Tutorial: Graph-based Methods for Similarity Searches

Larissa C. Shimomura[1], Daniel S. Kaster[2]

[1]Department of Mathematics and Computer Science
Eindhoven University of Technology
Eindhoven – Netherlands.

[2]Department of Computer Science
University of Londrina
Londrina, PR – Brazil

l.capobianco.shimomura@tue.nl, dskaster@uel.br

Abstract. *Similarity searches are based on retrieving similar data of one or more data used as reference according to an intrinsic characteristic of the data. Recently, graph-based methods have emerged as a very efficient option to execute similarity queries in metric and non-metric spaces. Graphs model the interconnectivity among data, enabling us to explore relationships and neighbors in an agile way. In this tutorial, we will give an introduction to similarity searches and present graph-based methods for similarity searches, the types of graphs used in such methods, their properties, open challenges, and research opportunities.*

Keywords: *Similarity Searches. Graph-based Methods. Proximity Graphs. Metric Spaces.*

1. Overview

Similarity searches are based on retrieving similar data of one or more data used as reference according to an intrinsic characteristic of the data. The similarity between two complex data is measured according to a (dis)similarity function (commonly, a distance function) to the feature vectors describing the compared data.Similarity retrieval of data is employed in a wide range of modern applications, for example, content-based image retrieval, pattern recognition, and recommender systems, to name a few[Zezula et al. 2010].

In this tutorial, we will introduce the area of similarity searches, its graph-based methods, and the open challenges of the area. This tutorial is an introduction-level tutorial and has not been presented before. The target audience of the tutorial includes but is not limited to practitioners, researchers, and students active in computer science or related fields such as information systems research, systems design, and engineering.

This tutorial will be presented in Portuguese, as we assume that most of the audience are Brazilians. If there is a high attendance of foreigners, we are open to presenting the tutorial in English.

2. Tutorial Outline

In this tutorial, we will introduce the basics of similarity searches, show how graphs have been used for similarity searches, and present recent advances in the subject. The tutorial is divided into three parts, covering the following topics.

Part I

Introduction to similarity searches – The first part of the tutorial introduces similarity searching. Similarity searches refer to the task of retrieving similar data of one or more data used as reference according to an intrinsic characteristic of the data. The intrinsic characteristic of the complex data is usually represented by a sequence of values composing feature vectors. The similarity between two complex data (images, videos, time series, etc.) is measured by applying a (dis)similarity function to the feature vectors describing the compared data. Distance functions are commonly used as the (dis)similarity; therefore, we can model the similarity search problem as a metric space search problem[Zezula et al. 2010]. In this first part of the tutorial, we will present an introduction to the area of similarity searches, feature extraction, distance measures used, and properties of metric spaces as well as types of similarity queries. We will also give an overview of methods frequently used for similarity searches, such as tree-based, permutation-based, and hashing-based methods. The following topics of this tutorial will focus on similarity graphs.

Part II

Properties of graph-based similarity searches – The second part of the tutorial discusses properties fundamental to similarity searching on proximity graphs, the main types of base graphs used in the graph-based methods, and search algorithms. Recently, graph-based methods have emerged as a very efficient option to execute similarity queries in metric and non-metric spaces and have been gaining attention in the research area. Graphs for similarity searches fall in the category of proximity graphs. A proximity graph is a graph defined as $G = (V, E)$, where V is the set of vertices (nodes) and E is the set of edges that connect pairs of vertices in V and each pair of vertices $(v, u) \in V$ is connected by an edge $e = (u, v)$, $e \in E$, if and only if u and v fulfill a defined property P [Shimomura et al. 2021]. The property P is called *neighborhood criterion* and defines the type of proximity graph.

The fundamental approach to perform similarity queries on a proximity graph is to employ the so-called *spatial approximation*, introduced by Navarro in [Navarro 2002]. Given a query element q and proximity measure δ, this approach consists in starting from a given vertex u in the graph and iteratively traverse the graph in a way to get spatially closer and closer to the elements that are the most similar to q. Every iteration propagates the search from a vertex u to its “neighbors” ($N(u)$) that are closer to q and consequently more likely to reach the answer, where vertices in $N(u)$ are adjacent to v in the graph.

However, not all graph types ensure that the spatial approximation approach can always reach the global optimal answer (most similar nodes/objects compared to the object being queried). In this part of the tutorial, we will present the spatial approximation property, why this property is essential for graph-based methods in similarity searches, and the limitations of the types of graphs that hold this property.

Types of base graphs and search algorithms of graph-based methods – Given the limitations of graph-based methods due to the spatial approximation property, the challenge of determining a good graph structure (neighborhood criterion) can be summarized by the following question: "How to connect the vertices (complex data) ensuring that all vertices can be easily retrieved according to the similarity space?". Some of the types of graphs that are frequently used as a base in graph-based methods are k-NN Graph, the Relative Neighborhood Graph (RNG), and the Navigable Small World Graph (NSW)[Paredes et al. 2006, Shimomura et al. 2021, Wang et al. 2021]. Each of these types of graphs works for different situations, and their properties are used in state-of-the-art graph-based methods. We will present the main graph types by showing the same example so the audience can visually understand their main differences and experimental results on how these graphs perform under different datasets using search algorithms[Hajebi et al. 2011]. We will use one-third of the tutorial time to present this and the previous topic.

Part III

Recent graph-based methods – In the third part of the tutorial, we will present some of the state-of-the-art methods, the most used techniques in such methods, and other related problems and applications of graph-based methods. One example of a technique used is to build a hierarchical graph structure, in which at each level is built a proximity graph, and any vertex (or an area of the graph) of an upper level can point to the next level and eventually guide the search closer to the similar vertices/nodes. An example of a method that uses this approach is the Hierarchical Navigable Small World Graph (HNSW)[Malkov and Yashunin 2020]. Another example is the use of the construction properties of the Relative Neighborhood Graph. The RNG does not connect vertices (objects) in the graph in which there is a closer neighbor, and such property is used in recent methods to prune edges when using, for example, a k-NN graph variation. An example of a method that uses this approach is the NSG (Navigating Spreading-out Graph)[Fu et al. 2019].

Parameter setting for graph-based methods – Different graph-based methods have different parameters and can have different performances according to the dataset. So, depending on the dataset, to achieve the desired execution time and memory consumption, choosing the best parameters are crucial. In this topic, we show how some of the main parameters of each one of these graphs can affect execution time and also present a method on how to choose such parameters automatically using meta-learning[Oyamada et al. 2020].

Other applications for graph-based methods – The graph-based methods presented were primarily proposed for similarity searches[Amagata et al. 2022]. However, there are other applications that such graph methods can also be used. One of the recent proposals is to use proximity graphs for outlier detection. Since the graph models the similarity space, we can use the proximity graph to verify which are outliers in the data. A second use case is the use of such graph-based methods for clustering since the structure of the graph in itself and the weights of the graphs can give much information about the distance distribution of the graph.

Conclusion and Research Opportunities – We will finish the tutorial by presenting our conclusions and open research questions based on the topics presented.

3. About the Authors

The tutorial will be presented by both authors, Larissa C. Shimomura and Daniel S. Kaster. It follows the biography of the authors.

Larissa C. Shimomura – Larissa C. Shimomura is a Ph.D. Candidate in the Department of Mathematics and Computer Science of Technische Universiteit Eindhoven, Netherlands. She obtained her bachelor's and master's degree in Computer Science from the University of Londrina. Her master's thesis was on graph-based methods for similarity searches, and she was awarded an honorable mention in the master thesis contest of SBBD 2021. The thesis's main contributions were an experimental evaluation of graph-based methods for similarity search and the HGraph method, which received the Gabriela e Roland Wagner Award in DEXA 2019. During her Ph.D., Larissa worked on the EU Horizon 2020 SmartDataLake project contributing to the Entity Resolution task. Her current research focus is on data dependencies for graph data and its applications in data profiling and data quality.

Daniel S. Kaster – Daniel S. Kaster has been a professor with the Department of Computer Science at the University of Londrina since 2001. He obtained his bachelor's from the University of Londrina (1998), master's degree from the University of Campinas (2001), and Ph.D. in Computer Science and Computational Mathematics from the University of São Paulo (2012). Daniel S. Kaster has also worked as a visiting professor in the Data Systems Group of the University of Waterloo in Canada (2019-2020). He is currently a member of the Steering Committee of the Special Committee for Databases (CEBD) of the Brazilian Computer Society (SBC). His main research topics are multimedia and spatiotemporal databases, indexing and optimization of similarity searches, trajectory data mining, content-based image retrieval, and data cleaning.

Aknowledgements

This work has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 825041 and the Brazilian agencies Araucaria Foundation, CNPq, and CAPES.

References

Amagata, D., Onizuka, M., and Hara, T. (2022). Fast, exact, and parallel-friendly outlier detection algorithms with proximity graph in metric spaces. *The VLDB Journal*, pages 1–25.

Fu, C., Xiang, C., Wang, C., and Cai, D. (2019). Fast approximate nearest neighbor search with the navigating spreading-out graph. *Proc. VLDB Endow.*, 12(5):461–474.

Hajebi, K., Abbasi-Yadkori, Y., Shahbazi, H., and Zhang, H. (2011). Fast approximate nearest-neighbor search with k-nearest neighbor graph. In *Int'l Joint Conf. on Artificial Intelligence IJCAI*, pages 1312–1317.

Malkov, Y. A. and Yashunin, D. A. (2020). Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(4):824–836.

Navarro, G. (2002). Searching in metric spaces by spatial approximation. *The VLDB Journal The Int'l Journal on Very Large Data Bases*, 11(1):28–46.

Oyamada, R. S., Shimomura, L. C., Junior, S. B., and Kaster, D. S. (2020). Towards proximity graph auto-configuration: An approach based on meta-learning. In Darmont, J., Novikov, B., and Wrembel, R., editors, *Advances in Databases and Information Systems*, pages 93–107, Cham. Springer International Publishing.

Paredes, R., Chávez, E., Figueroa, K., and Navarro, G. (2006). *Practical Construction of k-Nearest Neighbor Graphs in Metric Spaces*, pages 85–97. Springer Berlin Heidelberg.

Shimomura, L. C., Oyamada, R. S., Vieira, M. R., and Kaster, D. S. (2021). A survey on graph-based methods for similarity searches in metric spaces. *Information Systems*, 95:101507.

Wang, M., Xu, X., Yue, Q., and Wang, Y. (2021). A comprehensive survey and experimental comparison of graph-based approximate nearest neighbor search. *Proc. VLDB Endow.*, 14(11):1964–1978.

Zezula, P., Amato, G., Dohnal, V., and Batko, M. (2010). *Similarity Search: The Metric Space Approach*. Springer, 1st edition.