Interactive Metric Learning-based Visual Data Exploration: Application to the Visualization of a Scientific Social Network

Masaharu Yoshioka[1], Masahiko Itoh[23], and Michèle Sebag[4]

[1] Graduate School of Information Science and Technology, Hokkaido University
N14 W9, Kita-ku, Sapporo 060-0814, Japan
[2] Institute of Industrial Science, the University of Tokyo,
4-6-1, Komaba, Meguro-ku, Tokyo, 153-8505, Japan
[3] Social ICT Research Center, National Institute of Information and Communications Technology,
4-2-1, Nukui-Kitamachi, Koganei, Tokyo 184-8795, Japan
[4] TAO, LRI - CNRS
Univ. Paris-Sud bldg 650, Rue Noetzlin, 91190, Gif-sur-Yvette, France

Abstract. Data visualization is a core approach for understanding data specifics and extracting useful information in a simple and intuitive way. Visual data mining proceeds by projecting multidimensional data onto two-dimensional (2D) or three-dimensional (3D) data, e.g., through mathematical optimization and topology preserved in multidimensional scaling (MDS). However, this projection does not necessarily comply with the user's needs, prior knowledge and/or expectations. This paper proposes an interactive visual mining approach, centered on the user's needs and allowing the modification of data visualization by leveraging approaches from metric learning. The paper exemplifies the proposed system, referred to as *Interactive Metric Learning-based Visual Data Exploration* (IMViDE), applied to scientific social network browsing.

1 Introduction

Knowledge discovery from databases, the process of extracting knowledge from data [1], must be focused on the user needs: indeed, the desired knowledge properties (being new and useful) largely depend on the user's prior knowledge and expectations.

Data visualization is a core approach to understanding the data specifics, and extracting useful information in a simple and intuitive way [2], through projecting the multidimensional data in $\mathbb{R}^d$ onto $\mathbb{R}^2$, thus enabling its visual inspection. The quality of the projection thereby governs the quality of the knowledge extracted along data visualization. One of the best known data visualization approaches, Multi-Dimensional Scaling (MDS), proceeds by minimizing the topology loss induced by the projection from $\mathbb{R}^d$ onto $\mathbb{R}^2$ [3] (more in section 2).

However, the MDS projection does not necessarily comply with the users' prior knowledge and/or expectations about the problem domain. For this reason, several approaches have been proposed to interactively modify the MDS projection [3, 4]. In particular, Brown *et al.* [5] proposed to leverage the distance metric learning pioneered by Large Margin Nearest Neighbor (LMNN) [6] in the context of supervised machine learning (section 3). Specifically, LMNN [6] is aimed at the Mahalanobis distance on the data space such that it maximizes the classification accuracy of the k-nearest neighbor process, and shows that this problem reduces to a convex optimization problem[5].

This paper focuses on distance metric learning in the context of multidimensional data visualization for data exploration. The proposed *Interactive Metric Learning-based Visual Data Exploration* (IMViDE) system is an iterative 5-step process, using the standard Euclidean distance on $\mathbb{R}^d$ as initial distance:

1. The data is displayed in $\mathbb{R}^2$ using MDS together with the current distance.
2. The user specifies some distance-related constraints by labeling a few data points; the requirement is that a labeled point should be close to some other points with the same label, and further away from points with different labels.
3. The distance on $\mathbb{R}^d$ is optimized to account for the constraints, based on the ideas from [6].
4. Most importantly, IMViDE provides the user with feedback, displaying the features most relevant/impacted by the metric changes. This feedback allows the user to make sense of the search path and clarify his/her intention about the exploratory data analysis.
5. IMViDE relaunches MDS with the new metric and updates the data visualization. In this visualization result, data points that share the same label form a cluster as a result of distance metric learning, and the user can find data points that are as close to the cluster as similar ones in the context of this exploratory analysis. If the user is not satisfied with the visualization results, he/she goes back to step 2 to revise the visualization result that fits his/her intention.

This paper is organized as follows. Section 2 briefly reviews related works of data visualization and distance metric learning. For the sake of completeness, distance metric learning is described in section 3. An overview of the IMViDE system is detailed in section 4. IMViDE is exemplified in section 5, considering the visualization of a social network. The paper concludes with a discussion and some perspectives for further research.

2 Related Works

Data visualization techniques are used to represent characteristic information in the target data to the user's intuitive ways [2]. In particular, for multidimensional

[5] Note that the classification accuracy maximization can also be tackled by feature selection, that is, a combinatorial optimization problem.

data, there are several methods of projecting multidimensional data in $\mathbb{R}^d$ onto $\mathbb{R}^2$ such as MDS [3], PCA [7], SOM [8], GTM [9], and t-SNE [10]. The results of visualization using such methods sometimes differ from the user's intention; the interactive visualization is, therefore, required for modifying visualization results based on the user's intention and intuition.

Some studies provided functions for interactively changing parameters for dimension reduction and visualization. iPCA [11] and XGvis [12] enable users to interactively adjust dials or sliders to modify influential parameters in PCA or MDS respectively. However, it is difficult for users with no mathematical knowledge to predict the results caused by varying parameters. They therefore rely on trial-and-error to obtain desirable responses.

InterAxis [13] and Dust & Magnet [14] enable users to intuitively define and modify axes by dragging data points on the side of the x or y axes or attributes on a scatter plot respectively. iVisClassifier [15], using semisupervised Linear Discriminant Analysis (LDA), allows users to interactively label data and recompute clusters and projections. However, they did not provide functions for directly defining relationships between data points such as closeness and remoteness.

Another approach is using the concept of distance metric learning [16]. Distance metric learning is a framework for calculating appropriate distance metrics to classify labeled data more accurately. Most of these algorithms are formalized as supervised Mahalanobis distance learning. There are two main approaches. One is driven by nearest neighbors, such as Neighborhood Components Analysis (NCA) [17] and LMNN[6] and the other covers information-theoretic approaches, such as Information-Theoretic Metric Learning (ITML) [18] and Sparse Distance Metric Learning (SDML) [19].

There are some studies that use distance metric learning for constructing appropriate distance metric that fits the users' prior knowledge [20–23, 5]. LAMP [21] provided a multidimensional projection technique enabling users to build local transformations from some control points directly specified by users. Mizuno *et al.* presented an approach for manipulating arrangements of the local features and global categories of images by projecting the overall feature space onto a two-dimensional (2D) screen space [22]. V2PIs [20] and its extension [23], and Dis-function [5] allowed users to move data points in a 2D projected space to update the weight of a weighted-MDS model and the distance function of MDS respectively. Their method is similar to our method in that they allow users to explicitly reflect their intention by directly manipulating data points. However, their purpose of interaction on the scatter plot is mostly to provide a global optimum projection or distance functions from labeled or sampled data points based on the user's prior knowledge. By contrast, our purpose is exploring the user's classification standards based on distance metric learning through interactive manipulation of data points, and constructing an information retrieval system enabling users to retrieve related and/or similar information from their interesting data points. In addition, because our system would like to learn new distance metric by using a few numbers of labeled data, it is difficult to use an information-theoretic approach for our problem.

In the information retrieval research, there are several methods for providing feedback information to show the characteristics of a document that attract users' intention. For example, DualNavi [24] provides characteristic terms from selected retrieved results to modify the original retrieved query. Scatter/gather [25] is an interactive document clustering technique that is widely used in several domains [26, 27]. In this framework, the system conducts document clustering in the original document collection and provides information about the cluster by using topical words of the cluster (scatter). From the clustering results, the user selects one or more clusters that attract his/her attention and make a new document collection for further analysis (gather). The user iterates the scatter/gather process to find out the useful information. Although the framework of the system is different from our approach; i.e., IR starts with a query and our approaches start with selecting interesting data, it is helpful to show such feedback information to understand the characteristics of the results.

3 Distance Metric Learning for kNN Classification

The k-nearest neighbors (kNN) method, is one of the oldest and simplest methods for pattern classification that associates an instance with the majority class of its k nearest neighbors. The performance of this method critically depends on the distance metric used to identify nearest neighbors. In a supervised machine learning context, optimizing the distance metric based on labeled examples in such a way that it maximizes the kNN performance, comes naturally.

3.1 LMNN Classification

Weinberger *et al.* [6] formalized the problem of metric learning in terms of optimizing a linear change in representation, such that the Euclidean distance in the new representation yields optimal kNN performances as follows. Let us first introduce some notations:

- Let the training set $\mathcal{E}$ be defined as:

$$\mathcal{E} = \{(\boldsymbol{x_i}, y_i), \boldsymbol{x_i} \in \mathbb{R}^d, y_i \in \{-1, 1\}, i = 1 \ldots n\}$$

- For each pair (i, j) with $1 \leq i, j \leq n$, let $y_{i,j}$ be 1 iff $y_i = y_j$ and 0 otherwise.
- Let $j \rightsquigarrow i$ denote that $\boldsymbol{x_j}$ is a target neighbor of $\boldsymbol{x_i}$ (that is, $\boldsymbol{x_j}$ is among the k nearest neighbors of $\boldsymbol{x_i}$ with same label as $\boldsymbol{x_i}$, $y_j = y_i$).
- Finally, let $[z]_+ = max(z, 0)$ denote the standard hinge loss of z.

With these notations, the goal is to find a linear change in representation on $\mathbb{R}^d$, with L, a $d \times d$ matrix, such that the distance D_L on $\mathbb{R}^d$ is defined as:

$$D_L(\boldsymbol{x}, \boldsymbol{x'}) = ||L(\boldsymbol{x} - \boldsymbol{x'})||, \tag{1}$$

optimize two cost functions, respectively noted as $\epsilon_{pull}(L)$ and $\epsilon_{push}(L)$. The cost function $\epsilon_{pull}(L)$, to be minimized, is the sum of the distances between any $\boldsymbol{x_i}$ and its target neighbors:

$$\epsilon_{pull}(L) = \sum_{j \rightsquigarrow i} D_L(\boldsymbol{x_i}, \boldsymbol{x_j})^2$$

The cost function $\epsilon_{push}(L)$, to be minimized, measures the excess distance between a point $\boldsymbol{x_i}$ and its target neighbor $\boldsymbol{x_j}$, *compared to another neighbor* $\boldsymbol{x_l}$ which belongs to another class than $\boldsymbol{x_i}$:

$$\epsilon_{push}(L) = \sum_{i,j \rightsquigarrow i} \sum_{l} (1 - y_{il})[1 + D_L(\boldsymbol{x_i}, \boldsymbol{x_j})^2 - D_L(\boldsymbol{x_i}, \boldsymbol{x_l}))^2]_+$$

Finally, with α the weight parameter balancing the two criteria, the optimization problem is defined as:

$$\text{Find } L^* = \arg\max_{L} \left(\alpha \epsilon_{pull}(L) + (1 - \alpha)\epsilon_{push}(L)\right) \tag{2}$$

.

For the sake of convex optimization, one rather seeks $M = L^t L$ with L^t the transpose matrix of L, such that

$$D_L(\boldsymbol{x}, \boldsymbol{x'})^2 = ||L(\boldsymbol{x} - \boldsymbol{x'})||^2 = (\boldsymbol{x} - \boldsymbol{x'})^t L^t L(\boldsymbol{x} - \boldsymbol{x'})$$

For simplicity of notation, D_L is denoted D_M in the following.

This change in representation enables to reformulate Pb (2) as a semidefinite programming problem (SDP):

$$\begin{aligned} \text{Minimize } \; & (\alpha)\epsilon_{pull}(M) + (1 - \alpha)\epsilon_{push}(M) && (3) \\ \text{s.t. } \; & (\boldsymbol{x_i} - \boldsymbol{x_j})^t M(\boldsymbol{x_i}.\boldsymbol{x_j}) \leq 1 - \xi_{ijl} && (4) \\ & \xi_{ijl} \geq 0 && (5) \\ & M \succeq 0 && (6) \end{aligned}$$

.

The constraint $M \succeq 0$ indicates that matrix M is required to be positive and semidefinite. While general-purpose solvers can solve this SDP, such solvers tend to scale poorly when the number of constraints increases. Therefore, they propose to use a special- purpose solver based on a combination of subgradient descent in both matrices L and M.

3.2 Efficient computation

The gradient computation can be done most efficiently by careful book-keeping from one iteration to the next. Let M_t denote the current solution at step t. As a simplifying notation, let matrix C_{ij} be defined as:

$$C_{ij} = (\boldsymbol{x_i} - \boldsymbol{x_j})(\boldsymbol{x_i} - \boldsymbol{x_j})^t$$

The loss function in Eq. 6 is rewritten as:

$$\epsilon(M_t) = (1-\mu)\sum_{j \rightsquigarrow i} tr(M_t C_{ij}) + \mu \sum_{i,j \rightsquigarrow i} \sum_l (1 - y_{il}) \left[1 + tr(M_t C_{ij}) - tr(M_t C_{il})\right]_+ \quad (7)$$

with $tr(A)$ denoting the trace of matrix A.

Note that Eq. 7 is piecewise linear with respect to M_t. Let N_t be the set of triplets (i, j, l), such that the indices (i, j, l) satisfy

$$1 + tr(M_t C_{ij}) - tr(M_t C_{il} > 0$$

(they trigger the hinge loss in Eq. 7). With this definition, the gradient G_t of $\epsilon(M_t)$ can be written as:

$$\begin{aligned} G_t &= \frac{\partial \epsilon(M_t)}{\partial M_t} \\ &= (1-\mu)\sum_{j \rightsquigarrow i} C_{ij} + \mu \sum_{i,j \rightsquigarrow i} \sum_l (1 - y_{il})(C_{ij} - C_{il}) \end{aligned} \quad (8)$$

.

4 Overview of Interactive Metric Learning-based Visual Data Exploration

MDS is a popular method for projecting a set of data points $\boldsymbol{x_1} \ldots \boldsymbol{x_m}$ (not necessarily in a metric space) onto $\mathbb{R}^2$ based on the matrix of their dissimilarities or distances. Formally, to each $\boldsymbol{x_i}$ MDS associates a projection $z_i \in \mathbb{R}^2$, in such a way that the Euclidean distance $d(z_i, z_j)$ in $\mathbb{R}^2$ approximates the dissimilarity between $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$. This projection in the 2D plane enables visual inspection of the data. However, the initial dissimilarities and the associated visualization might not reflect the user's prior knowledge and desires, hindering the visual data mining process.

4.1 The IMViDE algorithm

The proposed *Interactive Metric Learning-based Visual Data Exploration* (IMViDE) system aims at addressing this drawback, by allowing the user to interactively modify the MDS visualization results. In the following, it is assumed that the data points are real-value vectors ($\boldsymbol{x_1} \in \mathbb{R}^d$; further research is concerned with extending the proposed approach to the general case.

The user interacts with IMViDE by specifying that some data points should or should not be close to each other in the representation 2D space. The IMViDE algorithm is a four-step process:

1. The MDS projection is applied on the basis of the current distance matrix; the resulting projection of the data points is displayed in the 2D plane.

2. The user interacts with IMViDE by selecting pairs of points as similar or dissimilar.
3. The metric is revised to account for the above constraints (pairs of similar or dissimilar data points).

4. A new distance matrix is computed according to the new distance and the process is iterated (go to step 1).

4.2 Distance Metric Learning for MDS

The inspiration for the IMViDE algorithm was taken from the metric learning approach presented in section 3 to find a linear transformation of the initial feature space, complying with the user-specified constraints.

In contrast to the standard kNN context, however, the number of neighborhood-related constraints is low as they result from the interaction with the user. We, therefore, adapt the optimization objective (Eq. 6). The original pull cost function "penalizes" small distances between every data point and close data points with different labels. In our case, as few points are "labeled", we penalize the small distances between every labeled $\boldsymbol{x_i}$ and all $\boldsymbol{x_j}$ that do not have the same label as $\boldsymbol{x_j}$.

Finally, the optimization criterion used to find the Mahalanobis distance complying with the current constraints, where y_{il} is set to 1 if $\boldsymbol{x_i}$ and $\boldsymbol{x_j}$ share the same label, and 0 in all other cases is the following:

$$\epsilon_{pull}(M_t) = \sum_{i,j \rightsquigarrow l} D_{M_t}(\boldsymbol{x_i}, \boldsymbol{x_j}) \tag{9}$$

$$\epsilon'_{push}(M_t) = \sum_{i,j \rightsquigarrow l} \sum_{l} (1 - y_{ij}) y_{il} \tag{10}$$

$$[1 + D_{M_t}(\boldsymbol{x_i}, \boldsymbol{x_l}) - D_{M_t}(\boldsymbol{x_i}, \boldsymbol{x_l})]_+ \tag{11}$$

$$\epsilon'(M_t) = (1 - \mu)\epsilon'_{push}(M_t) + \mu \epsilon_{push}(M_t) \tag{12}$$

.

The gradient of $\epsilon'(M_t)$, noted G'_t, reads:

$$\begin{aligned} G'_t &= \frac{\partial \epsilon'(M_t)}{M_t} \\ &= (1 - \mu) \sum_{i,j \rightsquigarrow l} C_{ij} + \mu \sum_{i,j \rightsquigarrow l} \sum_{l} (C_{ij} - C_{il}). \end{aligned} \tag{13}$$

The minimization of the cost function is handled using the same gradient algorithm as in [6].

4.3 Functionalities of IMViDE

We implemented the IMViDE system based on the discussion above. This system is made of two components:

- The first component, *Visualization*, takes charge of the visualization of the members of the social network. The current distance matrix is used as input of MDS to yield a projection of the members on the 2D plane. The metric is initially the Euclidean metric ($M_0 = Id$) on the representation space.
- The second component, *Interaction and Metric Learning*, takes charge of the following operations:
  - The user selects similar and dissimilar pair(s) of nodes for distance metric learning. The system adds the same label for nodes of similar pair(s) and adds a different label for dissimilar pair(s).
  - An important functionality is to provide some feedback to the user, indicating what (the system thinks) are the main goals of his/her search.

The detailed procedure of updating the distance metric is as follows.

1. Selection of similar and dissimilar pair(s) for metric learning.
   From the MDS visualization results, the user selects nodes that belong to the same group for adding same labels. When the user selects nodes without a label, the system generates a new label for the nodes. When the user selects nodes with labels, all labels are merged as one label and add merged label are added to all related nodes. For example, at first the user selects n_1, n_2, n_3 for adding labels, these three nodes have the label l_1. Next, the user selects n_4, n_5 for adding labels, these two nodes have the label l_2. When the user selects n_1, n_4, n_6 for adding labels, labels l_1 and l_2 are merged as l_1 and all 6 nodes $n_1, ..., n_6$ are labeled as l_1.
2. Metric learning by using similar and dissimilar pair(s) information.
   Based on the information about labeled nodes, the system refines M for minimizing the cost function by using a linear programming problem with a positive semidefinite constraint [6]. In this process, the gradient G'_t (eq. 13) is used to refine M stepwise. As the total minimization process requires high computational cost and may change the distance among nodes drastically, there are several cases for which the visualization results change drastically and result interpretation is inappropriate. Therefore, the system produces an intermediate result of the stepwise refinement process for the MDS visualization.
3. Updating MDS results by using the Mahalanobis distance metric.
   Based on the stepwise refinement result of M, the system updates the MDS visualization result. To keep the continuity of the visualization results, the MDS visualization result is updated by using the SMACOF (scaling by majoring a convex function) algorithm [28] and a previous visualization result is used as the initial input. As a result, the position of all nodes slightly moves based on this update process. The user can continue this minimization process in step 2 to see the effect of the distance metric learning (e.g.,

some unlabeled nodes move in the same direction and some unlabeled nodes do not move). In addition, the user can also go back to step 1 for modifying labels.

The IMViDE system produces feedback information to the users by using the difference between the most important words for each class before and after the interaction. Formally, let $\boldsymbol{c_i}$ and $\boldsymbol{c'_i}$ respectively denote the center of mass of the i-th class in the initial representation (respectively in the current representation):

$$\boldsymbol{c_i} = \sum_{j \in Cl_i} \boldsymbol{x_j}/|Cl_i| \tag{14}$$

$$\boldsymbol{c'_i} = M_t c_i \tag{15}$$

The contribution of the initial j-th dimension in the current representation, denoted as $\boldsymbol{ra_j}$ is defined, where $\boldsymbol{r_j}$ is a vector whose j-th element is 1 and 0 otherwise:

$$\boldsymbol{ra_j} = M_t r_j \tag{16}$$

Finally, noting c_{ij}, c'_{ij}, ra_{ij} the j-th coordinates of the c_i, c'_i, ra_i vectors, and $P(w|z_j)$ the probability of the word w for class j as computed by probabilistic latent semantic analysis (PLSA), we compute the characteristic score vectors I_{wi} and I'_{wi}, indicating the relevance of every term for class i, with respect to the initial and current metric:

$$I_{wi} = \sum_{j=0}^{T} c_{ij} P(w|z_j) \tag{17}$$

$$I'_{wi} = \sum_{j=0}^{T} \sum_{k=0}^{T} c'_{ij} ra_{jk} P(w|z_k) \tag{18}$$

The top-ℓ words ($\ell = 10$ in the experiments) relevant to each class before and after the interaction are displayed, giving the user feedback about the most important aspects of the i-th class, as interpreted through the metric learning and PLS preprocessing.

5 Visualization of a Scientific Social Network

The proposed IMViDE algorithm was empirically assessed on the visualization of a social network. For the sake of reproducibility and easy assessment, we used the social network of scientists involved in the European Network of Excellence PASCAL (*Pattern Analysis, Statistical Modelling and Computational Learning*, 2003-2013), where each scientist member of the network is described by his/her papers.

5.1 Pre-processing

We used the data made public for the Pascal Visualization Challenge, available at: http://analytics.ijs.si/˜blazf/pvc/data.html. The goal was to visualize the relationship between the authors based on the similarity among the contents of their paper (as opposed to the similarity induced by the coauthorship and citations).

The available data were preprocessed as follows:

1. Construction of the paper database with abstract and author information.
   Noun, adjective, adverb, and verb are selected and normalized by using TreeTagger [6] as candidates for the index keywords.
2. Selection of keywords.
   Keywords with the minimum document frequency and listed in stop list (e.g., be, do, one, etc.) were removed from the index keyword lists. We used a minimum document frequency of 1 in this experiment.
3. Construction of feature vectors.
   For each author, index keyword information on all his/her papers were collected and coded as his/her feature vector. In this vector, all index keywords correspond to one dimension in the feature vector space, and the value for that dimensions are calculated by TF · IDF.
4. Construction of reduced dimension feature vector by PLSA [29].
   To avoid the effect of the sparseness of the keyword feature vectors, we applied PLSA for constructed feature vectors for dimension reduction.

From the Pascal challenge data, we constructed feature vectors for 313 authors with 2986 index keywords and the feature vectors were reduced into 40 dimensions by using PLSA.

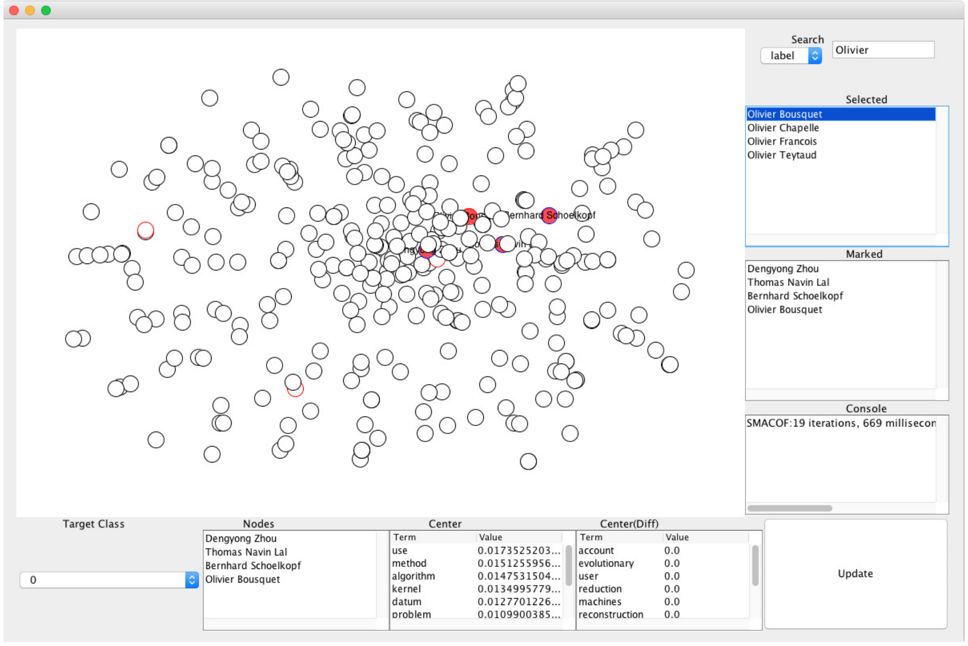
5.2 Experiments

Figure 1 shows an example of initial MDS visualization results. In this case, the user selected four coauthors of a paper; Bernhard Schoelkopf (who is the author with the largest number of papers in this database), Thomas Navin Lal, Dengyong Zhou, Olivier Bousquet who belong to the same group (red points in Figure 1). These authors have multiple articles in the database and Table 1 shows the number of subject category articles for each researcher[7].

Figure 2 shows a zoomed image of an MDS visualization based on a result of distance metric learning, showing that the system found a new distance metric in which the three authors are closer to each other. Figure 3 shows the characteristic terms in Figure 1 and 2, with Center and Center(diff) the list of top ranked keywords respectively based on the I'_{wi} value and $I'_{wi} - I_{wi}$.

[6] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/

[7] The total differs from the sum of the categories as each article may have more than one subject category.



Search: label Olivier

Selected:
Olivier Bousquet
Olivier Chapelle
Olivier Francois
Olivier Teytaud

Marked:
Dengyong Zhou
Thomas Navin Lal
Bernhard Schoelkopf
Olivier Bousquet

Console:

```
SMACOF:19 iterations, 669 millisecon
```

Target Class: 0

Nodes:
Dengyong Zhou
Thomas Navin Lal
Bernhard Schoelkopf
Olivier Bousquet

Center

Term	Value
use	0.0173525203...
method	0.0151255956...
algorithm	0.0147531504...
kernel	0.0134995779...
datum	0.0127701226...
problem	0.0109900385...

Center(Diff)

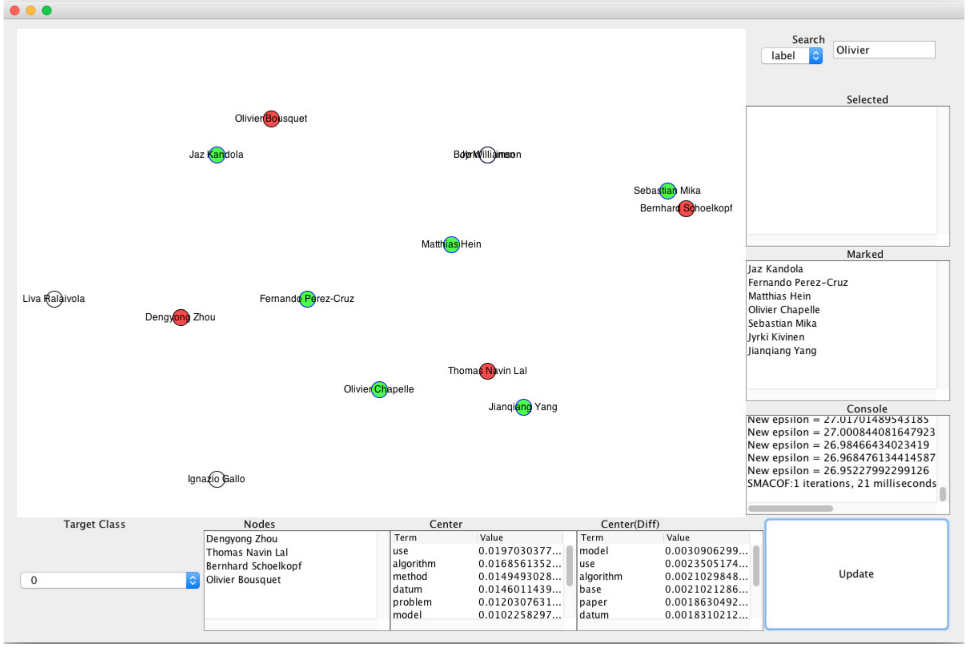
Term	Value
account	0.0
evolutionary	0.0
user	0.0
reduction	0.0
machines	0.0
reconstruction	0.0

Fig. 1. Initial MDS visualization results

Table 1. Number of articles for categorized topics

Name	BC	CS	IT	LO	MV	TA	total
Bernhard Schoelkopf	4	15	1	21	8	19	34
Thomas Navin Lal	4	1	0	3	0	3	7
Dengyong Zhou	0	3	1	5	0	8	8
Olivier Bousquet	0	11	0	14	0	17	19

BC: brain–computer interface, CS: computational, information-theoretic learning with statistics, IT: information retrieval and textual information access, LO: learning/statistics and optimization, MV: machine vision, TA: theory and algorithms



Jaz Kandola
Fernando Perez-Cruz
Matthias Hein
Olivier Chapelle
Sebastian Mika
Jyrki Kivinen
Jianqiang Yang

```
New epsilon = 27.01701489543185
New epsilon = 27.000844081647923
New epsilon = 26.98466434023419
New epsilon = 26.968476134414587
New epsilon = 26.95227992299126
SMACOF:1 iterations, 21 milliseconds
```

Dengyong Zhou
Thomas Navin Lal
Bernhard Schoelkopf
Olivier Bousquet

Term	Value
use	0.0197030377...
algorithm	0.0168561352...
method	0.0149493028...
datum	0.0146011439...
problem	0.0120307631...
model	0.0102258297...

Term	Value
model	0.0030906299...
use	0.0023505174...
algorithm	0.0021029848...
base	0.0021021286...
paper	0.0018630492...
datum	0.0018310212...

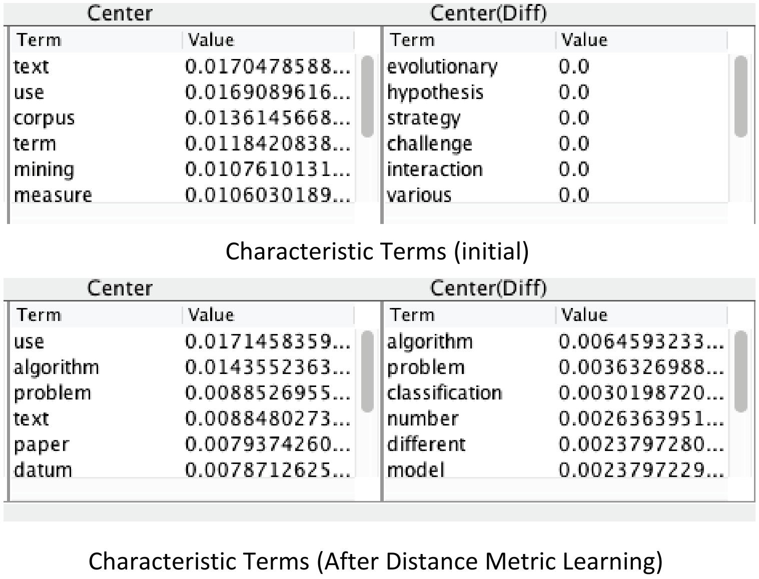
Fig. 2. Zoomed MDS visualization results based on DML

At the initial stage, as $I'_{wi} = I_{wi}$, all values for Center(diff) equal zero. Initially, keywords related to these authors common topic categories "CS", "LO", and "TA" (e.g., "algorithm" and "method") had a higher value. After distance metric learning those values are increased that represents keywords related to those categories are important features for calculating similarities. Followings is the list of top-ranked keywords based on Center:I'_{wi} value and Center(diff): $I'_{wi} - I_{wi}$ value were as follows.

Center:I'_{wi} use, algorithm, problem, text, paper, datum, approach, base, model

Center(diff):$I'_{wi} - I_{wi}$ algorithm, problem, classification, number, different, model, result, statistical, evolutionary

In addition, from Figure 2, we could determine some other authors (depicted as green points), who have a research topic that is also related to "CS", the application of machine learning algorithms to text data. Although those authors were somewhat close to Bernhard Schoelkopf, Thomas Navin Lal, Dengyong Zhou, or Olivier Bousquet in the initial visualization, there were many other researchers around them (Figure 4). This result shows how the metric learning state could help retrieving researchers with similar research interest during the interaction with the user.



Center		Center(Diff)	
Term	Value	Term	Value
text	0.0170478588...	evolutionary	0.0
use	0.0169089616...	hypothesis	0.0
corpus	0.0136145668...	strategy	0.0
term	0.0118420838...	challenge	0.0
mining	0.0107610131...	interaction	0.0
measure	0.0106030189...	various	0.0

Characteristic Terms (initial)

Center		Center(Diff)	
Term	Value	Term	Value
use	0.0171458359...	algorithm	0.0064593233...
algorithm	0.0143552363...	problem	0.0036326988...
problem	0.0088526955...	classification	0.0030198720...
text	0.0088480273...	number	0.0026363951...
paper	0.0079374260...	different	0.0023797280...
datum	0.0078712625...	model	0.0023797229...

Characteristic Terms (After Distance Metric Learning)

Fig. 3. List of characteristics terms in Figure 1 and 2

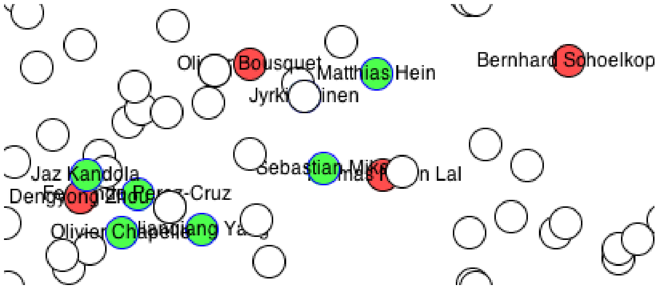


Fig. 4. Original positions of related authors in the initial MDS visualization

5.3 Discussion

There are several issues to be discussed in this system. One is the scalability issue. Because of the high computational complexity of LMNN based on SDP, LMNN does not scale well for a large data set [30]. The random sampling algorithm proposed by Wu *et al.* [30] may be a possible solution. Another solution is similar to the concept of scatter/gather [25]. At the initial stage a limited number of nodes (e.g., selection of researchers based on the number of articles) are used for initial visualization and distance metric learning. When the user is satisfied with the result at a certain level, the system selects nodes close to the labeled nodes and expands nodes by adding nodes that are close to these selected nodes by using the Mahalanobis distance. This approach is also good for improving the readability of the data presented on the screen, as it is quite difficult to read through the label of nodes more than thousands. Another issue is related to the technique for projecting the multidimensional data in $\mathbb{R}^d$ onto $\mathbb{R}^2$. There are several other techniques for this projection. First, we will investigate how the nonlinear t-distributed stochastic neighbor embedding (t-SNE) [10] compares to MDS. In addition to these further research directions, we also plan to extend our framework with multi-user functionalities, when several users interact with a large map.

6 Conclusions

This paper shows how metric learning can be embedded in an interactive visual data mining system, providing an intuitive and easy control of the visualization functionality. A main contribution of the approach is the provision of feedback, indicating the “angles” of users’ queries in terms of the dimensions (here, terms) most relevant to the new display. We also discuss the future research directions of this approach.

Acknowledgement

We would like to thank Prof. Jean-Daniel Fekete for many suggestions and discussion about this work. The first author was partially supported by JSPS KAKENHI Grant Number 25280035.

References

1. Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P.: Advances in knowledge discovery and data mining. American Association for Artificial Intelligence, Menlo Park, CA, USA (1996) 1–34
2. Keim, D.: Information visualization and visual data mining. IEEE Transactions on Visualization and Computer Graphics **8**(1) (Jan 2002) 1–8
3. Buja, A., Swayne, D.F., Littman, M.L., Dean, N., Hofmann, H., Chen, L.: Data visualization with multidimensional scaling. Journal of Computational and Graphical Statistics **17**(2) (June 2008) 444–472

4. Broekens, J., Cocx, T.: Object-centered interactive multi-dimensional scaling: Ask the expert. In: Proceedings of the Eighteenth Belgium-Netherlands Conference on Artificial Intelligence (BNAIC2006). (2006) 59–66
5. Brown, E.T., Liu, J., Brodley, C.E., Chang, R.: Dis-function: Learning distance functions interactively. In: Visual Analytics Science and Technology (VAST), 2012 IEEE Conference on, IEEE (2012) 83–92
6. Kilian Q. Weinberger, L.K.S.: Distance metric learning for large margin nearest neighbor classification. Journal of Machine Learning Research **10** (2009) 207–244
7. Jolliffe, I.T.: Principal Component Analysis. Springer (2002)
8. Kohonen, T.: Self-Organizing Maps. Springer (2001)
9. Bishop, C.M., Svensén, M., Williams, C.K.I.: GTM: the generative topographic mapping. Neural Computation **10**(1) (1998) 215–234
10. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of Machine Learning Research **9**(2579-2605) (2008) 85
11. Jeong, D.H., Ziemkiewicz, C., Fisher, B.D., Ribarsky, W., Chang, R.: ipca: An interactive system for pca-based visual analytics. Comput. Graph. Forum **28**(3) (2009) 767–774
12. BUJA, A., SWAYNE, D.F., Michael L. LITTMAN and, N.D.: Xgvis: Interactive data visualization with multidimensional scaling. Journal of Computational and Graphical Statistics (2001)
13. Kim, H., Choo, J., Park, H., Endert, A.: Interaxis: Steering scatterplot axes via observation-level interaction. IEEE Trans. Vis. Comput. Graph. **22**(1) (2016) 131–140
14. Yi, J.S., Melton, R., Stasko, J.T., Jacko, J.A.: Dust & magnet: multivariate information visualization using a magnet metaphor. Information Visualization **4**(3) (2005) 239–256
15. Choo, J., Lee, H., Kihm, J., Park, H.: ivisclassifier: An interactive visual analytics system for classification based on supervised dimension reduction. In: Proceedings of the IEEE Conference on Visual Analytics Science and Technology, IEEE VAST 2010, Salt Lake City, Utah, USA, 24-29 October 2010, part of VisWeek 2010. (2010) 27–34
16. Bellet, A., Habrard, A., Sebban, M.: A survey on metric learning for feature vectors and structured data. CoRR **abs/1306.6709** (2013)
17. Goldberger, J., Hinton, G.E., Roweis, S.T., Salakhutdinov, R.: Neighbourhood components analysis. In: Advances in neural information processing systems. (2004) 513–520
18. Davis, J.V., Kulis, B., Jain, P., Sra, S., Dhillon, I.S.: Information-theoretic metric learning. In: Proceedings of the 24th international conference on Machine learning, ACM (2007) 209–216
19. Qi, G.J., Tang, J., Zha, Z.J., Chua, T.S., Zhang, H.J.: An efficient sparse metric learning in high-dimensional space via l1-penalized log-determinant regularization. In: Proceedings of the 26th Annual International Conference on Machine Learning. ICML '09, New York, NY, USA, ACM (2009) 841–848
20. Leman, S.C., House, L.L., Maiti, D., Endert, A., North, C.: Visual to parametric interaction (v2pi). PLOS (3 2013)
21. Joia, P., Coimbra, D.B., Cuminato, J.A., Paulovich, F.V., Nonato, L.G.: Local affine multidimensional projection. IEEE Trans. Vis. Comput. Graph. **17**(12) (2011) 2563–2571
22. Mizuno, K., Wu, H., Takahashi, S.: Manipulating bilevel feature space for category-aware image exploration. In: IEEE Pacific Visualization Symposium, PacificVis 2014, Yokohama, Japan, March 4-7, 2014. (2014) 217–224

23. Hu, X., Bradel, L., Maiti, D., House, L., North, C., Leman, S.: Semantics of directly manipulating spatializations. IEEE Trans. Vis. Comput. Graph. **19**(12) (2013) 2052–2059
24. Takano, A., Niwa, Y., Nishioka, S., Hisamitsu, T., Iwayama, M., Imaichi, O.: Associative information access using dualnavi. In: Proceedings of the Sixth Natural Language Processing Pacific Rim Symposium. (2001) 771–772
25. Cutting, D.R., Pedersen, J.O., Karger, D., Tukey, J.W.: Scatter/gather: A cluster-based approach to browsing large document collections. In: Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. (1992) 318–329
26. Gong, X., Ke, W., Khare, R.: Studying scatter/gather browsing for web search. Proceedings of the American Society for Information Science and Technology **49**(1) (2012) 1–4
27. Zhang, Y., Broussard, R., Ke, W., Gong, X.: Evaluation of a scatter/gather interface for supporting distinct health information search tasks. Journal of the Association for Information Science and Technology **65**(5) (2014) 1028–1041
28. Leeuw, J.D., Mair, P.: Multidimensional scaling using majorization: Smacof in r. Journal of statistical software **31**(3) (2009)
29. Hofmann, T.: Probabilistic latent semantic indexing. In: SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, New York, NY, USA, ACM (1999) 50–57
30. Wu, K., Zheng, Z.: Fast lmnn algorithm through random sampling. In: 2015 IEEE International Conference on Data Mining Workshop (ICDMW). (Nov 2015) 871–876