Sky-Metaphor Visualisation for Self-Organising Maps

Khalid Latif and Rudolf Mayer
Institute of Software Technology & Interactive Systems
Vienna University of Technology
Favoritenstrasse 9-11/188, A-1040 Vienna, Austria
{klatif, mayer}@ifs.tuwien.ac.at

Abstract: Self-Organising Maps are utilised in many data mining and knowledge management applications. Although various visualisations have been proposed for SOM, these techniques lack in distinguishing between the items mapped to the same unit. Here we present a novel technique for the visualisation of Self-Organising Maps that displays inputs not in the centre of the map units, but shifts them towards the closest neighbours, the degree of the movement depending on the similarity to the neighbours. The night-sky visualisation facilitates better understanding of the underlying data. We report results from applying our method on two synthetic and a real-life data set.

Key Words: Self-organising Map, Visualisation, Night-sky
Category: H.3.3, H.5.2, I.2.6

1 Introduction

The Self-Organising Map (SOM) is a prominent tool for data mining and knowledge management. Part of its popularity can be attributed to the various visualisation methods which summarise the characteristics of the data set and help the user in understanding and analysing the underlying structure in the input data. The location of the input objects on the map allows the user to quickly identify similar and different objects.

However, the mapping of an input onto single map units is coarse and inaccurate to some extent. Depending on the resolution of the map, i.e. the number of units, inputs mapped onto the same unit might bear significant differences, which are not easy to transmit or visualise. Therefore, we propose a novel visualisation technique that takes into account not only the best-matching unit of an input object, but also the input's distances to the neighbouring units. As a result, the objects will not be placed at the centre of the map unit, but drift towards some of the neighbouring units. This helps the user on one hand to more easily distinguish between the items in the same unit, and on the other hand to grasp the similarities between data objects across unit boundaries.

This paper is organised as follows: Section 2 gives a brief introduction to the SOM algorithm, a survey of the most relevant SOM visualisations, and other systems using a night-sky metaphor for visualisation. Section 3 introduces our new visualisation technique, and Section 4 demonstrates its applicability. In Section 5 we conclude our findings and provide an outlook on future work.

2 Related Work

In this section we give an overview of the Self-Organising Map (SOM), its various visualisations, and other systems using the sky metaphor for data visualisation.

2.1 Self-Organising Map

The SOM [3] is a unsupervised neural network model that provides a mapping from a high-dimensional input space to a lower, often two-dimensional, output space. An important property of this mapping is that it is topology preserving – elements which are located close to each other in the input space will also be closely located in the output space. The generated map can help the user in getting a quick overview of the patterns in the input space.

The input space consists of any kind of data collection that can be represented in the numerical form - e.g. a vector space bag-of-words representation of text documents, features extracted from audio or images, or any other kind of numerical data. The output space is in many applications organised as a rectangular grid of units, a representation that is easily understandable for users due to its analogy to 2-D maps. Each of the units on the map is assigned a *weight vector* $\mathbf{m}_i$, which is of the same dimensionality as the vectors $\mathbf{x}_i$ in the input space. During the training process, the vectors $\mathbf{x}_i$ are presented to the Self-Organising Map, and the unit with the most similar weight vector to this input vector, the *best-matching unit*, is determined. The weight vector of this unit, and, to a lesser extent, of the neighbouring units, are adapted towards the input vector, i.e. their distance in the input space is reduced – the output space 'folds' as closely as possible into the input space. After the training is finished, the inputs are mapped onto their ultimate best-matching unit. Some units might accumulate a lot of inputs, while others, probably located between clusters, may be left empty. Further details on the SOM training process can be found in [3].

The SOM provides clustering of the data without explicitly assigning data items to the clusters or identifying cluster boundaries as opposed to, for example, the k-Means method. To allow an easier interpretation of the cluster structures and correlations in the content, visualisation techniques highlighting cluster boundaries and cumulations in the map are needed.

2.2 Self-Organising Map Visualisations

SOM visualisations can utilise the output space as a platform [13], where quantitative information is most commonly depicted as colour values or markers of different sizes. More advanced approaches use e.g. the analogy to geography [9].

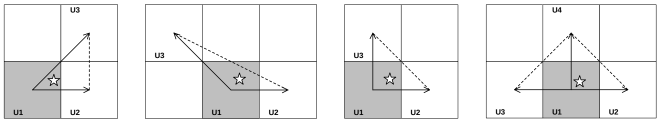
Weight-vector based techniques, rely solely on the weight vectors. Among them, Component Planes are projections of single dimensions of the weight vectors $\mathbf{m}_i$. By plotting all dimensions, all information about the weight-vectors is revealed. With increasing dimensionality, however, it becomes more difficult to perceive important information such as clustering structure and underlying dependencies. The unified distance matrix (U-Matrix [12]) is a visualisation technique that shows the local cluster boundaries by depicting pair-wise distances of neighbouring weight vectors. It is the most common method associated with Self-Organising Maps and has been extended in numerous ways. The Gradient Field [6] has some similarities with the U-Matrix, but applies smoothing over a broader neighbourhood and uses a different style of representation. It plots a vector field on top of the lattice where each arrow points to its closest cluster centre. This can be used to contrast different groups of Component Planes [7].

A second category of visualisation techniques take into account the **data distribution**. The most simple ones are hit histograms, which show how many data samples are mapped to a unit, and labelling techniques, which plot the names and categories, provided they are available, of data samples onto the map lattice. More sophisticated methods include Smoothed Data Histograms [5], which show the clustering structure by mapping each data sample to a number of map units, or graph-based methods [8], showing connections for units that are close to each other in the feature space. The P-Matrix [11] depicts the number of samples that lie within a sphere of a certain radius, namely a quartile of the pair-wise distances of the data vectors, around the weight vectors. Our newly proposed method falls into this category, and has certain similarities to hit histograms.

Emergent SOMs [10] work with a very high number of map units, i.e. provide a very high resolution. While a higher resolution can achieve to some extent similar results as our visualisation, namely distinguishing on a more detailed level between similar input objects, it does not completely solve the issue – input objects might still be mapped onto the same unit, and the map does not give hints on the similarity to inputs mapped on neighbouring units. Moreover, the ideal size of the SOM would need to be known, determined through experimental trials. Increasing the map size also has implications on performance.

2.3 Sky-metaphor Data Visualisation Techniques

The IN-SPIRE [14] tool builds on galaxy visualisation by making use of the metaphor of stars in the night sky. Each star represents an individual document, and clusters around centre points represent themes. The galaxy metaphor is also investigated in a prior work [2] to visualise document similarity. The night sky metaphor is also used in InfoSky [1] with a different approach. InfoSky is contingent on the assumption that documents are already organised in a hierarchy of collections. The collections are rendered as Voronoi cells, and hierarchically related collections are placed alongside each other. In contrast, the SOM is used for organizing the collections, and sky metaphor as a novel visualization of the SOM. documents



(a) (b) (c) (d)

Figure 1: Neighbouring forces on an element

3 Sky-metaphor Visualisation for Self-Organising Maps

In this section, we present our Sky-metaphor visualisation technique. The map uses a black background to resemble the night sky. Individual objects from the input space are represented as stars, which together with other similar objects may form star clusters. This effect can be enhanced by using a Smoothed Data Histograms [5] visualisation on top of the background, resembling galaxies. Units that do not contain any inputs remain black and will resemble dark nebulae.

Different interaction strategies such as zooming & panning, individual document and area selection are not specific to the sky metaphor, but are supported by our SOM toolkit [4].

Star Clusters

Traditionally, the SOM algorithm assigns input objects only to a discrete map unit. We however want to reveal more details about the relations between the objects that are mapped onto the same unit, and also the similarities of the objects to other objects in neighbouring units. Therefore, we propose to place the input objects not in the centre of a unit cell, but spread them across the cell.

Neighbourhood Forces

We calculate the exact location of an input $\mathbf{x}$ which is mapped onto its best-matching unit U. Our assumption is that the location of the input $\mathbf{x}$ in unit U is driven by the position of the next closest units, with the distance of $\mathbf{x}$ to these units acting as a pull force to the input. The pull force ($\mathbf{F}$) of a unit is inversely proportional to the distance of the input from the unit and is relative to the distance of the input to its best matching unit:

$$\mathbf{F}_i \propto \frac{d(x, U_1)}{d(x, U_i)} \quad \text{for} \quad i > 1 \tag{1}$$

where d denotes the metric measuring the distance from the input to the weight vector of a unit.

As the second-best matching unit is nearer to the input than third-best-matching unit, its pull force is higher in magnitude. For this reason, the displacement effect is insignificant for farther units. In most of the cases the second and third closest units, denoted as U_2 and U_3, are sufficient for calculating the displacement of the input $\mathbf{x_i}$ from the centre of the unit U. Their pull forces make up a virtual triangle, as illustrated in Figure 1. There is one rare exception to this assumption, namely in cases where both the second and third best matching unit are found to be on one axis with U. This implies that the input $\mathbf{x_i}$ would drift along only one dimension as a triangle effect can not be realized. In those cases, the fourth closest unit U_4 is taken into account (c.f. Figure 1(d)).

The x and y coordinates of the exact position $\mathbf{p}$ of input $\mathbf{x}$ on unit U can then be defined as:

$$\mathbf{p}_{<x,y>} = \left\langle \lambda * \sum_{i=2}^{k} \mathbf{F}_i * \frac{1}{U_{i<x>} - U_{1<x>}}, \lambda * \sum_{i=2}^{k} \mathbf{F}_i * \frac{1}{U_{i<y>} - U_{1<y>}} \right\rangle$$

where k is an index over the two or three nearer units U_2, U_3 and U_4 respectively, i.e. $k = 3$ or $k = 4$. A grid-constant λ is used to reconcile the displacement according to the display co-ordinates and is initially set to approximately a quarter of the unit's pixel size. In some cases two or more inputs may overlap each other too much due to very high similarity. In such a situation we marginally shift the inputs apart by applying a force of repulsion, where overlapping units push each other in opposite direction.

Constellations as Interconnection Trails

In the physical world, entities are usually interconnected either by physical or by semantic means. In the proposed night-sky visualization, the interconnections are realized by exploiting the notion of constellations. Closely related stars form a pattern and highlights the relationship between the inputs, which may otherwise be mapped to different units. We allow both user defined and automatic trails (such as based on meta-data) to illustrate usefulness of constellations by drawing connection lines between the stars.

4 Experiments and Results

For the experiments described in this section, we used two synthetic data sets to demonstrate the visualisation, and one text data set to test its applicability to a large real-life corpus.

Figure 2 shows experiments with the 'chain-link' data set, i.e. two intertwining rings in three-dimensional space. This data-set cannot be projected to two dimensions while preserving the ring-structures, the normal behaviour is for the rings to 'break'. The visualisation resembles the structure of the two rings well, with the points stretching over the cell space in such a way that an almost continuous line is formed. This is very similar to the original data, which also

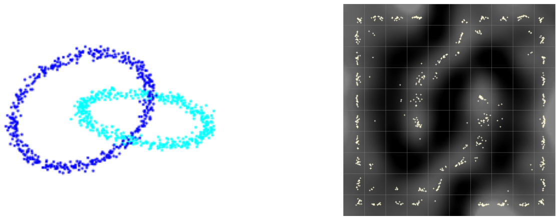


Figure 2: Chain-link data set and a trained map

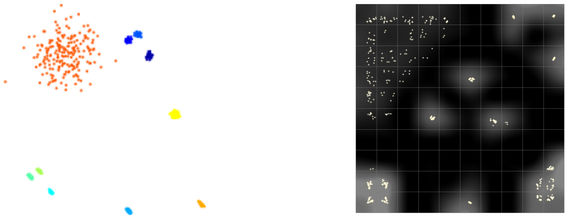


Figure 3: A data set of several different Gaussian clusters and a trained map

doesn't form the ring as a continuous data chain, but rather as several small clusters of data points.

Figure 3 depicts a plot of the two principal components of a ten-dimensional data set, generated using several Gaussian distributions with different centres and kernel widths. By not placing the data items in the centre of the units, the Sky-metaphor visualisation shows the concentration of inputs more effectively and also provides clear cluster boundaries.

The text corpus we used for our last experiment is the *20 newsgroups* data set. It consists of 1000 newsgroup postings for each of its 20 different newsgroups, such as *alt.atheism* and *comp.sys.mac.hardware*. We considered only the subject and the message body, but omitted other header lines. A standard *bag-of-words* indexing approach was used, applying a manually created stop-word list and document frequency threshold to reduce the dimensionality. A $tf \times idf$ weighting scheme was employed to obtain the vector values for the 2896 remaining terms. Finally, we trained a SOM of the size of 50 × 40 units.

Figure 4 depicts the overview of the trained map. Due to spreading the inputs over the SOM cells and the tendency to the inputs being moved towards the cluster centres, these become more compact and dense, while the areas between two clusters become larger – it becomes easier to identify groups of similar inputs.

Figure 5 depicts two sections of the map. The left image in the figure illustrates the concept of constellations: postings that are in relation to each other,

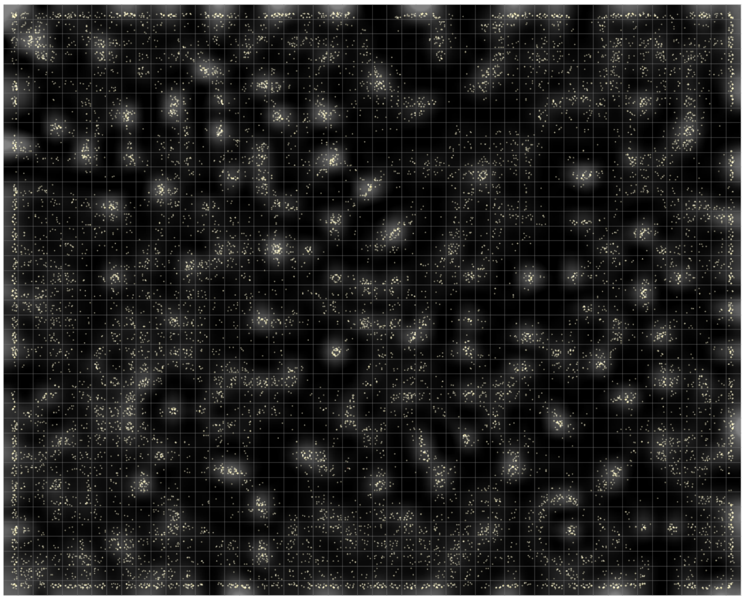


Figure 4: Sky Visualisation of 20 Newsgroups maps – overview

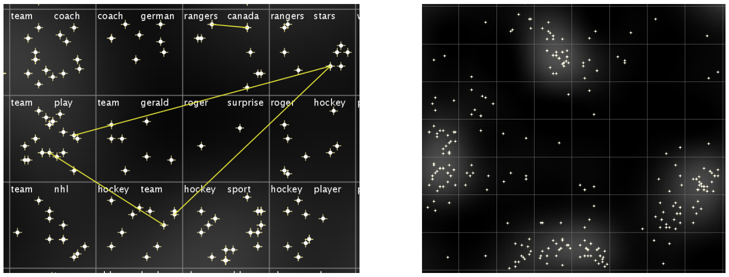


Figure 5: Detailed view of the 20 Newsgroups map

here direct replies to other postings, are linked. Such associative referencing allows instantly recalling other linked items in the data set. The right image shows a detailed view of cluster boundaries between two *sci.med* clusters in the upper-left and upper-middle area, and two *rec.motorcycles* and *rec.autos* clusters located in the lower-middle and right-middle area. Even though there are only few or no empty units between the cluster centres, the inputs on the units between those centres have been placed closer towards the centres, and therefore the cluster boundaries become easily visible.

5 Conclusions

In this paper we presented a novel method for visualising Self-Organising Maps. The night-sky metaphor is used to represent and interactively explore the underlying data set. The relationship of similarity between the inputs was depicted through star clusters and other complex interconnections by constellations. Our experiments with different data sets show that even a large stockpile of data could be turned into very useful knowledge map with effective visualisations.

References

1. Keith Andrews, Wolfgang Kienreich, Vedran Sabol, Jutta Becker, Georg Droschl, Frank Kappe, Michael Granitzer, Peter Auer, and Klaus Tochtermann. The InfoSky visual explorer: Exploiting hierarchical structure and document similarities. *Information Visualization*, 1(3/4):166–181, December 2002.
2. Beth Hetzler, Michelle Harris, Susan Havre, and Paul Whitney. Visualizing the full spectrum of document relationships. In *Proc. of 5th Intl. Conf. on Structures and Relations in Knowledge Organization*, pages 168–175, Lille, France, August 1998.
3. Teuvo Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer, 1995.
4. Robert Neumayer, Michael Dittenbach, and Andreas Rauber. PlaySOM and PocketSOMPlayer: Alternative interfaces to large music collections. In *Proc. of the Sixth Intl. Conf. on Music Information Retrieval*, pages 618–623, London, UK, September 11-15 2005.
5. Elias Pampalk, Andreas Rauber, and Dieter Merkl. Using Smoothed Data Histograms for Cluster Visualization in Self-Organizing Maps. In *Proc. of the Intl. Conf. on Artifical Neural Networks (ICANN'02)*, pages 871–876, Madrid, Spain, August 27-30 2002. Springer.
6. Georg Pölzlbauer, Michael Dittenbach, and Andreas Rauber. A visualization technique for Self-Organizing Maps with vector fields to obtain the cluster structure at desired levels of detail. In *Proc. of the Intl. Joint Conf. on Neural Networks*, pages 1558–1563, Montreal, Canada, July 31 - August 5 2005.
7. Georg Pölzlbauer, Michael Dittenbach, and Andreas Rauber. Advanced visualization of Self-Organizing Maps with vector fields. *Neural Networks*, 19(6-7):911–922, July-August 2006.
8. Georg Pölzlbauer, Andreas Rauber, and Michael Dittenbach. Advanced visualization techniques for Self-Organizing Maps with Graph-Based Methods, May 30 -June 1 2005.
9. André Skupin. A picture from a thousand words. *Computing in Science and Engineering*, 6(5):84–88, Sept.-Oct. 2004.
10. Alfred Ultsch. *Data mining and knowledge discovery with emergent self-organizing feature maps for multivariate time series*, pages 33–45. Elsevier, 1999.
11. Alfred Ultsch. Maps for the visualization of high-dimensional data spaces, 2003.
12. Alfred Ultsch and Hans Peter Siemon. Kohonen's self-organizing feature maps for exploratory data analysis, July 1990.
13. Juha Vesanto. SOM-based data visualization methods. *Intelligent Data Analysis*, 3(2):111–126, 1999.
14. Pak Chung Wong, Beth Hetzler, Christian Posse, Mark Whiting, Susan Havre, Nick Cramer, Anuj Shah, Mudita Singhal, Alan Turner, and Jim Thomas. IN-SPIRE InfoVis 2004 Contest Entry. In *Proc. of the IEEE Symposium on Information Visualization*, pages 216–217, Washington, DC, 2004.