

Dynamic Data Visualizing Using Maps

¹Parvathy.R Assistant Professor, ²M.Kalidass Assistant Professor

¹Department of Computer Science and Engg T K M Institute Of Technology, Karuvelil, Kollam, Kerala

²Department of Computer Science and Engg RVS Educational Trust's Group of Institution, Dindigul

ABSTRACT

Visualizing large-scale dynamic relational data by taking advantage of the geographic map metaphor is considered. A map-based visualization system uses animation to convey dynamics in large data sets, and which aims to preserve the viewer's mental map while also offering readable views at all times is described. One of the main problems in dynamic visualization is that of obtaining individually readable layouts for each moment in time, while at the same time preserving the viewer's mental map. This way of visualizing the trends in popularity of dynamic data by the use of maps overcomes the shortcomings of data analysis via animation.

Keywords: Maps, Dynamic Data, Data Mining, GMap1 Algorithm, Clustering

Date of Submission: 15-08-2017

Date of acceptance: 20-09-2017

I. INTRODUCTION

1.1 Overview

Visualization is a technique for creating images, diagrams, or animations to communicate a message. Dynamic data are real time, high volume data. Dynamic visualization techniques allow users to quickly grasp these data without requiring intensive training in the use of specific visualization tools, thereby facilitating human understanding of the dynamic data. This creates an appealing and informative visualization for the general public on the various trends and patterns of dynamic data sets.

Static maps of relational data lead to visually appealing representations, which show more than just the underlying vertices and edges. Specifically, by explicitly grouping vertices into different coloured regions, viewers of the data can quickly identify clusters and relations between clusters. Moreover, this explicit grouping leads to easy identification of central and peripheral vertices within each cluster. Extending traditional graph drawing algorithms

From static to dynamic graphs is a difficult problem. In most proposed solutions, the typical challenges are those of preserving the mental map of a viewer and ensuring readability of each drawing.

Changes are visualized by animation, which can be generated by concatenating static maps, thus providing continuity from one layout to the next. Whereas in dynamic graph drawing it is perfectly reasonable to have vertices move from one moment in time to the next. Also, if the layout from one time to the next is significantly different, it is likely that viewers will quickly get lost. A

common way to deal with this problem is anchoring some vertices that appear in two or more subsequent drawings. Additionally, the way to encode metrics and changes into the map metaphor needs to be considered.

1.2 Data Mining

Data mining is the method of automatically searching large amount of data to discover patterns and trends that go beyond simple analysis. Data mining uses sophisticated mathematical algorithms to divide the data and evaluate the probability of that in future events. Data mining is the process of analyzing the steps of the "knowledge discovery in databases" process, or KDD. The main properties of data mining are:

- Pattern Discovery
- Prediction of probable outcomes
- Generation of actionable information
- Focus on very large data sets and large databases. Data mining can answer questions that cannot be addressed using simple query and reporting techniques.

1.2.1 Automatic Discovery of data

Data mining is done by building models. It uses an algorithm to act on a set of data. The concept of automatic discovery refers to the working of the data mining models. The models can be used to mine the data on which they are constructed, but most types of models are reduce to a general form of new data. The process of applying a model to new data is called **scoring**.

1.2.2 Prediction Probability of data

Many forms of data mining are prediction based. For example, a model might predict income based on teaching and other factors that deals with population. Predictions is associated with probability (How probably is be true?). Prediction probabilities are called **confidence** some forms of predictive data mining will create **rules**, which are conditions that imply on a given outcome of several. For example, a rule might specify that a person who has a master's degree and lives in a USA is likely to have an income greater than the 1000USD. Rules have an associated with certain **support**.

1.2.3 Grouping or gathering of data

Different forms of data mining will identify natural clustering in the data. For example, a model might identify the segment of the people that has an income within a 1000USD, that has a good leadership quality, and that guide 10 persons.

1.2.4 Production of Actionable Information

Data mining can produce actionable information from a large amount of data. For example, a planner might use a model that predicts income based on population based to develop a plan for low-income housing.

Knowledge discovery consists of an iterative sequence of the following steps:

- Data cleaning (to remove noise and inconsistent data).
- Data integration (where multiple data sources may be combined).
- Data selection (where data relevant to the analysis task are retrieved from the database).
- Data transformation (where data are transformed and consolidated into forms appropriate for mining)
- Data mining (an essential process where intelligent methods are applied to extract data patterns)
- Pattern evaluation (to identify the truly interesting patterns representing knowledge based on some interestingness measures).
- Knowledge presentation (where visualization and knowledge representation techniques are used to present the mined knowledge to the user)

1.3 Purpose of Data Mining

Data mining is one component of the exciting area of machine learning and adaptive computation. The goal of building computer systems that can adapt to their environments and learn from their experience has attracted researchers from many fields, including computer science,

engineering, mathematics, physics, neuroscience, and cognitive science. Out of this research has come a wide variety of learning techniques that have the potential to transform many scientific and industrial fields. The purpose of data mining serves to discover (hidden, non trivial) patterns in large amounts of data records in order to be used very effectively for (ex post) analysis and (ex ante) forecasting. Data mining is widely used in many fields such as,

- Telecommunication
- Banking
- Insurance
- University
- Tourism

II. PROPOSED SOLUTION

2.1 Data Structure Design

The input to the algorithm is a relational data set, from which a graph $G = (V, E)$ is extracted. The set of vertices V corresponds to the objects in the data, and the set of edges E corresponds to the relationship between pairs of objects. In its full generality, the graph is vertex-weighted and edge weighted, with vertex weights corresponding to some notion of the importance of a vertex, and edge weights corresponding to some notion of the closeness between a pair of vertices.

2.1.1 GMap1 Algorithm

For visualizing dynamic data using maps, GMap1 algorithm for static data is used and later on converted for dynamic setting. GMap1, an algorithm to represent general graphs as maps. Clearly, there are theoretical limitations to what graphs can be represented exactly by touching polygons, namely, subclasses of planar graphs. The input to the algorithm is a relational data set, from which a graph $G = (V, E)$ is extracted. The set of vertices V corresponds to the objects in the data, and the set of edges E corresponds to the relationship between pairs of objects. In its full generality, the graph is vertex-weighted and edge weighted, with vertex weights corresponding to some notion of the importance of a vertex, and edge weights corresponding to some notion of the closeness between a pair of vertices. GMap1 Algorithm is as follows:

Step 1: The graph is embedded in the plane using a scalable force direct algorithm or multidimensional scaling (MDS).

Step 2: Cluster analysis of the underlying graph is done or the point set from step one is performed. This is done in order to group vertices into clusters, using a modularity- based clustering algorithm. The clustering and embedding algorithm should match,

embedding derived from a force-directed algorithm, a modularity based clustering could be a better fit. The two algorithms are strongly related and therefore vertices that are in the same cluster also be physically close to each other in the embedding are expected.

Step 3: Two-dimensional embedding is done along with the clustering is used to create the map. In the final step, the nodes are coloured using a coloring algorithm to maximize colour differences between neighboring nodes .The geographic map corresponding to the data set is created, based on a modified Voronoi diagram of the vertices, which in turn is determined by the clustering and embedding.

2.1.2 Classification

Classification creates a model based on which a new instance can be classified into the existing classes or determined classes. To start with classification the CSV file related to publications is stored. A data structure using this CSV file is created for classifying new or unknown subjects. A generic approach for classification is used. This approach supports for any CSV format. Each line in the CSV file is considered as an entry in the CSV Data structure i.e. Similarity based on keyword is used.

2.1.3 Clustering

Clustering is formation of groups on the basis of its attributes and is used to determine patterns from the data. Advantage of clustering over classification is each and every attribute is used to define a group but disadvantage of clustering is a user must know beforehand how many groups he wants to form.

There are two types of clustering:

- Hierarchical clustering: This approach uses measure (generally squared Euclidean) of distance for identifying distance between objects to form a cluster. Process starts with all the objects as separate clusters. Then on the basis of shortest distance between clusters two objects closest are joined to form a cluster. And this cluster represents the new object. Now again the process continues until one cluster is formed or the required number of cluster is reached.
- Non-Hierarchical Clustering: It is the method of clustering in which partition of observations (say n in number) occur into k clusters. Each observation is assigned to nearest cluster and cluster mean is recalculated.

In this project hierarchical clustering method based on the occurrence of a particular subject is used. Given a set of N items to be clustered, and take a

N*N distance (or similarity) matrix, the method of hierarchical clustering is as follows:

Step 1: Start the process by assigning each item to a cluster that is if there are N items, N clusters are formed, each of which contains just one item. Let the distances between the clusters is as same as the distances between the items they contained in it.

Step 2: Find the pair that are closer to each other in the cluster and make it in a single cluster, so that one cluster is reduced.

Step 3: Calculate the distances between the clusters, the new and the old one.

Step 4: Repeat steps 2 and 3 until all items are clustered to form a single cluster of size N.

Step3, done with modified form of single-Linkage Clustering algorithm.

2.1.4 Single-Linkage Clustering Algorithm

Single-Linkage Clustering Algorithm is an example of agglomerative hierarchical clustering method. This is a bottom-up strategy: compare each point with each

point. Each object is placed in a separate cluster, and at each step we merge the closest pair of clusters, until certain termination conditions are satisfied. This requires defining a notion of cluster proximity.

For the single link, the proximity of two clusters is defined as the minimum of the distance between any two points in the two clusters. Using graph terminology, it starts with all points, each on a separate cluster on its own (called a singleton cluster), and then add links between all points one at a time, shortest links first, then these single links combine the points into clusters. (I.e. the points with the shortest distance between each other are combined into a cluster first, then the next shortest distance are combined, and so on) The N*N proximity matrix is $D = [d(i, j)]$. The clusters are assigned sequence numbers 0, 1... (n-1) and L

(k) is the level of the kth clustering. A cluster with sequence number m is denoted (m) and the proximity between clusters (r) and (s) is denoted $d[(r), (s)]$. Algorithm is as follows:

Step 1: Plot the objects in n-dimensional space (where n is the number of attributes).(Here its two dimensional).

Step 2: Calculate the distance from each object (point) to all other points, using Euclidean distance measure, and place the numbers in a distance matrix.

The formula for Euclidean distance between two points i and j is:

$$d(i, j) = \sqrt{|x_{i1} - x_{j1}|^2 + |x_{i2} - x_{j2}|^2 + \dots + |x_{ip} - x_{jp}|^2}$$

Where x_{i1} is the value of attribute 1 for i and x_{j1} is the value of attribute 1 for j, and so on, as many

attributes we have ... shown up to $p - x_{ip}$ in the formula.

- Step 3: Identify the two clusters with the shortest distance in the matrix, and merge them together. Re-compute the distance matrix, as those two clusters are now in a single cluster, (no longer exist by themselves).
- Step 4: Repeat Step 3 until all clusters are merged

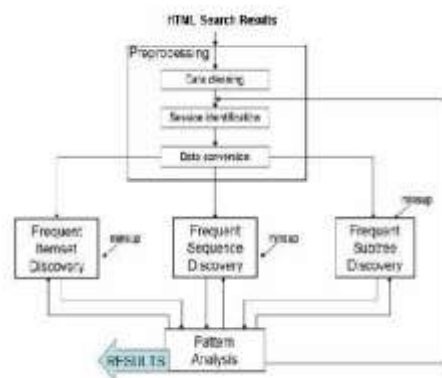
4.2. View definition to the data structure

Creating maps from extracted graph data by providing animation to the relevant attributes. Graphs capture relationships between objects and graph drawing allows us to visualize such relationships. Typically vertices are represented by points in two or three dimensional space, and edges are represented by lines between the corresponding vertices. The layout optimizes some aesthetic criteria, such as, showing underlying symmetries, or minimizing the number of edge crossings. There is also a large body of work on representing planar graphs as contact graphs where vertices are embodied by geometrical objects and edges are shown by two objects touching in some specified fashion.

4.3 GUI (Application design and development)

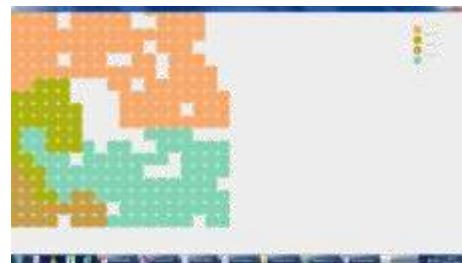
Integrating the above mentioned modules to develop a graphical user interface. In computing, a graphical user interface is a type of user interface that allows users to interact with electronic devices using images rather than text commands. GUIs can be used in computers, hand-held devices such as MP3 players, portable media players or gaming devices, household appliances, office and industry equipment. A GUI represents the information and actions available to a user through graphical icons and visual indicators such as secondary notation, as opposed to text-based interfaces, typed command labels or text navigation. The actions are usually performed through direct manipulation of the graphical elements.

Block diagram for web data mining



III. RESULT

A map with clusters of subjects is shown in figure .The map shown in the figure has four clusters of data based on the subjects computer, geology, science and literature. And four different colors are used to represent different clusters. User can easily perform pattern analysis in this map. The pattern analyzed and concluded from figure5.5 is that the dataset contains more number of publications related to the subject computer science.



IV. CONCLUSION AND FUTURE SCOPE

In this project, a way to visualize large-scale dynamic relational data with the help of the geographic map metaphor is explored. Some challenges created by the dynamics in the data were addressed. The map applicability of our approach is not limited to one dataset. For example, this scheme can be used, with minor modification in the data collection module, to visualize trends in the popularity of websites, TV shows, etc., where similarity and popularity information are easy to define.

The major component of future work is the evaluation of the effectiveness of visualization through the user study. This would also include the calibration of parameters (duration of animation, interval for difference calculation, etc.).Phase1 is implemented for static data setting and hierarchical clustering method is used, a functional interactive interface on using non hierarchical clustering has to be implemented. As the underlying data are map, a pan-and-zoom Google-Maps-like interaction has to be explored. In this direction, a searchable interface for static maps which allows users to zoom in and out of our maps and explore by means of intuitive mouse operations has to be developed.

The integration of trend visualization into such an interface is a part of future work. The resulting system can be further enhanced by allowing access to external online content by clicking on node labels. Finally, offering several metrics for visualization would result in a more powerful system.

REFERENCES

- [1]. Daisuke Mashima, Stephen G. Kobourov, and Yifan Hu, “Visualizing Dynamic Data with Maps”, IEEE Transactions on Visualization and Computer Graphics, Vol. 18, No. 9, Sep 2012.
- [2]. C.-C. Liao, H.-I.Lu, and H.-C. Yen, “Compact floor-planning via orderly spanning trees.Journal of Algorithms”, 48:441–451, 2003.
- [3]. M. B. Dillencourt, D. Eppstein, and M. T. Goodrich, Choosing colors for geometric graphs via color space embeddings. In 14th Symposium on Graph Drawing (GD), pages 294–305, 2006.
- [4]. Y. F. Hu, “Efficient and high quality force-directed graph drawing”. *Mathematica Journal*, 10:37–71, 2005.
- [5]. E. R. Gansner, Y. F. Hu, and S. G. Kobourov. GMap1, “Visualizing graphs and clusters as maps”. In IEEE Pacific Visualization Symposium, pages 201–208, 2010.
- [6]. Herman, G. Melancon, and M. S. Marshall.
- [7]. “Graph Visualization and Navigation in Information Visualization”: A survey. *IEEE Transactions on Visualization and Computer Graphics*, 6(1):24–43, 2000.
- [8]. J.Branke, “Dynamic Graph Drawing”, drawing graphs, vol.2025, pp.228-246, 2001.
- [9]. C. Collins, G. Penn, and S. Carpendale, “Bubble Sets: Revealing Set Relations with Isocontours over Existing Visualizations,” *IEEE Trans. Visualization and Computer Graphics*, vol. 15, no. 6, pp. 1009-1016, Nov./Dec. 2009.
- [10]. S. Diehl and C. Görg, “Graphs, They Are Changing,” *Proc. 10th Symp. Graph Drawing (GD)*, 23-30, 2002.
- [11]. C. Erten, P.J. Harding, S.G. Kobourov, K. Wampler, and G. Yee, “GraphAEL: Graph Animations with Evolving Layouts,” *Proc. 11th Symp. Graph Drawing (GD)*, pp. 98-110, 2003.
- [12]. T. Fruchterman and E. Reingold, “Graph Drawing by Force Directed Placement,” *Software—Practice and Experience*, vol. 21, pp. 1129-1164, 1991.
- [13]. E.R. Gansner and Y.F. Hu, “Efficient Node Overlap Removal Using a Proximity Stress Model,” *Proc. 16th Symp. Graph Drawing (GD)*, vol. 5417, 206-217, 2008.
- [14]. E.R. Gansner, Y.F. Hu, S. Kobourov, and C. Volinsky, “Putting Recommendations on the Map: Visualizing Clusters and Relations,” *Proc. Third ACM Conf. Recommender Systems (RecSys)*, pp. 345-348, 2009.

Parvathy.R Assistant Professor. “Dynamic Data Visualizing Using Maps.” *International Journal of Engineering Research and Applications (IJERA)*, vol. 7, no. 9, 2017, pp. 28–32.