

Exemplar-based Visualization of Large Document Corpus

Yanhua Chen, *Student Member, IEEE*, Lijun Wang, Ming Dong, *Member, IEEE*, and Jing Hua, *Member, IEEE*

Abstract—With the rapid growth of the World Wide Web and electronic information services, text corpus is becoming available on-line at an incredible rate. By displaying text data in a logical layout (e.g., color graphs), text visualization presents a direct way to observe the documents as well as understand the relationship between them. In this paper, we propose a novel technique, Exemplar-based Visualization (EV), to visualize an extremely large text corpus. Capitalizing on recent advances in matrix approximation and decomposition, EV presents a probabilistic multidimensional projection model in the low-rank text subspace with a sound objective function. The probability of each document proportion to the topics is obtained through iterative optimization and embedded to a low dimensional space using parameter embedding. By selecting the representative exemplars, we obtain a compact approximation of the data. This makes the visualization highly efficient and flexible. In addition, the selected exemplars neatly summarize the entire data set and greatly reduce the cognitive overload in the visualization, leading to an easier interpretation of large text corpus. Empirically, we demonstrate the superior performance of EV through extensive experiments performed on the publicly available text data sets.

Index Terms—Exemplar, large-scale document visualization, multidimensional projection.

1 INTRODUCTION

With the rapid growth of the World Wide Web and electronic information services, text corpus is becoming available on-line at an incredible rate. No one has time to read everything, yet in many applications we often have to make critical decisions based on our understanding of large document collections. For example, when a physician prescribes a specific drug, he frequently needs to identify and understand a comprehensive body of published literature describing an association between the drug of interest and an adverse event of interest. Thus, text mining, a technique of deriving high-quality knowledge from text, has recently drawn great attention in the research community. Research topics in text mining include, but not limited to, language identification, document clustering, summarization, text indexing and visualization. In particular, text visualization refers to the technology that displays text data or mining results in a logical layout (e.g., color graphs) so that one can view and analyze documents easily and intuitively. It presents a direct way to observe the documents as well as understand the relationship between them. In addition, visualization allows people to explore the inside logic of the model and offers users a chance to interact with the mining model so that questions can be answered.

In general, it is convenient to transform document collections into a data matrix [5], where the columns represent documents and the row vectors denote keyword counting after pre-processing. Thus, textual data sets have a very high dimensionality. A common way of visualizing text corpus is to map the raw data matrix into a d -dimensional space with $d = 1, 2, 3$ by employing dimensionality reduction techniques. The objective is to preserve in the projected space the distance relationships among the documents in their original space. Depending on the choice of mapping functions, both linear (e.g., principle component analysis (PCA) [13]) and nonlinear (e.g., ISOMAP [24]) dimensionality reduction techniques have been proposed in the literature. Facing the ever-increasing amount of available documents, a major challenge of text visualization is to develop scalable approaches that are able to process tens of thousands of documents. First, from

a computational point of view, large text corpus significantly raises the bar on the efficiency of an algorithm. For a collection of more than ten thousand documents, typical data projection methods, such as PCA, will fail to run due to insufficient memory. Second, since all documents are shown at once in the resulting space, overlaps of highly related documents are inevitable. Hierarchical clustering-based methods can partially solve the memory problem and produce a tree structure for document exploration. However, these algorithms run extremely slow. More important, they are not mathematically rigorous due to lacking a well defined objective function. Finally, knowledge or information is usually sparsely encoded in document collections. Thus, main topics of a text corpus are more accurately described by a probabilistic model [10]. That is, a document is modeled as a mixture of topics, and a topic is modeled based on the probabilities of words.

In the paper, we propose an Exemplar-based approach to Visualize (EV) extremely large text corpus. Capitalizing on recent advances in matrix approximation and decomposition, our method provides a means to visualize tens of thousands of documents with high accuracy (in retaining neighbor relations), high efficiency (in computation), and high flexibility (through the use of exemplars). Specifically, we first compute a representative text data subspace \mathbf{C} and a low-rank approximation $\tilde{\mathbf{X}}$ by applying the low-rank matrix approximation method. Next, documents are clustered through the matrix decomposition: $\tilde{\mathbf{X}} = \mathbf{C}\mathbf{W}\mathbf{G}^T$, where \mathbf{W} is the weight matrix, and \mathbf{G} is the cluster indicator matrix. To reduce the clutter in the visualization, the exemplars in each cluster are first visualized through Parameter Embedding (PE) [11], providing an overview of the distribution of the entire document collection. When desired, on the clicking of an exemplar, documents in the associated cluster or in a user-selected neighborhood are shown to provide further details. In addition, hierarchical data exploration can also be implemented by recursively applying EV in an area of interest.

In summary, a novel method is proposed here to visualize large document data sets in the low-rank subspace. From a theoretical perspective, EV presents a probabilistic multidimensional projection model with a sound objective function. Based on the rigorous derivation, the final visualization is obtained through iterative optimization. By selecting the representative rows and columns, EV obtains a compact approximation of the text data. This makes the visualization efficient and flexible. In addition, the selected exemplars neatly summarize the document collection and greatly reduce the cognitive overload in the visualization, leading to an easier interpretation of the text mining results. Through extensive experiments performed on the publicly available text data sets, we demonstrate the superior performance of EV when compared with existing techniques. The remainder of the paper is organized as follows. We first review related work in Section 2. Then, we present our algorithm in Section 3. In Section 4, we provide

• Yanhua Chen, Lijun Wang, Ming Dong are with Machine Vision and Pattern Recognition Lab, Department of Computer Science, Wayne State University, Detroit, MI48202, E-mail: {chenyanh, ljwang, mdong}@wayne.edu.

• Jing Hua is with Graphics and Imaging Lab, Department of Computer Science, Wayne State University, Detroit, MI 48202, E-mail: jinghua@wayne.edu.

Manuscript received 31 March 2009; accepted 27 July 2009; posted online 11 October 2009; mailed on 5 October 2009.

For information on obtaining reprints of this article, please send email to: tvcg@computer.org.

thorough experimental evaluation. Finally, we conclude in Section 5.

2 RELATED WORK

Visualization enables us to browse intuitively through huge amounts of data and thus provides a very powerful tool for expanding the human ability to comprehend high dimensional data. A number of different techniques [21, 3, 5] were proposed in the literature for visualizing a large data set, among which multidimensional projection is the most popular one. In document visualization, let $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\} \in \mathbb{R}^{m \times n}$ be a word-document matrix where columns represent the documents and rows denote the words appearing in them. In other words, the documents are treated as vectors with word frequency as their features. Multidimensional projection is to find the embedding of documents $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_n\} \in \mathbb{R}^{d \times n}$ in the visualization space, usually $d = \{1, 2, 3\}$ and minimize $|\delta(\mathbf{x}_i, \mathbf{x}_j) - D(f(\mathbf{x}_i), f(\mathbf{x}_j))|$, where $\delta(\mathbf{x}_i, \mathbf{x}_j)$ is the original dissimilarity distance and $D(f(\mathbf{x}_i), f(\mathbf{x}_j))$ is the Euclidean distance between the corresponding two points in the projected space, and $f: \mathbf{X} \rightarrow \mathbf{Y}$ is a mapping function [23].

In general, multidimensional projection techniques [13, 4, 24, 20] can be divided into two major categories based on the function f employed: *Linear Projection* methods and *Non-linear Projection* methods. Linear projection creates an orthogonal linear transformation that transforms the data to a new coordinate system such that the new variable is a linear combination of the original variables. Among such techniques, the widely known is PCA [13]. PCA finds a low-dimensional embedding of the data points that best preserves their variance as measured in the high-dimensional input space.

However, many data sets contain essential nonlinear structures that are invisible to PCA. For those cases, non-linear projection methods, using information not contained in the covariance matrix, are more appropriate. Several approaches, such as multidimensional scaling (MDS) [4] and ISOMAP [24], have been proposed for reproducing nonlinear higher-dimensional structures on a lower-dimensional display, and they differ in how the distances are weighted and how the functions are optimized. Classical MDS produces a low-dimensional representation of the objects such that the distances (e.g., the Euclidean distance (L2 norm), the Manhattan distance (L1, absolute norm), and the maximum norm) among the points in the new space reflect the proximities of the data in the original space. MDS is equivalent to PCA when the distance measure is Euclidean. ISOMAP extends metric MDS by incorporating the geodesic distances defined as the sum of edge weights along the shortest path between two nodes in a weighted graph (e.g., computed using Dijkstra’s algorithm). Then, the top d eigenvectors of the geodesic distance matrix are used to represent the coordinates in the new d -dimensional Euclidean space. The most recently developed text visualization systems based on the above traditional projection techniques include Infosky¹ and IN-SPIRE².

Although current multidimensional projection techniques can extract a low-dimensional representation of a document based on the word frequency, most of them take no account of the latent structure in the given data, i.e., topics in the document collection. To this end, Least Square Projection (LSP) [17] first chooses a set of control points using k -medoids method [1] based on the number of topics and then obtains the projection through the least square approximation, in which the data are projected following the geometry defined by the control points. Recently, incorporating probabilistic topic models into analyzing documents has attracted great interest in the research community [12] since it can provide a higher quality (i.e., more meaningful) visualization. In Probabilistic Latent Semantic Analysis (PLSA) [10], a topic is modeled as a probability distribution over words, and documents with similar semantics (i.e., topics) are embedded closely even if they do not share any words. The topic proportions estimated by PLSA can be embedded in the Euclidean space by Parametric Embedding (PE) [11], which employs a set of topic proportions as the input. Consequently, the documents that tend to be associated with the

same topic would be embedded nearby, as would topics that tend to have the similar documents associated with them.

Unfortunately, all the aforementioned methods are inapplicable to visualize an extremely large-scale text corpus. When dealing with tens of thousands of documents, for example, PCA will fail to run due to insufficient memory and the high computational cost of solving the eigen problem. Similarly, PLSA model is also computationally expensive. Actually, all of the above models have a time complexity greater than $\mathcal{O}(n)$. The ever-increasing online document collections present unprecedented challenges for the development of highly scalable methods that can be implemented in a linear polynomial time. Therefore, hierarchical-clustering based visualization methods [9, 16] are proposed to partially solve the memory and computation problem, in which a hierarchical cluster tree is first constructed using a recursive partitioning process, and then the elements of that tree are mapped to the d -dimensional space to create a visual representation. However, these methods are derived intuitively, lacking a mathematically rigorous objective function to minimize f . In addition, all determinations are strictly based on local decisions, and the deterministic nature of the hierarchical techniques prevents reevaluation after points are grouped into a node of tree. Therefore, an incorrect assignment made earlier in the process may not be modified, and the optimal hierarchy has to be found through reconstruction.

In order to achieve high accuracy with low computational cost for visualizing large-scale data sets, we present a novel method, called Exemplar-based Visualization (EV). In the following, we will derive a theoretically sound algorithm for EV and apply it to visualize large document corpus.

3 EXEMPLAR-BASED VISUALIZATION

In this section, we first present the EV model and derive the algorithm in Section 3.1. Then, we give some theoretical results in Section 3.2, including the correctness and convergence of the algorithm, time and space complexity analysis, and advantages of EV when compared with other visualization models.

3.1 Model Formulation and Algorithm

The proposed EV model takes a three-step approach to visualize large-scale text corpus. First, low rank matrix approximation is employed to select the representative subspaces and generate the compact approximation of the word-document matrix $\mathbf{X}_{m \times n}$. Among various matrix approximation methods, near-optimal low-rank approximation has gained increasing popularity in recent years due to its great computational and storage efficiency. The representative ones include Algorithm 844 [2], CUR [18] and CMD [22]. Typically, a near-optimal low-rank approximation algorithm first selects a set of columns \mathbf{C} and a set of rows \mathbf{R} as the left and right matrices of the approximation. Then, the middle matrix \mathbf{U} is computed by minimizing $\|\mathbf{X} - \mathbf{CUR}\|_F^2$. Thus, at the end of the first step, we obtain the low-rank approximation $\tilde{\mathbf{X}} = \mathbf{CUR}$, the representative subspaces \mathbf{C} (data exemplar set) and \mathbf{R} (feature set).

In the second step, we need to obtain the “soft” cluster indicators in the low-rank exemplar subspace, representing the probability of each document proportion to the topics in the topic model [7]. We formulate this task as an optimization problem,

$$\begin{aligned} J &= \min_{\mathbf{W} \geq 0, \mathbf{G} \geq 0} \|\tilde{\mathbf{X}} - \mathbf{CWG}^T\|_F^2 \\ &= \text{Tr}(\tilde{\mathbf{X}}^T \tilde{\mathbf{X}} - \tilde{\mathbf{X}}^T \mathbf{CWG}^T - \mathbf{GW}^T \mathbf{C}^T \tilde{\mathbf{X}} + \mathbf{GW}^T \mathbf{C}^T \mathbf{CWG}^T) \end{aligned} \quad (1)$$

where \mathbf{W} is the weight matrix and \mathbf{G} is the cluster indicator matrix with each element $g_{ih} \in [0, 1]$, indicating the probability distribution over topics for a particular document. In the optimization process, we propose an iterative algorithm to get non-negative \mathbf{W} and \mathbf{G} while fixing arbitrarily signed \mathbf{C} and $\tilde{\mathbf{X}}$. The updating rules are obtained by using the auxiliary functions and the optimization theory:

$$\mathbf{W}_{(i,h)} \leftarrow \mathbf{W}_{(i,h)} \sqrt{\frac{(\mathbf{A}_1^+ \mathbf{G})_{(i,h)} + (\mathbf{A}_3^- \mathbf{W} \mathbf{G}^T \mathbf{G})_{(i,h)}}{(\mathbf{A}_1^- \mathbf{G})_{(i,h)} + (\mathbf{A}_3^+ \mathbf{W} \mathbf{G}^T \mathbf{G})_{(i,h)}}} \quad (2)$$

¹<http://www.infovis-wiki.net/index.php?title=InfoSky>

²<http://in-spire.pnl.gov>

$$\mathbf{G}_{(i,h)} \leftarrow \mathbf{G}_{(i,h)} \sqrt{\frac{(\mathbf{A}_2^+ \mathbf{W})_{(i,h)} + (\mathbf{G} \mathbf{W}^T \mathbf{A}_3^- \mathbf{W})_{(i,h)}}{(\mathbf{A}_2^- \mathbf{W})_{(i,h)} + (\mathbf{G} \mathbf{W}^T \mathbf{A}_3^+ \mathbf{W})_{(i,h)}}} \quad (3)$$

where $\mathbf{A}_1 = \mathbf{C}^T \tilde{\mathbf{X}}$, $\mathbf{A}_2 = \tilde{\mathbf{X}}^T \mathbf{C}$ and $\mathbf{A}_3 = \mathbf{C}^T \mathbf{C}$.

The third step is to use PE [11] to embed documents into a low-dimensional Euclidean space such that the input probabilities $\mathbf{G} = p(L_h | \mathbf{x}_i)$ (where L is the topic label of a document) are approximated as closely as possible by the embedding-space probabilities $p(L_h | \mathbf{y}_i)$. The objective is to minimize the difference between input probabilities and the corresponding embedding-space probabilities using a sum of Kullback-Leibler (KL) divergences for each document: $\sum_{i=1}^n KL(p(L_h | \mathbf{x}_i) || p(L_h | \mathbf{y}_i))$. Minimizing this sum $\sum_{h=1}^z p(L_h | \mathbf{y}_i)$ is equivalent to minimizing the following sum of KL divergences:

$$E(\mathbf{y}_i, \phi_h) = - \sum_{i=1}^n \sum_{h=1}^z p(L_h | \mathbf{x}_i) \log p(L_h | \mathbf{y}_i) \quad (4)$$

The unknown parameters, a set of coordinates of documents \mathbf{y}_i and coordinates of topics ϕ_h in the embedding space, can be obtained with a gradient-based numerical optimization method. The gradients of Equation (4) with respect to \mathbf{y}_i and ϕ_h are:

$$\frac{\partial E}{\partial \mathbf{y}_i} = \sum_{h=1}^z (p(L_h | \mathbf{x}_i) - p(L_h | \mathbf{y}_i)) (\mathbf{y}_i - \phi_h) \quad (5)$$

$$\frac{\partial E}{\partial \phi_h} = \sum_{i=1}^n (p(L_h | \mathbf{x}_i) - p(L_h | \mathbf{y}_i)) (\phi_h - \mathbf{y}_i) \quad (6)$$

Thus, we can find the locally optimal solution for embedding coordinates \mathbf{y}_i for each document given ϕ_h .

The complete EV algorithm is given in Algorithm 1.

Algorithm 1 Exemplar-based Visualization

INPUT: word-document matrix $\mathbf{X} \in R^{m \times n}$, selected number of documents and words $r, c \in \mathbb{Z}^+$ s.t. $1 \leq r \leq m$, $1 \leq c \leq n$, number of topics $z \in \mathbb{Z}^+$ s.t. $1 \leq h \leq z$, and the label set of topics $L_{h=1}^z$

OUTPUT: Visualization of documents $\mathbf{Y} = \{\mathbf{y}_i\} \in R^{d \times n}$ ($1 \leq i \leq n$) in the embedding space

1. Use a near-optimal low-rank approximation method to get $\mathbf{C}_{m \times c}$, $\mathbf{U}_{c \times r}$, $\mathbf{R}_{r \times n}$ and $\tilde{\mathbf{X}}_{m \times n}$;
2. Initialize \mathbf{W} and \mathbf{G} with non-negative values;
3. Iterate by the following updating rules for each i and h until convergence;
 - (a) Let $\mathbf{A}_1 = \mathbf{C}^T \tilde{\mathbf{X}}$, $\mathbf{A}_2 = \tilde{\mathbf{X}}^T \mathbf{C}$ and $\mathbf{A}_3 = \mathbf{C}^T \mathbf{C}$, then split each matrix into the positive and negative parts:

$$\mathbf{A}_q^+ = (|\mathbf{A}_q| + \mathbf{A}_q) / 2; \quad \mathbf{A}_q^- = (|\mathbf{A}_q| - \mathbf{A}_q) / 2;$$

where $q \in \{1, 2, 3\}$;

(b)

$$\mathbf{W}_{(i,h)} \leftarrow \mathbf{W}_{(i,h)} \sqrt{\frac{(\mathbf{A}_1^+ \mathbf{G})_{(i,h)} + (\mathbf{A}_3^- \mathbf{W} \mathbf{G}^T \mathbf{G})_{(i,h)}}{(\mathbf{A}_1^- \mathbf{G})_{(i,h)} + (\mathbf{A}_3^+ \mathbf{W} \mathbf{G}^T \mathbf{G})_{(i,h)}}}$$

$$\mathbf{G}_{(i,h)} \leftarrow \mathbf{G}_{(i,h)} \sqrt{\frac{(\mathbf{A}_2^+ \mathbf{W})_{(i,h)} + (\mathbf{G} \mathbf{W}^T \mathbf{A}_3^- \mathbf{W})_{(i,h)}}{(\mathbf{A}_2^- \mathbf{W})_{(i,h)} + (\mathbf{G} \mathbf{W}^T \mathbf{A}_3^+ \mathbf{W})_{(i,h)}}}$$

4. Normalize cluster indicator $\mathbf{G} = p(L_h | \mathbf{x}_i)$ such that $\sum_{h=1}^z p(L_h | \mathbf{x}_i) = 1$;
 5. Use parameter embedding to obtain the embedding-space coordinates \mathbf{y}_i for each document.
-

3.2 Theoretical Analysis

In this section, we first show that our algorithm is correct and converges under the updating rules given in Equations (2)-(3). In addition, we show the efficiency of EV by analyzing its space and time requirements. Finally, we point out the advantages of EV when compared with other visualization methods.

3.2.1 Correctness and Convergence of EV

The correctness and convergence of the EV algorithm can be stated as the following two propositions.

Proposition 1 (Correctness of EV) *Given the object function of Equation (1), the constrained solution satisfies KKT complementary conditions under the updating rules in Equations (2)-(3).*

Proposition 2 (Convergence of EV) *The object function of Equation (1) is monotonically decreasing under the updating rules in Equations (2)-(3).*

Due to the space limit, we give an outline of the proof of the propositions and omit the details. Motivated by [6], we plan to render the proof based on optimization theory, auxiliary function and several matrix inequalities. First, following the standard theory of constrained optimization, we fix one variable \mathbf{G} and introduce the Lagrangian multipliers λ_1 and λ_2 to minimize the Lagrangian function $L(\mathbf{W}, \mathbf{G}, \lambda_1, \lambda_2) = \|\tilde{\mathbf{X}} - \mathbf{C} \mathbf{W} \mathbf{G}^T\|_F^2 - \text{Tr}(\lambda_1 \mathbf{W}) - \text{Tr}(\lambda_2 \mathbf{G}^T)$. Second, based on the KL complementarity condition, we set the gradient descent of $\frac{\partial L}{\partial \mathbf{W}}$ to be zero while fixing \mathbf{G} . Then, we successively update \mathbf{W} using Equation (2) until J converges to a local minima. Similarly, given \mathbf{W} , we can set $\frac{\partial L}{\partial \mathbf{G}}$ to be zero and update \mathbf{G} using Equation (3) until J converges to a local minima. \mathbf{W} and \mathbf{G} should update alternatively. Third, we construct auxiliary functions to prove that Equation (1) decreases monotonically under the updating rules. An auxiliary function $Z(\mathbf{W}^{t+1}, \mathbf{W}^t)$ should satisfy the two conditions: $Z(\mathbf{W}^{t+1}, \mathbf{W}^t) \geq J(\mathbf{W}^t)$, and $Z(\mathbf{W}^t, \mathbf{W}^t) = J(\mathbf{W}^t)$ for any \mathbf{W}^{t+1} and \mathbf{W}^t . We define $\mathbf{W}^{t+1} = \min_{\mathbf{W}} Z(\mathbf{W}, \mathbf{W}^t)$, then we obtain the following equation $J(\mathbf{W}^t) = Z(\mathbf{W}^t, \mathbf{W}^t) \geq Z(\mathbf{W}^{t+1}, \mathbf{W}^t) \geq J(\mathbf{W}^{t+1})$. Thus, with a proper auxiliary function, $J(\mathbf{W}^t)$ is decreasing monotonically. Similarly, we can also prove $J(\mathbf{G}^t)$ is decreasing monotonically under an appropriate auxiliary function.

3.2.2 Time and Space Complexity

To visualize a large data set, efficiency in both space and speed is essential. In the following, we provide detailed analysis on the time and space complexity of EV. To simplify the analysis, we assume $n = m$ and $r = c$ though they are not necessarily equal in the algorithm.

In Algorithm 1, the near-optimal matrix approximation is very efficient, having time complexity of $\mathcal{O}(nc^2)$. Details are given in [2]. In the decomposition step, even though $\tilde{\mathbf{X}}$ is used in the description of the algorithm, the computation is actually done using the three small matrices, \mathbf{C} , \mathbf{U} and \mathbf{R} . Specifically, we first need to compute \mathbf{A}_1 , \mathbf{A}_2 and \mathbf{A}_3 with the following time,

$$\mathbf{A}_1 : c(n \times c + c^2 + c \times n)$$

$$\mathbf{A}_2 : c(n \times c + c^2 + c \times n)$$

$$\mathbf{A}_3 : c^2 n$$

Then, we need to compute \mathbf{W} and \mathbf{G} in Equations (2) and (3). Assuming that the number of iteration $t = 1$, the time for computing \mathbf{W} and \mathbf{G} are,

$$\mathbf{W} : 2(c^2 z + cz^2 + z^2 n + cnz)$$

$$\mathbf{G} : 2(c^2 z + cz^2 + z^2 n + cnz)$$

Thus, the total time for matrix decomposition is $\mathcal{O}(c^2 n + (c^2 z + z^2 n + cnz))$. In addition, the time complexity of PE is $\mathcal{O}(nz)$. Since $z \ll \min(c, n)$ and $c \ll n$, the overall computational complexity is $\mathcal{O}(n)$.

Regarding the space complexity, EV needs $2cn + c^2$ units to store \mathbf{C} , \mathbf{U} and \mathbf{R} , and needs cz and nz units for \mathbf{W} and \mathbf{G} , respectively. In addition, the temporal storage for computing \mathbf{A}_q and updating \mathbf{W} and \mathbf{G} require $\mathcal{O}(cn)$ units. Since $c \ll n$, the total space used is $\mathcal{O}(n)$.

In summary, both the time and space complexity of EV are linear, and thus it is highly scalable, suitable for visualizing a very large document collection.

3.2.3 Advantages of EV

From a theoretical point of view, EV has the following unique properties for visualizing large-scale text corpus when compared with other visualization methods:

- **Accuracy:** EV is a probabilistic multidimensional projection model with a well-defined objective function. Through iterative optimization, it can preserve the proximity in the high-dimensional input space and thus provide accurate visualization results.
- **Efficiency:** EV has a high computational and spacial efficiency, and thus it is especially useful to visualize large document data. Compared with the time complexity of other visualization approaches, EV has a linear running time. Moreover, EV only needs to compute the non-zero entries of the approximation matrix, which further reduces the computational time for a sparse matrix (e.g., word-document matrix). EV also has the space complexity of $\mathcal{O}(n)$ while other algorithms typically require $\mathcal{O}(n^2)$ storage units.
- **Flexibility:** EV decomposes a word-document matrix into three matrices with the representative data subspace \mathbf{C} , which contains the exemplar documents from the collection. By choosing the subspace dimensions, EV can visualize text corpus with different granularity, effectively reducing the clutter/overlap in the layout and cognitive overload.

4 EXPERIMENTS

In this section, we compared EV with PLSA+PE, LSP, ISOMAP, MDS and PCA for visualizing text data sets. Specifically, we implemented two EV models: EV-844 and EV-CUR, in our experiments. In EV-844, Algorithm 844 [2] is used to successively select a column or row at a time with the largest norm from text data, resulting in a unique subspace; while EV-CUR uses CUR [18] to pick the representative samples based on their probability distributions computed by the norms. Note that duplicates may exist in the CUR subspace because the samples with large norms are likely to be selected more than once. In the following, Section 4.1 gives the details of the data sets we used. In Section 4.2, we discussed the quantitative evaluation methods used to report the experimental results. On several public text data sets (including two large ones with 18,864 and 15,565 documents, respectively), we demonstrated the superior visualization results by EV in Section 4.3, in which we also compared the computational speed of all the algorithms.

4.1 Data Sets

For the experiments on document visualization, we used the *20Newsgroups* data [14] and *10PubMed* data.

20Newsgroups data consists of documents in the 20 Newsgroups corpus. The corpus contains 18,864 articles categorized into 20 discussion groups³ with a vocabulary size 26,214. Note that at its full size the data here is too large to be processed by all the algorithms except EV. In order to make the comparison with existing methods, we constructed two subsets of *20Newsgroups* through uniform random sampling: *20Newsgroups-I* and *20Newsgroups-II*, shown in Table 1.

10PubMed data consists of published abstracts in the MEDLINE database⁴ from 2000 to 2008, relating to 10 different diseases. We used “MajorTopic” tag along with the disease-related MeSH terms as queries to MEDLINE. Table 2 shows the 10 document sets (15,565 documents) retrieved. From all the retrieved abstracts, the common and stop words are removed, and the words are stemmed using Porter’s suffix-stripping algorithm [19]. Finally, we built a word-document matrix of the size 22437×15565 .

³<http://www.cs.uiuc.edu/homes/dengcai2/Data/TextData.html>

⁴<http://www.ncbi.nlm.nih.gov/pubmed/>

Table 1. Summary of data subsets from *20Newsgroups* used in the experiments.

Data Name	Groups	No. Docs per Group	Total Docs
<i>20Newsgroups-I</i>	{comp.sys.ibm.pc.hardware}, {rec.sport.baseball},{sci.med}	100	300
<i>20Newsgroups-II</i>	all 20 groups	50	1000

Table 2. Summary of *10PubMed* data used in the experiments.

	Document Name	No. of Docs
1	Gout	543
2	Chickenpox	732
3	Raynaud Disease	343
4	Jaundice	503
5	Hepatitis A	796
6	Hay Fever	1517
7	Kidney Calculi	1549
8	Age-related Macular Degeneration	3283
9	Migraine	3703
10	Otitis	2596

4.2 Evaluation Measurement

We evaluated the visualization results quantitatively based on the label predication accuracy with the k -nearest neighbor (k -NN) method [8] in the visualization space. Documents are labeled with discussion groups in the *20Newsgroups* data, and with disease names in the *10PubMed* data. Majority voting among the training documents in the k neighbors of a test document is used to decide its predicted label. The accuracy generally becomes high when documents with the same label are located together while documents with different labels are located far away from each other in the visualization space.

Quantitatively, the accuracy $AC(k)$ is computed as,

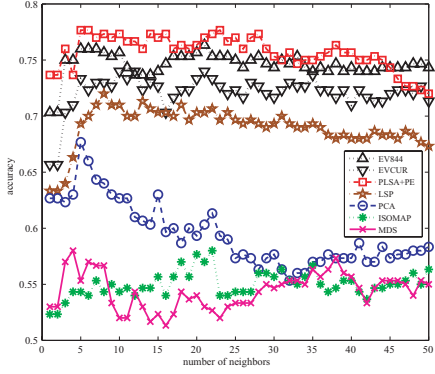
$$AC(k) = \frac{1}{n} \sum_{i=1}^n I(l_i, \hat{l}_k(\mathbf{y}_i)), \quad (7)$$

where n denotes the total number of documents in the experiment, l_i is the ground truth label of the i th document, $\hat{l}_k(\mathbf{y}_i)$ is the predicted label by k -NN in the embedding space, and I is the delta function that equals one if $\hat{l}_k(\mathbf{y}_i) = l_i$, and zero otherwise.

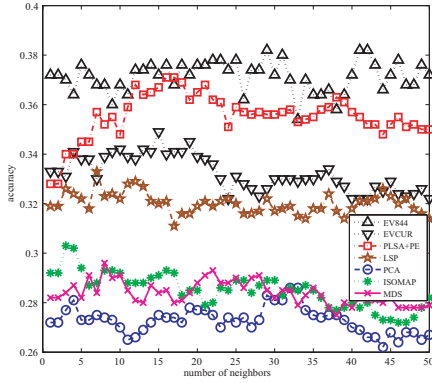
4.3 Results

First, on the data sets *20Newsgroups-I* and *20Newsgroups-II* we compared the neighbor-preserving accuracy in two-dimensional visualization generated by EV-844, EV-CUR, PLSA+PE, LSP, ISOMAP, MDS, and PCA. Through uniform random sampling, we created 10 independent evaluation sets for each data set, with given number of topics (3 for *20Newsgroups-I* and 20 for *20Newsgroups-II*) and documents (100 for *20Newsgroups-I* and 50 for *20Newsgroups-II*). The average accuracy values are obtained using k -NN over the 10 sets with $k = \{1, 2, \dots, 50\}$, shown in Figure 1.

Generally, the AC values obtained by the seven methods are higher for a small number of topics (e.g., $z=3$ in Figure 1(a)) than those with a large number of topics (e.g., $z=20$ in Figure 1(b)). Moreover, the accuracy achieved by the topic models (i.e., EV-844, EV-CUR, PLSA+PE and LSP) is significantly higher than the traditional projection methods (i.e., PCA, MDS and ISOMAP). These results indicate that topic information is very helpful for the data visualization. When visualizing real-world text corpus, particularly the ones collected from the World Wide Web, the number of topics is typically unknown and thus has to be estimated through topic model detection. Some well-known approaches include Bayesian Inference Criteria (BIC) and Minimum Message Length (MML). A detailed discussion of model detection can



(a)



(b)

Fig. 1. Accuracy with k -NN in the two-dimensional visualization space with different k : (a) *20Newsgroups-I* (3 topics), (b) *20Newsgroups-II* (20 topics).

be found in [15]. In our experiments, the number of topics for all the topic models is simply set based on the ground truth. Another important observation from Figure 1 is that EV-844 constantly provides a higher accuracy value than EV-CUR. This is mainly because Algorithm 844 selects unique columns (exemplars) while CUR may choose replicated ones to build the subspace. Thus, we used EV-844 in the rest of our experiments and referred it to EV without special mention. Finally, as shown in Figure 1(a), the two probabilistic topic models (i.e., EV and PLSA+PE) have comparable performance on *20Newsgroups-I*. However, as the number of topics increases, EV clearly outperforms PLSA+PE on *20Newsgroups-II* in Figure 1(b). These results imply that EV can appropriately embed documents in a two-dimensional Euclidean space while keeping the essential relationship of the documents, especially for a data set with a large number of topics.

Figures 2 and 3 show the visualization results obtained by EV, PLSA+PE, LSP, ISOMAP, MDS, and PCA on *20Newsgroups-I* and *20Newsgroups-II*, respectively. Here, each point represents a document, and the different color shapes represent the topic labels. For example, there are three different color shapes in Figure 2, representing three groups of news: black diamond for “comp.sys.ibm.pc”, green triangle for “rec.sport.baseball” and red circle for “sci.med”. In the EV visualization (Figure 2(f)), documents with the same label are nicely clustered together while documents with different labels tend to be placed far away. In PLSA+PE and LSP (Figures 2(e) and (d)), documents are located slightly more mixed than those in EV. On the other hand, with PCA, MDS and ISOMAP (Figures 2(a)-(c)), documents with different labels are mixed, and thus the AC values of the corresponding layout are very low. These results also imply that the topic

Table 3. Comparison of computation time (in seconds) for: EV, PLSA+PE, PCA, LSP, MDS and ISOMAP. A cross x indicates that an algorithm does not provide a result in a reasonable time.

Data size	EV	PLSA+PE	PCA	LSP	MDS	ISOMAP
n	$\mathcal{O}(n)$	$\mathcal{O}(n^2)$	$\mathcal{O}(n^2)$	$\mathcal{O}(f(n,s))$	$\mathcal{O}(n^3)$	$\mathcal{O}(n^3)$
1×10^3	0.49	0.42	0.40	15.25	20.48	200.05
2×10^3	0.95	1.50	1.36	30.40	216.62	1611.72
3×10^3	1.43	3.20	2.24	80.62	801.30	x
4×10^3	1.93	5.49	3.78	160.10	1881.00	x
5×10^3	2.55	8.38	x	x	x	x
1×10^4	5.79	x	x	x	x	x

models generally provide better visualization layout. Figures 3(a)-(f) show 20-topic news groups visualized by the six methods. Similarly, EV provides the best view since news in similar topics are closer while news of distinct topics are placed further away.

As discussed earlier, by choosing the dimension of the subspace, EV can visualize documents with different granularity and enhance the interpretability of the visualization. In Figures 2(g)-(i) and 3(g)-(i), the representative documents selected in the low-rank subspace are embedded in a two-dimensional layout, for *20Newsgroups-I* and *20Newsgroups-II*, respectively. In Figures 2(g)-(i), we provided a series of visualization for *20Newsgroups-I*, from the most abstract view to the visual layout with considerable amount of details as the number of selected exemplars increases from 10 to 40. This result demonstrates that EV can use exemplars to summarize the distribution of the entire document collection. Similarly, Figures 3(g)-(i) illustrate the visualization from abstract to details when the number of exemplars increases from 100 to 400 in *20Newsgroups-II*. In these figures, the overlapping in the original layout (Figure 3(f)) is greatly reduced, making users easier to understand the relations between news documents.

Second, we compared the computational speed of the six visualization methods: EV, PLSA+PE, PCA, LSP, MDS and ISOMAP. From a theoretical perspective, the time complexity of EV is $\mathcal{O}(n)$, PLSA+PE and PCA are $\mathcal{O}(n^2)$, LSP is $\mathcal{O}(f(n,s)) = \mathcal{O}(\max\{n^{\frac{3}{2}}, n\sqrt{s}\})$, and MDS and ISOMAP are $\mathcal{O}(n^3)$, where n is the number of documents and s is the condition number in LSP. Our experiments are performed on a machine with Quad 3GHz Intel Core2 processors and 4GB RAM. In order to compare under the same condition, the running time are reported based on a single iteration if an algorithm uses the iterative approach. Table 3 summarizes the computation time in seconds for all six methods with increasing number of documents. From Table 3, EV clearly is the quickest among the six, followed by PLSA+PE and PCA, while the computing time of LSP, MDS and ISOMAP increases quickly with the number of documents. More important, we observed that some algorithms fail to provide a result within a reasonable time for relatively large document sets. Specifically, ISOMAP is the slowest and cannot give a result when the matrix contains more than 3,000 documents due to insufficient memory. When we have more than 10,000 samples, only EV can provide a result within a reasonable computation time, while all other methods fail (indicated by a cross x in the table). Clearly, EV is suitable to visualize large text corpus we are increasingly facing these days thanks to its high computational efficiency.

We also developed an Exemplar-based Visualization software tool to offer a range of functions of creating visualization with user-specified configuration and thus supporting visual exploration of document data. First, when choose “VIEW All” menu, the system can show all the documents at once for the *20Newsgroups* and *10PubMed* data sets. In this case, EV is the only one among the six algorithms that can produce a projection in a reasonable time. For example, Figure 4(a) shows visualization by EV for the 18,864 documents in *20Newsgroups*. Again, each point represents a document, and the different color shapes represent the topic labels. Note that it is difficult to see the details because the number of documents is very large,

leading to extremely heavy overlapping. If one clicks “View Exemplars” and sets the number of exemplars at 1,000, Figure 4(b) shows the representative documents selected by EV to summarize the whole document collection. Clearly, the cognitive overload and serious overlapping are greatly reduced. Here, a big color shape indicates the mean coordinate of documents for one group, calculated by $\mu_l = \frac{1}{n_l} \sum_{i=1}^{n_l} I(i=l) \mathbf{y}_i$, where n_l is the number of documents labeled with l . Obviously, documents with the same label are clustered together, and similar documents with closely related labels are placed nearby, such as “comp.graphics”, “comp.os.ms.windows.misc” and “comp.windows.x” in the “computer” category, or “rec.autos”, “rec.motorecycles”, “rec.sport.baseball” and “rec.sport.hockey” in the “recreation” news group. Based on the visualized exemplars, EV provides several additional options for a user to further explore the data set. For example, on the click of “View Clusters”, a magnified layout of all corresponding documents in the groups of “comp.graphics”, “comp.os.ms.windows.misc” and “comp.windows.x” is given in Figure 4(c), which provides further details. Similarly, a user can specify a neighborhood (the rectangle in Figure 4(b)), clicking “Zoom In” will generate a magnified view of all or representative documents in the selected area. Also, if desired, further clustering and visualization can be performed in an area of interest, leading to a hierarchical structure for data exploration.

Figure 5 shows the exemplar-based visualization for the 15,565 documents in the *10PubMed* data set. Exemplars and means of *10PubMed* data illustrated in Figure 5(a) help us gain a better understanding on the distribution and relations of these documents. It is clear that documents with same disease are likely to be located closely while documents with different diseases are moved further away. We noticed that there is less overlapping in the *10PubMed* data set than in *20Newsgroups*. One reason is that the number of topics in *10PubMed* is less than in *20Newsgroups* while another one is that the abstracts in the literature for various diseases is actually easier to be separated than the documents in different news groups. The average value of *AC* is about 60% in the *10PubMed* data set; it is only approximately 30% in *20Newsgroups*. If desired, users can further explore the data set by clusters. In Figure 5(b), documents related to two diseases (“Gout” and “Chickenpox”) are shown, where the selected exemplars (100 in total) are emphasized by the bigger black shapes. First, our method provides a clear visualization with little clutter. Second, users can quickly browse the large document collection by reading only the representative documents (exemplars) in each cluster. The actual time required by EV to produce visualization for *20Newsgroups* and *10PubMed* (with 1,000 exemplars and 1,000 iterations) are 30 and 25 minutes, respectively. These results clearly show that EV provides a very powerful tool for visualizing large text data sets.

5 CONCLUSIONS AND FUTURE WORK

In the paper, we propose an Exemplar-based approach to Visualize (EV) extremely large text corpus. In EV, a representative text data subspace is first computed from the low-rank approximation of the original word-document matrix. Then, documents are soft clustered using the matrix decomposition and visualized in the Euclidean embedding space through parameter embedding. By selecting the representative documents, EV can visualize tens of thousands of documents with high accuracy (in retaining neighbor relations), high efficiency (in computation), and high flexibility (through the use of exemplars).

The algorithms discussed in this paper have been fully integrated into a visualization software package, which will be released publicly shortly after the Infovis Conference⁵. In the future, we plan to conduct practical user studies to solicit feedbacks so that the software can be improved with more convenient and user-friendly features. We also intend to pursue incorporating topic detection model into our system, making it more appropriate for real-world data visualization. Another direction we are considering for the future work is to develop an interaction tool based on the EV model for the visualization of other types of data.

⁵<http://vii.wayne.edu>

ACKNOWLEDGMENTS

This research was partially funded by U. S. National Science Foundation under grants IIS-0713315 and CNS-0751045, and by the 21st Century Jobs Fund Award, State of Michigan, under grant 06-1-P1-0193.

REFERENCES

- [1] P. Berkhin. Survey of clustering data mining techniques. Technical report, Accrue Software, San Jose, CA, 2002.
- [2] M. W. Berry, S. A. Pulatova, and G. W. Stewart. Algorithm 844: Computing sparse reduced-rank approximations to sparse matrices. *ACM Trans. Math. Softw.*, 31(2):252–269, 2005.
- [3] K. Borner, C. Chen, and K. Boyack. Visualizing knowledge domains. *Ann. Rev. Information Science and Technology*, 37:1–51, 2003.
- [4] T. Cox and M. Cox. *Multidimensional Scaling*. Chapman and Hall/CRC, 2nd. ed., 2001.
- [5] M. C. F. de Oliveira and H. Levkowitz. From visual data exploration to visual data mining: a survey. *IEEE Trans. on Visualization and Computer Graphics*, 9(3):378–394, 2003.
- [6] C. Ding, T. Li, and M. I. Jordan. Convex and semi-nonnegative matrix factorizations. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, accepted by 2008, to appear.
- [7] C. Ding, T. Li, and W. Peng. On the equivalence between non-negative matrix factorization and probabilistic latent semantic indexing. *Computational Statistics and Data Analysis*, 52(8):3913–3927, 2008.
- [8] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley, New York, 2nd edition, 2001.
- [9] Y.-H. Fua, M. Ward, and E. A. Rundensteiner. Hierarchical parallel coordinates for exploration of large datasets. In *proc. of VIS*, pages 43–50, 1999.
- [10] T. Hofmann. Probabilistic latent semantic analysis. In *proc. of UAI*, pages 289–296, 1999.
- [11] T. Iwata, K. Saito, N. Ueda, S. Stromsten, T. L. Griffiths, and J. B. Tenenbaum. Parametric embedding for class visualization. *Neural Computation*, 19:2536–2556, 2007.
- [12] T. Iwata, T. Yamada, and N. Ueda. Probabilistic latent semantic visualization: Topic model for visualizing documents. In *proc. of KDD*, pages 363–371, 2008.
- [13] I. T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, 2nd edition, 2002.
- [14] K. Lang. News weeder: Learning to filter netnews. In *proc. of ICML*, pages 331–339, 1995.
- [15] G. McLachlan and D. Peel. *Finite Mixture Models*. New York: John Wiley & Sons, Inc., first edition, 2002.
- [16] F. Paulovich and R. Minghim. Hipp: A novel hierarchical point placement strategy and its application to the exploration of document collections. *IEEE Trans. on Visualization and Computer Graphics*, 14(6):1229–1236, 2008.
- [17] F. Paulovich, L. G. Nonato, R. Minghim, and H. Levkowitz. Least square projection: a fast high-precision multidimensional projection technique and its application to document mapping. *IEEE Trans. on Visualization and Computer Graphics*, 14(3):564–575, 2008.
- [18] D. Petros, K. Ravi, and W. M. Michael. Fast monte carlo algorithms for matrices iii: Computing a compressed approximate matrix decomposition. *SIAM Journal on Computing*, 36:184–206, 2006.
- [19] M. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980.
- [20] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323 – 2326, 2000.
- [21] T. Soukup and I. Davidson. *Visual Data Mining: Techniques and Tools for Data Visualization and Mining*. John Wiley & Sons, Inc., 1st edition, 2002.
- [22] J. Sun, Y. Xie, H. Zhang, and C. Faloutsos. Less is more: Sparse graph mining with compact matrix decomposition. *Statistical Analysis and Data Mining*, 1(1):6–22, 2008.
- [23] R. Tejada, R. Minghim, and L. Nonato. On improved projection techniques to support visual exploration of multidimensional data sets. *Information Visualization*, 2(4):218–231, 2003.
- [24] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

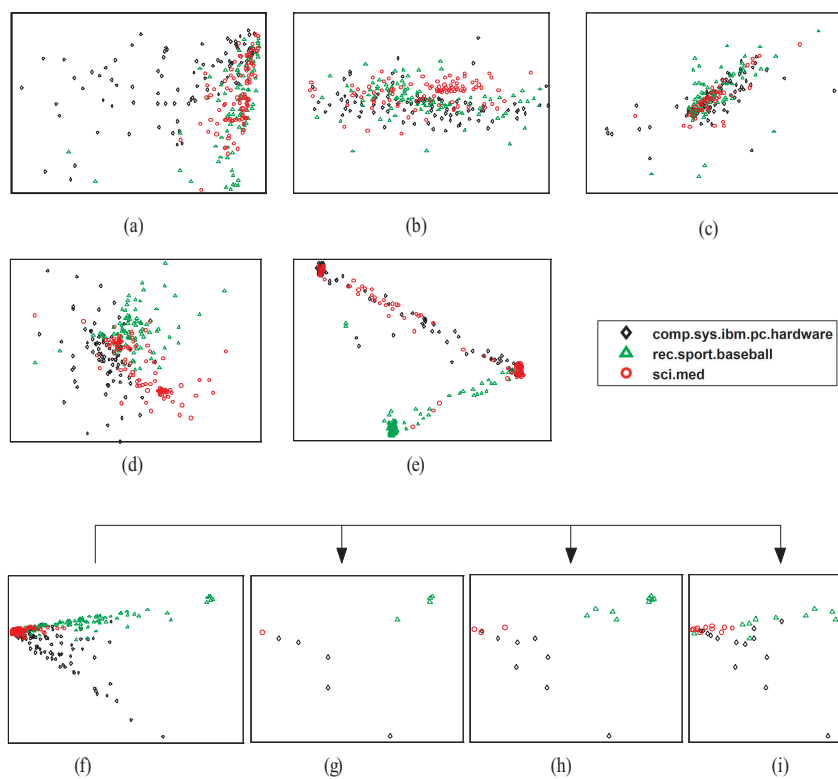


Fig. 2. Visualization of documents in *20Newsgroups-I* (300 documents, 3 topics) by (a)PCA, (b)MDS, (c)ISOMAP, (d)LSP, (e)PLSA+PE, and (f)EV, and visualization of (g)10 exemplars, (h)20 exemplars, (i)40 exemplars by EV.

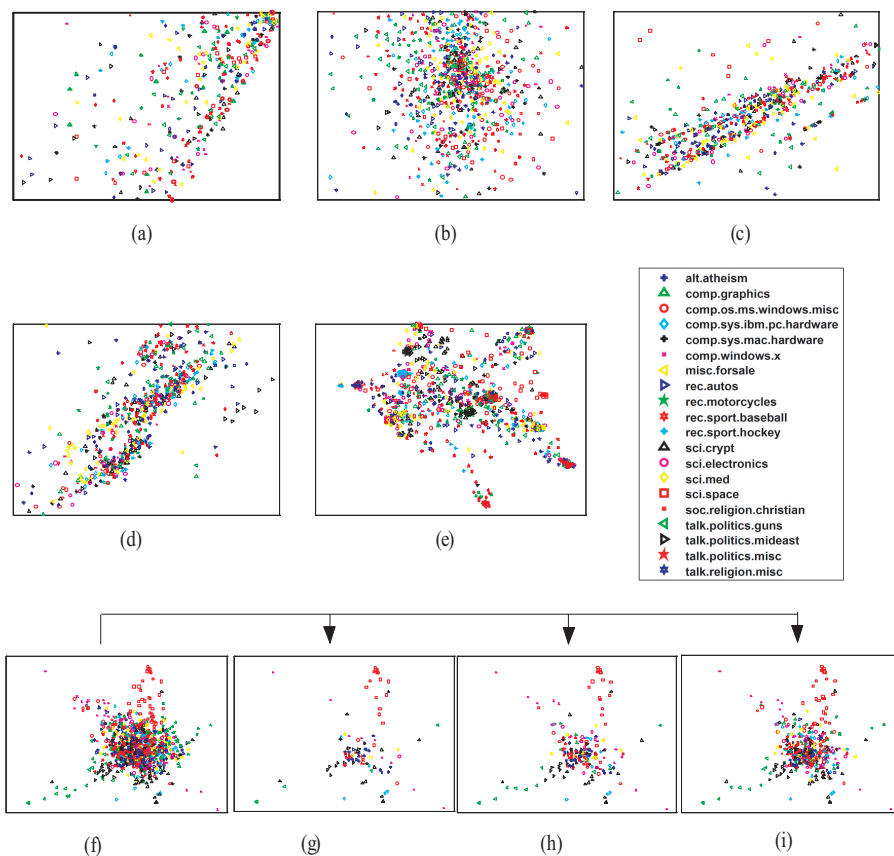


Fig. 3. Visualization of documents in *20Newsgroups-II* (1000 documents, 20 topics) by (a)PCA, (b)MDS, (c)ISOMAP, (d)LSP, (e)PLSA+PE, and (f)EV, and visualization of (g)100 exemplars, (h)200 exemplars, (i)400 exemplars by EV.

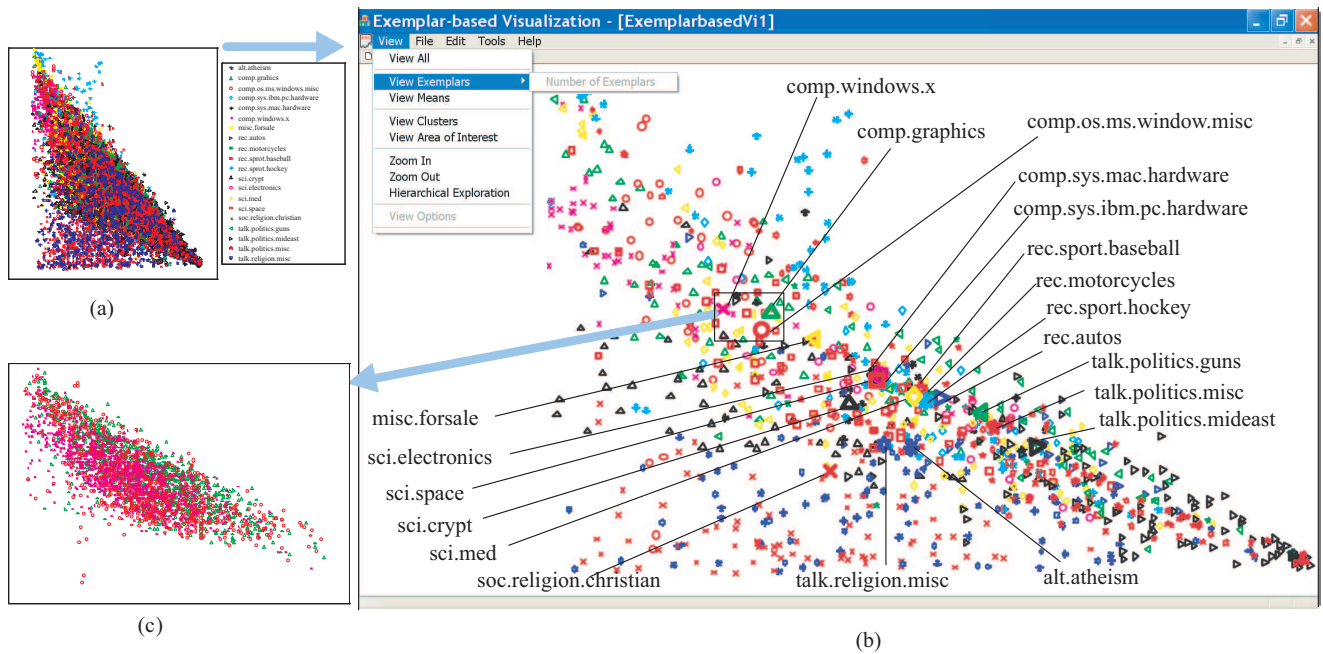


Fig. 4. Visualization of documents in *20Newsgroups* (18,864 documents, 20 topics) by EV. Each point represents a document; each color shape represents a news topic; and the corresponding big color shape indicates the mean of a news group. Visualization of (a) all documents, (b) 1000 exemplars with their means, (c) three similar groups of news: “comp.graphics”, “comp.os.ms.window.misc” and “comp.windows.x”.

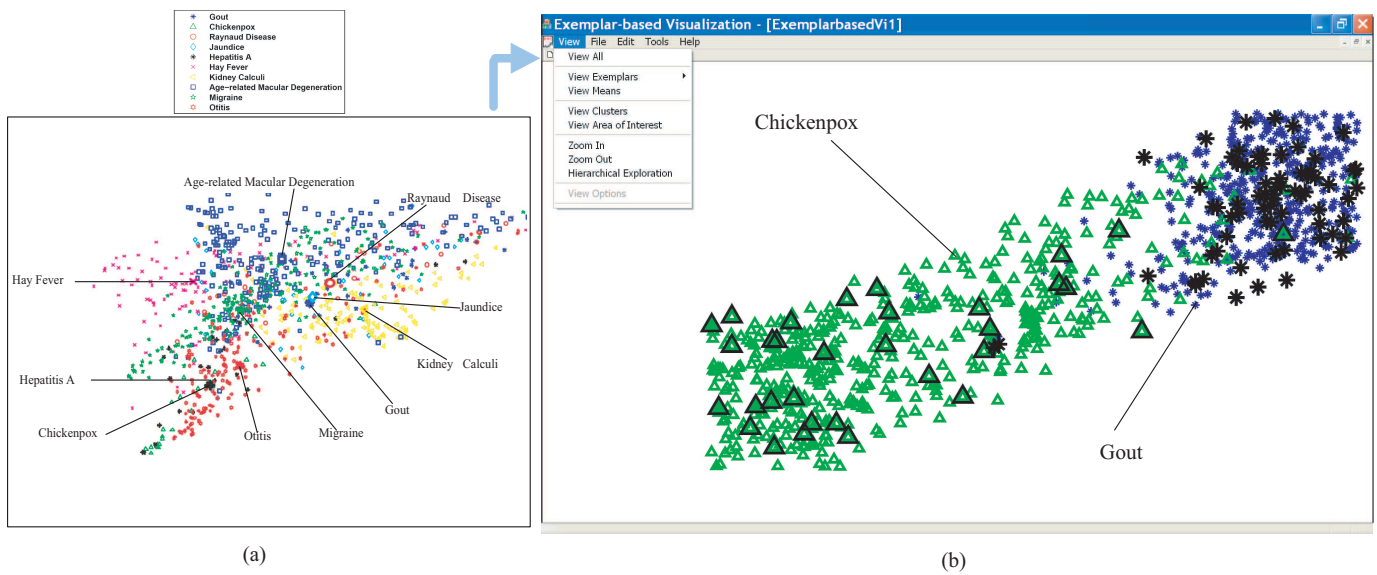


Fig. 5. Visualization of abstracts in *10PubMed* (15,565 documents, 10 topics) by EV. Each point represents an abstract; each color shape represents a disease; and the corresponding big color shape indicates the means of an abstract group. Visualization of (a) 1000 exemplars with their means, (b) two distinct groups of diseases: “Gout” and “Chickenpox” with the selected exemplars (100 in total), emphasized by the bigger black shapes.