Scalable and Multi-modal Neural Graph Retrieval

(Research Statement)

Indradyumna Roy

My current research plan is strongly motivated by the goal of building a unified framework for graph retrieval — which, given a query graph, returns a top-K list of relevant graphs or subgraphs. Graphs provide a common formalism across natural language queries, parse trees, Knowledge Graphs (KG), graphs extracted from tables and charts, images (scene graphs), *etc*., which allows data from different modalities to inter-operate seamlessly during retrieval. To align and connect across graphs from diverse modalities, we need node and edge representations that transcend traditional schema unification. Neural graph models provide a natural alternative. I therefore work primarily within a neural representation and retrieval framework.

Problem Setup and Challenges:

The corpus graph(s) are sourced from diverse information sources, and are linked together as needed in a preprocessing step. Examples of this step may be named entity detection in text and tables, linking mention spans to canonical entities in a KG such as Wikidata [46]; or connecting an object ('puppy') in an image to generic types ('pet', 'animal') in a KG [2, 9, 20].

Subsequent retrieval belongs to two distinct paradigms. The first involves searching for subgraphs within a single, large, real-world knowledge graph, such as Wikidata, DBpedia, or Freebase [3, 12, 46]. This requires identifying, scoring, and ranking relevant subgraphs from the larger corpus graph in response to a given query. The second paradigm involves a collection of smaller corpus graphs, typically resulting from scene graph extractions from images [22, 24, 30], local extractions from text passages and tables, molecular graphs [34], and others. Here, the retrieval task involves ranking smaller corpus graphs based on a given query.

In either paradigm, the goal is to report a ranked list of subgraphs, in response to a given query which may be provided as a natural language query utterance, a query image, or a query graph. An effective neural graph retrieval model should aim to achieve (i) *high retrieval accuracy*, ensuring that subgraphs are ranked according to their relevance to the query; (ii) *scalability*, enabling the efficient processing of large-sized graphs or a large number of graphs; and (iii) *interpretability*, allowing the model to justify its responses through alignment-based explanations.

Against this backdrop, my current research explores the following directions:

1. **Designing and learning relevance models for graph retrieval:** Designing neural models for graph retrieval and alignment, subgraph matching, designing neural analogs to combinatorial graph optimizations.
2. **LSH compatibility of graph search models:** Designing symmetric and asymmetric LSH techniques compatible with graph similarity scoring functions to enable sublinear time retrieval.

1 Designing and Learning Relevance Models for Graph Retrieval

As with other information retrieval tasks, defining relevance in graph search applications can be complex and nuanced. Conventional methods to measure relevance rely on combinatorial approaches like graph edit distance (GED), maximum common subgraph counting (MCS), and various graph or subgraph matching techniques [6, 28]. There has also been a growing interest in graph matching methods to assess similarity between image pairs [10, 16, 26, 42, 48]. However, these existing methods usually require a known ground truth correspondence between nodes and edges, which can be expensive and time-consuming to produce.

I believe that neural retrieval models should be trainable under distant supervision using pairwise relevance judgements, without relying on any prior information about ground truth correspondence between nodes or edges. Therefore, any neural graph retrieval model should be able to capture the underlying latent relevance between and query and corpus graphs In this line of research, my earlier works [39, 40] introduced neural architectures aligned with combinatorial concepts for identifying subgraph isomorphism [40] and maximum common subgraph [39]. Despite their complex alignment tasks, these architectures are fully differentiable

and support backpropagation, enabling end-to-end training. This is made possible through the use of the Gumbel-Sinkhorn network [33], which provides a soft approximation of the underlying node and edge alignment permutation as a doubly stochastic matrix This approximation allows us to create an alignment-driven asymmetric scoring layer, tailored to each specific combinatorial matching task. Furthermore, during inference, our neural graph retrieval systems are able to justify relevance scores with approximate *alignment witnesses*. This renders our model interpretable, as they pinpoint the underlying reason behind a given relevance score, e.g., which substructure of the corpus graph is playing the key role in relevance.

Future work: In this line of research, I am eager to continue working on three key areas:

• *Unifying different types of graph scoring layers:* Instead of designing custom scoring layers for each combinatorial notion, as I did in previous works [39, 40], I am currently focusing on building a unified neural model capable of accommodating a broad spectrum of late interaction scoring layers. A promising approach involves modeling Graph Edit Distance (GED), because the variable costs associated with adding or deleting nodes and edges allow GED to represent both symmetric and asymmetric relationships between graph pairs. This flexibility makes GED suitable for various graph comparison tasks, such as identifying the Maximum Common Subgraph and verifying Subgraph Isomorphism [13]. Although recent research [6, 7, 19, 36, 51] has utilized Graph Neural Networks (GNNs) to create neural models for GED computation, these approaches often overlook the potential for edit operations with different costs, leading to an overemphasis on cost-invariant edit sequences. My current work aims to address this limitation by incorporating GED with diverse edit costs, providing a more accurate and flexible framework for graph comparison.

• *Proposing node-edge consistent alignments:* Our current differentiable alignment module frequently generates inconsistent alignments for node and edge. This inconsistency is somewhat expected when trying to approximate the inherently complex Quadratic Assignment Problem (QAP) with polynomial-time neural network solvers. However, there is an opportunity to reduce the optimality gap, especially when considering the distributional characteristics of any given dataset. To improve alignment consistency, I plan to incorporate ideas from the Knowledge Graph (KG) alignment community [43, 47, 49, 50], where there is an emphasis on the dual role of node and edge alignments. In this approach, aligning nodes can provide additional information for edge alignment, and vice versa. Preliminary experiments focusing on improving edge alignment consistency, using the Kronecker product of the node alignments, have shown a significant improvement in the quality of the proposed matching. This approach offers a promising path toward develop more reliable and unified scoring mechanisms for neural graph models.

• *Improved interpretability via discrete structure matching:* While the Gumbel-Sinkhorn network [33] offers effective soft neural alignments for optimizing subjective human relevance judgments, proposing interpretable justifications for relevance scoring in richly attributed graphs requires distinct alignment explanations which account for both graph structure and node/edge labels. This necessitates consideration of both discrete structural alignment between graph pairs as well as similarity across dense features. Consequently, developing neural models for combinatorial graph alignment becomes crucial, where neural networks must make discrete decisions about alignment for nodes and edges in graph pairs. A current area of research explores designing optimization processes to train network architectures that make discrete decisions [8, 11, 35]. During my earlier work on "circuit neural networks" for logic synthesis and optimization at Google DeepMind, I faced similar challenges in discrete decision-making. I plan to work on designing more effective neural optimization routines to address these challenges.

2 LSH Compatibility of Graph Search Models

Ideally, a graph search system needs to rank all the corpus graphs by similarity scores for a given query graph. However, computing these scores can be prohibitively expensive when dealing with large graph databases. This bottleneck can be addressed through graph indexing, followed by locality-sensitive hashing (LSH). Yet, most graph matching models [19, 27], including those from our previous works [39, 40], use cross-graph neural alignment networks, which leave the resultant graph embeddings *dependent* on query-corpus pair.

In my recent work FourierHashNet [41], we approached a simpler scenario specific to subgraph isomorphism-based relevance [29]. In this context, the key task is to check whether the query graph is contained within a corpus graph. This is done by encoding each graph into fixed-size vectors and then checking for element-wise vector dominance. This notion of containment check using order embeddings [32], is widely used across various domains, including images, graphs [29], and text [25, 44]. However, conventional LSH typically caters to symmetric or straightforward asymmetric distance functions, which makes it unsuitable for this type of asymmetric dominance check.

To overcome this limitation, I proposed an asymmetric LSH, FourierHashNet [41], which transforms the dominance distance into a bounded dominance similarity measure, which is then converted into a form amenable to LSH through a Fourier transformation, resulting in an expectation of inner products of functions in the frequency domain. Finally, this expectation is approximated using importance-sampled estimates, thereby enabling the application of traditional LSH but within the frequency domain. This approach, inspired by the seminal work by Rahimi and Recht [37], presents an initial solution to the challenge of indexing graph databases while maintaining relevance for asymmetric similarity measures driven by subgraph isomorphism.

Future work: In this line of research, I am eager to continue working on three key areas:

• *Extension to other scoring functions:* Our proposed asymmetric LSH framework in FourierHashNet [41] can be extended to work with any shift-invariant function, potentially making several useful scoring functions LSH-compatible, such as Box Embedding-based volume scores [15] and facility location scores used in ColBERT [23]. Many such scoring functions, despite their modeling benefits, have often not yet been incorporated into mainstream industrial recommendation systems due to a lack of LSH compatibility. To explore this extension, I plan to create datasets and conduct experiments to test the performance of our FourierHashNet framework in retrieval scenarios that use these exotic scores. Finally, one of my aspirational goals is to design an LSH for the Sinkhorn Distance for Optimal Transport [18, 45], which will make it immediately applicable to graph retrieval applications that rely on Sinkhorn networks [33] for alignment-driven scoring.

• *Benchmarking data-driven LSH performance:* In situations where the data is not isotropic, one can train a data-sensitive LSH to balance the distribution across buckets. My previous work [38, 41] showed that trainable data-sensitive LSH offers a significantly improvement in trade-off between retrieval accuracy and query-time latency compared to traditional data-agnostic Random Hyperplanes LSH. Moreover, in FourierHashNet [41], data-driven LSH outperformed other non-LSH indexing methods like IVF [21] and HNSW [31]. Unfortunately, data-driven LSH is often neglected in practice and is absent from major indexing tools like Falconn [1] and Faiss [21], as well as key ANN benchmarks [4]. To address this gap, I am working on a comprehensive benchmarking evaluation to assess the performance of trained Random Hyperplanes LSH in comparison with other indexing approaches.

• *Broader applicability of asymmetric LSH with exotic collision probabilities.* There is a growing interest in LSH due to emerging applications beyond search. For example, the LGD sampler [14] uses LSH functions that are sensitive to classification loss functions, some of which are non-symmetric. This indicates that new asymmetric LSH approaches could lead to more efficient optimization routines for a variety of problems. Similarly, the efficient kernel-matrix multiplication algorithm by Backurs et al. [5] relies on LSH but currently only supports symmetric kernels. Asymmetric LSH could significantly expand the applicability of this framework. Similarly, an asymmetric LSH designed for linear regression and classification loss has been used to develop differentially private classifiers[17], showing that new asymmetric LSH functions could increase the versatility and utility of such frameworks. I plan to explore additional domains and applications where our asymmetric Fourier hashing framework can prove useful.

Plans

Over the next year, I plan to gather graphs from various modalities (including images, text, and tables), along with other datasets for diverse scoring functions, and set up QA test-beds and benchmarks. I will then proceed with the future work detailed above. We will release code in the public domain as we make progress.

References

[1] Alexandr Andoni, Piotr Indyk, Thijs Laarhoven, Ilya Razenshteyn, and Ludwig Schmidt. Practical and optimal lsh for angular distance. *Advances in neural information processing systems*, 28, 2015.

[2] Ghulam Ahmed Ansari, Amrita Saha, Vishwajeet Kumar, Mohan Bhambhani, Karthik Sankaranarayanan, and Soumen Chakrabarti. Neural program induction for kbqa without gold programs or query annotations. In *IJCAI*, pages 4890–4896. Macao, China, 2019.

[3] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer, 2007.

[4] Martin Aumüller, Erik Bernhardsson, and Alexander Faithfull. Ann-benchmarks: A benchmarking tool for approximate nearest neighbor algorithms. In *International conference on similarity search and applications*, pages 34–49. Springer, 2017.

[5] Arturs Backurs, Piotr Indyk, Cameron Musco, and Tal Wagner. Faster kernel matrix algebra via density estimation. In *International Conference on Machine Learning*, pages 500–510. PMLR, 2021.

[6] Yunsheng Bai, Hao Ding, Song Bian, Ting Chen, Yizhou Sun, and Wei Wang. Simgnn: A neural network approach to fast graph similarity computation. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pages 384–392, 2019.

[7] Yunsheng Bai, Hao Ding, Ken Gu, Yizhou Sun, and Wei Wang. Learning-based efficient graph similarity computation via multi-scale convolutional set matching. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 3219–3226, 2020.

[8] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.

[9] Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1533–1544, 2013.

[10] Florian Bernard, Christian Theobalt, and Michael Moeller. Ds*: Tighter lifting-free convex relaxations for quadratic matching problems. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4310–4319, 2018.

[11] Quentin Berthet, Mathieu Blondel, Olivier Teboul, Marco Cuturi, Jean-Philippe Vert, and Francis Bach. Learning with differentiable pertubed optimizers. *Advances in neural information processing systems*, 33:9508–9519, 2020.

[12] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250, 2008.

[13] Horst Bunke. On a relation between graph edit distance and maximum common subgraph. *Pattern Recognition Letters*, 18(8):689–694, 1997.

[14] Beidi Chen, Yingchen Xu, and Anshumali Shrivastava. Fast and accurate stochastic gradient estimation. *Advances in Neural Information Processing Systems*, 32, 2019.

[15] Tejas Chheda, Purujit Goyal, Trang Tran, Dhruvesh Patel, Michael Boratko, Shib Sankar Dasgupta, and Andrew McCallum. Box embeddings: An open-source library for representation learning using geometric structures. *arXiv preprint arXiv:2109.04997*, 2021. URL https://www.iesl.cs.umass.edu/box-embeddings/main/index.html.

[16] Minsu Cho, Jungmin Lee, and Kyoung Mu Lee. Reweighted random walks for graph matching. In *European conference on Computer vision*, pages 492–505. Springer, 2010.

[17] Benjamin Coleman and Anshumali Shrivastava. A one-pass distributed and private sketch for kernel sums with applications to machine learning at scale. In *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, pages 3252–3265, 2021.

[18] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

[19] Khoa D Doan, Saurav Manchanda, Suchismit Mahapatra, and Chandan K Reddy. Interpretable graph similarity computation via differentiable optimal alignment of node embeddings. 2021.

[20] Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. Allennlp: A deep semantic natural language processing platform. 2017.

[21] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547, 2019.

[22] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3668–3678, 2015.

[23] Omar Khattab and Matei Zaharia. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 39–48, 2020.

[24] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017.

[25] Alice Lai and Julia Hockenmaier. Learning to predict denotational probabilities for modeling entailment. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 721–730, 2017. URL https://www.aclweb.org/anthology/E17-1068.pdf.

[26] Marius Leordeanu and Martial Hebert. A spectral technique for correspondence problems using pairwise constraints. 2005.

[27] Yujia Li, Chenjie Gu, Thomas Dullien, Oriol Vinyals, and Pushmeet Kohli. Graph matching networks for learning the similarity of graph structured objects. In *International conference on machine learning*, pages 3835–3845. PMLR, 2019.

[28] Yujia Li, Chenjie Gu, Thomas Dullien, Oriol Vinyals, and Pushmeet Kohli. Graph matching networks for learning the similarity of graph structured objects, 2019.

[29] Zhaoyu Lou, Jiaxuan You, Chengtao Wen, Arquimedes Canedo, Jure Leskovec, et al. Neural subgraph matching. *arXiv preprint arXiv:2007.03092*, 2020.

[30] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *European conference on computer vision*, pages 852–869. Springer, 2016.

[31] Yu A. Malkov and D. A. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE Trans. Pattern Anal. Mach. Intell.*, 42(4):824–836, apr 2020. ISSN 0162-8828. doi: 10.1109/TPAMI.2018.2889473. URL https://doi.org/10.1109/TPAMI.2018.2889473.

[32] Brian McFee and Gert Lanckriet. Partial order embedding with multiple kernels. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 721–728, 2009.

[33] Gonzalo Mena, David Belanger, Scott Linderman, and Jasper Snoek. Learning latent permutations with gumbel-sinkhorn networks. *arXiv preprint arXiv:1802.08665*, 2018.

[34] Christopher Morris, Nils M Kriege, Franka Bause, Kristian Kersting, Petra Mutzel, and Marion Neumann. Tu-dataset: A collection of benchmark datasets for learning with graphs. *arXiv preprint arXiv:2007.08663*, 2020.

[35] Max B Paulus, Chris J Maddison, and Andreas Krause. Rao-blackwellizing the straight-through gumbel-softmax gradient estimator. *arXiv preprint arXiv:2010.04838*, 2020.

[36] Can Qin, Handong Zhao, Lichen Wang, Huan Wang, Yulun Zhang, and Yun Fu. Slow learning and fast inference: Efficient graph similarity computation via knowledge distillation. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021.

[37] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. *Advances in neural information processing systems*, 20, 2007.

[38] Indradyumna Roy, Abir De, and Soumen Chakrabarti. Adversarial permutation guided node representations for link prediction. *arXiv preprint arXiv:2012.08974*, 2020.

[39] Indradyumna Roy, Soumen Chakrabarti, and Abir De. Maximum common subgraph guided graph retrieval: Late and early interaction networks. *Advances in Neural Information Processing Systems*, 35:32112–32126, 2022.

[40] Indradyumna Roy, Venkata Sai Velugoti, Soumen Chakrabarti, and Abir De. Interpretable neural subgraph matching for graph retrieval. 2022.

[41] Indradyumna Roy, Rishi Agarwal, Soumen Chakrabarti, Anirban Dasgupta, and Abir De. Locality sensitive hashing in fourier frequency domain for soft set containment search. *Advances in Neural Information Processing Systems*, 36, 2024.

[42] Christian Schellewald and Christoph Schnörr. Probabilistic subgraph matching based on convex relaxation. In *International Workshop on Energy Minimization Methods in Computer Vision and Pattern Recognition*, pages 171–186. Springer, 2005.

[43] Xiaobin Tang, Jing Zhang, Bo Chen, Yang Yang, Hong Chen, and Cuiping Li. Bert-int: a bert-based interaction model for knowledge graph alignment. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 3174–3180, 2021.

[44] Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. Order-embeddings of images and language. *ICLR 2016*, 2015. URL https://arxiv.org/pdf/1511.06361.pdf.

[45] Cédric Villani et al. *Optimal transport: old and new*, volume 338. Springer, 2009.

[46] Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014.

[47] Yuting Wu, Xiao Liu, Yansong Feng, Zheng Wang, Rui Yan, and Dongyan Zhao. Relation-aware entity alignment for heterogeneous knowledge graphs. *arXiv preprint arXiv:1908.08210*, 2019.

[48] Tianshu Yu, Junchi Yan, Yilin Wang, Wei Liu, and Baoxin Li. Generalizing graph matching beyond quadratic assignment model. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 861–871, 2018.

[49] Renbo Zhu, Meng Ma, and Ping Wang. Raga: Relation-aware graph attention networks for global entity alignment. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 501–513. Springer, 2021.

[50] Yao Zhu, Hongzhi Liu, Zhonghai Wu, and Yingpeng Du. Relation-aware neighborhood matching model for entity alignment. *arXiv preprint arXiv:2012.08128*, 2020.

[51] Wei Zhuo and Guang Tan. Efficient graph similarity computation with alignment regularization. *Advances in Neural Information Processing Systems*, 35:30181–30193, 2022.