On Learning Paradigms for the Travelling Salesman Problem

Chaitanya K. Joshi[1], Thomas Laurent[2], and Xavier Bresson[1]

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

[1] Nanyang Technological University, Singapore
[2] Loyola Marymount University, LA, USA

Correspondence: chaitanya.joshi@ntu.edu.sg

Paper + Code

Contributions

End-to-end learning for TSP
We unify recent ideas in learning for Combinatorial Problems through a generic end-to-end pipeline.

Zero-shot generalization
We design controlled experiments to study the impact of architecture and learning paradigms on solving TSP instances larger than those seen in training (when training on fixed sizes).

Studying the end-to-end pipeline for TSP

1

Graph Definition

Define TSP as a decision problem:
- Full graphs
- *k*-NN graphs

2

Graph Embedding

Obtain embeddings of nodes and edges:
- GCN/GAT
- Transformer
- Simple MLP

3

Solution Decoding

Assign probability to each node or edge:
- Autoregressive
- Non-autoregressive

4

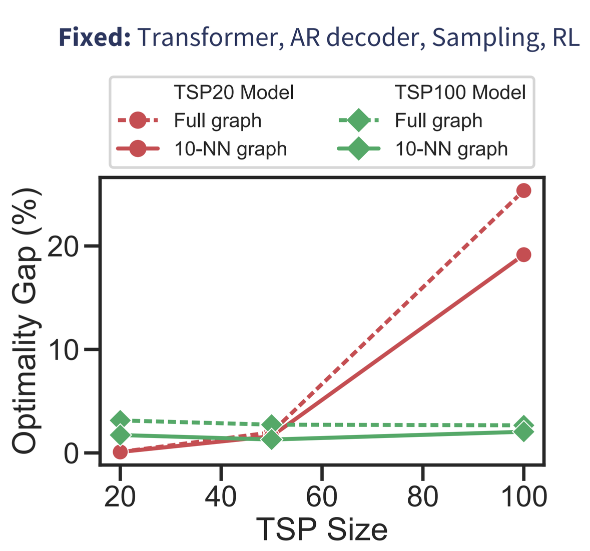
Graph Search

Select feasible solution set:
- Greedy search
- Beam search
- Sampling

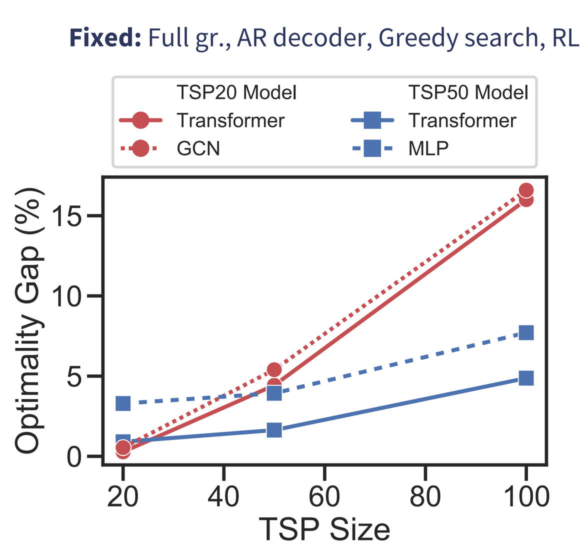
5

Policy Learning

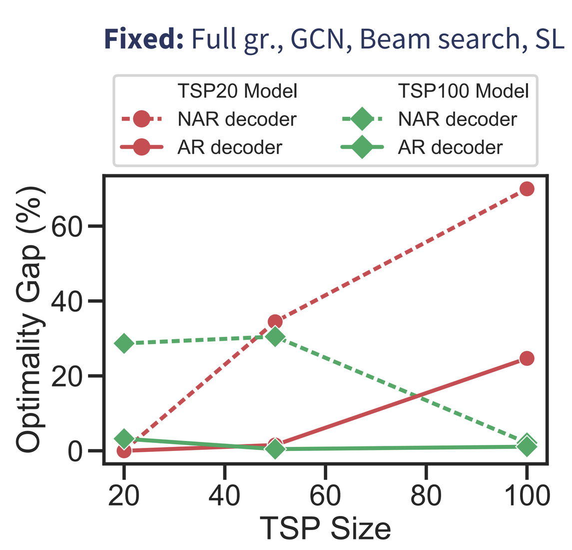
Train prediction policy end-to-end:
- Supervised Learning
- Reinf. Learning



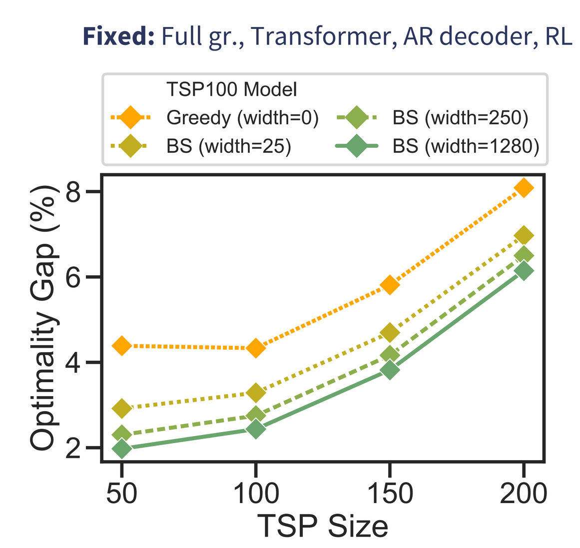
- Generalization, esp. to larger sizes, is difficult.
- **Sparse graphs** lead to faster learning.



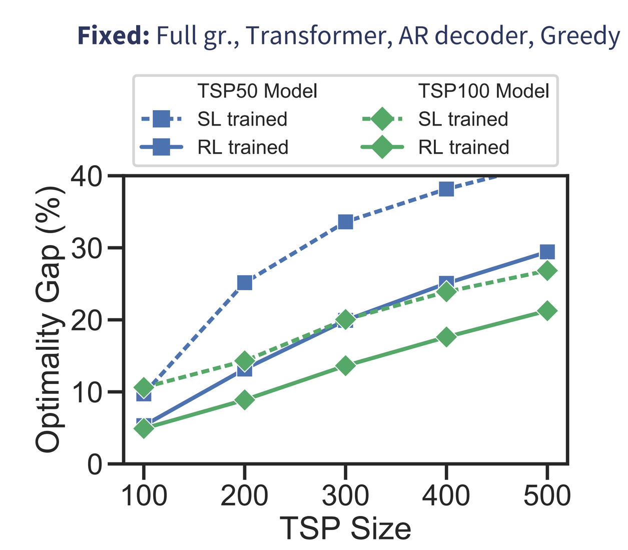
- Modern GNNs do not have **implicit scale-invariance**, *i.e.*, cannot generalize zero-shot.



- **Inductive bias:** AR decoding generalizes better than NAR.
- Step-by-step AR is slower than NAR.



- Search/sampling is a **trade-off** between performance and inference time.

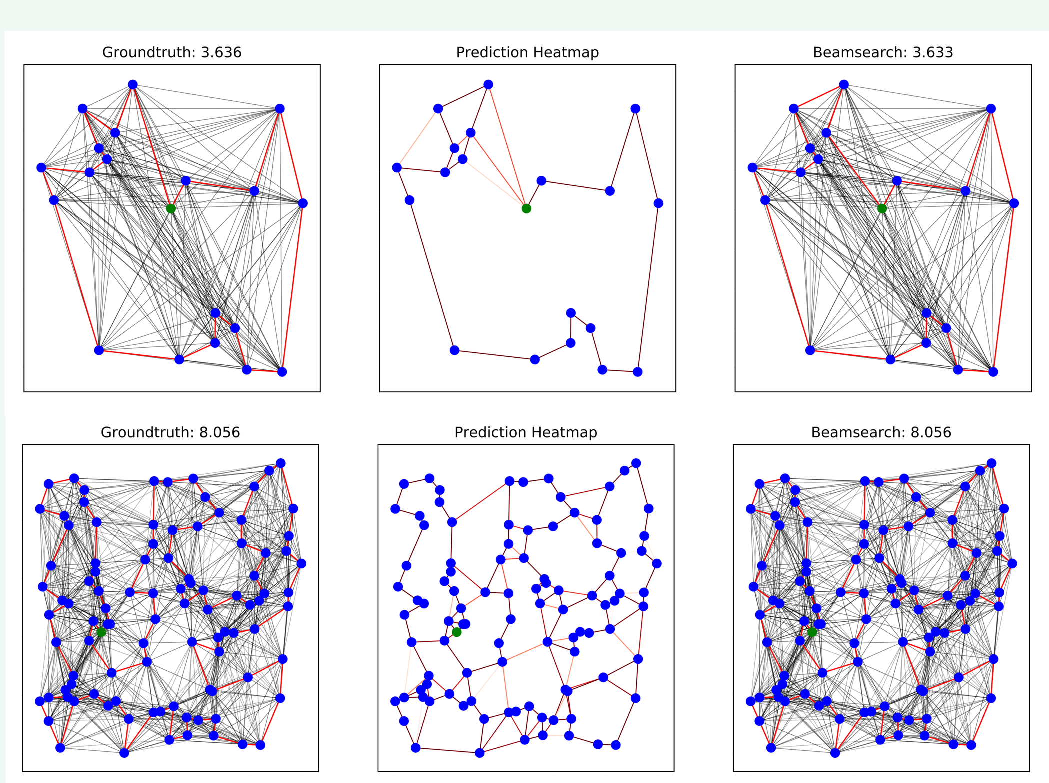


- SL overfits to specific graph sizes.
- Comparatively, RL **generalizes** better.

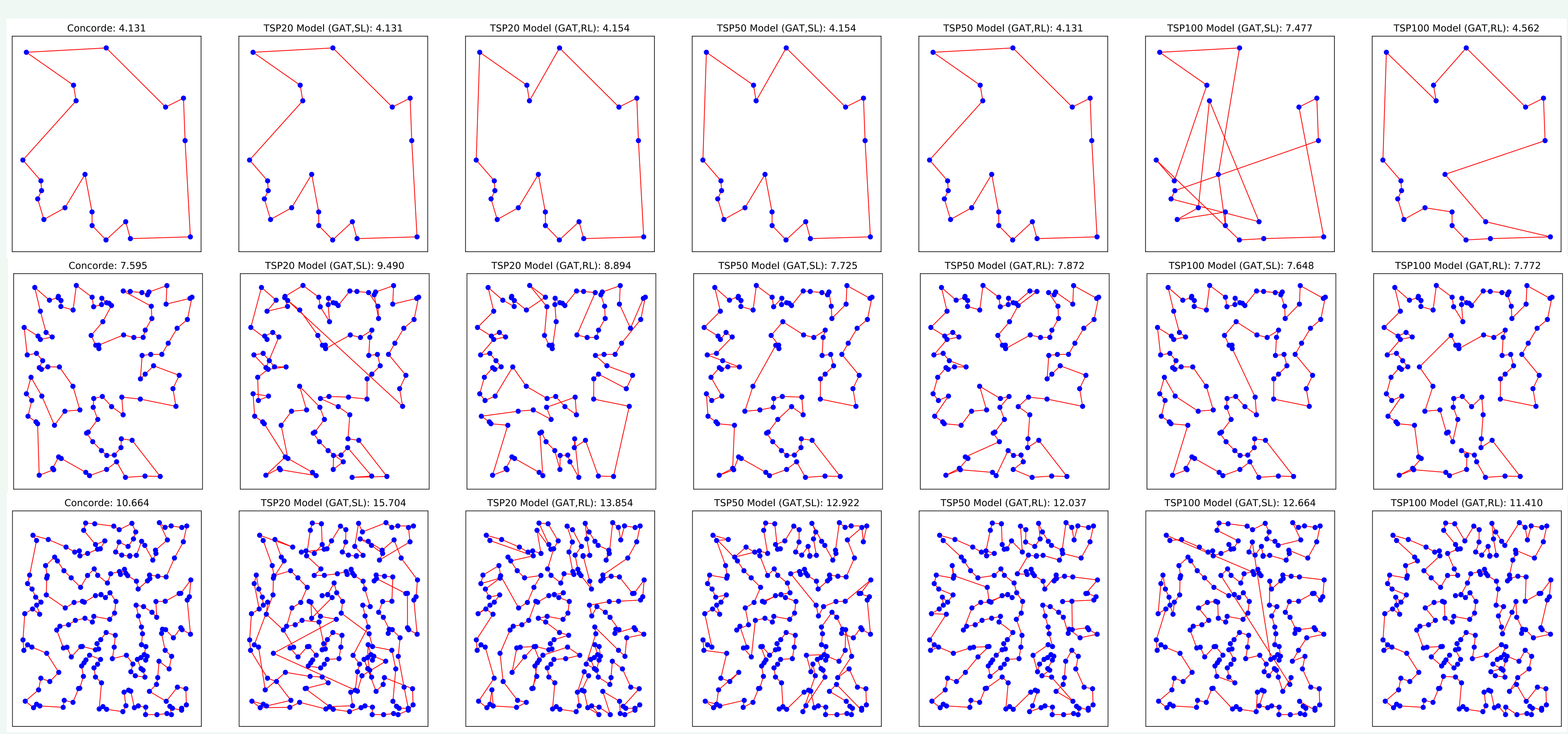
GNN-based solvers generalize poorly to TSPs larger than training size.

Prediction heatmaps

Visualizing probability to be on TSP tour for each edge before performing graph search:



Visualizing poor generalization



What is needed for zero-shot generalization?

Related Work

1. Vinyals, Fortunato, Jaitly. Pointer networks. NeurIPS, 2015.
2. Bello, Pham, Le, Norouzi, Bengio. Neural combinatorial optimization with RL. ICLR, 2017.
3. Nowak, Villar, Bandeira, Bruna. A note on learning algorithms for quadratic assignment. arXiv, 2017.
4. Khalil, Dai, Zhang, Dilkina, Song. Learning combinatorial optimization algorithms over graphs. NeurIPS, 2017.
5. Deudon, Cournut, Lacoste, Adulyaksak, Rousseau. Learning heuristics for the TSP by policy gradient. CPAIOR, 2018
6. Kool, van Hoof, Welling. Attention, learn to solve routing problems! ICLR, 2019.

Acknowledgement: XB is supported in part by NRF Fellowship NRFF2017-10.