© 2025 ADYPJIET, Volume 1(1), Online ISSN: 3049-1517, March 2025, pp. 16-34

Ajeenkya DY Patil Journal of Innovation in Engineering & Technology

Journal Homepage: https://www.adypsoe.in/adypjiet

# A Comprehensive Survey about Applications of Graph Theory in Computer Science and Social Networks

**Satish P. Hande[1], Geeta Kameri [2], Vijay Teli [3]**

[1]KLS, Vishwanathrao Deshpande Institute of Technology, Haliyal, Karnataka, India,
[2]Ajeenkya DY Patil School of Engineering Lohegaon, Pune, Maharashtra, India
[2]Research Scholar KLS, Vishwanathrao Deshpande Institute of Technology, Haliyal, Karnataka, India,
[3]Bangur nagar Arts, Science and Commerce College, Dandeli, Karnataka, India,

***Article History:***
***Received:*** *06-03-2025*
***Revised:*** *02-04-2025*
***Accepted:****5-04-2025*

**Abstract:**

Graph Theory (GT) plays a significant role in various areas of Computer Science (CS), offering applications such as web document clustering, cryptography, and algorithm execution analysis. Additionally, GT is valuable in the simplification and analysis of electronic circuits. In recent years, graphs have been widely utilized in Social Networks (SNs) for modelling and analysing network structures, understanding user interactions, and optimizing network operations. Given the extensive use of GT in SNs, this paper comprehensively reviews its applications in this domain. The objectives of this survey are twofold: first, to outline key applications of GT in CS with practical examples, and second, to explore its relevance in SNs with sufficient concepts and illustrations, highlighting the importance of graph-based modelling and analysis in social network research.

**Keywords:** Graph theory; clustering; social networks; social network analysis; cryptography

## 1.0Introduction

The definition of a graph is a statistical abstraction of conditions of this type. A diagram with multiple points and lines joining multiple pairs of these points can be easily represented for a variety of real-world contexts. For instance, the points could depict individuals with lines joining couples with friends, or the points could be contact centers with lines showing joning connections. It should be noted that the main concern in such diagrams is whether a line connects two defined points or not; the manner in which they are joined is irrelevant.

A statistical abstraction of these needs is called a graph. The concepts of graph theory are widely used in many different academic fields and perform various applications [1]. This entails researching chemicals, creating researching atoms and chemical bonding. For example, graph theory in sociology is utilized to determine an actor's level of popularity or to look at diffusion mechanisms. With edges denoting migratory or transit routes between regions, a vertex represents the regions where a particular species is located in graph theory's application to biodiversity and conservation. This data is essential for analyzing parasite and disease breeding patterns and investigating how migration affects other animals. This information is crucial. Concepts from graph theory are widely applied in computer science [1]. Algorithms like Bellman Ford, the Dijkstra algorithm, Minimum Trees, the Algorithm of Kruskal, Topological Sort, Breadth First Search, and Depth First Search are all used in graph theory.

## 2.0 History of Graph Theory

The Königsberg bridge problem, proposed in 1735, played a crucial role in the development of graph theory. This problem led to the Eulerian graph principle, formulated by Leonhard Euler as a framework for solving network-based challenges. In 1840, A.F. Möbius introduced the concepts of total graphs and bipartite graphs, which Kuratowski later demonstrated as planar structures of theoretical significance [2].

Gustav Kirchhoff, in 1845, introduced the concept of a connected graph as part of the tree principle, excluding cycles, and applied it to electrical circuits by using graphical techniques to measure current flow. The well-known four-color problem was first proposed by Thomas Guthrie in 1852 and later explored by P. Kirkman and William Hamilton in 1856, leading to the study of polyhedral cycles and the development of Hamiltonian graphs [2].

H. Dudeney, in 1913, examined puzzle-related problems involving graph theory. The four-color theorem, which remained unsolved for nearly a century, was eventually proven by Wolfgang Haken and Kenneth Appel using computational methods [2]. Around the same period, Arthur Cayley contributed to graph theory by analyzing tree structures through differential calculus, influencing theoretical chemistry and leading to the advancement of enumerative graph theory [2].

Sylvester further expanded the field by introducing the term "graph" and drawing connections between algebra and molecular diagrams, likening covariants to quantum invariants [3]. In 1941, Ramsey’s experiments on color patterns led to the development of Ramsey theory, a significant branch of graph science. Later, in 1969, Heinrich used computational techniques to tackle the four-color problem. Additionally, studies on asymptotic graph connectivity contributed to the formation of principles in random graph theory [2].

## 3.0 Applications of Graph Theory

Graph theory concepts are widely applied across multiple disciplines for modeling and analysis. These applications include studying molecular structures, understanding chemical bonding, and analyzing complex compounds. In sociology, graph theory is used to assess an individual's influence within a network or to examine diffusion processes. In the fields of biology and conservation, it plays a crucial role in mapping animal habitats, where vertices represent specific regions and edges indicate movement across geographical areas. This data is essential for studying breeding patterns, monitoring the spread of parasites and diseases, and evaluating the impact of migration on wildlife populations [4,5].

Scientific research and operations frequently rely on theoretical graph concepts. For instance, the traveling salesperson problem demonstrates how finding the shortest path in a weighted graph helps optimize tasks such as workforce scheduling and route planning. Graph theory is also integral to game theory, operational networks, and transportation systems, providing efficient solutions for logistics and network modeling [6].

Directed graphs (digraphs) are particularly useful in game theory, where vertices represent possible moves and edges indicate transitions between different states. The principles of graph theory are extensively applied in technology and computational research. Algorithms such as Dijkstra's, Prim's, and Kruskal's are widely used in computer graphics for pathfinding and network optimization.

Graph structures are also fundamental in data processing and communication networks. They help model contact networks, organize structured data, and optimize workflows. Graph transformation techniques enable modifications according to predefined rules, ensuring effective data management. Graph databases provide structured, secure, and continuous data storage while supporting efficient querying mechanisms.

A key application of graph theory is in navigation systems, such as Google Maps, where locations are represented as vertices and roads as edges. By analyzing graph structures, the system determines the shortest path between two points, facilitating efficient route planning. These applications highlight the importance of graph-based modeling in solving real-world problems across various domains.

### 3.1 Uses of Graph Theory in Algorithms

In Computer Science (CS), algorithms play a crucial role in developing and improving applications. Typically, software developers first create a detailed design before starting development, ensuring the final application functions smoothly and efficiently. Graph Theory (GT) is especially important in algorithm design, as many algorithms rely on graph-based structures to solve real-world problems.

**Some key algorithms that utilize graphs include:**

1. Depth First Search (DFS) and Breadth First Search (BFS): Used in data structures for searching nodes in both directed and undirected graphs.

2. Minimum Spanning Tree (MST) algorithms: Essential for optimizing network connections with minimal total weight.
3. Shortest Path Algorithms: Help find the most efficient routes in networks, such as Dijkstra's and Bellman-Ford algorithms.
4. Graph Planarity Algorithm: Determines whether a graph can be drawn on a plane without overlapping edges.
5. Data Transfer Representation: Graphs are widely used in modeling and optimizing data flow in complex applications.

Many programming languages support graph-related computations, allowing developers to implement and manipulate GT concepts efficiently. One such language is GIRL (Graph Information Retrieval Language), which is specifically designed for processing graph-based data.

### 3.1.1 GIRL (Graph Information Retrieval Language)

Graph Information Retrieval Language (GIRL) is a specialized programming language developed to streamline information retrieval, insertion, and deletion (IRD) operations within graph structures. It plays a vital role in three primary areas: (1) generating syntactic and semantic networks, (2) identifying syntactic and semantic relationships, and (3) executing dynamic IRD operations on graph-based data. A practical application of GIRL is demonstrated in Figure 1, where it is used to retrieve relevant information about an animal's origin.

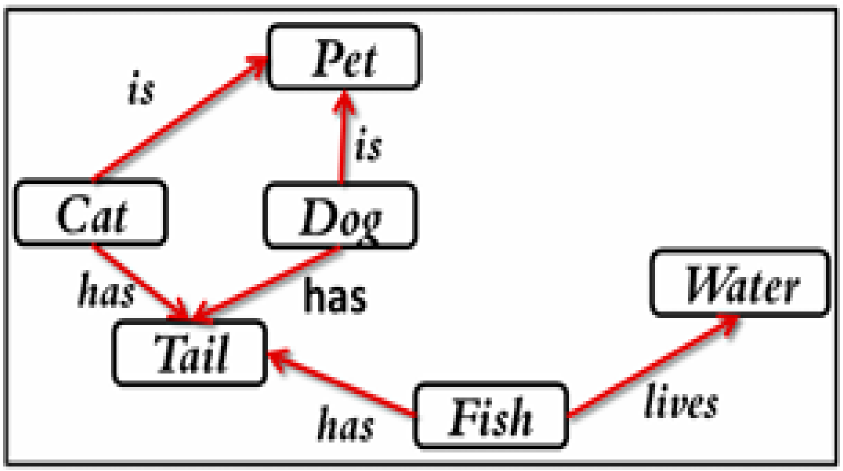



**Figure 1**: Semantic network retrieving animal source information

### 3.1.2 GASP (Graph Algorithm Software package)

GASP is a collection of algorithms designed for efficiently processing and analyzing graph-based data. It offers a comprehensive set of tools to solve various graph-related problems, including shortest path computation, minimum spanning tree construction, network flow optimization, and graph traversal. GASP plays a crucial role in both research and practical applications, such as network analysis, circuit design, and data structure optimization. By providing optimized implementations of fundamental graph algorithms, it enables developers and researchers to handle complex graph computations with ease.

As a software library, GASP is specifically designed to solve graph-related problems efficiently. It includes essential algorithms for pathfinding, spanning trees, flow optimization, and traversal, making it a valuable resource for applications in transportation networks, communications, and system optimization. For instance, to determine the shortest path between two cities in a transportation network, GASP can implement Dijkstra's Algorithm to identify the most efficient route.

**Steps involved**

1. Represent the cities as nodes and roads as weighted edges in a graph.
2. Use GASP's built-in shortest path function.
3. Retrieve the shortest route and its total distance.

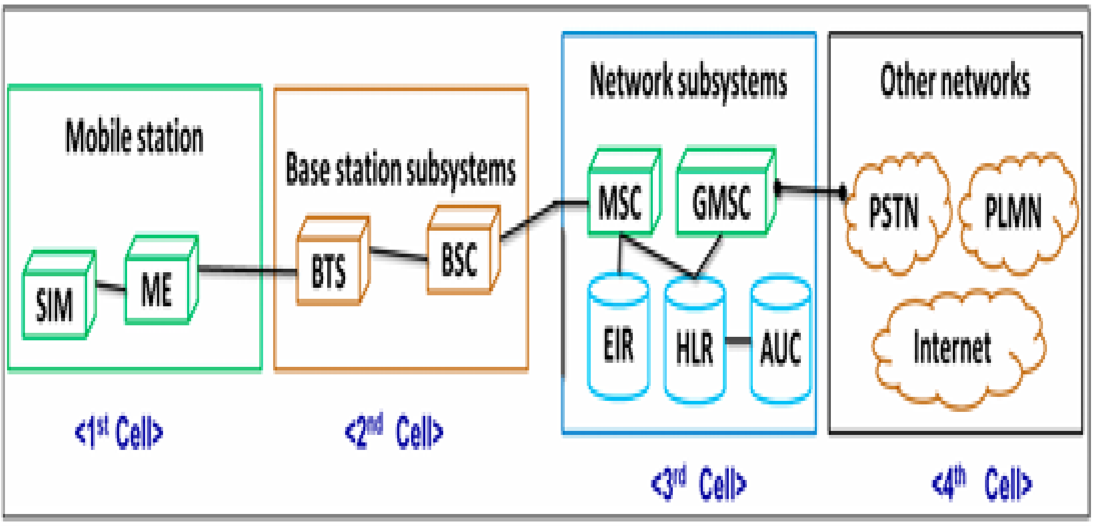
### 3.1.3. GTPL (Graph Theoretic Programming Language)

A specific programming language called Graph Theoretic Programming Language (GTPL) was created to effectively manage computations involving graphs. With the help of built-in functions and structures designed for graph theory applications, users may define, work with, and examine graphs. In domains like network modeling, database systems, artificial intelligence, and optimization issues, GTPL is especially helpful.

## 3.2 Group Special Mobile Networks and Maps Coloring using Graphs

Mobile phones operate within a geographical network known as Group Special Mobile (GSM). This network is divided into various-sized cells, typically represented as hexagonal regions, with each cell connected to a dedicated communication tower. Mobile phones within a specific cell establish connections through the corresponding tower, and the GSM network ensures seamless communication by identifying and linking neighbouring cells.

GSM networks operate across four distinct frequency bands. This allows cellular areas to be represented as a graph using a four-color mapping system. By applying vertex coloring techniques, each GSM network can be assigned four unique frequencies, ensuring efficient network management and minimizing interference. This approach not only simplifies network operations but also enhances frequency allocation based on user demands.




### 3.3 Uses of Graph Algorithms in Network Security Monitoring

Concepts from Graph Theory (GT) can be used in network security to simulate offline and real-time network attacks. Consider a graph G with n vertices, for example, where we must find a vertex of at most size k. Finding a minimum vertex cover is the goal, with the edges representing the connections between the chosen vertices, which serve as routing servers. This method is further explored in References [7, 8]. After establishing the network structure, it is essential to create a defense mechanism to stop network threats by concentrating on particular vital vertices and to find a solution for worm circulation. In this sense, if every vertex in a graph G appears in at least one edge in an edge set E, the graph is said to be covered. Figure 3 provides an example showing a fully covered graph with the required vertices.

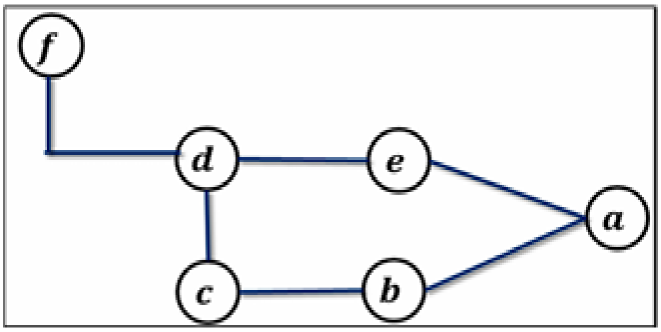



**Figure 3**: Vertex set V = {b, d, e} covers all nodes in graph G.

Additionally, graphs can be utilized for real-time threat analysis, enabling security analysts to implement appropriate protective measures [9]. They also provide a visual representation of failed access attempts on a system, helping to identify potential security vulnerabilities. Figure 4 illustrates this concept.

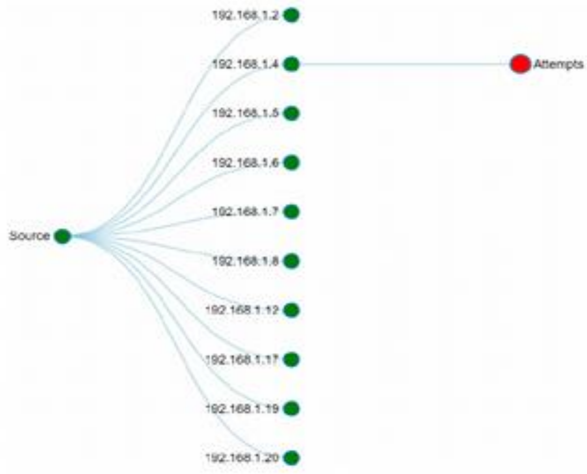



Figure 4: Login failures attempt analysis on a specific system using graphs.

### 3.4 Web Documents Clustering Using Graph Theory Concepts

Search engines can organize web documents into clusters, a technique known as web document clustering. This method involves grouping related types of online documents into a single class or server [10,11]. By storing related web pages together in a cluster, the efficiency of queries is improved, leading to more relevant results [12]. Clustering is commonly used in web document collections [16] and plays a key role in information retrieval processes [13–15]. Figure 5 illustrates the concept of online document clustering.

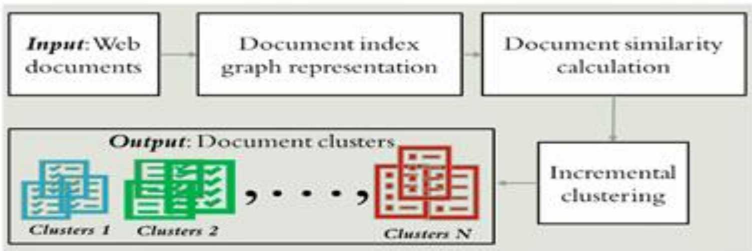



**Figure 5** Conceptual overview of web documents clustering.

In the exclusive k-means clustering algorithm, each data point is assigned to a single cluster. Every cluster is represented by its mean point, also known as the centroid or cluster center. The implementation of the k-means algorithm involves four main steps:

1. Divide the documents into k non-empty subsets.
2. Determine the mean point (centroid or seed point) of each cluster.

© 2025 ADYPJIET, Volume 1(1), Online ISSN: 3049-1517, March 2025, pp. 16-34

3. Assign each document to the nearest cluster's seed point.
4. Return to step 2 and repeat until no document changes clusters.

Figure 6 illustrates an example of the k-means clustering process in action.

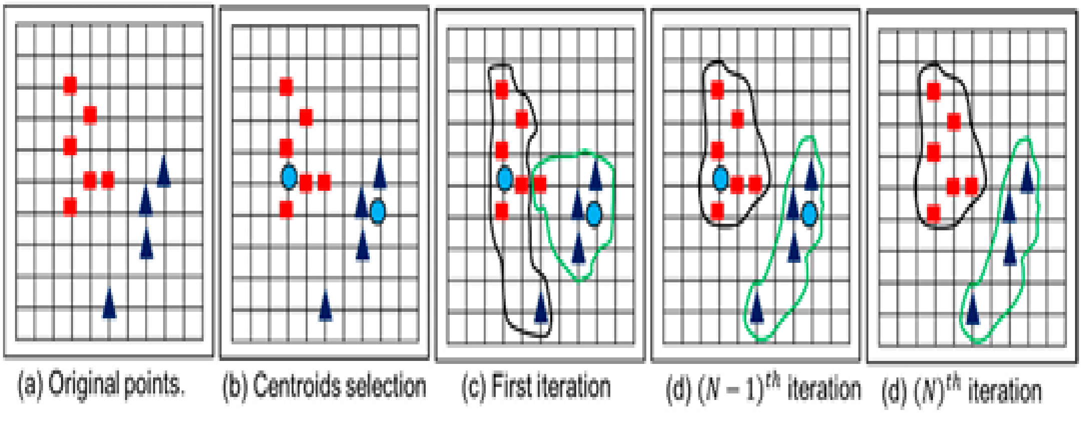



**Figure 6**. The formulation of different coloring options used in k-mean clustering algorithm.

The k-means clustering algorithm has various applications, including:

1. Modeling customer characteristics for target marketing and campaigns.
2. Segmenting customer needs and preferences.
3. Image compression to reduce space complexity in computer vision applications.
4. Network security monitoring.
5. Social Network Analysis (SNA) and data mining.

Another clustering method is the median graph, which is an undirected graph used for clustering. In this method, any three vertices, say x, y, and z, have a unique median. The median is a vertex (x, y, z) that is connected to the shortest path between any two of these vertices (x, y, and z). Figure 7 shows the median graph formed by three vertices.

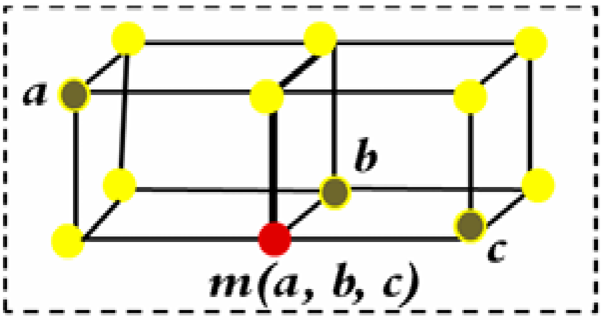



**Figure 7**. The median of three vertices in a median graph.

There are two key advantages to representing web documents as graphs:

1. It leverages the inherent structure of the original web documents, rather than relying on vectors that represent their frequencies.
2. It eliminates the need to develop a completely new clustering algorithm for every case.

Figure 8 provides an example of the graphical representation of web documents, where the entities are represented as vertices, and the connections between them are shown as edges.

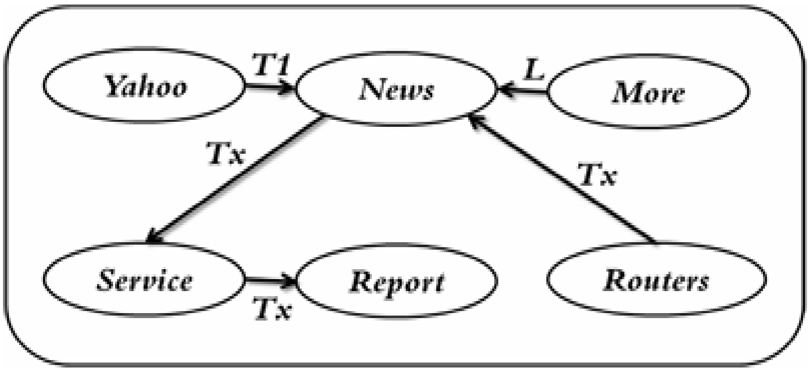



**Figure 8**. Graphical representation of web document clustering.

### 3.5 Graph theory's applications in operational research problems

Graph theory plays a crucial role in solving real-world problems, especially in operational research (OR). It helps businesses and organizations make smarter decisions by modeling complex scenarios as networks of connections.
Here's how it comes in handy in different areas:

1. **Optimizing Routes and Networks**

- **Finding the Best Route:** Whether it's Google Maps helping you find the fastest way home or a delivery company planning efficient routes, graph theory powers these decisions using shortest-path algorithms.
- **Building Efficient Networks:** Companies like telecom providers use graph models to lay down cables and cell towers in a way that ensures maximum coverage with minimal cost.

2. **Smart Scheduling and Planning**

- **Project Management:** Ever heard of the Critical Path Method (CPM)? It helps project managers figure out which tasks are most crucial to finish a project on time.
- **Timetabling Made Easy:** Universities use graph colouring techniques to schedule exams without conflicts—ensuring students don't have two tests at the same time.

3. **Making Transportation and Logistics Smoother**

- **Optimizing Deliveries:** Delivery services like Amazon and FedEx use graph-based algorithms to determine the best routes for their trucks, cutting down on costs and delays.
- **Traffic Management:** City planners analyze traffic patterns with graphs to reduce congestion and improve road networks.

3. **Choosing the Best Locations for Facilities**

- **Where Should a New Store Go?** Big retailers use graph models to decide where to open new branches, ensuring they serve the maximum number of customers efficiently.
- **Placing Wi-Fi Hotspots:** Companies use the concept of "dominating sets" in graphs to decide where to place Wi-Fi routers for optimal coverage.

5. **Helping Businesses Make Better Decisions**

- **Strategic Planning:** Businesses use graph-based models to analyze competition, predict customer behavior, and optimize pricing strategies.
- **Problem-Solving with AI:** Many modern AI techniques rely on graph-based heuristics to find the best solutions in complex scenarios.

6. **Managing Supply Chains and Inventory**

- **Efficient Warehousing:** Companies use graph theory to figure out the best way to store and distribute goods efficiently.
- **Balancing Supply and Demand:** Flow networks help businesses decide how much stock to keep and where to distribute it.

7. **Understanding Social Influence and Detecting Fraud**

- **Marketing and Social Media:** Platforms like Facebook and Twitter analyse user connections (graph structures) to suggest friends or target ads effectively.
- **Spotting Fraud:** Banks and cybersecurity firms use graph algorithms to detect suspicious transactions and prevent fraud.

### 3.6: Use of Graph Coloring

Graph coloring might sound like an abstract math concept, but it's actually a powerful tool that helps solve real-world problems in computer science. It's all about assigning "colors" (or labels) to things in a way that avoids conflicts. Let's look at some of the ways this concept is used in technology and everyday computing:

© 2025 ADYPJIET, Volume 1(1), Online ISSN: 3049-1517, March 2025, pp. 16-34

**1. Smarter Task Scheduling**

- Imagine you're trying to schedule exams at a university. You can't have two exams for the same student at the same time, right? Graph coloring helps assign time slots so there are no conflicts.
- Operating systems use a similar trick to schedule processes, making sure tasks don't compete for the same CPU resources.

**2. Making Computer Programs Run Faster (Register Allocation)**

- Computers have a limited number of registers (tiny storage spaces inside the CPU). Graph coloring helps compilers assign these registers efficiently to avoid slowdowns.
- It's like organizing a small closet—you need to fit everything in neatly without overlapping!

**3. Coloring Maps and Recognizing Images**

- When you look at a map, neighbouring regions should have different colors so you can tell them apart. Graph coloring helps achieve this.
- In image processing, it helps segment an image into different objects—for example, distinguishing people from the background in a photo.

**4. Avoiding Wi-Fi and Mobile Signal Clashes**

- Ever wondered why your Wi-Fi doesn't interfere too much with your neighbours'? That's because different Wi-Fi channels (or colors) are assigned to avoid signal overlap.
- Mobile networks also use this technique to ensure nearby cell towers don't broadcast on the same frequency, preventing dropped calls.

**5. Solving Puzzles Like Sudoku**

- Sudoku is basically a graph coloring problem! Each number is a "color," and the game's rules ensure no two connected cells have the same number.
- Puzzle games and AI systems use similar logic to find the best solutions.

**6. Speeding Up Computers (Parallel Processing)**

- When computers run big programs, they break them into smaller tasks that can run at the same time. Graph coloring helps make sure tasks that depend on each other don't run simultaneously, preventing crashes and delays.

**7. Finding Patterns in Social Networks**

- Ever wonder how Facebook suggests friends? Graph coloring helps group people into communities based on their connections.
- It's also used to analyze and visualize relationships between users, making sense of huge amounts of social data.

### 3.7 Applications of Graph Theory in the Internet of Things (IoT)

With the rapid advancement of Information and Communication Technologies (ICT), the Internet of Things (IoT) has become a key area of research. Graph theory (GT) plays an essential role in several IoT applications, including:

1. Graph-based clustering in two-tier architectures.
2. Risk assessment in IoT environments.
3. Modeling data functional networks and domain functional networks.
4. Analyzing information diffusion in narrowband IoT.
5. Utilizing IoT-generated big data for smart transportation systems.
6. Assessing vulnerabilities in Industrial IoT (IIoT) through attack graphs and maximum flow methods.
7. Integrating information from various sources to mitigate intentional cyber threats.

## 3. Applications of the Graph Theory in Social Networks (SN)

With the rapid advancements in Information and Communication Technology (ICT), the use of social networks (SNs) has grown exponentially [17]. In recent years, SNs have become a widely utilized platform for communication and information sharing, particularly among teenagers. Furthermore, businesses and service providers leverage user data from SNs for various purposes, such as recommending trending brands and running targeted marketing campaigns [18-20].

Analytics firms heavily rely on SN data to study social trends, understand user attitudes toward brands, conduct intent mining, perform sentiment analysis, and create personality profiles. Extensive research has been conducted on the concepts, applications, and technological advancements of SNs [21].

Social networks function as complex networks that can be represented as a graph G(U,V), where U represents the users or entities, and V represents the edges that define the relationships between them [22]. These relationships may include friendships, romantic connections, family ties, or sibling bonds, as seen on platforms like Facebook. A graphical representation of SN users typically consists of vertices (users) and edges (relationships), which can be either directed or undirected [23,24]. These edges reflect various factors such as similarity, trust, communication cost, interaction frequency, and influence between users in the network.

© 2025 ADYPJIET, Volume 1(1), Online ISSN: 3049-1517, March 2025, pp. 16-34

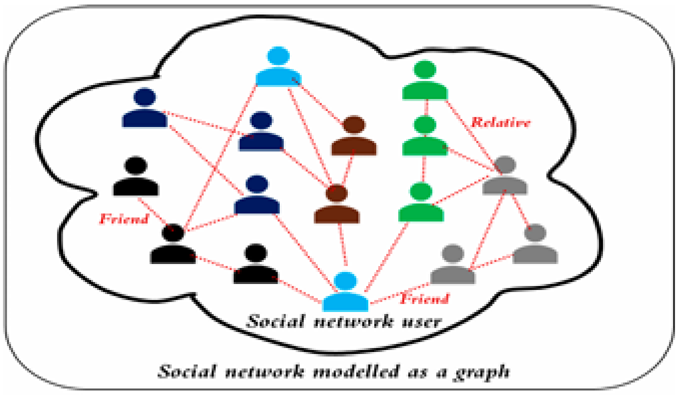
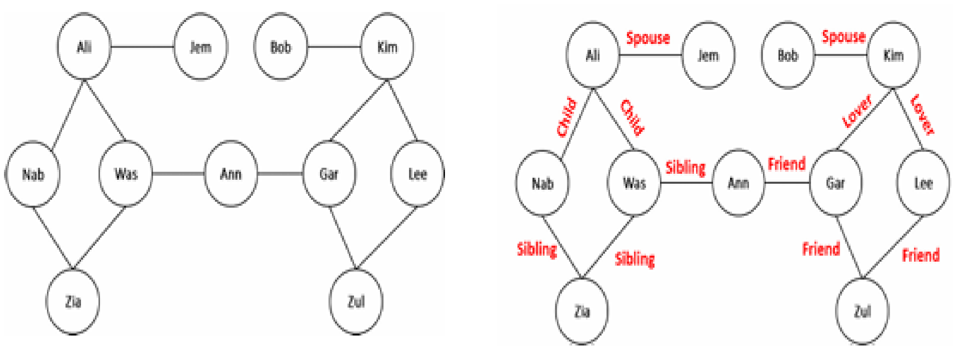



**Figure 9**. Social network (SN) users modelled as a graph G.

In a social network (SN), users and their connections can be represented as a graph, where the nodes (or vertices) are the users, and the edges (or links) represent their relationships, such as friendship, trust, or similarity. These graphs can be labelled in two ways:

- **Vertex-labelled graphs**: Here, the nodes (users) are labelled with their names or user IDs, while the edges indicate relationships between them. For example, in a Facebook-like network, each node represents a person, and an edge between two nodes signifies a friendship.
- **Edge-labelled graphs**: In this type of graph, the edges carry labels that describe the nature of the relationship between users. For instance, users in a network might have multiple types of connections, such as being friends, colleagues, family members, or sharing common interests.

Depending on the application, these connections can be **directed** (one-way, like a "follow" on Twitter) or **undirected** (mutual, like a Facebook friendship). Understanding these structures helps in analyzing user interactions, trust levels, and influence within a social network.



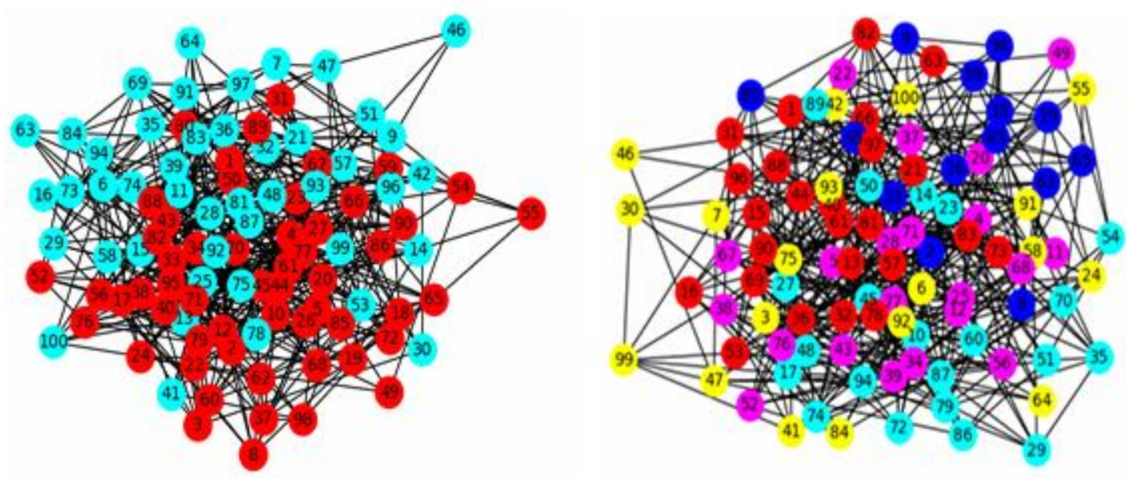

(**a**) Vertex-labelled graph (**b**) Edges-labelled graph

© 2025 ADYPJIET, Volume 1(1), Online ISSN: 3049-1517, March 2025, pp. 16-34

**Figure 10**. Example of the labelled-edges and -vertices graph of SN users.

**4.1** Use of Graphs for Community Clustering in a Social Networks

A community is a collection of individuals or users who share certain characteristics, interests, pastimes, locality, preferences, and/or backgrounds [25]. Target advertising, cooperative filtering, group recommendation, and interest are just a few benefits of community detection in an SN. Personalized advertising, opinion leader selection, information diffusion, information based marketing, information contagion, and speeding up the dissemination of information. Figure 11. provides an example of a community with 100 users. The users are divided into five and two communities, respectively.



(**a**) Two communities in a graph

(**b**) Five communities in a graph

**Figure 11**. Examples of the detected communities from a graph of 100 nodes in a SN.

Figure 12. shows a relatively complex example of clustering taken from Tabrizi et al. [26]. Vertices with identical colors correspond to the clusters.

**Figure12**. Visualization of the clusters found by the clustering algorithm Ref. [26]

There are several methods for identifying communities within social networks (SNs). For example, a node may be assigned to a community if its attribute values are similar to those of other users. Additionally, users can be grouped into different communities based on their preferences, behaviors, interests, and other characteristics. A community can be viewed as a group of users who share common attitudes or behaviors related to a specific topic or event. Furthermore, user communities can be formed based on activities within online social networks (OSNs), such as posting content related to a specific subject or leaving comments on relevant topics.

### 4.2 Users' Influence/Trust Score Representation in a Social Networks Via Graphs

1. **Influence Score:** This score quantifies a user's ability to affect others' perceptions or behaviors. It can be computed using metrics like degree centrality, betweenness centrality, or PageRank, which evaluate a user's connections and their connectivity within the network.
2. **Trust Score:** Trust is often modeled based on the quality of relationships. Metrics such as trust propagation algorithms determine how trustworthiness spreads through the network, factoring in user interactions and feedback.
3. **Visualization**: Graphs allow for visual representation of users' influences and trust scores, making it easier to identify key influencers and potential trust flaws. Various layout algorithms (e.g., force-directed layouts) can represent these complexities more intuitively.
4. **Applications:** Businesses leverage influence and trust scores for targeted marketing, recommendations, and identifying brand advocates. In contrast, communities may assess trust to improve collaboration and mitigate misinformation.

   By analyzing graphs representing users' influence and trust, stakeholders can make informed decisions to enhance engagement, foster relationships, and promote overall network health.

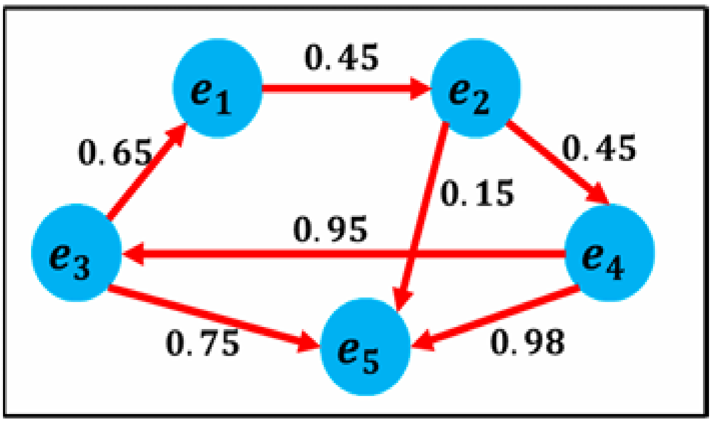


**Figure.13**. An example of the experts' influence network (Ref.27)

### 4.3 Analyzing the Modularity in a Social Network Users' Graph

In many situations, we can break down a network into several communities (or subgraphs). To evaluate how well this clustering represents the overall social network (SN) structure, we can use a measure called the modularity score. A **higher modularity score** indicates that the network is effectively divided into cohesive subgroups, meaning the connections within these communities are strong and meaningful. Conversely, a **lower score** suggests that these clusters might be random and less insightful, which can lead to misguided recommendations or analyses. It's generally preferred to partition the network in a way that reflects strong correlations within the subgraph. In simpler terms, modularity assesses how good the clustering is. The accompanying figures illustrate examples of both high and low modularity scores. With the growing use of social networks in recent years' researchers have delved deep into the concept of modularity, making it an important area of study in the field [28]

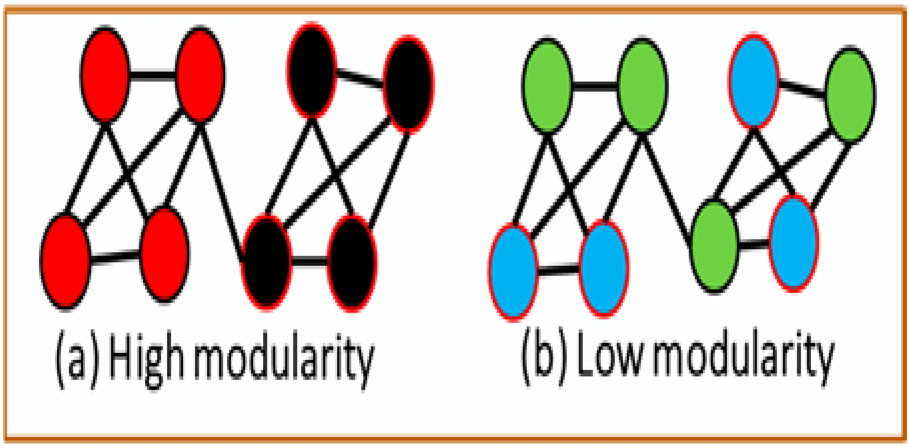



**Figure 14.** Overview of the high and low modularity in a graph (Ref. [28]).

### 4.4 Topic of Interest Modelling Using Graphs in Social Networks

Graphs are a powerful tool for modelling interests, especially with the rising popularity of social networks (SNs). Topic modelling has been increasingly utilized to extract and organize complex information. Essentially, topic modelling is a method for identifying groups of related information, such as themes or subjects, from a collection of documents—like posts, opinions, or comments from users. In practical terms, especially in micro-blogging, nouns and topics in posts are crucial for understanding and generating content. They capture personal interests effectively. For instance, in the example shown in Figure 15a, five topics extracted from micro-blogs are illustrated, where the vertices represent the topics, and edges denote their relationships. These topics can be categorized based on the type of information they contain, as detailed in Figure 15b. Using topic modelling, we can improve information

© 2025 ADYPJIET, Volume 1(1), Online ISSN: 3049-1517, March 2025, pp. 16-34

retrieval and processing, making it more efficient. Additionally, it can help analyse how users behave on social networks providing insights, for targeted advertisements. Recently, topic modelling has played a key role in the field of sentiment analysis, helping analyse opinions about products, shows, and more, especially with the advancements in natural language processing (NLP). This has opened new avenues for understanding user feedback through machine learning and deep learning techniques

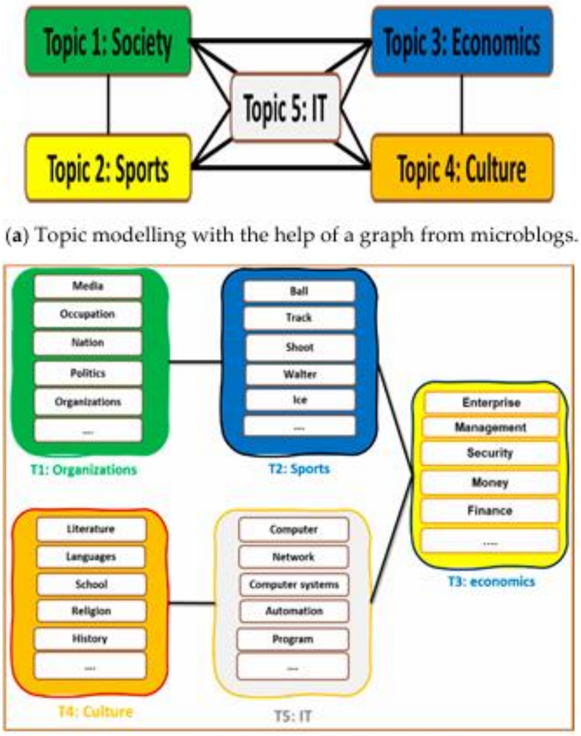



**Figure 15**. Overview of topic modeling and information contained in each topic by leveraging graphs.

## Conclusion:

In this paper, we reviewed various studies in the fields of computer science (CS) and social networks (SN) that apply the principles of graph theory (GT). Our objective is to assist researchers in effectively analyzing graph properties and selecting the most appropriate types of graphs for their specific problems. We highlighted key applications of graph theory, aiming to make its concepts more accessible to beginner researchers and demonstrating its relevance across different domains in CS and SN. Additionally, we provided practical examples and clear explanations to showcase the potential of graph theory, emphasizing its importance in contemporary research. Overall, our goal is to underscore the crucial role that graphs play in advancing research and to encourage further exploration in this area.

**REFERENCES:**

[1] G. Marcialis, F. Roli, Graph Based and Structural Methods for Fingerprint Classification, Springer verlag, Berlin Heidelberg, 9,1, 20018), 1–202.

[2] S.D.R. Zabih, Introduction to the special section on graph algorithms in computer vision, IEEE on pattern analysis, 23(10) (2016), 114–122.

[3] B. Hong Liu, W. Chieh KE, constructing a message pruning tree with minimum cost for tracking moving objects in wireless sensor networks, IEEE, 57(6) (2017), 16–22.

[4] S. Skiena S. Pemmaraju, Implementing Discrete Mathematics-Combinatory and Graph Theory with Mathematica, Addison-Wesley Publishing Company, 3(9) (2019), 1–448.

[5] X. Zhou, T. Nishizeki, Edge-coloring algorithms, Technical report, Graduate School of Information Sciences, Tohoku University, Sendai, 1(3) (2020), 120–142.

[6] A. Vince, C. Haaley, Star chromatic number, Journal of Graph Theory, 12(4) (2020), 551–559.

**[7]** H. Mokhtari, M.A. Dadgar, Flexible Job Shop Scheduling Problem with Controllable Processing Times to Optimize Total Cost of Delay and Processing. Int. J. Supply Oper. Manag. 2015, 2, 871.

[8] H.A. Dawood, R. William, Graph T Theory and Cyber Security. In Proceedings of the 2014 3rd International Conference on Advanced Computer Science Applications and Technologies, Amman, Jordan,29–30 December 2014; pp. 90–96.

[9] A. Majeed, R. Farooq, A. Masoom, A. Nadeem , Near—Miss situation based visual analysis of SIEM rules for real time network security monitoring. J. Ambient. Intell. Humaniz. Comput. 2019, 10, 1509–1526.

[10] A. Schenker, M. Last, H. Bunke, A. Kandel, Chapter? Clustering of Web Documents Using a Graph Model. In Web Document Analysis: Challenges and Opportunities; World Scientific Publishing Company: Singapore, 2003; pp. 3–18.

[11] B. J. Jain, Jain, K. Obermayer, Graph quantization. Comput. Vis. Image Underst. 2011, 115, 946–961.

[12]. A. Kalogeratos, A. Likas, Data & Knowledge Engineering Document clustering using synthetic cluster prototypes. Data Knowl. Eng. 2011, 70, 284–306.

[13]. S. L. Jarvenpaa, P. A. Todd, Consumer reactions to electronic shopping on the World Wide Web. Int. J. Electron. Commer. 1996, 1, 59–88.

[14]. R. Zhao, W.I. Grosky, Narrowing the Semantic Gap—Improved Text-Based Web Document Retrieval Using Visual Features. IEEE Trans. Multimed. 2002, 4, 189–200.

[15]. V. A. Zeithaml, A. Parasuraman, A. Malhotra, Service quality delivery through web sites: a critical review of extant knowledge. J. Acad. Mark. Sci. 2002, 30, 362–375.

[16] A. Schenker, M. Last, M. H. Bunke, A. Kandel, Graph Representations for Web Document Clustering. In Iberian Conference on Pattern Recognition and Image Analysis; Springer: Berlin/Heidelberg, Germany, 2003; pp. 935–942.

[17] R. Jin, H. Zhang, Y. Zhang, The social negative mood index for social networks. In Proceedings of the 2018 IEEE Third International Conference on Data Science in Cyberspace (DSC), Guangzhou, Chin, 18–21 June 2018; pp. 1–5.

[18] N. Kolli, N. Balakrishnan, Analysis of e-mail Communication Using a Social Network Framework for Crisis Detection in an Organization Science Direct. Procedia—Soc. Behav. Sci. 2013, 100, 57–67.

[19] C. He, H. Li, X. Fei, Y. Tang, J. Zhu, A Topic Community-based Method for Friend Recommendation in Online Social Networks via Joint Nonnegative Matrix Factorization. In Proceedings of the 2015 Third International Conference on Advanced Cloud and Big Data, Yangzhou, China, 30 October–1 November 2015; pp. 28–35.

[20] J. Wieringa, P.K. Kannan, X. Ma, T. Reutterer, H. Risselada, B. Skiera, Data analytics in a privacy-concerned world. J. Bus. Res. 2019.

[21]. F. Liu, H. Joo, Expert Systems with Applications Use of social network information to Enhance Collaborative filtering performance. Expert Syst. Appl. 2010, 37, 4772–4778.

[22]. D. Liu, L. Wang, J. Zheng, K. Ning, L. Zhang, Social Network. In Proceeding of the 2013IEEEInternational Conference on Services Computing, Santa Clara, CA, USA, 28 June–3 July 2013; pp. 368–375.

[23] A. Beach, M. Gartrell, R. Han, Social-K: Real-TimeK-AnonymityGuaranteesforSocialNetworkApplications. In Proceedings of the 2010 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), Mannheim, Germany, 29 March–2 April 2010; pp. 600–606.

[24] T.T. Zin, P.Tin, H. Hama, T. Toriu, Knowledge-based Social Network Applications to Disaster Event Analysis. In Proceedings of the International Multi-Conference of Engineers and Computer Scientists, Hong Kong, China, 13–15 March 2013.

[25] Y. Li, H. Hsiao, Y. Lee, Information Sciences Recommending social network applications via social filtering mechanisms. Inf. Sci. 2013, 239, 18–30.

[26] S.A. Tabrizi, A. Shakery, M. Asadpour, M. Abbasi, M.A. Tavallaie, Personalized PageRank Clustering: Agraph clustering algorithm based on random walks. Phys. A Stat. Mech. Appl. 2013, 392, 5772–5785.

[27] Y. Zheng, J. Xu, A trust transitivity model for group decision making in social network with intuitionistic fuzzy information. Filomat, 2018, 32, 1937–1945.

[28] M. E. J. Newman, Modularity and community structure in networks. Proc. Natl. Acad. Sci. USA 2006, 103, 8577–8582.