A Comparative Review of Core Models and Techniques in Discrete Mathematics

Khin Myo Myo Minn[1] and Hlaing Htake Khaung Tin[2]

[1]Faculty of Computing, [2]Faculty of Information Science

[1,2]University of Information Technology Yangon, Myanmar

DOI: https://doi.org/10.51584/IJRIAS.2026.11060085

Received: 16 May 2026; Accepted: 22 May 2026; Published: 24 June 2026

ABSTRACT

The investigation of only finite and countable structures becomes the foundation for many applications, which serve as a base for various computations and analysis in discrete mathematics. The comparative study on four fundamental areas of discrete mathematics – Graph Theory, Combinatorics, Mathematical Logic and Discrete Probability is presented in this paper. It will help to analyze their characteristics, computational and practical aspects of application in the different domains like computer science, artificial intelligence, networking and optimization. All four techniques were considered by providing an overview of the technique and describing the advantages and disadvantages of the approach. Such techniques like Graph Theory were praised for their capability to represent relationships and networks. Also, the exactness of Combinatorics for counting and ordering was mentioned while criticizing their inefficiency because of exponential growth. The possibility of using mathematical logic for reasoning and decision-making systems is discussed. Finally, the capabilities of Discrete Probability to manage uncertainties and perform predictive modeling in data driven systems were analyzed. Comparative Assessment will be developed to consider four techniques by a few selected criteria, such as Computational Complexity and Scalability. Results indicate that there is no ideal approach; all approaches are suitable depending on certain types of problems. Furthermore, the paper identifies contemporary trends in the area, among them being the application of these approaches to problems such as those solved by machine learning models using graphs and probabilistic logic. The current review will contribute towards improving knowledge on discrete mathematics approaches as well as generate ideas regarding which approach to apply for computing challenges.

Keywords: Discrete Mathematics, Graph Theory, Combinatorics, Mathematical Logic, Discrete Probability.

INTRODUCTION

Discrete mathematics is one of the fundamentals of mathematics which studies finite and distinct structures. The key differences between continuous and discrete mathematics are based on the nature of objects involved in their study. While continuous mathematics deals with quantities that can change continuously, discrete mathematics studies structures such as integers, graphs, logical propositions, sets, and combinations of the latter. These are the foundations for current computer architectures and very important for computer science, data analysis, cryptography, artificial intelligence, and operational research.

Modern times have seen growing demand for discrete mathematical tools due to the rapid development of information and digital technologies. Efficient and effective use of discrete mathematics is essential for solving many contemporary issues in networks, decision-making, algorithms and uncertainty models. The key areas of discrete mathematics include graph theory, combinatorics, mathematical logic, and discrete probability. Each of them is a separate theory, as well as an approach to problem-solving. For example, graph theory may be used for modeling relations in the network, combinatorics for counting and sorting items, logic in artificial intelligence to draw conclusions about the object, and probability for decision-making in conditions of uncertainty.

However, each of these theories differs from others in terms of computational costs, scalability, and application possibilities. Knowledge of all these differences is essential when choosing the proper theory to use in solving practical issues. Unfortunately, prior studies have been focused on only one theory at a time, and there has been no analysis of several mathematical theories simultaneously.

That is why the aim of this paper is to conduct a comparative study of some critical approaches to discrete mathematics. There are three objectives in this research: (i) to explore the theoretical principles underlying the fundamental areas of discrete mathematics, (ii) to consider the computational aspects of the areas, and (iii) to estimate the applicability of the areas to solving particular tasks. The objective of this paper is to give a general understanding of the situation in discrete mathematics.

This paper will be organized as follows: Firstly, after introducing basic concepts of discrete mathematics (Section 2), we will discuss methods (Section 3), differences in methodology (Section 4), fields of application (Section 5), and outline some ideas for future research (Section 6).

OVERVIEW OF CORE AREAS IN DISCRETE MATHEMATICS

Discrete mathematics is a collection of integrated fields which offer vital tools to model, analyze and solve computational challenges. This section introduces some of the basic topics of graph theory, as well as combinatorics, mathematical logic, and discrete probability, and their theoretical backgrounds and applications.

Graph Theory

Graphs are used to model pairwise relationships among objects and graph theory is the study of these. Graphs are made up of a set of vertices (also known as nodes) and a set of edges (also known as connections) that relate two vertices. Depending on the nature of the relationships they represent, a graph can be directed or undirected, weighted or unweighted.

Problems involving connectivity, shortest paths and network flows are important and fundamental problems in graph theory. There are many classic algorithms that are known for optimizing networks including Dijkstra's algorithm and Kruskal's algorithm. These applications include communication systems, transportation networks and social network analysis [1][2].

Shortest Path Problem

One of the most widely used graph algorithms is Dijkstra’s shortest path algorithm, which determines the minimum distance between nodes in a weighted graph.

$$d(v) = \min\,(d(v), d(u) + w(u,v)) \tag{1}$$

where:

- $d(v)$represents the shortest distance to vertex v
- $w(u,v)$represents the edge weight between vertices

This algorithm is widely applied to:

- GPS navigation systems
- Internet routing protocols
- Transportation optimization
- Social network analysis

The computational complexity of Dijkstra's algorithm is:

$$O(E\log V) \tag{2}$$

when implemented using a priority queue.

Combinatorics

Combinatorics is the study of discrete objects in terms of the way that they can be counted, arranged and selected. It contains essentials like Permutation, Combination and Pigeonhole principle. Combinatorial methods play important role in solving problems with discrete structures and finite configurations.

$$n! = n \times (n-1) \times (n-2) \times \cdots \times 1 \tag{3}$$

Even though combinatorics is very helpful, it often faces the problem of combinatorial explosion due to its exponential complexity. Combinatorics has wide applicability in fields such as cryptography, scheduling algorithms, and even algorithm design [1][3].

Mathematical Logic

Logic is a tool used in making decisions and reasoning. The three branches of logic are propositional logic, predicate logic, and Boolean algebra. These techniques are applied in the representation of knowledge and verifying their validity. Computer programming languages, database querying languages, and even artificial intelligence are all founded upon logical principles. Several approaches may be used to evaluate logical expressions, such as truth tables, inference rules, and logical equivalences [1][4].

Discrete Probability

Countably finite experiments are termed as discrete probability experiments. This provides a means for quantifying and analyzing random events using discrete methods. Concepts that are worth noting include probability mass function, expected values, and conditional probability.

$$P(A) = \frac{\text{Number of favorable outcomes}}{\text{Total number of outcomes}} \tag{4}$$

Discrete probability finds application in several areas such as data science, machine learning, and risk management. Discrete probabilities may be used to aid in decision-making because they assist in assessing uncertainty and predicting the future using the existing data [5].

The four disciplines stated above constitute the basic foundations of discrete mathematics. Graphs involve the study of structure, combinatorics involves counting, logic involves inference, and probability involves uncertainty. It is the connection between the modules of these disciplines that enables the creation of algorithms capable of solving difficult real-life problems.

COMPARATIVE ANALYSIS OF DISCRETE MATHEMATICS TECHNIQUES

This chapter presents a comparative study of four key areas within discrete sciences including Graph Theory, Combinatorics, Discrete Probability, and Mathematical Logic. The comparisons have been drawn by focusing on certain key elements including scalability, flexibility, applicability, and computational complexity among others.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue VI June 2026

Theoretical comparison is done based on the theoretical characteristics.

Discrete Mathematics domains have different theoretical bases and hence have different problem-solving approaches.

- Graph Theory is about structuring and relations between entities. Very visual and intuitive; useful for modeling networks and connections.
- Counting and arrangement are the core concepts of combinatorics, which provide precise mathematical solutions, but can become quickly complex.
- Mathematical Logic focuses on formal reasoning and symbolic representation, which is used for a rigorous framework to validate correctness and consistency.
- Discrete Probability adds uncertainty to a discrete system, allowing for modelling of random events and probabilistic outcomes.

Computational Complexity Analysis

The most important part of comparing is how fast the algorithm runs. The shortest path and minimum spanning tree problems are examples of graph algorithms that are generally of polynomial time complexity but may be costly for dense graphs. Because of the many different configurations, combinatorial problems can have an exponential time complexity. Logical operations tend to be relatively cheap to compute but may take time in the case of complex logical inference systems. All discrete probability computations are typically manageable but can become intensive if one has large sample spaces or conditional dependencies.

Scalability and Performance

Scalability determines how well a technique performs as the problem size increases.

Table 1. Performance Comparison

Graph Theory	Scales effectively with optimized algorithms, though very large networks may require approximation methods.
Combinatorics	Limited scalability due to combinatorial explosion.
Logic	Scales well in structured systems, particularly in rule-based and knowledge-based applications.
Probability	Highly scalable when supported by statistical and computational tools.

Flexibility and Adaptability

Flexibility refers to the ability of a method to adapt to different problem domains.

- Graph theory is highly adaptable for modeling diverse systems such as biological networks, communication systems, and transportation.
- Combinatorics is less flexible, as it is primarily suited for structured counting problems.
- Logic provides precision but can be rigid when dealing with uncertain or incomplete information.
- Probability is highly flexible, allowing integration with data-driven and machine learning approaches.

Table 2. Comprehensive Comparison of Discrete Mathematics Techniques

Criteria	**Graph Theory**	**Combinatorics**	**Mathematical Logic**	**Discrete Probability**
Primary Focus	Relationships & structures	Counting & arrangements	Reasoning & inference	Uncertainty & randomness
Mathematical Nature	Structural	Enumerative	Symbolic	Statistical
Problem Type	Network-based problems	Finite configurations	Decision & verification	Predictive modeling

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue VI June 2026

Computational Complexity	Moderate to High	High (often exponential)	Moderate	Moderate
Scalability	Good with optimized algorithms	Limited due to explosion	Good in structured systems	Very good with data support
Flexibility	High	Moderate	Low–Moderate	High
Deterministic/ Probabilistic	Deterministic	Deterministic	Deterministic	Probabilistic
Typical Methods	BFS, DFS, shortest path	Permutations, combinations	Truth tables, inference rules	PMF, expectation, Bayes rule
Strengths	Visual modeling, real-world networks	Exact solutions	Formal correctness	Handles uncertainty
Limitations	Large graph complexity	Combinatorial explosion	Rigid structure	Data dependency
Real-world Applications	Social networks, routing	Cryptography, scheduling	AI reasoning, databases	Machine learning, risk analysis
Integration Capability	High	Moderate	High	Very high

From comparative analysis, one can easily conclude that there is a difference between each method in terms of the properties. In this section, we are going to compare four fields of discrete science: graph theory, combinatorics, mathematical logic, and discrete probability. There are some features that we will consider such as scalability, flexibility, problem-solving abilities, and complexity.

As an illustration, graph theory fits well in the description of complex network systems, whereas combinatorics may help in finding an accurate answer to counting problems. At the same time, both techniques are not efficient in case the problem is related to scalability issues. Speaking of mathematical logic, this method is vital in formal deductions and system verifications (including artificial intelligence). Moreover, one cannot underestimate the role of discrete probability in case of prediction uncertainties. Nevertheless, it should be noted that all these methods may be considered complementary, which means that solving a problem usually involves several techniques.

Table 3. Algorithmic and Computational Comparison

Technique	**Example Algorithm**	**Time Complexity**	**Space Complexity**	**Real-World Application**
Graph Theory	Dijkstra’s Algorithm	$O(V^2)$ or $O(E \log V)$	$O(V)$	GPS routing, network optimization
Combinatorics	Backtracking Algorithm	$O(2^n)$	$O(n)$	Scheduling, cryptography
Mathematical Logic	Resolution Inference	$O(n^2)$ to exponential	$O(n)$	Expert systems, theorem proving
Discrete Probability	Naïve Bayes Classification	$O(n \cdot m)$	$O(m)$	Spam filtering, prediction systems

As evident from the comparison matrix provided in Table 3, a more systematic approach for the assessment of discrete mathematics approaches in terms of efficiency, computation requirements, and application has been undertaken. The algorithms using graphs are scalable and suitable for network applications, while combinatorial algorithms usually have an exponential complexity increase. Mathematical logic is highly precise when applied to formal inference processes but may require considerable computation resources depending on the complexity of the process. The use of discrete probabilities continues to be relevant in machine learning.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue VI June 2026

METHODOLOGICAL DIFFERENCES AMONG DISCRETE MATHEMATICS TECHNIQUES

In this part, the differences between graph theory, combinatorics, mathematical logic, and discrete probability in terms of methodology will be considered. All the four disciplines use discrete mathematics as their basis; however, the difference lies in how problems are solved and modeled.

Modeling Approaches

Each technique adopts a unique way of representing problems:

- Graph Theory models problems using vertices and edges, emphasizing relationships and connectivity. It is particularly effective for representing networks and interdependent systems.
- Combinatorics relies on enumeration and structural arrangement, focusing on counting all possible configurations within defined constraints.
- Mathematical Logic represents problems symbolically using propositions, predicates, and logical operators, enabling formal reasoning and rule-based analysis.
- Discrete Probability models systems using probabilistic frameworks, incorporating randomness and uncertainty into discrete structures.

These differences in representation directly influence how problems are interpreted and solved.

Problem-Solving Strategies

The techniques also vary in their methodological strategies:

Table 4. Problem-solving strategies

Graph Theory	Uses algorithmic approaches such as traversal (BFS, DFS), shortest path, and optimization algorithms.
Combinatorics	Applies formula-based and recursive methods, often involving permutations, combinations, and generating functions.
Logic	Employs deductive reasoning, truth tables, and inference rules to derive conclusions from given premises.
Probability	Utilizes statistical reasoning, probability distributions, and expectation-based analysis to predict outcomes.

Thus, graph theory and combinatorics are more constructive, while logic and probability are more analytical.

Deterministic vs. Probabilistic Nature

A key methodological distinction lies in how uncertainty is handled:

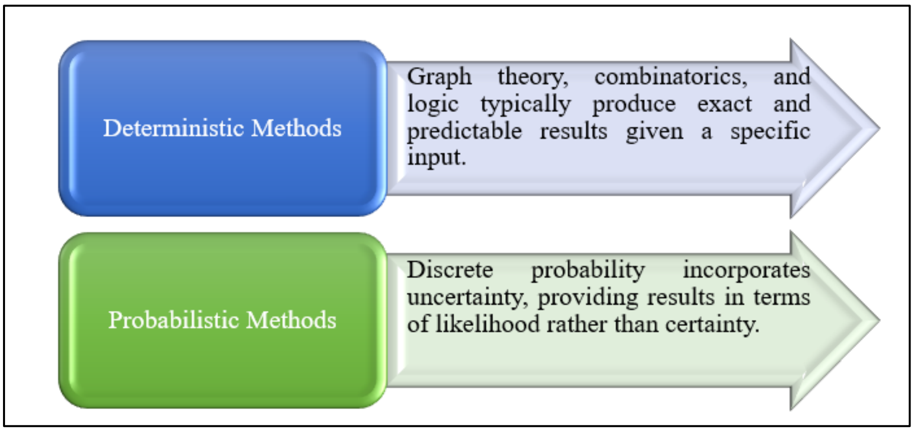

Figure 1. Methodological distinction

This distinction is crucial in real-world applications where data may be incomplete or uncertain.

Table 5. Algorithmic vs. Analytical Orientation

Algorithmic	Analytical Orientation
Graph Theory	Strongly algorithmic, focusing on step-by-step procedures for solving problems efficiently.
Combinatorics	Combines algorithmic and mathematical analysis, often requiring enumeration techniques.
Logic	Primarily analytical, emphasizing correctness and formal proof structures.
Probability	Analytical and data-driven, often supported by computational simulations and statistical tools.

Scalability and Complexity Handling

Various methods deal with complexity in their unique way. Algorithms and data structures assist graph theory in managing complexity. Exponential growth and approximations aid combinatorics in dealing with complexity. The logic system employs consistency to manage complexity but may turn out to be complicated due to the existence of many rules.

Integration of Techniques

In practice, these methodologies are often combined to solve complex problems:

- Graph-based models may incorporate probabilistic weights for uncertain networks
- Combinatorial optimization problems often use logical constraints
- Probabilistic reasoning can be integrated with logical inference (e.g., probabilistic logic models)

This proves beyond doubt that diverse methodologies are not indicative of separation but rather complement each other in an interdisciplinary context. The diverse methodologies that accompany the discrete mathematics methodologies seek to ensure that the advantages and limitations of every methodology are made apparent. Graph theory seeks to model structures; combinatorics emphasizes enumeration and exactness; logic emphasizes reasoning, while probability emphasizes management of uncertainties.

APPLICATION DOMAINS OF DISCRETE MATHEMATICS TECHNIQUES

Discrete mathematics techniques have a wide range of uses because they may be used in the modeling of complex structures, as well as providing answers to optimization problems and decision-making. The practical use of graph theory, combinatorics, mathematical logic, and discrete probability is compared below.

Computer Science and Algorithms

Discrete Mathematics serves as the foundation for Computer Science. Some of the areas where Graph Theory finds its application in computer science include path finding, graph traversal, and dependency resolution while designing algorithms. The algorithms consist of shortest path algorithms and optimizing network flows. Moreover, the Combinatorics domain is applied in the analysis of algorithms and development of algorithms based on their complexity. Mathematical Logic forms the core of Programming Languages, Compilers, and Formal Verification Systems.

Artificial Intelligence and Machine Learning

Discrete mathematics mathematical modeling aids in generating models for Artificial Intelligence Models. The approach of mathematical modeling has been widely applied to reasoning and learning within AI. Logic is widely applied in areas such as rule-based systems, expert systems, and knowledge representation. Graph theory can be applied in areas such as knowledge graphs, neural networks, and graph learning. Probability can be applied in machine learning models and probability inference. Combinatorics has been widely applied in feature selection during the training process of the machine learning model.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue VI June 2026

Network Analysis and Communication Systems

Efficient communication systems need discrete structures. Graph Theory deals with modeling communication networks such as the Internet, mobile communication networks, and social media networks. Combinatorics deals with designing and allocating resources of the communication network. Probability is utilized for analyzing reliability, traffic analysis, and failure rate of communication systems. Logic is required for making sure of the correct functioning of communication protocols.

Cryptography and Security

Security systems rely heavily on discrete mathematics to protect data and ensure privacy.

- Combinatorics is used in key generation and cryptographic algorithms.
- Graph Theory supports secure communication protocols and blockchain structures.
- Logic is applied in authentication mechanisms and formal verification of security protocols.
- Probability is used to evaluate attack likelihood and system vulnerabilities.

These techniques ensure secure data transmission and storage in digital environments.

Operations Research and Optimization

Discrete mathematics is essential in solving optimization problems in logistics, transportation, and resource management.

- Graph Theory is used in routing, scheduling, and supply chain optimization.
- Combinatorics provides methods for solving allocation and assignment problems.
- Probability supports decision-making under uncertainty in operational systems.
- Logic helps define constraints and rules in optimization models.

These applications improve efficiency and reduce operational costs.

Data Science and Analytics

In the era of big data, discrete mathematics plays a vital role in extracting meaningful insights.

- Probability is fundamental for statistical modeling, prediction, and hypothesis testing.
- Graph Theory is used in clustering, recommendation systems, and social network analysis.
- Combinatorics aids in data arrangement and pattern discovery.
- Logic ensures data integrity and supports query processing in databases.

These techniques enable data-driven decision-making across industries.

Table 6. Comparative Summary of Applications

Domain	**Most Relevant Technique**	**Key Contribution**
Computer Science	Graph Theory, Logic	Algorithm design, system correctness
Artificial Intelligence	Logic, Probability	Reasoning and learning
Networks	Graph Theory	Modeling and optimization
Cryptography	Combinatorics	Secure key generation
Operations Research	Graph Theory, Combinatorics	Optimization
Data Science	Probability	Prediction and analysis

The above application examples clearly illustrate how complementary the techniques are with regards to their application in discrete mathematics. Graph theory is necessary when addressing structural issues, while combinatorics will be the solution for any problem dealing with counting; logic and probability are needed for addressing correctness and prediction respectively.

Unfortunately, not many practical problems will ever be straightforward. As such, the combination of various techniques to solve one problem will always arise. This may include the use of graphs to represent users in a recommendation system, prediction based on probability and logic to ensure correctness.

Table 7. Providing versatile techniques

Technique	Strength	Limitation	Best Use Case
Graph Theory	Network modeling	Large-scale complexity	Routing, networks
Combinatorics	Exact counting	Exponential growth	Optimization, cryptography
Logic	Formal reasoning	Less flexible	AI, verification
Probability	Handles uncertainty	Requires data	Prediction, analytics

Discrete mathematics provides versatile tools that are applicable across a wide range of domains. The effectiveness of each technique depends on the nature of the problem, data availability, and computational requirements. A combined approach often yields the most efficient and robust solutions in modern applications.

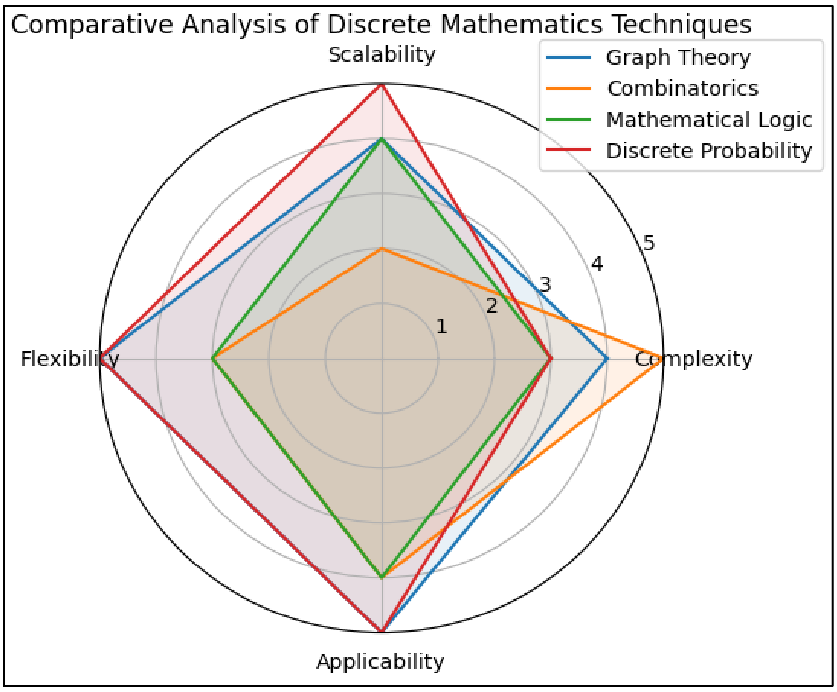

Figure 2. Comparative radar chart of discrete mathematics techniques based on computational complexity, scalability, flexibility, and applicability.

The figure above is a radar chart comparison for four major techniques: Mathematical probability, Discrete probability, Graph theory, and Combinatorics. The assessment is conducted under four broad areas of criterion evaluation: computational complexity (relative to typical computer memory usage and speed), scalability (in terms of the number of processors that can be scaled up or down), flexibility (ability to adapt to various applications), and applicability (ability to be implemented in real-world applications).

As can be seen, discrete probability (and graph theory) achieves the best overall level of performance in terms of scalability and applicability, which makes it very suitable for modern data-driven and network-based applications. Graph theory works well for instances of structured relationships, and probability is excellent for situations of uncertainty and large-scale data analysis. On the other hand, while combinatorics is very costly to compute, it suffers from low scalability because the size of the problem grows exponentially. Combinatorics, on the other hand, are very expensive when computing but low scalable when working in large systems because of their exponential growth of problem size. Mathematical logic exhibits balanced performance, especially in terms of scalability and applicability to structured reasoning contexts like formal verification systems or AI applications.

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue VI June 2026

The radar chart also depicts the balance between efficiency and scalability. The Graph Theory and Discrete Probability are more adaptable for large-scale real-world systems since they take advantage of efficient algorithms and probability approximations. On the other hand, combinatorial approach is likely to have exponential increases in computation that reduce scalability, especially when solving optimization problems. This chart provides clear comparison on why there is an increased adoption of approaches using graph theory, probability, and logic.

CONCLUSION AND FUTURE RESEARCH DIRECTIONS

Throughout this research paper, we have provided an extensive comparative analysis of four important disciplines in discrete mathematics, namely graph theory, combinatorics, mathematical logic, and discrete probability. Our main objective was to conduct an analysis of the theoretical foundation and methodology involved in each discipline and their methods of calculation and implementation in different spheres such as computer science, artificial intelligence, network science, and cryptography.

Graph Theory Application in Social Media Analysis

Many social media websites are studied with graph theory techniques since users can be modeled as vertices whereas the relationships between them as edges. Algorithms like PageRank and community detection algorithms are commonly used to find influential users and how information propagates among users.

Discrete Probability Applications in Machine Learning

Discrete probability theories and models form the backbone of many machine learning systems for prediction and classification purposes. For instance, naïve Bayes classifier and Bayesian network techniques are utilized in spam and fraud detection systems.

Combinatorics Applications in Optimization

Optimization problems in scheduling and logistics use various combinatorial methods to determine optimal solutions such as the minimum cost and shortest path algorithms which find the most efficient routes.

Logic Applications in Artificial Intelligence

Logic systems play an essential role in building expert systems and automated reasoning systems. They provide the basis for logical inference techniques used in artificial intelligence and verification systems.

As it turns out, both approaches have certain advantages and drawbacks. The use of graph theory as an approach is extremely effective and efficient in terms of building networks important for communication or transport purposes. Counting and optimization questions can be answered with perfect accuracy with the help of combinatorics, but the process can be extremely complex, with exponential growth. The main advantage of mathematical logic is its logical precision, while the drawback is a lack of flexibility when it comes to uncertainty. Finally, the use of discrete probability is indispensable in terms of modelling uncertainty and predictive analytics. Another core concept that is introduced here is the idea that there is no single approach to discrete mathematics that can be considered superior to the others. Instead, depending on the problem that must be solved, the type of algorithm used would change according to various factors such as computational power limitations as well as what needs to be achieved through the calculations done by the computer. Future research could focus on developing more integrated frameworks through the combination of graph theory approaches, logic, and probabilistic systems as well as combinatorial optimization.

REFERENCES

1. Rosen, K. H. (2019). *Discrete Mathematics and Its Applications* (8th ed.). McGraw-Hill Education.
2. West, D. B. (2001). *Introduction to Graph Theory* (2nd ed.). Prentice Hall.
3. Grimaldi, R. P. (2004). *Discrete and Combinatorial Mathematics: An Applied Introduction* (5th ed.).

ISSN No. 2454-6194 | DOI: 10.51584/IJRIAS |Volume XI Issue VI June 2026

Pearson.
4. Enderton, H. B. (2001). *A Mathematical Introduction to Logic* (2nd ed.). Academic Press.
5. Feller, W. (1968). *An Introduction to Probability Theory and Its Applications* (Vol. 1, 3rd ed.). Wiley.
6. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to Algorithms* (3rd ed.). MIT Press.
7. Diestel, R. (2017). *Graph Theory* (5th ed.). Springer.
8. Korte, B., & Vygen, J. (2018). *Combinatorial Optimization: Theory and Algorithms* (6th ed.). Springer.
9. Russell, S., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson.
10. Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Springer.
11. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
12. Hamilton, W. L. (2020). Graph Representation Learning. Morgan & Claypool Publishers.
13. Pearl, J. (1988). Probabilistic Reasoning in Intelligent Systems. Morgan Kaufmann.
14. Newman, M. (2018). Networks (2nd ed.). Oxford University Press.