Investigating Ecological Networks based on Structural Identity

Paras Vasant Mistry
210877245
Athen Ma
Artificial Intelligence

***Abstract* - Networks of trophic links (food webs) are used to describe and understand mechanistic routes for the translocation of energy (biomass) between species. However, due to the challenges of collecting data on a large number of species, food web methodologies have only been used to study a small part of ecosystems. In this research, we show that field sample data may be used to create believable and testable food webs using machine learning of food webs using a logic-based technique.**

Keywords— food webs, species traits

I. INTRODUCTION (*HEADING 1*)

Networks of trophic connections (food webs), which are the main channels for the transfer of energy between species, are crucial for describing the structure and dynamics of ecosystems and may result in broad hypotheses about how ecosystems react to environmental change. Since establishing trophic interactions between the many hundreds of species in an ecosystem is resource-demanding and necessitates significant investment in field observation and laboratory investigation, food webs have only been used to describe and detail a small number of ecosystems. Due to the lack of essential background information about the network, such as whether any two species are even likely to interact, it is frequently impossible to increase the efficiency of searches for trophic linkages by filtering out unlikely interactions. To convert from the ecological "language" of sample data to the network language of linkages within a trophic network, could take a lot of study and interpretation.

A method known as "graph embedding" is used to convert nodes, edges, and their attributes into vector space (a lower dimension) while maintaining the most amount of the graph's information and structure. Graphs can differ in scale, specificity, and subject, which makes them challenging. For data exploration and high-dimensional data visualisation, we are utilizing the t-SNE (Distributed Stochastic Neighbor Embedding) methodology, which is an unsupervised, non-linear method.

Studying the different aspects and research papers of social analysis and learning about the deep network concept made me interested to carry out my work in this area to grasp more in-depth knowledge and idea about how the whole network is built and executed using different algorithms and predictions are made.

II. RELATED WORK

A. Literature Review

Ecological networks are a very vast and in-depth topic to discuss. It's the interaction between species in natural ecosystems. Food-web theory shows that the highest predator-prey body mass ratios found in natural food webs may be especially important because they create weak interactions with slow dynamics. Identifying these vital interactions in real communities typically requires the arduous identification of interactions in complex food webs as said by authors Ulrich Brose and Alison C. Iles (2019). David A. Bohan along with Geoffrey Caron-Lormier (2011)" after depth research stated that networks of trophic links (food webs) are used to describe and understand mechanistic routes for translocation of energy (biomass) between species. However, a relatively low proportion of ecosystems have been studied using food web approaches due to difficulties in making observations on large numbers of species. Some analysis of the food web suggests that the disagreements are based on the selective use of relatively few food webs, as well as analytical decisions that obscure important variability in the data as mentioned by Jennifer A. Dunne, Richard J. William and Neo D. Martinez in their research (2002). A large-scale assessment of agro-ecosystem responses has also been performed by Athen Ma, Xueke Lu, Clare Gray, Alan Raybould and Alireza Tamaddoni-Nezhad (2018) by analysing a case study of 502 replicated food webs, from fields on the farm scale evaluations (FSE) of GMHT crops. Leonardo F.R. Ribeiro, Pedro H.P. Savarese, and Daniel R. Figueiredo (2017) [online] stated that to measure node similarity at different scales and construct a multilayer graph to encode structural similarities and generate a structural context for nodes, Structure identity is been used with representational learning techniques. *Struc2vec* is a framework for learning latent representations for the structural identity of nodes. It uses a hierarchy to measure node similarity at different scales and constructs a multilayer graph to encode structural similarities and generate a structural context for nodes.

B. Background Research

Department of Computer Science (2017). Representation Learning on Graphs: Methods and Applications [online] stated that Machine learning on graphs is an important and ubiquitous task with applications ranging from drug design to friendship recommendation in social networks. The primary challenge in this domain is finding a way to represent, or encode, graph structure so that it can be easily exploited by

machine learning models. Here they have provided a conceptual review of key advancements in this area of representation learning on graphs, including matrix factorization-based methods, random-walks-based algorithms, and graph convolutional networks. They developed a unified framework to describe these recent approaches and highlight a number of important applications and directions for future work. Ross M. et al (2012) stated that reconciling biodiversity and ecosystem function in a single conceptual framework is best achieved through the application of a food-web approach. Food webs are maps of the trophic interactions between species, usually simplified into networks of species and the energy links between them. These networks have a suite of attributes which can be calculated to describe food web structure. food webs provide a natural framework for understanding species' ecological roles and the mechanisms through which biodiversity influences ecosystem function. Ahmed Hassan et al (2021) stated that to track covid-19 disease, Machine learning (ML) can be deployed effectively. ML techniques have been anticipated in areas that need to identify dangerous negative factors and define their priorities. Four standard models anticipate COVID-19 prediction, which are Neural Network (NN), Support Vector Machines (SVM), Bayesian Network (BN) and Polynomial Regression (PR). Five measures parameters were used to evaluate the performance of each model, namely root mean squared error (RMSE), mean squared error (MAE), mean absolute error (MSE), Explained Variance score and r2 score (R^2). The results showed NN outperformed the other models, while in the available dataset the SVM performs poorly in all the predictions.

III. METHODOLOGY

We have used a global dataset of traits and food web architecture by Ulrich Brose (2018). We are building a network of traits consisting of two properties "con.taxonomy" and "res.taxonomy". Machine learning algorithm t-SNE will be applied to the network graph embeddings which we achieved from the Word2Vec model by using the deep walk algorithm.

Dataset: The dataset consists of various environmental factors such as body mass, species traits along with their ecosystem and taxonomical resolution of the consumer species which we are going to work on in our project further. The dataset consists of more than 2 lacs of rows with respect to 46 columns that describe every aspect of the traits. Column 1 & 2 contains the highest available taxonomic resolution description of the consumer species and taxonomic resolution description of resource species. The selection of taxonomy resolution of consumer and resource species to achieve research objectives in the field of machine learning and scientific papers.

con.taxonomy	res.taxonomy
Vulpes vulpes	Tipulidae
Emberiza schoeniclus	Coleoptera
Emberiza schoeniclus	Araneidae
Vulpes vulpes	Tettigoniidae
Emberiza schoeniclus	Chironomidae

Table 1: Sample data of taxonomic resolution of consumer & resource species

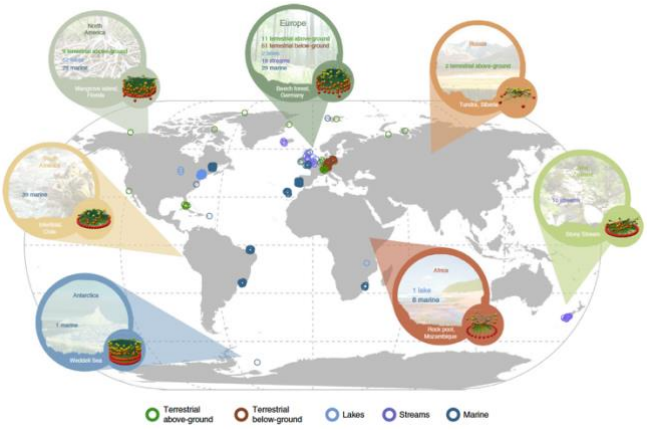

Fig: Global distribution of food webs. The global distribution of food webs in the GATEWAy database

A. Requirements

The provided data must be analyzed using word embedding algorithms like word2vec. To get the most accurate word embedding results that may be used to correctly depict the ecological network, the model will run constantly on the Python platform. The word embeddings using machine learning (ML) in Python language is a method of taking text input and transforming it into real-valued vectors that encodes the meaning of the word with the assistance of the processing tool that follows:

Jupyter Notebook: It is the most recent interactive web-based development environment for data, code, and notebooks. Users may create and arrange workflows in data science, scientific computing, computational journalism, and machine learning using the interface's flexibility.

Numpy: It is a library for the Python programming language that supports big, multi-dimensional arrays and matrices as well as a sizable number of high-level mathematical operations on these arrays.

Pandas: Pandas is a software library for manipulating and analyzing data that was created for the Python programming language. It contains data structures and procedures specifically for working with time series and numerical tables.

Matplotlib: Python's Matplotlib toolkit is a complete tool for building static, animated, and interactive visualizations. Matplotlib makes difficult things possible and simple things simple.

Sklearn: For machine learning in Python, Scikit-learn (Sklearn) is the most reliable and practical library. It offers a number of effective techniques for statistical modelling and machine learning, including dimensionality reduction, clustering, and regression, through a Python consistency interface.

Gensim: It is an open-source framework for current statistical machine learning that performs natural language processing tasks such as retrieval by similarity, document indexing, unsupervised topic modelling, and other similarity-based functions.

NetworkX: It is a Python library for building, modifying, and researching the composition, dynamics, and purposes of complicated networks.

B. Implementation

We are going to use *networkx* to plot the graphical representation of the data and will use the *word2vec* model along with the *deep walk* algorithm to get the embeddings of the graph. NetworkX provides different types of graphs to build the graphical representation of the data such as the following:

Networkx Class	Type	Self-loops allowed	Parallel edges allowed
Graph	undirected	Yes	No
DiGraph	directed	Yes	No
MultiGraph	undirected	Yes	Yes
MultiDiGraph	directed	Yes	Yes

Fig 1: Different Graphs of NetworkX

DeepWalk models a series of brief random walks to learn embeddings (social representations) of a graph's vertices. Social representations are latent characteristics of the vertices that indicate community and local affinities. These latent representations store social relationships as continuous vectors in a limited number of dimensions. It expands neural language models to process a particular language made up of a collection of walks that were created randomly.

Algorithm 1 DeepWalk(G, w, d, γ, t)

Input: graph $G(V, E)$
window size w
embedding size d
walks per vertex γ
walk length t
Output: matrix of vertex representations $\Phi \in \mathbb{R}^{|V| \times d}$
Initialization: Sample Φ from $\mathcal{U}^{|V| \times d}$
Build a binary Tree T from V
for $i = 0$ to γ **do**
  $\mathcal{O}$ = Shuffle(V)
  for each $v_i \in \mathcal{O}$ **do**
    $\mathcal{W}_{v_i} = RandomWalk(G, v_i, t)$
    SkipGram(Φ, $\mathcal{W}_{v_i}$, w)
  end for
end for

Fig 2: *DeepWalk* algorithm

DeepWalk has a pretty straightforward concept. It uses the skip-gram approach to extract the vector representation of the node by using the sentence-length sequence produced by the random walk as input. There are the following steps performed for a deep walk such as:

Step 1 – Generating random walks

A random walk of a predetermined size is first conducted by DeepWalk starting from each node. A basic graph traversal technique called random walk begins at a particular node and then randomly walks on to the next node that is neighbouring, repeating this process forever. However, in the case of DeepWalk, the maximum walk distance is an adjustable hyperparameter. Better results may be obtained with more random walks per node, but the computing workload would increase.

Step 2 – Apply *Word2vec*

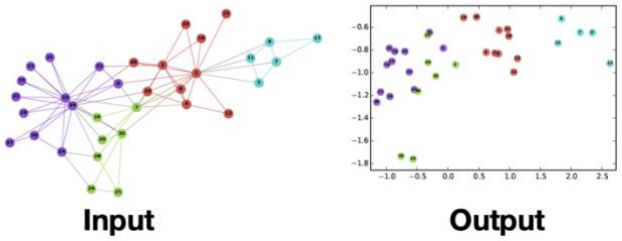

Fig 3: *Word2vec* to learn node embeddings on graph

(Mikolov, 2013) stated that, the Word2Vec technique produces effective word embeddings for each word in a corpus of text. Word2Vec's basic premise is that words with similar meanings frequently appear in comparable settings. Let's take an example of a simple text corpus like this:

1. I like to eat hamburgers.
2. Hamburgers are delicious.
3. I like eating fries.
4. Fries and hamburgers are tasty.
5. Fries are delicious.

Fig 4: Simple text corpus example

In the above example, you can see that "Fries" and "Hamburgers" has similar context. The similarity of the context will be handled by *Word2vec* and it would generate the vector embedding for the text "Fries" and "Hamburgers". Now the words will be arranged according to their similarity between them. This differs based on the parameter "window_size" of word2vec. If the window size is small then the text may consider only nearby neighbours.

In the case of DeepWalk, we do not have a text corpus but regardless it gets the data from random walks. Word2vec performs well even with random walks and helps to generate vector embeddings which help to put adjacent nodes together.

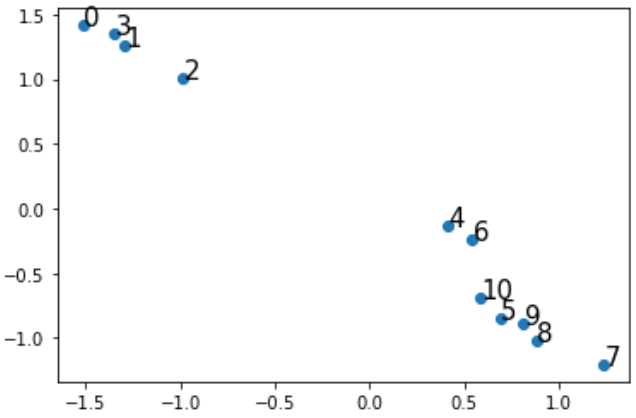

Fig 5: Vector space with 2 dimensions for visualization

Once we get the embeddings our next step is to build the t-SNE (Distributed Stochastic Neighbor Embedding) model. It is an algorithm for studying high-dimensional data that uses non-linear dimensionality reduction. It converts multidimensional data into two or more dimensions that are easy for people to see. You might need to plot fewer exploratory data analysis plots the next time you deal with high-dimensional data with the aid of the t-SNE algorithms.

Because the cost function for t-SNE is not convex, we can achieve different outcomes with various initializations.

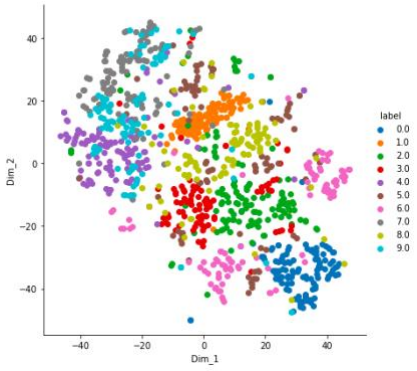

Fig 6: t-SNE model

Once the t-SNE model is plotted, the next step is to get the pairwise cosine similarity. For this, we have used *seaborn* which is a python data visualization package that uses *matplotlib* as its foundation. For creating eye-catching and educational statistics visuals, it offers a high-level interface.

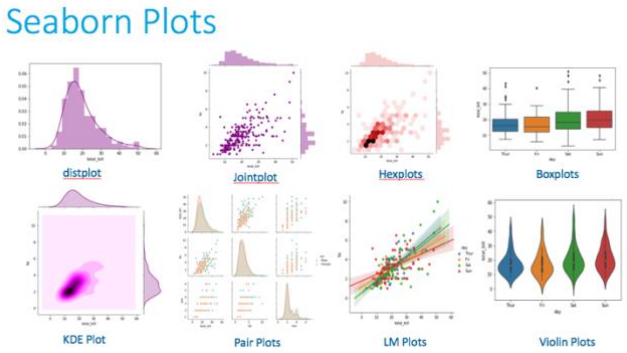

Fig 7: Different plots for seaborn

In our dataset, we have a plot “distplot” for getting the cosine similarity.

Network graph properties

NetworkX is also used to retrieve different network properties of the graph such as density, average clustering, and transitivity.

The density of the undirected graph is given by:

$$d = \frac{2m}{n(n-1)},$$

Fig 8: Density formula for undirected graph

The clustering coefficient for the graph is the average:

$$C = \frac{1}{n}\sum_{v \in G} c_v,$$

Fig 9: Average clustering formula for graph

The transitivity for the graph is given by:

$$T = 3\frac{\#triangles}{\#triads}$$

Fig 10: Transitivity of the graph

Centrality Measures

We'll illustrate several centrality metrics. We define the most typical as follows:

1. A node's ***degree centrality*** in a network is determined by the number of connections (vertices) that intersect it.
2. By calculating the total of the shortest routes (geodesic distances) between a node and every other node in the network, closeness centrality calculates how "near" a node is to other nodes in a network.
3. The amount of traffic passing through a node to reach other nodes in the network is measured using betweenness centrality to estimate the relative significance of a node. This is accomplished by counting the proportion of pathways that contain the node of interest and linking all node pairs. Group Measured by betweenness centrality, the volume of traffic passing through a collection of nodes.

Cliques are a subset of undirected graph vertices where each pair of different vertices in the clique is connected to an adjacent vertex. In other words, a fully induced subgraph of graph G is a clique of that graph.

We have used the attribute “find_cliques” which will return all maximal cliques in an undirected graph.

Recommendations

Although there are several algorithms, one of them is based on the "Open Triangles," a social network theory idea. The feature of three nodes A, B, and C known as triadic closure states that if a strong tie exists between nodes A-B and A-C, then a weak or strong tie also exists between nodes B-C. However, it is a valuable simplification of reality that may be used to comprehend and anticipate networks. This trait is too severe to remain true over very vast, complicated networks.

We have made the top ten suggestions based on the “Open Triangles” algorithm

C. Testing

As compared to the research done on this particular dataset. I think that the *word2vec* algorithm has done its job of creating the embeddings and DeepWalk has given better outcomes for the graph provided.

D. Results

The ecological networks are very large in size. I have tested the dataset based on the taxonomic resolution for consumer and resource species. We tried using the *word2vec* model along with the *DeepWalk* algorithm to visualize the data and receive graph embeddings.

Machine learning methods like t-SNE have been applied to the embeddings provided by the graph to match the similarity between the data and arrange it in the better format possible. We can observe the graph below which states the progress of our project.

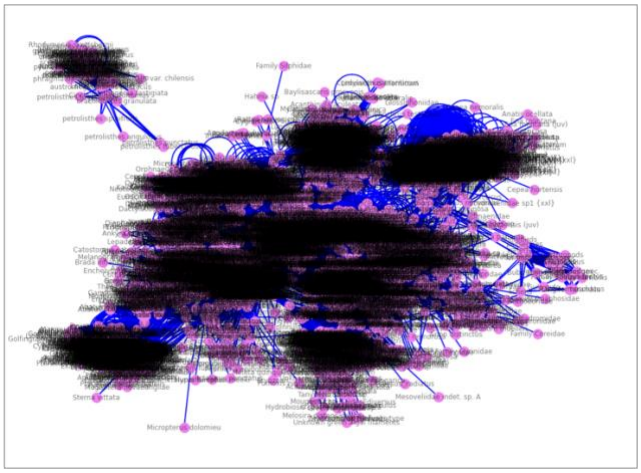

Fig 11: Graph of the ecological network

	con.taxonomy	res.taxonomy
0	5216	4.000000
1	3777	0.853190
2	1318	-0.386378
3	1918	0.126870
4	5065	-0.790162
5	2667	-0.329990
6	4242	0.758275
7	2612	-0.402961
8	3885	-0.695983
9	1365	-0.140308

Fig 12: Graph embeddings

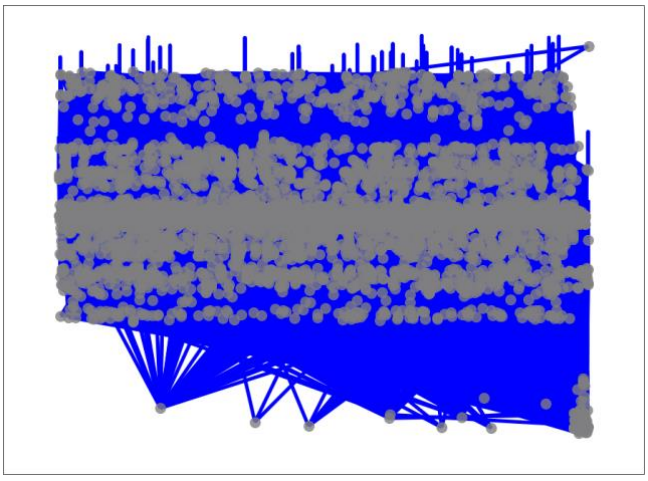

Fig 13: Graph after applying the DeepWalk algorithm

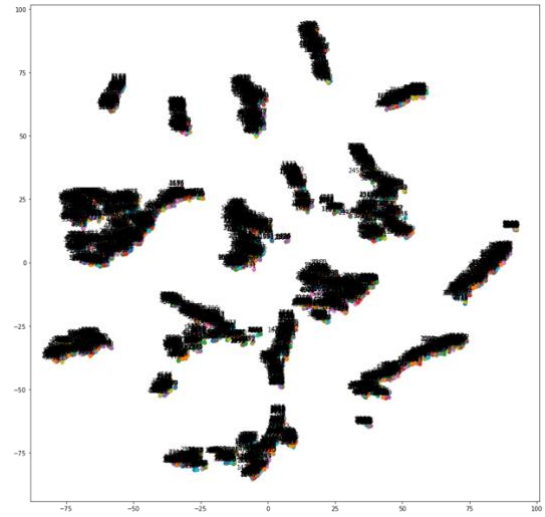

Fig 14: t-SNE model using word2vec model

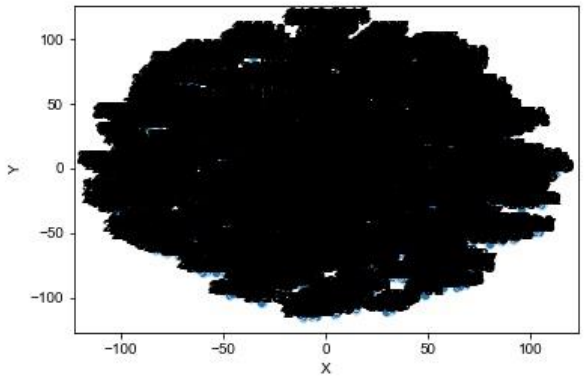

Fig 15: t-SNE model using seaborn

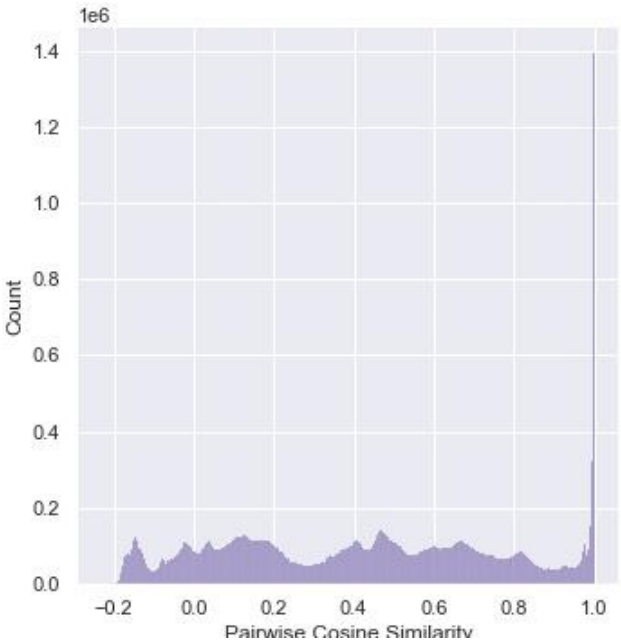

Fig 16: Pairwise Cosine Similarity

IV. DISCUSSIONS

The ecological food web is very large, so applying machine learning algorithms to each structural entity is impossible which can lead to some incomplete data about the species. In this paper, we are trying to implement a graph structure for the given dataset of traits and food web architecture. We have used two properties of the species that is taxonomical resolution description for both consumer and resource species. The graph represents the connectivity between the consumer and resource species with respect to taxonomical resolution. By using the *word2vec* model along with the *DeepWalk* algorithm we get a clear representation of the data in the form of graph embeddings which indicates the connection of nodes and edges.

The larger cannot be handled by the jupyter notebook environment which may lead to a restart of the kernel. Got to learn different techniques to represent data in the graphical format along with machine learning algorithms to visualize the data to get in-depth knowledge about the data on which the task was performed. The research can be used for an honest appraisal and can be utilized to modify the techniques used for the implementation of the task.

V. FUTURE WORK

We can use more advanced machine learning algorithms on the dataset to get more in-depth knowledge about the ecosystem and use advanced visualizing techniques which can provide high-resolution output for better understanding. Preprocess the data in a better way and extract information and use it to plot the data in a more detailed manner.

REFERENCES

Ulrich Brose, Phillippe Archambault, Alison C. Lles, Andrew D. Barnes – Predator traits determine food-web architecture across ecosystem. https://doi.org/10.1038/s41559-019-0899-x

Bohan DA, Caron-Lormier G, Muggleton S, Raybould A, Tamaddoni-Nezhad A (2011) Automated Discovery of Food Webs from Ecological Data Using Logic-Based Machine Learning. https://doi.org/10.1371/journal.pone.0029028

Jennifer A. Dunne, Richard J. Williams and Neo D. Martinez - Food-web structure and network theory: The role of connectance and size https://www.pnas.org/doi/epdf/10.1073/pnas.192407699

Athen Ma, Xueke Lu, Clare Gray, Alan Raybould, David A. Bohan and Alireza Tamaddoni-Nezhad, - Ecological networks reveal the resilience of agroecosystems to changes in farming management https://doi.org/10.1038/s41559-018-0757-2

Department of Computer Science (2017). Representation Learning on Graphs: Methods and Applications [online] https://www-cs.stanford.edu/people/jure/pubs/graphrepresentation-ieee17.pdf

Ross M. Thompson et al (2012). Food webs: Reconciling the structure and function of biodiversity. Available from: https://doi.org/10.1016/j.tree.2012.08.005

Leonardo F.R. Ribeiro, Pedro H.P. Saverese, Daniel R. Figueiredo – Struc2vec: Learning Node Representations from Structural Identity https://dl.acm.org/doi/10.1145/3097983.3098061

Ahmed Hassan Mohammed Hassan et al (2021). Visualization & Prediction of COVID-19 Future Outbreak by Using Machine Learning. https://www.mecs-press.org/ijitcs/ijitcs-v13-n3/IJITCS-V13-N3-2.pdf

Tomas Mikolov, Kai Chan et al (2013) Efficient Estimation of Word Representations in Vector Space. https://arxiv.org/pdf/1301.3781.pdf

Brose, U. et al. (2018) GlobAL daTabasE of traits and food Web Architecture (GATEWAy) v.1.0. (iDiv Data Repository, accessed 17 April 2019); https://doi.org/10.25829/iDiv.283-3-756