Retrieval of Graph Structured Objects: Theory and Applications

Indradyumna Roy
IIT Bombay, India
indraroy15@cse.iitb.ac.in

Soumen Chakrabarti
IIT Bombay, India
soumen@cse.iitb.ac.in

Abir De
IIT Bombay, India
abir@cse.iitb.ac.in

Abstract

Graph-structured data is ubiquitous across diverse domains like social networks, search, question answering, and drug discovery. Effective retrieval of (sub-)graphs with relevant substructures has become critical to the success of these applications. This proposed tutorial will introduce attendees to state-of-the-art neural methods for graph retrieval, highlighting architectures that effectively model relevance through innovative combinations of early and late interaction mechanisms.

Participants will explore relevance models that represent graphs as sets of embeddings, enabling alignment-driven similarity scoring between query and corpus graphs and supporting diverse cost functions, both symmetric and asymmetric. We will also discuss compatibility with Approximate Nearest Neighbor (ANN) methods, covering recent advances in locality-sensitive hashing (LSH) and other indexing techniques that significantly enhance scalability in graph retrieval.

The tutorial includes hands-on experience with an accessible, PyTorch-integrated toolkit that provides downloadable graph retrieval datasets and baseline implementations of recent methods. Participants will learn to adapt these methods for multi-modal applications — such as molecule, text, and image retrieval — where graph-based retrieval proves particularly effective. Designed for researchers and practitioners, this session delivers both foundational concepts and practical tools for implementing and scaling neural graph retrieval solutions across interdisciplinary applications.

CCS Concepts

• **Information Retrieval**; • **Scalable Search**; • **Neural Combinatorial Approximation**; • **Approximate Nearest Neighbor Search**;

Keywords

Scalable Search and Retrieval, Graph Similarity Scoring

1 Topic and relevance

With the proliferation of graph-structured data, efficient retrieval of relevant (sub-)graphs has become foundational across numerous applications. From social networks to e-commerce platforms to semantic search engines, large-scale graph retrieval powers everything from personalized recommendations to fraud detection and knowledge-based question answering [2, 8, 19, 34, 37, 38]. Beyond web-based applications, graph retrieval is essential across diverse fields, including drug discovery [46], image and video analysis [18, 40], software analysis [21], frequent subgraph mining [45], circuit design [25], and community detection [8], where uncovering complex structural relationships within vast datasets is critical for generating insights and driving innovation.

Recent tutorials related to graph ML in comparable conferences such as CIKM 2024, CIKM 2023, CIKM 2022, WebConf 2025 emphasize scalable *analytics*, clustering, spatio-temporal modeling, user community profiling, blockchain optimization, few shot learning, domain shift, LLM integration, etc., leaving relatively unserved the critical problems of indexing and interpretable retrieval.

As the volume and complexity of graph data grow, retrieval methods must adapt to accommodate a wide range of multi-modal data sources, such as knowledge graphs [3, 43], molecular and chemical structures [23], user interaction networks, and product or content recommendation graphs. Effective graph retrieval demands a balanced approach to similarity, integrating both structural similarity and feature or label similarities to capture a nuanced understanding of relationships. This challenge extends beyond approximate matching to include sophisticated graph similarity measures, such as subgraph isomorphism, graph edit distance (GED), and maximum common subgraph (MCS) size, each offering unique relevance cues. Building retrieval systems that address these varied measures is essential for achieving the accuracy, scalability, and interpretability needed in diverse real-world scenarios.

The design of neural graph retrieval systems necessitates two key qualities: interpretability and scalability. In terms of interpretability, it is essential that the system not only retrieve highly relevant results, but also offer clear, understandable justifications for its choices. An ideal system would generate approximate *alignment witnesses*, which show the underlying reasons for a given match by highlighting important alignments between nodes and edges in query and corpus graphs. Furthermore, similar to traditional information retrieval systems, graph retrieval systems should be trainable under *distant supervision*, being provided only broader relevance indicators rather than relying on detailed alignment labels.

Scalability, meanwhile, is a central challenge for graph retrieval, especially as data volumes increase. Traditional graph matching approaches often rely on cross-graph neural alignment, which renders each corpus graph representation query-dependent and prevents the precomputation of embeddings for efficient retrieval. This limits the ability to implement high-speed, offline indexing techniques,

such as locality-sensitive hashing (LSH) and other Approximate Nearest Neighbor (ANN) methods.

In recent years, there has been significant advances in the area of neural graph retrieval [4, 5, 17, 20, 22, 26, 47]. These methods encompass a wide landscape of techniques, design choices, applications and ML methods. A clearer understanding on these methods among web researchers will allow them to seamlessly use these techniques for a plethora of applications, from multimodal retrieval [44] to question answering [38, 39].

The organizers of the tutorial have extensive experience in neural graph retrieval methods, other applications in web data and general areas in web data driven modeling [6, 7, 11–15, 17, 19, 27–33, 35, 36, 38, 41]. Two of the proposers share experience of presenting several comparable or longer tutorials at WebConf, SIGIR, CIKM, SIGKDD, NeurIPS and AAAI. Their expertise would help in the dissemination of this hands-on tutorial.

2 Tutorial content

At the outset, this tutorial will comprehensively cover the recent advancements on neural graph retrieval methods, guiding participants through the evolution from early neural graph matching models to modern alignment-driven approaches. We will explore both traditional and recent methods for scalable graph retrieval, delving into ANN compatibility, interpretability enhancements, and set alignment-based scoring functions, equipping attendees with a thorough understanding of the latest advancements in neural graph retrieval. Next, we explain the tutorial contents in details.

Overview and applications   Measuring relevance score or similarity between graphs dated back to graph kernels [42]. In recent years, deep graph matching has been a primary focus of the computer vision community, with numerous neural models developed to identify node-level alignments within graphs. These models typically require explicit supervision, relying on ground truth alignments to compute various loss functions. More recent advancements in neural graph retrieval have focused on training with distant supervision [4, 5, 20, 22, 26, 47]. These models typically encode graphs as single, graph-level representations that are compared to compute similarity scores. We will begin with a high level overview of the above works, in the first part of the tutorial.

Graph similarity based on whole graph embeddings   Next, we will describe how we can represent an entire graph into a vector, so that we can approximate graph similarity using vector based similarity. In this context, we will discuss SimGNN [4], NeuroMatch [22], ERIC [47], GEN [20]. We will highlight their modeling architecture, training methods, accuracy and performance on current datasets.

Graph similarity based on graph alignments   While efficient, this single-vector approach is limited in flexibility and struggles to adapt to different similarity measures, such as subgraph isomorphism, MCS, and GED, resulting in suboptimal performance. Additionally, the black-box nature of these similarity scoring models does not provide interpretable justifications, leaving out explicit alignment information.

More recent approaches have made progress by introducing hybrid architectures that combine early and late interaction mechanisms to better handle combinatorial graph matching tasks. These models integrate neural architectures aligned with combinatorial

concepts to handle subgraph isomorphism [33] , MCS [30], and GED [17]. Such techniques improve interpretability by providing alignment-based justifications, making it possible to see which elements within the graphs contribute to their similarity scores.

Efficient retrieval   For scalability, earlier approaches based on coarse graph-level representations could leverage Approximate Nearest Neighbor (ANN) techniques, enabling sublinear-time retrieval for various symmetric [9, 10] and asymmetric scoring functions [24, 29]. More recent efforts are also underway to adapt ANN compatibility to set alignment-based scoring functions [1, 16], aiming to achieve both scalability and flexibility in complex graph retrieval scenarios.

PyTorch Toolkit for Graph Retrieval   To support practical experimentation and application, this tutorial introduces a PyTorch-integrated toolkit designed to make graph retrieval techniques accessible and adaptable. This toolkit offers readily downloadable datasets and baseline implementations of recent methods, providing researchers with the tools to adapt retrieval models for multimodal applications, including molecule, text, and image retrieval. By combining the theoretical underpinnings of neural graph retrieval with hands-on experience using this toolkit, participants will leave equipped to implement scalable, interpretable graph retrieval models that are adaptable to a range of applications.

3 Tutorial Style

This will be a hands-on tutorial, combining interactive lecture segments with practical exercises. Participants will work with a preconfigured Jupyter notebook, requiring them to have Python and PyTorch installed along with specific packages for graph processing (e.g., PyTorch Geometric). Prior to the session, attendees will receive detailed setup instructions, including links to the toolkit and datasets, to ensure a smooth and engaging experience.

4 Schedule

The tutorial will be structured as a half-day session, blending lectures with hands-on exercises.

(A) Overview of Neural Graph Retrieval: **(1)** Duration: 20 minutes. **(2)** Presenter: Soumen Chakrabarti. **(3)** Description: An introduction to graph-based applications across various domains (e.g., web search, KGs, social networks, drug discovery, image retrieval), emphasizing the unique challenges in graph retrieval and the relevance of neural models.

(B) Graph similarity based on whole graph embeddings: **(1)** Duration: 30 minutes **(2)** Presenter: Abir De **(3)** Description: Overview of graph representation learning, including historical neural modeling approaches and the evolution of embedding-based relevance measures. Discussion of trade-offs and considerations in choosing graph similarity techniques for different applications.

(C) Recent Advances in Alignment-Driven Modeling: **(1)** Duration: 25 minutes **(2)** Presenter: Indradyumna Roy **(3)** Description: Presentation on recent advancements in alignment-driven modeling for graph retrieval. Explanation of alignment techniques, relevance scoring, and interpretability through alignment witnesses.

(D Break, 10 minutes

(E) Introduction to ANN and LSH for Graph Retrieval: **(1)** Duration: 20 minutes **(2)** Presenter: Soumen Chakrabarti **(3)** Description:

Introduction to Approximate Nearest Neighbor (ANN) methods and Locality-Sensitive Hashing (LSH) with a historical perspective from web search to modern applications in graph retrieval. Discussion on scalability benefits and indexing for real-time applications.

(F) Hands-On Demo: PyTorch Toolkit for Graph Retrieval: **(1)** Duration: 40 minutes **(2)** Presenter: Indradyumna Roy **(3)** Description: Interactive session on using the PyTorch-based toolkit for graph retrieval, including implementations of both retrieval and LSH/ANN techniques. Participants will work with datasets and try out model adaptations for retrieval tasks.

(G) Applications, Open Challenges, and Future Directions: **(1)** Duration: 20 minutes **(2)** Presenter: Abir De **(3)** Description: Discussion of current and potential applications of neural graph retrieval in fields such as knowledge graphs, NLP, and drug discovery. Overview of open challenges and promising research directions.

(H) Q&A and Discussion: **(1)** Duration: 15 minutes **(2)** Description: An open session for participants to ask questions, clarify doubts, and discuss future possibilities in neural graph retrieval.

5 Audience

This tutorial is intended for a diverse audience, including students, researchers, and industry professionals with an interest in applying neural graph retrieval methods to real-world challenges. We recommend that attendees have a foundational understanding of machine learning and neural networks, as well as familiarity with Python and PyTorch. Knowledge of graph theory and basic concepts in deep learning would be beneficial but is not required, as the tutorial will cover introductory concepts before progressing to more advanced topics.

Prerequisite Knowledge **(1)** Basic understanding of machine learning principles and neural network architectures. **(2)** Familiarity with programming in Python and experience with PyTorch. **(3)** (Optional) Prior knowledge of graph theory and representation learning concepts, although introductory material will be provided.

Potential Learning Outcomes **(1)** An understanding of fundamental concepts in neural graph retrieval, including relevance models, embedding techniques, and alignment-based scoring. **(2)** Practical experience implementing scalable graph retrieval models and adapting them for specific applications, such as drug discovery, video and image retrieval, knowledge graphs (KG) retrieval, chip design, and recommendation systems, where graph problems frequently occur. **(3)** Hands-on experience with a PyTorch-based toolkit for graph retrieval, enabling them to work with downloadable datasets and experiment with baseline models.

6 Previous Editions

This is the first edition of this tutorial, and it has not been presented at any prior conference or workshop.

7 Tutorial Materials

The following materials will be provided to attendees to enhance learning and ensure a comprehensive, hands-on experience:
- **Website:** A dedicated tutorial website will host all materials, including setup instructions, download links for datasets, and relevant resources.
- **Slides:** Presentation slides will be made available for download, providing a structured overview of the concepts discussed.
- **Supplementary Materials:** Attendees will have access to curated references and supplementary readings for further study.
- **Software and Setup Instructions:** Detailed setup instructions will be provided for Python, PyTorch, and required libraries (e.g., PyTorch Geometric) to ensure compatibility with the tutorial's Jupyter notebooks.
- **Source Code and Demos:** Attendees will receive access to source code for example implementations, hands-on demos, and baseline graph retrieval models, ensuring practical familiarity with the techniques covered.
- **Recorded Videos and Demos:** Videos demonstrating key techniques and sample workflows will be shared, along with recordings of the tutorial for future reference.

All materials provided will be open-source and free of copyright issues, with source code and datasets available under open licenses.

8 Organizers

Indradyumna Roy He is a final-year Ph.D. candidate in the Department of Computer Science and Engineering at IIT Bombay. His research specializes in neural graph retrieval, combining deep learning methods with graph combinatorial optimization, scalable retrieval techniques, and interpretability in graph models. His work is supported by prestigious fellowships, including the Prime Minister's Research Fellowship (PMRF), Qualcomm Innovation Fellowship, and Google Ph.D. Fellowship. He has published at leading venues such as NeurIPS, AAAI, and InterSpeech, and contributed as a reviewer for major conferences like NeurIPS, ICLR, AAAI, and AISTATS. His practical expertise in scalable and interpretable graph retrieval systems brings valuable insights to this tutorial, providing participants with an in-depth understanding of state-of-the-art developments and applications in the field.

Soumen Chakrabarti He is a Professor in the Department of Computer Science and Engineering at IIT Bombay, internationally recognized for his work on graph-based representation learning, knowledge graphs, and advanced search technologies. His research has been instrumental in enhancing graph neural networks, knowledge graph embeddings, and multi-modal question answering. He has published extensively in prestigious venues like WWW, SIGIR, SIGKDD, EMNLP, and VLDB and received numerous awards, including the WWW Best Paper Award (1999), the ICDE 10-Year Influential Paper Award (2012), and the Bhatnagar Prize (2014). He authored one of the earliest books on Web search and mining, a foundational reference in the field, and holds 13 U.S. patents. His roles in organizing premier conferences, such as founding the ACM Web Search and Data Mining (WSDM) Conference series, underscore his ability to lead discussions at the intersection of graphs and machine learning. With his deep knowledge and practical insights, Prof. Chakrabarti is exceptionally qualified to deliver a thorough and engaging introduction to neural graph retrieval.

Abir De He is an Assistant Professor in the Department of Computer Science and Engineering at the Indian Institute of Technology (IIT) Bombay. Prof. De's research focuses on developing neural architectures for graph-based tasks, data-efficient learning methods, and human-in-the-loop models—foundational elements in neural

graph retrieval. His numerous awards, including the INAE Young Engineer Award (2021), the Prof. Krithi Ramamritham Award for Creative Research at IIT Bombay (2020), and the INAE Best Ph.D. Project Award (2019), underscore his impact and expertise in these areas. Prof. De has organized influential events such as (1) the Workshop on Subset Selection in Machine Learning at ICML 2021, (2) the Workshop on Temporal Point Processes at NeurIPS 2022, and (3) the IndoML Symposium in 2023, demonstrating his ability to lead discussions on complex machine learning topics.

References

[1] Alexandr Andoni, Piotr Indyk, and Ilya Razenshteyn. 2018. Approximate nearest neighbor search in high dimensions. In *Proceedings of the International Congress of Mathematicians: Rio de Janeiro 2018*. World Scientific, 3287–3318.

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

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

[4] Yunsheng Bai, Hao Ding, Song Bian, Ting Chen, Yizhou Sun, and Wei Wang. 2019. Simgnn: A neural network approach to fast graph similarity computation. In *WSDM*. 384–392.

[5] Yunsheng Bai, Hao Ding, Ken Gu, Yizhou Sun, and Wei Wang. 2020. Learning-based efficient graph similarity computation via multi-scale convolutional set matching. In *AAAI*.

[6] Soumen Chakrabarti, Harkanwar Singh, Shubham Lohiya, Prachi Jain, and Mausam . 2022. Joint Completion and Alignment of Multilingual Knowledge Graphs. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang (Eds.). Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 11922–11938. https://doi.org/10.18653/v1/2022.emnlp-main.817

[7] P. Chakraborty, S. Ranu, K. S. I. Mantri, and A. De. 2023. Learning and Maximizing Influence in Social Networks Under Capacity Constraints. In *WSDM*.

[8] Rong-Ching Chang and Jiawei Zhang. 2024. CommunityKG-RAG: Leveraging Community Structures in Knowledge Graphs for Advanced Retrieval-Augmented Generation in Fact-Checking. *arXiv preprint arXiv:2408.08535* (2024).

[9] Moses S Charikar. 2002. Similarity estimation techniques from rounding algorithms. In *STOC*. 380–388.

[10] Mayur Datar, Nicole Immorlica, Piotr Indyk, and Vahab S Mirrokni. 2004. Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of the twentieth annual symposium on Computational geometry*. 253–262.

[11] A. De, S. Bhattacharya, P. Bhattacharya, N. Ganguly, and S. Chakrabarti. 2014. Learning a Linear Influence Model from Transient Opinion Dynamics. In *CIKM*.

[12] A. De, S. Bhattacharya, and N. Ganguly. 2018. Demarcating Endogenous and Exogenous Opinion Diffusion Process on Social Networks. In *WWW*.

[13] Abir De, Sourangshu Bhattacharya, Sourav Sarkar, Niloy Ganguly, and Soumen Chakrabarti. 2016. Discriminative link prediction using local, community, and global signals. *IEEE Transactions on Knowledge and Data Engineering* 28, 8 (2016), 2057–2070.

[14] P. Deshpande, K. Marathe, A. De, and S. Sarawagi. 2021. Long Horizon Forecasting with Temporal Point Processes. In *WSDM*.

[15] V. Gupta, S. Bedathur, S. Bhattacharya, and A. De. 2021. Learning Temporal Point Processes with Intermittent Observations. In *AISTATS*.

[16] Piotr Indyk and Rajeev Motwani. 1998. Approximate nearest neighbors: towards removing the curse of dimensionality. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*. 604–613.

[17] Eeshaan Jain, Indradyumna Roy, Saswat Meher, Soumen Chakrabarti, and Abir De. 2024. Graph Edit Distance with General Costs Using Neural Set Divergence. *arXiv preprint arXiv:2409.17687* (2024).

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

[19] Mandar Joshi, Uma Sawant, and Soumen Chakrabarti. 2014. Knowledge graph and corpus driven segmentation and answer inference for telegraphic entity-seeking queries. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 1104–1114.

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

[21] Xiang Ling, Lingfei Wu, Saizhuo Wang, Gaoning Pan, Tengfei Ma, Fangli Xu, Alex X Liu, Chunming Wu, and Shouling Ji. 2021. Deep Graph Matching and Searching for Semantic Code Retrieval. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 15, 5 (2021), 1–21. https://dl.acm.org/doi/pdf/10.1145/3447571

[22] Zhaoyu Lou, Jiaxuan You, Chengtao Wen, Arquimedes Canedo, Jure Leskovec, et al. 2020. Neural Subgraph Matching. *arXiv preprint arXiv:2007.03092* (2020).

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

[24] Behnam Neyshabur and Nathan Srebro. 2015. On symmetric and asymmetric lshs for inner product search. In *International Conference on Machine Learning*. PMLR, 1926–1934.

[25] Miles Ohlrich, Carl Ebeling, Eka Ginting, and Lisa Sather. 1993. Subgemini: Identifying subcircuits using a fast subgraph isomorphism algorithm. In *Proceedings of the 30th International Design Automation Conference*. 31–37.

[26] Can Qin, Handong Zhao, Lichen Wang, Huan Wang, Yulun Zhang, and Yun Fu. 2021. Slow Learning and Fast Inference: Efficient Graph Similarity Computation via Knowledge Distillation. In *Thirty-Fifth Conference on Neural Information Processing Systems*.

[27] V. Raj, I. Roy, A. Ramachandran, S. Chakrabarti, and A. De. 2025. Charting the Design Space of Neural Graph Representations for Subgraph Matching. In *ICLR*.

[28] Ashwin Ramachandran, Vaibhav Raj, Indrayumna Roy, Soumen Chakrabarti, and Abir De. 2024. Iteratively Refined Early Interaction Alignment for Subgraph Matching based Graph Retrieval. In *NeurIPS*. https://neurips.cc/virtual/2024/poster/93261

[29] Indradyumna Roy, Rishi Agarwal, Soumen Chakrabarti, Anirban Dasgupta, and Abir De. 2023. Locality Sensitive Hashing in Fourier Frequency Domain For Soft Set Containment Search. *NeurIPS* (2023).

[30] Indradyumna Roy, Soumen Chakrabarti, and Abir De. 2022. Maximum Common Subgraph Guided Graph Retrieval: Late and Early Interaction Networks. *Advances in Neural Information Processing Systems* 35 (2022), 32112–32126.

[31] Indradyumna Roy, Abir De, and Soumen Chakrabarti. 2020. Adversarial Permutation Guided Node Representations for Link Prediction. *arXiv preprint arXiv:2012.08974* (2020).

[32] I. Roy, E. Jain, S. Chakrabarti, and A. De. 2025. Clique Number Estimation via Differentiable Functions of Adjacency Matrix Permutations. In *ICLR*.

[33] Indradyumna Roy, Venkata Sai Velugoti, Soumen Chakrabarti, and Abir De. 2022. Interpretable Neural Subgraph Matching for Graph Retrieval. (2022).

[34] Amrita Saha, Ghulam Ahmed Ansari, Abhishek Laddha, Karthik Sankaranarayanan, and Soumen Chakrabarti. 2019. Complex program induction for querying knowledge bases in the absence of gold programs. *Transactions of the Association for Computational Linguistics* 7 (2019), 185–200.

[35] A. Saha, B. Samanta, N. Ganguly, and A. De. 2018. CRPP: Competing Recurrent Point Process for Modeling Visibility Dynamics in Information Diffusion. In *CIKM*.

[36] B. Samanta, A. De, and N. Ganguly. 2017. STRM: A sister tweet reinforcement process for modeling hashtag popularity. In *INFOCOM*.

[37] Uma Sawant and Soumen Chakrabarti. 2013. Learning joint query interpretation and response ranking. In *Proceedings of the 22nd international conference on World Wide Web*. 1099–1110.

[38] Uma Sawant, Saurabh Garg, Soumen Chakrabarti, and Ganesh Ramakrishnan. 2019. Neural architecture for question answering using a knowledge graph and web corpus. *Information Retrieval Journal* 22, 3 (2019), 324–349.

[39] Apoorv Saxena, Aditay Tripathi, and Partha Talukdar. 2020. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 4498–4507.

[40] Kim Shearer, Horst Bunke, and Svetha Venkatesh. 2001. Video indexing and similarity retrieval by largest common subgraph detection using decision trees. *Pattern Recognition* 34, 5 (2001), 1075–1091.

[41] Harkanwar Singh, Prachi Jain, Soumen Chakrabarti, et al. 2021. Multilingual Knowledge Graph Completion with Joint Relation and Entity Alignment. *arXiv preprint arXiv:2104.08804* (2021).

[42] S Vichy N Vishwanathan, Nicol N Schraudolph, Risi Kondor, and Karsten M Borgwardt. 2010. Graph kernels. *Journal of Machine Learning Research* 11 (2010), 1201–1242.

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

[44] Kaiye Wang, Qiyue Yin, Wei Wang, Shu Wu, and Liang Wang. 2016. A comprehensive survey on cross-modal retrieval. *arXiv preprint arXiv:1607.06215* (2016).

[45] Xifeng Yan, Philip S Yu, and Jiawei Han. 2004. Graph indexing: A frequent structure-based approach. In *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*. 335–346.

[46] Xiangxiang Zeng, Xinqi Tu, Yuansheng Liu, Xiangzheng Fu, and Yansen Su. 2022. Toward better drug discovery with knowledge graph. *Current opinion in structural biology* 72 (2022), 114–126.

[47] Wei Zhuo and Guang Tan. 2022. Efficient graph similarity computation with alignment regularization. *NeurIPS* (2022).