Deep Learning

Mohammad Ali Keyvanrad

Lecture 17: Neural Text Generation
• Introduction

• Machine Translation
  – Bidirectional LSTM
  – Attention Mechanism
  – Google’s Multilingual NMT
• **Introduction**

• Machine Translation
  – Bidirectional LSTM
  – Attention Mechanism
  – Google’s Multilingual NMT
Introduction

• Predominant techniques for text generation
  – Template or rule-based systems
  – Require infeasible amounts of hand-engineering

• Deep learning recently achieved great empirical success on some text generation tasks.

• Using end-to-end neural network models
  – An encoder model to produce a hidden representation of the source text
  – Followed by a decoder model to generate the target
Introduction

• Modeling discrete sequences of text tokens
  – Given a sequence $U = (u_1, u_2, \ldots, u_S)$

  $$p(U) = \prod_{t=1}^{S} p(u_t|u_{<t})$$

  $u_1, u_2, \ldots, u_{t-1}$

• General Form of model
  – Input sequence $X$
  – Output sequence $Y$

  $$p(Y|X) = \prod_{t=1}^{T} p(y_t|X, y_{<t})$$
Introduction

- For example: machine translation tasks
  - $X$ might be a sentence in English
  - $Y$ the translated sentence in Chinese

$$p(Y|X) = \prod_{t=1}^{T} p(y_t|X, y_{<t})$$
Introduction

• Other examples

\[ p(Y|X) = \prod_{t=1}^{T} p(y_t|X, y_{<t}) \]

<table>
<thead>
<tr>
<th>Task</th>
<th>(X) (example)</th>
<th>(Y) (example)</th>
</tr>
</thead>
<tbody>
<tr>
<td>language modeling</td>
<td>none (empty sequence)</td>
<td>tokens from news corpus</td>
</tr>
<tr>
<td>machine translation</td>
<td>source sequence in English</td>
<td>target sequence in French</td>
</tr>
<tr>
<td>grammar correction</td>
<td>noisy, ungrammatical sentence</td>
<td>corrected sentence</td>
</tr>
<tr>
<td>summarization</td>
<td>body of news article</td>
<td>headline of article</td>
</tr>
<tr>
<td>dialogue</td>
<td>conversation history</td>
<td>next response in turn</td>
</tr>
<tr>
<td>speech transcription</td>
<td>audio / speech features</td>
<td>text transcript</td>
</tr>
<tr>
<td>image captioning</td>
<td>image</td>
<td>caption describing image</td>
</tr>
<tr>
<td>question answering</td>
<td>supporting text + knowledge base + question</td>
<td>answer</td>
</tr>
</tbody>
</table>
• Introduction

• **Machine Translation**
  – Bidirectional LSTM
  – Attention Mechanism
  – Google’s Multilingual NMT
Machine Translation

• The classic test of language understanding
  – Both language analysis & generation

• Translation is a US$40 billion a year industry

• Huge commercial use
  – Google
    – translates over 100 billion words a day
  – Facebook
  – eBay
Machine Translation

• Machine Translation
  – A naive word-based system would completely fail
    – location of subject, verb, ...
  – Historical Approaches were based on probabilistic models
    – **Translation model**: telling us what a sentence/phrase in a source language most likely translates into
    – **Language model**: telling us how likely a given sentence/phrase is overall.
  – LSTMs can generate arbitrary output sequences after seeing the entire input
    – They can even focus in on specific parts of the input automatically
Progress in Machine Translation

- Phrase-based SMT
- Syntax-based SMT
- Neural MT


Progress trends over the years.
Neural Machine Translation

- Neural Machine Translation
  - The approach of modeling the entire MT process via one big artificial neural network
  - Sometimes we compromise this goal a little
Neural MT: The Bronze Age

- En-Es translator
  - Constructed on 31 En, 40 Es words
  - Max 10 word sentence
  - Binary encoding of words
  - 50 inputs, 66 outputs
  - 1 or 3 hidden 150-unit layers
  - Ave WER: 1.3 words

[Allen 1987 IEEE 1st ICNN]
Neural Machine Translation

• **Sequence-to-sequence (Seq2Seq) model**
  – An end-to-end model made up of two recurrent neural networks (or LSTM)
    – **Encoder**: takes the model’s input sequence as input and encodes it into a fixed-size "context vector"
    – **Decoder**: uses the context vector from above as a "seed“ from which to generate an output sequence.
  – Seq2Seq models are often referred to as "encoder decoder models"
Neural Machine Translation

- **Seq2Seq architecture – encoder**
  - Read the input sequence to Seq2Seq model and generate a fixed-dimensional context vector $C$
  - Encoder will use a recurrent neural network cell – usually an LSTM – to read the input tokens
Neural Machine Translation

- It’s so difficult to compress an arbitrary-length sequence into a single fixed-size vector
- encoder will usually consist of stacked LSTMs
- The final layer’s LSTM hidden state will be used as C.

[Sutskever et al. 2014]
Neural Machine Translation

- A deep recurrent neural network

[Sutskever et al. 2014]
Neural Machine Translation

• **Process the input sequence in reverse**
  – Last thing that the encoder sees will (roughly) corresponds to the first thing that the model outputs
  – This makes it easier for the decoder to "get started" on the output
  – Once it has the first few words translated correctly, it’s much easier to go on to construct a correct sentence
Neural Machine Translation

- **Seq2Seq architecture – decoder**
  - The decoder is also an LSTM network
  - We’ll run all layers of LSTM, one after the other, following up with a softmax on the final
  - We pass output word into the first layer

- Both the encoder and decoder are trained at the same time
Four big wins of Neural MT

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

• **Distributed representations share strength**
  – Better exploitation of word and phrase similarities

• **Better exploitation of context**
  – NMT can use a much bigger context – both source and partial target text – to translate more accurately

• **More fluent text generation**
  – Deep learning text generation is much higher quality
Neural Machine Translation

- NMT aggressively rolled out by industry!
  - 2016/02: Microsoft launches deep neural network MT running offline on Android/iOS.
  - 2016/08: Systran launches purely NMT model
    - One of the oldest machine translation companies that has done extensive work for the United States Department of Defense.
  - 2016/09: Google launches NMT
• Introduction
• Machine Translation
  – Bidirectional LSTM
  – Attention Mechanism
  – Google’s Multilingual NMT
Bidirectional LSTM

- A word can have a dependency on another word before or after it.
- Bidirectional LSTM fix this problem
  - Traversing a sequence in both directions
  - The hidden states are concatenated to get the final context vector
• Introduction

• Machine Translation
  – Bidirectional LSTM
  – **Attention Mechanism**
  – Google’s Multilingual NMT
Attention Mechanism

• Vanilla seq2seq & long sentences
  – Problem: fixed-dimensional representation $Y$
Attention Mechanism

- Solution
  - Pool of source states
Attention Mechanism

- Word alignments
  - Phrase-based SMT aligned words in a preprocessing-step, usually using EM
Attention Mechanism

- Learning both translation & alignment
Attention Mechanism

• Different parts of an input have different levels of significance.
  – Example: “the ball is on the field”
    – "ball“, "on“, and "field“ are the words that are most important

• Different parts of the output may even consider different parts of the input "important“
  – The first word of output is usually based on the first few words of the input
  – The last word is likely based on the last few words of input

• Attention mechanisms make use of this observation.
Attention Mechanism

- Attention mechanisms
  - Decoder network look at the entire input sequence at every decoding step
  - Decoder can then decide what input words are important at any point in time
Attention Mechanism

• Our input is a sequence of words $x_1, \ldots, x_n$ that we want to translate

• Our target sentence is a sequence of words $y_1, \ldots, y_m$

• Encoder
  – Capture contextual representation of each word in the sentence
  – All $h_1, \ldots, h_n$ are the hidden vectors representing the input sentence
  – These vectors are the output of a bi-LSTM for instance
Attention Mechanism

• Decoder
  – We want to compute the hidden states $s_i$ of the decoder
    – $s_{i-1}$ is the previous hidden vector
    – $y_{i-1}$ is the generated word at the previous step
    – $c_i$ is a context vector that capture the context from the original sentence
      – context vector captures relevant information for the $i$-th decoding time step
      – unlike the standard Seq2Seq in which there’s only one context vector

\[ s_i = f(s_{i-1}, y_{i-1}, c_i) \]
Attention Mechanism

- For each hidden vector from the original sentence, compute a score
  \[ e_{i,j} = a(s_{i-1}, h_j) \]
  - **Alignment model**: \( a \) is any function with values in \( \mathbb{R} \)
    - for instance a single layer fully-connected neural network

- Computing the context vector \( c_i \)
  - weighted average of the hidden vectors from the original sentence
  - The vector \( \alpha_i \) is called the attention vector
    \[
    \alpha_{i,j} = \frac{\exp(e_{i,j})}{\sum_{k=1}^{n} \exp(e_{i,k})} \quad \alpha_i = (\alpha_{i,1}, \ldots, \alpha_{i,n}). \quad c_i = \sum_{j=1}^{n} \alpha_{i,j} h_j
    \]
Attention Mechanism

- The graphical illustration of the proposed model
  - generate the $t$-th target word $y_t$ given a source sentence $(x_1; x_2; \ldots; x_T)$
Attention Mechanism

- **Attention vector** for machine translation
  - English to French
  - Each pixel shows the weight $\alpha_{ij}$ of the annotation of the $j$-th source word for the $i$-th target word
Attention Mechanism

- **Alignment model**
  - Needs to be evaluated $T_x \times T_y$ times for each sentence
  - In order to reduce computation, we use a single layer multilayer perceptron

$$a(s_{i-1}, h_j) = v^\top_a \tanh (W_a s_{i-1} + U_a h_j)$$

$$e_{i,j} = a(s_{i-1}, h_j)$$

$$\alpha_{i,j} = \frac{\exp(e_{i,j})}{\sum_{k=1}^{n} \exp(e_{i,k})}$$

$$c_i = \sum_{j=1}^{n} \alpha_{i,j} h_j$$

$$W_a \in \mathbb{R}^{n \times n} \ldots$$

$$U_a \in \mathbb{R}^{n \times 2n}$$

$$y_{t-1} \quad y_t$$

$s_{t-1} \quad s_t \quad \ldots$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$

$\cdot \quad \cdot \quad \cdot$
Attention Mechanism

• **Global vs. Local**
  – Avoid focusing on everything at each time

**Global:** all source states.

**Local:** subset of source states.
Attention Mechanism

- The major advantage of attention-based models is their ability to efficiently translate long sentences.

[Minh-Thang Luong, 2015]
• Introduction

• Machine Translation
  – Bidirectional LSTM
  – Attention Mechanism
  – Google’s Multilingual NMT
Google’s Multilingual NMT

- State-of-the-art in Neural Machine Translation (NMT)
  - Bilingual
Google’s Multilingual NMT

- State-of-the-art in Neural Machine Translation (NMT) - Multilingual
Google’s Multilingual NMT

• Google’s Multilingual NMT System
  – Simplicity: single model
  – Low-resource language improvements
  – Zero-shot translation
    – Translate between language pairs it has never seen in this combination
      – Train: Portuguese → English + English → Spanish
      – Test: Portuguese → Spanish
Google’s Multilingual NMT

- Architecture
Google’s Multilingual NMT

- A token at the beginning of the input sentence to indicate the target language

Hello, how are you? -> ¿Hola como estás?

Add <2es> to indicate that Spanish is the target language

<2es> Hello, how are you? -> ¿Hola como estás?
Dealing with the large output vocabulary

• NMT systems have a hard time dealing with large vocabulary size
  – softmax can be quite expensive to compute
    – Scaling softmax
      – Hierarchical Softmax
    – Reducing vocabulary
      – simply limit the vocabulary size to a small number and replace words outside the vocabulary with a tag <UNK>
  – Handling unknown words

French: Guillaume et Cesar ont une voiture bleue a Lausanne.

English: Guillaume and Cesar have a blue car in Lausanne.
References


• Thang Luong, Hieu Pham, and Chris Manning. “Effective Approaches to Attention-based Neural Machine Translation.” EMNLP’15.


Would you like to be aware of what is better than fasting, prayer and almsgiving?” All replied: “yes.” He said: “It is reconciling between people, for making relations between people strained will eradicate anything.”