CSI 436/536
Introduction to Machine Learning

Other Deep Learning Models

Professor Siwei Lyu
Computer Science
University at Albany, State University of New York
RNN

- Recurrent neural networks
- NN with feedback links
- Unrolling in time leads to a network with (conceptually) infinite number of layers
- Applications: sequential data analysis text, speech, DNA, video, etc

Image from https://colah.github.io/posts/2015-08-Understanding-LSTMs/
RNN

• Model structure
  • Sequential link of RNN cells

• Input from previous time step carries “historic” information from all previous input
  • h is known as the “latent” or “hidden” state
  • From hidden state output can be generated
Training RNN

- BPTT (back propagation through time) with fixed window
  - All weights are tied together

- Vanishing gradient as in feedforward NN
  - “Future” loss has small effect on “present” cell
  - Squashing nonlinearity of hidden state is one problem

Image from https://colah.github.io/posts/2015-08-Understanding-LSTMs/
LSTM (Hochreiter & Schmidhuber '94)

- Stands for long short-term memory cell

- LSTM-RNN cell has
  - A short term “public” memory channel (bottom)
  - A long term “private” memory channel (top)
  - Private channel is used to carry long term history through the use of “gates”

Image from https://colah.github.io/posts/2015-08-Understanding-LSTMs/
Why LSTM

- The public hidden state in vanilla RNN cell “forgets” too fast

- The memory channel carries information that is strongly affected by current input

- There are too many blocks in the way

Image from https://colah.github.io/posts/2015-08-Understanding-LSTMs/
How LSTM solves this

• Providing a more persistent information channel

only simple multiplication and addition on the way

• Modulate this channel with gates: sigmoid function followed by point-wise multiplication
How LSTM solves this

- LSTM-RNN is “stateful” while CNN and RNN are stateless
  - Think about the long-term memory as a global variable in a function

- Three gates: forget gate, input gate, output gate

Image from https://colah.github.io/posts/2015-08-Understanding-LSTMs/
Summary

• RNN solves sequential learning problem through recursion

• One problem with RNN is the fast fading of history, which also leads to vanishing gradients in BPTT

• One solution to this problem is to introduce long term memory as in LSTM or GRU

• Training of LSTM-RNN is much complex and slow so recent trend is to replace RNN completely with CNN of many layers and skip connections
Seq2seq

• Sequence to sequence learning
  • Unlike CNN, we need to map one sequence to another sequence directly
  • Unlike RNN, it is hard to put symbols in order and handle different # of symbols

• Example machine translation

Economic growth has slowed down in recent years.
Das Wirtschaftswachstum hat sich in den letzten Jahren verlangsamt.
La croissance économique s'est ralenti ces dernières années.
Seq2seq

• Applications
  • Neural machine translation
  • Neural speech recognition
  • Video dubbing and captioning

S2VT: A herd of zebras are walking in a field.
Encoder-decoder model [Suskever eval 2014]

- Consider a simple machine translation task

"The weather is nice"  "[START]Il fait beau"

- The model has two modules
  - Encoder: an LSTM-RNN convert input into states (private & public)
  - Decoder: an LSTM-RNN generate output taking encoder’s states

https://blog.keras.io/a-ten-minute-introduction-to-sequence-to-sequence-learning-in-keras.html
Encoder-decoder in seq2seq

- Many variants about how to transfer LSTM-RNN states from encoder to decoder
  - Last state or the average of all states
- Encoder and decoder can be trained individually
Attention in seq2seq

- “No symbols are created equal”, so using only the summary state is too crude
Summary

- Seq2seq learning uses RNN as building blocks but handles locally out of sequential order and different length of input and output

- Two new aspects in seq2seq become independently important in ML
  - Attention mechanism
  - Encoder-decoder architecture
GANs

• Generative Adversary Network (GAN)
• Goodfellow et.al., 2015
• Discriminative training of generative models

Discriminator
Goal: produce counterfeit money that is as similar as real money

Generator
Goal: distinguish between real and counterfeit money
GAN

- The goal of GAN is to learn a generative model for data but using a discriminative NN as helper.
GAN image synthesis

Text description

This bird is blue with white and has a very short beak
This bird has wings that are brown and has a yellow belly
A white bird with a black crown and yellow beak
This bird is white, black, and brown in color, with a brown beak
The bird has small beak, with reddish brown crown and gray belly
This is a small, black bird with a white breast and white on the wingbars.
This bird is white black and yellow in color, with a short black beak

Stage-I images

Stage-II images