Summary and Comparison of Sequence Based Models

Hao Liang

January 18, 2023

Abstract

In daily life, we have to deal with loads of sequenced data. For example languages, audios and gene sequences of human being. So in this work I will mainly focus on the work people have developed to deal with the sequence data. I will start with the RNN models and then briefly present the TCN model. Then I will introduce attention and transformers and the powerful models based on it. After that I will compare these models and discuss its applications. At last I presented my perspective of future work.

1 RNN Based Models

1.1 Intuitions of RNN Models

When dealing with sequence data, we need the information of previous data, and that was the intuition of RNN. Since neural networks has shown strong power in machine learning, we added internal state to memory the previous data.

1.2 RNN models

$$h_t = \sigma_h \left(W_h x_t + U_h h_{t-1} + b_h \right)$$
$$y_t = \sigma_y \left(W_y h_t + b_y \right)$$

In RNN[13] model, you may use the memory of last step(h_{t-1}) and the current data(x_t) to predict output and next memory. The process is shown in Figure 1.

1.3 LSTM & GRU

LSTM[9] added forget gate to RNN, shown in Figure 2.
GRU[6] combines the forget and input gates into a single to reduce the number of parameters, shown in Figure 3.

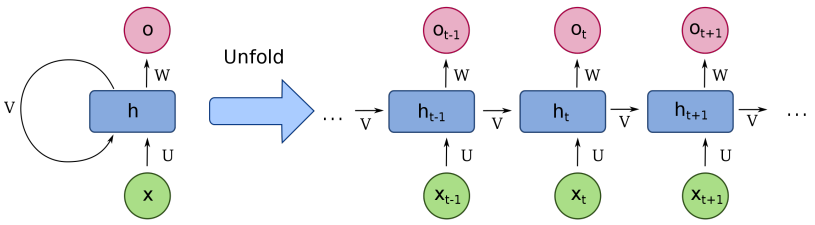

Figure 1: RNN cell [22]

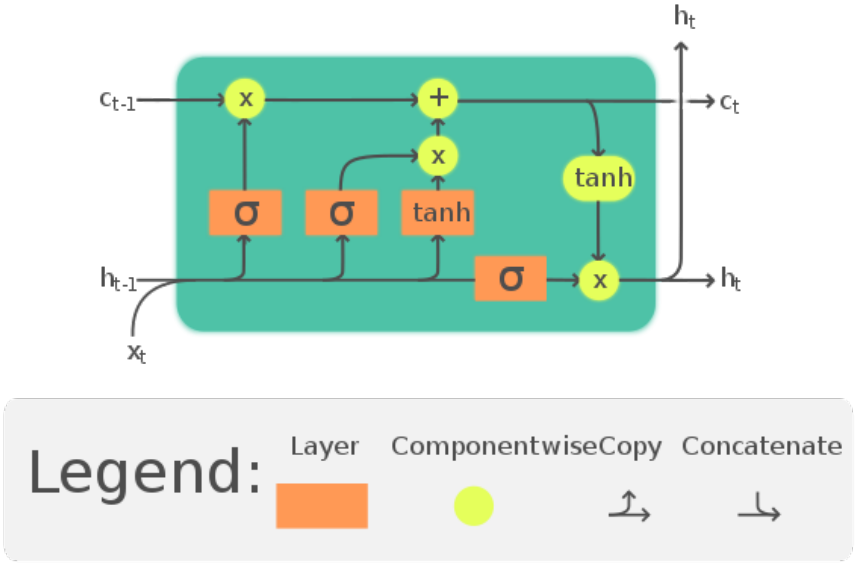

Figure 2: LSTM cell [21]

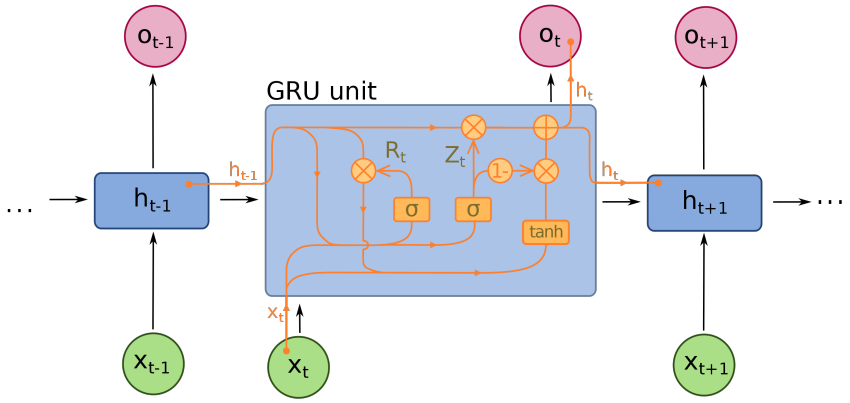

Figure 3: GRU cell [22]

1.4 Bi-RNN

For every step, Bi-RNN[16] can use data from both direction and both sides can be trained simultaneously.
The application of this model is clear. In semantics we need sentences before as well as sentences after to fully understand a word or sentence.

2 TCN Based Models

As convolution has been very successful in CV, then TCN is proposed for sequence data. TCN was proved that it can out perform RNN models across a diverse range of tasks.[3]

2.1 TCN Model

The full name of TCN is temporal convolutional network, which is shown in Figure 4.
It can use data t time ago with only $log(t)$ layers, which means it can have better memory than the LSTM model with same amount of parameters. It is proven by dwipam(github)[1] using a simple code.

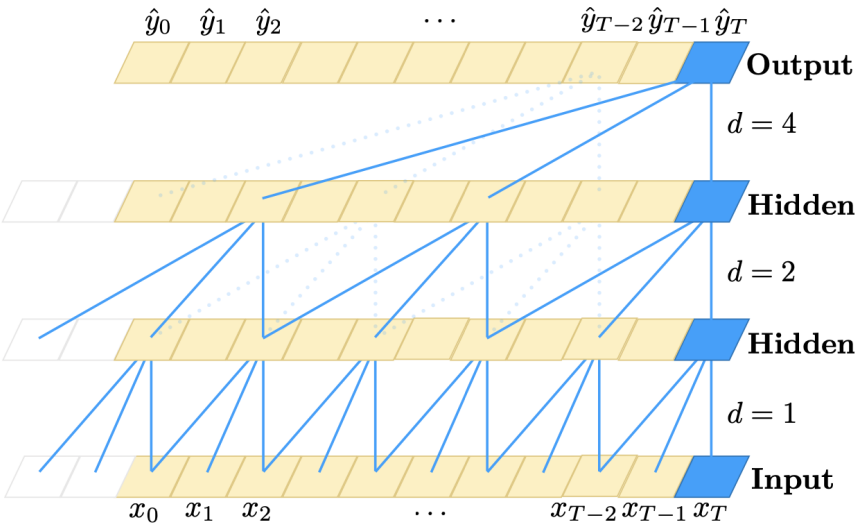

Figure 4: GRU cell [3]

2.2 Compare TCN model and RNN model

Advantages of TCN:

- Parallelism
- More Flexible. TCN can skip some data.
- Stable gradient. RNN often has the problem of gradient disappearance and gradient explosion, which is mainly caused by sharing parameters in different time periods. TCN usually does not have the problem of gradient disappearance and explosion.
- Lower memory.[19]

Disadvantages of TCN:

- We can have Bi-RNN for both sides but TCN is one side.

3 Attention and Transformer Based Models

3.1 Attention

There are lots of Attentions [5].

- Distinctive Attention[2]
- Co-Attention[12]
- For example google proposed self attention.[20]

It has been proven by numerous experiments that attention is an expert at extracting features.

3.2 Transformer

Transformer[20] is a multi-layer self attention(multi-head self attention) model which is proven experimentally to have strong feature extraction ability. [7] The Structure of Transformer is shown in Figure 5.

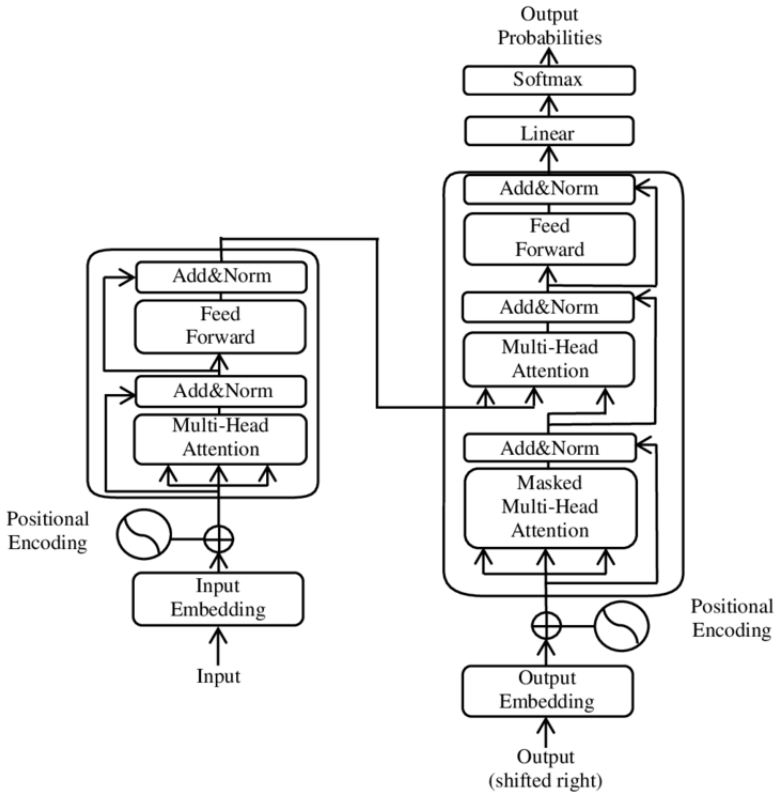

Figure 5: Transformer Structure [23]

3.3 Applications of Attention and Transformer

- The state-of-art NLP models.[4]
- CV models if we need to concentrate on some parts of a figure. For example scene segmentation tasks.[8]
- Attention can also be applied to GNNs to extract feature.[11]

4 Comparison of models

The fact is Attention models are becoming more and more popular recent years. It is the state-of-art model in many research fields, for example the Chatgpt[4] and Alphafold2[10]. However this doesn't mean we should forget RNN and TCN. I will discuss why Attention out performs RNN and TCN and then show the research fields where we can still apply RNN and TCN.

4.1 Attention compared to RNN and TCN

Advantages of Attention:

- Attention is a global algorithm. The advantage of the attention model also becomes more apparent while translating longer sentences. [2]
- Accuracy of long-range dependencies.[18]
- The result of transformer-based model is better.[14]

Disadvantages of Attention:

- The computational complexity of attention is $O(n^2 \cdot d)$. However whether attention has more computational complexity depends on the hyper parameters.[19]
- Attention needs extra positional encoding.[20]

5 Future Work

5.1 Explore the relationship between RNN and GNN

When reading materials of GNN, I found out some of the embedding ideas of Graphs are inspired by language models. Also attention can be applied to both RNN and GNN. As the geometric course is going to explain CNN, RNN, and GNN from geometric aspect, maybe I can bridge the GNN and the RNN models using geometric deep learning.

5.2 Relation between different Deep Learning models

After Reading the Chatgpt paper, I understand Chatgpt can achieve good results because of the Reinforcement Learning[15] from Human Feedback. This technique make me recall the AlphaGo[17] paper. Maybe using deep learning to optimize the parameters first and then use reinforcement learning to adjust the parameters can largely improve the performance of huge Deep Learning models.

5.3 Basic information about me and Preliminary reading plan

- For NLP, I have done a research of NLP last winter, so I am familiar with the latest NLP models and the sequence models.
- For Generative model, I also have done a summer internship training GAN. So I am familiar with VAE, GAN as well as diffusion and the poisson flow models.
- For CV, I know some basic neural network models, so I still need to spend some time to read the latest works.
- For GNN, I know very few and just started to learn some embedding methods for Graphs.

So my preliminary plans of the eight weeks are as follow.

1. For the first week, Sequence based models
2. For week two and three, Latest NLP or CV models.

3. For week four, maybe I will read some more about reinforcement learning and how it can be applied to training Deep Learning models.

4. For week five and six, I will get used to the basic models of graphs. At that time I will be able to read the state-of-art results of GNN.

5. For GNN, I know very few and just started to learn some embedding methods for Graphs.

6. For the last two weeks, I will summarize the relationship between the models using geometric deep learning.

References

[1] Compare of tcn and lstm. https://github.com/dwipam/medium-3.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[3] Shaojie Bai, J Zico Kolter, and Vladlen Koltun. An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271*, 2018.

[4] Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback. *arXiv e-prints*, page arXiv:2204.05862, April 2022.

[5] Sneha Chaudhari, Varun Mithal, Gungor Polatkan, and Rohan Ramanath. An attentive survey of attention models. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 12(5):1–32, 2021.

[6] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv e-prints*, page arXiv:1810.04805, October 2018.

[8] Jun Fu, Jing Liu, Haijie Tian, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. *CoRR*, abs/1809.02983, 2018.

[9] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[10] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *Nature*, 596(7873):583–589, 2021.

[11] John Boaz Lee, Ryan A Rossi, Sungchul Kim, Nesreen K Ahmed, and Eunyee Koh. Attention models in graphs: A survey. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 13(6):1–25, 2019.

[12] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. *Advances in neural information processing systems*, 29, 2016.

[13] Larry R Medsker and LC Jain. Recurrent neural networks. *Design and Applications*, 5:64–67, 2001.

[14] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

[15] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[16] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE transactions on Signal Processing*, 45(11):2673–2681, 1997.

[17] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

[18] Gongbo Tang, Mathias Müller, Annette Rios, and Rico Sennrich. Why self-attention? a targeted evaluation of neural machine translation architectures. *arXiv preprint arXiv:1808.08946*, 2018.

[19] Ashish Vaswani, Samy Bengio, Eugene Brevdo, Francois Chollet, Aidan N Gomez, Stephan Gouws, Llion Jones, Łukasz Kaiser, Nal Kalchbrenner, Niki Parmar, et al. Tensor2tensor for neural machine translation. *arXiv preprint arXiv:1803.07416*, 2018.

[20] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[21] Wikipedia contributors. Long short-term memory — Wikipedia, the free encyclopedia, 2022. [Online; accessed 16-January-2023].

[22] Wikipedia contributors. Recurrent neural network — Wikipedia, the free encyclopedia, 2022. [Online; accessed 16-January-2023].

[23] Wikipedia contributors. Transformer (machine learning model) — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Transformer_(machine_learning_model)&oldid=1132766570, 2023. [Online; accessed 18-January-2023].