Transformers in Language and Speech Processing

Transformers in Automatic Speech Recognition

Marc Evrard, Camille Guinaudeau, François Yvon

LISN, Université Paris-Saclay

2020-2021

Attention in MNT†

- Core idea: On each step of the decoder, use a **direct connection encoder** to **focus on a particular part** of the source sequence
- Main aims of **attention**:
    - Provide a solution to the seq-to-seq **bottleneck** problem
        - Raymond Mooney (2014): *You can't cram the meaning of a whole %&!$# sentence into a single $&!#* vector!*
        - Decoder can **look directly** at the source, bypassing the bottleneck
    - Help with the **vanishing gradient** problem
        - Provides shortcuts to distant states
    - Provides some **interpretability**
        - Can inspect what the decoder was focusing on
        - We learn a structure (soft alignment), without an explicit loss

† Inspired by Stanford cs224n Lecture 7 (2021)

Attention in general†

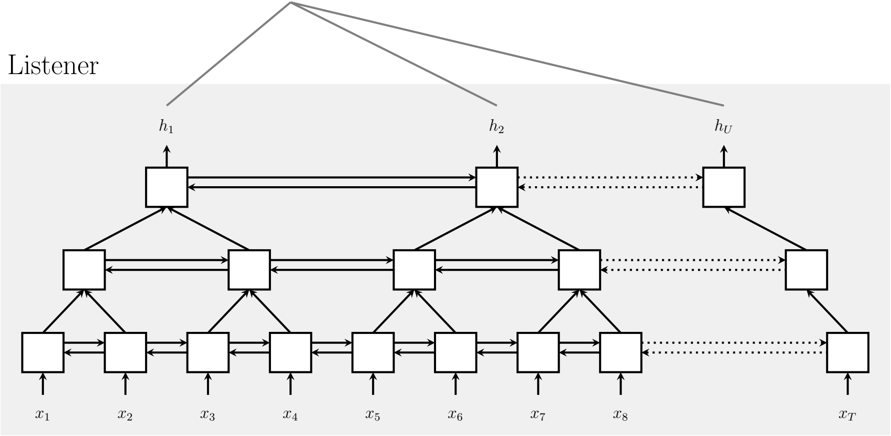
- General definition of attention:
  - Technique to compute a **weighted sum of vector values**, dependent on a **vector query**
- *The query attends to the values*
  - E.g., in the seq2seq + attention model:
    - **Query**: Each **decoder hidden state** (attending to)
    - **Values**: The **encoder hidden states**
  - Intuition: **Attention** is
    - **Weighted sum**: **Selective summary** of the information contained in the values (the query determines which values to focus on)
    - Way to obtain a **fixed-size representation** of a set of representations (values), dependent on some other repr. (the query)

† Inspired by Stanford cs224n Lecture 7 (2021)

Attention in Speech: Listen Attend and Spell

- Listen Attend and Spell (Chan et al. 2016)
- NN that learns to transcribe speech utterances to characters
- Learns all components of a speech recognizer jointly
  (Unlike traditional DNN-HMM models)
- 2 components:
  - **Listener**: Pyramidal RNN encoder with filter bank spectra as inputs
  - **Speller**: Attention-based RNN decoder with characters as outputs
- Produces character sequences without independence assumptions
  (Key improvement over previous end-to-end models)
- Results on a Google voice search task subset:
  - WER = 14.1% without dictionary or LM
  - WER = 10.3% with LM rescoring over the top 32 beams
    (vs. WER = 8.0% for Sainath et al. (2015)'s CLDNN-HMM model)

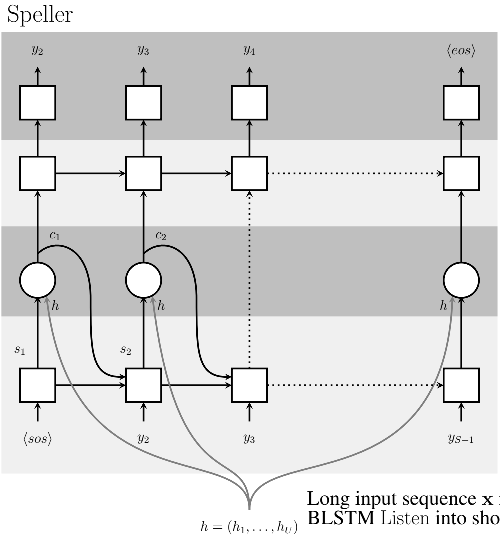
Listen Attend and Spell: Listener Module†



Pyramidal BLSTM encoding input sequence **x** into high-level features **h**

†Fig. from Chan et al. (2016)

Listen Attend and Spell: Speller Module†



Grapheme characters y_i are modelled by the CharacterDistribution

AttentionContext **creates context vector c_i from $\mathbf{h}$ and s_i**

Long input sequence $\mathbf{x}$ is encoded with the pyramidal BLSTM Listen **into shorter sequence $\mathbf{h}$**

† Fig. from Chan et al. (2016)

Early Research (2018)

- Transformer:
  - Why not use only attention for representation?
  - Represent different *features* using different layers of attention
- Early transformer-based architectures in speech recognition: *Speech-Transformer: A No-Recurrence Sequence-to-Sequence Model for Speech Recognition* (Dong et al. 2018)
- Minimal changes in the architecture (vs. the original Transformer):
  - Mainly: Input embeddings through CNNs
- Slightly lower performance than traditional SOTA models (proof of concept: transformer-based ASRs can work)

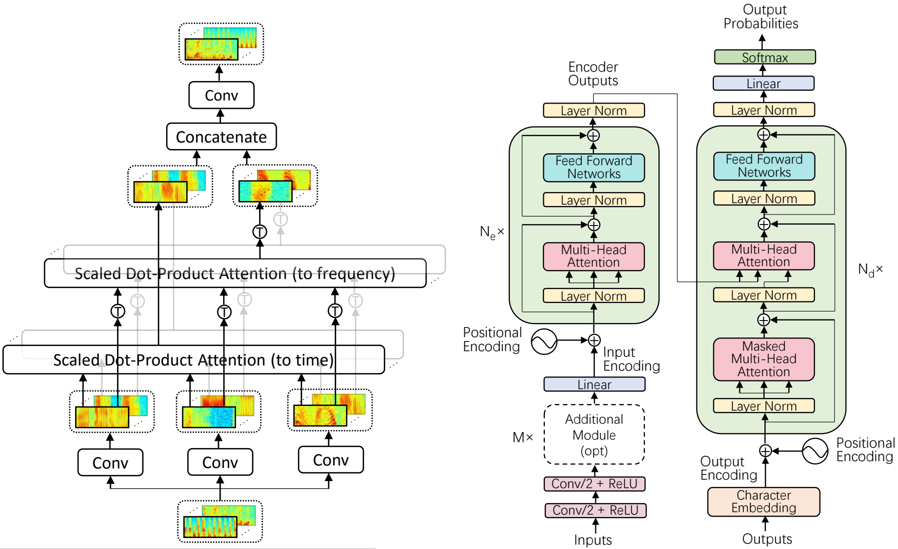
Speech-Transformer (Dong et al. 2018)

- Motivation:
  - Recurrent sequence-to-sequence models using encoder-decoder architecture yielded performances improvements in speech recognition
  - Drawback: Slow (internal recurrence limits the training parallelization)
- Speech-Transformer: Model relying entirely on attention mechanisms to learn the positional dependencies
  - 2D-Attention mechanism attending jointly (time and frequency axes)
- Evaluated on the Wall Street Journal (WSJ) speech recognition dataset (vs. Zhang et al. (2017)'s seq2seq + deep CNN model)
  - Best model: Word error rate (WER) of 10.9% (vs. 10.5%)
  - Training time: 1.2 days on 1 GPU (vs. 5 days on 10 GPUs)

Features

- Input feature sequence: 2-dim. spectrograms (time/frequency)
  - 80-dim. filterbanks: hop size = 10ms and window size = 25ms
  - Including dynamic features: Temporal 1$^{\text{st}}$ and 2$^{\text{nd}}$ order differences
  - Per-speaker mean subtraction and variance normalization
  - Training batch = 20,000 frames
- CNNs are used to model the input spectrograms to mitigate the length mismatch along time (few speech frames per character)
- Output alphabet = 31 classes: 26 lowercase letters, apostrophe, period, space, noise marker, and end-of-sequence tokens
- Learned character-level embeddings are used to convert the character sequence

2D-Attention Mechanism†



†Fig. from Dong et al. (2018)

Proposed Improvements (2019)

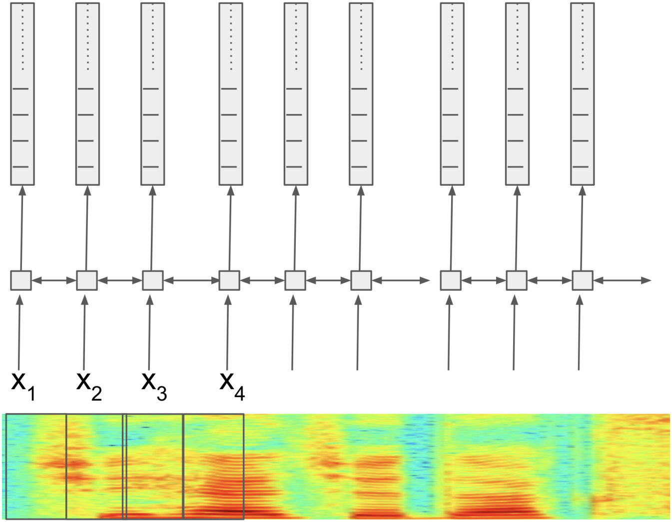
Several key Speech-Transformer improvements in different directions:

- Integration of the Connectionist Temporal Classification (CTC) loss into Speech-Transformers (Karita et al. 2019)
- Replacement of Sinusoidal Positional Encoding (PE) (Mohamed et al. 2019)
- Adaptations for streaming recognition (Moritz et al. 2020)
- Hybrid Architecture: using only the encoder blocks of the transformer for the acoustic modeling, and HMM or RNN modeling for the rest of the architecture (Wang et al. 2020)

Connectionist Temporal Classification Loss

- Connectionist Temporal Classification (CTC) is a loss function associated with RNNs (Graves et al. 2006)
- It is tailored to sequence modeling where timing differs between the input and output sequences
- E.g., typically used for modeling phonemes in speech audio
- Find the best path through a matrix of softmax at each frame (targeting the whole dictionary and a blank token)
- Can be solved efficiently through a dynamic programming algorithm
- Gradients can be calculated from the CTC scores and be back-propagated to update the neural network weights
- CTC is independent of the underlying neural network structure but is often applied at the output of BLSTMs

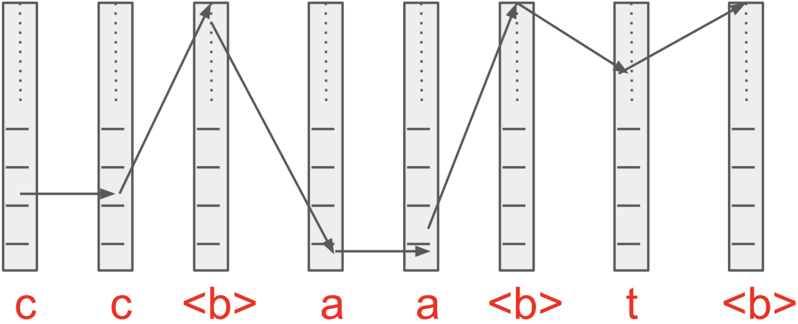
CTC Loss†



†From Stanford cs224n Lecture 12 (2017)

CTC Loss[†]

- Find the best path through the softmax at each frame (for “cat”)



[†] From Stanford cs224n Lecture 12 (2017)

CTC Loss and Transformers

- Karita et al. (2019) proposed to integrate CTC loss into Speech-Transformer
- CTC loss has several advantages:
  - Allows the alignment of audio frames to transcription characters
  - Ease the integration of the language model into the learning process
- They propose a hybrid architecture combining Transformer and RNN-based ASR
- They found that the learning curve converges faster than with an only Transformer architecture
- Evaluations:
  - WER = 4.5% on Wall Street Journal
  - WER = 11.6% on TED-LIUM

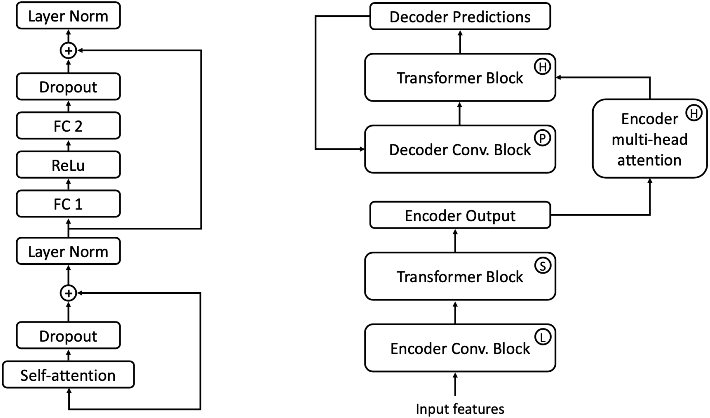
Replacement of Sinusoidal Positional Encoding

- Sinusoidal PE was proposed in the original Transformer paper (Vaswani et al. 2017)
- It may cause performance degradations for longer sequences that have similar acoustic—or semantic—information at different positions (Zhou et al. 2019)
- Alternative approaches:
  - Replacing absolute PE with relative PE (Zhou et al. 2019)
  - Replacing PE with pooling layers (Tsunoo et al. 2019)
  - Replacing PE with trainable convolutional layers (Mohamed et al. 2019)

Positional Encoding through Convolutional Layers

- Combining PE with speech features
- Replacing the sinusoidal PE with convolutionally learned input contextual representations (Mohamed et al. 2019):
  - 2-D convolutional layers over input speech features in the encoder
  - 1-D convolutional layers over previously generated outputs in the decoder
- Transformer's inductive bias is most likely able to mimic convolution filters but yields an unstable optimization process
- Adding early convolutional layers allows the model to learn implicit relative PE, which improves stability
- The model achieves 4.7% and 12.9% WER on the LibriSpeech `test clean` and `test other` subsets, respectively (no extra LM text)

Positional Encoding through Convolutional Layers†



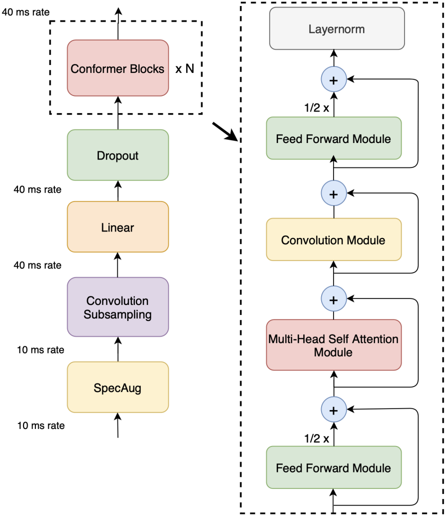
Left: Components of one transformer block
Right: Block diagram of the full end-to-end model

† Fig. from Mohamed et al. (2019)

Conformer (Gulati et al. 2020)

- Main strengths of transformer-based architectures:
  - High efficiency
  - Ability to capture the global context
- CNNs capture local context effectively
- Combine CNNs and transformers to model both local and global contexts
  - Add a convolution module after the Multi-Head Attention block
- **Conformer**: Convolution-augmented transformer for speech recognition
- LibriSpeech: WER = 1.9%/2.1% (with/without using a LM)

Conformer†



†Fig. from Gulati et al. (2020)

Conclusion

- Transformers for ASR is a very active field of research
- Here, just an overview of some chosen paper are given
- In a short amount of time, vast improvements have been made
- Architectures are still changing but seem to converge toward a mix of CNNs and Transformers
- It seems that the revolution lead by Transformers in machine translation (and in NLP in general) may be about to happen in speech processing as well

Bibliography I

Chan, William, Navdeep Jaitly, Quoc Le, and Oriol Vinyals. 2016. "Listen, Attend and Spell: A Neural Network for Large Vocabulary Conversational Speech Recognition." In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4960–64. IEEE.

Dong, Linhao, Shuang Xu, and Bo Xu. 2018. "Speech-Transformer: A No-Recurrence Sequence-to-Sequence Model for Speech Recognition." In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5884–88. IEEE.

Graves, Alex, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. "Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks." In *Proceedings of the 23rd International Conference on Machine Learning*, 369–76.

Gulati, Anmol, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, et al. 2020. "Conformer: Convolution-Augmented Transformer for Speech Recognition." *Proc. Interspeech 2020*, 5036–40.

Karita, Shigeki, Nelson Enrique Yalta Soplin, Shinji Watanabe, Marc Delcroix, Atsunori Ogawa, and Tomohiro Nakatani. 2019. "Improving Transformer-Based End-to-End Speech Recognition with Connectionist Temporal Classification and Language Model Integration." *Proc. Interspeech 2019*, 1408–12.

Bibliography II

Mohamed, Abdelrahman, Dmytro Okhonko, and Luke Zettlemoyer. 2019. "Transformers with Convolutional Context for ASR." *arXiv Preprint arXiv:1904.11660*.

Moritz, Niko, Takaaki Hori, and Jonathan Le. 2020. "Streaming Automatic Speech Recognition with the Transformer Model." In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6074–78. IEEE.

Sainath, Tara N, Oriol Vinyals, Andrew Senior, and Haşim Sak. 2015. "Convolutional, Long Short-Term Memory, Fully Connected Deep Neural Networks." In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4580–84. IEEE.

Tsunoo, Emiru, Yosuke Kashiwagi, Toshiyuki Kumakura, and Shinji Watanabe. 2019. "Transformer ASR with Contextual Block Processing." In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 427–33. IEEE.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. "Attention Is All You Need." In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 6000–6010.

Bibliography III

Wang, Yongqiang, Abdelrahman Mohamed, Due Le, Chunxi Liu, Alex Xiao, Jay Mahadeokar, Hongzhao Huang, Andros Tjandra, Xiaohui Zhang, and Frank Zhang. 2020. "Transformer-Based Acoustic Modeling for Hybrid Speech Recognition." In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6874–78. IEEE.

Zhang, Yu, William Chan, and Navdeep Jaitly. 2017. "Very Deep Convolutional Networks for End-to-End Speech Recognition." In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 4845–49. IEEE.

Zhou, Pan, Ruchao Fan, Wei Chen, and Jia Jia. 2019. "Improving Generalization of Transformer for Speech Recognition with Parallel Schedule Sampling and Relative Positional Embedding." *arXiv Preprint arXiv:1911.00203*.