Notes on Self-Attention for Speech

Laurent Besacier (LIG)

A) Background (Transformer Model for NMT & Self Attention)

see http://nlp.seas.harvard.edu/2018/04/03/attention.html

Goal

-Reducing sequential computation of RNN-based encoder/decoder models
-Limitation of CNNs for NMT (CONVS2S, ByteNet) : number of operations required to relate signals from two arbitrary input or output positions grows in the distance between positions (linearly for ConvS2S and logarithmically for ByteNet)
-Transformer: this is reduced to a constant number of operations (interesting for long input/output sequences)

Transformer and Self Attention

-Transduction model that do not use sequence aligned RNNs or convolution
-Self attention mechanism : aggregates information from all of the other words of the sequence, generating a new representation per word informed by the entire context ; repeated multiple times in parallel for all words, successively generating new representations.
-Decoding: at each step the model is auto-regressive generates one word at a time, from left to right. It attends not only to the other previously generated words, but also to the final representations generated by the encoder
-see https://ai.googleblog.com/2017/08/transformer-novel-neural-network.html

Multi-head attention

-Attention function described as mapping a query and a set of key-value pairs to an output (query, keys, values, and output are all vectors). The output is computed as a weighted sum of the values (weight assigned to each value is computed by a compatibility function of the query with the corresponding key).
-For instance in Badhanau's NMT, "values" correspond to encoder hidden states, "keys" also correspond to hidden states, "query" correspond to the previous decoder hidden state and "output" corresponds to the context vector obtained through attention
-Multihead attention: linearly project key, values and queries h times with different learned linear projections => jointly learn to attend to information from different representation subspaces at different positions (h: number of heads ; h=8 for instance)
-Different heads can learn different relationships between sequence tokens

Encoder blocks

-Each layer has two sub-layers. The first is a multi-head self-attention mechanism, and the second is a simple, position-wise fully connected feed-forward network.

Decoder blocks

-In addition to the two sub-layers in each encoder layer, the decoder inserts a third sub-layer, which performs multi-head attention over the output of the encoder stack.
-modify the self-attention sub-layer in the decoder stack to prevent positions from attending to subsequent positions (ensures that the predictions for position i can depend only on the known outputs at positions less than i)

Positional encoding

-Model contains no recurrence and no convolution,
-To make use of the order of the sequence, we must inject some information about the relative or absolute position of the tokens in the sequence.
-Add “positional encodings” at the bottoms of the encoder and decoder stacks. They have the same dimension as the embeddings, so that the two can be summed.
-They can be learned or fixed (fixed=sort of trigonometric position encoding - see details in the Transformer paper)

B) Self-Attentional Acoustic Models

Interspeech 2018 - https://arxiv.org/pdf/1803.09519.pdf

Foreword

First attempt to introduce self attention in acoustic modeling ; experimental results are not so great but keep in mind that this is the early stage of self attention for speech

Goal

Apply self-attention to acoustic modeling (for a speech transcription task) and to do so addresses the following issues:
(1) self-attention memory grows quadratically in the sequence length (and for speech, input frame sequences are longer than BPE sequences)
(2) current (Transformer) approach to incorporate position information into the model is not suitable (it is weird and probably will not work to add position vector to handcrafted acoustic features !)
(3) how should we stress the importance of local context in the acoustic signal (biased self attention?)

(1) Sequence length

-we can get very long training utterances, up to 2026 frames (average 800)
-simple downsampling by reshaping the sequence before self attentional layers
-reduce sequence length by factor *a* and increase the vector state dimension accordingly

(2) Position modeling

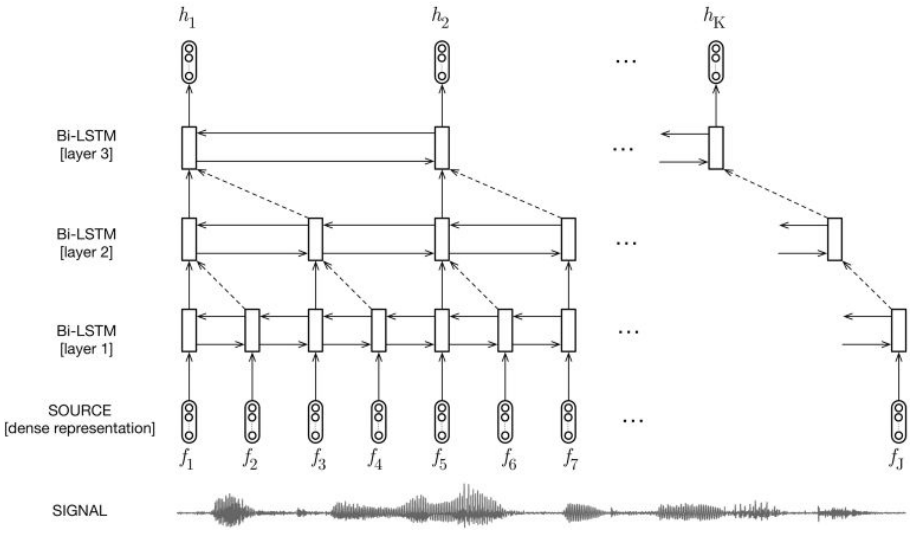
-concatenated position representation instead of summed
(fixed or learnt) to the word embeddings ; 3d approach (unclear) is concatenating separately learned position embeddings to the queries and keys (Q,K)
-re-introducing RNNs (hybrid models) into the encoder: for instance (1) stack two LSTM blocks on top of self attention layers (2) interleaved hybrid model (did not get exactly what they do here) seems that replace FF NN in fig1.c by a LSTM

(3) importance of local context (attention biasing)

-why? intuition that locality of context plays a special role in acoustic modeling
-general idea: bias the self attention toward attending in a local range around each frame
-local masking: all attention weights outside the band b are set to 0, so that the self-attention is restricted to a local region of size b
-soft Gaussian mask (soft gaussian rather than hard square)

Experiments

-ASR on TED Talks (English) ; baseline to be compared with = Listen-Attend-and-Spell (LAS) model



LAS Pyramidal Encoder

-Use XNMT/Dynet

-Input: Speech features: 40 log-filterbank coeff
-Output: Characters (26 English characters, apostrophe, whitespace, and special start-of-sequence and unknown character tokens)

Results

Position modeling (see table 2)
-poor results without RNNs most disappointing aspect of this article
-re-introducing RNNs (hybrid models) significantly improve results
Attention biasing (see table 3)
-positive effect on the results (gaussian masking > local masking)
-btw: Gaussian masking could be applied to Transformer/NMT ?
Comparison to SOTA (see table 1 and 3)
-interleaved hybrid model + attention biasing better than Listen/Attend/Spell (LAS) but it does not beat a more recent baseline (LSTM/NiN model) (Y. Zhang, W. Chan, and N. Jaitly, "Very Deep Convolutional Networks for End-to-End Speech Recognition," in ICASSP 2017) - did not read that one

Interpretability of attention heads

Hypothesize that certain attention heads correspond to certain types of acoustic events (see table 4)
Retrain hybrid models with phones as targets instead of characters and obtain a soft alignment of phoneme labels for each frame
Then correlate these phoneme activations to each of the first layer's 8 attention heads
=> Attention heads seem to focus on different phonetic events/categories

C) ASR with the Transformer in Mandarin Chinese

https://arxiv.org/abs/1804.10752

Foreword
Paper sometimes poorly written… but new SOTA result obtained on a chinese ASR task

Goal
Introduce Transformer model for ASR in Mandarin Chinese
Compare syllable symbols (pinyins with tones) output versus phoneme output

Model
Exact Transformer NMT model

Log-filterbank features linearly transformed to convert the input dimension to the model dimension d_model (i assume this is the dimension of the original transformer implementation they use?)
Nothing is said about position encoding so it is probably summed whereas previous paper said that it diverged (!!!!)
=>some information is missing here...

Experiments

HKUST corpus (telephone speech in Mandarin chinese)
-Input: Speech features: 80 log-filterbank coeff
-Output: 1284 syllables symbols or 118 phoneme symbols + extra tokens
-base model and big model from Transformer initial paper
-a paragraph mentions the need for forced alignments… did not understand why...weird ...

Results

A CER is computed (character error rate) by recontructing the output sequence from phones or syllables to a character sequence (with another seq2seq model)
Comparison with SOTA (see table 3)
-syllable based system on par with a joint CTC-attention based encoder-decoder network with separate RNN-LM (and better than phone targets)