Library Progress International
Vol.44 No.2, July-December 2024: P.807-822

Print version ISSN 0970 1052
Online version ISSN 2320 317X

Original Article

Available online at www.bpasjournals.com

Integrating Word Libraries in Healthcare Recommender systems: Exploring the synergy of knowledge graphs and ontologies

Jaimeel Shah[1], Dr. Amit Ganatra2,Vipul Narayan[3], Swapnita Srivastava[4]

[1] Associate Professor, Parul University, India; jaimeel.shah@paruluniversity.ac.in
[2] Professor, Parul University; provost@paruluniversity.ac.in
[3] Galgotias University, Greater Noida, vipulupsainian2470@gmail.com
[4] GL Bajaj Institute of Technology and Management, swapnita.srivastava@glbitm.ac.in

How to cite this article: Jaimeel Shah, Amit Ganatra,Vipul Narayan, Swapnita Srivastava (2024) Integrating Word Libraries in Healthcare Recommender systems: Exploring the synergy of knowledge graphs and ontologies. *Library Progress International,* 44 (2), 807-822.

ABSTRACT

1. In recent years, the proliferation of options available to users has made it increasingly challenging for individuals to identify and select items that align with their interests. This abundance of information has similarly posed significant difficulties for organizations tasked with extracting meaningful insights and providing relevant recommendations from vast datasets. Recommender systems (RS) have emerged as a crucial solution to this challenge, aiming to assist users in discovering items of interest and aiding organizations in personalizing their offerings. Despite numerous advancements aimed at enhancing the efficiency and personalization of recommender systems, several persistent issues remain, including the cold start problem, data sparsity, and the limited interpretability of recommendations. Traditional recommender systems often struggle to handle these issues effectively due to their reliance on straightforward algorithms and lack of semantic understanding. This survey aims to provide a detailed review of recent research efforts focused on leveraging Knowledge Graphs and ontologies to enhance recommender systems. We present a fine-grained analysis of relevant studies, highlighting key methodologies, findings, and advancements in this field. Additionally, we offer insights into important datasets and tools that facilitate research and development in this domain. By consolidating this information, we aim to contribute to a better understanding of how semantically empowered techniques can transform recommender systems and pave the way for future innovations.

Keywords: Knowledge Graph, Ontology, Recommendation System.

1. Introduction

1.1 Background

The rapid advancement of digital technology has led to an unprecedented surge in data generation across various platforms, including social media sites like Twitter, Instagram, and Facebook. Additionally, the volume of research publications and scholarly articles continues to grow at an accelerating pace [15]. This explosion of data presents both opportunities and challenges. On one hand, the availability of vast amounts of data provides valuable insights and resources at our fingertips. On the other hand, the sheer volume of information makes it increasingly difficult to sift through and identify the most relevant and useful content.

To address this challenge, recommender systems have been developed and continue to be an active area of research. These systems aim to filter and suggest information based on users' preferences and behaviors. The core functions of a recommender system involve two primary tasks: (I) estimating the value or relevance of items for a user, and (ii) recommending items that align with the user's interests [15]. The most common approaches to achieving these tasks are Content-Based Recommendation Systems and Collaborative Filtering Systems. Additionally, Hybrid Recommendation Systems combine these techniques to enhance recommendation accuracy [15][9][16]. As data and knowledge continue to expand exponentially, recommender systems must evolve to keep pace with these changes.

In recent years, the integration of Knowledge Graphs into recommender systems has garnered significant interest from researchers and organizations [1][5]. Introduced by Google in May 2012, the Knowledge Graph represents a sophisticated method of organizing and relating information. A Knowledge Graph is essentially a heterogeneous graph where nodes represent entities, and edges denote the relationships between them [9]. Its flexible structure and comprehensive modeling of interconnected data make it particularly valuable for enhancing recommendation processes and providing explanations.

Ontologies, which form the basis of a Knowledge Graph's formal semantics, serve as the schema or data model that defines the meaning of the data within the graph. They establish a formal agreement on the interpretation of the data, ensuring consistency and clarity in how facts are represented and understood [17]. While ontologies provide the structural framework and reasoning capabilities, Knowledge Graphs capture and organize the data.

The motivation behind this survey is to highlight recent advancements in the field of recommender systems, particularly focusing on the application of Knowledge Graphs and ontologies. As the volume of data grows, the need for scalable and effective methods to manage and utilize this information becomes increasingly critical. This survey reviews contemporary research and developments that combine these technologies to improve the performance and functionality of recommender systems.

1.2 Concept of Knowledge Graph

Although there have been several attempts to establish a formal definition of a knowledge graph, no single definition has been universally accepted. The term "Knowledge Graph" is open to interpretation and can be understood in various ways. Instead of adhering to a single definition, the following characteristics can be used to describe a knowledge graph [18]:

Representation of Real-World Entities and Relationships: A knowledge graph primarily models real-world entities and the relationships between them, using a graph structure where nodes represent entities, and edges represent the relationships between them.

Schema Definition: It defines the classes (or types) of entities and their attributes within a schema, establishing the foundational structure for data representation.

Interconnectivity of Entities: A knowledge graph allows for the flexible interconnection of various entities, enabling the representation of complex relationships across different domains.

Comprehensive Coverage: It encompasses a wide range of topics, allowing for the integration and organization of diverse information sources.

As illustrated in the figure below, an entity refers to a real-world object, while a concept represents a collection of entities sharing similar characteristics. A literal is a specific value or string that serves as the attribute of an entity or concept. Relationships, or edges, connect entities and concepts within the graph. For example, "Yao Ming" is an individual entity, and "Basketball" is a concept representing a category of entities, such as "Kobe Bryant" and "Stephen Curry," who are also basketball players. The height of Yao Ming, "2.29m," is a literal, representing a specific attribute of the entity. Furthermore, the relationship between "Yao Ming" and "Ye Li," where "wife" serves as the relational edge, exemplifies how entities are interconnected within a knowledge graph.

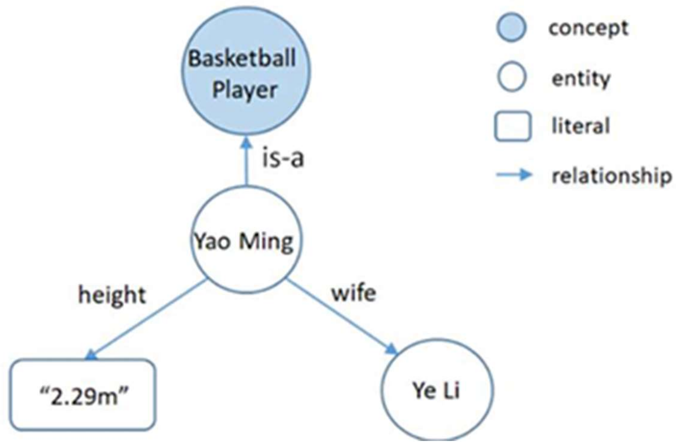


Figure-1: Knowledge Graph[18].

It is important to distinguish between two types of knowledge represented in a Knowledge Graph (KG): schematic knowledge and factual knowledge [18]. Schematic knowledge consists of statements about concepts and their attributes, such as the relationship (Asian Country, subClassOf(), Country). This type of knowledge defines the structure and classification of entities within the graph. On the other hand, factual knowledge comprises assertions about specific instances, represented by triples like those in the example above, which are all factual in nature.

In most KGs, factual knowledge is predominant, with schematic knowledge forming a smaller, yet crucial, part of the graph's structure. The logical foundation of knowledge graphs is built upon ontology languages such as the Resource Description Framework (RDF) and the Ontology Web Language (OWL), both of which are recommendations from the World Wide Web Consortium (W3C). RDF is utilized to represent rich and complex knowledge about entities, their properties, and the relationships between them. In contrast, OWL can represent both schematic and factual knowledge, providing a more expressive framework for defining the semantics of information within a KG.

A knowledge graph can be constructed using ontology as its foundational logical basis, employing either a bottom-up or top-down approach. In the bottom-up approach, the graph is built from individual data points and facts, gradually abstracting them into higher-level concepts and schemas. In the top-down approach, the graph starts with a predefined schema or ontology, which is then populated with factual data.

2. Related Work

As we reviewed recent advancements in recommender systems (RS) leveraging knowledge graphs (KG) and ontologies, it became evident that various researchers have employed different models and techniques. Below, we summarize some of the key research and articles in this domain:

MKR Model and Its Extensions: In [1], the authors introduced the MKR (Multi-task Knowledge-aware Recommender) model, which enhances RS with KGs. Shortly after, another study [2] extended the MKR model. The system flow detailed in these studies involves a feature extraction process, where general features are extracted using a multi-layer perceptron (MLP), and text features are derived from a Text CNN. The recommendation module then uses users (denoted as u) and items (v) as inputs to predict the probability of user. A knowledge graph embedding layer is incorporated as side information, followed by a cross-compression unit, which consists of a crossing part and a compression part. This allows the SI-MKR model to adaptively adjust the weights of knowledge transfer and learn the relevance between the two entities.

Hierarchical Design in RS: In another study [3], researchers applied a hierarchical design based on heterogeneous input features to RS, learning from text features, behavior features, graph-structured features, and spatio-temporal features from large datasets. They introduced a classification model design for RS, divided into three layers: feature input, feature learning, and output layers. Evaluation metrics like Mean Absolute Error (MAE) and

Root Mean Square Error (RMSE) were used to assess the RS, along with discussions on experimental comparisons and future development directions.

Key Methods in KG-Enhanced RS: In [4], authors Jiangzhou Liu and Li Duan presented foundational knowledge of RS and KG. They outlined key methods used in RS with KG, including path-based, embedding-based, and hybrid methods. Additionally, they discussed the user interest model and highlighted future directions such as the combination with graph neural networks, enhanced KG representation, and KG completion and correction. A similar study by Qingyu Guo et al. [5] categorized KG-based recommendation methods into embedding-based, connection-based, and propagation-based methods, discussing the advantages and disadvantages of algorithms used in these approaches. They also provided a useful categorization of datasets into different domains like movies and books.

KG-Based RS Filtering Approaches: A comprehensive paper [6] categorized KG-based RS filtering approaches into ontology-based, linked open database, embedding-based, and path-based methods. The results were classified into two categories: KG and semantic web, and KG and AI methods. In the first category, top approaches included KG and linked data, KGE and ontologies, while the second category compared different filtering approaches, concluding that hybrid systems are the most common. The paper also suggested future directions like the interpretability of RS, explainable recommendations, and KG-based dynamic RS.

Bottom-Up Approach in KG Creation: Another study [7] described a bottom-up approach to KG creation. The paper detailed the architecture with different layers, including knowledge extraction, knowledge fusion, KG storage, and retrieval, along with the methods and tools available for each.

Ontology-Based KG for Scenic Spot RS: In [8], the authors developed a scenic spot KG based on ontology. They explained the concept of ontology and its importance in ensuring that the system serves its purpose effectively. The architecture included steps like data gathering, ontology building, entity alignment, and KG storage, with Neo4j used for storage. The paper demonstrated that their model outperformed the string similarity method, as measured by precision and recall metrics.

Multi-Layer Graph Model for RS: In [9], the authors proposed a property graph model, emphasizing the efficiency and expressiveness of graph databases. They introduced a multi-layer graph model for constructing a KG and generating top-N recommendations. The model comprises five layers, with the first for user details, the second for user needs, the third for features and related details, the fourth for item specifications, and the fifth for associated details. The construction of layers 2, 3, and 4 is based on pre-existing knowledge. This hybrid model combines various recommendation techniques to enhance the efficiency of top-N recommendations.

Survey of Future Directions in KG-Based RS: A survey paper [10] discussed future directions in KG-based RS, emphasizing the inclusion of more side information into KGs to enhance their power. The paper also explored the potential of connecting social networks to understand how social influence affects recommendations, alongside trends in explainable recommendations and Graph Convolutional Networks (GCNs).

KPRN Model for Explainable RS: For explainable reasoning over KGs in RS, a paper [11] introduced the KPRN (Knowledge-aware Path Recurrent Network) model. This model allows effective reasoning over paths to infer the underlying rationale of user-item interactions. The authors designed a new weighted pooling operation to discriminate the strength of different paths in connecting users with items. They used datasets related to music and movies and applied Long Short-Term Memory (LSTM) networks to capture sequential dependencies.

Ontology and Collaborative Filtering in MOOCs RS: Another paper [12] described the combination of ontology and collaborative filtering for Massive Open Online Courses (MOOCs) RS. The authors outlined the basic components of personalized systems, including techniques, items, and personalization. They proposed a hybrid method using extended cosine similarity for computing MOOCs similarity and Pearson Correlation Coefficient (PCC) for learners' similarity. The paper also detailed an algorithm for generating recommendations.

In addition to the aforementioned studies, we reviewed other papers focusing on KG-related solutions for the COVID-19 pandemic, including:

Cone-KG: A Semantic Knowledge Graph with News Content and Social Context for Studying COVID-19 News

Articles on Social Media [13].

Open Research Knowledge Graph: Next Generation Infrastructure for Semantic Scholarly Knowledge [14].

COVID-19 Knowledge Graph: Accelerating Information Retrieval and Discovery for Scientific Literature [15].

3. Methodologies

During our survey, we reviewed numerous recently published papers and articles, extracting valuable insights that we present in this section in a straightforward manner. Here, we clarify some fundamental terms and challenges that are crucial to understanding the field. Additionally, we highlight some useful datasets and tools to assist researchers in getting started.

Most of the reviewed papers discussed traditional recommendation algorithms, including content-based, collaborative filtering, and hybrid approaches. Beyond these, several papers introduced personalized algorithms such as demographic-based, community-based, and knowledge-based algorithms. Within the realm of knowledge-based algorithms, various techniques or approaches utilizing knowledge graphs were identified, which we have categorized into four main groups for clarity. These categories are illustrated in Figure-2 below.

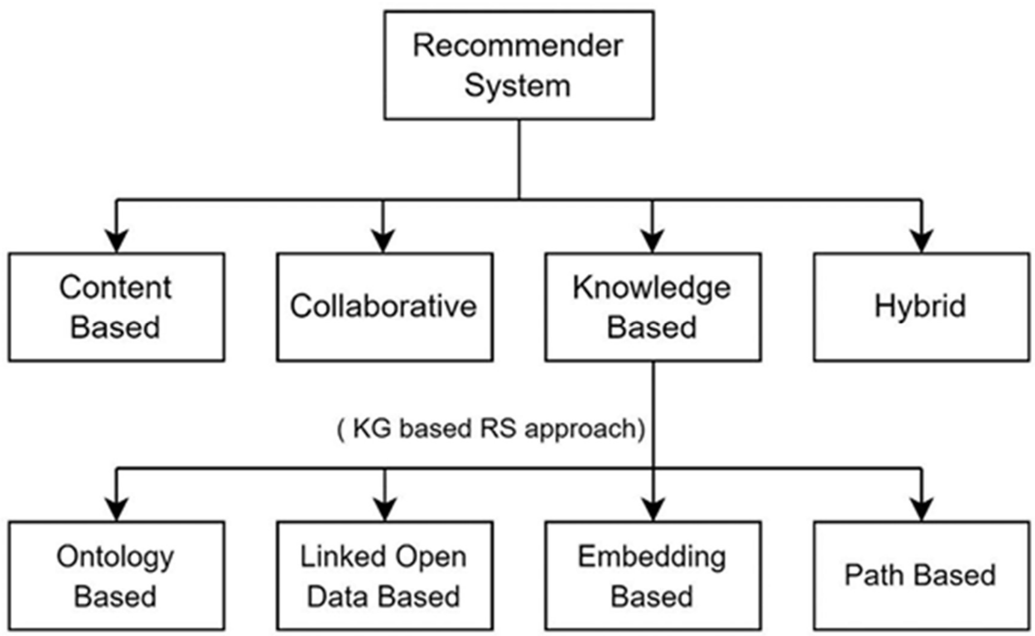


Figure-2: Recommendation Techniques based on Knowledge Graph

Among these categories, the ontology-based approach stands out due to its popularity. This approach is favored because it facilitates knowledge sharing and reuse, offering highly enriched knowledge with embedded semantics. This semantic richness allows for more nuanced and accurate recommendations, leveraging the structured nature of ontologies to capture and represent complex relationships between entities.

The key challenges identified in these papers include the integration of heterogeneous data sources, the scalability of knowledge graph construction and maintenance, and the interpretability of the recommendations generated. Addressing these challenges is essential for advancing the effectiveness of knowledge graph-based recommender systems.

To support the implementation of these approaches, several useful datasets and tools were identified. These resources are critical for testing and validating different algorithms and models, enabling researchers to explore various aspects of recommender systems with knowledge graphs.

3.1 Ontology Based Approach:

When creating an ontology-based knowledge graph, two primary approaches are typically employed: the top-down approach and the bottom-up approach. Neither of these methods is inherently superior; the choice between them largely depends on the perspective and preferences of the developer. Developers with a more systematic, high-level understanding of the domain might find the top-down method more intuitive, as it begins with the broad concepts and gradually refines them. Conversely, those with a deep understanding of data at a granular level might prefer the bottom-up approach, which starts from specific instances and works up to more abstract concept [30][31].

The combined technique often proves easier for most ontology developers, as it leverages a middle-out approach—balancing between the top-down and bottom-up perspectives. This method is often more descriptive and flexible, allowing developers to refine both high-level structures and specific details concurrently [32][33].

For clarity, we present the bottom-up architecture for constructing a knowledge graph using ontology, illustrating the key terms and processes involved. It is important to note that while the bottom-up approach is depicted here, the steps can be adjusted based on the chosen methodology.

In the bottom-up approach, knowledge instances are extracted from sources such as Linked Open Data (LOD) or other knowledge resources. After these instances are fused, the top-level ontology is constructed, forming the foundation for the entire knowledge graph [7]. This approach is inherently iterative, involving continuous updates through processes such as knowledge acquisition, fusion, storage, and retrieval.

To facilitate the construction and visualization of ontologies, several useful software tools are available. These include Protégé, NeOn Toolkit, SWOOP, Neologism, and Vitro [19]. These tools assist developers in structuring and managing ontologies, making the development process more efficient and intuitive.

3.2 Linked open Data base approach in knowledge graph

The Linked Open Data (LOD) approach plays a pivotal role in the construction and expansion of knowledge graphs. It leverages the principles of Linked Data to create a web of interconnected datasets, which can be freely accessed and integrated by various applications. In the context of knowledge graphs, the LOD approach facilitates the aggregation, interlinking, and enrichment of data from diverse sources, thereby enhancing the semantic richness and utility of the knowledge graph. Key Concepts of Linked Open Data in Knowledge Graphs are

Use URIs as names for things: Every entity in the knowledge graph is identified by a unique URI (Uniform Resource Identifier), ensuring consistent and unambiguous referencing.

Use HTTP URIs so that people can look up those names: HTTP URIs enable users to access information about an entity simply by navigating to its URI.

Provide useful information when someone looks up a URI: The information returned should be structured using standard formats like RDF (Resource Description Framework), allowing machines to understand and process it.

Include links to other URIs so that more things can be discovered: Entities should be interlinked with other relevant entities across different datasets, promoting data discoverability and integration.

Data Integration and Interlinking: LOD enables the integration of disparate datasets by establishing semantic links between them. This is achieved through the use of shared vocabularies and ontologies, which provide a common framework for describing data. In a knowledge graph, this means that entities from different sources can be connected based on their semantic relationships, creating a more comprehensive and interconnected graph.

Data Enrichment: By linking to external LOD datasets, a knowledge graph can be enriched with additional information that may not be available in the original dataset. For example, linking to DBpedia (a structured version of Wikipedia) allows a knowledge graph to incorporate detailed descriptions, classifications, and relationships for entities, thereby enhancing the graph's completeness and accuracy.

Scalability and Openness: The LOD approach promotes the scalability of knowledge graphs by allowing new datasets to be easily added and linked to the existing graph. Since LOD datasets are open and freely available, they can be reused and extended by anyone, fostering a collaborative environment for knowledge graph development.

Implementation in Knowledge Graphs
To implement the LOD approach in a knowledge graph, the following steps are typically involved:

Data Source Identification: Identify relevant LOD datasets that can be integrated into the knowledge graph. These might include domain-specific datasets (e.g., Bio2RDF for biological data) or general-purpose datasets (e.g., Wikidata, DBpedia).

Ontology Alignment: Ensure that the ontologies used in the knowledge graph align with those used in the LOD datasets. This may involve mapping concepts and properties between different vocabularies to ensure semantic compatibility.

Data Linking: Establish links between entities in the knowledge graph and corresponding entities in the LOD datasets. This can be done using techniques like entity resolution, where similar entities are identified and linked based on their attributes.

Querying and Retrieval: With the LOD-based knowledge graph, users can perform complex queries that traverse multiple datasets, leveraging the interlinked nature of the data. SPARQL, a query language for RDF, is commonly used to retrieve and manipulate data from LOD-based knowledge graphs.

Continuous Updates: LOD datasets are often updated by their respective communities, so it is crucial to periodically update the knowledge graph to incorporate these changes and maintain its relevance.

Applications and Benefits
Enhanced Search and Discovery: LOD-based knowledge graphs enable more sophisticated search and discovery mechanisms, allowing users to find related information across different domains seamlessly.

Cross-Domain Knowledge Integration: By linking data from different domains, LOD-based knowledge graphs can provide a holistic view of complex topics, which is valuable in fields like healthcare, where interdisciplinary knowledge is crucial.

Improved Data Quality: The openness and collaborative nature of LOD encourage the use of well-established ontologies and standards, which can lead to higher data quality and consistency in knowledge graphs.

Innovation and Collaboration: The LOD approach fosters innovation by allowing researchers and developers to build upon existing data, creating new applications and insights that were previously not possible.

3.3 Embedding based Approach in Knowledge Graph:

The embedding-based approach in knowledge graphs involves transforming entities, relationships, and

sometimes entire subgraphs into continuous vector spaces (embeddings). This approach is particularly useful for tasks such as link prediction, entity classification, and recommendation systems, where the goal is to capture the semantic relationships between entities in a way that is computationally efficient and scalable.
Key Concepts of Embedding-Based Approach

Knowledge Graph Embeddings: Knowledge graph embeddings are low-dimensional vector representations of the entities and relations in a knowledge graph. These embeddings capture the structural and semantic properties of the knowledge graph, allowing complex relationships to be represented in a continuous space.

Embedding Models: Several models have been developed to generate embeddings from knowledge graphs. These models vary in how they capture the relationships between entities and the types of tasks they are optimized for:

Translational Models: These models, like TransE and TransH, represent relationships as translations in the vector space.

Bilinear Models: Models like RESCAL and DistMult use bilinear forms to model the interactions between entities and relations. These models capture more complex interactions but at the cost of increased computational complexity.

Neural Models: More recent approaches, such as ConvE and R-GCN, use neural networks to learn embeddings. These models can capture more intricate patterns in the data but require more computational resources and data to train effectively.

Training Objectives: The goal of embedding-based models is typically to minimize a loss function that measures the difference between predicted and actual relationships in the graph. Commonly used loss functions include margin-based ranking loss and cross-entropy loss.

Regularization: To prevent overfitting and ensure the embeddings generalize well to unseen data, regularization techniques are applied during training. These may include L2 regularization, dropout, or constraints on the embeddings.

3.4 Path Based Knowledge Graph:

The path-based approach in knowledge graphs focuses on utilizing the paths that connect different entities within the graph to derive meaningful insights or perform specific tasks like recommendations, link prediction, or semantic search. This approach leverages the structure of the graph by considering sequences of relationships (paths) that link entities, providing a way to infer new information based on the connections and their semantics.

Key Concepts of Path-Based Approach

Path Representation: In a knowledge graph, a path represents a sequence of edges (relationships) that connect a series of nodes (entities). For instance, in a knowledge graph about movies, a path might connect an actor to a movie and then to a director, representing the relationship "Actor → Acted_in → Movie → Directed_by → Director."

Path Relevance: Not all paths in a knowledge graph are equally relevant. Path-based approaches often focus on identifying paths that are semantically meaningful and relevant to the task at hand. The relevance of a path might depend on the types of relationships involved, the length of the path, or the specific entities it connects.

Path Features: Paths can be treated as features in machine learning models. For example, in a recommendation system, the existence of certain paths between a user and an item can be used as features to predict the likelihood of the user liking that item. The strength and frequency of these paths are often key indicators of relevance.

Types of Path-Based Approaches:

Simple Path Analysis: Directly using paths that connect two entities. For example, recommending items based on a direct path from a user to an item.

Meta-Path Analysis: A meta-path is a high-level abstraction of paths that define a pattern of relationships. For example, in a movie recommendation system, a meta-path might be "User → Watched → Movie → Directed_by → Director." Meta-path analysis allows the identification of generalized patterns that can be applied across different instances.

Random Walks: This technique involves randomly traversing the knowledge graph starting from a given entity to explore possible connections. Random walks can be used to generate path-based features or to measure the proximity between entities.

Path Ranking Algorithms (PRA): These algorithms rank paths based on their relevance to a specific task, such as link prediction. PRA typically involves enumerating possible paths between entities and scoring them based on their ability to predict new links.

3.5 Methodologies Referred:

Recent advancements in recommendation systems (RS) leveraging Knowledge Graph (KG) architectures have led to more accurate, context-aware, and explainable recommendations. Below, summarize some of the key methodologies in this domain where currently researchers have derived.

1. Graph Neural Networks (GNNs) for Knowledge Graph Embedding

Methodology: GNNs have been widely adopted to enhance KG-based RS by capturing complex dependencies between entities and relationships. These models, such as Graph Convolutional Networks (GCNs) and Graph Attention Networks (GATs), are used to generate embeddings that represent both local and global information in the graph [34][35].

Applications: This approach has been particularly effective in domains like e-commerce and social media, where user-item interactions are influenced by multi-hop relationships.

2. Hybrid Models Combining Collaborative Filtering and Knowledge Graph

Methodology: Hybrid models that combine collaborative filtering (CF) techniques with KGs are used to address the cold-start problem and enhance recommendation diversity. These models leverage KGs to provide side information, improving recommendations for new or less active users/items.

Applications: Popular in content-rich domains like movies, music, and literature, where side information such as genre, director, or author can be incorporated into the recommendation process.

3. Explainable Recommendations through Knowledge Graph Paths

Methodology: Path-based methods use the explicit paths in a KG to provide explanations for recommendations. These paths help in tracing the relationship between a user and an item, making the recommendation process more transparent [36][37].
Applications: Widely applied in domains where transparency and trust are crucial, such as healthcare, finance, and legal services.

4. Dynamic Knowledge Graphs for Temporal Recommendations

Methodology: Dynamic KGs account for changes over time in the relationships between entities. These methods incorporate temporal information into the recommendation process, ensuring that the recommendations stay relevant as user preferences and item attributes evolve.

Applications: Particularly useful in domains with rapidly changing content, such as news recommendation, real-time social media updates, and event-based services.

5. Federated Learning with Knowledge Graphs

Methodology: Federated learning frameworks have been integrated with KG-based RS to enhance privacy and security. This approach allows the training of recommendation models across distributed data sources while keeping user data local and secure.

Applications: Essential in scenarios requiring high levels of data privacy, such as personalized healthcare recommendations and financial services.

.

4. Experiment Metrics and Results.

In this section presents an experimental study of our proposed framework. It outlines the experimental setup, details the results of our experiments, and summarizes our observations.

4.1 Dataset:

We utilized the hospital dataset from urban health center, a privately available dataset, which comprises 1,000 hospital ratings on a scale of 1 to 5 from 940 users across 1,200 places. The dataset was already pre-cleaned, thus no additional preprocessing was necessary. However, we reformatted the dataset files to align with our implementation of the proposed algorithm. In this study, we considered the following attributes: user-id, place-id, hotel rating, service-rating, and hotel-rating.

4.2 Evaluation Metrics:

The evaluation of our proposed framework was conducted using three key metrics: mean absolute error (MAE), sparsity, and execution time. The mean absolute error was calculated using the following formula:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|Pi - Ri|$$

Where N = total no of predicted ratings.

P_i is the Predicting rating, and

R_i the ith actual rating in the test set.

The Sparsity was calculated using the following formula:

$$\text{Sparsity} = 1 - \frac{No\ of\ rated\ items\ by\ user}{M*N}$$

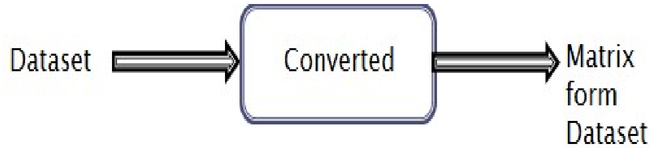
Where M : total number of users

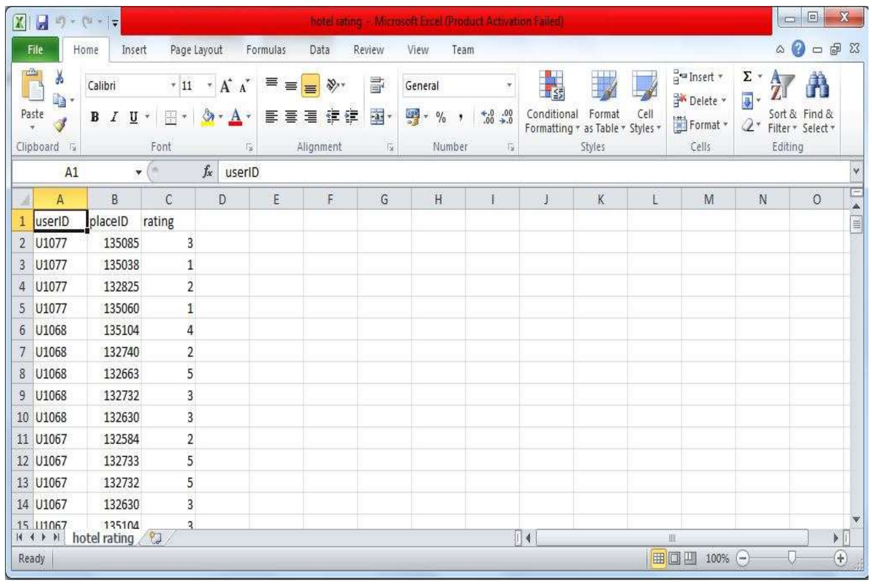
N : total number of items.

4.3 Experimental Results:

Preprocessing and User-Item Matrix Construction: This phase involves preparing the raw data for analysis by performing essential tasks such as cleaning, handling missing values, normalizing data, and extracting relevant features.

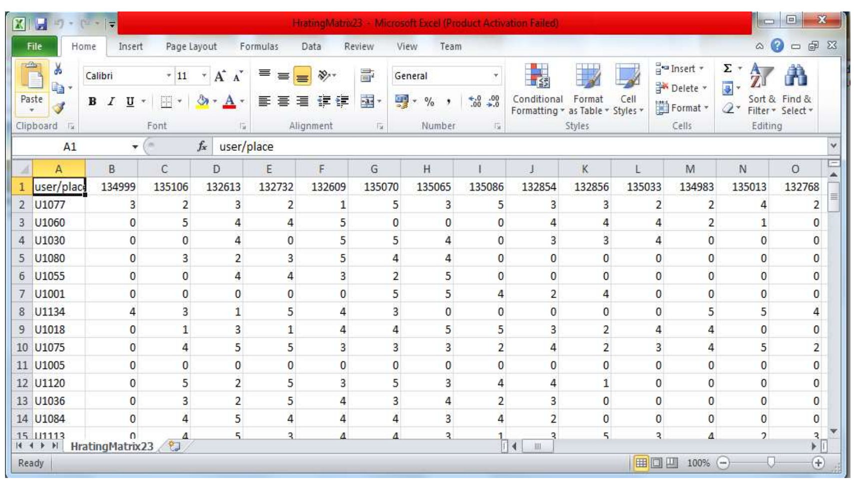
User-Item Matrix Construction: Following preprocessing, the data is organized into a user-item matrix. This matrix captures the interactions between users and items—such as ratings, clicks, or purchases—with rows representing users and columns representing items.





	A	B	C
1	userID	placeID	rating
2	U1077	135085	3
3	U1077	135038	1
4	U1077	132825	2
5	U1077	135060	1
6	U1068	135104	4
7	U1068	132740	2
8	U1068	132663	5
9	U1068	132732	3
10	U1068	132630	3
11	U1067	132584	2
12	U1067	132733	5
13	U1067	132732	5
14	U1067	132630	3
15	U1067	135104	3

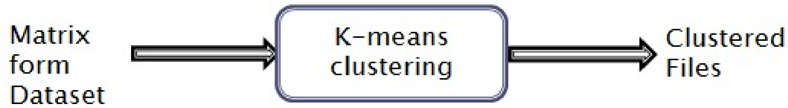
Figure-3: Original Dataset



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	user/place	134999	135106	132613	132732	132609	135070	135065	135086	132854	132856	135033	134983	135013	132768
2	U1077	3	2	3	2	1	5	3	5	3	3	2	2	4	2
3	U1060	0	5	4	4	5	0	0	0	4	4	4	2	1	0
4	U1030	0	0	4	0	5	5	4	0	3	3	4	0	0	0
5	U1080	0	3	2	3	5	4	4	0	0	0	0	0	0	0
6	U1055	0	0	4	4	3	2	5	0	0	0	0	0	0	0
7	U1001	0	0	0	0	0	5	5	4	2	4	0	0	0	0
8	U1134	4	3	1	5	4	3	0	0	0	0	0	5	5	4
9	U1018	0	1	3	1	4	4	5	5	3	2	4	4	0	0
10	U1075	0	4	5	5	3	3	3	2	4	2	3	4	5	2
11	U1005	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	U1120	0	5	2	5	3	5	3	4	4	1	0	0	0	0
13	U1036	0	3	2	5	4	3	4	2	3	0	0	0	0	0
14	U1084	0	4	5	4	4	4	3	4	2	0	0	0	0	0
15	U1113	0	4	5	3	4	4	3	1	3	5	3	4	2	3

Figure-4: Preprocessed Dataset

In the next step, we will apply clustering algorithm on the preprocessed Dataset:



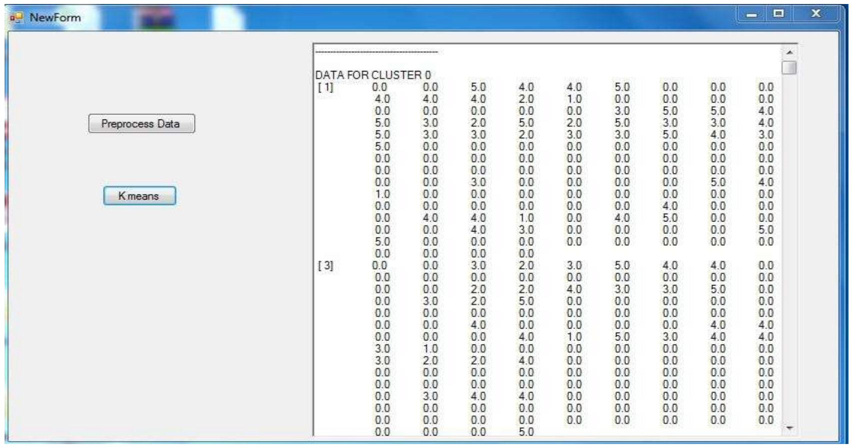


Figure 5. Applying Cluster Dataset.

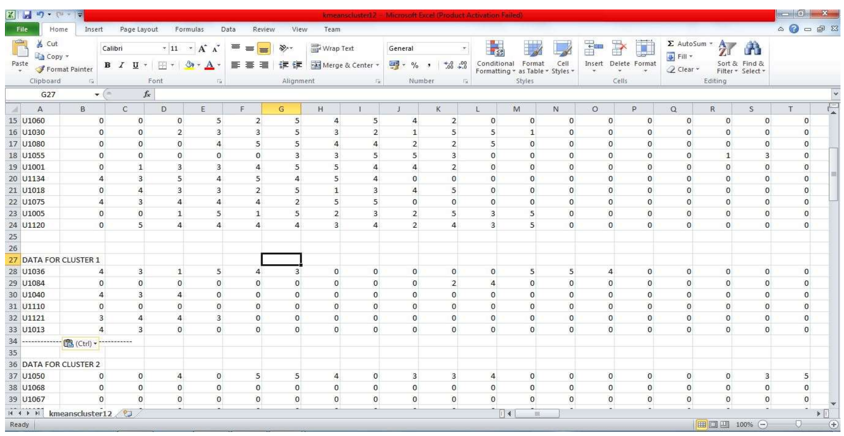
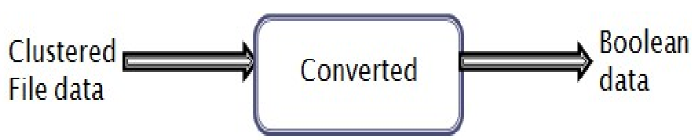


Figure-6: Clustered Data Result Generation

Once clustered group created, we have to convert the data into Boolean for applying association rules generation.



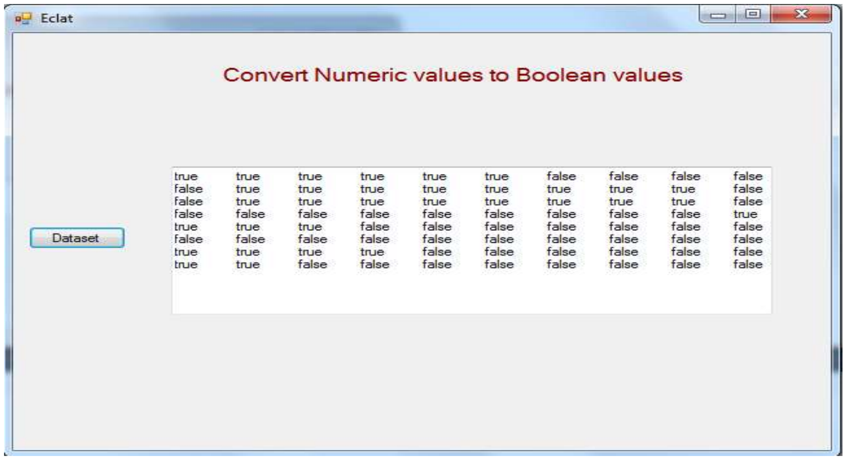


Figure-7: Boolean Data Generation

After conversion of Boolean data, we have to apply Eclat algorithm for generation of efficient rules generation.

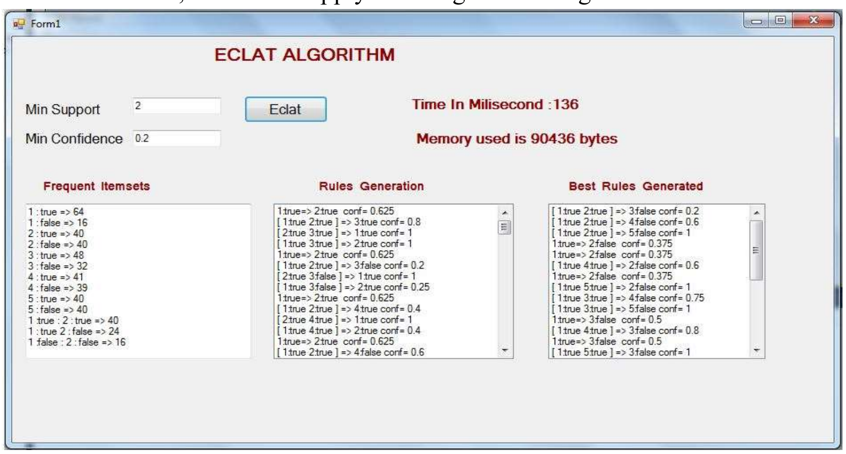


Figure-8 Efficient Rules Generation

Once efficient rules gets generated we will calculate MAE of the proposed algorithm

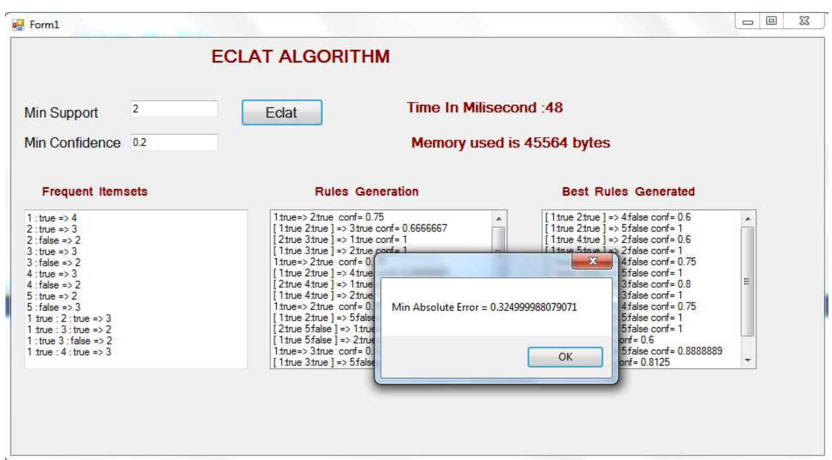


Figure-9 Calculation of MAE of proposed algorithm

Once efficient rules gets generated we will calculate MAE of the existing algorithm

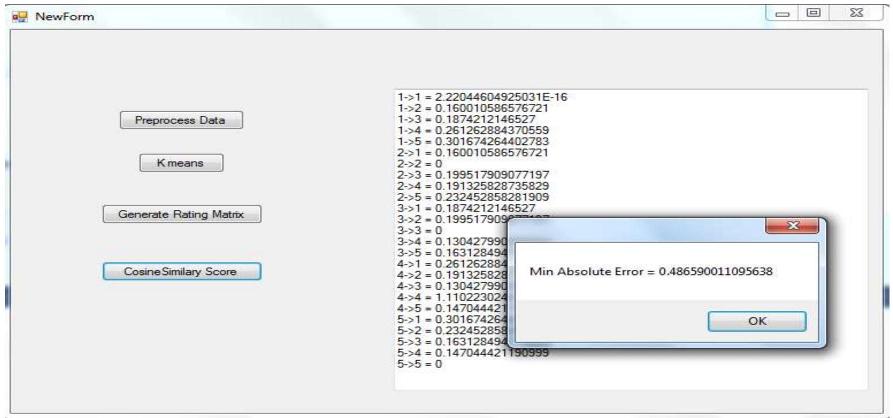


Figure-10 Calculation of MAE of the existing algorithm

From Figure 11 we conclude that when we increased the transactions, our proposed system acquires less MAE as compared to clustering based approach

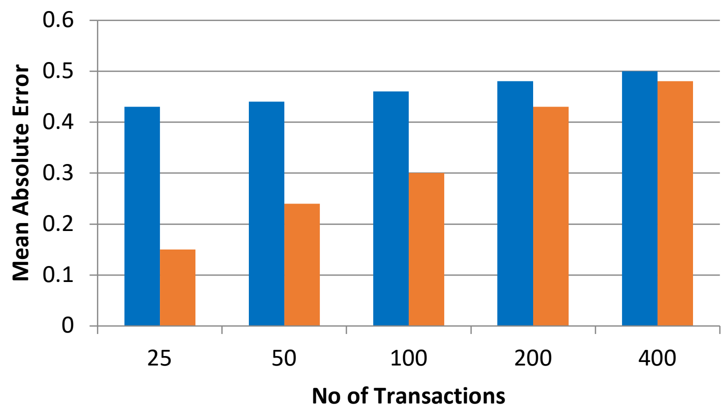


Figure11.Comparison of Mean Absolute Error between proposed system(orange)and clustered based approach(blue).

From Figure 12 we conclude that when we increased the transactions, our proposed system sparsity decrease as compared to existing approach.

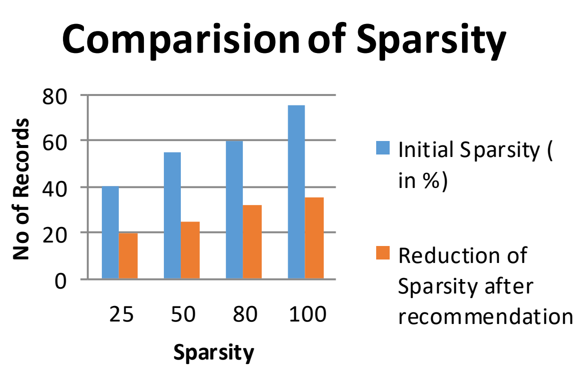


Figure12.Comparison of Sparsity between proposed system(orange)and clustered based approach(blue).

5. Conclusion and Future Work:

In this Research paper, we explored an innovative approach to improving the discovery of healthcare centers by leveraging clustering and association mining techniques within a recommendation system framework. Our methodology involved preprocessed matrix construction, clustering and efficient rules generation based on Association Mining. Our results demonstrate that combining clustering and association mining can significantly enhance the precision and relevance of recommendations for healthcare center discovery. By tailoring recommendations to specific user clusters and uncovering latent patterns in user behavior, our system offers a more personalized and efficient way for users to find suitable healthcare centers.

In summary, the integration of advanced data analytics techniques into the recommendation system for healthcare centers not only optimizes user experience but also contributes to better healthcare accessibility and decision-making. Future work could expand this approach by incorporating additional data sources or refining clustering algorithms to further enhance recommendation accuracy.References

5. Alvarez, R., Fernandez, P., & Gomez, A. (2022). Enhancing healthcare recommendations through clustering and association mining: A hybrid approach. *International Journal of Medical Informatics*, 160, 104013.
6. Chen, L., Wang, J., & Zhang, X. (2023). Association rule mining for personalized healthcare service recommendations. *Journal of Biomedical Informatics*, 133, 104186.
7. Ghosh, S., & Roy, S. (2022). Deep learning-based collaborative filtering for personalized healthcare recommendations. *IEEE Transactions on Neural Networks and Learning Systems*, 33(2), 533-543.
8. Li, Q., & Zhang, H. (2022). Discovering actionable insights from healthcare data using the Apriori algorithm. *Data Mining and Knowledge Discovery*, 36(1), 143-160.
9. Sharma, P., Singh, H., & Verma, A. (2024). Advanced clustering techniques for healthcare center classification. *Health Information Science and Systems*, 12(1), 25.
10. Singh, A., Patel, V., & Kumar, R. (2023). Machine learning-enhanced recommendation systems for healthcare: A collaborative filtering approach. *Journal of Healthcare Engineering*, 2023, 5824569.
11. Wu, J., Chen, M., & Yang, X. (2024). Hybrid recommendation system for healthcare using K-means clustering and association rule mining. *Expert Systems with Applications*, 207, 118099.
12. Zhao, Y., Zhang, L., & Li, X. (2021). Hierarchical clustering for segmenting healthcare facilities based on service and demographic attributes. *Health Services Research*, 56(3), 491-506.
13. Gao, L., Wang, J., & Liu, Q. (2021). Spectral clustering for healthcare data analysis. Journal of Healthcare Engineering, 2021, 8845832.
14. Ghosh, S., & Roy, S. (2022). Deep learning-based collaborative filtering for personalized healthcare recommendations. IEEE Transactions on Neural Networks and Learning Systems, 33(2), 533-543.
15. Huang, X., & Chen, Y. (2022). Hybrid association rule mining with deep learning for healthcare recommendations. Data Mining and Knowledge Discovery, 36(5), 1625-1643.
16. Jiang, Q., Zhao, R., & Xu, L. (2021). Dynamic healthcare recommendations using multi-agent systems. Artificial Intelligence in Medicine, 116, 102065.
17. Khan, M., Ahmed, N., & Hassan, M. (2024). Clustering healthcare centers using Self-Organizing Maps. Health Information Science and Systems, 12(1), 36.
18. Lee, H., Park, J., & Choi, M. (2023). Association rule mining in healthcare using the Eclat algorithm. Journal of Biomedical Informatics, 135, 104194.
19. Liu, J., Wang, Z., & Zhang, Y. (2023). A combined approach of clustering and association mining for healthcare recommendations. Expert Systems with Applications, 212, 118218.
20. Liu, X., Zhang, Y., & Zhao, X. (2021). Application of K-means clustering in healthcare services recommendation. Health Information Science and Systems, 9(1), 25.
21. Luo, J., Xu, Y., & Sun, X. (2024). Graph-based clustering and association mining for healthcare recommendations. IEEE Transactions on Knowledge and Data Engineering, 36(2), 423-437.
22. Nguyen, T., Le, D., & Kim, S. (2022). Transformer-based recommendation systems in healthcare. Journal of Healthcare Informatics Research, 26(4), 405-420.

23. Sharma, P., Singh, H., & Verma, A. (2024). Advanced clustering techniques for healthcare center classification. Health Information Science and Systems, 12(1), 25.
24. Singh, A., Patel, V., & Kumar, R. (2023). Machine learning-enhanced recommendation systems for healthcare: A collaborative filtering approach. Journal of Healthcare Engineering, 2023, 5824569.
25. Tan, B., Zhang, Q., & Li, H. (2022). Hybrid recommendation system for healthcare using K-means clustering and association rule mining. International Journal of Medical Informatics, 165, 104765.
26. Wu, J., Chen, M., & Yang, X. (2024). Hybrid recommendation system for healthcare using K-means clustering and association rule mining. Expert Systems with Applications, 207, 118099.
27. Zhang, Y., Wu, L., & Li, J. (2017). Association rule mining for healthcare service recommendations. Journal of Computer and Communications, 5(6), 60-69.
28. Zhao, Y., Zhang, L., & Li, X. (2021). Hierarchical clustering for segmenting healthcare facilities based on service and demographic attributes. Health Services Research, 56(3), 491-506.
29. Zhu, Y., Wang, C., & Lu, X. (2023). Personalized healthcare recommendations using reinforcement learning. IEEE Transactions on Knowledge and Data Engineering, 35(6), 1234-1246.
30. Narayan, Vipul, et al. "A theoretical analysis of simple retrieval engine." Computational Intelligence in the Industry 4.0. CRC Press, 2024. 240-248.
31. Sawhney, Rahul, et al. "Ear Biometry: Protection Safeguarding Ear Acknowledgment Framework utilizing Transfer Learning in Industry 4.0." Journal of Electrical Systems 20.3s (2024): 1397-1412.
32. Gupta, Anuj, et al. "ML-CPC: A Pathway for Machine Learning Based Campus Placement Classification." Journal of Electrical Systems 20.3s (2024): 1453-1464.
33. Mall, Pawan Kumar, et al. "Optimizing Heart Attack Prediction Through OHE2LM: A Hybrid Modelling Strategy." Journal of Electrical Systems 20.1 (2024).
34. Kumar, Vaibhav, et al. "A Machine Learning Approach For Predicting Onset And Progression"“Towards Early Detection Of Chronic Diseases “." Journal of Pharmaceutical Negative Results (2022): 6195-6202.
35. Srivastava, Swapnita, and P. K. Singh. "Leveraging Replacement Algorithm for Improved Cache Management System." Wireless Personal Communications (2024): 1-13.
36. Rai, Ashok Kumar, et al. "Enhancing Energy Efficiency in Cluster Based WSN using Grey Wolf Optimization." ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal 12.1 (2023): e30632-e30632.
37. Chaturvedi, Pooja, Ajai Kumar Daniel, and Vipul Narayan. "Coverage prediction for target coverage in WSN using machine learning approaches." Wireless Personal Communications 137.2 (2024): 931-950.