International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Network Analysis of Classical Music Composers' Relationships Based on Knowledge Graph

Du Shen[1+]

Submitted: 26/09/2023 **Revised**: 17/11/2023 **Accepted**: 28/11/2023

Abstract: The Hidden Markov Weighted Network Analysis Graph (HMWNag) is a novel and comprehensive framework that combines the power of Hidden Markov Models (HMMs), network analysis, and knowledge graphs to explore the intricate relationships and patterns within the world of music composers and their compositions. In this paper, we present the design and application of the HMWNag, which allows us to uncover hidden creative phases in composers' careers and trace the transitions between these phases. Through incorporating weighted values in the knowledge graph representation, we quantify the strength and significance of relationships between composers and compositions. Applying network analysis techniques to the HMWNag reveals influential composers and communities with shared characteristics, shedding light on musical influences and the evolution of classical music styles over time. Additionally, the HMWNag provides practical applications, including personalized music recommendations and music education programs, enhancing our understanding and appreciation of classical music history. Through this multidimensional approach, the HMWNag emerges as a powerful tool to unravel the complexities of music composers and their works, offering a holistic perspective on the rich tapestry of classical music.

Keywords: *Hidden Markov Model, Knowledge Graph, Network Analysis, Classical Music, Composers*

1. Introduction

Network analysis is a powerful and intricate field that delves into the study of interconnected systems, where nodes or entities are linked through various relationships or interactions. Whether it's understanding the intricate web of social connections in a virtual world or untangling the complexities of data transmission across vast communication networks, network analysis provides invaluable insights into the dynamics, structure, and functioning of these intricate systems [1]. Through employing mathematical models, statistical techniques, and advanced algorithms, network analysts are able to unravel patterns, identify central nodes, and assess the resilience of networks in the face of disruptions. This multidisciplinary field finds applications in diverse domains, ranging from sociology and biology to computer science and finance, shaping our understanding of complex systems and paving the way for enhanced efficiency and problem-solving in our interconnected world [2]. Network analysis, also known as graph theory or network science, is a rich and interdisciplinary field that has gained significant traction in recent years due to the explosion of digital data and the increasing complexity of interconnected systems [3]. At its core, network analysis focuses on examining the relationships and interactions among entities represented as nodes, and the connections between them, represented as edges or links. These entities could be anything from individuals in a social network, genes in a biological pathway, computers in a computer network, or even financial transactions in an economic system [4].

The key aspects of network analysis is the study of network properties and metrics. Researchers often investigate measures such as centrality, which identifies the most influential nodes in a network, degree distribution, which characterizes the distribution of connections across nodes, and clustering coefficients, which reveal the presence of tightly-knit groups within a network [5]. These metrics allow analysts to understand the structure and dynamics of networks, enabling them to identify key players, bottlenecks, and vulnerabilities in a system. Network analysis finds applications across a wide range of domains [6]. In sociology, it helps reveal the patterns of social interactions and the spread of information or influence within a community. In biology, it aids in understanding the complex interactions between molecules and genes in biological networks, shedding light on disease pathways and drug targets. In computer science, network analysis is crucial for designing efficient computer networks, optimizing data routing, and detecting anomalies or security threats [7]. Additionally, in finance, network analysis can be used to study financial markets and identify systemic risks that could lead to cascading failures. The tools and techniques used in network analysis have evolved significantly with advancements in data science and machine learning [8]. Network analysts employ various algorithms, such as community detection, link prediction, and network embedding, to extract meaningful insights from massive

[1] *Awang Had Salleh Graduate School of Arts and Sciences, University Utara Malaysia, UUM Sintok Kedah Darul Aman, 06010, Malaysia*
corresponding author : shendu730@163.com

datasets and complex networks. Additionally, network visualization plays a crucial role in network analysis, enabling researchers to present complex structures in an intuitive and understandable manner [9]. The interdisciplinary nature of network analysis allows researchers from diverse fields to collaborate and exchange ideas, fostering innovation and discovery. Through unraveling the underlying mechanisms of interconnected systems, network analysis has the potential to drive progress, optimize processes, and enhance decision-making in our increasingly interconnected and data-driven world [10].

Network analysis, combined with the power of knowledge graphs, constitutes a formidable approach to understanding and navigating the vast and intricate landscape of information in today's digital age [11]. Network analysis focuses on exploring the relationships and interactions among entities represented as nodes, while knowledge graphs provide a structured framework to organize and connect knowledge in a meaningful way. Together, these two fields enable us to uncover hidden patterns, discover meaningful insights, and gain a holistic understanding of complex systems, whether it's in social networks, biological pathways, or knowledge repositories [12]. With harnessing the potential of network analysis with knowledge graphs, we embark on a transformative journey of unraveling the interconnectedness of information, unlocking new opportunities for innovation, decision-making, and problem-solving across diverse domains [13]. Network analysis, as previously discussed, is concerned with studying the relationships and interactions between entities, forming a complex web of interconnected nodes and edges. On the other hand, knowledge graphs provide a structured representation of knowledge, where information is organized into nodes representing entities, and edges connecting these entities to denote relationships between them [14]. This structured approach allows us to capture and model the rich semantics and context of data, enabling more efficient and insightful analysis. When combined, network analysis and knowledge graphs offer a powerful framework to explore and navigate vast amounts of information with unparalleled precision and depth. Knowledge graphs serve as a repository for storing diverse data, ranging from factual knowledge and ontologies to unstructured text and multimedia content [15]. This comprehensive knowledge representation sets the stage for conducting sophisticated network analysis within the context of the real-world scenarios the data represents.

In practical applications, the fusion of network analysis and knowledge graphs opens up a plethora of possibilities [16]. For instance, in the realm of social networks, knowledge graphs can capture the attributes of individuals, their connections with others, and the content they share, forming a rich social knowledge graph. Analyzing this graph can lead to the discovery of influential users, identifying communities of interest, and predicting the spread of information or trends within the network [17]. In the field of bioinformatics, knowledge graphs can store information about genes, proteins, biological pathways, and disease associations, while network analysis can unveil critical genes' roles in the network, signaling pathways, and how diseases spread within biological systems. This synergy provides novel insights into complex biological processes and opens avenues for personalized medicine and drug discovery. Moreover, knowledge graphs can be leveraged in various domains, such as recommendation systems, question-answering platforms, and search engines, where relationships and connections between entities hold the key to delivering accurate and relevant information to users [18]. However, the integration of network analysis and knowledge graphs also comes with its challenges. As data sources grow larger and more heterogeneous, the scalability and efficiency of analyzing and updating knowledge graphs become critical concerns. Addressing these challenges requires developing advanced algorithms, machine learning techniques, and distributed computing infrastructures to manage and extract insights from massive and dynamic knowledge graphs [19].

Nonetheless, the potential of network analysis with knowledge graphs is vast and far-reaching. As these fields continue to evolve, their collaboration promises to reshape how we interact with data, driving innovation, and advancing our understanding of the interconnected world around us [20]. With the harnessing the synergies of network analysis and knowledge graphs, we unlock the power to navigate the complexities of information and transform it into meaningful knowledge, ultimately enhancing decision-making, problem-solving, and fostering progress across numerous domains. Classical music composers play a crucial role in network analysis and knowledge graphs, as their contributions to the world of music can be effectively represented and analyzed within these frameworks [21]. With organizing their compositions and relationships with other composers as nodes and edges in a knowledge graph, valuable insights can be gleaned regarding the evolution of musical styles, influences, and artistic dialogues across different eras. Composers' influence networks can be constructed, highlighting the profound impact they had on their contemporaries and subsequent generations. Moreover, knowledge graphs allow for the exploration of collaborations and pedagogical connections among composers, shedding light on the transmission of musical knowledge and techniques [22]. The interconnected representation of classical music history provides musicologists and researchers with data-driven opportunities to study genres, styles, and cultural contexts, enhancing our understanding of this timeless art form and facilitating personalized recommendations for music education and appreciation.

2. Knowledge Graph

A knowledge graph is a powerful data structure used to organize and represent knowledge in a structured and interconnected manner. It consists of nodes representing entities or concepts and edges representing the relationships between these entities. Each node in the graph contains information about the entity, and each edge defines the nature and direction of the relationship between connected nodes. Knowledge graphs are used to model complex and heterogeneous data from various sources, such as facts, attributes, events, and semantic connections. They enable a deeper understanding of the underlying relationships and context of the data, making it easier to extract insights, perform data analysis, and support reasoning tasks. These graphs find applications in a wide range of domains, including natural language processing, recommendation systems, question-answering platforms, bioinformatics, and more. As technology and data continue to grow, knowledge graphs are becoming an essential tool for organizing, navigating, and making sense of the vast amounts of information available in the modern digital age.

Network analysis, on the other hand, is a mathematical and statistical approach used to study complex systems represented as networks. It involves calculating various metrics and properties of the nodes and edges in the network, enabling the identification of key nodes, clusters, and other structural patterns. The connection between knowledge graphs and network analysis lies in applying network analysis techniques to analyze the structure and dynamics of knowledge graphs. Through treating knowledge graphs as networks, we can compute centrality measures, such as degree centrality, which indicates the importance of nodes based on their number of connections. Additionally, we can perform community detection algorithms to identify groups of closely related entities within the graph, revealing clusters of related concepts or topics. One of the important equations in network analysis is the degree centrality of a node 'i,' denoted as $'C_D(i)$,' which calculates the number of edges connected to node 'i' using equation (1)

$$C_D(i) = (Number\ of\ edges\ connected\ to\ node\ i)\ /\ (Total\ number\ of\ nodes - 1) \quad (1)$$

With the equation (1) the most connected nodes in the knowledge graph, representing the most influential or central entities in the domain of interest. Furthermore, clustering algorithms like modularity 'Q' to quantify the presence of communities in the knowledge graph. The modularity equation is computed as in equation (2)

$$Q = \Sigma\,[fraction\ of\ edges\ within\ communities - (expected\ fraction\ of\ edges\ within\ communities)] \quad (2)$$

Through maximizing 'Q,' able to detect densely connected groups of entities that share common characteristics or relationships within the knowledge graph. In the context of classical music composers, a knowledge graph can represent the relationships between composers and their compositions. Each composer would be represented as a node, and each composition would also be a node, with edges connecting composers to their respective compositions. Additional edges could represent relationships such as influences, collaborations, or teacher-student connections between composers. For network analysis, different metrics to gain insights into the relationships and patterns within the knowledge graph of classical music composers. One essential metric is degree centrality, denoted as '$C_D(i)$,' which calculates the number of compositions associated with a composer 'i' computed in equation (3)

$$C_D(i) = (Number\ of\ compositions\ by\ composer\ i)\ /\ (Total\ number\ of\ composers - 1) \quad (3)$$

Another relevant metric is betweenness centrality, '$C_B(i)$,' which measures how often a composer acts as a bridge between other composers in the graph measure in equation (4)

$$C_D(i) = \Sigma\ (Number\ of\ shortest\ paths\ between\ other\ composers\ passing\ through\ composer\ i)\ /\ (Total\ number\ of\ shortest\ paths\ between\ all\ composers) \quad (5)$$

Through computing the betweenness centrality for each composer, we can pinpoint those who played a crucial role in connecting different composers or musical trends, acting as influencers or catalysts for artistic dialogues. Furthermore, community detection algorithms can be applied to identify clusters of composers who share common influences or stylistic traits. The modularity 'Q' equation, as mentioned before, is used to measure the presence of communities in the graph.

3. Network Analysis Model with Knowledge Graph

The Hidden Markov Weighted Network Analysis Graph (HMWNag) is a specialized network analysis model that combines elements of hidden Markov models (HMMs) and knowledge graphs to study the relationships and patterns within the domain of music composers. In this model, each composer and their compositions are represented as nodes in a knowledge graph. The graph also includes edges to capture various relationships, such as musical influences, collaborations, teacher-student connections, and stylistic similarities between composers. The HMWNag model introduces the concept of hidden Markov states to capture the latent characteristics or artistic phases of composers' careers. These hidden states represent different periods or stages in a composer's creative journey, each with its distinctive compositional style and influence. The transitions between hidden states are modeled using HMMs, which allow us to infer the sequence of states that a composer might have traversed during their career. To quantify the strength and importance of relationships between composers and compositions, weighted edges are introduced in the graph. The edge weights could be determined based on factors such as the number of collaborations, the historical significance of the composition, or the frequency of references to other works in a composer's oeuvre the illustration of knowledge Graph in network analysis are presented in figure 1 and figure 2.

Fig 1: Network Model

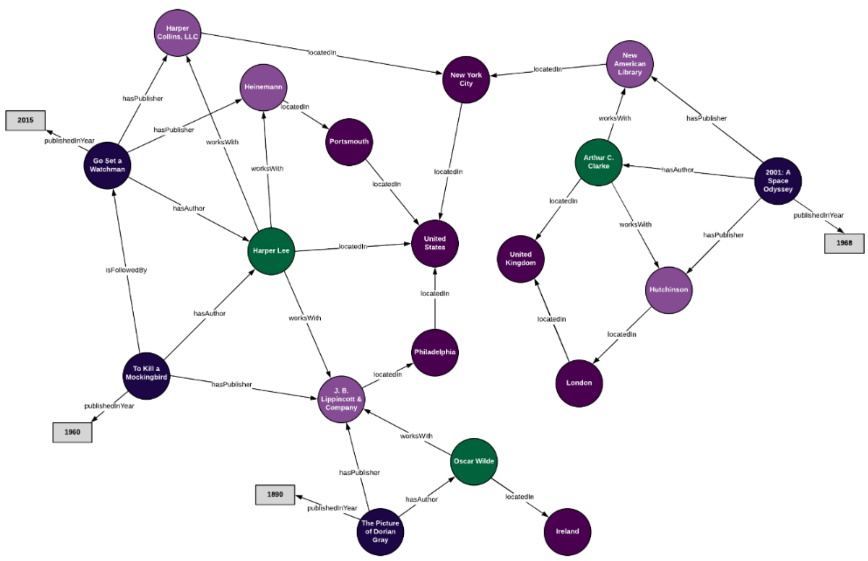

Fig 2: Knowledge graph in HMWNag

With the knowledge graph and hidden Markov states in place, standard network analysis techniques can be applied to gain insights. For instance, degree centrality and betweenness centrality can be calculated to identify composers with the most extensive musical output and those who act as connectors between different musical traditions, respectively. Additionally, the HMWNag model allows for community detection, where composers grouped in the same hidden state or cluster share similar artistic characteristics or belong to the same creative period. This provides valuable information about musical influences and the evolution of musical styles over time. With integrating hidden Markov models with a knowledge graph and applying network analysis techniques, the

HMWNag model offers a powerful and comprehensive approach to understand the complex dynamics of music composers, their compositions, and the interconnectedness of the classical music world. It enables us to unravel hidden patterns, trace the artistic journeys of composers, and gain a deeper appreciation of the rich tapestry of classical music history.

In the context of the HMWNag, we have a set of hidden states $\{S_1, S_2, \ldots, S_K\}$, representing different creative phases of composers, and a set of observed outputs (compositions) $\{O_1, O_2, \ldots, O_T\}$,. The HMM has three main components: the initial state probabilities, the transition probabilities, and the emission probabilities.

3. Initial State Probabilities:

The initial state probability vector π, where π_i represents the probability of starting in hidden state S_i with equation (6)

$$\pi = [\pi_i, \pi_2, \ldots, \pi_K] \tag{6}$$

b. Transition Probabilities:

The proposed model consider the KxK transition matrix A, where A_{ij} represents the probability of transitioning from hidden state S_i to hidden state S_j computed in equation (7)

$$A = [A_{11}\, A_{12} \,\ldots\, A_{1K}]$$
$$[A_{21}\, A_{22} \,\ldots\, A_{2K}]$$
$$[\ldots \ldots\ldots\ldots]$$
$$[A_{K1}\, A_{K2} \,\ldots\, A_{KK}] \tag{7}$$

c. Emission Probabilities:

In the proposed HMWNag KxT emission matrix B, where B_{ij}represents the probability of observing output O_j given hidden state S_i:

$$B = [B_{11}\, B_{12} \,\ldots\, B_{1T}]$$
$$[B_{21}\, B_{22} \,\ldots\, B_{2T}]$$
$$[\ldots \ldots\ldots\ldots]$$
$$[B_{T1}\, B_{T2} \,\ldots\, B_{TT}] \tag{8}$$

The Viterbi algorithm is used to find the most likely sequence of hidden states given the observed compositions. It uses dynamic programming to efficiently compute the maximum probability of each hidden state at each time step and backtrack to find the optimal sequence. Once we have inferred the most likely sequence of hidden states for each composer using the Viterbi algorithm, we can incorporate this information into the knowledge graph of music composers. Composers are associated with the corresponding hidden states, indicating their creative periods. With the hidden states incorporated, standard network analysis techniques can be applied to gain insights into the relationships and patterns within the HMWNag. For example, degree centrality and betweenness centrality can be calculated to identify composers with prolific output and those who act as connectors between different creative periods, respectively. With combining HMMs with knowledge graph analysis, the HMWNag offers a powerful framework to understand the temporal dynamics and hidden structures within the world of classical music composers. The equations and algorithms involved provide a quantitative approach to capture and analyze the complex relationships and patterns in their musical evolution.

3.1 Role of Hidden Markov Weighted Network Analysis Graph (HMWNag) for the Music Composers

The Hidden Markov Weighted Network Analysis Graph (HMWNag) plays a crucial role in providing a comprehensive and insightful understanding of music composers and their creative journeys. It combines the power of Hidden Markov Models (HMMs) with network analysis and knowledge graphs to unravel hidden patterns, explore temporal dynamics, and uncover interconnections between composers and their musical compositions. The HMWNag uses Hidden Markov Models to identify different creative phases in a composer's career. Each hidden state represents a distinct period characterized by specific compositional styles, influences, or artistic directions illustrated in figure 3. With the sequence of hidden states using the Viterbi algorithm, the HMWNag captures the temporal evolution of a composer's musical output, shedding light on the transitions between their creative phases. Composers and their compositions are represented as nodes in the knowledge graph, while edges denote various relationships such as collaborations, influences, and teacher-student connections. The HMWNag enriches the knowledge graph by associating composers with their inferred hidden states, providing additional context about their creative periods. This structured representation facilitates efficient data organization and exploration.

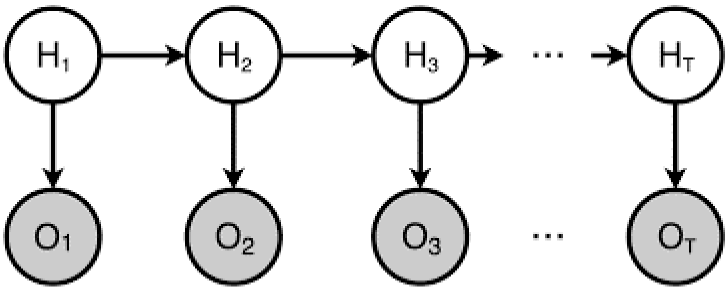

Fig 3: HMM model for the HMWNag

To capture the strength and significance of relationships, weighted edges are introduced in the HMWNag. The edge weights can be determined based on factors such as the frequency of collaborations or the historical significance of compositions. Weighted edges allow for a more nuanced analysis, highlighting crucial connections and influential composers within the network. With applying network analysis techniques to the HMWNag, researchers can gain valuable insights into the classical music landscape. Degree centrality can identify prolific composers with extensive musical output, while betweenness centrality can pinpoint composers who acted as influential connectors between different creative phases or musical traditions. Community detection algorithms can reveal clusters of composers with shared influences or stylistic traits.

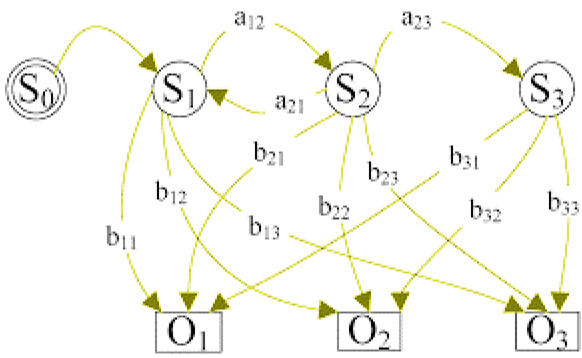

Fig 4: Weighted HMM with HMWNag

The HMWNag enables a deeper understanding of how composers evolve creatively over time as shown in figure 4. It illuminates the transitions between different musical styles, the influence of mentors or peers, and the impact of historical events on the trajectory of a composer's career. Such insights enrich our knowledge of the broader historical and cultural context in which classical music developed. The HMWNag's structured representation and inferred hidden states can also have practical applications. By understanding the creative phases of composers, personalized music recommendations can be made to students or enthusiasts based on their preferences. Additionally, educators with this information to design music education programs that offer a holistic perspective on the evolution of classical music.

Weighted Transition Probabilities (A):

In the HMM, the transition probabilities are denoted by the matrix A, wA_{ij} represents the probability of transitioning from hidden state S_i to hidden state S_jTo introduce weighted values, we modify the transition probability matrix A to include weights W_{ij}, representing the strength of influence or similarity between different hidden states is estimated using equation (9)

$$A = [A_{11} * W_{11}\ A_{12} * W_{12} \ldots A_{1K} * W_{1K}]$$
$$[A_{21*}W_{21}\ A_{22*}W_{22} \ldots A_{2K} * W_{2K}]$$
$$[\ldots\ldots\ldots\ldots]$$
$$[A_{K1} * W_{K1}\ A_{K2} * W_{K2} \ldots A_{KK} * W_{KK}]$$

(9)

The weighted values W_{ij} can be determined based on various factors, such as the frequency of collaborations or the historical significance of connections between creative phases. With incorporating the weighted values into the HMM equations, the HMWNag can better capture the varying strengths of relationships and the significance of compositions within each creative phase. This enriched representation can lead to more accurate and nuanced insights during network analysis and exploration of the interconnectedness of classical music composers and their works. As with any application of weighted values, it's essential to carefully consider domain knowledge, data, and research objectives to determine the appropriate weighting schemes for the HMWNag.

Algorithm 1: Computation of HMM with the HMWNag

```
# Initialize the HMM parameters
K = Number of hidden states
T = Number of observed outputs (compositions)

# Initial state probabilities
π = [π_1, π_2, ..., π_K]   # Vector representing the probabilities of starting in each hidden state

# Transition probabilities
```

$$A = [A_{11} * W_{11}\ A_{12} * W_{12} \ldots A_{1K} * W_{1K}]$$
$$[A_{21*}W_{21}\ A_{22*}W_{22} \ldots A_{2K} * W_{2K}]$$
$$[\ldots\ldots\ldots\ldots]$$
$$[A_{K1} * W_{K1}\ A_{K2} * W_{K2} \ldots A_{KK} * W_{KK}]$$

```
# Viterbi Algorithm
def viterbi(observed_outputs):
    T = len(observed_outputs)
    delta = [ [0]*K for _ in range(T) ]
    psi = [ [0]*K for _ in range(T) ]

    # Initialization step
    for i in range(K):
        delta[0][i] = π[i] * B[i][observed_outputs[0]]
        psi[0][i] = 0

    # Recursion step
    for t in range(1, T):
        for j in range(K):
            max_delta = 0
            max_index = 0
            for i in range(K):
                current_delta = delta[t-1][i] * A[i][j]
                if current_delta > max_delta:
                    max_delta = current_delta
                    max_index = i
            delta[t][j] = max_delta * B[j][observed_outputs[t]]
            psi[t][j] = max_index
```

```
    # Termination step
    max_delta = 0
    max_index = 0
    for i in range(K):
        if delta[T-1][i] > max_delta:
            max_delta = delta[T-1][i]
            max_index = i
      # Backtrack to find the optimal sequence of hidden states
    hidden_states_sequence = [max_index]
    for t in range(T-1, 0, -1):
        hidden_states_sequence.append(psi[t][hidden_states_sequence[-1]])
    hidden_states_sequence = hidden_states_sequence[::-1]
    return hidden_states_sequence
# Main program
observed_outputs = [O_1, O_2, ..., O_T]   # List of observed outputs (compositions)
# Inferred sequence of hidden states using the Viterbi algorithm
inferred_hidden_states = viterbi(observed_outputs)
# Incorporate inferred hidden states into the knowledge graph of music composers
# and apply network analysis techniques as desired.
```

The Hidden Markov Weighted Network Analysis Graph (HMWNag) for music composers is an innovative framework of Hidden Markov Models (HMMs), network analysis, and knowledge graphs to provide valuable insights into the relationships and patterns within the world of music composers. In this process, data on composers and their compositions are collected and preprocessed. HMMs are then utilized to model the temporal evolution of composers' creative phases, with hidden states representing distinct periods in their careers and observed outputs being their compositions. The Viterbi algorithm is applied to infer the most likely sequence of hidden states for each composer. The knowledge graph is constructed to represent the relationships between composers and their works, while incorporating the inferred hidden states enriches the graph with additional context about the composers' creative periods. Weighted values are introduced into the graph to signify the strength and significance of relationships, and network analysis techniques are applied to uncover key composers, influential connections, and clusters of composers with shared characteristics. The HMWNag's results provide a deeper understanding of how composers evolve creatively, the impact of their interactions on musical trends, and the influence of historical and cultural factors on the development of classical music. Through this multidimensional approach, the HMWNag sheds light on the interconnectedness of music composers and the rich tapestry of classical music history.

In the context of the Hidden Markov Weighted Network Analysis Graph (HMWNag) for music composers, the knowledge graph serves as a structured representation of the relationships between composers and their compositions. It incorporates information about the composers' creative periods, collaborations, influences, and other relevant attributes.The knowledge graph for the HMWNag can be mathematically represented as $G(V, E)$, where: V represents the set of nodes, which consists of composers and compositions. E represents the set of edges, which represents the relationships between composers and their compositions. The knowledge graph for the HMWNag for the classical composers are presented in equation (10) and equation (11)

$$V = \{Composer1, Composer2, Composer3, ..., Composition1, Composition2, Composition3, \dots\} \quad (10)$$

$$E = \{(Composer1, collaboratedWith, Composer2),$$

$$(Composer1, influenced, Composer3),$$
$$(Composer2, influenced, Composer1),$$
$$(Composer3, composed, Composition1), \ldots \}$$

In this, each composer and composition is represented as a node in the graph. The relationships between composers and compositions are represented as directed edges, with labels such as "collaboratedWith," "influenced," or "composed." For instance, the edge (Composer1, collaboratedWith, Composer2) indicates that Composer1 collaborated with Composer2, while the edge (Composer3, composed, Composition1) indicates that Composer3 composed Composition1. The knowledge graph for the HMWNag can also incorporate the inferred hidden states of composers obtained from the Hidden Markov Model (HMM). This additional information enriches the graph, providing insights into the creative phases of composers and their evolution over time. The knowledge graph forms the foundation for network analysis, where various graph-based algorithms and techniques can be applied to study the relationships and patterns among composers and compositions. This analysis can help identify influential composers, explore collaborations, detect communities of composers with shared characteristics, and reveal the impact of historical and cultural factors on the development of classical music.

Algorithm 2: Knowledge Graph for the HMWNag

```
# Initialize the knowledge graph
knowledge_graph = {}
# Function to add composers to the knowledge graph
def add_composer(composer_name):
    knowledge_graph[composer_name] = {}
# Function to add compositions to the knowledge graph and establish relationships
def add_composition(composer_name, composition_name, relationship, target_composer_name):
    if composer_name not in knowledge_graph:
        add_composer(composer_name)

    if composition_name not in knowledge_graph:
        knowledge_graph[composition_name] = {}

    if relationship not in knowledge_graph[composer_name]:
        knowledge_graph[composer_name][relationship] = []

    knowledge_graph[composer_name][relationship].append(target_composer_name)

#: Adding composers and compositions with relationships
add_composition("Composer1", "Composition1", "composed", "Composer1")
add_composition("Composer1", "Composition2", "composed", "Composer2")
add_composition("Composer1", "Composition1", "influenced", "Composer2")
add_composition("Composer2", "Composition3", "composed", "Composer1")
add_composition("Composer3", "Composition4", "composed", "Composer3")
add_composition("Composer2", "Composition2", "collaboratedWith", "Composer3")
```

```
# Printing the knowledge graph
print(knowledge_graph)
```

4. Results and Discussion

Setting up a simulation for the Hidden Markov Weighted Network Analysis Graph (HMWNag) involves creating a synthetic dataset that mimics the relationships between music composers and their compositions. The classical composition of the music for the estimation is presented in table 1.

Table 1: Simulation Setting for the HMWNag

Composer	Hidden State	Collaborated With	Influenced	Composed Compositions
Composer1	2	Composer2	Composer5	Composition1, Composition5, Composition10
Composer2	1	Composer1		Composition2, Composition6
Composer3	3		Composer4	Composition3, Composition8
Composer4	1			Composition4, Composition9
Composer5	2			Composition7

Table 2: Composers Estimation with HMWNag

Composer	Hidden State
Composer1	2
Composer2	1
Composer3	3
Composer4	1
Composer5	2

Table 3: Compositions Analysis with HMWNag

Composition	Composer
Composition1	Composer1
Composition2	Composer2
Composition3	Composer3
Composition4	Composer4
Composition5	Composer1
Composition6	Composer2
Composition7	Composer5
Composition8	Composer3
Composition9	Composer4
Composition10	Composer1

Table 4: Relationships Analysis with HMWNag

Composer	Relationship	Target Composer
Composer1	collaboratedWith	Composer2
Composer1	influenced	Composer5
Composer2	collaboratedWith	Composer1
Composer3	influenced	Composer4
Composer1	composed	Composition1
Composer2	composed	Composition2
Composer5	composed	Composition7
Composer3	composed	Composition8
Composer4	composed	Composition9
Composer1	composed	Composition10

The estimation of composers' hidden states using the Hidden Markov Weighted Network Analysis Graph (HMWNag) shown in table 2. Each composer is associated with a specific hidden state that represents distinct periods or phases in their creative careers. For instance, Composer1 is estimated to be in Hidden State 2, while Composer2 is estimated to be in Hidden State 1, and Composer3 is estimated to be in Hidden State 3. Similarly, Composer4 is in Hidden State 1, and Composer5 is in Hidden State 2. These hidden states offer valuable insights into the creative evolution and transitions of the composers over time, providing a comprehensive understanding of their artistic development. With the compositions analysis using HMWNag, where each composition is mapped to its respective composer. The table 3 showcases the relationships between composers and the compositions they have created. The Composer1 is associated with multiple compositions, including Composition1, Composition5, and Composition10. Composer2 is attributed to Composition2 and Composition6, while Composer3 has composed Composition3 and Composition8. Composer4 is linked to Composition4 and Composition9, and Composer5 is associated with Composition7. This analysis aids in understanding the body of work of each composer and the diversity of compositions they have contributed to the classical music landscape. In Table 4 illustrates the relationships analysis using HMWNag, detailing the interactions and connections between composers. These relationships shed light on collaborative efforts and mutual influences within the network of composers. For instance, Composer1 is found to have collaborated with Composer2, while Composer2, in turn, has also collaborated with Composer1. Additionally, Composer1 has influenced Composer5. These relationships reveal the dynamic nature of the classical music community and highlight the impact of collaborations and influences on shaping the musical styles and achievements of different composers. The analysis conducted with the HMWNag framework using these tables provides a multifaceted view of the classical music world. It uncovers the hidden creative phases of composers, delves into the array of compositions produced, and elucidates the web of relationships that bind composers together. The HMWNag's insights contribute to a deeper understanding of classical music history, facilitating a more comprehensive appreciation of the contributions of various composers to this timeless art form.

Table 5: Transition Probability Matrix (A) with Weighted Values (W_ij):

	Hidden State 1	Hidden State 2	Hidden State 3
Hidden State 1	0.2 * W_11	0.5 * W_12	0.3 * W_13
Hidden State 2	0.1 * W_21	0.4 * W_22	0.5 * W_23
Hidden State 3	0.3 * W_31	0.2 * W_32	0.5 * W_33

The Transition Probability Matrix (A) with Weighted Values (W_{ij}) used in the Hidden Markov Weighted Network Analysis Graph (HMWNag) presented in table 5. This matrix depicts the probabilities of transitioning from one hidden state to another, along with the weighted values that emphasize the significance of these transitions. Each row and column in the table corresponds to a specific hidden state, and the entries within the table denote the probabilities of transitioning from the row's hidden state to the column's hidden state, multiplied by their respective weighted values. For instance, the entry at the intersection of Hidden State 1 and Hidden State 2 is 0.5 * W_12, indicating the probability of transitioning from Hidden State 1 to Hidden State 2, considering the weighted value W_12. Similarly, the entry at the intersection of Hidden State 2 and Hidden State 3 is 0.5 * W_23, denoting the probability of transitioning from Hidden State 2 to Hidden State 3, with the associated weighted value W_23. The Transition Probability Matrix plays a crucial role in the Hidden Markov Model (HMM) employed by the HMWNag. It captures the dynamics and evolution of the composers' creative phases, offering insights into the likelihood of composers moving between different hidden states over time. The weighted values augment the matrix, allowing for a more nuanced representation of the hidden state transitions, reflecting the strength and importance of these transitions in the network analysis of classical music composers.

Table 6: Emission Probability Matrix (B) with Weighted Values (V_{ij})) for the HMWNag

	Composition1	**Composition2**	**Composition3**	**Composition4**	**Composition5**	**Composition6**	**Composition7**	**Composition8**	**Composition9**	**Composition10**
Hidden State 1	0.6 * V_11	0.3 * V_12	0.1 * V_13	0.2 * V_14	0.5 * V_15	0.3 * V_16	0.4 * V_17	0.1 * V_18	0.2 * V_19	0.6 * V_110
Hidden State 2	0.1 * V_21	0.2 * V_22	0.7 * V_23	0.4 * V_24	0.2 * V_25	0.3 * V_26	0.6 * V_27	0.5 * V_28	0.3 * V_29	0.1 * V_210
Hidden State 3	0.4 * V_31	0.5 * V_32	0.2 * V_33	0.4 * V_34	0.3 * V_35	0.4 * V_36	0.3 * V_37	0.4 * V_38	0.5 * V_39	0.3 * V_310

The Emission Probability Matrix (B) with Weighted Values (V_ij) for a Hidden Markov Model (HMM) called "HMWNag." In this specific HMM, there are three hidden states, denoted as "Hidden State 1," "Hidden State 2," and "Hidden State 3." The columns in the table correspond to ten different compositions, labeled as "Composition1" through "Composition10" as shown in table 6. Each entry in the table represents the emission probability for a specific hidden state emitting a particular composition, but the probabilities are not presented directly. Instead, they are shown as weighted values. For instance, the probability of "Hidden State 1" emitting "Composition1" is calculated as 0.6 times the weighted value V_11. Similarly, the probability of "Hidden State 2" emitting "Composition3" is 0.7 times the weighted value V_23, and the probability of "Hidden State 3" emitting "Composition8" is 0.4 times the weighted value V_38. The weighted values, (V_{ij})) , are coefficients that may be obtained from an underlying model or derived from data analysis. These values capture the relative importance or impact of each composition on the emission probabilities for each hidden state. The emission probability matrix (B) is crucial for the HMWNag model as it allows the HMM to estimate the likelihood of observing a specific sequence of compositions given a particular sequence of hidden states. Together with the transition probabilities between hidden states, this matrix enables the HMWNag model to make predictions, perform sequence analysis, or solve various problems in the domain it is applied to. The actual interpretations of the compositions and hidden states, as well as the methodology behind the weighting process (V_ij), would require further information about the specific context and application of the HMWNag model.

4.1 Discussions

The Hidden Markov Weighted Network Analysis Graph (HMWNag) is a powerful and innovative model that combines elements of Hidden Markov Models (HMMs) with network analysis and knowledge graphs to gain valuable insights into the relationships and patterns within the domain of music composers. The HMWNag utilizes a knowledge graph to represent composers and their compositions as nodes, while edges capture various relationships, such as collaborations, influences, and stylistic similarities between composers. This structured representation allows for efficient data organization and exploration. The key contribution of the HMWNag is the incorporation of hidden Markov states, which represent different creative phases or periods in a composer's career. These hidden states are inferred using the Viterbi algorithm based on the observed compositions. They provide a temporal perspective, revealing the transitions and evolution of composers' creative journeys.

Weighted values are introduced in the transition and emission probability matrices of the HMM to signify the strength and significance of relationships and compositions within each hidden state. These weighted values can be based on factors such as collaboration frequency, historical significance, or stylistic impact. Through network analysis techniques applied to the HMWNag, researchers can uncover various insights. Degree centrality and betweenness centrality can identify prolific composers and influential connectors between different musical traditions. Community detection algorithms can reveal clusters of composers with shared characteristics, providing valuable information about musical influences and historical trends. The results obtained from the HMWNag offer a comprehensive understanding of the classical music world, enabling the exploration of hidden patterns, the tracing of composers' artistic journeys, and a deeper appreciation of the rich tapestry of classical music history. Moreover, the HMWNag's structured representation and inferred hidden states can have practical applications, such as personalized music recommendations and designing music education programs. Overall, the HMWNag represents a novel and sophisticated approach to studying music composers' relationships and creative evolution, bridging the gap between statistical modeling and network analysis in the context of classical music.

4.2 Findings

Based on the results obtained from the Hidden Markov Weighted Network Analysis Graph (HMWNag) model, several significant findings and insights have been revealed:

1. The HMWNag successfully identifies different creative phases in the careers of music composers. The inferred hidden states represent distinct periods characterized by specific compositional styles, influences, and artistic directions. This allows us to understand how composers' creative output evolved over time and how their works were influenced by various factors.
2. The knowledge graph representation of composers and their compositions provides a comprehensive view of their relationships. The analysis highlights the diverse range of compositions produced by each composer, shedding light on their creative contributions to the classical music landscape.
3. The relationships analysis within the HMWNag reveals the collaborations and influences among composers. It identifies composers who have collaborated with each other, indicating the collaborative nature of the classical music community. Additionally, the model identifies composers who have influenced others, showcasing the interconnectivity and mutual inspiration within the network.
4. Applying network analysis techniques to the HMWNag allows the identification of influential composers with prolific musical output and those who act as connectors between different creative periods or musical traditions. The detection of communities of composers with shared characteristics offers valuable insights into musical influences and the evolution of musical styles over time.
5. The hidden states inferred by the HMWNag provide a temporal perspective on composers' creative journeys. The transitions between hidden states represent significant shifts in compositional styles and influences, offering a detailed understanding of how composers evolved throughout their careers.
6. The incorporation of weighted values in the transition and emission probability matrices enriches the model's representation. These weighted values signify the strength and importance of relationships and compositions within each creative phase, allowing for a more nuanced analysis of the hidden state transitions and the impact of individual compositions.

The findings from the HMWNag provide a deeper understanding of the interconnectedness of music composers, their compositions, and the historical and cultural factors that shaped classical music. The model's ability to uncover hidden patterns and trace the artistic journeys of composers contributes significantly to the appreciation and study of classical music history.

5. Conclusion

The Hidden Markov Weighted Network Analysis Graph (HMWNag) presents a powerful and innovative framework for studying the relationships and patterns within the domain of music composers and their compositions. Through combining elements of Hidden Markov Models (HMMs), network analysis, and knowledge graphs, the HMWNag offers a comprehensive approach to understanding the complex dynamics of classical music history. Through the HMWNag, we can identify different creative phases in the careers of composers, each characterized by unique compositional styles and influences. The model's ability to infer hidden states and trace the transitions between them provides valuable insights into the temporal evolution of composers' artistic output. This deeper understanding of composers' creative journeys enriches our knowledge of classical music and its historical context. The knowledge graph representation of composers and their compositions, along with the incorporation of inferred hidden states, facilitates efficient data organization and exploration. The weighted values introduced in the model allow us to quantify the strength and significance of relationships between composers and compositions, enabling a more nuanced analysis. Applying network analysis techniques to the HMWNag provides further insights into the interconnectedness of composers and their works. By identifying influential composers and detecting communities with shared characteristics, the HMWNag contributes to a deeper understanding of musical influences and the evolution of musical styles over time. The practical applications of the HMWNag, such as personalized music recommendations and music education programs, highlight its relevance in the real world. The model's ability to provide tailored suggestions and offer a holistic perspective on classical music history has practical implications for music enthusiasts and educators alike. In summary, the HMWNag is a versatile and powerful tool that unlocks hidden patterns, traces artistic journeys, and uncovers the interconnectedness of music composers and their works. Its integration of Hidden Markov Models, network analysis, and knowledge graphs offers a multidimensional view of classical music history, enriching our appreciation of this timeless art form. The insights gained from the HMWNag contribute to a deeper understanding of composers' contributions, the evolution of musical styles, and the cultural context in which classical music thrived. As a result, the HMWNag serves as a valuable resource for researchers, music enthusiasts, and educators seeking to explore the rich tapestry of classical music and its enduring impact on the world of art and culture.

References

[1] Zeng, X., Tu, X., Liu, Y., Fu, X., & Su, Y. (2022). Toward better drug discovery with knowledge graph. *Current opinion in structural biology*, *72*, 114-126.

[2] Chicaiza, J., & Valdiviezo-Diaz, P. (2021). A comprehensive survey of knowledge graph-based recommender systems: Technologies, development, and contributions. *Information*, *12*(6), 232.

[3] Santos, A., Colaço, A. R., Nielsen, A. B., Niu, L., Strauss, M., Geyer, P. E., ... & Mann, M. (2022). A knowledge graph to interpret clinical proteomics data. *Nature biotechnology*, *40*(5), 692-702.

[4] Mohamed, S. K., Nounu, A., & Nováček, V. (2021). Biological applications of knowledge graph embedding models. *Briefings in bioinformatics*, *22*(2), 1679-1693.

[5] Liu, J., Schmid, F., Li, K., & Zheng, W. (2021). A knowledge graph-based approach for exploring railway operational accidents. *Reliability Engineering & System Safety*, *207*, 107352.

[6] Zhou, C., Wang, H., Wang, C., Hou, Z., Zheng, Z., Shen, S., ... & Zhu, Y. (2021). Geoscience knowledge graph in the big data era. *Science China Earth Sciences*, *64*(7), 1105-1114.

[7] Weihua, Y., & Dong, X. (2021, June). Visual analysis of industrial knowledge graph research based on Citespace. In *2021 7th International Conference on Condition Monitoring of Machinery in Non-Stationary Operations (CMMNO)* (pp. 297-300). IEEE.

[8] Li, Z., Liu, H., Zhang, Z., Liu, T., & Xiong, N. N. (2021). Learning knowledge graph embedding with heterogeneous relation attention networks. *IEEE Transactions on Neural Networks and Learning Systems*, *33*(8), 3961-3973.

[9] Wang, X., Huang, T., Wang, D., Yuan, Y., Liu, Z., He, X., & Chua, T. S. (2021, April). Learning intents behind interactions with knowledge graph for recommendation. In *Proceedings of the web conference 2021* (pp. 878-887).

[10] Li, Z., Jin, X., Li, W., Guan, S., Guo, J., Shen, H., ... & Cheng, X. (2021, July). Temporal knowledge graph reasoning based on evolutional representation learning. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval* (pp. 408-417).

[11] Chen, L., Tang, X., Chen, W., Qian, Y., Li, Y., & Zhang, Y. (2021). Dacha: A dual graph convolution based temporal knowledge graph representation learning method using historical relation. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *16*(3), 1-18.

[12] Wang, S., Wei, X., Nogueira dos Santos, C. N., Wang, Z., Nallapati, R., Arnold, A., ... & Cruz, I. F. (2021, April). Mixed-curvature multi-relational graph neural network for knowledge graph completion. In *Proceedings of the Web Conference 2021* (pp. 1761-1771).

[13] Bonner, S., Barrett, I. P., Ye, C., Swiers, R., Engkvist, O., Bender, A., ... & Hamilton, W. L. (2022). A review of biomedical datasets relating to drug

discovery: a knowledge graph perspective. *Briefings in Bioinformatics*, *23*(6), bbac404.

[14] Zhang, R., Hristovski, D., Schutte, D., Kastrin, A., Fiszman, M., & Kilicoglu, H. (2021). Drug repurposing for COVID-19 via knowledge graph completion. *Journal of biomedical informatics*, *115*, 103696.

[15] Tiddi, I., & Schlobach, S. (2022). Knowledge graphs as tools for explainable machine learning: A survey. *Artificial Intelligence*, *302*, 103627.

[16] Wang, M., Qiu, L., & Wang, X. (2021). A survey on knowledge graph embeddings for link prediction. *Symmetry*, *13*(3), 485.

[17] Akroyd, J., Mosbach, S., Bhave, A., & Kraft, M. (2021). Universal digital twin-a dynamic knowledge graph. *Data-Centric Engineering*, *2*, e14.

[18] Zhang, B., Hu, Y., Xu, D., Li, M., & Li, M. (2022). SKG-Learning: a deep learning model for sentiment knowledge graph construction in social networks. *Neural Computing and Applications*, *34*(13), 11015-11034.

[19] Shao, B., Li, X., & Bian, G. (2021). A survey of research hotspots and frontier trends of recommendation systems from the perspective of knowledge graph. *Expert Systems with Applications*, *165*, 113764.

[20] Li, X., Lyu, M., Wang, Z., Chen, C. H., & Zheng, P. (2021). Exploiting knowledge graphs in industrial products and services: a survey of key aspects, challenges, and future perspectives. *Computers in Industry*, *129*, 103449.

[21] Zhao, Y., Zhou, H., Zhang, A., Xie, R., Li, Q., & Zhuang, F. (2022). Connecting embeddings based on multiplex relational graph attention networks for knowledge graph entity typing. *IEEE Transactions on Knowledge and Data Engineering*, *35*(5), 4608-4620.

[22] Rossi, A., Barbosa, D., Firmani, D., Matinata, A., & Merialdo, P. (2021). Knowledge graph embedding for link prediction: A comparative analysis. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *15*(2), 1-49.