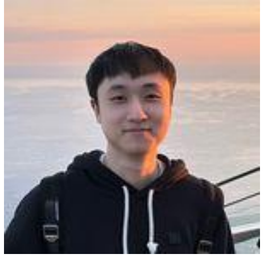
THE UNIVERSITY of EDINBURGH

August 3-7, 2025 KDD2025

TSPRank: Bridging Pairwise and Listwise Methods with a Bilinear Travelling Salesman Model





Waylon Li[1]   Yftah Ziser[2]   Yifei Xie[1]   Shay Cohen[1]   Tiejun Ma[1]

[1]University of Edinburgh   [2]Nvidia Research

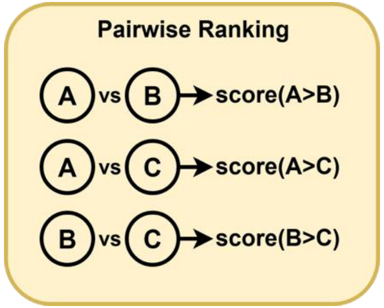
aiai Artificial Intelligence and its Applications Institute   ilcc Institute for Language, Cognition and Computation   NVIDIA

Influencing the world since 1583

Table of Content

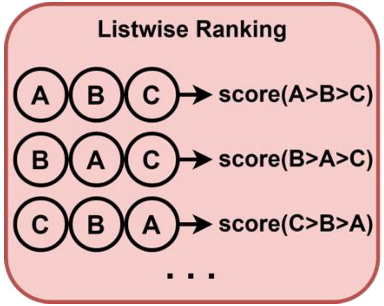
- Motivation
  - Limitations of pairwise and listwise methods
  - Similarity between ranking and travelling salesman problem (TSP)
- Methodology
- Experiments & Results
- Conclusion & Future Work

Limitations of pairwise and listwise methods



✅ Robust (usually GBDT-based)
❌ Not optimized on list level, leading to sub-optimal results [2]

Typical representative: LambdaMART [1]



✅ Capture the list-level information, optimized for listwise order
❌ Less robust and require complex tuning to achieve marginal gains over pairwise models like LambdaMART on information retrieval benchmarks [3]

Typical representative: deep learning based (SetRank [4], Rankformer [5])

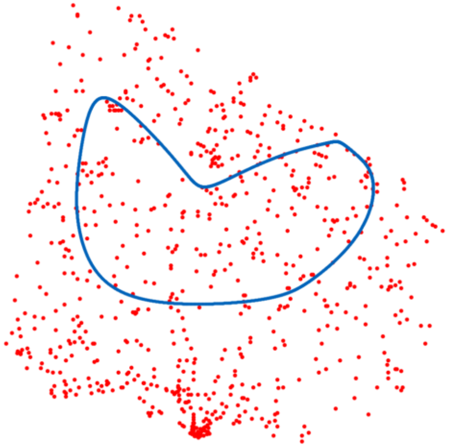
Question: Is it possible to combine the advantages of both pairwise and listwise methods?

Predicting the order of a list is challenging because ranking N entities from 1 to N is complex. However, breaking it down into $(N \times N)$ pairwise comparisons simplifies the task, as each pairwise comparison is more straightforward than ranking the entire list.

- Motivation
- Methodology
  - Travelling salesman problem
  - Rethink pairwise ranking in a graph
  - TSPRank
  - Local learning & global learning
- Experiments & Results
- Conclusion & Future Work

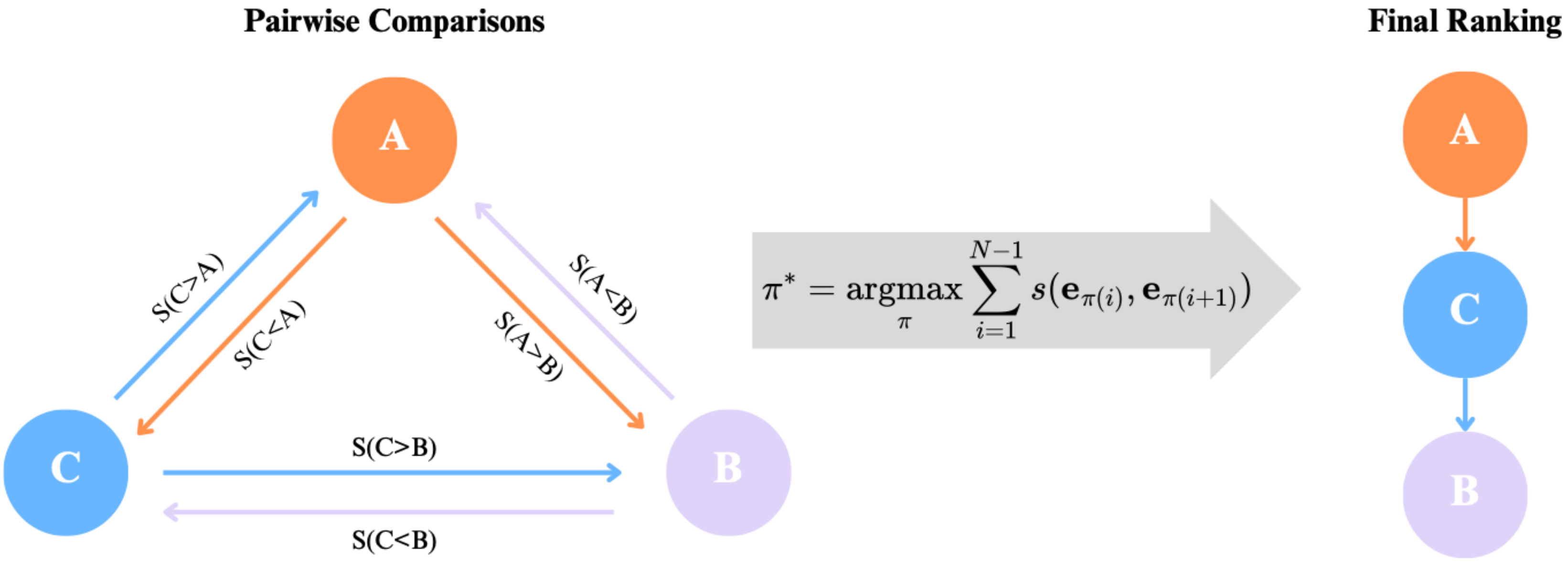
Travelling salesman problem (TSP)

Given a list of cities and the distances between each pair of cities, what is the shortest possible route that visits each city exactly once and returns to the origin city? It is an NP-hard problem in combinatorial optimization, important in theoretical computer science and operations research.

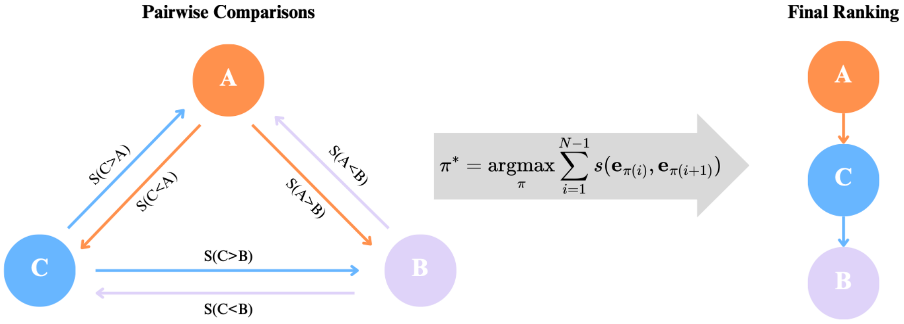


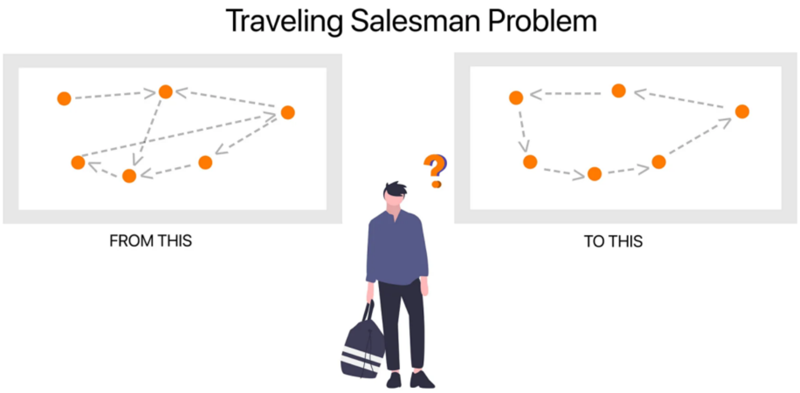
Adapted from https://www.linkedin.com/pulse/traveling-salesman-problem-14-different-solutions-sandeep-kella/

Rethink pairwise ranking in a graph...



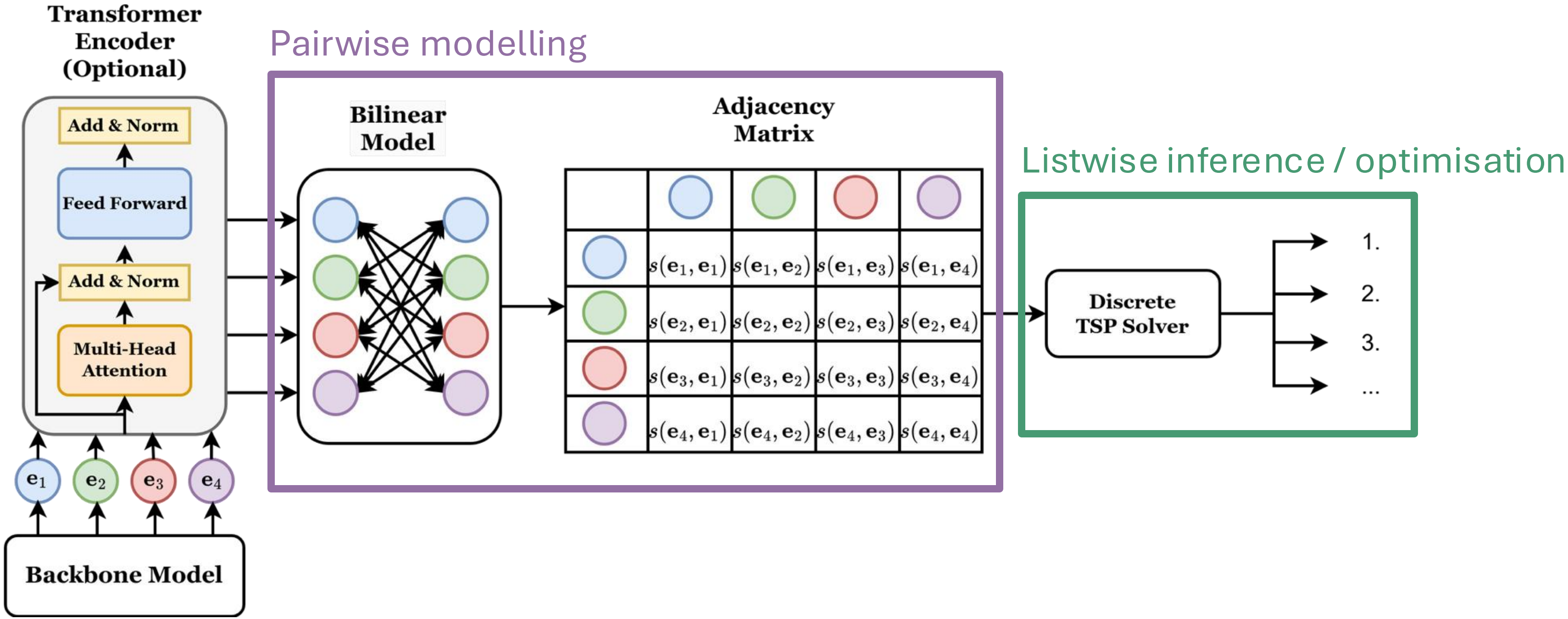
Does this look familiar?





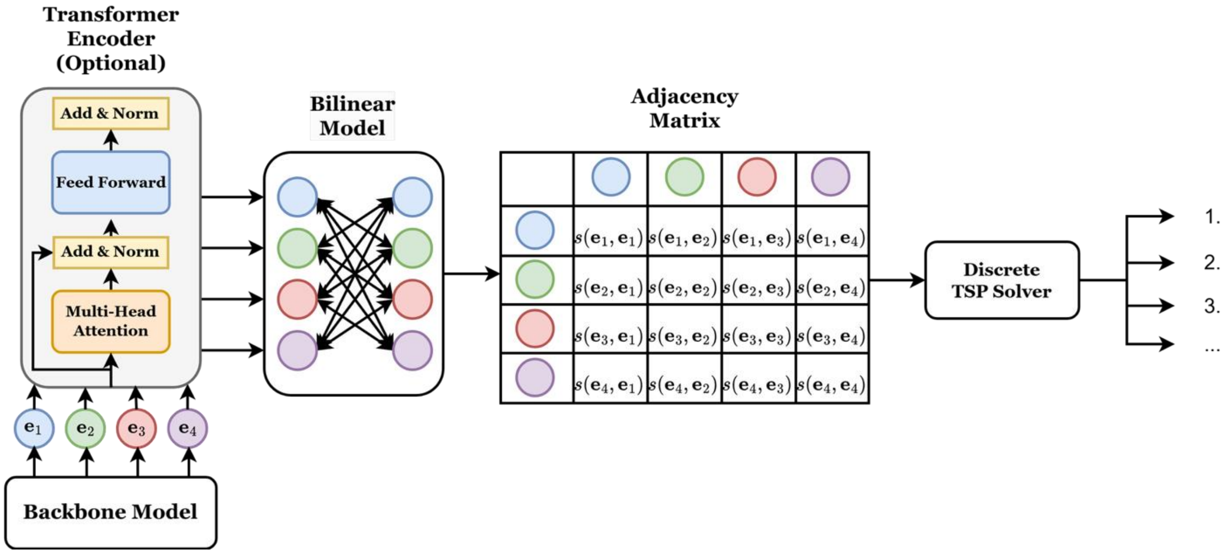
We consider ranking as a TSP where the traveller does not go back to the start point at the end. It is also referred as the Open-Loop TSP.

TSPRank: A generic ranking model for existing backbone encoders

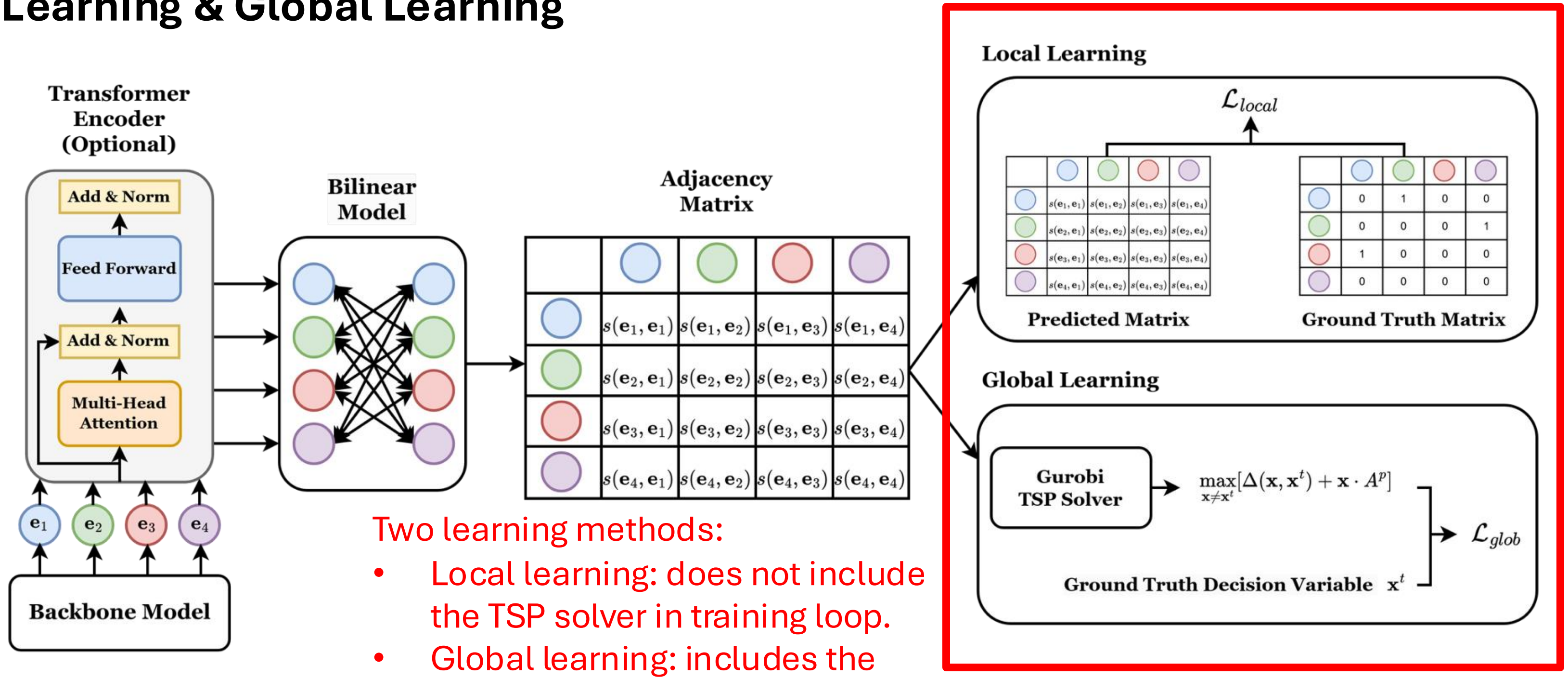


Remaining Question

As the TSP solver is discrete, it does not produce gradients for backpropagation.



Local Learning & Global Learning



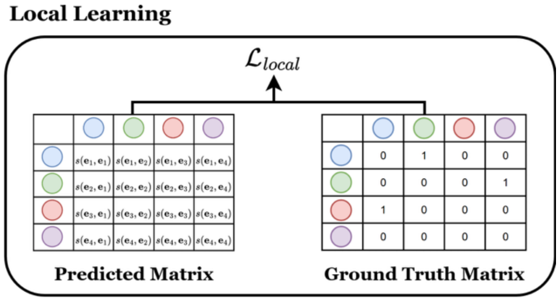
Two learning methods:

- Local learning: does not include the TSP solver in training loop.
- Global learning: includes the solver for end-to-end training.

Local Learning (weighted cross-entropy)

Objective: determine if entity e_j should be ranked one position after e_i in a given pair of entities.

$$\mathcal{L}_{local}(A^p, A^t) = -\sum_{i=1}^{N} y_k \log \frac{e^{A^p_{ik}}}{\sum_{j=1}^{N} e^{A^p_{ij}}}, \; k = \arg\max_j A^t_{ij}$$



- A^P: predicted pairwise scores matrix (adjacency matrix).
- A^t : ground-truth adjacency matrix.
- y_k : weighted term, true ordinal ranking for the true consecutive entity after entity i. (penalties vary based on the actual ranking positions)
- N : number of ranking entities in the list.

Note: y_k can be adjusted to $N + 1 - y_k$ depending on whether y_k represents ascending or descending order.

Global Learning (end-to-end, max-margin)

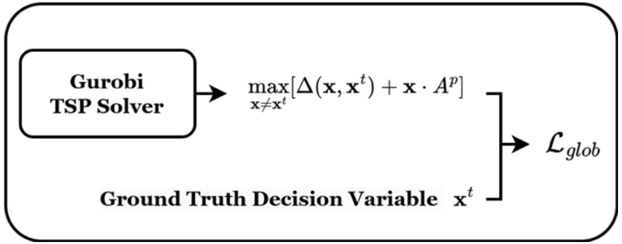
Objective: incorporating the TSP solver in the training procedure to better align the model with the inference process.

$$\mathbf{x}^t \cdot A^p \geq \mathbf{x} \cdot A^p + \Delta(\mathbf{x}^t, \mathbf{x}), \text{ for all } \mathbf{x},$$

$$\mathcal{L}_{glob}(A^p, \mathbf{x}^t) = \max(0, \max_{\mathbf{x} \neq \mathbf{x}^t}[\Delta(\mathbf{x}, \mathbf{x}^t) + \mathbf{x} \cdot A^p] - \mathbf{x}^t \cdot A^p)$$

$$\Delta(\mathbf{x}, \mathbf{x}^t) = \sum_{i=1}^{N} \sum_{j=1}^{N} \max(0, x_{ij} - x_{ij}^t)$$

Global Learning



- A^P: predicted pairwise scores matrix (adjacency matrix).
- x^t : target decision variables.
- x : predicted decision variables.
- Δ : enforce a margin for each incorrectly identified edge.

$$x_{ij} = \begin{cases} 1, & \text{if entity } \mathbf{e}_j \text{ is ranked immediately after } \mathbf{e}_i, \\ 0, & \text{otherwise.} \end{cases}$$

Local Learning vs. Global Learning

Local learning:

- Weighted cross entropy
- Greedily modelling $P(e_j \mid e_i)$

Global learning:

- Max-margin
- End-to-end. Use the output from the discrete TSP solver to guide the training procedure

- Background
- Motivation
- Methodology
- Experiments & Results
  - Datasets and benchmark models
  - Results
  - Visualisation analysis
- Conclusion & Future Work

Dataset	Task
Stock Ranking: introduced by Feng et al. [7], which includes historical trading data from 2013 to 2017 for NASDAQ and NYSE.	Rank next day stocks in the same sector and choose the top-K to invest.
Information Retrieval: MQ2008-list [8] from Microsoft.	Rank a list of documents based on their technical indicators.
Event Ordering: “On This Day 2” (OTD2) [9]	Event Ordering: chronologically ordering historical events given their text embeddings.

Benchmark Models

We choose the SOTA generic pairwise and listwise algorithms (not specifically tailored for any task).

LambdaMART [1] (pairwise, GBDT-based)

Rankformer [5] (listwise, transformer-based)

Metrics

Financial metrics
- IRR@K: investment return ratio of investing the top K stocks.
- SR@K: sharpe ratio of investing the top K stocks.

Ranking metrics
- MAP@K: mean average precision at K.
- Kendall's tau (τ): a statistical measure evaluating the correlation between two ordinal rankings.
- MRR: mean reciprocal rank of the true top entity.
- NDCG@K: normalized discounted cumulative gain at K, measuring ranking quality.

Other metrics
- RMSE: root mean squared error.
- EM: exact match rate.

Results: Stock Ranking

Market	Model	τ	IRR@1	SR@1	MAP@1	IRR@3	SR@3	MAP@3	IRR@5	SR@5	MAP@5
NASDAQ	Feng et al. + MLP (Original)	0.0093	0.1947	0.5341	**0.1690**	0.2366	0.9881	0.3253	0.1892	0.9682	0.5871
	Feng et al. + LambdaMART	0.0071	0.0310	-0.0873	0.1539	0.0340	0.0445	0.3144	0.0505	0.2678	0.5858
	Feng et al. + Rankformer	0.0110	0.2257	0.5464	0.1620	0.2857	1.1245	0.3216	0.2309	1.0943	0.5860
	Feng et al. + TSPRank-Local	0.0291	0.5353	1.2858	0.1658	0.4416	1.7401	0.3297	0.2537	1.2623	0.5932
	Feng et al. + TSPRank-Global	**0.0447**	**0.7849**	**1.7471**	0.1633	**0.5224**	**2.0359**	**0.3364**	**0.2937**	**1.4331**	**0.5999**
NYSE	Feng et al. + MLP (Original)	0.0162	0.4170	1.0755	**0.1791**	0.2574	1.2367	**0.2841**	0.2257	1.3186	0.4649
	Feng et al. + LambdaMART	0.0054	0.1005	0.1367	0.1307	0.0732	0.4192	0.2592	0.1063	0.6882	0.4574
	Feng et al. + Rankformer	0.0181	0.2924	0.9113	0.1535	0.2701	1.2890	0.2758	0.2515	1.4200	0.4651
	Feng et al. + TSPRank-Local	0.0313	**0.5012**	**1.5710**	0.1424	**0.3974**	1.9735	0.2756	0.2788	1.6662	0.4680
	Feng et al. + TSPRank-Global	**0.0422**	0.4787	1.4552	0.1392	0.3889	**1.9976**	0.2756	**0.2816**	**1.7350**	**0.4732**

Table 1: Performance comparison of Feng et al., LambdaMART, Rankformer, and TSPRank on the NASDAQ and NYSE stock ranking dataset, averaged across all filtered sectors.

Results: Information Retrieval & Historical Events Ordering

		Top 10					Top 30				
Model	Type	NDCG@3	NDCG@5	NDCG@10	MRR	τ	NDCG@3	NDCG@5	NDCG@10	MRR	τ
LambdaMART	Pairwise	0.6833	0.7222	0.8707	0.4259	0.1474	0.7340	0.7298	0.7403	0.3617	0.2372
Rankformer	Listwise	0.7220	0.7565	0.8865	0.4661	**0.2317**	0.7486	0.7470	0.7596	0.3732	**0.2834**
TSPRank-Local	Pairwise-Listwise	0.6858	0.7213	0.8719	0.4266	0.1544	0.7189	0.7240	0.7362	0.3206	0.2054
TSPRank-Global	Pairwise-Listwise	**0.7281**	**0.7585**	**0.8884**	**0.4861**	0.2212	**0.7582**	**0.7558**	**0.7631**	**0.3895**	0.2647

Table 2: Evaluation of LambdaMART, Rankformer, and TSPRank on MQ2008-list information retrieval dataset for top 10 and top 30 documents.

Group Size		10				30				50			
Model	Type	$\tau \uparrow$	EM ↑	MRR ↑	RMSE ↓	$\tau \uparrow$	EM ↑	MRR ↑	RMSE ↓	$\tau \uparrow$	EM ↑	MRR ↑	RMSE ↓
te-3-small + LambdaMART	Pairwise	0.6297	0.3008	0.7554	**1.993**	0.5929	0.1064	0.6122	5.969	0.6000	0.0639	0.5596	9.618
te-3-small + Rankformer	Listwise	0.6190	0.2899	0.7361	1.998	0.5859	0.0921	0.4911	5.973	0.5724	0.0527	0.3526	10.069
te-3-small + TSPRank-Local	Pairwise-Listwise	0.5658	0.2856	0.7679	2.296	0.5095	0.0873	0.5739	6.930	0.4713	0.0460	0.3949	12.084
te-3-small + TSPRank-Global	Pairwise-Listwise	**0.6301**	**0.3350**	**0.7936**	2.057	**0.6302**	**0.1384**	**0.7300**	**5.770**	**0.6207**	**0.0871**	**0.6618**	**9.602**

Table 3: Evaluation of LambdaMART, Rankformer, and TSPRank on OTD2 dataset for historical events ordering for group sizes of 10, 30, and 50. "*te-3-small*" stands for "*text-embedding-3-small*".

Visualisation Analysis

Purpose: empirically explore why TSPRank-Global performs better.

We use the OTD2 dataset as the starting point as textual data is more interpretable.

We arbitrarily sample 3 events each from the US, UK, and China.

Event Title	Year	Rank	Label
1st US store to install electric lights, Philadelphia	1878	3	US-1
1st sitting US President to visit South America, FDR in Colombia	1934	5	US-2
75th US Masters Tournament, Augusta National GC: Charl Schwartzel of South Africa birdies the final 4 holes to win his first major title, 2 strokes ahead of Australian pair Adam Scott and Jason Day	2011	8	US-3
Charles Watson-Wentworth, 2nd Marquess of Rockingham, becomes Prime Minister of Great Britain	1782	2	UK-1
1st main line electric train in UK (Liverpool to Southport)	1904	4	UK-2
UK Terrorism Act 2006 becomes law	2006	7	UK-3
A Mongolian victory at the naval Battle of Yamen ends the Song Dynasty in China	1279	1	CN-1
US Senate rejects China People's Republic membership to UN	1953	6	CN-2
China's Hubei province, the original center of the coronavirus COVID-19 outbreak eases restrictions on travel after a nearly two-month lockdown	2020	9	CN-3

Table 4: Event titles in the constructed group. Labels indicate the order of occurrence within each country, e.g., "US-1" denotes the earliest event in the US within the group.

Event Title	Year	Rank	Label
1st US store to install electric lights, Philadelphia	1878	3	US-1
1st sitting US President to visit South America, FDR in Colombia	1934	5	US-2
75th US Masters Tournament, Augusta National GC: Charl Schwartzel of South Africa birdies the final 4 holes to win his first major title, 2 strokes ahead of Australian pair Adam Scott and Jason Day	2011	8	US-3
Charles Watson-Wentworth, 2nd Marquess of Rockingham, becomes Prime Minister of Great Britain	1782	2	UK-1
1st main line electric train in UK (Liverpool to Southport)	1904	4	UK-2
UK Terrorism Act 2006 becomes law	2006	7	UK-3
A Mongolian victory at the naval Battle of Yamen ends the Song Dynasty in China	1279	1	CN-1
US Senate rejects China People's Republic membership to UN	1953	6	CN-2
China's Hubei province, the original center of the coronavirus COVID-19 outbreak eases restrictions on travel after a nearly two-month lockdown	2020	9	CN-3

Table 4: Event titles in the constructed group. Labels indicate the order of occurrence within each country, e.g., "US-1" denotes the earliest event in the US within the group.

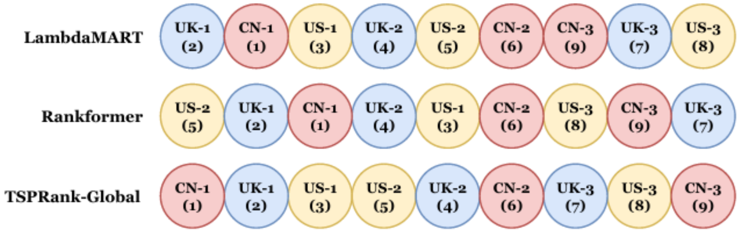


Figure 2: Visualisation of predictions by LambdaMART, Rankformer, and TSPRank-Global on the constructed group. Numbers in parentheses indicate the true ranking.

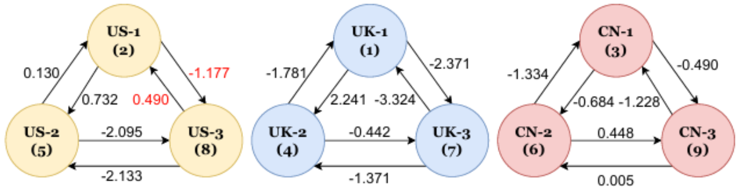


Figure 3: Illustration of the intra-country pairwise comparison graph. Edges between pairs of events from different countries are omitted for clarity. Scores highlighted in red indicate errors in the pairwise prediction for TSPRank-Global.

- Background
- Motivation
- Methodology
- Experiments & Results
- Conclusion & Future Work

Conclusion: Main Findings

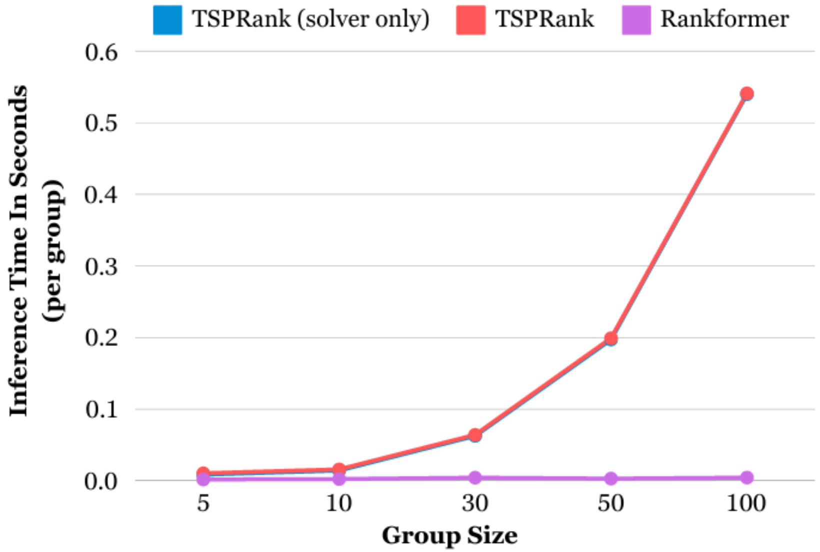
- Better Performance of TSPRank, which is a hybrid method, across diverse tasks.
- Global learning outperforms local learning.
- GBDT-based pairwise ranking method does not always outperform deep learning based listwise ranking method as indicated by existing literatures.
- With the help of the listwise optimisation provided by the TSP solver, TSPRank is more tolerant to errors and uncertainties in pairwise comparisons.

Future Work

99.8% of the inference time is consumed by the discrete TSP solver.

⇒ Currently suitable for small-scale ranking problems such as the reranking stage in information retrieval, etc.

⇒ Future work can be replacing the Gurobi TSP solver by other heuristic algorithms or NN-based TSP solvers.



[1] Burges, C.J., 2010. From ranknet to lambdarank to lambdamart: An overview. *Learning*, *11*(23-581), p.81.
[2] Cao, Z., Qin, T., Liu, T.Y., Tsai, M.F. and Li, H., 2007, June. Learning to rank: from pairwise approach to listwise approach. In Proceedings of the 24th international conference on Machine learning (pp. 129-136).
[3] Qin, Z., Yan, L., Zhuang, H., Tay, Y., Pasumarthi, R.K., Wang, X., Bendersky, M. and Najork, M., 2021, May. Are neural rankers still outperformed by gradient boosted decision trees?. In International conference on learning representations.
[4] Pang, L., Xu, J., Ai, Q., Lan, Y., Cheng, X. and Wen, J., 2020, July. Setrank: Learning a permutation-invariant ranking model for information retrieval. In Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval (pp. 499-508).
[5] Buyl, M., Missault, P. and Sondag, P.A., 2023, August. Rankformer: Listwise learning-to-rank using listwide labels. In Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (pp. 3762-3773).
[7] Feng, F., He, X., Wang, X., Luo, C., Liu, Y. and Chua, T.S., 2019. Temporal relational ranking for stock prediction. ACM Transactions on Information Systems (TOIS), 37(2), pp.1-30.
[8] https://www.microsoft.com/en-us/research/project/letor-learning-rank-information-retrieval/letor-4-0/
[9] https://github.com/ltorroba/machine-reading-historical-events

More details...

Poster:

#195 Exhibit Hall F, Tuesday, August 5, 5:30 - 8:00 PM

Paper

Code & Data

Email:
waylon.li@ed.ac.uk

Remaining Questions

1. Choose and setup the TSP solver.

- x_{ij}: decision variable.
- s_{ij}: $s(e_i, e_j)$
- N: the total number of entities to be ranked.
- z_i: the number of entities ranked before entity i.

$$x_{ij} = \begin{cases} 1, & \text{if entity } \mathbf{e}_j \text{ is ranked immediately after } \mathbf{e}_i, \\ 0, & \text{otherwise.} \end{cases}$$

Objective function:

$$\max_{x_{ij}} \sum_{i=1}^{N} \sum_{j=1, j \neq i}^{N} s_{ij} x_{ij}$$

Constraints:

$$s.t. \quad \sum_{j=1, j \neq i}^{N} x_{ij} \leq 1 \quad \text{for all } i$$

$$\sum_{i=1, i \neq j}^{N} x_{ij} \leq 1 \quad \text{for all } j$$

$$\sum_{i=1}^{N} \sum_{j=1, j \neq i}^{N} x_{ij} = N - 1$$

$$z_i + 1 \leq z_j + N(1 - x_{ij}) \quad i, j = 2, \ldots, N, \ i \neq j$$

$$z_i \geq 0 \quad i = 2, \ldots, N$$

[6] Gurobi Optimization, LLC. 2023. Gurobi Optimizer Reference Manual. https://www.gurobi.com

THE UNIVERSITY of EDINBURGH

Remaining Questions

1. Choose and setup the TSP solver.

- x_{ij}: decision variable.
- s_{ij}: $s(e_i, e_j)$
- N: the total number of entities tobe ranked.
- z_i: the number of entities ranked before entity i.

$$\max_{x_{ij}} \sum_{i=1}^{N} \sum_{j=1, j\neq i}^{N} s_{ij} x_{ij}$$

Each entity has at most one predecessor and one successor in the ranking

$$s.t. \quad \sum_{j=1, j\neq i}^{N} x_{ij} \leq 1 \quad \text{for all } i$$

$$\sum_{i=1, i\neq j}^{N} x_{ij} \leq 1 \quad \text{for all } j$$

$$\sum_{i=1}^{N} \sum_{j=1, j\neq i}^{N} x_{ij} = N - 1$$

$$z_i + 1 \leq z_j + N(1 - x_{ij}) \quad i, j = 2, \ldots, N, \ i \neq j$$

$$z_i \geq 0 \quad i = 2, \ldots, N$$

[6] Gurobi Optimization, LLC. 2023. Gurobi Optimizer Reference Manual. https: //www.gurobi.com

Remaining Questions

1. Choose and setup the TSP solver.

- x_{ij}: decision variable.
- s_{ij}: $s(e_i, e_j)$
- N: the total number of entities tobe ranked.
- z_i: the number of entities ranked before entity i.

$$\max_{x_{ij}} \sum_{i=1}^{N} \sum_{j=1, j \neq i}^{N} s_{ij} x_{ij}$$

$$s.t. \quad \sum_{j=1, j \neq i}^{N} x_{ij} \leq 1 \quad \text{for all } i$$

$$\sum_{i=1, i \neq j}^{N} x_{ij} \leq 1 \quad \text{for all } j$$

Ensures that the total number of pairwise comparisons is exactly $N-1$ (open-loop).

$$\sum_{i=1}^{N} \sum_{j=1, j \neq i}^{N} x_{ij} = N - 1$$

$$z_i + 1 \leq z_j + N(1 - x_{ij}) \quad i, j = 2, \ldots, N, \ i \neq j$$

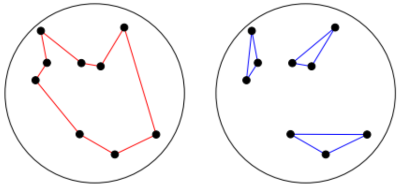
$$z_i \geq 0 \quad i = 2, \ldots, N$$

[6] Gurobi Optimization, LLC. 2023. Gurobi Optimizer Reference Manual. https://www.gurobi.com

Remaining Questions

1. Choose and setup the TSP solver.

- x_{ij}: decision variable.
- s_{ij}: $s(e_i, e_j)$
- N: the total number of entities tobe ranked.
- z_i: the number of entities ranked before entity i.



Introduce variables z to eliminate multiple separate sequences (subtours) and enforce that there is a single, complete ranking that includes all entities.

$$\max_{x_{ij}} \sum_{i=1}^{N} \sum_{j=1, j \neq i}^{N} s_{ij} x_{ij}$$

$$s.t. \quad \sum_{j=1, j \neq i}^{N} x_{ij} \leq 1 \quad \text{for all } i$$

$$\sum_{i=1, i \neq j}^{N} x_{ij} \leq 1 \quad \text{for all } j$$

$$\sum_{i=1}^{N} \sum_{j=1, j \neq i}^{N} x_{ij} = N - 1$$

$$z_i + 1 \leq z_j + N(1 - x_{ij}) \quad i, j = 2, \ldots, N, \ i \neq j$$

$$z_i \geq 0 \quad i = 2, \ldots, N$$

[6] Gurobi Optimization, LLC. 2023. Gurobi Optimizer Reference Manual. https: //www.gurobi.com