Neural Machine Translation by Jointly Learning to Align and Translate

作者: Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio

研究目的

- 為了解決 auto-encoder 在處理 fixed-length vector 時的問題，特別在長句子的 case

研究方式

- 提出稱為 (soft-)alignments 的方式 -> learns to align and translate jointly
  - 每次翻譯的時候 (whenever generating a word)，(soft-)search 句子中最相關的位置
  - Encodes the input sentence into a sequence of vectors and chooses a subset of these vectors

- 文中針對英文對法文的翻譯問題進行測試

- Common Encoder-Decoder (via RNN) ref-1 (https://arxiv.org/abs/1406.1078), ref-2 (http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural):

  - Encoder:
    - 透過當下的 input x_t 以及前一個時間點的 hidden state h_{t-1} 來定義當下的 hidden state,
    $$h_t = f(x_t, h_{t-1})$$
    - 再轉換所有的 hidden state 成 context vector c,
    $$c = q(h_1, \ldots, h_{T_x})$$
    - f, q 為 nonlinear function
  - Decoder:
    - 利用 Encoder 中所得的 context vector c 以及過往所 predict 的字 $y = y_1, \ldots, y_{T_y}$ 進行轉換，求出機率最高的結果 $p(y)$,
    $$p(y) = \prod_{t=1}^{T} p(y_t | \{y_1, \ldots, y_{t-1}\}, c)$$
    其中
    $$p(y_t | \{y_1, \ldots, y_{t-1}\}, c) = g(y_{t-1}, s_t, c) \quad \text{with an RNN}$$
    - g is a nonlinear function, s_t is the hidden state of the RNN.

- Proposed approach:

- Encoder - Bidirectional RNN ref-3 (https://pdfs.semanticscholar.org/4b80/89bc9b49f84de43acc2eb8900035f7d492b2.pdf)

$$h_j = [\overrightarrow{h}_j^{\intercal}; \overleftarrow{h}_j^{\intercal}]$$

- Decoder
  - Conditional probability is

    $$p(y_t|\{y_1, \ldots, y_{t-1}\}, c_i) = g(y_{t-1}, s_t, c)$$

    and s_i is an RNN hidden state for time i, computed by

    $$s_i = f(s_{i-1}, y_{i-1}, c_i)$$

    跟前面不同的是， y_i 會對應到一個獨立的 context vector c_i
  - each h_i contains information about the whole input sequence with a strong focus on the parts surrouding the i-th word of the input sequence.

    $$c_i = \sum_{j=1}^{T_x} a_{ij} h_j$$

    a_{ij} is the weight of each tation h_j,

    $$a_{ij} = \frac{\exp(e_{ij})}{\sum_{k=1}^{T_x} \exp(e_{ik})} \text{ where } e_{ij} = a(s_{i-1}, h_j)$$
  - e_{ij} 的分數根據 s_{i-1} (RNN hidden state) 以及 h_j (j-th hidden layer) 而定 -> 定義了 position j 附近的 input 跟 position i 的 output 間的關係
  - 直觀上，這裡可理解成計算期望值的概念 (expected annotation)，a_{ij} 是 y_i 可轉換成 x_j 的機率，而 c_i 就會是在這個機率下的 expected annotation

- 模型設計
  - 用了兩種模型: RNN Encoder-Decoder, Proposed approach

研究貢獻

- The intuitive of attention mechanism