THE INTRODUCTION OF RL IN COMBINATORIAL OPTIMIZATION PROBLEM

CHEN JINGJING

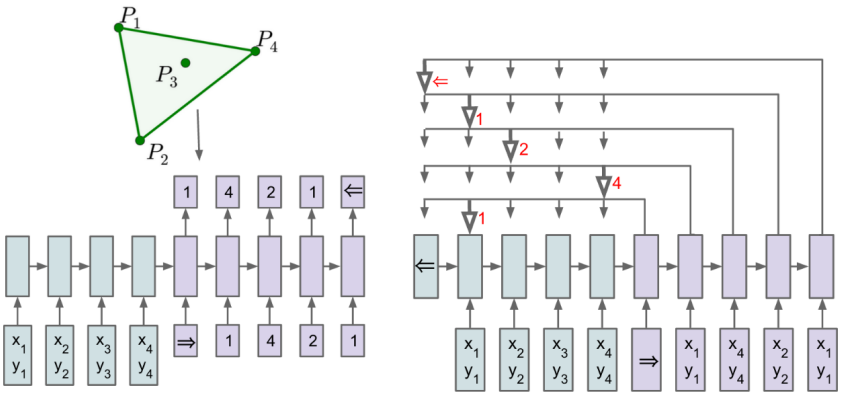
In the last two weeks, I read some papers related to Combinatorial optimization problem(COP) and completed the code which can be used in the co-worked paper with Huiling.

1. Related works

These papers gave the basement of solving Combinatorial Optimization Problem(COP) with reinforcement learning. Based on them, We know how to use RL to solve TSP and VRP, because they are all COP.

In NIPS2015, Vinyals [1] proposed a network structure to solve TSP and Knapsack problem– Pointer Network(Ptr-Net). Pointer Network is based on Seq2Seq and Attention Mechanism.

Seq2Seq model is an important application scenario of RNN, which realizes the function of transforming one sequence into another, and does not require equal length of input sequence and output sequence. An English sentence "who are you" and its corresponding Chinese sentence "ni shi shui" are two different sequences, and what seq2seq model does is to match such sequences. Seq2Seq is consist of two RNNs which are often used in machine translation. They are called Encoder and Decoder respectively. The framework of Seq2Seq is shown in Figure 1(a).



(a) Sequence-to-Sequence

(b) Ptr-Net

and pointer network.png

Figure 1: seq2seq and pointer network

For Encoder, the input is a sequence and after the encoder network(RNN), we get the output(a fixed-size vector representation)-it's a sequence of vectors. For Decoder, the input is the encoded information),after the decoder network(RNN) we get the output(sequence).

Attention mechanism means choose the critical information and it's a bridge to help the Decoder gain the optimal output after it collected the information from the Encoder. Attention mechanism is another neural network which attends to the entire encoder RNN states, this mechanism allows

the encoder focus on the important locations of source sequence to output better sequences. It contributes to the effectiveness of Seq2Seq and the original updating formula of Attention mechanism in Seq2Seq is as follows:

$$\begin{aligned} u_j^i &= v^T \tanh\left(W_1 e_j + W_2 d_i\right) j \in (1, \ldots, n) \\ a_j^i &= \operatorname{softmax}\left(u_j^i\right) \quad j \in (1, \ldots, n) \\ d_i' &= \textstyle\sum_{j=1}^n a_j^i e_j \end{aligned} \tag{1.1}$$

Attention mechanism in Seq2Seq is to assign the weight to the output of Encoder(vector), while the updating formula in Pointer Network is to overcome the drawbacks of Seq2Seq–the length of the output vector tends to be the length of the dictionary, which is predetermined (for example, the English word dictionary has n=8000 words). In combinatorial optimization problems, such as TSP problem, the input is the coordinate sequence of the city, and the output is also the coordinate sequence of the city, while the TSP problem solved every time the city size n is not fixed. The output of Decoder is actually the probability vector with a dimension of n, the same length as the sequence vector entered by encoder.

$$\begin{aligned} u_j^i &= v^T \tanh\left(W_1 e_j + W_2 d_i\right) \quad j \in (1, \ldots, n) \\ p\left(C_i | C_1, \ldots, C_{i-1}, \mathcal{P}\right) &= \operatorname{softmax}\left(u^i\right) \end{aligned} \tag{1.2}$$

They use attention mechanisms to calculate the Softmax probability and use them as Pointers to elements in the input sequence, combine with the input sequence, and finally use supervised methods to train the model. The framework of the Pointer Network is showed in Figure 1(b).

Motivated by pointer networkBello et.al. [2] proposed a novel research work-tackling combinatorial optimization problems using neural networks and reinforcement learning. Considering the Pointer network is trained based on supervised learning which needs an expensive experience to gain the optimal labels. Reinforcement Learning is an neural method for sequential decision problem with the reward as feedback. Integrating the advantage of RL and Pointer Network is a neural idea to solve the COP. Bello et al. used Actor-Critic algorithm to train Pointer Network and obtained the approximate optimal solution for TSP with node length n=100. Using negative tour length as the reward signal, we optimize the parameters of the recurrent neural network using a policy gradient method.

The tour length is

$$L(\pi|s) = \left\|\mathbf{x}_{\pi(n)} - \mathbf{x}_{\pi(1)}\right\|_2 + \sum_i^{n-1} \left\|\mathbf{x}_{\pi(i)} - \mathbf{x}_{\pi(i+1)}\right\|_2 \tag{1.3}$$

The Actor-Critic algorithm is as follows:

$$\begin{aligned} &\cdot \text{ Policy gradient: } \nabla_\theta J(\theta) \approx \tfrac{1}{B} \textstyle\sum_{i=1}^B \left(L\left(\pi_i|s_i\right) - b\left(s_i\right)\right) \nabla_\theta \log p_\theta\left(\pi_i|s_i\right) \\ &\cdot \text{ Critic: } \mathcal{L}\left(\theta_v\right) = \tfrac{1}{B} \textstyle\sum_{i=1}^B \left\|b_{\theta_v}\left(s_i\right) - L\left(\pi_i|s_i\right)\right\|_2^2 \end{aligned} \tag{1.4}$$

Pointing mechanism

$$\begin{aligned} u_i &= \begin{cases} v^\top \cdot \tanh\left(W_{ref} \cdot r_i + W_q \cdot q\right) & \text{if } i \neq \pi(j) \text{ for all } j < i \\ -\infty & \text{otherwise} \end{cases} \\ A\left(ref, q; W_{ref}, W_q, v\right) &\stackrel{\text{def}}{=} \operatorname{softmax}(u) \\ p(\pi(j)|\pi(<j), s) &\stackrel{\text{def}}{=} A\left(enc_{1:n}, \mathrm{dec}_j\right) \end{aligned} \tag{1.5}$$

Beside this, there are two kinds of search strategies: (1)sampling-sample multiple candidate tours from stochastic policy and choose the shortest one. They control the diversity with temperature parameter(in softmax). (2)Active Search-training during inference time and minimizing L on a single test input.

VRP is a dynamic programming problem with the varying demand of each customer. and it's also a sequential decision problem. This paper [3] presents an end-to-end framework for solving VRP problems with RL.

In realitycustomers are dependent. There is no sequential information in inputs, just a set of unordered locations. Once one customer is visited, it would not emerge in the input. While Encoder in the Pointer Network need to be retrained according to the varied demand of customers. It would cost much to update the encoder to deal with the dynamic problem. RNN is no longer appropriate in the Encoder. Figure 2 shows the disadvantage of the Pointer Network.

disadvantage of Pointer Network.png

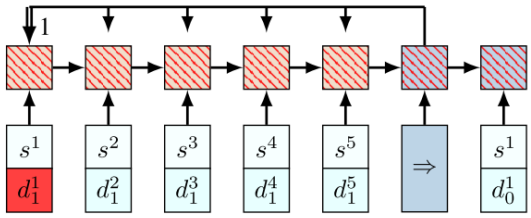


Figure 2: The disadvantage of Pointer Network

The innovation in this article is to replace RNN with an embedding layer in the encoder-mapping inputs to the D-dimensional vector space. Decoder and Attention is similar to the part of pointer network. Then Training the network with Policy Gradient algorithm and the trained model generating a series of continuous actions in real time without re-train for repeated training of each new problem..

There is an simple example about the framework of this article in Figure 3. It shows that the attention layer is based on the embedding layer. The only one connection between Encoder and Decoder is the attention layer and without considering the hidden state.

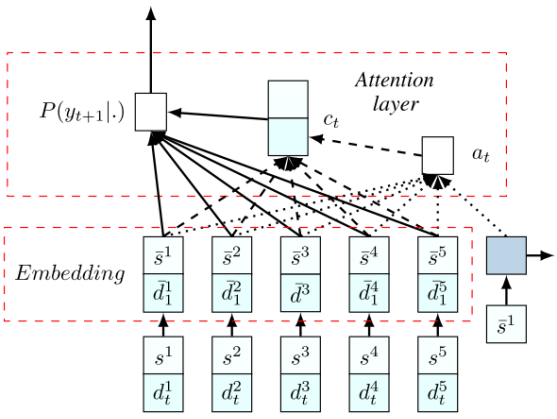


Figure 3: RL and Pointer Network

Motivated by paper [4] and paper [5], paper [6] drops REINFORCE to train the networks to solve COP. This article is also an Encoder-Decoder framework which contains Graph Attention Encoder and attention model Decoder.

2. CONCLUSION

To sum up, the main idea of TSP and VRP is Encoder-Decoder architecture, mapping sequence to sequence. There are many methods for Encoder, such as graph embedding, attention mechanism, glimpse, multi-head, and also Decoder. But the robustness of a trained model has yet to be tested. There are also some works solving COP based on graph theory, like paper [**?**, 4].

REFERENCES

[1] Vinyals O, Fortunato M, Jaitly N. Pointer networks[C]//Advances in Neural Information Processing Systems. 2015: 2692-2700.
[2] Bello I, Pham H, Le Q V, et al. Neural combinatorial optimization with reinforcement learning[J]. arXiv preprint arXiv:1611.09940, 2016.
[3] Nazari M, Oroojlooy A, Snyder L, et al. Reinforcement learning for solving the vehicle routing problem[C]//Advances in Neural Information Processing Systems. 2018: 9839-9849.
[4] Khalil E, Dai H, Zhang Y, et al. Learning combinatorial optimization algorithms over graphs[C]//Advances in Neural Information Processing Systems. 2017: 6348-6358.
[5] Veličković P, Cucurull G, Casanova A, et al. Graph attention networks[J]. arXiv preprint arXiv:1710.10903, 2017.
[6] Kool W, van Hoof H, Welling M. Attention, Learn to Solve Routing Problems![J]. arXiv preprint arXiv:1803.08475, 2018.