Deep Learning
Review report: Attention Is All You Need

Johana and Truong

May 2019

1 Summary

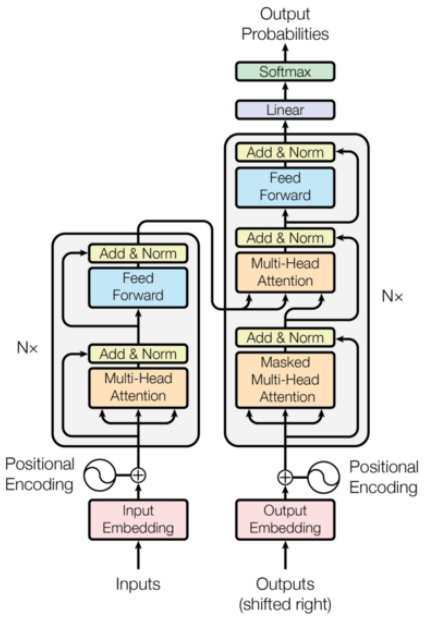

Figure 1: The Transformer

This paper presents a new neural sequence network transduction model named the Transformer. This model is based solely on attention mechanism: without recurrences and convolutions. Transformers have an encoder-decoder structure. The encoding component is composed of a stack of 6 identical encoders and the decoding component is a stack of decoders of the same number. Each encoder has two sub-layers: A multi-head self attention mechanism and a fully connected feed forward network. Each decoder has three sub-layers: A masked

multi-head attention mechanism, a multi-head self attention mechanism and a fully connected feed forward network. One detail in this approach is that each sub-layer has a residual connection followed by a layer-normalization.

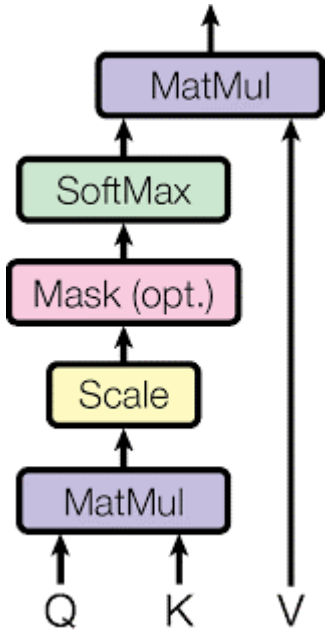

Figure 2: The Scaled Dot-product Attention

Scale dot product attention is calculated as described bellow and leads to multi-head self and masked multi-head attention mechanisms. First,it is necessary to create three vectors from each of input vectors (embeddings) of encoder as well as decoder: Query, Key, and Value vectors. These vectors are created by multiplying the embedding by three matrices that we trained during the training process. Also,they have a smaller dimension than the embedding vector. This choice make the computation of multi-head attention constant. Second, it is needed to score each word of the input sentence against the current word. It determines how much focus to place on other parts of the input sentence as we encode a sequence at a certain position. This score is calculated by taking the dot product of the Query with the Key of the respective sequence we're scoring. Third, scores are divided by the inverse square root of the dimension of the Key to have more stable gradients. The result is passed through a softmax operation. Softmax normalizes the scores so they're all positive and add up to 1. Then, each value vector is multiplied by the softmax score. The output is produced by summing the weighted value vectors. In practice, the attention function is computed with Query, Key, and Value matrices by packing our embeddings into a matrix X, and multiplying it by the weight matrices trained.

Multi-head attention is this calculation done in matrix form. The idea is to

capture many different attentions to each word considering a question in other words, different aspects of the input. To learn diverse representations, the Multi-Head Attention applies various linear transformations to the values, keys, and queries for each "head" of attention. It just applies multiple blocks in parallel, concatenates their outputs, then applies one single linear transformation. The Masked multi-head attention mechanism in the decoder side is done to mask future output to leak into the network. Just before the softmax, positions are set to $-\infty$. This step ensures that the predictions for position k can depend only on the known outputs at positions less than k. Besides, the paper explains the importance of positional encoding and why the choice of sine and cosine functions.

Finally, it is the training regime of the model which will be described and discussed in the section 3. Overall, they used two datasets WMT 2014 English-German and English-French, trained the model in 12 hours with the Adam optimizer and a varied learning rate. They employed residual dropout and label smoothing for normalization.

2 Contribution

Building this architecture is a big step forward for NLP applications. Because of the capacity of learning long-range dependencies in a minimum number of sequential operations and a low complexity per self-attention layer, this architecture is becoming popular. Transformers are used by OpenAI in the GPT2-language model [3]. GPT-2 is trained to predict the next word, given all of the previous words within some text. DeepMind is also using Transformers for AlphaStar [4]. : a program to defeat a top professional Starcraft player

Much recently, transformers have been used to build a new language representation model called BERT which stands for "Bidirectional Encoder Representation model from Transformers" [1]. This model obtains new state-of-arts results on text classification, entity recognition and others NLP tasks.

3 Results and discussion

3.1 Experimental results

There are totally three results including the performance, variances and other-task orientation. Overall, this technique surpassed other popular ones such as RNN, CNN in NLP area and be able to apply for other aspects related in sequences such as signal, sound or genetic analysis.

In translation task, Transformer architecture recorded the highest BLEU score on the WMT 2014 EN-DE and EN-FR comparing with other popular ones. Moreover, the significant advantage is the training cost reduction. While other approaches obeying the fact that the more complex they are, the more training cost we spend, the author debated such that this cost consumption is smaller constraint regardless of the different datasets.

Throughout the development period, they had discovered reasonable hyper-parameters which strongly affects their model. Basically, the size of input keys and values should be large enough to cover all symbol representations, while it also limits the number of attention heads to maximize the capability. In addition, regularization methods (dropout, label smoothing) positively support model's performance.

The last experiment is related in English consistency conversion. Despite the lack of model tuning, the model performs significantly well. Therefore, it yields the preliminary approach for other purposes mentioning in the section 2.

3.2 Discussion

The main advantage of Transformer is that reducing the complexity per layer with the multi-head which provides an effective way for simultaneous training, remarkably saves the computation cost as well as memory consumption. In addition, self-attention layers maintain long-range dependencies in sentences which is the main reason leading to expected result.

About the clarity, reader should have experience in NLP as well as Machine Learning aspects to clearly understand content of the paper. For example, the explanation of keys, values, queries and the method to extract them from input data is ambiguous. The acknowledgment of the embedding algorithms causes challenges for beginner's observation.

Despite of minor drawbacks, the author contributes not only a variable solution in sequence processing (discussed in section 2), but also the framework called Tensor2Tensor[1] integrating the architecture implementation part. Therefore, it supports the research community in the analysis as well as deploying to reality.

4 Lecture link

This paper provides another application of Encoder-Decoder structure which is machine translation task. The usage of principal layers such as linear, softmax and layer normalization is also mentioned in the architecture. Furthermore, they employ residual connections to learn parameters more effectively. In training period, the author optimized their model with the Adam Optimizer, and used Dropout regularization layers to prevent from over-fitting and keep their model learn effectively. Obviously, these things that we have learned in our class are fundamental components which are popularly recognized and applied in not only this model, but also other Neural Network ones.

[1] https://github.com/tensorflow/tensor2tensor

5 Innovation Idea

The Transformer architecture powerfully exhibits in sequence-related tasks, since the attention mechanism maintains the long-range dependencies and encoder-decoder structure plays main role in context determination. In some machine question-answering task as known as chat bot applications, it may be considered as the state-of-the-art approach. Furthermore, Transformer can answer and predict questions, while making related questions could be trained by another one (similar to GAN[2]). For further vision, two or more adversarial/ collaborated Transformers can handle more complexity issues which are acquired long-term dependency chains.

References

[1] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[2] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[3] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1:8, 2019.

[4] Oriol Vinyals, Igor Babuschkin, Junyoung Chung, Michael Mathieu, Max Jaderberg, Wojciech M. Czarnecki, Andrew Dudzik, Aja Huang, Petko Georgiev, Richard Powell, Timo Ewalds, Dan Horgan, Manuel Kroiss, Ivo Danihelka, John Agapiou, Junhyuk Oh, Valentin Dalibard, David Choi, Laurent Sifre, Yury Sulsky, Sasha Vezhnevets, James Molloy, Trevor Cai, David Budden, Tom Paine, Caglar Gulcehre, Ziyu Wang, Tobias Pfaff, Toby Pohlen, Yuhuai Wu, Dani Yogatama, Julia Cohen, Katrina McKinney, Oliver Smith, Tom Schaul, Timothy Lillicrap, Chris Apps, Koray Kavukcuoglu, Demis Hassabis, and David Silver. AlphaStar: Mastering the Real-Time Strategy Game StarCraft II. https://deepmind.com/blog/alphastar-mastering-real-time-strategy-game-starcraft-ii/, 2019.