Paper Reading on “Sequence to Sequence Learning with Neural Networks”

Ilya Sutskever
Google
ilyasu@google.com

Oriol Vinyals
Google
vinyals@google.com

Quoc V. Le
Google
qvl@google.com

***Abstract*—Although Deep Neural Networks work well whenever large labeled training sets are available, they cannot be used to map sequences to sequences. Paper presents a general end to end approach to sequence learning using a LongShort-Term Memory (LSTM) to map the input Sequence to a vector(fixed dimensions) and another deep LSTM to deconde the target sequence from a vector.**
English to French Translation
Calculation Metrics: BLEU (Bilingual Evaluation Understudy) Score Result: 34.8 on entire dataset
Increased results to 36.5 when LSTM was used to rerank 1000 hypotheses
Increase in LSTM performance when reversing the order of words in a sentence, reasons unclear.

I. Introduction

- DNN powerful as they can perform arbitrary parallel computation for a modest number of steps.Neural Networks are related to conventional statistical models and can learn an intricate computation with supervised back propagation.
  Significantly limited to problems having input and output vectors of fixed dimensionality, or inability of DNN to perform when a sequential problem of variable length. example, Speech Recognition, Text Translation
- LSTM, read input sequence one step at a time, to obtain a large fixed vector representation. The second LSTM is a RNN
  $$[28, 23, 30]$$
  conditioned on the input sequence. LSTM, learns on a data with long range temporal dependencies.
- The idea of reversing the words in a long sentence introduced several short term dependencies that made the optimization problem simpler. LSTM, variable input sequence to a fixed dimension vector.

II. Model Description

- Unclear to apply RNN to problems whose output sequences have different lengths with complicated and non-monotonic relationships.
- Sequence learning strategy is to map the input sequence to a fixed-sized vector using one RNN, and then to map the vector to the target sequence with another RNN.
- But RNN is difficult to train for Long term Dependencies, LSTM Introduced.
- The goal of the LSTM is to estimate the conditional probability
  $$p(y_1 \ldots y_T | x_1 \ldots x_T) \tag{1}$$
  where x terms represent an input sequence and y terms represent an output sequence whose length may vary.

$$p(y_1, \ldots, y_{T'} | x_1, \ldots, x_T) = \prod_{t=1}^{T'} p(y_t | v, y_1, \ldots, y_{t-1})$$

- The LSTM computes this conditional probability by first obtaining the fixed- dimensional representation v of the input sequence
  $$x_1 \ldots x_T$$
  given by the last hidden state of the LSTM, and then computing the probability of
  $$y_1 \ldots y_T$$
  with a standard LSTM-LM formulation whose initial hidden state is set to the representation v of
  $$x_1 \ldots x_T$$
  .
- Use of two different LSTMs, each for input and output sequence.
- Deep LSTM, 4 layered, casue it performs better than shallow ones
- Reversing the words of the sentences.

III. Experiments

- WMT’14 English to French MT
- Directly translate the input sequence without any reference SMT system

Fuse Machines Inc.

IV. Dataset details

WMT'14 English to French dataset

Model trained on a subset of 12M sentences consisting of 348M French words and 304M English words

160,000 of the most frequent words for the source language and 80,000 of the most frequent words for the target language.

Replacement of out of vocabulary words with ”UNK” token

V. Decoding and Rescoring

- Trained it by maximizing the log probability of a correct translation T given the source sentence S,so the training objective is:

$$1/|\mathcal{S}| \sum_{(T,S)\in\mathcal{S}} \log p(T|S)$$

- where S is the training set. Once training is complete, we produce translations by finding the most likely translation according to the LSTM:

$$\hat{T} = \arg\max_{T} p(T|S)$$

VI. Reversing the Source Sentences

- Learns much better with long term dependencies when sentences are reversed
- Still unclear why, but introduction of several short term dependencies is key
- Reversing the words in the source sentence, the average distance between corresponding words in the source and target language is unchanged.However, the first few words in the source language are now very close to the first few words in the target language, so the problem's minimal time lag is greatly reduced.

VII. Training details

- 4 layered Deep LSTM with 1000 cells at each layer and 1000 dimensional word embeddings.
- Input Vocabulary of 160K words, output vocabulary of 80K words
- Naive Softmax over 80K words
- LSTM, 380M parameters, 64M are recurrent connections (32M for encoders and 32M for decoders)
- Uniform distribution between -0.08 and 0.08
- Stochastic Gradient without momentum
- Batch Size=128
- Suffers from Vanishing Gradients .
- All sentences within a minibatch were roughly of the same length.

VIII. Results and Conclusion

Best results are obtained with an ensemble of LSTMs that differ in their random initializations and in the random order of minibatches.

- Ensemble of 5 reversed LSTMs, beam size, 12 = **34.81**
- Rescoring the baseline 1000-best with an ensemble of 5 reversed LSTMs = **36.5**

Hence, A large deep LSTM with a limited vocabulary can outperform a stan- dard SMT-based system whose vocabulary is unlimited on a large-scale MT task. Introduction of reversal of words produced surprising results.

- A simple, straightforward and a relatively unoptimized approach can outperform a mature SMT system,

,