Technical Report

N°10/2005

Visualization of semantic similarity

Alessio Bertone,

Riccardo Albertoni,

Monica De Martino

Istituto per la Matematica Applicata e Tecnologie Informatiche

CNR - Genova

Index

1 Introduction

This technical report aims to investigate the problems related to the visualization of similarity and to present an overview of the main available tools.
Despite the similarity (then the distance measure) plays a central role in several activity as information retrieval, exploration and analysis, the most of research activity concerning the similarities has been carried out within the field of ontology alignment, and the representation of the similarity is limited to the visualization of ontologies e.g. in form of trees, avoiding to solve problems such as the overriding of elements when too many elements are displayed, the lack of proper search capabilities, and so on.
The report is organized as follows: firstly the main rules and navigational techniques to be taken into account are illustrated, then it is outlined the connection between the choice of a certain distance measure and its representation; finally a brief overview of the related works and the available tools to visualize the similarity is given.

2 Information Visualization and similarity

Information visualization is a complex research area. It builds on theory in information design, computer graphics, human-computer interaction and cognitive science.
Practical application of information visualization in computer programs involves selecting, transforming and representing abstract data in a form that facilitates human interaction for exploration and understanding. Important aspects of information visualization are the interactivity and dynamics of the visual representation. Strong techniques enable the user to modify the visualization in real-time, thus affording unparalleled perception of patterns and structural relations in the abstract data in question. Thus, to make Information Visualization effective the human factor and how it affects the human perception must be taken into account.

2.1 Human Perception and Graphical Primitives

One of the most important issues in scientific data visualization is mapping attributes of data into graphical primitives which effectively convey the informational content of data. In general, this mapping defines an abstract visualization technique for the given data. However, there are several possible mappings which may lead to different visualization technique designs. Selecting and creating the most effective design among all the alternatives for a given situation usually requires considerable knowledge and creativity on the part of the visualization technique designer. While the knowledge about characteristics of data, such as types, units, scales, and spacing among measurement points, as well as graphical primitives, which eventually compose a design, is important in constructing visualization techniques, the knowledge about comprehensibility of the resulting image is essential for effective presentation of the information inherent in the data. Usually, the latter type of knowledge is in the form of heuristic rules and principles that are acquired through experience and experimentation. On the contrary, the former one can be more formally defined and in particular, this paragraph focuses on the definition of the graphical primitives.

Bertin (Bertin 1981) identifies 7 graphical primitives from which the images are built:

- Size
- Brightness
- Colour
- Saturation
- Orientation
- Shape
- Texture

There is a variety of studies about the perception of these primitives, here it follows some statements:

- **size**
  - human eye can distinguish between up to 20 different sizes using a ratio of 1:10 between smallest and biggest size.
  - Differences in size are better perceived for dark surfaces.

- **brightness:**
  - human eye can distinguish between 60-70 grey levels.
  - For representing nominal data only 5 – 6 levels should be used.

- **colour:**
  - number of distinguishable colours levels relates directly to the size of the coloured surface.
  - Diameter should be at least 1,5 mm to perceive colour differences.
  - Colour perception also depends on the adjacent colours.

- **saturation:**
  - selecting property is best for pure (fully saturated) colours. Pure colours should be used for small surfaces which represent extreme values.
  - Saturation differences are harder to perceive than brightness differences.

- **orientation:**
    - direction is best perceived for icons with a longish shape.
    - Directions are well distinguishable if angle is chosen between 30° and 60°.
    - Number of direction in a visualization is limited.

- **shape:**
    - map from data to shape is complicated.
    - Shape is appropriate as visual metaphor.
    - Shapes can be obtained by combinations e.g. of directions for star shaped coordinates.

- **texture:**
    - A collection of small objects is generally perceived as texture / pattern.
    - A texture with few and isolated objects is associated with „less“; a texture with many objects is associated with many objects is associated with „many“.

Further rules**:**

- A difference of the shape of two objects can easier be perceived than a difference in the size.
- Contrast facilitates perception as well as lighting between lower and upper border.
- Regular geometric shapes are easier perceivable than asymmetric shapes.
- It is important not to overload the human visual system in a visual representation. (The solutions is the adoption of techniques such as focus & context, linking techniques,…).

2.2 Typical Layout for Graph Visualization

Since the similarities are often represented as graphs, here it follows a brief overview of the typical layouts for the graph visualization and of the main navigational techniques.

Here it follows the typical layouts for graph visualizations:

- A *Tree Layout* will position children nodes “below” their common ancestor.

- *H–tree layouts* are also classical representations for binary trees which only perform well on balanced trees.

- The radial positioning (or “*Radial View*”) places nodes on concentric circles according to their depth in the tree A subtree is then laid out over a sector of the circle and the algorithm ensures that two adjacent sectors do not overlap

- The cone tree algorithm can be used to obtain a “*balloon view*” of the tree by projecting it onto the plane, where sibling subtrees are included in circles attached to the father node.

- The *tree–maps* and the onion graphs represent trees by sequences of nested boxes. Note that, in tree–maps, the size of the individual rectangles is significant. For example, if the tree represents a file system hierarchy, this size may be proportional to the size of the respective file.

- The *hyperbolic layout* of graphs provides a distorted view of a tree (similar to the use of Fish-eye lenses on traditional tree layouts).

2.3 Navigation and Interaction Techniques

Here it follows the main navigation and interaction techniques:

- *Focus+context*: This approach is defined as a viewing approach that provides users with a detailed view of a small focus area and a global view of the overall context, that is, it provides a set of techniques that allow the user to focus on some detail *without* losing the context. Typical focus+context techniques are Fisheye Views, Polyfocal Display, Bifocal Lens, Perspective Wall, Hyperbolic Browser, etc.

        - Fisheye views imitate the well–known fisheye lens effect, by enlarging an area of interest, and showing other portions of the image with successively less detail. The distortion created by the fisheye view is the consequence of the form of the function, which has a faster increment around 0 (hence affecting the nodes around the focus), with the increment slowing down when closing up.

- *Zooming+filtering*: This approach is defined as a viewing approach that works by reducing the amount of context in the display. The reduction is done by filtering the information in the form of selecting a subset of the data along a range of numerical values of one or more dimensions. The typical zooming (along with Pan) filtering techniques are Starfield Display, Tree-Maps, Pad, Pad++ or the more recent version called Piccolo, etc.

        - Zooming can take on two forms. *Geometric zooming* simply provides a blow up of the graph content. *Semantic zooming* means that the information content changes and more details are shown when approaching a particular area of the graph.

- *Incremental exploration*: This approach is defined as a viewing approach that displays only a small portion of the full hierarchy incrementally following the user's exploration of information space. Thus, these techniques are able to handle huge data sets where it is impossible to display the entire hierarchy on the screen at a time.

3 Similarity and Representation of Similarity

Sticking to the question that gives the title to this report, and before analysing the available tools, there are two issues regarding the use of the distance-similarity metaphor that should be addressed to take full advantage of the potential the metaphor has to offer for exploration of complex spaces: encoding and decoding similarity distance.

3.1 Encoding similarity distance

“Everything is related to everything else, but near things are more related than distant things” (first law of Geography - Tobler).
Starting from this statement, it is possible to affirm that the similarity distance metaphor has to map how related are data content into a chosen distance measure, so that similar data items are placed closer to one another in an multi-dimensional attribute space than less similar ones (Fabrikant and Buttenfield, 2001).

Therefore the problems that may rise are related to the distance measure that is going to be chosen.
According to the different measure, a metric space can or cannot be defined, thus different distance measures will strongly affect the choice of the representation (i.e. the visualization) of the similarity

3.2 Decoding similarity distance

The question is how to represent the similarity measure in a not misleading manner: the risk lies in the perceptual and cognitive level, since viewers may attach meaning to metric distances visible in the display although non-metric proximity underlies the data are not related (e.g. SOM).

Here it follows the main techniques to represent the similarity:

- SOM (Kohonen, T. 1995)

The SOM is an algorithm used to visualize and interpret large high-dimensional data sets. Typical applications are visualization of process states or financial results by representing the central dependencies within the data on the map
The map consists of a regular grid of processing units, "neurons". A model of some multidimensional observation, eventually a vector consisting of features, is associated with each unit. The map attempts to represent all the available observations with optimal accuracy using a restricted set of models. At the same time the models become ordered on the grid so that similar models are close to each other and dissimilar models far from each other

Pros:

- It reduces a multidimensional concept space to a 2D/3D space
- It reveals clusters of related concepts, overall patterns within a discourse

Cons:

- no interaction with users (in the chosen example, but other development could solve the problem)
- users might relate close data that are not related (e.g. elements on close “hills” could be not related!)

- Concept Graphs: They help to reveal the more general themes in a discourse. The user can expand each concept to drill down to ever more specific terms (Gahegan 2003)

  Pros:

  - It reveals more general themes in a discourse

  Cons:

  - It works well for small graphs

- MDS - Multidimensional scaling aims to detect meaningful underlying dimensions that allow the researcher to explain observed similarities or dissimilarities (distances) between the investigated objects (any kind of similarity or dissimilarity matrix, in addition to correlation matrices). [http://www.statsoft.com/...]

- Graphs using FDP (a lot of algorithms have been implemented – they only differ in the computational time... e.g. Kamada and Kawai, 1989 , Fruchterman and Reingold 91, ... Other algorithms similar to FDP is Spring Embedder (Eades 1984))

  Basic idea of FDP: system of forces similar to subatomic particles and celestial bodies.
  In order to lay out a graph to replace the vertices by steel rings and replace each edge with a spring to form a mechanical system (Figure 1). The vertices are placed in some initial layout and let go so that the spring forces on the rings move the system to a minimal energy state. An important deviation from the physical reality is the application of the forces: repulsive forces are calculated between every pair of vertices, but attractive forces are calculated only between neighbours (this reduces the time complexity).
  The repulsive force is $-k^2/d$, the attractive force is $-d^2/k$, k represents the optimal distance between vertices and in a diagram of the forces it is the point where the 2 forces cancel each out.

  Pros of FDP (and similar techniques):
  - works well in practice for small graphs with regular structure
  - relatively simple to implement (many tools implement them)
  - extendible to 3D
  - often able to detect and display symmetries

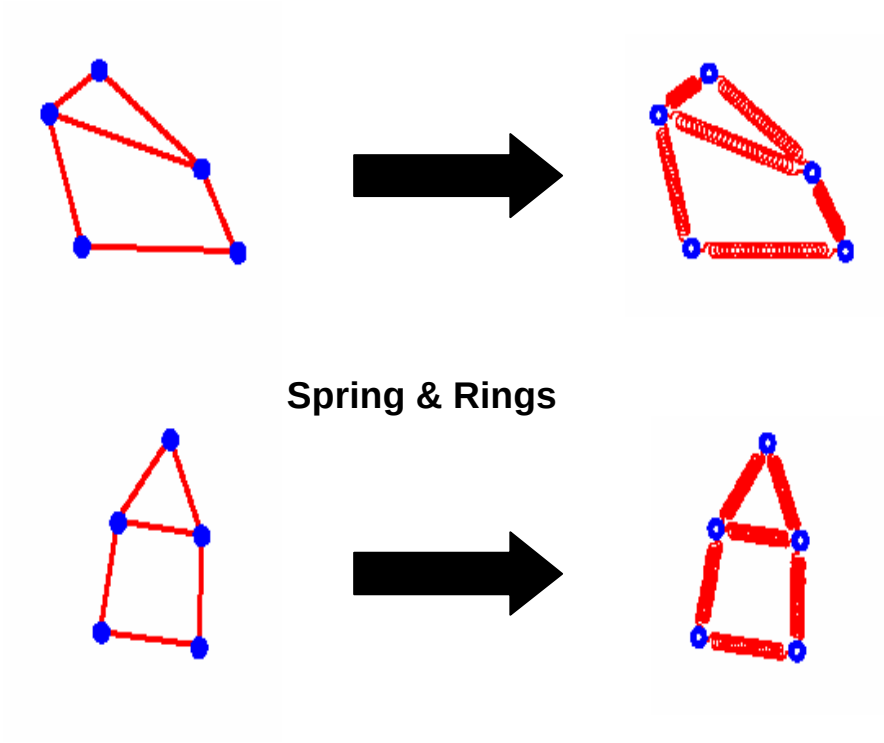

Figure 1: Initial (A) and final (B) configuration

Cons of FDP (and similar techniques):

- slow running time
- few theoretical results on the quality of the drawings produced
- difficult to extend to orthogonal and polyline drawings
- limited constraint satisfaction capability

Semantic Treemaps (Feng Y. & Börner, 2002) use the traditional Treemap (Shneiderman, B. 1992) space filling technique to represent the tree structure of the best partitions by alternatively dividing rectangles in horizontal and vertical direction, resulting in a set of nested rectangles showing the layers of nodes in the tree. FDP algorithm is used to layout the web pages in each Treemap rectangle according to their semantic similarities: by associating the semantic distances between data items with the spring force coefficients similar items will be drawn close to each other and vice versa.

4 Related works & Tools

In the semantics world, the main works are related to the ontology exploration.

- SEWASIE (Catarci 2004) An ontology based visual tool for query formulation support - The intelligence of the interface is driven by an ontology describing the domain of the data in the information system. The final purpose of the tool is to generate a conjunctive query ready to be executed by some evaluation engine associated to the information system
- SWETO (SWETO): a public use testbed with OWL schema. It uses TouchGraph visualization (with zoom, rotate and locality functionalities)
- GODE (GODE 2004) Graphical Ontology Designer Environment proposes 3 areas for the visual query (main concept, background concept, temporal area); a background system can lay under GODE (e.g. WordNet, Sesame, RDF repository) and return semantic or lexical related concepts. The tool mainly aims to create/edit the ontology which is constructed starting from the inserted text. Then a semantic analyser (OntoExtract) elaborates the text, and a graph based on Spring Embedder algorithm displays the result.
- ISWIVE (ISWIVE 2004) Integrated Semantic Web Interactive Visualization Environment aims at visualizing the information of Topic Maps and RDF. It proposes 3 areas for the visual query: semantic query (=to search the SW resources by subject, predicate or object), dual panel viewer to display RDF graph (via a multi-scale force directed algorithm) and/or Topic Tree (via an extended hv-tree drawing algorithm), local viewer panel (that gives a detailed view of the relationships of the selected nodes and surrounding resources).
- WIDE (WIDE 2004) aims to solve a common problem during information search, that is, different user groups do not have the same backgrounds and use different terminologies to talk about the same things. The components are a user interface, a meta level and a content level. The proposed visualization is similar to TgViz with the usual interaction functionalities.
- SWAP.it (Seeling and Becks, 2004) Semantic Web analysis portal for intelligent text analysis. It integrates a Document Map to show interdocument similarity (a

visual text mining tool based on DocMINER), a domain ontology in form of tree that serves as a workspace for metadata-database navigation, and some analysis functionalities (e.g. fulltext, search, statistics, list of URL and metadata,…)

There exist several tools to represent similarities. Here it follows some of main ones:

- IVC The Information Visualization CyberInfrastructure: it provides an unified architecture in which diverse data analysis, modelling and visualization algorithms can be plugged in and run (IVC 2004)

  Main features: completely open-source, it allows to integrate different programming languages (e.g., Java, Perl, C, C++), math packages and graphic already implemented (e.g. Latent Semantic Analysis, Topics Model, Pathfinder Network Scaling, Multidimensional Scaling, Clustering, Parallel Coordinates, Spring Embedding Algorithm, Radial Tree, Hyperbolic Tree, Fisheye Table, …)

- GeoVISTA (GeoVISTA Studio, 2000): GeoVISTA *Studio* is an open software development environment designed for geospatial data. *Studio* is a programming-free environment that allows users to quickly build applications for geo-computation and geographic visualization.

  Main features: open software, modularly designed interface that allows the integration of various forms of geographic data to be analysed and displayed in a dynamic environment

- Prefuse (Prefuse, 2004) is a user interface toolkit for building highly interactive visualizations of structured and unstructured data. This includes any form of data that can be represented as a set of entities (or nodes) possibly connected by any number of relations (or edges). Using this toolkit, developers can create responsive, animated graphical interfaces for visualizing, exploring, and manipulating these various forms of data.

Main features: open-source, it includes hierarchies (organization charts, taxonomies, file systems), networks (computer networks, social networks, web site linkage) and even non-connected collections of data (timelines, scatterplots). It is written in Java, using the Java2D graphics library and is designed to integrate with any application written using the Java Swing.

- The InfoVis Toolkit (InfoVis, 2004) is a Interactive Graphics Toolkit written in Java to ease the development of Information Visualization applications and components.

  Main features: Extensible, implements nine types of visualization: Scatter Plots, Time Series, Parallel Coordinates and Matrices for tables; Node-Link diagrams, Icicle trees and Treemaps for trees; Adjacency Matrices and Node-Link diagrams for graphs.

- Piccolo.Java (Piccolo , 2004) is a toolkit that supports the development of 2D structured graphics programs, in general, and Zoomable User Interfaces (ZUIs - a ZUI is a new kind of interface that presents a huge canvas of information on a traditional computer display by letting the user smoothly zoom in, to get more detailed information, and zoom out for an overview). These types of interfaces include the concept of semantic zoom by which the zoomed representation of an object is not simply the scaling of its geometric shape, but the shape or representation that is most suitable at that scale to convey the meaning of the object and ease the understanding of its nature. For example, at a certain scale level an object can be just a dot, at another it can be depicted as a labelled box while still at another it can be a rectangle with little characters.

  Main features: written in 100% java, it is based on the Java2D API, semantic zoom, hierarchical structure of graphical objects and cameras, great visualisation flexibility.

- JUNG (Jung, 2005) the Java Universal Network/Graph Framework is a software library that provides a common and extendible language for the modelling, analysis, and visualization of data that can be represented as a graph or network.

  Main features: open-source , written in Java, it includes implementations of algorithms from graph theory, data mining, and social network analysis, such as routines for clustering, decomposition, optimisation, random graph generation, statistical analysis, and calculation of network distances, flows, and importance measures (centrality, PageRank, HITS, etc.).
  It also provides a visualization framework that makes it easy to construct tools for the interactive exploration of network data.

- Graphviz - Graph Visualization Software

  Main features: open source, several main graph layout programs

- Pajek (Pajek 2003) Program for Large Network Analysis,

  Main features: free, graph visualizations, several algorithms (FDP, Spring Embedder, etc.)

5 Conclusion and Discussion

This technical report investigated the problems related to the visualization of similarity, the main rules and navigational techniques to be taken into account the connection between the choice of a certain distance measure and its representation; and finally a brief overview of the related works and the available tools to visualize the similarity.

6 Bibliography

Bertin, J. Graphics and Graphic Information Processing, Berlin: Walter de Gruyter & Co., 1981

Tobler, W. R. (1970). A Computer Movie Simulating Urban Growth in the Detroit Region. E*conomic Geography,* vol. 46, no. 2: 234-240.

Fabrikant, S. I. and Buttenfield, B. P. (2001). Formalizing Semantic Spaces For Information Access. *Annals of the Association of American Geographers,* vol. 91, no. 2: 263-280.

Kohonen, T. (1995), Self-Organizing Maps, Berlin: Springer-Verlag, 1995.
William Pike, Mark Gahegan: Constructing Semantically Scalable Cognitive Spaces. COSIT 2003: 332-348

MDS http://www.statsoft.com/textbook/stmulsca.html

Kamada, T. and Kawai, S. (1989). An Algorithm for Drawing General Undirected Graphs. *Information Processing Letters,* vol. 31, no. 1: 7-15.

Fruchterman, T.M.J & Reingold, E.M. (1991), Graph drawing by force directed placement. Software: Practice and Experience, 21(11), 1991.

P. Eades, 'A heuristic for graph drawing', *Congressus Nutnerantiunt,* 42, 149–160 (1984).

Feng Y. & Börner K. Using semantic treemaps to categorize and visualize bookmark files. In Proceeding of SPIE - Visualization and Data Analysis. Volume 4665, January 2002,San Jose, CA, USA, pp. 218-227.

Shneiderman, B. (1992). Tree Visualization with Treemaps: 2D Space-Filling Approach. ACM Transactions on Graphics 11, 1 (Jan. 1992), 92 - 99.

Tiziana Catarci, Paolo Dongilli, Tania Di Mascio, Enrico Franconi, Giuseppe Santucci, Sergio Tessaris: An Ontology Based Visual Tool for Query Formulation Support. ECAI 2004: 308-312.

SWETO http://lsdis.cs.uga.edu/Project/Semdis/sweto (Sheth, Avant)

Leendert W. M. Wienhofen. "Using Graphically Represented Ontologies for Searching Content on the Semantic Web," *iv*, vol. 00, no. , pp. 801-806, Eighth 2004.

Ing-Xiang Chen, Chun-Lin Fan, Pang-Hsiang Lo, Li-Chia Kuo, Cheng-Zen Yang: ISWIVE: An Integrated Semantic Web Interactive Visualization Environment. AINA 2005: 701-706.

Dirk Burmeister, Stephan Grimm, Jörg Haist, A Semantic Approach for User Depending Information Visualization, 8th International Conference on Information Visualisation, IV 2004, 14-16 July 2004, London, UK. pp. 302-307.

Seeling C., Becks A.: Exploiting Metadata for Ontology-Based Visual Exploration of Weakly Structured Text Documents. Proceedings of the 7th International Conference on Information Visualisation (IV03), London, U.K., IEEE Press, ISBN 0-7695-1988-1, July 2003, pp.652-657.

IVC http://iv.slis.indiana.edu/sw/papers/ivc-framework.pdf

GeoVISTA Studio: A Geocomputational Workbench (PDF) Gahegan, M. et al. (2000) "GeoVISTA Studio: A Geocomputational Workbench", The Proceedings of the 5th International Conference on GeoComputation, August.

Prefuse: a toolkit for interactive information visualization. Jeffrey Heer, Stuart K. Card, and James A. Landay. In *CHI 2005, Human Factors in Computing Systems*, 2005. http://prefuse.sourceforge.net/index.html

The InfoVis Toolkit, Jean-Daniel Fekete, in Proceedings of the 10th IEEE Symposium on Information Visualization (InfoVis'04), IEEE Press, 2004, pp. 167-174.

Piccolo Toolkit Design for Interactive Structured Graphics Bederson, B. B., Grosjean, J., & Meyer, J. (2004). IEEE Transactions on Software Engineering, 30 (8), pp. 535-546.

JUNG http://jung.sourceforge.net/index.html

Graphviz http://www.graphviz.org/

Chapter about Pajek: V. Batagelj, A. Mrvar: Pajek - Analysis and Visualization of Large Networks. in Jünger, M., Mutzel, P., (Eds.) Graph Drawing Software. Springer, Berlin 2003. p. 77-103