International Journal of Information Management 56 (2021) 102012

Contents lists available at ScienceDirect

International Journal of Information Management

journal homepage: www.elsevier.com/locate/ijinfomgt

# A novel architecture to identify locations for Real Estate Investment

Sandeep Kumar E[a,*], Viswanath Talasila[a], Ramkrishna Pasumarthy[b]

[a] Department of Telecommunication Engg., M.S Ramaiah Institute of Technology, Bengaluru 560054, India
[b] Department of Electrical Engg., Indian Institute of Technology Madras, Chennai 600036, India

ARTICLE INFO

*Keywords:*
Time complexity
Complex network science
Machine learning
Real estate investment
Data privacy
Blockchains

ABSTRACT

The identification of a favorable location for investment is a key aspect influencing the real estate market of a smart city. The number of factors that influence the identification easily runs into a few hundreds (including floor space area, crime in the locality and so on). Existing literature predominantly focuses on the analysis of price trends in a given location. This paper aims to develop a set of tools to compute an optimal location for investment, a problem which has received little attention in the literature (analysis of house price trends has received more attention). In previous work the authors proposed a machine learning approach for computing optimal locations. There are two main issues with the previous work. All real estate factors were assumed to be independent and identically distributed random variables. To address this, in the current paper we propose a network structure to derive the relational inferences between the factors. However, solving the location identification problem using only a network incurs computational burden. Hence, the machine learning layers from the previous work is combined with a network layer for computing an optimal location with proven lower computational cost. A second issue is that the computations are performed on an online database which has inherent privacy risks. The online data, user information and the algorithms can be tampered through privacy breaches. We present a privacy preservation technique to protect the algorithms, and use blockchains to secure the identity of the user. This paper presents solutions to two interesting problems in the analysis of real estate networks: a) to design tools that can identify an optimal location for investment and b) to preserve the privacy of the entire process using privacy preserving techniques and block chains.

## 1. Introduction

There are two (related) problems that are of crucial interest in real estate investment: identifying an optimal location for investment and the ability to do so in a secure manner. Software tools now enable users to compare across real estate investment opportunities. These tools are now online, thereby posing a privacy risk for the user leading to security threats. These two problems are now explained in detail.

Firstly, location in a city plays a crucial role in real estate investment. People consider many criteria when they decide to invest in a location. These criteria range from public transportation service, availability of schools and restaurants, parks, water availability, temperature and rainfall in that location, safety and crime rate, and so on. This leads to a large set of attributes that increases the complexity in decision making for determining an optimal location due to choice overload (Reutskaja et al., 2018). There is a need for intelligent algorithms and methods to assist a user with best locations for investment considering all the attributes in which a user is interested. In existing literature data science is used for providing solutions for the problems associated with real estate investment (Byeonghwa park & Jae, 2015; Liu, Mavrin, Niu, & Kong, 2016; Wei, Guang-ji, & Hong-rui, 2010; Xue, 2015; Liu et al., 2016). A majority of the literature focus on the hedonic modeling of house price and predictions, price forecasting (Zhang et al., 2009)by using machine learning and st

atistical modeling. A key assumption here is that investors are aware of the location where they want to invest. However, this assumption is not true in general, and there are many factors which make it very difficult for a user to know a good investment location. Locations in large cities can easily compromise thousands of dwellings and commercial properties; with no clear homogeneous characteristics (such as in demography, schools, religious and other establishments, facilities, etc.). Focusing only on price trends, and assuming an investor knows the investment location precisely, are strong assumptions which are do not hold in general. Hence there is a need to develop tools and techniques which allow an investor to identify an *optimal* location for investment given a wide range of real estate factors.

Secondly, there is a growing trend of online software equipped with powerful algorithms (Reply, 2019; Ubitquity, 2019) to help users

[*] Corresponding author.
*E-mail addresses:* sandeepe31@gmail.com (S.K. E), viswanath.talasila@msrit.edu (V. Talasila), ramkrishna@ee.iitm.ac.in (R. Pasumarthy).

https://doi.org/10.1016/j.ijinfomgt.2019.09.008
Received 18 February 2019; Received in revised form 6 September 2019; Accepted 24 September 2019
Available online 31 October 2019

identify good investment options. These algorithms run on the back of extensive data sets that comprise detailed information of many neighborhoods' in a city. It is important that such online software, and their supporting databases, be protected from cybercrime that attack data privacy. Few such attacks include similarity and difference attacks, joint attacks (Soontornphand & Natwichai, 2016), neighborhood attacks (Zhou & Pei, 2008), and so on.

In the previous work (Sandeep Kumar, Viswanath Talasila, Naphtali Rishe, Suresh Kumar, & Iyengar, 2019) the authors extensively use concepts derived from data science for finding locations for real estate investment. However, in Sandeep Kumar et al. (2019), it is not straightforward to derive the inter-attribute dynamics since the attributes are treated independent and identically distributed.

In this paper, we solve the same location identification problem using network science [1] approach. To the best of author's knowledge, network modeling has not been adopted for identifying locations for real estate investment. We first use a complete bipartite network that identifies landmarks and the then use a different bipartite network structure to identify locations (condominiums) in a landmark. The selection of best condominium happens using eigen and alpha centrality network measures. Network science provides a useful method to compute and visualize the dynamic interactions between various network entities. It also provides information about the topological nature of the dynamically changing interconnections. But, as the network becomes complex (w.r.t network size), the time complexity in the location identification algorithm convergence increases. Our comparative study in this paper, demonstrates that the data science approach is computationally superior than the network science approach, in providing optimal locations for real estate investment. However, network modeling provides a better inference on the dynamic interactions and topology of the real estate investment network.

A novel approach is proposed in this paper (to address the first problem, of location identification), where the data science techniques are combined with network science. We present a three-layer hierarchical solution for the proposed problem. In layer 1, we retain the statistical modeling for attribute selection from our previous work (Sandeep Kumar et al., 2019) along with decision trees for landmark identification. In layer-2, PCA and k-means clustering are used for identifying locations in that landmark. The third layer is derived from network science. The selected condominiums obtained from the layer-2 is passed onto layer-3 where a complex network architecture of the condominiums is obtained. The centrality measures on this network infer the best condominium for investment. The purpose of adding a network model is to select the best condominium considering their mutual influences with respect to real estate attributes. The proposed method in this paper, outputs a set of condominiums that matches the user's preferences and a network that is constructed out of these condominiums that rank the nodes (condominiums and attributes) according to their centrality scores. A user interested in the best condominium selects the condominium with the highest score.

Concerning the second problem (of privacy), an adversary who is interested in the algorithmic details may pass multiple queries and can get the underlying working notion of the algorithm that can lead to privacy breach leading to security threats and vulnerabilities. Hence, it is necessary to protect the privacy of the algorithm using data privacy techniques. Specifically, the paper uses differential privacy as a privacy preservation method for the complex network architecture of the condominiums, which is the output of our real estate location identification algorithm. However, the addition of privacy preservation methods comes with a trade-off of accuracy due to the induced noise in the process. In this paper, differential privacy retains the top condominiums and attributes and shuffles/changes the centrality of the rest. The method will be discussed in detail in the later sections of this paper. This differential privacy technique is an add-on on the location identification algorithm. Suppose, a user is a realtor and uses this software as a 'decision tool' to his/her clients, then the algorithm proprietor (realtor) identity is preserved using block chains with specially developed smart contracts using Ethereum chain for the purpose of location identification. The results of the algorithm (list of best locations and the real estate network) is transferred to the requester anonymously. The anonymity provided by block chain plays a key role in safeguarding the identity of these kinds of users. This is also an add-on on the location identification algorithm. In addition, usage of block chains for real estate investment opens up a new path for the developers to incorporate block chains in the real estate investment for purchasing, selling, transferring property deeds, and so on. Hence, the proposed location identification architecture in the paper consists of a combined data and network science approach to identify the locations embedded with differential privacy techniques and blockchains.

The definition of the smart city by European Commission was quoted in Ismagiloiva, Hughes, Rana, and Dwivedi (2019); Israilidis, Odusanya, and Mazhar (2019) as "smart city is a place where the traditional networks and services are made more efficient through the use of digital and telecommunication technologies, for the benefit of its inhabitants and businesses". In addition authors state there is a need for a unified and integrated strategic thinking and planning for future developments in smart cities. In Wu and Chen (2019), it was highlighted that the government and private bodies working on smart cities should give more importance to the citizen centric solutions thereby promoting citizen satisfaction. Further it was stated that use of innovative technical solutions to the smart city problems will create economic competitiveness among other cities which attracts more stake holders such as migrants, tourists and investors. The work discussed in this paper satisfies all these requirements from a perspective of a smart city, where a combination of approaches from various fields like finance, data science, network science, data privacy and blockchains is used for providing an efficient technological architecture for location identification in real estate for a smart investment in a city, considering an investor's requirement.

The rest of the paper is as follows: Section 2 discusses the related works, our previous work and state of art work comparison, Section 3 deals with complex network visualization of the real estate investment scenario and its time complexity calculations, Section 4 discusses the data and network science combined approach for location identification, Section 5 discusses the network dynamics, Section 6 deals with the framework to extend the proposed method for multiple factors, Section 7 discusses the differential privacy, Section 8 discusses the application of block chains for real estate investment and finally Section 9 discusses the implications of this work from a smart city viewpoint, and Section 10 deals with the conclusions of the paper.

## 2. Related works

In this section, a comprehensive study on the existing literature is presented. In Liu, Yang, and Liu (2007), authors discuss a complex network view of residential real estate markets. and construct a network having entities like banks, house producers and so on, and the links representing the cash flow between various entities. Firm size and individual wealth were characterized by power law and Zipf's law distributions. The goal of the work was to study the dynamics of the various entities in a residential real estate network. The simulation was carried out on the Shanghai residential market during 2001-2005. In Guo and Xue (2009), a network is constructed comprising of accounts opened in a bank as nodes and the links being the capital transferred among accounts.The work focuses on the clustering analysis on the network and using shortest path algorithm to find the correlation ship

[1] Network science provides a graphical representation of interactions between various random, semi-random or deterministic in nature. This graphical structure/network view of a given scenario outputs various inferences on a given problem.

among two nodes in a network. In DArcangelis and Rotundo (2016), a sample of equity mutual funds investing in European stocks are studied based on the complex network approach applied to stock holdings. The results shows that there is a community formation having stocks that are connected through the mutual fund owners they have in common though they are geographically dispersed in different European countries. The discussed works prove that the application of complex networks are often seen in finance and economics, nevertheless, a strategic way of constructing a network by identifying the right entities and linking parameters are always a challenge.

In Liu, Wang, and Wei (2017), the data of 2751 stock companies of 2012 and the data of 2578 stock companies of 2013 were collected from Chinese stock market and were converted into a complex network using visibility graph method. For these complex networks, degree distribution and clustering coefficient were considered as the research parameters. The result shows that the complex network has a power law distribution and a small world characteristics. In Battiston, Glattfelder, Garlaschelli, Lillo, and Caldarelli (2010), authors discuss the use of networks in finance and economics by using similarity based networks, hierarchical networks, control networks and transaction networks. In Jiajia Ren (2019), authors propose a complex network view of the stock market and obtain various inferences before the stock market crash.The goal of this work is to find the reasons for the crash by detecting the changes in the dynamics of the network before and after the crash. While we can see numerous applications of network modeling in finance and economics, surprisingly, inferring on the relational status between various entities for location identification in real estate networks have not been dealt so far.

In Timothy and Sharma (2016b), the authors design a hedonic regression model that investigates the spatial dependency of various attributes on the real estate price using mutual information and variance of the inflation factor. They used linear regression and regression trees for this purpose.

In Timothy and Sharma (2016a), authors discuss hedonic modeling using machine learning techniques like Support Vector Machines (SVM), Principal Component Regression (PCR) and k-means nearest neighbor techniques and proved that PCR is better than other techniques that was inferred using spearman's correlation coefficient. After a careful literature survey, it was observed that data science and network science are applied widely in finance and economics, although independently. Further, their applications on location identification are not available in the existing literature.

The web applications like magicbricks (2019) and 99acres (2019) are query based technologies that ask the user an exact location and shortlist the houses and condominiums based on the user attribute preferences. These are query based applications without machine learning (data science concepts) or network science methods as compared to our proposed work. Data science helps in selecting the best attributes, identifying a street and set of locations for investment, where as network science focuses on providing an abstract view of the relation between various entities of real estate investment for identifying best location, which are novel aspects of our work and not available in the web applications like magicbricks (2019); 99acres (2019).

In Goldberg (2019), Nguyen, Imine, and Rusinowitch (2016), authors use differential privacy for network release. However, the techniques are in general, and not specific to real estate investment. Compared to the techniques mentioned in Goldberg (2019), Nguyen et al. (2016)Goldberg, 2019Goldberg (2019), Nguyen et al. (2016), in this paper, we present a novel differential privacy technique to preserve the edge privacy of a network called *camouflage differential privacy* together with naive differential privacy techniques, specifically targeted towards real estate investment network. In Li, Yang, Sun, and Zhang (2017) authors propose a technique called MB-CI (Merging Barrels and Consistency Inference) that provides edge privacy for a graph using the edge histogram of a graph. However, it is not specific to the real estate network and in addition, this work is different than our proposed method.

The companies like Reply and Ubitquity (Reply, 2019; Ubitquity, 2019) provide a platform for the exchange of land records over blockchains. Authors in ZHANG, Yinghui, Ximeng, and ZHENG (2018) propose an architecture that provides secure and fair payment of outsourcing services without relying on the third parties. They claim that the architecture achieves a robust fairness and is resilient for eavesdropping and malleability attacks. In Karamitsos, Papadaki, and Barghuthi (2018), authors propose a concept of using Ethereum smart contracts for real estate investment. The smart contract is between a landlord/real estate owner and tenants. The purpose of the contract is to make sure that the rental agreement is signed, the rental amount is paid on time, and the termination of the contract is executed correctly. In Krupa and Akhil (2019), authors propose an architecture to use blockchains in real estate investment where they claim that various paper works involved in real estate can be digitized using a blockchain like improved property search though blockchain enabled multiple listing service, property visit and inspection, negotiation of terms, values and signing of the letter of intent, lease due diligence by using smart identities, automated agreement, payments, and cash flow using smart contracts, execution of sale and real-time data analysis. However, there are no existing works that use blockchains for transferring locations or network specific to real estate investment.

To summarize, researchers have used regression to analyze the dependence of attributes like the number of beds, bathrooms, and so on, on the real estate price and using this dependence they were able to predict the future price. However, the attributes were considered independently (i.i.d). In reality, these attributes are not independent and are always influenced by their fellow attributes. This dependence is not considered in any hedonic and regression related works. On the other side, in network science part, there are no specific works specific to location identification in real estate investment. There are related literature in finance and economics, where the relationship between attributes is analyzed. However, the derived relationships are not used for classification and location identification. But, for applications like real estate investment, classification, identification, together with the study of influences and relationships among the attributes is highly essential. In this direction, the work proposed in this paper is a novel attempt. This is the first work that uses network modeling location identification in real estate investment. In addition, the application of differential privacy and blockchains for location identification in real estate investment is not available in the current literature and it is also a novel aspect of our work. The architecture (block diagram) of the proposed method is shown in Fig. 1.

### 2.1. Data description and our previous work on location identification

#### 2.1.1. Data set

For the proposed work, real estate Multiple Listing Service (MLS) data from*Terra Fly* (database maintained by Florida International University) database (Terrafly, 2019) of nine landmarks (we call streets, roads, boulevards, etc. as landmarks in our previous work and the work in the current paper) was considered. This MLS data is the condominium data available in those landmarks. The data is available as a downloadable file and has nearly 350 columns describing the attributes of the condominium and 200 rows being the condominium IDs. We have approximately 7000 condominiums in Alton Rd, 7000 in Bay Rd, 9000 in Collins Ave,1500 in Dade Blvd,2000 in Lincoln Rd, 2000 in Lincoln CT, 4000 in Washington Ave, 2000 in West Ave and 2000 in James Ave, which belongs to Miami Beach City of Florida.

#### 2.1.2. Previous work

Miami-Beach city is considered as the case study. The roads, streets, boulevards and so on (which are called landmarks) are clustered into groups. Nearest landmark is given preference for grouping. Initially, nine landmarks are considered. These landmarks has numerous condominiums (we call them as locations)and every condominium has

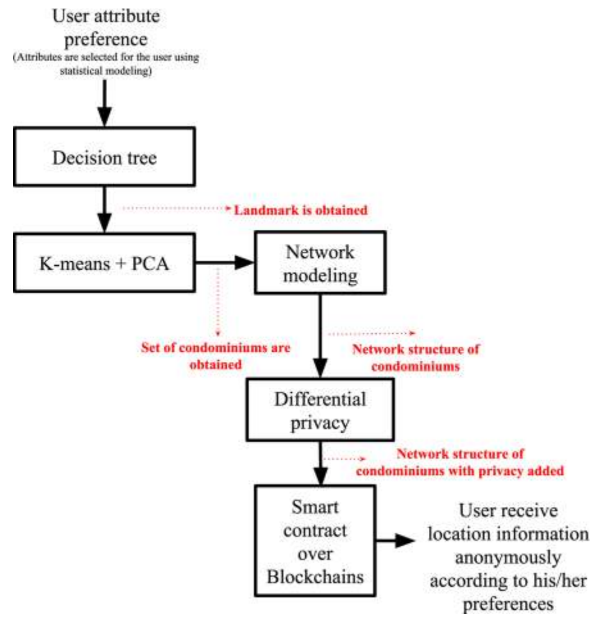
User attribute preference
(Attributes are selected for the user using statistical modeling)
Decision tree
Landmark is obtained
K-means + PCA
Network modeling
Set of condominiums are obtained
Network structure of condominiums
Differential privacy
Network structure of condominiums with privacy added
Smart contract over Blockchains
User receive location information anonymously according to his/her preferences

**Fig. 1.** General system block diagram.

condominium-units. These units are associated with 350 attributes (no. of car parks, no. of garage spaces, and so on).

These nine landmarks are analyzed as a single cluster, and this cluster has a set of top attributes based on the $\chi$ value [2] For the nine landmarks the top attributes obtained were:

- Number of Beds: Number of Bedrooms available in a unit of the condominium building.
- Number of Full baths: Number of full bathrooms (tub,shower,sink and toilet)available in the unit.
- Living area in sq.ft: The space of the property where people are living.
- Number of garage spaces: Number of spaces in the property available for parking vehicles.
- List Price:Selling price of the property to the public
- Application fee:Fee paid for owner's association.
- Year Built:Year in which the condominium/apartment complex is built.
- Family Limited Property Total Value 1: The property value accounted for taxation after all exemptions. This is for the district that does not contain schools and other facilities.
- Tax amount:The amount paid as tax for the property every year.

Suppose, *Number of Beds* attribute is considered, this particular attribute has different values of $\chi$ in every landmark, same for the other attributes as well. Based on the values that an attribute picks, a set of top attributes are selected for further processing. These attributes are given a choice for a user, based on the users choice an interest vector (which is a binary string) is created and passed onto a decision tree, whose leaf nodes are landmarks. The tree selects the best landmark based on *highest magnitude win approach*. In addition, the magnitude entered by a user (say, if he is interested in the number of bedrooms then he will enter its count) is passed onto the second layer. The principal component analysis (Kong et al., 2017) is used to calculate the principal component score for every condominium in a landmark and the PC scores are clustered using k-means clustering. One cluster is selected which is the closest match to the user's preferences using euclidean distance metric between the k-means centroid and the PC score generated from the user's choice magnitude. The selected cluster has set of condominiums. Through out this paper condominiums and locations are used interchangeably, they both mean the same in the context of the work in this paper. The method is shown in Fig. 2. For the work discussed in this paper, same data set is used with Miami Beach city as a case study.

### 2.2. Block diagram and contributions of our current work

Among a large set of attributes, few are shortlisted based on statistical modeling (Sandeep Kumar et al., 2019), and the attribute preference of the user is passed onto the stacks of learning algorithms to obtain the set of locations. The obtained set of locations with their attributes form a complete bipartite network with attributes as one partition and the condominiums as the other; in the network modeling block. This network is privacy preserved by edge differential privacy and the obtained privacy preserved network along with the list of locations (condominiums) is transacted over the Ethereum blockchain using interplanetary file system (IPFS) (Narayan Prusty, 2017) for Ethereum chains.

In this section, we will highlight the specific contributions of our proposed work.

- We identify locations [3] for real estate investment based on user preferences.
- We compare the time complexity for two approaches: data science approach and the network science approach for location identification for real estate investment and infer on the superiority of the technique in terms of time complexity.
- We construct a novel technique that combines both data science and network science for location identification
- We conduct dynamic perturbations in the network weights and check the impact of this random variations of link weight on the network centrality measures.
- The obtained network is checked for privacy preservation of the edges in the real estate complex network using existing differential privacy techniques and a novel method called "*camouflage differential privacy*" is proposed for privacy preservation.
- As a case, the transaction of the resultant network over the blockchain using a dedicated Ethereum smart contract using IPFS is discussed.

## 3. Complex network visualization

In this section, we solve the location identification problem dealt in Sandeep Kumar et al. (2019) using complex network science. The algorithm used to find the top attributes in our previous work is retained. Every top attribute has an associated $\chi$ value computed over an entire landmark, which is used as a link weight. In our previous work, the stack of machine learning algorithms was used to identify the set of condominiums (locations). In the approach discussed in this section, the machine learning based layers of our previous work (refer to Section 2.1.2), are replaced by networks/graphs. The decision tree in the layer-1 is replaced by a complete bipartite graph [4], where one partition are the attributes and the other being landmarks. When a user enters his choice of attributes, only the links of those attributes are retained and rest are nullified. By application of eigenvector centrality on this

[2] $\chi$ value is the correlation representative of the attribute with the real estate price in a landmark. An attribute is selected as the top attribute based on the $\chi$ value it gains competing with all the other attributes in the given cluster landmarks.

[3] In this paper, the term locations and the condominiums mean the same and are used interchangeably. The proposed method in this paper will identify the set of condominiums that match to the user's requirements.

[4] Two party graph hence called *bipartite*. The links always flow from vertices of partition-1 to vertices of partition-2, and there are no intra-partition links.

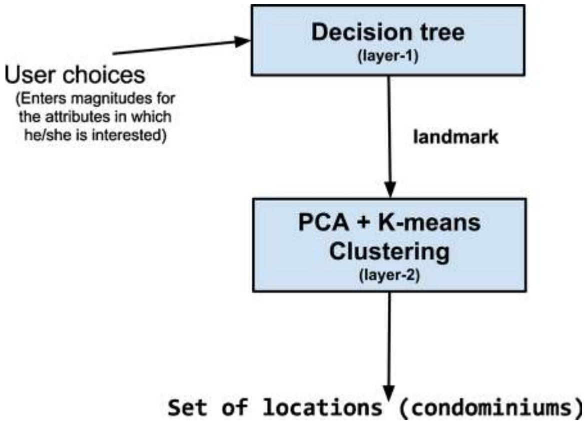
Decision tree
(layer-1)
User choices
(Enters magnitudes for the attributes in which he/she is interested)
landmark
PCA + K-means Clustering
(layer-2)
Set of locations (condominiums)

**Fig. 2.** Data science based approach for identification of location.

weighted bipartite network, the central landmark is selected. Once the landmark is identified, we proceed towards the identification of best condominiums for investment. For layer-1 only user's choice was considered i.e, only interests of a user. However, for layer-2 the magnitudes of the user entered attributes are considered. The user entered magnitude is normalized by the attribute maximum value of that landmark. These normalized values are injected as exogenous factors into the layer-2 which is again a bipartite graph of attributes and condominiums. The links between the various entities are the $\chi$ values fetched by an attribute within that landmark. Application of alpha centrality[5] Lerman, Lain, Ghosh, Kang, and Kumaraguru (2013) to the layer-2 will select the best (central) condominium.

In addition, other condominiums can also be ranked according to the centrality values, and alpha centrality replaces PCA and k-means clustering of layer-2 in Sandeep Kumar et al. (2019). Hence, this is a two-layer architecture where in layer-1, Eigen centrality is used and in layer-2 Alpha centrality, and the entire architecture is solely constructed using network analytics. The current architecture is shown in Fig. 3. In Fig. 2 there are two layers derived from machine learning techniques, of which in layer-1 decision trees for landmark identification and in layer-2 PCA with k-means clustering is used for best condominium selection in a landmark, respectively. In Fig. 3 there are two layers comprised of bipartite networks, where the layer-1 uses Eigen centrality for landmark identification and layer-2 uses alpha centrality for the identification of best condominium in a landmark.

**Definition 3.1.** Real estate complete bipartite network A graph $G(V, E)$ is said to be a complete real estate bipartite graph if and only if there exists a partition $V(G) = c \cup f$ and $c \cap f = \phi$, where $c$ is the set of condominiums or even landmarks, and $f$ is the set of attributes, such that all the edges share a vertex from both set $c$ and $f$ and all possible edges that join vertices from set $c$ to set $f$ are drawn.

### 3.1. *Time complexity calculation–complex networks analogy*

In this section, the time complexity of the network architecture of the real estate investment is calculated. Here time complexity is considered as an indirect metric to measure the complexity that results due to increase in the network size which in turn is dependant on the number of attributes, locations, and so on.

Eigenvector centrality (Anand and Pandia, 2015) computation takes $O(V^3)$ time, where V is the total number of vertices in the graph (Lei Tang, 2019), which is based on the power-iteration algorithm, however in this case the graph is fully connected or with a graph with non-sparse adjacency matrix. Alpha centrality is also eigenvector centrality but with exogenous factors, hence the time complexity is $O(V^3) + O(V)$, where the first term is due to eigenvector calculation and the second term is due to the addition of exogenous factors to every node in a network. However, for a bipartite network with lesser connections than a fully connected network the complexity of eigenvector centrality is less than $O(V^3)$ and similarly alpha centrality is less than $O(V^3) + O(V)$. Let us consider an example graph shown in Fig. 4.

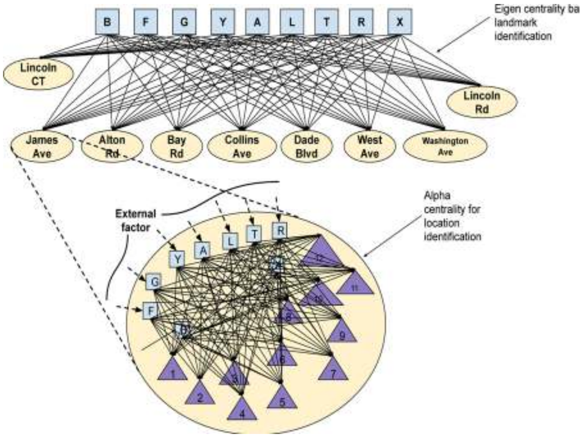

**Fig. 3.** Complex network visualization of real estate scenario where B: Number of Beds, F: Number of full baths, G: Number of garage spaces, Y: Year built, A: Application fee, L: List price, T: FLP Total value, X: Tax amount. Attributes are represented by square shape and condominiums & landmarks by triangle and ellipsoidal shapes respectively.

The adjacency matrix for this graph is:

$$\begin{array}{c c} & \begin{matrix} a & b & 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} a \\ b \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \begin{bmatrix} 0 & 0 & X & X & X & X & X \\ 0 & 0 & X & X & X & X & X \\ X & X & 0 & 0 & 0 & 0 & 0 \\ X & X & 0 & 0 & 0 & 0 & 0 \\ X & X & 0 & 0 & 0 & 0 & 0 \\ X & X & 0 & 0 & 0 & 0 & 0 \\ X & X & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \end{array}$$

where $X \in \mathbb{R}$.

Observe that the matrix is sparse with majority of the entries being 0. For a fully connected graph/non-sparse adjacency matrix the time complexity is calculated as $O(V^2)$. $O(V) = O(V^3)$, where $O(V^2)$ is due to the $V^2$ number of multiplications in the power iteration method and $O(V)$ is the maximum number of steps the algorithm can run before it converges. However for the matrix which is derived from the bipartite network, assuming $V_1$ and $V_2$ as the number of vertices available in each group, exactly $V_1^2 + V_2^2$ number of multiplications are reduced from the total multiplications of power-iteration method. That is because, a computer algorithm will not consider the zero entries for the iterations. Hence, in every step there is a reduction in the complexity with zero entries in the matrix.

Hence the eigenvector calculation in a bipartite network has a complexity of $O(V^2 - V_1^2 - V_2^2)$. $O(V) = O(V^3 - V_1^2 . V - V_2^2 . V)$, where $V = V_1 + V_2$. For the network in the Fig. 4, the total time complexity is $O_{eigen}(.) = 20$. Similarly, for alpha centrality the time complexity is $O(V^3 - V_1^2 . V - V_2^2 . V) + O(V)$ and $O_{alpha}(.) = 27$.

Suppose $V_1$, $V_2$ are the vertices in the first layer and $v_3$, $V_4$ are the vertices in the second layer, and $V = V_1 + V_2$, $W = V_3 + V_4$ then the overall time complexity of both layers is: $O(V^2 - V_1^2 - V_2^2). O(V) + O(W^2 - V_3^2 - V_4^2). O(W) + O(W)$. An example calculation for the nine landmarks is as follows: In the first stage, we have 18 vertices hence complexity is $O(5832 - 1458 - 1458) = O(2916)$. In the second stage (say if Alton Road was selected by first

[5] One of the advantage of alpha centrality is it considers the exogenous factors for finding the central node in a given graph. However, by limiting the value of alpha (called information perturbation control parameter), we can approximate alpha centrality to eigen centrality. Alpha centrality is a modification of the eigen centrality method.

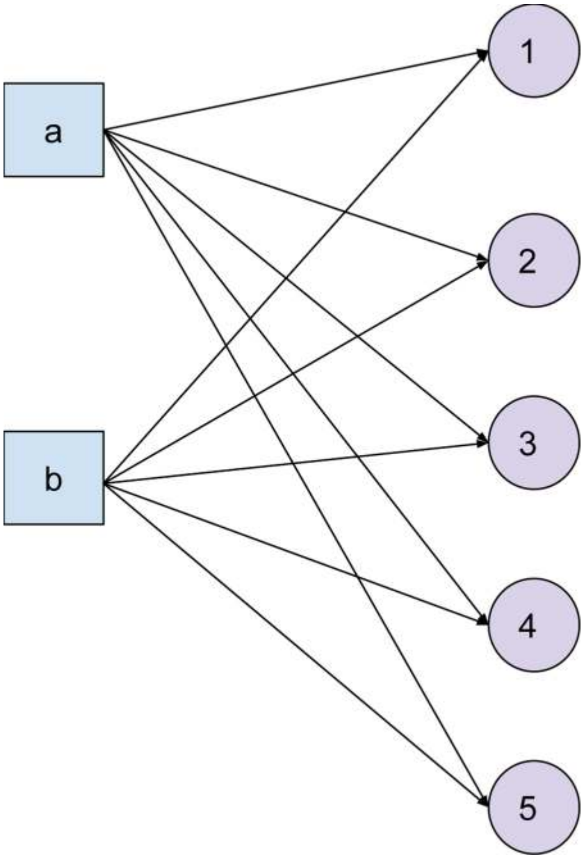

**Fig. 4.** Example network for complexity calculation.

stage), there are 7000 condominiums and 9 attributes hence time complexity is $O(7009^3 - (7000^2 * 7009) - (9^2 * 7009)) + O(7009) = O(883141009)$. Therefore, the overall complexity is O (883141009) units of time.

The time complexity for finding the top attributes for a given cluster of landmarks is neglected. The procedure of finding the top attributes is available in both network science and data science approach and both follows the same method and hence results in same computation time complexity. This will not add any value in terms of comparison of both approaches, hence the time complexity of finding top attributes is neglected.

### 3.2. Time complexity calculation–data science approach

In this section, we calculate the time complexity for the statistical modeling and machine learning approach used in Sandeep Kumar et al. (2019). Here also the time complexity for finding top attributes not considered.

*Part-1:* Decision-Trees (layer-1): This part was used to find the landmark, based on the user's interest vector, the time complexity of decision tree is $O(log_2 n)$, where n is the number of nodes in the tree (Sandeep Kumar et al., 2019).

*Part-2:* PCA and K-means clustering (layer-2): For PCA, time complexity is given by $O(f^2N' + f^3)$, where $f$ is the number of top attributes and $N'$ is the number of units in a condominium. To calculate PCA for $N$ condominiums in a landmark, time complexity is $O(f^2NN' + Nf^3)$.

To calculate the PC score of one unit in a condominium, time complexity is $O(2f)$. For $N'$ number of units in a condominium, we have $O(2N'f)$. For an entire condominium its the average, hence $O(2N'f + N' + 1)$. For $N$ such condominiums we have $O(2N'fN + N'N + N)$.

Principal score is calculated using the first principal component of every condominium and the average principal component of a landmark is given by $O(N+1)$ for $N$ condominiums in a landmark.

For K-means clustering,$O(NKI)$ (Aldrich, 2002) is the complexity, where, N is the number of data points, K is the number of clusters and I is the number of Iterations. In our case, N is the number of condominiums (since it is PC scores of condominiums to be clustered) in a landmark, K is the clusters required and I = 1. Therefore, total time complexity is given by: $O(log_2 n) + O(f^2N'N + Nf^3) + O(2N'fN + N'N + N) + O(N + 1) + O(NK)$.

Let us consider Alton Rd as an example, where $N' = 200$, $N = 7000$, $f = 9$, $K = 20$ clusters, and n = 512, then the overall time complexity of using data science approach is O(300702010) time units.

If we compare the time complexity of calculations from Section 3.1 and Section 3.2, it is clear that data science has lower time computational complexity than the network science approach. Even though we can solve the location identification problem using machine learning, but sole learning techniques do not infer on the dynamics and relations among the attributes. Hence, the two techniques are combined to incorporate the benefits of both approaches.

## 4. Data and network science combined approach

The statistical modeling used to select the top attributes and the machine learning layers to identify locations are retained from our previous work. A network is constructed from the condominiums that are output from the machine learning layers. The architecture is shown in Fig. 5, where there are three layers, the first layer inputs the attribute preferences from a user and selects the best landmark, the second layer selects the set of best condominiums from a landmark and the third layer is a complex network structure of these condominiums that ranks the condominiums using eigen centrality network measure. The architecture in Fig. 5 is called the combined approach in this paper since it is a combination of both data and network science approaches.

### 4.1. Time complexity of the combined approach

In this section, we prove that the time complexity of the combined approach is in between the time complexities of data science and network science approaches. Hence, there are three kinds of time complexities to compare: sole data science approach, sole network science approach and the combined approach. We prove that the combined approach is having complexity in between that of the other two, and hence achieves a trade-off between two methods. The properties of the real estate complex bipartite networks are:

**Property-1.** If $G(V, E)$ be a real estate complete bipartite network

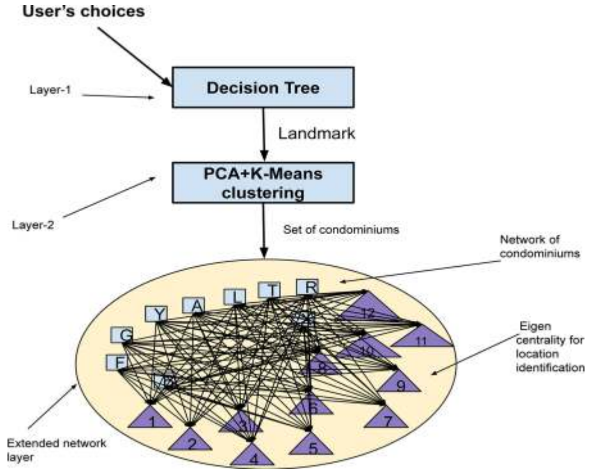

**Fig. 5.** Combination of data science with complex networks,where, B: Number of Beds, F: Number of full baths, G: Number of garage spaces, Y: Year built, A: Application fee, L: List price, T: FLP Total value, X: Tax amount.

with the bi-partitions $c$ and $f$, with $m$ and $n$ vertices respectively, then the total number of edges in $G$ is $mn$.

**Property-2.** If $G$ is complete real estate bipartite network with bi-partitions as $c$ and $f$ with $m$ and $n$ vertices in them respectively, then $\deg(v_i) = n$ and $\deg(v_j) = m \ \forall \ v_i \in c$ and $v_j \in f$.

**Property-3.** If a real estate complete bipartite network $G$ with a positive number of edges exists then $G$ is 2-colorable.

Suppose, $\mathfrak{T}$: time complexity of data analytics approach, t: time complexity of complex networks approach, $T$: time complexity of combined approach then,

**Proposition 1**: $\mathfrak{T} < t < T$.

*Proof.* Since in the *combined approach*, network structure comes as an add-on on the existing method, the time complexity of the latest method, will be greater than the previous method. Hence, $t > \mathfrak{T}$.

Let us consider layer-2 of complex networks approach and the extended network layer of combined approach. In Section 3.1 analysis, it was clear that the alpha centrality considers the entire condominiums of a landmark. Let the number of condominiums be $N$ and the available attributes be $f$. Therefore, total number of vertices in the graph is $V' = N + f$.In the combined approach, the number of condominiums obtained after optimization using machine learning layers is less than before $N$. Let us denote the number of condominiums after optimization as $n'$. The obtained graph from combined approach contains total vertices $V'' = n' + f$. We know that $n' < < N$. As per our simulation studies $n'$ was in the order of 20 and N is 7000 for Alton Rd.

Consider, $n' < < N \Rightarrow V'' - f < < V' - f$

Therefore, $V'' < < V' \Rightarrow O(V'') < < O(V')$

Therefore, t $<$ $T$, hence we can conclude that $\mathfrak{T} < t < T$.

Let us calculate the time complexity of our latest method: $O(log_2 n) + O(f^2NN' + Nf^3) + O(2N'fN + N'N + N) + O(N + 1) + O(NK) + O(V''^2 - n'^2 - f^2)$. $O(V'')$.

A simulation study was conducted by supplying values to the attributes like: Number of garage spaces = 3, application fee = 400, Number of full bathrooms = 3, Number of bedrooms = 2, Built Year = 1986, Taxable Property value = 1942446, living area = 1007 Sq. ft, Tax amount = 8633, List Price = 2000000. It was observed that the landmark selected by layer-1 was James Ave, 401 condominiums were selected by layer-2 in that landmark. We know that for James Ave, $N$ = 2000 and $n' = 401$ and $n' < < N$. Hence, the latest method (combined approach) gives a total time complexity of $O(44145786)$. For James Ave $N$ = 2000, $N' = 200$, $\mathfrak{T} = 41502010$, $T = 72328925$ and t = 44145786. From the obtained complexity values it is clear that $\mathfrak{T} < t < T$.

From property-1: the total number of edges in the network approach of Section 3.1 for James Ave is 9*9 = 81 in layer-1 and 9*2000 = 18000, with total number of edges being 18081. Where as in the combined approach the total number of edges is 9*401 = 3609. Similarly, from property-2, in the case of network approach, in layer-1, the degree of every node in partition-1 is 9 and in partition-2 it is 9. In layer-2 the degree of very node is 2000 in partition-1 and 9 in partition-2. In the combined approach, the degree of every node in partition-1 is 401 and in partition-2 it is 9. It is clear that in the combined approach there is a reduction in the number of edges and degree of nodes that in turn reduces the time complexity in centrality computation. This supports the notion that t $<$ $T$.

Fig. 6 shows the optimal set of condominiums obtained after PCA and K-means clustering in a bipartite network architecture, for the simulation inputs. There are two types of nodes in the network, the attributes are depicted as circles and the condominiums are shown in triangular shapes respectively. The link weights are the $\chi$ values which are the correlation representatives calculated w.r.t every attribute and the real estate price of respective condominiums (Sandeep Kumar et al., 2019). Eigen centrality was applied to find the best condominium. The simulation results showed that condominium-1693 was most central (best). The set of condominiums selected, changes based on the magnitude of attribute entered by a user.

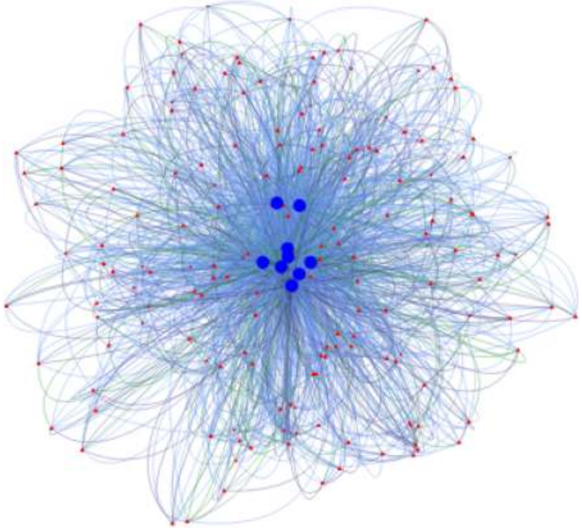

**Fig. 6.** Bipartite network view of condominiums, the blue colored circle are the attributes and the red colored triangles are the condominiums.The size of the nodes vary based on their centrality values. The links are green if their weights are above 1.

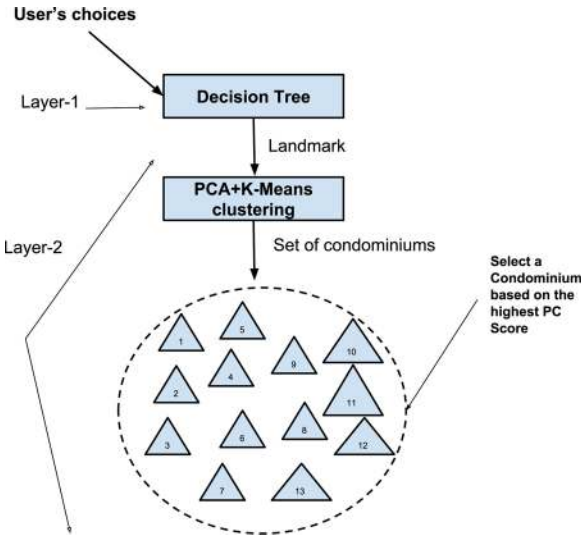

**Fig. 7.** Selection of best condominium based on PC scores (a complete data science approach).

Suppose if the complex networks part was not added and we continue to use the data science concepts to find the best condominium. Then optimal location selection can be carried out based on the PC score that was assigned to the condominium before. Whichever condominium has the highest PC score that would be the best condominium to invest and the scenario is shown in Fig. 7. [6] However, there will not be relational inference but just ranking among condominiums will be obtained.

When the time complexities of methods in Figs. 5 and 7 are compared, method in Fig. 7 is less time complex, since PC score comparison takes very minimal time, $O(n)$, where $n$ is the number of comparisons, than the complex network addition which brings an overhead $O(n^3)$, where $n$ is the number of vertices. However, the combined approach is

[6] There is another way to select the best condominium i.e., by matching the PC score generated by user's attribute entry with the PC scores of the condominiums in a landmark. Whichever is the closest match is the best condominium.

still preferred than the data science approach due to the advantage of the network to infer the relationships among the attributes of the real estate. Using networks as an extended layer is an example demonstration of the usage of network science concepts, however, this layer need not be always as an end layer. Any layer in the architecture can be replaced by graphs whenever necessary.

However, when we applied the method shown in Fig. 6, the obtained result was *condominium-1701*. The result of the method in Fig. 3 is different than the result of the method in Fig. 5 because of different cost functions.

**Proposition-2**: The optimization in the data science approach has a cost function $f_1$ and the combined approach has cost function $f_2$ then: $f_1 \neq f_2$.

*Proof.* Let $N'$ be number of units in a condominium. There are $N$ such condominiums in a landmark. Let there be $f$ number of top attributes for cluster of landmarks. The PC score for a condominium is calculated by,

$$P_q = \frac{1}{N'} \sum_{i=1}^{N'} \mathrm{pcs}_{\mathrm{iq}} \tag{1}$$

$pcs_i$ is calculated using (2),

$$\mathrm{pcs}_i = \sum_{j=1}^{f} \mathrm{pc}_{\mathrm{ji}} {*} T_{\mathrm{ji}} \tag{2}$$

Equation (1) denotes the principal component score of a condominium,where $pcs_{iq}$ is the principal score of $i^{th}$ unit of a $q^{th}$ condominium, and $T_*$ is the magnitude of the attributes of a unit in that condominium.

Substitute (2) in (1) we get,

$$P_q = \frac{1}{N'} \sum_{i=1}^{N'} \sum_{j=1}^{f} \mathrm{pc}_{\mathrm{jiq}} {*} T_{\mathrm{jiq}} \tag{3}$$

For q = 1, we have $P_1 = \frac{1}{N'}[(\mathrm{pc}_{1i1} {*} T_{1i1}) + (\mathrm{pc}_{1i2} {*} T_{1i2}) + (\mathrm{pc}_{1i3} {*} T_{1i3}). ..]$, similarly we have $p_2$, $p_3$ and so on, for all condominiums selected by PCA and K-means clustering.

The *pc* in equation (2) is the first principal component with maximum variance information out of all available components. Hence, we can write (3) as,

$$P_q = \frac{1}{N'} \underset{\mathrm{pc}}{\arg\max} \sum_{i=1}^{N'} \sum_{j=1}^{f} \mathrm{pc}_{\mathrm{qij}} {*} T_{\mathrm{qij}} \tag{4}$$

Therefore, the cost function for the data science part is

$$f_1 = \max(p_1, p_2, p_3. ..) = \max(p^*) \tag{5}$$

For the latest method (combined approach), from the available list of condominiums output by PCA and k-means clustering, we construct a graphical structure. The $\chi$ values are the link weights and Eigen centrality provides the best condominium.

Let G(V,$\chi$) be a graph, such that $V \in L$, $T$, where $L$ are the landmarks and $T$ are the attributes. Recall that $V = n' + f$[7] .

Let us define eigen centrality of a graph $G$ as

$$\mathfrak{C}_i = \sum_{j=1}^{V} A_{\mathrm{ij}} \mathfrak{C}_j \tag{6}$$

Therefore, the cost function is,

$$f_2 = \max(\mathfrak{C}) \tag{7}$$

For j = 1, $\mathfrak{C} = \frac{1}{\alpha_{\max}}[A_{11}\mathfrak{C}_1 + A_{12}\mathfrak{C}_2 + \ldots]$, similarly we can obtain $\mathfrak{C}_3, \mathfrak{C}_4, \mathfrak{C}_5. ..$

Similar to (4) we can write,

$$\mathfrak{C}_i = \frac{1}{\alpha_{\max}} \underset{\mathfrak{C}}{\arg\max} \sum_{j=1}^{V} A_{\mathrm{ij}} \mathfrak{C}_j \tag{8}$$

Because every time we select a condominium based on the centrality value of its adjacent nodes and every node expects the neighboring node centralities to be maximum.

By induction on (3) and (6) it is clear that: $\mathfrak{C}_1 \neq p_1$, $\mathfrak{C}_2 \neq p_2$, $\mathfrak{C}_3 \neq p_3$,...Therefore it is proved that $f_1 \neq f_2$.

The cost function for the combined approach can be derived similarly, since it is the combination of the two techniques. From $Proposition - 2$, it was clear that whether a complete data science approach or a network science approach or the combined method is used, the cost function is different and hence the obtained results will differ depending on the method of choice.

## 5. Analysis of network dynamics of combined approach

The motivation behind using network science is to study the relationship between the condominiums and the attributes and their influences on each other. Similar to Section 4 analysis, let us set the attributes for simulations like the following: number of beds-2, number of garage spaces-2, number of full bathrooms-2, application fee-126. According to our *combined approach*, Alton Rd was the result of the decision tree and 169 number of condominiums were selected by layer-2. According to eigen centrality, *condominium- 6487* was selected as the best condominium. The obtained network is as shown in Fig. 8.

To study the link weight dynamics, the link weights are varied and each time the eigen centrality values are noted. At no variation [8] of links weights, the obtained centrality values are shown in Table 1. This table contains the centrality values of the attributes calculated without perturbations in the link weights.

The weights of the links associated with the second central attribute *F* was increased in steps of 10% on the existing value. The result is shown in Fig. 9. It was observed that as the link weight increases, *F*'s centrality value also increases and at a point of 30% increase, *L* loses its top central position and *F* becomes the more central attribute. Hence, it was concluded that adjusting the weights of the links controls the centrality of the network. This also implies that higher the correlation of an attribute with the real estate price of a landmark, higher that attribute will become central in the network. The same explanation holds true for the case when the link weights are decreased.

In another experiment, the weights associated with all attributes were increased to a maximum of 10% of their values randomly, to check the most stable attribute. This helps us to understand the most stable attribute due to sudden uncertain inflations.

It was observed that the List Price (L) remained more stable as the most central node even after 100 iterations. In addition, when the weights were increased 40% of their link values, *F*, *B*, *T* mixed randomly and the system became inconsistent. However, List price (L) remained stable till 80% perturbations, later its position changed randomly between the top four positions.

This simulation indicates that during sudden changes in the correlation which may be due to natural calamities, inflation, and so on, *List Price* attribute remains invariant by being most influential in the real estate investment. The same analogy can be drawn on condominiums as well and most invariant condominium can be found.

## 6. Extending the framework for multiple factors of real estate investment

As discussed in Section 1, real estate investment is a complex phenomenon that comprises of multiple factors. Like real estate factors,

[7] Our bipartite network has $n'$ number of condominiums and $f$ number of attributes with total vertices being $V$

[8] The links were varied by finding 0%, 10%, 20% and so on, of the existing weight and adding it to the link weights back, such that it is an additive noise

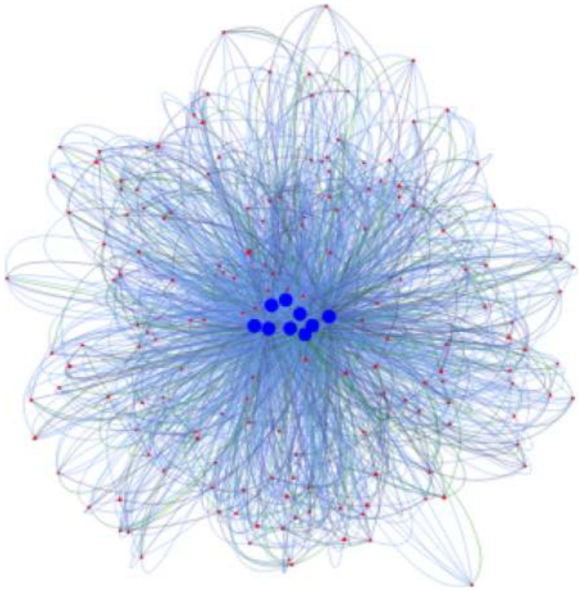

**Fig. 8.** Bipartite network view of Alton Rd condominiums, the blue colored circle are the attributes and the red colored triangles are the condominiums.The size of the nodes vary based on their centrality values. The links are green if their weights are above 1.

**Table 1**
Eigen centrality values without link perturbations.

| Attribute | Eigen centrality value |
|---|---|
| Year Built (Y) | 0.4426 |
| Number of garage spaces (G) | 0.5621 |
| Tax amount(X) | 0.6645 |
| Application fee (A) | 0.6921 |
| Living area (R) | 0.7502 |
| FLP total value 1 (T) | 0.7648 |
| Number of bedrooms (B) | 0.7651 |
| Number of full baths (F) | 0.7787 |
| List price (L) | 1.0000 |

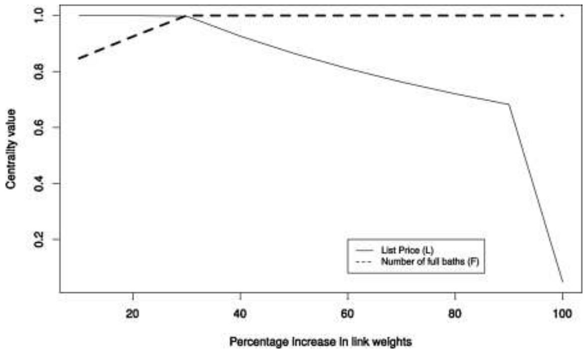

**Fig. 9.** Effect of the link weight change on the centrality values of the top two attributes.

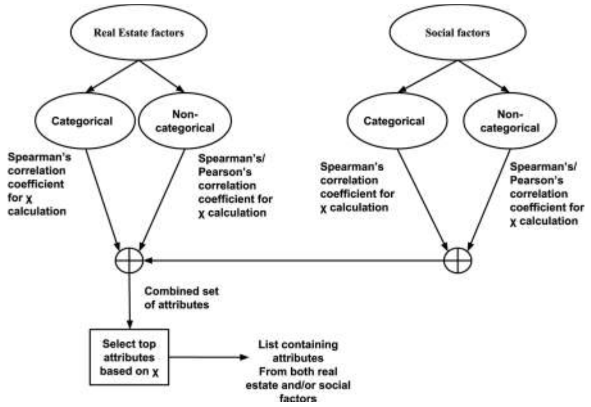
Real Estate factors
Social factors
Categorical
Non-categorical
Spearman's correlation coefficient for χ calculation
Spearman's/ Pearson's correlation coefficient for χ calculation
Categorical
Non-categorical
Spearman's correlation coefficient for χ calculation
Spearman's/ Pearson's correlation coefficient for χ calculation
Combined set of attributes
Select top attributes based on χ
List containing attributes From both real estate and/or social factors

**Fig. 10.** Finding top attributes.

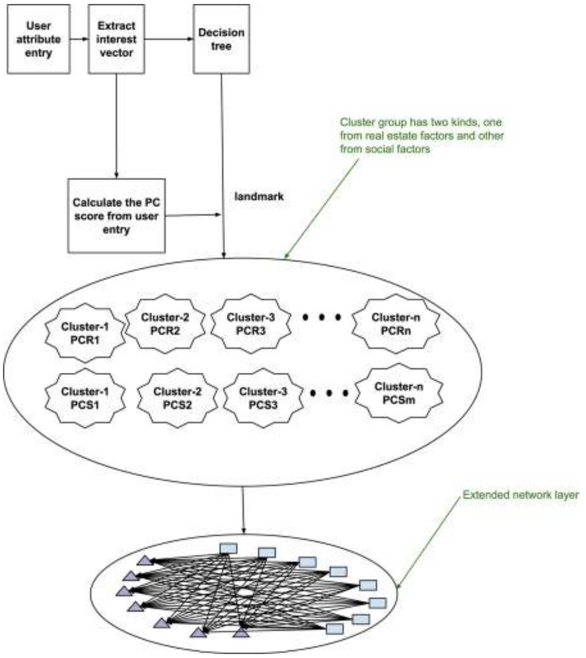

**Fig. 11.** Finding locations and constructing a complex network.

there are social factors (Bhat et al., 2018) that comprises attributes like language, ethnicity, religion, age, and so on, the same explanation is true for environmental and other economic factors. In this section, we address how to extend the architecture discussed in the previous sections to all these factors and set of locations are provided for a user considering the large space of attributes. The architecture is shown in Figs. 10 and 11 . The structure is explained considering two factors, the same could be extended to other kinds of factors.

Fig. 10, explains the steps for determining top attributes where, the real estate factors comprises of both categorical and non-categorical data. In our previous work (Sandeep Kumar et al., 2019) and the current work only numeric attributes are considered, however there are categorical attributes also in real estate factors. Few of them include–cooling and heating description, building construction type, dining hall type, view from condominium, pets permission, parking restrictions, and so on. To find $\chi$ values instead of Pearson's coefficient, Spearman's correlation coefficient (Szmidt & Kacprzyk, 2010) is recommended, since it considers the rank rather than the actual value. For the numeric attributes either Pearson or Spearman's correlation coefficients can be used. Once the $\chi$ values are calculated, the top attributes are selected from the entire pool of attributes. This final set contains attributes from social and/or real estate factors. The appearance of the attribute is entirely based on the $\chi$ value it fetches. The attribute selection algorithm is available in Sandeep Kumar et al. (2019) and Bhat et al. (2018). These obtained attribute set is passed onto the stacks of machine learning layers.

Fig. 11, shows the steps that are required to find the location. The attribute set is given for a user. He/she enters the options with the magnitudes for each, in whichever he/she is interested. First an interest vector is extracted from the user's entry. A decision tree comprising of both real estate and social factors is fed with the user's interest vector. The tree selects one landmark out of a cluster of landmarks. The *highest* *magnitude win approach* discussed in Sandeep Kumar et al. (2019) is used to obtain a landmark that matches to the users interest. The only change to be observed is that, the truth table used for the operation of the decision tree comprises of both real estate and social factors, and their associated landmarks. From, the user's option entry, magnitude principal component score is calculated. There are two sets of clusters of condominiums unlike real estate factors case, where there was only one. This is because of two kinds of factors considered for building the model. The method applied to form these clusters is discussed in Sandeep Kumar et al. (2019). The obtained sets of clusters have a centroid PC score value in each.The user-generated PC score is compared with both sets of clusters of condominiums. One cluster is selected from each set that is, one from real estate factors and one from social factors. A network of condominiums is constructed combining both of these clusters and using the centrality measures the condominiums (locations) are ranked. This architecture is scalable to any number of factors that influence real estate investment. The network layer ranks the condominiums considering all the factors and the relationships among them and selects the most central among all.

Privacy and security are the major concerns of any software solutions. The next two sections in detail are written with respect to these aspects, such that the paper discusses the solution to the location identification problem as a complete package of analytics with privacy and security concerns.

## 7. Differential privacy for Real estate complex network

In this section, privacy preservation scheme for real estate networks is discussed in detail.

Construction of a complex network needs a large amount of data. In this paper, Miami Beach city data is extensively used for this purpose. The network constructed from the approach discussed in the previous sections has condominiums and attributes as the vertices and the edges being the $\chi$ values, that depicts the relation of the condominium with the attributes in terms of its correlation with the real estate price. However, from the privacy viewpoint, the edge-weights are private information of a landmark and these weights encapsulates the entire information of the landmark. A malicious user who wants his condominiums to top the list would simply change the edge weights suitably such that the centrality of the condominiums changes. However, this is possible only if the malicious user taps the relational information between the edges. This information is necessary since eigen centrality is sensitive towards the magnitude of the edge weights and the relationship among the edge weights. The goal is to privacy preserve this relation information and still find the central condominium and attribute (preserve the ranking of top vertices). To begin with, existing differential privacy techniques Cynthia Dwork (2019) are applied on the proposed complex network and then we proceed to the *camouflage differential privacy* technique. Hence, to summarize, due to the privacy preservation, a end user, receives a list of best condominiums and the network that has the best condominium selected using eigen centrality, however the remaining information i.e, apart from the best condominium and attribute, the centrality values of the other condominiums and attributes,relation between the edge weights are noisy due to privacy preservation technique, such that a user cannot infer anything from such noisy information.

### 7.1. Understanding differential privacy

In the digital era of data transmission, more amount of data is shared online. It is important that the data does not get into the wrong hands. This invokes a potential need for preserving privacy of the data. Differential privacy (DP) in Cynthia Dwork (2019) is one such technique that protects user data privacy by adding a negligible amount of noise to the data. DP is based on the idea that the outcome of the statistical analysis is essentially equally likely independent of whether any individual joins or refrains from joining the database. The random noise is added in such a way that the output of the query made by the presence or the absence of a single entity will be covered up.

*Terminology*

- The data that needs to be protected is contained in a set called $D$, in which each element corresponds to information from an individual user.
- Quantity that we would like to compute from a database $D$ is modeled by $q(D)$ for some mapping $q$ (called query) that acts on $D$; the range of q is denoted as $Q$.
- Changes in the database is defined by a symmetric binary relation on $D\boldsymbol{X}D$ called adjacency relation and is denoted by Adj(.,.); two databases $D$ and $D'$ that satisfy $Adj(D, D')$ are called adjacent databases.
- Directly making $q(D)$ available to the public may cause users in the database to lose their privacy. In order to preserve privacy for any given query $q$ one needs to develop a mechanism $M$ that approximates $q$, $range(M) = range(q) = Q$. A mechanism that acts on a database is said to be differentially private if it is able to ensure that two adjacent databases are nearly indistinguishable (in a probabilistic sense) from just looking at the output of the mechanism.

**Definition 7.2.** $\epsilon$-differential privacy Given $\epsilon \geq 0$, a mechanism $M$ preserves an $\epsilon$-differential privacy if for all $R \subseteq range(M)$ and all adjacent databases $D$ and $D'$ in $D$, it holds that:

$P[M(D) \in R] \leq exp(\epsilon)P[M(D') \in R]$

**Definition 7.3.** $(\epsilon, \delta)$- differential privacy

Given $\epsilon, \delta \geq 0$, a mechanism $M$ preserves $(\epsilon, \delta)$ differential privacy if for all $R \subseteq range(M)$ and all adjacent databases $D$ and $D'$ in $D$, it holds that

$P[M(D) \in R] \leq exp(\epsilon)P[M(D') \in R] + \delta$When

$\delta = 0$, definition-7.2 reduces to $\epsilon$-differential privacy. When $\delta > 0$, even if $\epsilon$ is small, it can still happen that $P[M(D) \in R]$ is large compared to $P[M(D') \in R]$ as a result, one can potentially tell whether the input database is $D$ or $D'$ Cynthia Dwork (2019).

*Mechanisms of noise addition*

- Laplace mechanism (Hsu, 2014): Laplace mechanism works by introducing additive noise drawn from the Laplace distribution.For a given query $q$ with $range(q) = R$, let $\delta = \arg\max_{D,D'} |q(D) - q(D')|$ be the sensitivity of $q$. Then the mechanism $M(D) = q(D) + w$ with w $\sim$ Lap$(\frac{\delta}{\epsilon})$ preserves $\epsilon$-differential privacy.
  The Laplace mechanism reveals an intrinsic trade-off between privacy and accuracy of the result. Notice that the mean squared error of the result is given by:
  $$E[M(D) - q(D)]^2 = \text{var}(w) = \frac{2\delta^2}{\epsilon^2}.$$
  As $\epsilon$ becomes small, the result becomes less accurate and hence more privacy is preserved.The Laplacian noise is inversely proportional to the number of users in the database. In other words, with more users in the database, we can introduce less noise in order to achieve the privacy guarantee. Hence, it is easy to preserve individual privacy with more participating users.
- Exponential mechanism Lilla Tthmrsz (2019): This mechanism requires a scoring function $u: QXD \rightarrow R$. The exponential mechanism $M_E(D;u)$ guarantees $\epsilon$-differential privacy by randomly reporting $q$ according to the probability density function.
  $$\frac{\exp(\epsilon u(q, D))/2\delta_u}{\int_{q' \in Q} \exp(\epsilon u(q', D)/2\delta_u)\text{d}q'}$$

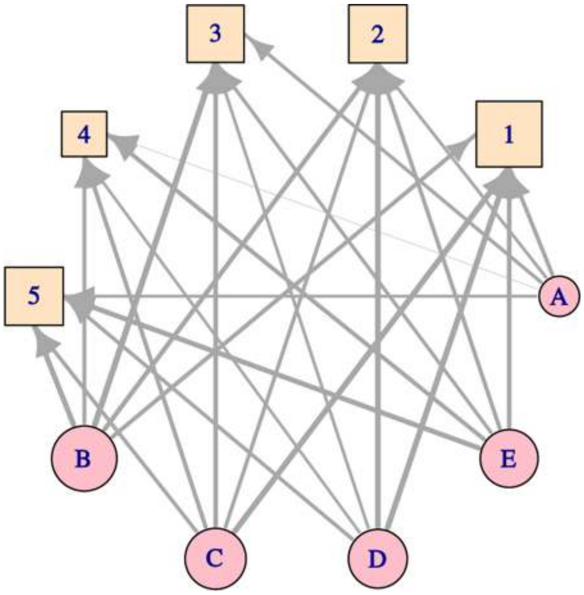

**Fig. 12.** Sample bipartite graph.

where $\delta_u = \underset{D,D'}{\arg\max}\ \underset{\mathrm{Adj}(D,D')}{\arg\max}\ |u(x, D) - u(x, D')|$ is the sensitivity of the scoring function u. Consider the exponential mechanism $M_E(D;u)$ acting on a database $D$ under a scoring function $u$. If $Q$ is finite i.e, $|Q| < \infty$, then $M_E$ satisfies: $P[u_{\mathrm{opt}} - u(M_E(D; u), D] \geq \frac{2\delta_u}{\epsilon}(\log|Q| + t)] \leq \exp(-t)$, where $u_{opt} = max_{q\in Q}\mathrm{u(q,D)}$.The exponential mechanism satisfies $M_E(D:u)$ satisfies: $\mathrm{E}[u_{opt} - u (M_E(D;u), D)] \leq 2\delta_u(1+\log|Q|)/\ \epsilon$.

- Gaussian mechanism (Balle Borja, 2019): For a given query $q$, let $\delta_2 = max_{D,D'||}q(D) - q(D')||_2$ be the $l_2$ sensitivity of q. Then for $\epsilon \in (0, 1)$ and $\delta > 0$, the mechanism $M(D) = q(D) + w$ preserves $(\epsilon, \delta)$-differential privacy when $w$ is a random vector whose entries are zero mean Gaussian with variance $\sigma = \delta_2\sqrt{(}2\log 1.25/\delta)/\varepsilon$

### 7.2. Simulations of existing DP on real estate complex network

Consider an example network that has five condominiums namely 1,2,3,4,5 and five attributes namely A,B,C,D,E. Let this be a complete bipartite graph with the links flowing from attributes to the condominiums. However, the weights are chosen similar to that of the actual network weights ($\chi$ values) and randomly assigned to the links, and the graph is shown in Fig. 12.

Let us consider another graph with one link removed. This graph in shown in Fig. 13, where one link from E to 5 is removed.

Global sensitivity is computed for these two graphs and Laplacian noise is added to the link weights. The noise intensity is defined by $\epsilon$. Let this value be 0.6. Fig. 14 shows a comparison of the link weight probability distribution before and after the addition of the noise. The dotted line shows the noisy link weight distribution (black colored and shifted PDF). The hyperparameters in the setting have to be chosen in such a way that it should be a balance between the actual and noisy PDFs. In the sense that the noisy PDF should not deviate too much from the actual PDF.

The idea of adding this noise was to perturb the weight magnitude with noise but eigen centrality values of the best condominium and attribute should not change. In addition, the relation among the edge weights should change due to the noise addition when noise is added. Consider Table 2, in which the edge weights before and after noise addition is shown. The order of the edge weights before noise addition is given by (in the ascending order), $E_{A4}$,$E_{D4}$,$E_{A5}$,$E_{D3}$,$E_{A2}$,$E_{A3}$, $E_{E3}$,$E_{C2}$,$E_{D5}$,$E_{C5}$,$E_{E2}$,$E_{A1}$,$E_{B4}$,$E_{B1}$,$E_{C4}$,$E_{E4}$,$E_{B2}$,$E_{E1}$,$E_{C3}$,$E_{E5}$,$E_{B5}$,$E_{D2}$,$E_{D1}$,$E_{C1}$,$E_{B3}$. Here $E_{A4} < E_{D4} < .\ .\ . < E_{B3}$. However, even after the noise

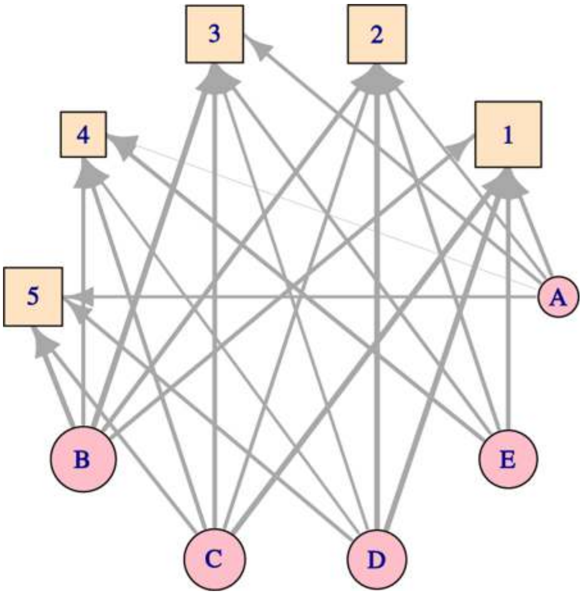

**Fig. 13.** Sample bipartite graph with one link removed.

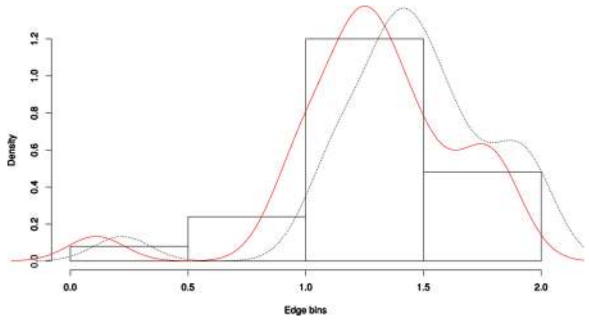

**Fig. 14.** Link weight distribution of the graph before (red colored non-dotted line) and after noise addition (black dotted line).

**Table 2**
Edge weights with and without Laplacian Noise(LN), *epsilon* = 0.6.

| Link | Without LN | With LN |
|---|---|---|
| A → 1,($E_{A1}$) | 1.230 | 1.4176548 |
| A → 2,($E_{A2}$) | 1.010 | 1.1824296 |
| A → 3,($E_{A3}$) | 1.110 | 1.2891907 |
| A → 4,($E_{A4}$) | 0.110 | 0.2319773 |
| A → 5,($E_{A5}$) | 0.990 | 1.1611083 |
| B → 1,($E_{B1}$) | 1.333 | 1.5224213 |
| B → 2,($E_{B2}$) | 1.440 | 1.6217841 |
| B → 3,($E_{B3}$) | 1.860 | 2.0146681 |
| B → 4,($E_{B4}$) | 1.240 | 1.4283780 |
| B → 5,($E_{B5}$) | 1.660 | 1.8270352 |
| C → 1, ($E_{C1}$) | 1.780 | 1.9395010 |
| C → 2, ($E_{C2}$) | 1.220 | 1.4069345 |
| C → 3, ($E_{C3}$) | 1.450 | 1.6310863 |
| C → 4, ($E_{C4}$) | 1.340 | 1.5289120 |
| C → 5, ($E_{C5}$) | 1.220 | 1.4069345 |
| D → 1, ($E_{D1}$) | 1.780 | 1.9395010 |
| D → 2, ($E_{D2}$) | 1.770 | 1.9301157 |
| D → 3, ($E_{D3}$) | 0.990 | 1.1611083 |
| D → 4, ($E_{D4}$) | 0.899 | 1.0642231 |
| D → 5, ($E_{D5}$) | 1.220 | 1.4069345 |
| E → 1, ($E_{E1}$) | 1.450 | 1.6310863 |
| E → 2, ($E_{E2}$) | 1.230 | 1.4176548 |
| E → 3, ($E_{E3}$) | 1.110 | 1.2891907 |
| E → 4, ($E_{E4}$) | 1.340 | 1.5289120 |
| E → 5, ($E_{E5}$) | 1.560 | 1.7335847 |

**Table 3**
Eigen centrality (EC) before and after Laplacian Noise(LN) addition, *epsilon* = 0.7.

| Vertex | EC before LN | EC after LN |
|---|---|---|
| A | 0.6127788 | 0.6468305 |
| B | 0.9924547 | 0.9943658 |
| C | 0.9279013 | 0.9381003 |
| D | 0.8953344 | 0.8792043 |
| E | 0.8801412 | 0.8970817 |
| 1 | 1.0000000 | 1.0000000 |
| 2 | 0.8862076 | 0.8990959 |
| 3 | 0.8691649 | 0.8805701 |
| 4 | 0.6859015 | 0.7150575 |
| 5 | 0.8868285 | 0.8993983 |

**Table 4**
Edge weights with and without Laplacian Noise(LN), *epsilon=0.6*

| Link | Without LN | With LN |
|---|---|---|
| A → 1,($E_{A1}$) | 1.230 | 1.4230 |
| A → 2,($E_{A2}$) | 1.010 | 1.1753 |
| A → 3,($E_{A3}$) | 1.110 | 1.2031 |
| A → 4,($E_{A4}$) | 0.110 | 1.9877 |
| A → 5,($E_{A5}$) | 0.990 | 1.0828 |
| B → 1,($E_{B1}$) | 1.333 | 1.4420 |
| B → 2,($E_{B2}$) | 1.440 | 1.5333 |
| B → 3,($E_{B3}$) | 1.860 | 1.9424 |
| B → 4,($E_{B4}$) | 1.240 | 1.4338 |
| B → 5,($E_{B5}$) | 1.660 | 1.7364 |
| C → 1, ($E_{C1}$) | 1.780 | 1.9379 |
| C → 2, ($E_{C2}$) | 1.220 | 1.3121 |
| C → 3, ($E_{C3}$) | 1.450 | 1.6325 |
| C → 4, ($E_{C4}$) | 1.340 | 1.5344 |
| C → 5, ($E_{C5}$) | 1.220 | 1.3121 |
| D → 1, ($E_{D1}$) | 1.780 | 1.9379 |
| D → 2, ($E_{D2}$) | 1.770 | 1.8286 |
| D → 3, ($E_{D3}$) | 0.990 | 1.0828 |
| D → 4, ($E_{D4}$) | 0.899 | 0.2870 |
| D → 5, ($E_{D5}$) | 1.220 | 1.3121 |
| E → 1, ($E_{E1}$) | 1.450 | 1.6325 |
| E → 2, ($E_{E2}$) | 1.230 | 1.4230 |
| E → 3, ($E_{E3}$) | 1.110 | 1.2031 |
| E → 4, ($E_{E4}$) | 1.340 | 1.5344 |
| E → 5, ($E_{E5}$) | 1.560 | 1.6339 |

addition the relation does not change. The eigen centrality before and after the noise addition is shown in Table 3.

Even though the eigen centrality of the top condominium (node-1) and the top attribute (node-B) do not change, the prime requirement was to preserve the relationship status of the weights of the graph. However, existing privacy preservation methods do not satisfy this requirement. This is the same for the other DP mechanisms like Gaussian and Exponential. The inference drawn out of this simulation reveals that unless the edge weights linked to a vertex shuffle among other weights, the problem persists. Hence, a new technique is developed called *camouflage DP* where an intelligent shuffling of the edges is carried out and following this stage is the naive DP noise addition.

### 7.3. Camouflage DP

The literal meaning of camouflage is the use of any combination of materials, coloration or illumination of concealment, either by making animals or objects hard to see or by disguising them as something else (Cambridge dictionary, 2019). The current technique discussed in this paper derives its name due to a similar analogy used to preserve network edge weights. 15

Let G (V,E) be a graph that has *E* edges and *V* vertices. The edges *E* is passed through camouflage layer.

*Camouflage layer:* Let $E_i$, where $i = 1, 2. .. n$ be the edges of the network. These edges are sorted in such a way that $E_i < E_{i+1}, \forall$ i=1,2,3...n. Let this sorted list be $E'$. Divide this list $E'$ into pairs where each pairs are the non-overlapping neighboring edges, such that $E'_i < E'_{i+1}$ then, the swapping is carried our pair wise i.e., $E'_i = E'_{i+1}$ and$E'_{i+1} = E'_i$, further DP noise is added to this, to obtain $E\tilde{(g)}$.

**Simulations:** Consider the same graph of Fig. 12 as an example,

**Step 1**: Sort the edge weights and group them pairwise with the immediate neighbors that leads to: (0.110 0.899), (0.990 0.990), (1.010 1.110), (1.110 1.220), (1.220 1.220), (1.230 1.230), (1.240 1.333), (1.340 1.340), (1.440 1.450),(1.450 1.560), (1.660 1.770) (1.780 1.780), 1.860

**Step 2:** swap pairwise: (0.889 0.110), (0.990 0.990), (1.110 1.010), (1.220 1.110), (1.220 1.220), (1.230 1.230), (1.333 1.240), (1.340 1.340), (1.450 1.440),(1.560 1.450), (1.770 1.660) (1.780 1.780), 1.860

The edge weight associated to a vertex swaps with its nearest magnitude edge.

**Step 3:** Perform suitable noise addition to all these swapped weights. Note that the vertex associativity of the edge is changed compared to its original associativity due to swap operation in Step-2. Obtained edge weights before and after noise addition is shown in Table 4.

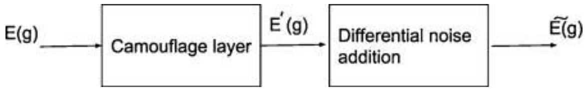

**Fig. 15.** Block diagram of camouflage DP.

Comparing the second and the third column of Table 4, it is clear that the relation between the edges is shuffled resulting in changed associativity with the vertices, and hence the most important criterion of privacy preservation is satisfied. We can see that in the column-2 $E_{A4}$ is the smallest weight whereas in the column-3 $E_{D4}$ is the smallest. In addition, the distribution of the edge weights before and after noise addition as shown in Fig. 16. It can be seen that there is no significant difference between the two PDFs, which satisfies the criterion of D. The eigen centrality values are listed in Table 5.

The inference drawn from Table 5 was after the camouflage DP the eigen centrality of the attributes and the condominiums change, but the most central vertices in the graph remain unaffected before and after noise addition. The trend in the eigen centrality remains unchanged irrespective of the changes in the $\epsilon$ value, this can be seen in Table 6, where eigen centrality of all the vertices after camouflage DP for the variations in $\epsilon$ from 0.1 to 0.5 is available. Observe that the $\epsilon$ values are small. Vertices 1 and *B* are the most central ones among the condominiums and the attributes respectively, and it remains same even though there is a change on $\epsilon$ or in other words if there is a negligible increase in the noise magnitude. This satisfies the second important criterion.

There are two key aspects in camouflage DP– shuffling in the camouflage layer and the noise addition in the DP layer. As stated earlier in this section, the key aspect of privacy protection rests in hiding the

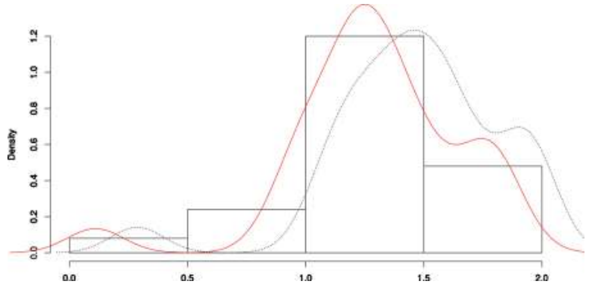

**Fig. 16.** Link weight distribution of the graph before (red colored non-dotted line) and after noise addition (black dotted line).

**Table 5**
Eigen centrality (EC) before and after Laplacian Noise(LN) addition, *epsilon* = 0.6.

| Vertex | EC before LN | EC after LN |
|---|---|---|
| A | 0.6127788 | 0.6468305 |
| B | 0.9924547 | 0.9943658 |
| C | 0.9279013 | 0.9381003 |
| D | 0.8953344 | 0.8792043 |
| E | 0.8801412 | 0.8970817 |
| 1 | 1.0000000 | 1.0000000 |
| 2 | 0.8996368 | 0.8990959 |
| 3 | 0.8810546 | 0.8805701 |
| 4 | 0.7153618 | 0.7150575 |
| 5 | 0.8999143 | 0.8993983 |

**Table 6**
Eigen centrality (EC) values for variations in $\epsilon$.

| Vertex | $\epsilon = 0.1$ | $\epsilon = 0.2$ | $\epsilon = 0.3$ | $\epsilon = 0.4$ | $\epsilon = 0.5$ |
|---|---|---|---|---|---|
| A | 0.8067 | 0.8104 | 0.8135 | 0.8161 | 0.8184 |
| B | 0.9612 | 0.9621 | 0.9630 | 0.9639 | 0.9648 |
| C | 0.9236 | 0.9253 | 0.9270 | 0.9288 | 0.9307 |
| D | 0.7754 | 0.7792 | 0.7826 | 0.7855 | 0.7880 |
| E | 0.8758 | 0.8788 | 0.8819 | 0.8851 | 0.8883 |
| 1 | 1.0000 | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| 2 | 0.8518 | 0.8552 | 0.8585 | 0.8617 | 0.8648 |
| 3 | 0.8471 | 0.8499 | 0.8523 | 0.8546 | 0.8568 |
| 4 | 0.8060 | 0.8097 | 0.8130 | 0.8160 | 0.8188 |
| 5 | 0.8386 | 0.8420 | 0.8453 | 0.8483 | 0.8513 |

relational information of the edges and nodes, and maintaining the same eigen centrality before and after applying privacy preserving algorithm. However, shuffling the edge weights comes with a trade-off of variation in the trends of eigen centrality values (trends here means the ranking of the nodes according to the eigen centrality values). Since, shuffling swaps the edge weights with the nearest other edge, it is preferable that the difference between the two edges is as small as possible (which is usually the case due to sorting), lesser the difference, more are the chances of shuffling not affecting the trends of eigen centrality values. For the example bipartite graph shown in Fig. 12, the variance is 0.134 and that resulted in 2 out of 10 number of nodes trend variations before and after camouflage DP. This is also evident in Table 5. However, in the actual real estate network, our calculations showed that the variance of the edge weights is not more than 0.05, since the network appears only at the end of the location identification algorithm and the link weights of this condominium network belongs to the same landmark as well. This small variance brings still lesser changes in the trends of centrality values, and usually do not affect the top condominiums and attributes. On the other hand, if the variance of edge weights are high, there can be high variations in the trends even affecting the top condominiums and attributes, which is not possible in a the current real estate network. After, the camouflage layer is the noise addition layer, which acts as a further coating to the shuffled edge weights and hides the actual edge weight information. However, additive noise is least significant in preserving the trend of eigen centrality values (as evident from Table 6). Hence, camouflage DP provides, privacy at two levels: hiding the relationship between the edges and nodes with a trade off in the changes in the trend of the eigen centrality values (usually negligible), and preserving the privacy of the edge weights by adding noise.

**Theorem 1.** *Given$G(V, E)$ real estate complete bipartite network with the bi-partitionsc and f, with m and nvertices respectively. Camouflage DP on$G(V, E)$ preserves the total number of edges inG which is mn.*

**Proof.** *Suppose there exists two vertices $v_i$ and $v_j$, $i = j = 1$ such that $v_i \in c$ and $v_j \in f$, wherecis the number of condominiums andfis the number of attributes/features then there exists only one edge sinceGis bipartite.*

$\therefore \deg(v_i) = \deg(v_j) = 1$

*Now, suppose there exists two vertices $v_j \in f$, such that$j = \{1, 2\}$ and $v_i \in c$*

*Then there exists edges connecting from every vertices inc to f.*

$\therefore \deg(v_i) = 2, \deg(v_j) = 1$. *Total number of edges inGis 2.*

*Suppose, there aren vertices in f then,* $\deg(v_j) = 1\ \forall j = \{1, 2, 3. .. n\}$ *and* $\deg(v_i) = n$.

*Suppose if there arem vertices in c, then* $\deg(v_i) = n$, $\deg(v_j) = m \forall\ i,j = \{1, 2, 3. . m\}$

*To find total edges it is sufficient to add the degrees of all the vertices in eitherc or f.*

*The sum of all degrees of vertices incis* $\sum_{i=1}^{m} \deg(v_i) = mn$.

*Suppose, if the edges are sorted and shuffled, and further added by noise the total number of edges still remain the same, that is,mn. Hence proved.* □

**Theorem 2.** *Camouflage DP preserves the vertex degree in real estate bipartite network*

**Proof.** *LetGis complete real estate bipartite network with bi-partitions asc and f with m and nvertices in them respectively, then* $\deg(v_i) = n$ *and* $\deg(v_j) = m\ \forall\ v_i \in c$ *and* $v_j \in f$. *Suppose there exists two vertices $v_i$ and $v_j$, $i = j = 1$ such that $v_i \in c$ and $v_j \in f$, wherecis the number of condominiums andfis the number of attributes/features then there exists only one edge sinceGis bipartite.*

$\therefore \deg(v_i) = \deg(v_j) = 1$

*Now, suppose there exists two vertices $v_j \in f$, such that$j = \{1, 2\}$ and $v_i \in c$*

*Then there exists edges connecting from every vertices inc to f.*

$\therefore \deg(v_i) = 2, \deg(v_j) = 1$. *Total number of edges inGis 2.*

*Suppose, there aren vertices in f then,* $\deg(v_j) = 1\ \forall j = \{1, 2, 3. .. n\}$ *and* $\deg(v_i) = n$.

*Similarly, we can prove by induction, that if there is only one vertex inf, and m vertices in c, then* $\deg(v_j) = m$. *Camouflage DP shuffles the edges and adds noise to the edge weights. This retains the position (or connection of an edge with a node) of an edge where as changes the magnitude. So, the degree of the nodes in the network remains unchanged. Hence proved.* □

**Theorem 3.** *Camouflage DP preserves the time complexity property of the location identification approach, specifically* $t < T$.

**Proof.** *Let us find the total number of edges for the architecture shown in Fig. 3 , which is complex network approach. In layer-1, if there arefnumber of attributes andLnumber of attributes then according to Theorem-1 there arefLnumber of edges. In layer-2, if there arefnumber of features andNnumber of condominiums then there arefNnumber of edges. Let the adjacency formed by edges in the layer-1 be$A_1$ and that of layer-2 be$A_2$. The time complexity involved in the centrality calculation be$h_1$ and $h_2$. Therefore the total time complexity is* $t = h_1 + h_2$. *The total number of edges in the network approach is$fL + fN = f(L + N)$. Let this be $Ed_1$.*

*In the combined approach, suppose the number of features arefand optimal condominiums aren′ then the total number of edges arefn′. Let this be $Ed_2$. The adjacency matrix formed by$Ed_2$ be $A_3$. Let Tbe the time complexity involved in the centrality calculation for$A_3$.*

*From Theorem-2, the Camouflage preserves the degree of the vertices. Hence, the$Ed_1$ and $Ed_2$ remains same after the camouflage privacy preservation.*

*Since,$n' < < N$ therefore $Ed_1 < < Ed_2$,* $t < T$. *Hence proved.* □

To summarize, camouflage DP shuffles the relation between the edges of a graph, keeping the positions of top condominiums and the attributes unchanged but, however, the network that a user receives is noisy (or camouflaged) due to network edge privacy preservation. This DP method can be extended for a group differential privacy, which is the future scope of this work.In the next section, another add-on on the location identification algorithm is discussed which is the use of blockchain for providing anonymity in the context of real estate investment.

## 8. Blockchains in real estate investment

Real estate investment is a business arena that contains potential competition between their peers. However, suggesting good condominiums in favor of one condominium owner (by using a software) may be offensive for others. This is the case if the user is a realtor and he suggests locations for investment to his clients using this developed algorithm. Revealing the realtor's identity may be a breach of privacy. Hence, in these cases, a blockchain can serve as a means of transaction. The idea is that, a client sends his attribute preference list over the chain. The remote software proprietor will reply with the list of condominiums and the network, which is privacy preserved over the blockchain anonymously.

Application of blockchains to solve societal problems are still in its infant stage, however, our work witnesses a method that could make use of the blockchains for real estate investment. In this section, first we introduce about blockchains and its types, Ethereum smart contracts in brief in general and later discuss the blockchain based real estate location identification in detail.

### *8.1. Understanding blockchains and smart contracts*

The blockchain came into existence in 2009 through the concept of crypto currencies (Conti, Sandeep Kumar, Lal, & Ruj, 2018). A blockchain contains secure history of data exchanges, utilizes peer to peer time stamp and verify the exchanges, and can be managed without the interference of a third party. The verification happens with the help of other peers in the network (through a consensus) and every transaction is saved in the block. Every user connected to the blockchain is entangled by two kinds of keys, private keys and public key linked to a wallet using which a user can perform transactions. A user can access his wallet using private keys and the public key (wallet address) is the one which is available for other peers in the network to perform transaction. Private keys gives a user, the power to digitally sign and validate every action initiated with his public key. Since, the wallet address is a key that comes out of encryption algorithm, it is a string of random characters for an unintended user. This makes a wallet owner anonymous to the outside world. The copy of the blockchain is kept in every machine connected to the network and hence there is no concept of centralized access. In addition, because of these multiple copies it is unable for anyone to tamper the contents of a chain. Anonymity and decentralization are the major backbones of blockchain technology. There are many kinds of blockchains (Fernández-Caramés & Fraga-Lamas, 2018), but majorly they are classified into two kinds (1) **Public blockchains:** The algorithms are open source and permission-less. Anyone can start downloading the code and create their own public node and access the network, free to see their transaction stored in the ledger (block) and anyone can view the transaction (they are transparent), but anonymously. For example: Bitcoin, Ethereum, and so on. (2) **Private blockchains:** write permissions are kept centralized to only one organization. Read permission can be public or restricted. A group of people may involve in the verification and others within a company. For example: MONAX, Multichain, and so on. A further granular level of classification can be achieved that involves, private permissioned, public permissioned, private permissionless and public permissionless blockchains.

### *8.2. Ethereum chains and smart contracts*

The underlying concept in Bitcoin and Ethereum (Ethereum, 2019) remain the same however, Bitcoin is a platform purely for payment and online currency transaction, in contradiction, Ethereum allows users to create a smart contract and tokens for transactions according to the application that they are building over the chain. A smart contract is more like conventional object-oriented programming and needs an Ethereum virtual machine (EVM) to be installed on the host machine. Once the code chunk executes, units of value may be transferred as easily as data. Ethereum is used to build economic systems in pure software. In other words, it is the software for business logic, wherein people can move money around with the speed and scale that we normally get with data. Like Bitcoin, Ethereum is also free and open source platform. *Ether* is the currency in Ethereum and its smallest unit is *Wei*. Every activity that modifies the state on Ethereum costs *Ether*. There is one more currency called *Gas* which is the internal currency of Ethereum. For every execution of a line in the code consumes *Gas* accordingly. The price of the *Gas* is fixed at the beginning and it won’t fluctuate with the market, unlike *Ether*. Ethereum blockchains can be coded using Go, Solidity, Rust, and C++. The current real estate smart contract application was developed using Solidity with Remix IDE (Chris Dannen, 2017). It is to be noted that every smart contract is having its own public address using which others can contact. This smart contract is run by a single or a group of owners. Ethers can be transferred between sender to smart contract, smart contract to another contract, and contract to the sender.

### *8.3. Real estate smart contract*

We assume that a service provider is a realtor who uses the developed algorithm for location identification and is unwilling to disclose his identity. Hence, the developed smart contract will provide a means of transaction, anonymously. First, a user (who wants to know the locations for investment) will send the request for locations to the smart contract using public address of smart contract. The smart contract will send the list of landmark clusters and its associated attributes to the user, together with the fee amount. The user sends the attribute preference list with the fee to the smart contract. This fee amount will be transferred to the realtor's account using his public address which is linked to the smart contract. The smart contract will send an instruction to the realtor [9] to execute the location identification algorithm and obtain the result (which is the list of condominiums and the network model which is privacy preserved). This result is uploaded to the IPFS (Inter Planetary File System; IPFS, 2019) and the web page links are requested by the contract and then these links are sent to the user, using which the user can access the result. The execution of the smart contract leads to Gas consumption. This will be deducted by the realtor's account for every execution (depending on the number of lines of codes a smart contract contains).In our case, as mentioned before, the smart contract account is same as the realtor account, however a separate account can be maintained for the smart contract. The smart contract is made available on the Ethernet network and the transactions are verified every time by miners. The blockchain architecture for real estate is shown in Fig. 17.

These are the following messages exchanged between a realtor, smart contract and the user.

**Cmessage():-** user sends a request message to the smart contract using its public address to send the list of top attributes.

**Smessage():-** Smart contract sends a message of that contains the list of attributes with its associated cluster of landmarks and the attribute maximum range within which a user is suppose to enter the numbers, with the fee details.

**CAmessage():-** This message contains the attributes with the magnitudes specified for every attribute.

**execute():-** smart contract sends a request to the realtor to execute real estate location identification algorithm. Realtor prepares the list of best condominiums and a network model (protected using DP).

[9] here a realtor is not willing to reveal his/her identity or instead of a realtor there can be a server which executes it can our proposed location identification algorithm every time a smart contract sends a message to it making the system automated. In this paper, the realtor is assumed to be a manual operator.

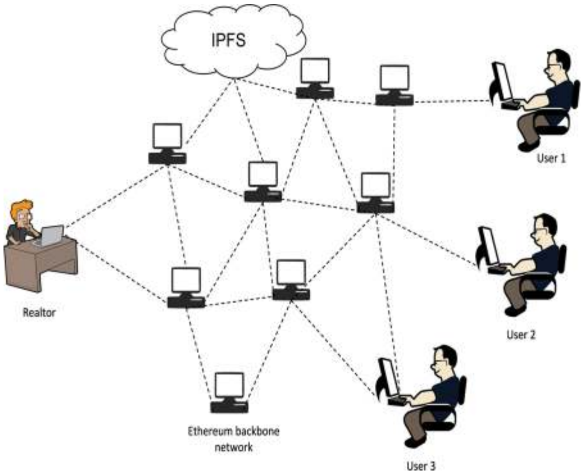

**Fig. 17.** Blockchain architecture for real estate investment location identification.

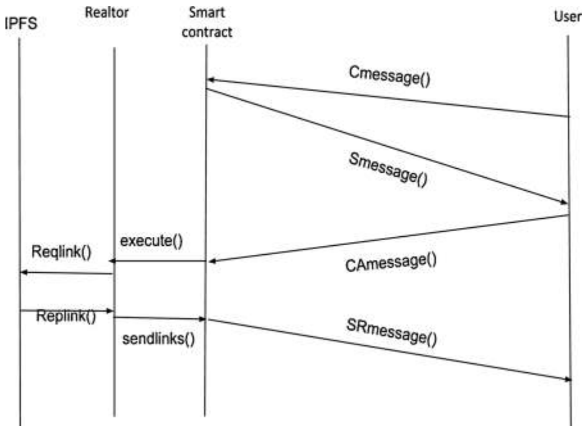

**Fig. 18.** Timing diagram for communication between user, smart contract and the Realtor.

**reqlink():-**Realtor uploads the image of the complex network and the list of condominiums and requests two links for the same respectively to the IPFS server.

**replink():-**IPFS replies with the links to the Realtor and these links are transfered to the smart contract using **sendlinks()**.

**SRmessage():-** This message contains the reply from the smart contract containing two links (say *link-1* and *link-2*) through which the network model and the list of best condominiums are made available to the client.

The timing diagram is shown in Fig. 18, that shows series of message exchanges between the various entities of blockchain architecture designed for real estate investment location transaction.

**Algorithm 1.** User side actions

```
Result: /* receive links from smart contract*/
Initialize acc_balance; /*Initialize the account of the user*/
Cmessage()
{
send request for landmark clusters and attributes to smart contract
}
receive reply from contract
CAmessage()
{
send attribute list with fee amount in Ethers}
decrement acc_balance
receive links
terminate transaction
```

**Algorithm 2.** Smart contract

```
Result link-1, link-2
Initialize Balance /* initializing the balance of the account to which the smart
    contract is linked*/
Receive request from user;
  Smessage()
  {
  send list of clusters with their landmarks and attributes with range
  send fee details for consultancy
  }
receive user_attributes
send service provider to execute the location identification algorithm()
/*service provider runs the function execute(), and run Reqlink(), receives through Replink() and send these links to smart contract using sendlinks()*/
deduct fee amount from the user's account
increment Balance
  SRmessage()
  {
  receive links from Realtor
  send links to the user
  }
terminate transaction
```

The entire smart contract was implemented using Solidity 0.4.0 using Remix IDE connecting to the Ropsten test network using dummy *Ether*coins from Ropsten faucet. The error free smart contract was deployed later on the main Ethereum network using myetherwallet (2019).

## 9. Implications of the current work from a smart city perspective

Technological development has a huge impact on converting a modern residential city into a smart city. A smart city comprises of smart mobility, smart living, smart environment, smart citizens, smart government, smart economy, smart architecture and technologies (Ismagilova, Hughes, Yogesh, Dwivedi, & Raman, 2019). According to Khalid, Eldrandaly, and Laila (2019), smartness of a city is determined by the structure and functions of its data, information and knowledge management system. Various technological studies and their application in solving the issues of smart cities is often seen in literature (Duan, Edwards, & Dwivedi, 2019; Dwivedi et al., 2019; Janssen, Luthra, Mangla, Rana, & Dwivedi, 2019; Li, Deng, Lee, & Wang, 2019; Singh et al., 2019). In this direction, the work dealt in this paper uses current technological trends in solving the location identification problem in real estate investment. With the growing awareness of smart cities, the use of sophisticated and intelligent techniques to solve issues of a city is very much required (Rana et al., 2019). Hence, the architecture that is discussed in this paper which combines the aspects of data science, network science, privacy and blockchains satisfies the need.

Identification of locations is not only an important aspect of real estate investment; there are numerous other use cases to which this work can be extended like: finding locations of a disease outbreak and spread (for smart healthcare), finding locations of criminal activities outbreak and spread (for smart infrastructure management), finding locations of traffic congestion and outbreak (for smart transportation system), and so on. In totality, wherever location plays a crucial role, the method presented in this paper can be applied for selecting the optimal ones, which is of a major interest for any relevant stakeholders.

## 10. Conclusions

In the previous work authors used tools from data science to compute investment locations for real estate investment. However, the relational dynamics between attributes of real estate investment are not easy to compute and/or visualize. Two novel solutions are presented to address this. The first solution uses tools from network science to derive the relational dynamics and compute the investment locations. However, the network science approach is shown to be computationally inferior to the data science approach. To get both computational advantage and relational inferences, the second solution in this paper combines the data science approach with the network science approach.

Further, since real estate investment tools and databases are online, there are inherent privacy risks. A novel algorithm, camouflage DP, is designed and implemented to demonstrate real estate network privacy preservation. In addition, blockchains are used as medium of providing anonymous transactions. As future scope, we aim to extend the DP techniques to enrich the privacy preservation in real estate networks and the use of blockchains for secure real estate transactions.

## References

Aldrich, C. (2002). *Exploratory Analysis of Metallurgical Process Data with Neural Networks and Related Methods.* Elsevier Publications Edition-1 (book).

Goldberg, A. K., "Towards Differentially Private Inference on Network Data", Thesis, Applied Mathematics, Harvard College, Cambridge.

Anand, Bihari, & Pandia, Manoj Kumar (2015). Eigenvector centrality and its application in research professionals' relationship network. *2015 International Conference on Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE),* 510–514.

"Inter Planetary File System", accessed on 22/10/2019 [online] (2019). Available at: https://ipfs.io/.

"myetherwallet", accessed on 22/10/2019 [online] (2019). Available at: https://www.myetherwallet.com/.

Balle Borja, Wang Yu-Xiang, Improving the Gaussian Mechanism for Differential Privacy: Analytical Calibration and Optimal Denoising arXiv:1805.06530.

Battiston, S., Glattfelder, J. B., Garlaschelli, D., Lillo, F., & Caldarelli, G. (2010). The structure of financial networks. *Network Science, 131-163*.

Bhat, Aditya, Santhosh Kumar, T., Kalburgi, Udith Manohar, Sandeep Kumar, E., Viswanath Talasila, T., Suresh Kumar, V., Rishe, Naphtali, & Kusuma, S. M. (2018). Significant Social Factors of Real Estate Investment. *7th Int. conference on communication and signal processing (ICCSP)-Tamil Nadu, India*.

Byeonghwa park, & Jae, K. B. (2015). Using machine learning algorithms for housing price prediction: The case ocf Fairfax County, Virginia housing data. *Expert Systems with Applications, 42*(6), 2928–2934 ISSN: 0957-4174.

Chris Dannen (2017). *Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners".* Apress Publications.

Conti, M., Sandeep Kumar, E., Lal, C., & Ruj, S. (2018). A Survey on Security and Privacy Issues of Bitcoin. *IEEE Communications Surveys & Tutorials, 20*(4), 3416–3452. https://doi.org/10.1109/COMST.2018.2842460 Fourthquarter.

"magicbricks", accessed on 22/10/2019 [online] (2019). Available at: https://www.magicbricks.com/.

"99acres", accessed on 22/10/2019, [online] (2019). Available at: https://www.99acres.com/.

Cambridge dictionary, "meaning of camouflage", accessed on 22/10/2019 [online] (2019). Available at: https://dictionary.cambridge.org/dictionary/english/camouflage.

Cynthia Dwork, "Differential Privacy", Microsoft research, white paper.

DArcangelis, A. M., & Rotundo, G. (2016). Complex Networks in Finance. In P. Commendatore, M. Matilla-Garca, L. Varela, & J. Cnovas (Eds.). *Complex Networks and Dynamics. Lecture Notes in Economics and Mathematical Systems, vol. 683.*. Springer Cham..

Duan, Y., Edwards, J. S., & Dwivedi, Y. K. (2019). Artificial intelligence for decision making in the era of Big Dataevolution, challenges and research agenda. *International Journal of Information Management, 48*, 63–71.

Dwivedi, Y. K., Hughes, L., Ismagilova, E., Aarts, G., Coombs, C., Crick, T., & Galanos, V. (2019). Artificial Intelligence (AI): Multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy. *International Journal of Information Management*. https://doi.org/10.1016/j.ijinfomgt.2019.08.002.

Guo, Y., & Xue, Y. (2009). Research on Cooperative Relations for Identifying Abnormal Vertices in Complex Financial Networks," 2009 International Conference on Electronic Commerce and Business Intelligence. *Beijing,* 37–40. https://doi.org/10.1109/ECBI.2009.70.

Nguyen, Hiep H., Imine, Abdessamad, & Rusinowitch, Michal (2016). Network Structure Release under Differential Privacy". *Trans. Data Privacy, 9*(3 (December)), 215–241.

Hsu, et al. (2014). Differential Privacy: An Economic Method for Choosing Epsilon. *2014 IEEE 27th Computer Security Foundations Symposium, Vienna,* 398–410.

Ismagiloiva, E., Hughes, L., Rana, N., & Dwivedi, Y. (2019b]). *Role of Smart Cities in Creating Sustainable Cities and Communities: A Systematic Literature Review. In International Working Conference on Transfer and Diffusion of IT.* Cham: Springer311–324 June.

Ismagilova, E., Hughes, L., Yogesh, K., Dwivedi, K., & Raman, R. (2019b]). Smart cities: Advances in research An information systems perspective. *International Journal of Information Management, 47*, 88–100. https://doi.org/10.1016/j.ijinfomgt.01.2019.004 ISSN 0268-4012.

Israilidis, John, Odusanya, Kayode, & Mazhar, Muhammad Usman (2019). Exploring knowledge management perspectives in smart city research: A review and future research agenda. *International Journal of Information Management*. https://doi.org/10.1016/j.ijinfomgt.07.2019.015 ISSN 0268-4012.

Janssen, M., Luthra, S., Mangla, S., Rana, N. P., & Dwivedi, Y. K. (2019). Challenges for adopting and implementing IoT in smart cities. *Internet Research*. https://doi.org/10.1108/INTR-06-2018-0252.

Jiajia Ren, "Stock Price Dynamics Before Crashes: A Complex Network Study on the U.S Stock Market", Thesis document, University of Lethbridge.

Karamitsos, I., Papadaki, M., & Barghuthi, N. (2018). Design of the Blockchain Smart Contract: A Use Case for Real Estate. *Journal of Information Security, 9*, 177–190. https://doi.org/10.4236/jis.2018.93013.

Khalid, A., Eldrandaly, Mohamed Abdel-Basset, & Laila, Abdel-Fatah (2019). PTZ-Surveillance coverage based on artificial intelligence for smart cities. *International Journal of Information Management*. https://doi.org/10.1016/j.ijinfomgt.04.2019.017 ISSN 0268-4012.

Kong, Xiangyu, Hu, Changhua, & Duan, Zhansheng (2017). *Principal Component Analysis Networks and Algorithms".* Springer Publications.

Krupa, K. S., & Akhil, M. S. (2019). Reshaping the Real Estate Industry Using Blockchain. In V. Sridhar, M. Padma, & K. Rao (Eds.). *Emerging Research in Electronics, Computer Science and Technology. Lecture Notes in Electrical Engineering, vol. 545*. Singapore: Springer.

Lerman, K., Lain, P., Ghosh, R., Kang, J. H., & Kumaraguru, P. (2013). *Limited Attention and Centrality in Social Intelligence and Technology.* PA: State Collegehttps://doi.org/10.1109/SOCIETY.2013.11 80-8.

Li, Xiaoye, Yang, Jing, Sun, Zhenlong, & Zhang, Jianpei (2017). Differential Privacy for Edge Weights in Social Networks. *Security and Communication Networks, 2017*, 10. https://doi.org/10.1155/2017/4267921 Article ID 4267921.

Li, Daming, Deng, Lianbing, Lee, Minchang, & Wang, Haoxiang (2019). IoT data feature extraction and intrusion detection system for smart cities based on deep migration learning. *International Journal of Information Management*. https://doi.org/10.1016/j.ijinfomgt.04.2019.006 ISSN 0268-4012.

Lilla Tthmrsz, "Differential Privacy", M.S Thesis.

Liu, D., Yang, D., & Liu, J. (2007). Complex Networks Model for Residential Real Estate Markets," 2007 International Conference on Wireless Communications, Networking and Mobile Computing. *Shanghai,* 5737–5740. https://doi.org/10.1109/WICOM.2007.1406.

Liu, B., Mavrin, B., Niu, D., & Kong, L. (2016). House Price Modeling over Heterogeneous Regions with Hierarchical Spatial Functional Analysis. *2016 IEEE 16th International Conference on Data Mining (ICDM), Bar-celona,* 1047–1052.

Liu, Fuyu, Wang, NIingkui, & Wei, Daijun (2017). Analysis of Chinese Stock Market by Using the Method of Visibility Graph. *The Open Cybernetics & Systemics Journal, 11*. https://doi.org/10.2174/1874110X01711010036.

Narayan Prusty (2017). *Building Blockchain projects".* Packt publishers.

Rana, N. P., Luthra, S., Mangla, S. K., Islam, R., Roderick, S., & Dwivedi, Y. K. (2019). Barriers to the Development of Smart Cities in Indian Context. *Information Systems Frontiers, 21*(3), 503–525.

Reutskaja, Elena, et al. (2018). Choice overload reduces neural signatures of choice set value in dorsal striatum and anterior cingulate cortex. *Nature Human Behavior* October.

Sandeep Kumar, E., Viswanath Talasila, Naphtali Rishe, T., Suresh Kumar, V., & Iyengar, S. S. (2019). *Location Identification for Real Estate Investment using Data Analytics, Int, Journal of Data Science and Analytics.* Springer Publications (Jan 2019)https://doi.org/10.1007/s41060-018-00170-0.

Singh, P., Dwivedi, Y. K., Kahlon, K. S., Sawhney, R. S., Alalwan, A. A., & Rana, N. P. (2019). Smart Monitoring and Controlling of Government Policies Using Social Media and Cloud Computing. *Information Systems Frontiers,* 1–23. https://doi.org/10.1007/s10796-019-09916-y.

Soontornphand, T., & Natwichai, J. (2016). Joint Attack: A New Privacy Attack for Incremental Data Publishing. *2016 19th International Conference on Network-Based Information Systems (NBiS), Ostrava,* 364–369.

Szmidt, E., & Kacprzyk, J. (2010). The Spearman rank correlation coefficient between intuitionistic fuzzy sets. *2010 5th IEEE International Conference Intelligent Systems, London,* 276–280. https://doi.org/10.1109/IS.2010.5548399.

Lei Tang, Huan Liu, "Community Detection and Mining in Social Media", Morgan and Claypool publishers(book).

Fernández-Caramés, T. M., & Fraga-Lamas, P. (2018). A Review on the Use of Blockchain for the Internet of Things. *IEEE Access, 6*, 32979–33001.

Timothy, Oladunni, & Sharma, Sharad (2016a]). "Hedonic Housing Theory" A Machine Learning Investigation. *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA),* 522–527.

Timothy, Oladunni, & Sharma, Sharad (2016b]). Spatial Dependency and Hedonic Housing Regression Model. *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA),* 553–558.

Wei, W., Guang-ji, T., & Hong-rui, Z. (2010). Empirical analysis on the housing price in Harbin City based on hedonic model. *2010 International conference on Management Science and Engineering 17th Annual Conference Proceeding, Melbourne, VIC,* 1659–1664.

Wu, Yenchun Jim, & Chen, Jeng-Chung (2019). A structured method for smart city project selection. *International Journal of Information Management*. https://doi.org/10.1016/j.ijinfomgt.2019.07.007 0268-4012.

Xue, H. (2015). The Prediction on Residential Real Estate Price Based on BPNN. *2015 8th International Conference on Intelligent Computation Technology and Automation (ICICTA), Nanchang,* 1008–1013.

ZHANG, D. E. N. G., Yinghui, Robert H., Ximeng, L. I. U., & ZHENG, Dong (2018). Blockchain based efficient and robust fair payment for outsourcing services in cloud computing. *Information Sciences, 462*, 262–277 Research Collection School Of Information Systems.

Zhang, Y., Liu, S., He, S., & Fang, Z. (2009). Forecasting research on real-estate prices in

Shangai. *2009 International Conference on Grey Systems and Intelligent Services (GSIS 2009), Nanjing,* 625–629.

Zhou, B., & Pei, J. (2008). Preserving Privacy in Social Networks Against Neighborhood Attacks. *2008 IEEE 24th International Conference on Data Engineering, Cancun,* 506–515.

"Terrafly" Database, accessed on 22/10/2019 [online] (2019). Available at: http://www.terrafly.com.

"Ethereum" accessed on 22/10/2019 [online] (2019). Available at - https://www.ethereum.org/.

"Reply", accessed on 22/10/2019 [online] (2019). Available at: https://www.reply.com/en/content/real-estate.

"Ubitquity", accessed on 22/10/2019 [online] (2019). Available at: https://www.ubitquity.io/.