Modeling Diffusion Problems via Graph-based Structure

Yoro R.E.
Department of Computer
Science,
Babcock University, Ilishan
Remo, Ogun State, Nigeria.

Adekunle Y.A.
Department of Computer
Science,
Babcock University, Ilishan
Remo, Ogun State, Nigeria.

Ebiesuwa Seun
Department of Computer
Science,
Babcock University, Ilishan
Remo, Ogun State, Nigeria.

ABSTRACT

Academics carry out research studies periodically that they need to report. Various problem arise during the course of these studies ranging from proper comprehension of the task or domain problem, its sensitivity and failure analysis via model creation, its visual result representation and its other ecstatic that help the proposed model to be easily readable, understandable and implemented. In modeling, a researcher may seek underlying relations or data feats of interest between observed versus computed data and/or values, from statistical perspective or vantage point. This study aims to discuss and unveil modeling a problem from a graph-based perspective as well as highlighting some of the feats for analysis.

Keywords

abstract, structure, evidence, rationale, research

1. INTRODUCTION

The future continually leave us curious as we seek to control every frontier of our daily activity and dealings in the society. Knowledge of such future will grant us insight and help us plan the magnitude of other events that are dependent on it. A farmer may inquire how temperature and rainfall affects his farm products. This will shed great light and grant the farmer insight to what factors need to be changed, and in turn – change his farm's output; Whereas, the immediate society may be more concerned about other impact of temperature and rainfall such as erosion etc. Chaotic, complex and dynamic events if properly predicted will help experts make better decisions about the future from observed data.

These results can be made more manifest via models and modeling. Thus, a model is tool that provides knowledge statement of the future we sought and real-time prediction as output that helps us to plan in time, the scale or magnitude of a probable event as it reaches maximum (Ojugo, 2012). Though, the reliability of models can be questioned as its results are seen as prophesies based on certain probability (Macy and Willer, 2002).

1.1 The Concept of Models

Models are a collection of finite objects along with its consequent relations or operations that are defined upon it. It is also a representation of entities and the relationships that ensue between them. The model simply learns by imitating and observing the thought process of how its entities work or behaves (Ojugo et al, 2015a) – and can be exemplified in a flock of bird flying in tight formation. These birds will collectively form an image with a goal to move as a single organism.

The flock choreographs in such grace even with no group-leader bird. Instead, each bird reacts in movement of its immediate neighbors, to result in hypnotic-pattern-rhythm and highly-nonlinear (Ojugo et al, 2015b). Modeling such elegance that is not governed by any system – can be quite tedious due to its dynamism, complexity and nonlinear nature.

Yet, we can model it as an aggregation of local feats interactions via 3-rules: (a) **separation**: each bird does not get too close to another, (b) **alignment**: each bird matches its direction and speed to nearest bird, and (c) **cohesion**: each bird stores in memory perceived center of the flock and its immediate neighbor (Reynolds, 1987). Thus, each bird is an agent with local feats interaction to yield a highly realistic flight formation via simple rules – to result in the theory of Agent Based Modeling – ABM (Ojugo et al, 2014).

ABM is best suited for graph-based problem/task domain designs. The graph-design allows the collection of entities to be represented as nodes (agents or actors) so that they are defined as highly self-organized, path-dependent, dynamic, autonomous and complex-in-nature such that they all have features that are local (innate) within them. As these agents interact, these feats eventually emerge and spread through the graph- structure as a global feat. These interactions are in response to varying external and internal influences received – as thus, and results in a global pattern (Axelrod, 1997). Various studies have harnessed the many potential of ABM as tools in relational method modeling with its fundamental focus on emergent social structure and social order via local interaction – because, it provides these position: (a) provides a theoretical framework of dynamic social graphs shaped via interactions of actor, and (b) it puts to the test, all social-learning theories that manipulate feats of the graph-structure such as its topology, stratification and others (Simon, 1998; Kaufman, 1996; Macy and Willer, 2002).

1.2 Types of Models

Models are of two types:

a. *Knowledge driven* models, governed by dynamic laws have their spatial/frequency distributions results of its predicted values obtained via observed-historic data that is used to calibrate and test the model. Its usage focuses on formulation of new parameters to help predict the model's internal process. Its major challenges includes (not limited to): (a) parameter choice selection, (b) its results must be validated, (c) heterogeneity nature of its historic data to be used, and (d) limited data availability due to the complex, dynamic and non-linear nature of most tasks (Ojugo, Yoro et al, 2013a).

b. *Data-Driven* models explore intelligent heuristic rules and search, aimed at learning data feats of interest as the data converges in time. Its spatial and statistical distribution is based on dynamic laws of evolution that allows entrant of new data feats that may not have been present at its outset, to be injected as model progresses. Thus, helps it to yield a solution of high quality (even with noise, imprecision, ambiguity and partial truth applied to its input). It rather focuses more on model validation than the results achieved, with a view to provide insight about the domain problem – even with

limited data to perform the much required tests (Ojugo et al, 2013a).

In summary, ***reasons*** for modeling include:

a. **Robustness**, which allow the researcher to tune up the model in anticipation or in view of the real system to help estimate the system's effectiveness even with noise implementation at its data input.

b. **Continuous adaptation** allows from statistics point of view, agents that are void of local minima, allows the introduction of new and random immigrants not present from the outset with high diversity to slow convergence in the search space as well as balances data exploitation and exploration so that in learning the properties of change, it yields an accordingly biased solution).

c. **Flexibility** – decisions made with uncertainty has its impacts in a system's future state. Thus, optimization aims to predicts the future needs with an algorithm that focuses on both its objective function, to make the system flexible and facilitate adaptation (if necessary) with the ease of black-box integration.

1.3 Properties of Agents in a Graph-Based Model

Properties of an agent in a graph-based agent modeling (ABM) include (Ojugo et al, 2015a,b):

a. *Agents[1] are autonomous* – System may not directly be modeled as globally-integrated entity; But, they emerge as a bottom-up pattern, coordinated not by centralized authority (as in environmental constraints) but via local interactions in autonomous decision makers known as self-organization.

b. *Agents follow rules* – Global pattern does not reflect cognitive complexity in an agent's ability. Agents build up behaviors by obeying rules as norms, morals and social habits. These rules yield the un-obvious global patterns that are sometimes very difficult to understand. The complexity of our behavior thus, is a reflection of our environmental complexity. Thus, ABMs explore simple rule set of behavioral assumptions required to generate a pattern of explanatory interest to influence the dynamics of population as well as determine their behavior from local feats and processes that are innate in agents as they interact with their neighbors.

c. *Agents Evolve* – As agents learn, their behavior/actions evolve allowing them respond appropriately to the task so as to meet specific needs/requirements as well as reach their desired state and set objectives.

d. *Agents are interdependent* – As they build social habits (such as affection, imitation and set-goals etc), agents influence themselves in response to signals that they receive. An agent's behaviors can change an aspect of the environment, which in turn affects the behavior of other agents. Thus, aftermath of each agent's decisions, depend in part on the choices of others.

e. *Agents are adaptive* through interaction to generate a system as *individual* and *population* learning, imitating and replicating; they may not calculate most efficient action. As individuals, they learn via processes such as reinforcement or back-propagation, which in turn alters the probability distribution of behaviors competing for attention within the system. As population, they learn via evolution process of selection, imitation, and social influence to alter frequency distribution of agent-types competing for reproduction within a population.

f. *Agents have Property* or characteristics that describe them. These properties may be intrinsic (unchanged, but we can change their values in time); while others are re-evaluated as agent interact within the system.

g. *Agents have Lifetime* – Agents are created as they enter the environment and are destroyed when they expire or die. A system has some form of control over the *lifetime* of an agent as the agent goes through its corresponding processes. Each agent is allotted a space and with more agents introduced – the more positions and memory is used up and the greater its complexity and performance of the system. Stochastic models aim to maximize an agent's potentials by creating and destroying them on a need-basis (Ojugo, 2008).

h. *Agents are instantiated* – Each agent belongs to a class – so that agents of the same class can perform the same action. Class determines what properties and methods are available to an agent. Each time an agent is thus created, it brings forth an instance of the class such an agent belongs to.

i. *Agents have scope* that defines the underlying structure of the system an agent has access to. They are strong, weak and no-ties defined through local interaction that ripples as patterns into a global effect on the system.

j. *Agents have Methods* as actions an agent can perform. The actions are determined by the nature of the task. Agents defined in a class can all perform same actions.

k. *Agents have Events* as reactions that an agent receives in response from its environment (and previous actions taken). Agents have events associated with them.

l. *Agents are encapsulated* – Is the inability of the system to define how feats are innate in its agents, irrespective of the system being able to redirect these feats. Thus, agents' feats are transparent to the system and we often refer to as the agent goes through its corresponding processes. Each agent is allotted a space and with more agents introduced – the more positions and memory is used up and the greater its complexity and performance of the system. Stochastic models aim to maximize an agent's potentials by creating and destroying them on a need-basis (Ojugo, 2008).

m. *Agents are instantiated* – Each agent belongs to a class – so that agents of the same class can perform the same action. Class determines what properties and methods are available to an agent. Each time an agent is thus created, it brings forth an instance of the class such an agent belongs to.

n. *Agents have scope* that defines the underlying structure of the system an agent has access to. They are strong, weak and no-ties defined through local interaction that ripples as patterns into a global effect on the system.

o. *Agents have Methods* as actions an agent can perform. The actions are determined by the nature of the task. Agents defined in a class can all perform same actions.

p. *Agents have Events* as reactions that an agent receives in response from its environment (and previous actions taken). Agents have events associated with them.

q. *Agents are encapsulated* – Is the inability of the system to define how feats are innate in its agents, irrespective of the system being able to redirect these feats. Thus, agents' feats are transparent to the system and we often refer to agents in a process as black-box tools (tools we can use without knowing its inner workings). The merit of encapsulation includes: (a) speeds up development processes, (b) allows reuse of agent in similar systems, and (c) makes transparency more effective.

r. *Agents have states* – The state of an agent is a set of all the values defined by all its properties. A person can be fair in complexion, 6.8feets tall, short hair, etc.

1.4 Issues with Agents in Graph-Based Models

Macy and Willer (2002) ABMs converge to two problems (both complementary to explain clustering in social ties) as:

1. *Emergent Social Structure* – Agents (via their behavior) traverse a physical space in response to influence and selection pressures. They may start off the same as the crowd; But, eventually change their behavior to avoid being different or isolated. Thus, they conform rather than be homogenous, and aggregates to a global pattern of cultural difference, stratification and homophilous clustering in graphs. Modeling such process starts off as a heterogeneous population to end in convergence, coordination, diffusion, and collapse of personal norms, institutions, beliefs, innovations, etc.

2. *Emergent Social Order* – Agents' egoistic adaptation often leads to successful collective action without the altruism or global (top-down) imposition of control. A key finds in many studies, is that cooperation achieved via social order and collective action as a function of trust, largely depends decisively on social interaction that are embedded therein in the social system.

1.5 Model Validation and ABM

Much confusion in result validation via different process, arise from varying semantics – since validation notes the confidence level in a model's ability to represent a problem. It emphasizes a model is not expected to be absolutely valid and compares predicted versus observed values in relation to the model design/calibration dataset (De Vos et al, 2005). There are 3-types of validation (Gaas, 1983) which includes: (a) *Replicative* validation ensures that prediction agrees with observed dataset used in its design and parameter estimates, (b) *Predictive* validation ensures model accurately predicts variable convergence in time, not used in calibration so that the model is structurally valid as it reflects the behaviour of real system, and (c) *Specific* validation ensures a generally accepted standard for model test and validating is adequate for a special purpose. Thus, some models maybe unrealistic to emphasize parameters for calibration and ad-hoc feats to will make validation less rigorous – so that even inadequate models will likely pass tests (Gaume and Gosset, 2003).

Some researchers used history-match to distinguish data used in training versus testing. Others used corroboration to describe extent a hypothesis passed a test and to express limited acceptance of the model's result. Valid, from *validus* implies powerful, well-grounded, defensible and sound. A more accepted semantics is needed to describe qualification of prediction. 'Verification' is inappropriate (used in some studies) to note establishment of truth as hardly possible both in science and absolutely not in modeling. Thus, the more concise use of validation here – is to imply *predictive* validity as proposed by Gaas (1983).

2. GRAPH AND ITS STRUCTURE

A graph is a structure consisting a set of vertices (nodes, agents, actors, entities) connected together by corresponding set of edges (relations). It helps to models task in a variety of task domain (not limited to) computing, mathematics, sociology, biology, engineering etc – by modeling various interactions within a system, in which agents represent nodal entities; while its edges represent their defined relations and interactions (via clusters formation) to effect a global feat. A powerful role of graphs is to bridge local feats that exist in nodes as they blossoms into global patterns to help explain how nodes and their corresponding relations impacts and ripple through the entire system. Each node helps to shape graph's evolution in time, adapting themselves to various forms (Ojugo et al, 2014).

Mathematically, a graph G = (V,E) has that each node $x \in$ V and each $m \in$ E. Edges can be directed from a node to another. The node from which an edge points away from is the source/predecessor, whereas that which an edge points to is destination/successor. Each node has a corresponding set of neighbors to which it is either linked to or is isolated from. The links can be weak, strong or isolated, in terms of relations status as measured through *dyads D* (West, 2001; Izquierdo and Hanneman, 2008; Ojugo et al, 2015a).

The edges describe what relations exist between the nodes. A network has nodes with corresponding one or more kinds of relations. Graph with a *single* type of relation amongst its nodes is called *simplex*; while those of more than one kind of relationship are *multiplex*. Multiplex graphs are analyzed differently and via different techniques (one for each type). An *undirected* relation is seen among siblings indicating a relation of co-occurrence, co-presence or bonded-tie where nodes are of same level); while, a *directed* relation is seen between a parent and child as data originates from parent as source node and reaches child as target node (in an instance) via an arc. This is called a ***digraph*** (directed graph) and these directed links may be reciprocated to indicate that both nodes can be source cum target at any instance. Edges may be weighted to indicate *binary* (presence or absence of relations), *signed* (negative, positive or no relations), *ordinal* (strong, weak or isolated relationship), or *numeric value* or *cost* scaled on ratio (Diestel, 2005).

2.1 Connectivity in Graphs

Nodes in a graph are connected as thus:

1. ***Walk*** – is a sequence of nodes and relations (edges) that begin and ends with nodes. A closed walk start and end with same node. Length of a walk is number of edges it uses. Examples of walks and its length for fig 1 include:

   {A, B, D} = 2
       {A, C, D} = 2

   {A, B, C, D} = 3
       {A, C, B, D} = 3
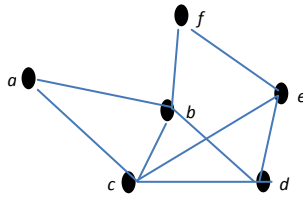
   {A, B, C, B, D} = 4

   {A, B, C, E, D} = 4

Fig 1: An Undirected G

{A, B, F, E, C, D}=5

{A, C, B, F, E, C, D} = 6

2. **Cycle** is a special, restricted walk used to examine the neighborhoods of nodes (points adjacent to a particular node). A cycle is a closed walk of 3 or more nodes, all of whom are distinct except for the origin or destination nodes. The cycle beginning and ending with A is {A, B, C}. The cycles beginning and ending with B are: {(B, A, C), (B, C, D), (B, D, E, F), (B, A, C, E, F) and (B, A, C, D, E, F)} respectively. And so on.

3. **Trail** between nodes is any walk that includes any given relation at most once. The same nodes, however – can be part of a trail multiple times. The length of a trail is the number of relations in it. All trails are walks, but not all walks are trails. A trail that begins and ends with the same node is **closed** trail. There are a number of trails between A and D. All tracings as listed above, are trails except {A, B, C, B, D} – which is a walk but not a trail. This is because the relation BC is used more than once.

4. **Path** – is a walk in which each node (or relation) in the graph is used at most once. Also, it is a simple path in which every vertex is incident to at most two edges. All paths are trails and walks; But, not all trails/walks are paths. From fig 1, example of paths includes: {A,B,D}, {A,B,C,D}, {A,C,D}, {A,B,F,E,D}. The length of a path is number of relations it has. The length of shortest path between two nodes is geodesic distance between them. The geodesic distance between A-D is shortest distance = 2 in {A,B,D} and 2 {A,C,D} respectively.
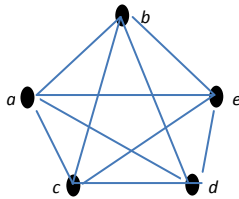

Fig 2: Undirected G with K₅ complete graph. If G is subgraph, its vertices forms

5. **Tree** – is a connected *acyclic* simple graph. The vertex of degree 1 is called a leaf or pendant vertex. An edge incident to a leaf is a leaf edge. A non-leaf vertex is an internal vertex. If a tree has a vertex, distinguished as its root, the tree is called *rooted*. Rooted trees are often treated as directed acyclic (in which case, the root is called its source node or delivery point) – with all its edges pointing away from the root. A subtree of the tree T is a connected subgraph of T. A forest is an acyclic simply graph. A subforest of the forest F is a subgraph of G. a spanning tree is a spanning subgraph that is a tree. Every graph has a spanning forest. But only a connected graph has a spanning tree. A special kind of tree called a star is $K_{1,k}$

6. **Cliques** – A complete graph K_n of order n is a simple graph with n-nodes in which every vertex is adjacent to every other node denoted by $K_n = \frac{n(n-1)}{2}$ edges (which is all possible choices of vertices pairs) as in Fig. 2. A *clique* is a set of pairwise adjacent vertices. A k-clique is a clique of order *k*. In fig 1, {A,B,C} is a triangular 3-clique. A maximal clique is a clique that is not a subset of any other clique. The clique number ω(G) is the order of a largest clique in G.

2.2 Graph Measures

To analyze graphs and seek data feats of interest, the basic measures (both at local and global pattern) include:

1. **Degree of Distribution** is number of edges incidence on any node. It helps us know how and where to add another node. If k is the number of edges incident on a node, and P(k) is probability that the degree of a randomly chosen node equals k. We may add a new node *x* with preference based on probability proportional to number of existing edges k, that node *x* has. Thus, the connectivity of a node x to a node y approximately follows the rule as in Eq. 1:

$$P(links\_to\_node\_x) = \frac{k_x}{\sum_y k_y} \quad (1)$$

Degree of distribution of *x* in *N*(G) denoted as *deg*(x) is total number of edges at *x* or the number of neighbors of *x* as in Eq. 2 (Mason and Verwoerd, 2006):

$$deg(x) = |N(u)| \quad (2)$$

For a digraph G, the number of head endpoints adjacent to a node is its in-degree; while, the number of tail endpoints is its out-degree. Thus, in-degree of a node is the number of edges that points to a node *x*, denoted as $deg_{in}(x)$ and out-degree is number of edges that points away from node *x* denoted by $deg_{out}(x)$. Also, in-degree is denoted as $deg^-(x)$ and out-degree is denoted as $deg^+(x)$. A node with $deg^-(x) = 0$ is a **source** node; while, a node with $deg^+(x) = 0$ is the destination node given by Eq. 3:

$$\sum_{v \in N} deg^+(x) = \sum_{v \in N} deg^-(x) = |A| \quad (3)$$

If for every node v ∈ N, $deg^-(x) = deg^+(x)$ – then, we have a balanced digraph. Thus, the nodes for both figures of 10 – so that we have v₁, v₂,…vₙ, then adjacency matrix A of G is given by Eq. 4 as:

$$a_{xy} = \begin{cases} 1 & if \ v_1 v_2 \in E(G) \\ 0 & if \ v_1 v_2 \notin E(G) \end{cases} \quad (4)$$
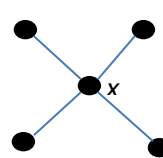

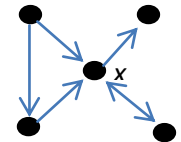**Fig 3a: Undirected G**   **Fig 3b: Directed G**

Fig 3a yields the Adjacency Matrix $A = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$

Fig 3b yields Adjacency Matric: $A = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$

For the undirected G, $deg(x) = 4$; while for the directed graph – we have that $deg_{in}(x) = 3$ and $deg_{out}(x) = 2$.

Degree of Distribution has become a prominent feat in the measure of network topology and graph models. It is a more reliable property for numerical estimation. Its visual representation of the properties of the network is achieved via 3-means namely: (a) each node is ranked according to their degree and values of the each node's degrees plotted against the ranks of each vertex, (b) plot a histogram of the nodes of G according against their degrees using a logarithmic scale, and (c) plot degree of the nodes sorted versus their degree distribution P(k) or their cumulative degree distribution P_c(k).

2. **Paths/Average Path Length** – If x,y are nodes in G, the sequence of nodes given by $x = v_1, v_2, \ldots, v_n = y$ is such that for $i = 1, 2, \ldots, k$-1, the path length k – 1 from x-to-y is given as Eq. 5:

$$\begin{aligned} (i) & v_i v_{i+1} \in E(G) \\ (ii) & v_i \neq v_j \ for \ i \neq j \end{aligned} \quad (5)$$

The geodesic distance δ(x,y) is the length of the shortest path from *x-to-y*. If no such path exists, then δ(x,y) = ∞. If for every pair of vertices x,y ∈ N(G), there is some path from x to y – then, it is G is connected. The average path length and diameter of s graph G is defined as the average and maximum value of δ(x,y) taken over all pairs of distinct nodes x,y in N(G) which are connected by atleast one path. Thus, for fig 4, the path length equals 3.
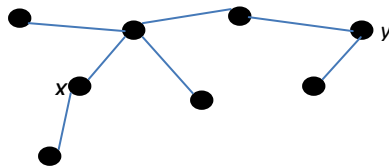

Fig 4: Undirected G

3. **Clustering Coefficient:** Clusters are formed by existence of cliques. Thus, clustering coefficient is the measurement that shows the tendency of a graph to be divided up into clusters. A cluster is a subset of vertices that contains lots of edges connecting vertices to each other. If x is a vertex with degree deg(x) = k. Its clustering coefficient of x for an undirected graph G, is given below such that for the node x of degree k with e-edges between these k-neighbors of x in G, we have that Eq 6:

$$C_x = \frac{2e}{k(k-1)} \quad (6)$$

C_x measures the ratio of the number of edges between the neighbors of x to the total possible number of such edges, which is $\frac{k(k-1)}{2}$. It takes a value $0 \leq C_x \leq 1$ and is the *local clustering coefficient*. Conversely, the *average* clustering coefficient of the graph G is defined as Eq. 7:

$$C_{average} = \frac{1}{N} \sum_{x=1}^{N} \frac{E_x}{k_x(k_x - 1)} \quad (7)$$

where N = |V| is number of vertices. The closer the local clustering is to 1, the more likely G form clusters, and the more the number of cliques.
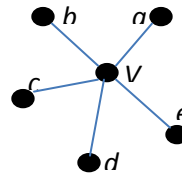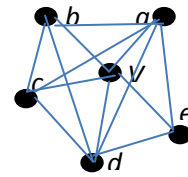

Fig 4a: **Fig 4b:**

From fig 4a, node V is a hub with a clustering coefficient C = 0. Conversely in fig 4b, node V does same function with a high clustering coefficient. The maximum number of all potential connections is given by Eq 8:

$$E_{max} = |V| * \left(\frac{|V| - 1}{2} \right) \quad (8)$$

If |V| = 5 (as the number of the neighbors of node V).

$$E_{max} = 5 * \left(\frac{5 - 1}{2} \right) \equiv 10$$

Neighbors of node V are connected with 8 edges between each other in the connection is: {(a,b), (b,c), (c,d), (d,e), (e,a), (a,c), (a,d) and (b,e)}. Thus, E = 8 connections and the clustering coefficient is given by:

$$C_x(k) = \left(\frac{E}{E_{max}} \right) = \frac{8}{10} = 0.8$$

The more complex a graph, the more significantly higher will be the average clustering coefficients, as compared to random graphs (helping to prove the modular nature of such graphs). Clustering coefficient in analysis, helps us to classify observations into two/more mutually exclusive unknown groups based on some combination of variables. Thus, clusters analysis is best employed in unsupervised classification (Duda et al, 2001 and Jain et al, 1999) – whose purpose is to group different objects together by observing common feats within the system. In biological networks, it helps us identify similar biological entities and units such as protein, enzymes and genes that are homologous in different organisms or that belong to same complex genes that are co-expressed (Borate et al, 2009; Perkins and Langston, 2009; Pavlopoulos et al, 2011).

Due to the difficulty of the analysis and prediction of elements in a system that are based on observation of the behaviors and properties of other elements in the same system – clustering algorithms has been grouped into various techniques namely: (a) *exclusive* (data are grouped so that a certain element is assigned to only one group exclusively, (b) *overlapping* (uses fuzzy sets to cluster data so that each point may belong to two or more clusters with different degrees of membership, (c) *hierarchical* (data is organized in hierarchies based on the union of two nearest clusters. It is best for microarray and sequence analysis, and (d) *probabilistic or spectral* where data is grouped through profiling (Ojugo et al, 2015a).

2.3 Graph Models and Network Topology

Graphs are basically divided into three (3) models namely:

1. **Erdos-Renyl (Random) model** describes the feats of a random graph in which nodes *n* are connected to each other by randomly selecting from $\frac{n(n-1)}{2}$ edges. Its

distribution is such that probability of a node of degree k is $P(k) \approx e^{-(k)} \frac{(k)^k}{k!}$, following Bernoulli Degree – with k as average connectivity of G. For small values of P probabilities, G has many isolated cliques; But with the value of $P > \frac{Log\ V}{V}$, almost all nodes are connected (Pavlopoulos et al, 2011).

2. **Watts-Strogatz (Small World)** model deviates from the concept of random graph in that each node is added to G sequentially and linked to an existing one based on a chosen probability proportional to its current degree in a hierarchical model. It assumes G grows via power-law degree distribution with the probability of n of degree k which is proportional to $P(k) = k^{-\gamma}$, where $\gamma = 3$ (Watts and Strogatz, 1998; Pavlopoulos et al, 2011). The graph is characterized or either influenced by: (a) *small* path length α that defines its average shortest path between pair of nodes and determines probability of nodes being connected given a number of common neighbors; And, so controls the extent to which G has small or densely connected components in that as α nears infinity, it yields a random graph, or (b) its *clustering* coefficient q is the average pair of neighbors of a node connected to another, which determines the probability of an edge being rewired to connect to a node in G. Small values of q implies high clustering coefficient cum large average path length so that G become a random graphs; Else, as q tends to 0.01 (small values of q), it becomes a small-world graph (Schnettler, 2009 and Ojugo et al, 2014).

3. **Scale-Free Graphs** describes Barabasi/Albert model whose graph reveals data about its dynamics from an evolutionary point. It deviates from a random network based on two feats namely: *growth* and *preferential attachment*. It views a graph as an evolving entity that models the dynamics of its growth. Given a positive integer m in an initial network G_o, G evolves based on these rules in discrete time-process as thus:

   i. **Growth** – At each time j, a new node of degree m is added to the network.
   ii. **Preferential Attachment** – For each node x in G, the probability a new node connects to it is proportional to the degree of x. We express G_j for the network at time j and $P(x,y)$ for the probability that the new node added at time k is linked to x in G_{j-1} as in Eq. 9:

$$P(x,y) = \frac{\deg(x)}{\sum_{v \in N(G_{j-1})} \deg(x)} \quad (9)$$

Simply put, each node inserted is sequentially linked to an existing one based on a chosen probability that is proportional to the existing node's current degree, in a hierarchical fashion. Thus, model generates G whose degree of distribution asymptotically approaches power law so that node x of degree k is proportional to $P(k) = k^{-\gamma}$ with range $2 < \gamma < +\infty$ (Albert and Barabasi, 2002; Pastor-Satorras and Vespignani, 2002; Pavlopoulos et al, 2011). Variants of this model all follow a power law degree of distribution with γ as above. If $\gamma \leq 3$, it yields a small world G; whereas if $\gamma \geq 3$, it yields a scale-free graphs. An evolving graph follows power law degree distribution with $2 < \gamma < +\infty$ (Barabasi and Albert, 1999; Dorogovtsev, Mendes and Samukhin, 2000).

3. DIFFUSION MODELS IN RELATION TO GRAPHS

Diffusion is the process or art of moving a product, idea or an innovation from a region of high concentration to that of a lower region/sphere. To this effect, every node on a G can may exist in any of these states: (a) *susceptible*: if node is exposed to the innovation and will likely adopt or is infected by the innovation, (b) *infected/adopted*: if node has adopted the innovation after being exposed and can pass it to others too, and (c) *removed/reject*: if a node had been exposed to the innovation and rejects it (in the case of a virus, the node has recovered, immunized or virus dies). Thus, we have two modes: susceptible-infect-susceptible (SIS) and susceptible-infect-remove (SIR). In SIS, a node may adopt innovation after exposure; But, is later cured but not immunized. Thus, such a node can be infected again. Conversely in the SIR mode, the node switches between being susceptible – to become immunized of the innovation, in which case it can never adopt it again. Thus, node is permanently immunized and can no longer participate in propagation.

3.1 Independent Cascade Model

It is a discrete-time model in which at $t = 0$, an adversary inserts d copies of innovation to some nodes on G. If node x is infected the first time at t, it has a single chance to infect any neighbor y not currently infected. Probability that x will succeed with y is P_{xy}. If x succeeds, y is infected at $t+1$; Else, x tries again (even in the future, if y gets infected by another neighbour). This process continues and stops after n-steps if no more nodes, are unexposed. It needs a node to be infected exactly once following Kempe et al (2003). Graph of size M, has M_d subset of nodes and d copies of virus placed on the network. With diffusion complete, $S(M_d, G)$ is expected number of infected nodes. Expectation exceeds all random choices made by propagation model. Eq. 10 is maximum expected number of infected nodes and maximum exceeds all possible initial virus placements.

$$S_d(G) = \max_{M_d} S(M_j, G) \quad (10)$$

The subset $A_d = arg \max_{M_d} S(M_d, G)$ corresponds to choices made by an inventor to spread the innovations. $S_d(G)$ is the epidemic spread in G and a similar definition of epidemic spread of randomize adversary as in Eq. 11 in which case, it defines the expected epidemic spread where the expectation takes over all possible positions of the d innovations placed on G and given by: $S'_d(G) = E_{M_d}[S(M_d, G)]$ (11)

3.2 Dynamic Propagation Model

In SIS, viruses are seen as dynamic birth-death process that evolves overtime. It continues to either propagate or eventually die. An infected node x spreads virus to node y in time t with infection rate of $\frac{\beta}{\delta}$ and probability β. At same time, an infected node may recover with probability δ. With adjacency matrix T, $\lambda_1(T)$ is largest eigen-value of T. The state $\frac{\beta}{\delta} < \frac{1}{\lambda_1(T)}$ is true as epidemic threshold and sufficient for quick recovery (Ganesh et al, 2005; Wang et al, 2003).

4. CONCLUSION / RECOMMENDATION

Models are predictive, educational tools to aid experts and researchers compile existing knowledge about a task. They serve as vehicle to communicate hypotheses, a means to investigate parameters crucial in estimation as well as help us

better understand a problem domain. Simple models may not provide enough new data, whereas complex models may not be understood. Model implementation as an intellectual tool, requires less accurate numeric agreement in predicted versus observed values as it displays feat of interest with its probabilities. But, rather requires feedback mechanism as more important. Only models that are understandable and easily manageable are fully explored. Thus, modelers must balance complexity and simplicity, which is crucial to studying the relevant processes of how a model works.

Prediction depends largely on predictive value accuracy. But, ABM is more concerned with theoretical model design and explanation of such prediction rather than the prediction itself. For highly abstract experiments that explore plausible mechanisms to observe underlying patterns, ABMs do not necessarily aim at accurate display of a particular empirical application. Instead, enriches our insight of the fundamental processes in a variety of tasks. Making a model realistic, inevitably add complexity can undermines its benefit as tool for theoretical research, if we cannot figure how the model yields a given result. Researchers are skeptic about validity of simulated results if a model is for theoretical exploration rather than empirical prediction (Axelord, 1997).

Global feats are sui generis, but also emerge from bottom up via local interactions – so that path-dependent, emergent feats, self-organize process (like informal control) are not mistaken for globally-coordinated institutions. ABM studies processes with no global-coordination (which, established) – forces a top-down approach. It is dedicated to how simple, predictable local interactions generate familiar but highly intricate, enigmatic global patterns such as data diffusion, coordination of conventions, norm emergence and collective action via participation. These local emergent feats appear unexpectedly to either transform or disappear in a society such as revolutions, market crashes, fads and feeding frenzies. ABMs are theoretical bridge to leverage between individualism and non-reductionist in which global patterns aggregates of interest more than individual feats; And at same time, is an emergent pattern not understood without the bottom-up dynamical model (Macy and Willer, 2002).

5. REFERENCES

[1] Axelrod R., (1997). The Complexity of Cooperation. Princeton, NJ: Princeton Univ. Press [textbooks first]

[2] David, L., (2011). Social network analysis: theory and application, http://train.ed.psu.edu/wfed-543/SocNet_theoryApp.pdf, last retrieved December 3, 2014 [Web accessed books].

[3] Kaufman S., (1996). At Home in the Universe: The Search for the Laws of Self-Organization and Complexity. UK: Oxford Univ. Press.

[4] Macy M and Willer, J., (2002). From factor to actors: computational sociology and agent based model, Annual Review Sociology, 28, p143–166, doi: 10.1146/annurev.soc.28.110601.141117.

[5] Ojugo, A.A., E. Ben-Iwhiwhu, O.D. Kekeje., M.O. Yerokun and I.J. Iyawah., (2014). Malware propagation on time varying networks: comparative study of machine learning frameworks, International Journal of Modern Education and Computer Science, 8: 25-33, doi: 10.5815/ijmecs.2014.08.04.

[6] Ojugo, A.A and Yoro, R.E., (2015). Thesis writing and research, Unpublished Technical report, 2(1): 46.

[7] Ojugo, A.A., F.O. Aghware., R.E. Yoro., M.O. Yerokun., A.O. Eboka., C.N. Anujeonye and F.N. Efozia., (2015a). Predicting behavioral evolution and diffusion innovation on a graph-based model, Advances in Networks, 3(2): 8-21. doi: 10.11648/j.net.20150302.11

[8] A.A. Ojugo., F.O. Aghware., R.E. Yoro., M.O. Yerokun., A.O. Eboka., C.N. Anujeonye and F.N. Efozia., (2015b). Evolutionary Model for Virus Propagation on Networks, Automation, Control and Intelligent, Systems, 3(4): 56-62. doi: 10.11648/j.acis.20150304.12

[9] Simon H. 1998. The Sciences of the Artificial. Cambridge, MA: MIT Press