Hierarchical Structural Information – Theory and Applications

Marzena Bielecka[1] [0000−0002−8472−684X], Andrzej Bielecki[1] [0000−0002−0192−3785], Aleksander Suchorab[1] [0009−0001−5006−6492], and Igor Wojnicki[1] [0000−0002−4406−4992]

AGH University of Krakow, al.Mickiewicza 30, 30-059 Kraków, Poland
bielecka@agh.edu.pl, bielecki@agh.edu.pl, asuchorab@agh.edu.pl, wojnicki@agh.edu.pl

Abstract. This paper introduces a novel measure to quantify structural information in hierarchical graphs. It addresses the limitation of current methods that do not adequately account for hierarchical structures. By considering inner structural information and distinguishability of higher-level vertices, the proposed measure captures the additional information generated by the hierarchy. The hypothesis that hierarchical graphs contain more structural information is validated using the "Countries" dataset. The results demonstrate a measurable increase in the information content when the hierarchical structure is considered, compared to a simple graph representation. This highlights the importance of recognizing and utilizing hierarchy to enhance the informational richness of graphs, potentially improving the performance of graph-based machine learning models.

Keywords: Information quantity · Structural information · Hierarchical information · Hierarchical data · Amount of information · Network analysis

1 Introduction and Motivation

The question of how much information is gathered in a graph, considering its topology and hence the structure, has been researched since the mid-20th century. The subject is brought to light by recent advances in machine learning in cases where there is a need to learn, classify, or regress information expressed with graphs. The reason for this is that information quantity addresses the quality of the data set. The more information there is in the graph, the more reliable Graph Neural Networks (GNNs) or graph embeddings would be.

Although the graph structure information quantity measure for general graphs is known, there is little research regarding hierarchical graphs. And this is the subject we want to address. We will focus on structural information in a case for which a hierarchical graph is available or a hierarchy can be identified unambiguously. In such a case, which is supported by real-world examples, additional information that is provided as edge or node labels, can express a hierarchy

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97632-2_4

within the graph, transforming it into a hierarchical graph. As a result, a part of the graph semantics is turned into its structure. Our hypothesis is that the amount of structural information in such a structurally extended graph is greater than the initial one. To support this hypothesis, we would like to present a measure and apply it to a real-world example.

It should be stressed that this paper addresses the problem of the structure of relational data in the context of machine learning. The proposed methodology, however, goes beyond pure computer science and refers to cybernetics. Namely, the method can be used to analyse the complexity of multi-levelled social networks [2] as well as biological systems that are hierarchical by their nature [31]. The biological application are crucial because, in contemporary biology, the role of information and its processing is regarded as fundamental [11, 19]. The presented approach can also be applied to theoretical foundations of embodied autonomous agents, that act in their environment and create knowledge about it [4]. Such knowledge is usually modelled as hierarchical formal structures.

In the following sections a brief introduction to complex data structures for machine learning and a proposal of an information quantity measure for hierarchical graphs are given. Furthermore, a real-world example dataset in the form of a graph with well established hierarchy is also presented along with the calculations of information quantity.

1.1 Graph Data Structures in Machine Learning

Graph data structures are fundamental for machine learning applications that require relational modeling. A graph consists of nodes (also called vertices) and edges, representing entities and their relationships. This structure allows machine learning models to effectively handle non-Euclidean data, making them well suited for social network analysis, recommendation systems, and biological modeling [24]. A major advantage of using graphs in machine learning is their ability to capture complex dependencies between data points. Unlike traditional vector-based models, graph-based approaches incorporate relational information, enabling better performance in structured data environments [21]. This is particularly valuable in semi-supervised learning, where labeled data is limited but unlabeled data can still contribute to learning by propagating information across connected nodes [28].

One of the most important techniques in graph-based learning is graph embedding, which transforms graph data into lower-dimensional vector representations while preserving structural properties.

As graph-based machine learning continues to evolve, more and more research is focused on enhancing scalability, interpretability, and robustness to expand its applicability across various domains.

DOI: 10.1007/978-3-031-97632-2_4

1.2 Hierarchical Graph Data Structures and Their Applications in Machine Learning

Hierarchical graph data structures extend beyond traditional graphs by introducing multiple levels of abstraction, where each level represents a transformation of the previous one. These structures effectively model complex relationships in large-scale datasets, enabling machine learning algorithms to leverage multi-scale information efficiently. Unlike simple graphs, hierarchical graphs allow nested relationships while maintaining graph-based flexibility, making them ideal for applications in NLP, computer vision, and bioinformatics [23].

A significant use case of hierarchical graphs is the Hierarchical Graph Neural Networks (HGNNs), where graph representations are refined across multiple levels to enhance feature extraction. In NLP, they help capture sentence structures and contextual dependencies, improving text classification and sentiment analysis [32]. In bioinformatics, hierarchical graphs model molecular interactions, improving drug discovery and protein structure prediction [20]. Hierarchical structures also benefit computer vision, where multiresolution graph representations improve image segmentation, object recognition, and scene understanding. In autonomous systems, hierarchical graph-based models facilitate efficient decision making by structuring sensor data for multiscale reasoning [38]. As machine learning tasks grow in complexity, dynamic hierarchical graph models, that adapt to changing relationships, are developed. This is particularly useful in fraud detection, social network analysis, and recommendation systems, where interactions between entities evolve. Recent advances in Graph Attention Networks (GATs) with hierarchical structures have improved model interpretability and efficiency in handling large-scale data [27].

With continued advancements, hierarchical graph-based learning is expected to further enhance performance across AI-driven applications.

1.3 Graph Embeddings

A graph embedding technique transforms relationships in a graph into lower dimensional space, namely a vector with fixed length, while preserving as much as possible of the structural information and properties. Having a set of fixed-length vectors that represent nodes makes them easier to analyze and more suitable for machine learning. It enables tasks such as node classification, recommendation, link prediction, graph completion, or clustering, to name a few, on graph data with vector-based machine learning concepts and technologies.

There are multiple graph embedding techniques that can be used with varying applicability depending on the actual characteristics of the graphs [34] with some benchmarks [13] and taxonomy [9], available. One of the challenges is to verify if the embeddings preserve semantics [15]. As it has been observed, the sparsity of graphs degrades embedding performance greatly [29]. Performance gains reported for one graph type may not translate to other types of graphs. It is also challenging to compare different embedding methods and predict which one will be best for a given graph [13]. There is no standard way to quantify

DOI: 10.1007/978-3-031-97632-2_4

the advantages of one approach over another. Many methods are evaluated on dense, curated datasets that do not reflect the complexities of real-world graphs.

1.4 Graph Neural Networks

Graph Neural Networks are a class of deep learning models designed to process graph-structured data. Unlike traditional neural networks that operate on Euclidean data (e.g., images and text represented as vectors), GNNs can model relationships in non-Euclidean spaces. They achieve this by iteratively aggregating information from neighboring nodes, capturing both local and global dependencies within the graph [39]. A key property of GNNs is their ability to leverage relational information rather than treating data points as independent entities [18]. This makes them particularly effective in domains where interconnected data is essential. Their inductive learning capability enables them to generalize unseen graphs [35], and their flexibility allows them to process various types of graphs, including directed, undirected, weighted, and heterogeneous structures [37].

Compared to traditional machine learning models that rely on handcrafted features, GNNs automatically learn hierarchical and context-aware representations, improving performance in node classification, link prediction, and graph clustering. They are widely applied in social network analysis (e.g., community detection and friend recommendation) [18], biomedical research (e.g., protein structure prediction and drug discovery) [37], and fraud detection by analyzing transaction networks for anomalies. In natural language processing (NLP), they enhance tasks such as semantic parsing and document classification by capturing contextual relationships [35, 8, 36]. Furthermore, GNNs contribute to traffic prediction and optimization in autonomous systems [22]. By exploiting graph structures, they enable more accurate and context-aware predictions, making them a crucial tool in modern deep learning.

2 Entropy-based Information Quantity for Graph-based Hierarchical Structures

Before starting any machine learning process, the dataset has to be assessed if it is up to providing proper input for the model. Especially, it has to be confirmed if the data quantity and quality are sufficient. In case of vectors, there are several heuristics and methods that work well. For graphs, one should begin with assessment of how much structural information there is in it, to make sure that there is enough information to feed the subsequent learning process.

To measure amount of structural information in a graph, there are two starting points to consider. One point is to start with the graph entropy [26], which is based on information theory. Another one begins with an ontology, which results in the Hellerman's approach [14]. Even though their starting points are different, the end results are quite similar, being a single measure of information quantity. In this paper, we choose to apply the Hellerman's approach.

However, in case of graphs where a hierarchy can be identified, the proposed Hellerman's approach is lacking. This includes hierarchical graphs or graphs in which such hierarchy could be identified or inferred, based on semantic information such as edge or node labeling.

In this section, we put forward a proposal of such a measure that is suitable for hierarchical graphs or graphs with identifiable hierarchy.

2.1 Hierarchical Information

Hierarchical information appears when substructures of a given structure or a group of elements of a given set constitute a single element in a new structure or set at the higher level of hierarchy. Then, among such new elements novel relations can appear. Let us formalize this idea on the basis of the concept of structural information signalized in [3], worked out in detail in [5] and applied to cognitive maps in [6]. Thus, let X be an k-element set with a relation $\mathcal{R}$ on it that generates the directed graph (digraph) $G(X, \mathcal{R})$. Let us recall that the nodes of $G(X, \mathcal{R})$ correspond to elements of X, whereas the directed edges connect the nodes that correspond to elements in relation to each other [5]. To put it briefly, the form of the graph $G(X, \mathcal{R})$ is the structural information on the set X, that is generated by relation $\mathcal{R}$. The amount of such information can be calculated numerically by applying the concept worked out by Hellerman [14], as follows.

$$H = -N \sum_{k=1}^{K} \frac{n_k}{N} log \frac{n_k}{N} \tag{1}$$

where N is the number of vertices in the graph, K is the number of equivalence classes and n_k is the number of vertices belonging to class n_k.

It should be mentioned that the way in which the information in graphs can be calculated was studied since the fifties of the 20th century, but it was referred to entropy in graphs and not to ontological aspects of structural information as such [12, 25, 26, 30, 33]. The formulae used to calculate the amount of information in graphs were consistent in Shanon information theory, Hellerman's concept, and the idea presented in [5].

Let us consider the way in which the amount of information can be calculated in hierarchical structures. The mentioned set X with relation $\mathcal{R}$ constitutes the first level of the hierarchy, on which both the information and the way the amount of it is calculated, are given in [5] – see Fig.1 as an example, where $X = \{x_1, \dots, x_{11}\}$ and relation $\mathcal{R}$ generate the digraph $G(X)$. Let $P(G)$ denote the set of all subgraphs of $G(X, \mathcal{R})$. Let N be the cardinality of $P(G)$. From the set $P(G)$ there are selected elements that will constitute components of the set, let us say $\mathscr{X}$, on the next level of hierarchy. Let us assume that $card\ \mathscr{X} = n \leq N$. Then, the amount of information H_c, generated by the choice of the number of the elements of $\mathscr{X}$, is equal to

$$H_c = -N \left(\frac{n}{N} \log \frac{n}{N} + \frac{N-n}{N} \log \frac{N-n}{N} \right). \tag{2}$$

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97632-2_4

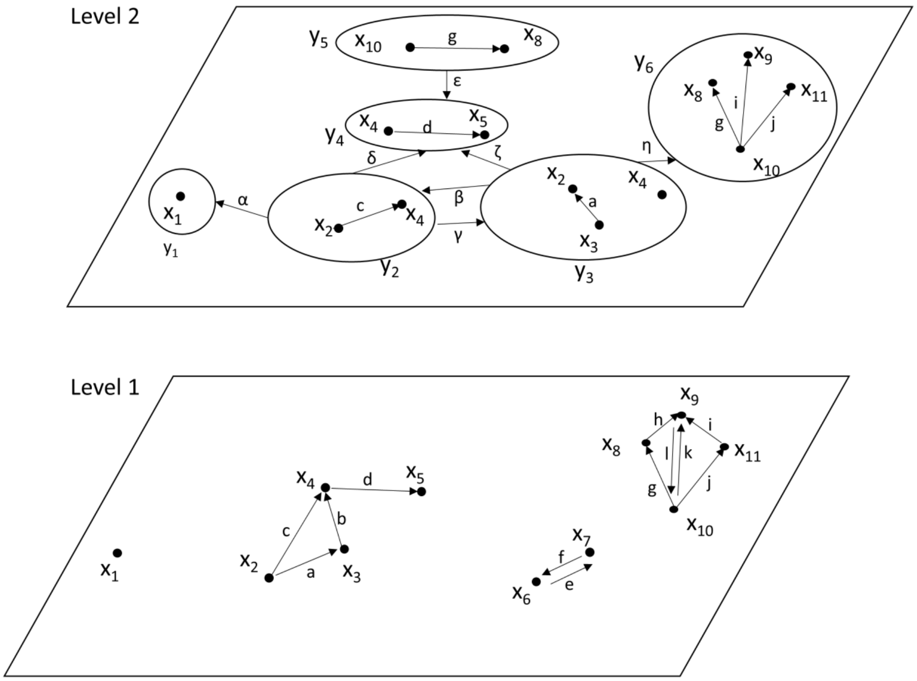

Fig. 1. Example of hierarchical information

DOI: 10.1007/978-3-031-97632-2_4

The elements of $\mathscr{X}$ are graphs themselves, so each element has an inner structure and, as a consequence, inner structural information. Thus, for each element of $\mathscr{X}$, its amount of structural information can be calculated in the same way as for graph $G(X, \mathcal{R})$. Let a relation $\mathscr{R}$ be defined on $\mathscr{X}$. It generates the digraph $\mathscr{G}(\mathscr{X}, \mathscr{R})$ at the second level of the hierarchy – see Fig.1. For this graph, the amount of information can be calculated in the same way as for graph $G(X, \mathcal{R})$.

In summary, selecting elements that constitute higher-level components in a hierarchical structure generates three types of information:

(a) information generated by number of the chosen elements – amount of this type of information is given by formula (2);
(b) inner structural information of every element on the higher level;
(c) information generated by the distinguishability of higher-level vertices resulting from their internal structure.

3 Computing the Hierarchical Information of an Example Dataset

Let us introduce an example, a real-world dataset, and calculate the amount of structural information for it. Then, let us identify its hierarchy and compare the amount of structural information when the hierarchy is taken into consideration.

3.1 The Dataset

The "Countries" dataset [7], provided by PyKEEN (Python KnowlEdge EmbeddiNgs) [1], will be used to demonstrate the computation of hierarchical information. The motivation for using this dataset is its relatively small size, ease of interpretation of actual data, and the fact that it can be turned into a hierarchical structure easily and consistently.

The dataset consists of countries, regions that group countries and continents that group regions. These three kinds of nodes are not explicitly differentiated, but the type of node can be deduced based on its connections, namely edge labels. The dataset contains two kinds of edges differentiated by labeling: "neighbor", a symmetric relationship denoting that two countries are next to each other, and "locatedin", which denotes a lower-level entity belonging to a higher-level entity. As a result, the semantic information encoded with the "locatedin" labels can be used to build the hierarchy providing additional structural information.

An excerpt from the dataset considering two countries, namely Czechia and Poland is given in Fig. 2. The hierarchical decomposition of the excerpt is given in Fig. 3. There are edges labeled "locatedin" between countries and regions, namely "Czechia"–"Eastern_Europe" and "Poland"–"Eastern_Europe", and between regions and continents, which is "Eastern_Europe"–"Europe". They create a three-level hierarchy with "Poland", "Czechia" and other neighboring countries at level 1, "Eastern_Europe" at the level 2, and "Europe" at the level 3.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97632-2_4

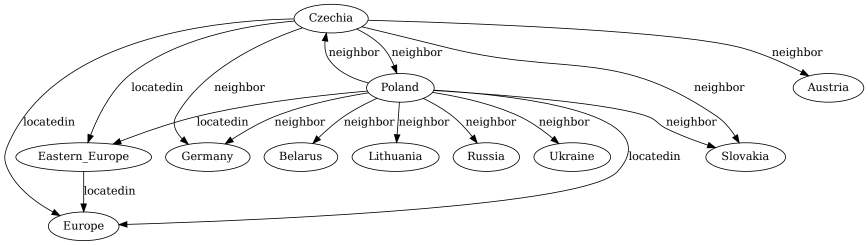

Fig. 2. An excerpt from the “Countries” dataset.

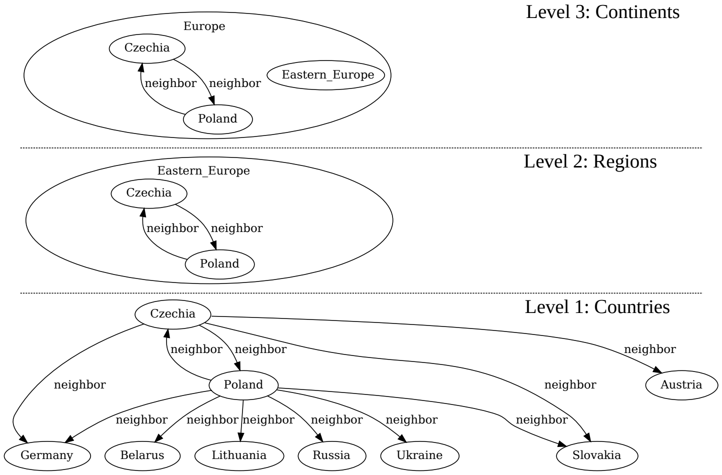

Fig. 3. An excerpt from the “Countries” dataset decomposed into a 3-level hierarchy, the decomposition is based on the “locatedin” label.

3.2 Data preparation

The dataset must be slightly adjusted, as there is a name conflict: *Micronesia* is both a country and a region, and in the original dataset there is a loop edge "Micronesia is located in Micronesia" which causes ontological ambiguity. It is just a mistake in the dataset. In order to fix it, the country of *Micronesia* will be assigned a non-conflicting name.

After converting the graph to a hierarchical form, there are 244 countries, 23 regions, and 5 continents, forming consecutive levels of hierarchy. The set of countries will constitute the initial set X from section 2.1. The "neighbor" relation remains as connections between countries, like the relation $\mathcal{R}$ from section 2.1, and the "locatedin" relation describes the composition of higher-level nodes from lower-level nodes. The initial graph $G(X)$ is generated by the relation $\mathcal{R}$.

3.3 Results

The calculations were performed using the Python programming language, with the *igraph* network analysis library [10] to handle the graphs, using the bliss isomorphism algorithm [16, 17].

As described in chapter 2.1, introducing hierarchy creates additional information which is dependent on two consecutive levels of hierarchy and includes information (a) generated by the number of chosen elements, (b) inner structural information and (c) generated by distinguishability.

Information (a) for the second layer (regions over countries) is 109.9, given by formula 2, choosing a set of 23 elements of higher level over a set of 244. Similarly, for the third layer (continents over regions), it is 17.4, choosing 5 elements over a set of 23.

Information (b) for the second layer is 570.8, as a sum of information given by formula 1 for each subgraph belonging to a higher-level vertex, where the equivalence classes are defined as in [5], equivalent to automorphism groups. Information (b) for the third layer is zero because there are no connections between regions in the dataset. It could be possible to infer missing neighbor relations between regions and continents based on connections of countries that belong to them, but it was decided against it, in order to avoid any unnecessary data transformations and also to prevent introduction of any inferred information which could affect final results.

Information (c) is 102.0 for the second layer and 4.9 for the third layer. It is calculated using formula 1 with vertices grouped into equivalence classes by inner structure distinguishability: two higher-level vertices are indistinguishable if the lower-level subgraphs belonging to them are isomorphic.

The amount of information in the entire graph in the Hellerman sense [14] as defined in [5] is 1956.1, without taking into account hierarchical structures. When interpreting the data set hierarchically, the amount of information in the initial graph is 1424.8, as the graph is now smaller.

The sum of information due to all the levels of hierarchy is 805.0, which when added to information generated by the initial graph constituting the first layer,

DOI: 10.1007/978-3-031-97632-2_4

is 2229.8. This value is greater than the amount of information in the graph in the non-hierarchical interpretation, which shows that the introduction of the hierarchy creates additional information, resulting in an increase of almost 14%. The comparison of the two interpretations can be seen in Fig. 4.

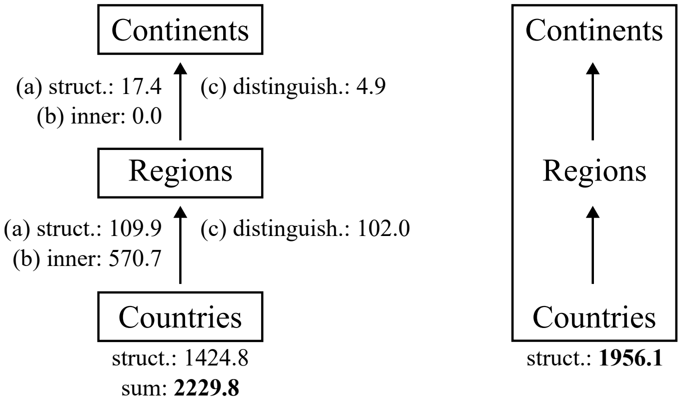

Fig. 4. Comparison of computing the information in the "Countries" dataset with a hierarchical interpretation (on the left) versus interpreting it as a single graph (on the right).

4 Summary

We propose a novel measure for quantifying information in hierarchical graphs which is influenced by existing concepts of graph entropy and especially Hellerman's ontological approach. It accounts for the additional information generated by the hierarchy itself, considering factors such as the number of chosen elements, inner structural information, and distinguishability of higher-level vertices.

To validate the hypothesis that hierarchical graphs contain more structural information, the proposed measure is applied to the "Countries" dataset. When comparing the information content, if treated as a simple graph versus a hierarchical one, the results demonstrate a significant increase in information if the hierarchical structure is taken into account. Identifying a single edge label that forms the hierarchy increases the amount of information by almost 14%. As a result, it supports the argument that recognizing and utilizing the hierarchy enhances the informational richness of graphs. It might provide better metrics for the quantity of information in graphs, indirectly improving the performance of graph-based machine learning models.

Further research focuses on investigating the relationship between the proposed measure and the quality of graph embeddings and the performance of graph neural networks. In particular, the following problems can be put as examples.

1. Will taking into account hierarchical information improve or optimize graph embedding algorithms?
2. There is a GNN with its input being a graph and output a vector. Can the hierarchical information within the input graph be taken into account and does it impact prediction quality?
3. Will including the hierarchical information in graph embedding algorithms pose any opportunities or risks.
4. How to encode the hierarchical information as a GNN input. How does it influence GNN's architecture?
5. Does the hierarchical information concept proposed in this paper optimize hierarchical graph embedding problem?

These are the most immediate goals, but the research can go further, beyond this scope, taking into account social networks, biology, biological systems or autonomous agents, as it was mentioned in the introduction.

References

1. Ali, M., Berrendorf, M., Hoyt, C.T., Vermue, L., Sharifzadeh, S., Tresp, V., Lehmann, J.: PyKEEN 1.0: A Python Library for Training and Evaluating Knowledge Graph Embeddings. Journal of Machine Learning Research **22**(82), 1–6 (2021), http://jmlr.org/papers/v22/20-825.html
2. Amini, M.H., Imteaj, A., Pardalos, P.M.: Interdependent networks: A data science perspective. Patterns **1**(1), 9 pages (2020)
3. Bielecki, A.: The general entity of life: a cybernetic approach. Biological Cybernetics **109**, 401–419 (2015)
4. Bielecki, A.: The systemic concept of contextual truth. Foundations of Science **26**, 807–824 (2020)
5. Bielecki, A., Schmittel, M.: The information encoded in structures: Theory and application to molecular cybernetics. Foundations of Science **27**, 1327–1345 (2022)
6. Bielecki, A., Stocki, R.: The concept of structural information and possible applications. Zagadnienia Filozoficzne w Nauce (Philosophical Problems in Science) **75**, 157–183 (2023)
7. Bouchard, G., Singh, S., Trouillon, T.: On approximate reasoning capabilities of low-rank vector spaces. In: AAAI spring symposia (2015)
8. Bunke, H.: Graph-based tools for data mining and machine learning. Springer (2003)
9. Cai, H., Zheng, V.W., Chang, K.C.C.: A Comprehensive Survey of Graph Embedding: Problems, Techniques and Applications (Feb 2018)
10. Csardi, G., Nepusz, T.: The igraph software package for complex network research. InterJournal **Complex Systems**, 1695 (2006), https://igraph.org
11. Davies, P.: The Demon in the Machine. How Hidden Webs of Information Are Solving the Mystery of Life. University of Chicago Press (2019)
12. Dehmer, M., Mowshowitz, A.: A history of graph entropy measures. Information Sciences **181**, 57–78 (2011)
13. Goyal, P., Huang, D., Goswami, A., Chhetri, S.R., Canedo, A., Ferrara, E.: Benchmarks for Graph Embedding Evaluation (Aug 2019). https://doi.org/10.48550/arXiv.1908.06543

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97632-2_4

14. Hellerman, L.: Representation of living forms. Biological Philosophy **21**, 537–552 (2006)
15. Jain, N., Kalo, J.C., Balke, W.T., Krestel, R.: Do Embeddings Actually Capture Knowledge Graph Semantics? In: Verborgh, R., Hose, K., Paulheim, H., Champin, P.A., Maleshkova, M., Corcho, O., Ristoski, P., Alam, M. (eds.) The Semantic Web. pp. 143–159. Lecture Notes in Computer Science, Springer International Publishing, Cham (2021). https://doi.org/10.1007/978-3-030-77385-4_9
16. Junttila, T., Kaski, P.: Engineering an efficient canonical labeling tool for large and sparse graphs. In: Applegate, D., Brodal, G.S., Panario, D., Sedgewick, R. (eds.) Proceedings of the Ninth Workshop on Algorithm Engineering and Experiments and the Fourth Workshop on Analytic Algorithms and Combinatorics. pp. 135–149. SIAM (2007). https://doi.org/10.1137/1.9781611972870.13
17. Junttila, T., Kaski, P.: Conflict propagation and component recursion for canonical labeling. In: Marchetti-Spaccamela, A., Segal, M. (eds.) Theory and Practice of Algorithms in (Computer) Systems – First International ICST Conference, TAPAS 2011, Rome, Italy, April 18–20, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6595, pp. 151–162. Springer (2011). https://doi.org/10.1007/978-3-642-19754-3_16
18. Khemani, B., Patil, S., Kotecha, K., Tanwar, S.: A review of graph neural networks: Concepts, architectures, techniques, challenges, datasets, applications, and future directions. Journal of Big Data (2024)
19. Konieczny, L., Roterman, I., Spólnik, P.: Systems Biology. Functional Strategies of Living Organisms. Springer, Heidelberg (2023)
20. Li, L., Xiang, J., Chen, B., Heaney, C., Dargaville, S., Pain, C.: Implementing the discontinuous-galerkin finite element method using graph neural networks with application to diffusion equations. Neural Networks **185** (2025)
21. Liquiere, M., Sallantin, J.: Structural machine learning with galois lattice and graphs (1998)
22. Liu, Z., Zhou, J.: Introduction to Graph Neural Networks. Cambridge University Press (2022)
23. Luo, B., Zhang, Z., Wang, Q., He, B.: Multi-chain graphs of graphs: A new approach to analyzing blockchain datasets. In: The Thirty-Eighth Conference on Neural Information Processing Systems (2024)
24. Makaro, I., Kiselev, D., Nikitinsky, N., Subelj, L.: Survey on graph embeddings and their applications to machine learning problems on graphs. PeerJ Computer Science (2021)
25. Mowshowitz, A.: Entropy and the complexity of graphs: I. an index of the relative complexity of the graph. Bulletin of Mathematical Biophysics **30**, 175–204 (1968)
26. Mowshowitz, A.: Entropy and the complexity of graphs: II. the information content of digraphs and infinite graphs. Bulletin of Mathematical Biophysics **30**, 225–240 (1968)
27. Park, J., Han, M., Lee, K., Park, S.: Hierarchical graph attention network with positive and negative attentions for improved interpretability: Isa-pn. Journal of Chemical Information and Modeling (2024)
28. Perraudin, N.: Graph-based structures in data science: fundamental limits and applications to machine learning. Ph.D. thesis, EPFL (2017)
29. Pujara, J., Augustine, E., Getoor, L.: Sparsity and Noise: Where Knowledge Graph Embeddings Fall Short. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. pp. 1751–1756. Association for Computational Linguistics, Copenhagen, Denmark (Sep 2017). https://doi.org/10.18653/v1/D17-1184

30. Rashevsky, N.: Life, information theory, and topology. Bulletin of Mathematical Biophysics **17**, 229–235 (1955)
31. Rosslenbroich, B.: On the Origin of Autonomy. A New Look at the Major Transitions in Evolution. Springer, Heidelberg (2014)
32. Shen, Y., Lin, Z.: Patentgrapher: A plm-gnns hybrid model for comprehensive patent plagiarism detection across full claim texts. IEEE Access (2024)
33. Trucco, E.: A note on the information content of graphs. Bulletin of Mathematical Biophysics **18**, 129–135 (1956)
34. Wang, Q., Mao, Z., Wang, B., Guo, L.: Knowledge Graph Embedding: A Survey of Approaches and Applications. IEEE Transactions on Knowledge and Data Engineering **29**(12), 2724–2743 (Dec 2017). https://doi.org/10.1109/TKDE.2017.2754499
35. Wu, Z., Pan, S., Long, G., Chen, F., Zhang, C.: A comprehensive survey on graph neural networks. IEEE Transactions on Neural Networks and Learning Systems **32**, 4–24 (2020)
36. Xia, F., Sun, K., Yu, S., Aziz, A., Wan, L., Pan, S.: Graph learning: A survey. IEEE Transactions on Artificial Intelligence **2**, 109–127 (2021)
37. Zhang, X., Liang, L., Liu, L., Tang, M.: Graph neural networks and their current applications in bioinformatics. Frontiers in Genetics **12** (2021)
38. Zhang, Y.: Constructing Knowledge Graphs with Language Models and Learning Hierarchies from Graphs Using Probabilistic Topic Modeling. Ph.D. thesis, University of Alberta (2024)
39. Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z.: Graph neural networks: A review of methods and applications. AI Open, Elsevier (2020)

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97632-2_4