Lecture 8: Translation, Seq2Seq, Attention

Shubham Gupta

February 10, 2020

1 Introduction

- SNT and NMT
- NMT: Translate language with one neural network
- Train seq2seq to translate from one language to another.
- Due to this, there will be N^2 translation systems.

2 Greedy decoding

- Choose best word and feed to next step during generation.
- **Problem**:
  - No way to undo decisions
- How to fix?
- Exhaustive search is not good. Wayyyy Too expensive.

3 Bean Search

- At each step, keep track of k most probable partial tranlations. These are called *hypotheses*. k is the beam size
- Score of hypothesis is log probability
- These scores are negative because of log probability
- Higher score is better
- Not guaranteed to find optimal solution
- Moarrr efficient though
- At each stage, store k most probable translations. Compute log prob. Pick the top candidate. Repeat process till the end. Once reached the end, go back in the tree to show the full translation
- **Stopping criterion**

- Each hypothesis can produce ¡END¿ token at different timesteps
- Remove these hypythesis and continue exploring others
- Iterate till reach timestep T
- OR have atleast n completed hypothesis.
- Both these metrics are predefined
- How to select top hypothesis from list of hypothesis?
  * You'll end up chosing shorter one.
  * Longer hypothesis have lower scores
  * Solution: Normalize by length of each hypothesis

3.1 Advanrages of NMT

- Better performance
  * More fluent
  * Better use of context
  * Better use of phrase similarities
- Single NN optimized end-to-end
- Less human engineering effort
  * No feature engineering
  * Same method for all language pairs

3.2 Disadvatanges of NMT

- Less intrepretable
- Difficult to control. Can't specify rules and guidelines for translation

4 Evalualte machine translation

- **BLEU** Bilingual Evaluation Understudy
- Compare machine-written translation to several human-written translations and compute similarity score based on:
  * n-gram precision
  * AND Penalty for too-short system translations aka brevity penalty
- Useful but imperfect

5 Attention

- Why do we need it?
  * Information bottleneck: Forcing all information to be captured in single vector. Problems with long sequences

    - On each step of the decoder, use direct connection to encoder to focus on a part of the sequence
- attention score: Dot product between decoder state and encoder state at any time step t
- Apply softmax to the attention scores to get prob distribution
- Use it to product *Attention output*
- Use it to influence output of the word