA NMT!
<table>
<thead>
<tr>
<th></th>
<th>This trader, Richard Usher, left RBS in 2010 and is understand to have be given leave from his current position as European head of forex spot trading at <strong>JPMorgan</strong>.</th>
</tr>
</thead>
<tbody>
<tr>
<td>human</td>
<td>Ce trader, Richard Usher, a quitté RBS en 2010 et aurait été mis suspendu de son poste de responsable européen du trading au comptant pour les devises chez <strong>JPMorgan</strong>.</td>
</tr>
<tr>
<td>trans</td>
<td>Ce unk₀, Richard unk₀, a quitté unk₁ en 2010 et a compris qu'il est autorisé à quitter son poste actuel en tant que leader européen du marché des points de vente au unk₅.</td>
</tr>
<tr>
<td>trans+unk</td>
<td>Ce négociateur, Richard Usher, a quitté RBS en 2010 et a compris qu'il est autorisé à quitter son poste actuel en tant que leader européen du marché des points de vente au <strong>JPMorgan</strong>.</td>
</tr>
</tbody>
</table>

• Translates well long sentences
• Correct: **JPMorgan** vs. **JPMorgan**.
Copy Mechanism – Old but useful!

• Later, we’ll discuss better techniques!

• But it’s useful when adapting to new tasks!
  • Text summarization: [Gu, Lu, Li, Li, ACL’16], [Gulcehre, Ahn, Nallapati, Zhou, Bengio, ACL’16]
  • Semantic parsing: [Jia, Liang, ACL’16]

Learn to decide when to copy.
3. Advancing NMT

a. The *vocabulary* aspect
   
   • *Goal*: extend the vocabulary coverage.

b. The *memory* aspect
   
   • *Goal*: translate long sentences better.

c. The *language complexity* aspect
   
   • *Goal*: handle more language variations.

d. The *data* aspect
   
   • *Goal*: utilize more data sources.
Vanilla seq2seq & long sentences

Problem: fixed-dimensional representations
Attention Mechanism

- **Solution**: random access memory
  - Retrieve as needed.

Started in computer vision!
[Larochelle & Hinton, 2010],
[Denil, Bazzani, Larochelle, Freitas, 2012]
Learning both translation & alignment

Attention Mechanism

Simplified version of (Bahdanau et al., 2015)
Attention Mechanism – Scoring

score\( (h_{t-1}, \bar{h}_s) \)

- Compare target and source hidden states.
Attention Mechanism – Scoring

\[ \text{score}(h_{t-1}, \overline{h}_s) \]

- Compare target and source hidden states.
Attention Mechanism – Scoring

\[ \text{score}(h_{t-1}, \bar{h}_s) \]

• Compare target and source hidden states.
Attention Mechanism – Scoring

$$\text{score}(h_{t-1}, \bar{h}_s)$$

• Compare target and source hidden states.
Attention Mechanism – *Normalization*

\[ a_t(s) = \frac{e^{\text{score}(s)}}{\sum_{s'} e^{\text{score}(s')}} \]

- Convert into *alignment weights*. 
Attention Mechanism – Context

\[ c_t = \sum_s a_t(s) \bar{h}_s \]

- Build context vector: weighted average.
Attention Mechanism – *Hidden State*

- Compute the **next hidden state**.
Attention Mechanisms+

• Simplified mechanism & more functions:

\[
\text{score}(h_t, \bar{h}_s) = \begin{cases} 
    h_t^\top \bar{h}_s \\
    h_t^\top W_a \bar{h}_s \\
    v_a^\top \tanh(W_a[h_t; \bar{h}_s])
\end{cases}
\]

Thang Luong, Hieu Pham, and Chris Manning. Effective Approaches to Attention-based Neural Machine Translation. EMNLP’15.
**Attention Mechanisms+**

- Simplified mechanism & more functions:

\[
\text{score}(h_t, \bar{h}_s) = \begin{cases} 
  h_t^T \bar{h}_s \\
  h_t^T W_a \bar{h}_s \\
  v_a^T \tanh(W_a[h_t; \bar{h}_s])
\end{cases}
\]

Bilinear form: well-adopted.

---

**Sequence-to-Sequence Learning with Attentional Neural Networks**

The attention model is from *Effective Approaches to Attention-based Neural Machine Translation*, Luong et al. EMNLP 2015. We use the *global-general-attention* model with the *input-feeding* approach from the paper. Input-feeding is optional and can be turned off.
Global vs. Local

- Avoid focusing on everything at each time

**Global:** all source states.

**Local:** subset of source states.

Potential for long sequences!

Thang Luong, Hieu Pham, and Chris Manning. *Effective Approaches to Attention-based Neural Machine Translation*. EMNLP’15.
Better Translation of Long Sentences

BLEU scores for different models:
- **Ours, no attn (BLEU 13.9)**
- **Ours, local–p attn (BLEU 20.9)**
- **Ours, best system (BLEU 23.0)**
- **WMT’14 best (BLEU 20.7)**
- **Jeans et al., 2015 (BLEU 21.6)**
Better Translation of Long Sentences

![Graph showing BLEU scores for different methods across sentence lengths.]

- Ours, no attn (BLEU 13.9)
- Ours, local-p attn (BLEU 20.9)
- Ours, best system (BLEU 23.0)
- WMT’14 best (BLEU 20.7)
- Jeans et al., 2015 (BLEU 21.6)

New SOTA!
## Sample English-German translations

<table>
<thead>
<tr>
<th>source</th>
<th>Orlando Bloom and <em>Miranda Kerr</em> still love each other</th>
</tr>
</thead>
<tbody>
<tr>
<td>human</td>
<td>Orlando Bloom und <em>Miranda Kerr</em> lieben sich noch immer</td>
</tr>
<tr>
<td>+attn</td>
<td>Orlando Bloom und <em>Miranda Kerr</em> lieben einander noch immer .</td>
</tr>
<tr>
<td>base</td>
<td>Orlando Bloom und <em>Lucas Miranda</em> lieben einander noch immer .</td>
</tr>
</tbody>
</table>

- Translates names correctly.
### Sample English-German translations

<table>
<thead>
<tr>
<th>source</th>
<th>We’re pleased the FAA recognizes that an enjoyable passenger experience is not incompatible with safety and security, said Roger Dow, CEO of the U.S. Travel Association.</th>
</tr>
</thead>
<tbody>
<tr>
<td>human</td>
<td>Wir freuen uns, dass die FAA erkennt, dass ein angenehmes Passagiererlebnis nicht im Wider- spruch zur Sicherheit steht, sagte Roger Dow, CEO der U.S. Travel Association.</td>
</tr>
<tr>
<td>+attn</td>
<td>Wir freuen uns, dass die FAA anerkennt, dass ein angenehmes ist nicht mit Sicherheit und Sicherheit unvereinbar ist, sagte Roger Dow, CEO der US - die.</td>
</tr>
<tr>
<td>base</td>
<td>Wir freuen uns über die &lt;unk&gt;, dass ein &lt;unk&gt; &lt;unk&gt; mit Sicherheit nicht ist mit Sicherheit und Sicherheit, sagte Roger Cameron, CEO der US - &lt;unk&gt;.</td>
</tr>
</tbody>
</table>

- Translates a doubly-negated phrase correctly.
| source | We’re pleased the FAA recognizes that an enjoyable passenger experience is *not incompatible* with safety and security, said Roger Dow, CEO of the U.S. Travel Association. |
| human | Wir freuen uns, dass die FAA erkennt, dass ein angenehmes Passagiererlebnis nicht im Wider-spruch zur Sicherheit steht, sagte Roger Dow, CEO der U.S. Travel Association. |
| +attn | Wir freuen uns, dass die FAA anerkennt, dass ein angenehmes ist nicht mit Sicherheit und Sicherheit *unvereinbar* ist, sagte Roger Dow, CEO der US - die. |
| base | Wir freuen uns über die <unk>, dass ein <unk> <unk> mit Sicherheit nicht *vereinbar* ist mit Sicherheit und Sicherheit, sagte Roger Cameron, CEO der US - <unk>. |

- Translates a *doubly-negated phrase* correctly.
More Attention! The idea of coverage

• Caption generation

How to not miss an important image patch?

Xu, Ba, Kiros, Cho, Courville, Salakhutdinov, Zemel, Bengio. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. ICML’15
Doubly attention

- Sum to 1 in both dimensions

\[ -\log(P(y|x)) + \lambda \sum_{i}^{L} (1 - \sum_{t}^{C} \alpha_{ti})^2 \]

Per image patch

Sum across caption words

Coverage set exists long time ago in SMT!

Xu, Ba, Kiros, Cho, Courville, Salakhutdinov, Zemel, Bengio. *Show, Attend and Tell: Neural Image Caption Generation with Visual Attention*. ICML’15
Extend to NMT – Linguistic insights

- [Cohn, Hoang, Vymolova, Yao, Dyer, Haffari, NAACL’16]: position (IBM2) + Markov (HMM) + fertility (IBM3-5) + alignment symmetry (BerkeleyAligner).

\[- \log(P(y|x)) + \lambda \sum_{i}^{L} (1 - \sum_{t}^{C} \alpha_{ti})^2\]

- [Tu, Lu, Liu, Liu, Li, ACL’16]: linguistic & NN-based coverage models.
If you feel jetlagged ... see when MT fails

Sale of chicken murder

Go back toward your behind

Deep fried baby

Meat muscle stupid bean sprouts
3. Advancing NMT

a. The **vocabulary** aspect
   • *Goal*: extend the vocabulary coverage.

b. The **memory** aspect
   • *Goal*: translate long sentences better.

c. The **language complexity** aspect
   • *Goal*: handle more language variations.

d. The **data** aspect
   • *Goal*: utilize more data sources.
Extend NMT to more languages

- “Copy” mechanisms are **not sufficient**.
  - Transliteration: Christopher ↦ Kryštof
  - Multi-word alignment: Solar system ↦ Sonnensystem

- Need to handle **large, open vocabulary**
  - Rich morphology: nejneobhospodařovávatelnějšímu ("to the worst farmable one")
  - Informal spelling: goooooood morning !!!!!

Be able to operate at sub-word levels.
Sub-word modeling

Again, lots of inspirations from neural language modeling!
Character-based LSTM

Bi-LSTM builds word representations

(Unfortunately)

Character-based LSTM

\[ \text{the} \quad \text{bank} \quad \text{was} \quad \text{closed} \]

\[ \text{the} \quad \text{bank} \quad \text{was} \]

\[ \text{un} \quad \text{ly} \quad \text{(unfortunately)} \]

Recurrent Language Model

Bi-LSTM builds word representations

Character ConvNet

Highway layer ≅ Like GRU but applied vertically.
Sub-word NMT: two trends

• **Same seq2seq architecture:**
  • Use smaller units.
  • [Sennrich, Haddow, Birch, ACL’16a], [Chung, Cho, Bengio, ACL’16].

• **Hybrid architectures:**
  • RNN for *words* + something else for *characters*.
  • [Costa-Jussà & Fonollosa, ACL’16], [Luong & Manning, ACL’16].
Byte Pair Encoding

• A compression algorithm:
  • Most frequent byte pair $\mapsto$ a new byte.

Replace bytes with character ngrams

Byte Pair Encoding

• A word segmentation algorithm:
  • Start with a vocabulary of characters.
  • Most frequent ngram pairs $\rightarrow$ a new ngram.
Byte Pair Encoding

• A word segmentation algorithm:
  • Start with a vocabulary of characters.
  • Most frequent ngram pairs $\Rightarrow$ a new ngram.

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 |

Vocabulary

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

Start with all characters in vocab

(Example from Sennrich)
Byte Pair Encoding

- A word segmentation algorithm:
  - Start with a vocabulary of characters.
  - Most frequent ngram pairs $\Rightarrow$ a new ngram.

<table>
<thead>
<tr>
<th>Dictionary</th>
<th>Vocabulary</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 low</td>
<td>l, o, w, e, r, n, w, s, t, i, d, es</td>
</tr>
<tr>
<td>2 lower</td>
<td></td>
</tr>
<tr>
<td>6 newest</td>
<td></td>
</tr>
<tr>
<td>3 widest</td>
<td></td>
</tr>
</tbody>
</table>

Add a pair (e, s) with freq 9

(Example from Sennrich)
Byte Pair Encoding

- A word segmentation algorithm:
  - Start with a vocabulary of characters.
  - Most frequent ngram pairs \(\mapsto\) a new ngram.

**Dictionary**

<table>
<thead>
<tr>
<th>5</th>
<th>low</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>lower</td>
</tr>
<tr>
<td>6</td>
<td>new est</td>
</tr>
<tr>
<td>3</td>
<td>widest</td>
</tr>
</tbody>
</table>

**Vocabulary**

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

Add a pair (es, t) with freq 9

(Example from Sennrich)
Byte Pair Encoding

- A word segmentation algorithm:
  - Start with a vocabulary of characters.
  - Most frequent ngram pairs $\mapsto$ a new ngram.

**Dictionary**

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

**Vocabulary**

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

Add a pair (l, o) with freq 7

(Example from Sennrich)
Byte Pair Encoding

- A **word segmentation** algorithm:
  - Start with a vocabulary of **characters**.
  - Most frequent **ngram pairs** $\mapsto$ a new **ngram**.

- **Automatically decide** vocabs for NMT
  - **Word-level**: asinine situation $\mapsto$ Asinin-Situation
  - **BPE-level**: as in ine situation $\mapsto$ As in in- Situation

---

**Top places in WMT 2016!**

https://github.com/rsennrich/nematus
BPE $\mapsto$ Characters

Works for many language pairs.

Sub-word NMT: two trends

- Same seq2seq architecture:
  - Use smaller units.
  - (Sennrich et al., ACL’16), (Chung et al., ACL’16).

- Hybrid architectures:
  - RNN for *words* + something else for *characters*.
  - [Costa-Jussà & Fonollosa, ACL’16], [Luong & Manning, ACL’16].
Character-level Encoder

- Useful when source language is complex:
  - Similar architecture [Kim, Jernite, Sontag, Rush, AAAI’15].

+3 BLEU for German-English translation.

Hybrid NMT

• A best-of-both-worlds architecture:
  • Translate mostly at the word level
  • Only go the character level when needed.

• More than 2 BLEU improvement over copy mechanism.

Hybrid NMT

Word-level (4 layers)

End-to-end training 8-stacking LSTM layers.
2-stage Decoding

- Word-level beam search
2-stage Decoding

- **Word-level** beam search
- **Char-level** beam search for `<unk>`.
English-Czech Results

- Train on WMT’15 data (12M sentence pairs)
  - newstest2015

<table>
<thead>
<tr>
<th>Systems</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Winning WMT’15 (Bojar &amp; Tamchyna, 2015)</td>
<td>18.8</td>
</tr>
<tr>
<td><strong>Word-level</strong> NMT (Jean et al., 2015)</td>
<td>18.3</td>
</tr>
</tbody>
</table>

30x data
3 systems
Large vocab + copy mechanism
English-Czech Results

- Train on WMT’15 data (12M sentence pairs)
  - newstest2015

<table>
<thead>
<tr>
<th>Systems</th>
<th>BLEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Winning WMT’15 (Bojar &amp; Tamchyna, 2015)</td>
<td>18.8</td>
</tr>
<tr>
<td><strong>Word-level</strong> NMT (Jean et al., 2015)</td>
<td>18.3</td>
</tr>
<tr>
<td><strong>Hybrid</strong> NMT (Luong &amp; Manning, 2016)*</td>
<td><strong>20.7</strong></td>
</tr>
</tbody>
</table>

30x data
3 systems
Large vocab
+ copy mechanism

New
SOTA!
Effects of Vocabulary Sizes

More than +2.0 BLEU over copy mechanism!
Rare Word Embeddings

- Word & character-based embeddings.
Rare Word Embeddings

- Word & character-based embeddings.
<table>
<thead>
<tr>
<th>source</th>
<th>Her <strong>11-year-old</strong> daughter, <em>Shani Bart</em>, said it felt a little bit <strong>weird</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>human</td>
<td>Její <strong>jedenáctiletá</strong> dcera Shani Bartová prozradila, že je to trochu zvláštní</td>
</tr>
<tr>
<td>word</td>
<td>Její &lt;unk&gt; dcera &lt;unk&gt; &lt;unk&gt; řekla, že je to trochu divné</td>
</tr>
<tr>
<td>hybrid</td>
<td>Její <strong>11-year-old</strong> dcera Shani, řekla, že je to trochu <strong>divné</strong></td>
</tr>
<tr>
<td>hybrid</td>
<td>Její &lt;unk&gt; dcera, &lt;unk&gt; &lt;unk&gt;, řekla, že je to &lt;unk&gt; &lt;unk&gt;</td>
</tr>
<tr>
<td></td>
<td>Její <strong>jedenáctiletá</strong> dcera, <strong>Graham Bart</strong>, řekla, že cítí trochu <strong>divný</strong></td>
</tr>
</tbody>
</table>

- **Hybrid**: correct, **11-year-old** – **jedenáctiletá**.
## Sample English-Czech translations

<table>
<thead>
<tr>
<th>source</th>
<th>Her <em>11-year-old</em> daughter, <em>Shani Bart</em>, said it felt a little bit <em>weird</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>human</td>
<td>Její <em>jedenáctiletá</em> dcera Shani Bartová prozradila, že je to trochu zvláštní</td>
</tr>
<tr>
<td>word</td>
<td>Její <em>&lt;unk&gt;</em> dcera <em>&lt;unk&gt;</em> řekla, že je to trochu divné</td>
</tr>
<tr>
<td>hybrid</td>
<td>Její <em>11-year-old</em> dcera Shani, řekla, že je to trochu <em>divné</em></td>
</tr>
<tr>
<td></td>
<td>Její <em>jedenáctiletá</em> dcera, <em>Graham Bart</em>, řekla, že cítí trochu <em>divný</em></td>
</tr>
</tbody>
</table>

- **Word-based:** identity copy fails.
3. Advancing NMT

a. The **vocabulary** aspect
   • *Goal*: extend the vocabulary coverage.

b. The **memory** aspect
   • *Goal*: translate long sentences better.

c. The **language complexity** aspect
   • *Goal*: handle more language variations.

d. The **data** aspect
   • *Goal*: utilize more data sources.
Can we utilize other data sources?

- **Multi-lingual**: learn from many language pairs?
- **SMT-inspired**: utilize monolingual data?
- **Multi-task**: combine seq2seq tasks?
Can we utilize other data sources?

- Multi-lingual: learn from many language pairs?
- **SMT-inspired**: utilize monolingual data?
- Multi-task: combine seq2seq tasks?

More later by Cho!
Integrating Language Models

- **Score interpolation:**
  \[
  \log p(y_t = k) = \log p_{TM}(y_t = k) + \beta \log p_{LM}(y_t = k)
  \]

- **Deep fusion:** combine hidden states instead.
  - Controller learns interpolation weights.
  - Better than shallow score interpolation.

---

**Improve low-resource language pairs**

Autoencoders

- **Shared** encoders & decoders: 3 tasks

  ![Diagram showing shared encoders and decoders](image)

- **Small** amount of mono data as **regularization**.
  - +0.9 BLEU improvements

**How to utilize more monolingual data?**

*Thang Luong, Quoc Le, Ilya Sutskever, Oriol Vinyals, Lukasz Kaiser.*

*Multi-task sequence to sequence learning.* ICLR 2016.
Enriching parallel data

- Dummy source sentences

<table>
<thead>
<tr>
<th>English</th>
<th>French</th>
<th>(parallel)</th>
<th>(mono)</th>
</tr>
</thead>
<tbody>
<tr>
<td>She loves cute cats</td>
<td>Elle aime les chats mignons</td>
<td></td>
<td>Elle aime les chiens mignons</td>
</tr>
<tr>
<td>&lt;null&gt;</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Small gain +0.4-1.0 BLEU. Difficult to add more mono data.

Enriching parallel data

- **Synthetic** source sentences

  She loves cute cats  |  Elle aime les chats mignons  (parallel)
  
  She likes cute cats  |  Elle aime les chiens mignons  (mono)

  Back translated

  Large gain +2.1-3.4 BLEU.

Prevent Over-fitting

With synthetic source
4. Future of NMT

a. Multi-task learning

b. Larger context

c. Mobile devices

d. Beyond Maximum Likelihood Estimation
4. Future of NMT

a. Multi-task learning

b. Larger context

c. Mobile devices

d. Beyond Maximum Likelihood Estimation
Multilingual Translation

Language-agnostic Continuous Space

[Dong et al., ACL2015; Luong et al., ICLR2016; Firat et al., NAACL2016]
Multilingual Translation: Expectations

1. Positive language transfer
2. # of parameters grows linearly w.r.t. # of languages
3. Multi-source translation [Zoph&Knight, NAACL2016]

[193]

[Dong et al., ACL2015; Luong et al., ICLR2016; Firat et al., NAACL2016]
Multilingual Translation with Shared Alignment

- Encoder per source language
- Seq. of source symbols $\rightarrow$ Seq. of context vectors

[Firat et al., NAACL2016]
Multilingual Translation with Shared Alignment

- **Shared Attention Mechanism**
  - Target hidden state, source context vector → Attention weight

[Firat et al., NAACL2016]
Multilingual Translation with Shared Alignment

- Decoder per target language
- Aligned context vector → Target symbol

[Firat et al., NAACL2016]
Multilingual Translation: Training

- No multi-way parallel corpus assumed
  - Bilingual sentence pairs only
  - Each sentence pair activates/updates one encoder, decoder and shared attention

[Dong et al., ACL2015; Luong et al., ICLR2016; Firat et al., NAACL2016]
Multilingual Translation: First Result

- 10 language pair-directions
  - En $\rightarrow$ \{Fr, Cs, De, Ru, Fi\} + \{Fr, Cs, De, Ru, Fi\} $\rightarrow$ En
- 60+ million bilingual sentence pairs
- *Comparable to 10 single-pair models*

![Bar charts comparing single and multilingual translation performance for various languages.](chart.png)
Multilingual Translation: Looking Ahead

- Low-resource translation
  - Positive language transfer from high-resource to low-resource language pair-directions

<table>
<thead>
<tr>
<th></th>
<th># Symbols</th>
<th># Sentence</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td># En</td>
<td>Other</td>
</tr>
<tr>
<td>En-Uz</td>
<td>1.361m</td>
<td>1.186m</td>
</tr>
<tr>
<td>En-Es</td>
<td>908.1m</td>
<td>924.9m</td>
</tr>
<tr>
<td>En-Fr</td>
<td>1.837b</td>
<td>1.911b</td>
</tr>
</tbody>
</table>

[Firat et al., under review]
Multilingual Translation: Looking Ahead

- Low-resource translation: Example

Uz-En: 6.45
Uz-En + Tr-En: 9.34
Uz-En + Tr-En + Es-En: 10.34
Uz-En + Tr-En + Es-En + En-Tr: 9.41
Ensemble: 12.99
- 3x Uz-En + Tr-En + Es-En
- 3x Uz-En + Tr-En + Es-En + En-Tr

[Firat et al., 2016c]
Multilingual Translation: Looking Ahead

- Zero-resource translation
  - Translation without any direct parallel resource
**Multilingual Translation: Looking Ahead**

- Zero-resource translation
  - Finetuning with *pseudo*-parallel corpus
    [Sennrich et al., ACL2016]
  - Closely related to unsupervised learning

![Diagram showing pseudo-corpus generation and finetuning processes.](Image)
Multilingual Translation: Looking Ahead

- Zero-resource translation
  - Some initial result, but long way to go…

<table>
<thead>
<tr>
<th>Pivot</th>
<th>Many-to-1</th>
<th>Pseudo Parallel Corpus</th>
<th>True Parallel Corpus</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1k 10k 100k 1m</td>
<td>1k 10k 100k 1m</td>
</tr>
<tr>
<td>√</td>
<td>No Finetuning</td>
<td>Dev: 20.64, Test 20.4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Dev</td>
<td>0.28 10.16 15.61 17.59</td>
<td>0.1 8.45 16.2 20.59</td>
</tr>
<tr>
<td></td>
<td>Test</td>
<td>0.47 10.14 15.41 17.61</td>
<td>0.12 8.18 15.8 19.97</td>
</tr>
<tr>
<td>√</td>
<td>Early</td>
<td>Dev 19.42 21.08 21.7 21.81</td>
<td>8.89 16.89 20.77 22.08</td>
</tr>
<tr>
<td></td>
<td>Late</td>
<td>Test 20.5 20.71 21.06 21.19</td>
<td>15.42 17.95 20.16 20.9</td>
</tr>
</tbody>
</table>

[Firat et al., EMNLP2016]
Multilingual Translation: Looking Ahead

• Multi-modal, Multitask Translation
  [Luong et al., ICLR2016; Caglayan et al., WMT2016]
4. Future of NMT

a. Multi-task learning
b. Larger context
c. Mobile devices
d. Beyond Maximum Likelihood Estimation
Larger-context NMT

- **Beyond sentence level:**
  - Paragraphs, articles, books, etc.

- **Challenges?**
  - Extremely long sequences.
  - Maintain across sentences:
    - Coherent style
    - Discourse structure
Solution: Hierarchical architectures?

- Effective attention mechanism for long sequences
**Solution: Hierarchical architectures?**

- Speech recognition [Chan, Jaity, Le, Vinyals, ICASSP’15].
Solution: Hierarchical architectures?

- Speech recognition [Chan, Jaity, Le, Vinyals, ICASSP’15].

Speech signals: thousands of frames
**Solution: Hierarchical architectures?**

- Speech recognition [Chan, Jaity, Le, Vinyals, ICASSP’15].

**Speech transcription:**

“how much would a woodchuck chuck”

**Speech signals:** thousands of frames
**Solution: Hierarchical architectures?**

- Effective attention mechanism for long sequences
  - Speech recognition [Chan, Jaity, Le, Vinyals, ICASSP’15].

- Tracking states over time
**Solution: Hierarchical architectures?**

- Dialogue systems [Serban, Sordoni, Bengio, Courville, Pineau, AAAI’15].

```plaintext
mom, I don't feel so good <s>  
what's wrong? <s>  
i feel like I'm going to pass out. <s>  
```

```
what's wrong? <s>  
encoder hidden state  
context hidden state  
prediction  
decoder initial hidden state  
```

```
utterance representation  
```

```
utterance representation  
```

```
mom, I don't feel so good <s>  
```
**Solution:** Hierarchical architectures?

- Dialogue systems [Serban, Sordoni, Bengio, Courville, Pineau, AAAI’15].

![Diagram](image-url)

- Utterance level solution:

  - **Utterance representation**
    - Encoder hidden state
    - Decoder initial hidden state
    - Context hidden state
    - Prediction
  
  - **Utterance level**
    - `what’s wrong? </s>`
    - `w_{2,1}, \ldots, w_{2,N_2}`
    - `i feel like i’m going to pass out. </s>`
    - `w_{3,1}, \ldots, w_{3,N_3}`
  
  - **Sample response**
    - `mom, i don’t feel so good </s>`
    - `w_{1,1}, \ldots, w_{1,N_1}`
    - `what’s wrong? </s>`
    - `w_{2,1}, \ldots, w_{2,N_2}`
Solution: Hierarchical architectures

- Dialogue systems [Serban, Sordoni, Bengio, Courville, Pineau, AAAI’15].
**Solution: Hierarchical architectures?**

- **Effective attention mechanism** for long sequences
  - Speech recognition [Chan, Jaity, Le, Vinyals, ICASSP’15].

- **Tracking states** over many sentences
  - Dialogue systems [Serban, Sordoni, Bengio, Courville, Pineau, AAAI’15].

*What else?*
4. Future of NMT

a. Multi-task learning
b. Larger context
c. Mobile devices
d. Beyond Maximum Likelihood Estimation
Mobile devices

There are officially more mobile devices than people in the world

The world is home to 7.2 billion gadgets, and they’re multiplying five times faster than we are

• NMT has **small memory footprint**:
  • No gigantic phrase tables & LMs compared to SMT.

• Still, require **large NNs** for SOTA results

Can we address this?
Model Pruning

- Explore the **redundancy structure** in NMT

![Graph showing BLEU score vs percentage pruned](image)

Model Pruning

- NMT redundancy via pruning & retraining:

![Graph showing performance of pruned models](image)

Figure 1: Performance of pruned models, immediately after pruning and after retraining.

Prune smallest weights

Model Pruning

- NMT redundancy via pruning & retraining:

\[
\text{BLEU score vs. percentage pruned}
\]

Prune 40% with little loss

Prune smallest weights

Model Pruning

• NMT redundancy via **pruning & retraining**:

![Graph showing performance of pruned models, immediately after pruning and after retraining.](image)

- Prune 80% without loss

---

It was just a baby!

Next, really putting NMT onto mobile devices!
Knowledge Distillation

Yoon Kim, Alexander M. Rush.

*Sequence-level knowledge distillation*. EMNLP’16.
Knowledge Distillation

Yoon Kim, Alexander M. Rush.

Sequence-level knowledge distillation. EMNLP’16.
Knowledge Distillation

Yoon Kim, Alexander M. Rush. 
*Sequence-level knowledge distillation.* EMNLP’16.
Knowledge Distillation

Yoon Kim, Alexander M. Rush.
Sequence-level knowledge distillation. EMNLP’16.
Sequence-level knowledge distillation:
- Match the final distribution over sequences
- Beam search to create new training data

Student model: no need beam search.

10 times faster with only 0.2 BLEU loss!

https://github.com/harvardnlp/nmt-android

4. Future of NMT

a. Multi-task learning
b. Larger context
c. Mobile devices
d. Beyond Maximum Likelihood Estimation
Maximum Likelihood Estimation for Sequence Modelling

• Given a ground-truth trajectory, maximize the predictability of a next action: \( \max \log p(x_t | x_{<t}) \)

• Maximum (log-)likelihood estimation

• Two issues
  1. Weak correlation with a true reward
  2. **Mismatch between training and inference**

\[ p(\text{the, cat, is, eating}) \]

\[ p(\text{the}) \quad p(\text{cat} | \ldots) \quad p(\text{is} | \ldots) \quad p(\text{eating} | \ldots) \]

\[ h_0 \quad h_1 \quad h_2 \quad h_3 \]

\[ \text{the} \quad \text{cat} \quad \text{is} \]
Beyond Maximum Likelihood

- Maximize the sequence-wise global loss
- Incorporate inference into training
  - Stochastic inference
    - Policy gradient [Ranzato et al., ICLR2016; Bahdanau et al., arXiv2016]
    - Minimum risk training [Shen et al., ACL2016]
  - Deterministic inference
    - Learning to search [Wiseman & Rush, arXiv2016]
What have we learnt today?

1. History of MT and where Neural MT fits in
2. Language modelling & Neural Machine Translation
   a. Feedforward and recurrent language models
   b. Recurrent neural network and its learning
   c. Conditional language model: learning and decoding
3. Advanced Neural machine translation
   a. Scaling softmax and copy mechanism
   b. Attention-based models
   c. Subword-level translation
   d. Incorporating monolingual corpora
4. And, the future!

https://sites.google.com/site/acl16nmt/home/resources

Thank you!
References (1)

- [Bahdanau et al., ICLR’15] Neural Translation by Jointly Learning to Align and Translate.  
- [Cohn, Hoang, Vymolova, Yao, Dyer, Haffari, NAACL’16] Incorporating Structural Alignment Biases into an Attentional Neural Translation Model.  
  http://www.aclweb.org/anthology/P15-1166
  http://deeplearning.cs.cmu.edu/pdfs/Hochreiter97_lstm.pdf
References (2)

References (3)