Dimensionality Reduction in Big Data using Unsupervised Learning An Overview

K Swanthana
Computer Science Engineering Department, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad, Telangana, INDIA

K Swapnika
Computer Science Engineering Department, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad, Telangana, INDIA

Dr Y VijayaLatha
Head, Information technology, Department, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad, Telangana, INDIA

Abstract—Unsupervised dimensionality reduction aims at representing high-dimensional data in lower-dimensional spaces by preserving important structural properties in a faithful way. Domains like text, image, video, audio contain large amounts of redundancies and ambiguities among the attributes which result in considerable noise effects. Retrieving the data from high dimensional datasets is a big challenge. Dimensionality reduction techniques have been a successful avenue for automatically extracting the latent concepts by removing the noise and reducing the complexity in processing the high dimensional data. In this paper we give an overview of unsupervised dimensionality reduction techniques for text retrieval task. The paper ends with a short review of the techniques for dimensionality reduction in the previous proceedings.

Index Terms— Dimensionality reduction, latent, redundancy, Unsupervised.

I. INTRODUCTION

Dimensionality Reduction (DR) methods work in unsupervised way: they process data features without taking into account additional information like class labels, which are then sometimes used to assess DR quality [14]. Dimensionality reduction can be used for different purposes, ranging from exploratory data analysis (visual inspection) to data compression before subsequent processing. In the latter case, DR can be seen as a way to defeat the so-called curse of dimensionality, which makes many complex analysis tasks like regression or classification much more difficult in high-dimensional spaces than in low-dimensional ones.

If visualization is difficult in high-dimensional space, perhaps an (almost) equivalent representation in a lower-dimensional space could improve the readability of data. This is precisely the idea that underlies the field of dimensionality reduction (DR). This domain includes various techniques that are able to construct meaningful data representations in a space of given dimensionality. Beside visualization, other applications of DR are: data compression and denoising. Dimensionality reduction can also preprocess data, with the hope that a simplified representation can accelerate any subsequent processing or improve its outcome.

Data collection and storage becomes easier and cheaper every day. Processing large amounts of data raises many issues, in terms of algorithmic complexity (time and memory consumption), workload distribution (vectorised, parallel, or distributed architectures), and efficient visual presentation of the results. Politics and media have coined the term “Big Data” to refer to these problems and the effort to alleviate them. In a recent interview for the INNS Big Data conference, though, Jurgen Schmidhuber said: “At any given moment, big data is more data than most people can conveniently store". Thereby he pointed out nicely that big data was, is, and will always remain an open question, although it only became popular very recently.

II. DIMENSIONALITY REDUCTION

Dimensionality reduction is an unsupervised task that allows high-dimensional data to be processed in lower-dimensional spaces. As the dimensionality of data increases query performance decreases, and thus demand for processing power and storage space increases. This problem of high dimensionality is defined as the curse of dimensionality [23]. As a result of this, efficiency of data indexing structure decreases rapidly with the increase in the number of dimensions. Existing indexing structures perform well in low dimensionality spaces and poorly in high dimensionality spaces. Solution for this problem is to reduce the dimensionality of the search space before indexing

the data. The dimensionality reduction can be made in two different ways: Using feature selection that keeps the most relevant variables from the original dataset or by using DR that exploits the redundancy of the input data and by finding a smaller set of new variables, each being a combination of the input variables, containing basically the same information as the input variables. In fact, one of the most widely used dimensionality reduction techniques, Principal Component Analysis (PCA), dates back to Karl Pearson in 1901 [18]. In this paper, we study about high dimensionality and evaluate four popular DR techniques for text retrieval task.

DR techniques are proposed as a data pre-processing step. This process identifies a suitable low dimensional representation of original data. Reducing the dimensionality improves the computational efficiency and accuracy of the data analysis. Mathematically the problem of dimension reduction can be defined as: Given a *r*-dimensional random vector $\mathbf{X}=(x1,x2,\ldots xr)$T, the objective is to find a representation of lower dimension $\mathbf{S}=(s1,s2,\ldots,sk)$T, where $k<r$, which preserves the content of the original data, as much as possible according to some criterion. DR techniques are classified as supervised and unsupervised techniques based on the learning process.

Supervised algorithms need a training set with the class label information to learn the lower dimensional representation according to some criteria and then predict the class labels on unknown text data. Linear Discriminant Analysis (LDA), Maximum Margin Criterion (MMC), Orthogonal Centroid (OC) algorithm are some of the supervised DR techniques [3, 23].

Unsupervised approaches such as SVD project the original data to a new lower dimensional space without utilizing the label information. DR operates either by transforming the existing features to a new reduced set of features or by selecting a subset of the existing features.

A. Feature transformation

Feature transformation techniques aim to reduce the dimensionality of data to a small number of dimensions which are linear or non-linear combinations of the vector coordinates in the original dimensions. These techniques are believed to be successful in uncovering the latent structures in the datasets [9]. Examples of various feature transformation techniques include PCA, ICA, Projection pursuit and Factor analysis.

1. Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is by far one of the most popular algorithms for dimensionality reduction [18, 22, 8, 12]. Given a set of observations **X**, with dimension *M* (they lie in R^M), PCA is the standard technique for finding the single best (in the sense of least-square error) subspace of a given dimension, *m* . Without loss of generality, we may assume the data is zero-mean and the subspace to fit is a linear subspace (passing through the origin).

2. Independent Component Analysis (ICA)

Independent Component Analysis (ICA) tries to linearly transform the original data into components that are maximally independent of each other. ICA assumes that the observed multivariate data are linear or non-linear mixtures of some unknown latent variables with unknown mixing coefficients. These latent variables are called the independent components of the data. ICA technique seeks linear projections that are as independent as possible. However, these projections are not necessarily orthogonal to each other. For DR, ICA finds *k* components that effectively capture variability of the original data. ICA factors a data matrix, **A** of size *t* x *d* as

A =C.F

where **C** is described as the mixing matrix with *t* rows and *k* columns and **F** is the matrix of independent components with *k* rows and *d* columns. In this study, Fast ICA algorithm is used to identify the latent dimensions in the data [10]. To speed up the iteration process, the observed data can be uncorrelated by a linear transformation called

„pre-whitening". ICA is well studied by researchers in signal processing. ICA defines interestingness in terms of the directions that are statistically independent and least normally distributed. Applications of ICA can be found on text data in [14].

3. Projection Pursuit

Projection Pursuit (PP) is a type of statistical technique which involves finding the most "interesting" possible projections in multidimensional data. Often, projectionswhich deviate more from a normal

distribution are considered to be more interesting. As each projection is found, the data are reduced by removing the component along that projection, and the process is repeated to find new projections; this is the "pursuit" aspect that motivated the technique known as matching pursuit. The idea of projection pursuit is to locate the projection (or projections) from high dimensional space to low-dimensional space that reveals the most of the details about the structure of the data set. Once an interesting set of projections has been found, existing structures (clusters, surfaces, etc.) can be extracted and analyzed separately [21].

4. Factor analysis

A data reduction technique is designed to represent a wide range of attributes on a smaller number of dimensions. Suppose that a bank asked a large number of questions about a given branch. Consider how the following characteristics might be more parsimoniously represented by just a few constructs (factors) [10, 11].

Service:

- Friendliness of staff.
- Time Spent in line-up.
- Assistance via telephone. Convenience
- Distance of bank from home
- Hours of operation
- Availability of parking.
- Proximity to other stores where you frequently shop.

Cost

- Monthly account fee
- Change for with-drawls and deposits.
- Lone interest rate.

Benefits include: (1) a more concise representation of the marketing situation and hence communication may be enhanced; (2) fewer questions may be required on future surveys; and, (3) perceptual maps become feasible. Ideally, interval data (e.g., a rating on a 7 point scale), regarding the perceptions of consumers are required regarding a number of features, such as those noted above for the bank are gathered.

B. Feature selection

Unsupervised feature selection techniques are much harder than the supervised techniques. The goal of feature selection for unsupervised learning is to find the smallest feature subset that best uncovers "interesting natural" groupings (clusters) from data according to the chosen criterion [1, 4]. Feature selection techniques for unsupervised learning can be categorized as filter, wrapper and embedded approaches [14]. But in unsupervised learning, we are not given class labels. The problem is that not all features are important. Some of the features may be redundant, some may be irrelevant, and some can even misguide clustering results. In addition, reducing the number of features increases comprehensibility and ameliorates the problem that some unsupervised learning algorithms break down with high dimensional data [1, 17, 13].

1. Filtering approach

The filter approach basically pre-selects the features, and then applies the selected feature subset to the clustering algorithm. This approach ranks features or feature subsets independently of the predictor (classifier) by using univariate methods (i.e by consider one variable at a time) or by using multivariate methods (i.e by consider more than one variables at a time)[11,14,15].

Figure 1: Filtering approach for unsupervised learning

2. Wrapper approach

The wrapper approach [19] incorporates the clustering algorithm in the feature search and selection. It uses a classifier to assess (many) features or feature subsets. The wrapper approach is used to cluster the data as best we can in each candidate feature subspace according to what "natural" means, and select the most "interesting" subspace with the minimum number of features. This framework is inspired by the supervised wrapper

approach [11, 14], but rather than wrap the search for the best feature subset around a supervised induction algorithm, we wrap the search around a clustering algorithm [19].

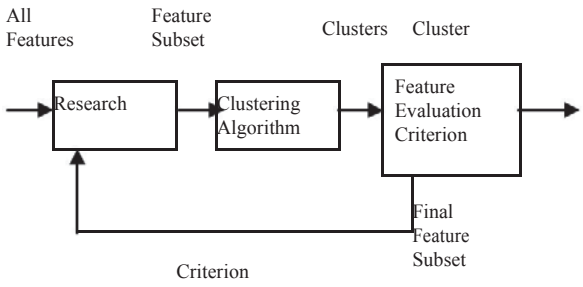

Figure 2: Wrapper approach for unsupervised learning

3. Embedded approach

Embedded methods learn which features best contribute to the accuracy of the model while the model is being created. The most common type of embedded feature selection methods are regularization methods. Regularization methods are also called penalization methods that introduce additional constraints into the optimization of a predictive algorithm (such as a regression algorithm) that bias the model toward lower complexity (less coefficients). Examples of regularization algorithms are the LASSO, Elastic Net and Ridge Regression [11, 17, 21].

Therefore the Main goal of these three approaches is to rank subsets of useful features**.** The difference between the approaches is shown in the below figure.

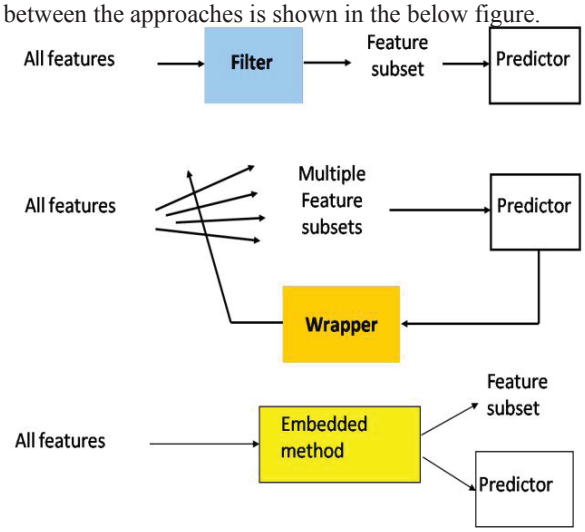

Figure 3: filters vs. wrappers vs. Embedding

C. Singular Value Decomposition

A document collection of t terms and d documents is represented by a term-document matrix with t rows, d columns and with rank r. Vectors representing documents and queries are projected in new, low dimensional space obtained by truncated SVD. The SVD of a term-document matrix **A** is written as [1, 20].

$$\mathbf{A} = \mathbf{U}.\mathbf{\Sigma}.\mathbf{V}^{T}$$

If the term-document matrix **A** is *t* x *d*, then **U** is a *t* x *r* orthogonal matrix, **V** is a *d* x *r* orthogonal matrix and **Σ** is *r* x *r* diagonal matrix where the values on the diagonal of **Σ** are called the singular values. Singular values can then be sorted in decreasing order and the top *k* (*k*<*r*) values are selected as a means of developing a latent semantic representation of original matrix. The geometric interpretation of SVD is to consider the columns of **V**T as defining the new axes, the rows of **U** as coordinates of the objects in the space spanned by these new axes and **Σ** as a scaling factor indicating the relative importance of each new axis [4]. By changing (*r*-*k*) rows of **Σ** to zero rows a low rank approximation to **A** called $\mathbf{A}_K$ can be created through the truncated SVD as,

$$\mathbf{A}_K = \mathbf{U}_K . \mathbf{\Sigma}_K . \mathbf{V}_K^T$$

Where $\mathbf{U}_K$ is the t x k term-concept matrix, $\mathbf{\Sigma}_K$ is the k x k concept-concept matrix, $\mathbf{V}_K^T$ is the k x d concept-document matrix. Only the first k columns are retained in $\mathbf{U}_K$ and k rows are retained in $\mathbf{V}_k^T$. By applying the SVD on a term-document matrix, documents will be represented in a vector space of artificial concepts. Each of the k reduced dimensions corresponds to a latent concept which helps to discriminate the documents.

D. Non-Negative Matrix Factorization

Non-negative Matrix Factorization (NMF) is another development in the field of DR and clustering [16]. The semantic space derived by NMF contains each axis capturing the base topic of a particular document cluster and each document is represented as an additive combination of base topics [15]. NMF is proved to be useful in approximating high dimensional data comprised of non-negative components [2].

For the data matrix **A** of size t x d with each column of t dimensional non-negative vector of original database (d vectors), NMF factorizes **A** as

$$\mathbf{A}=\mathbf{W}.\mathbf{H}$$

Where **W** is t x k and **H** is k x d and $k \leq d$. Each column of **W** contains a basis vector and each column of **H** contains the weights needed to approximate the corresponding columns in **A** using the basis from **W**. Here the choice of k is mostly dependent upon the characteristics of the particular database within the application. In contrast to SVD, NMF does not need to be orthogonal and each document is guaranteed to take only non-negative values in all the latent semantic directions.

E. Fuzzy K-Means Clustering

In [5] Dhillon and Modha have used centroids of clusters which are created using spherical K-means algorithm for lowering rank of the term-document matrix. The IR technique using clustering for decomposition is called Concept Indexing (CI). Concept index is a space containing linear combinations of centroids of the clusters. CI is computationally more efficient and requires less memory than LSI. Dobsa and Dalbelo-Basic [6] have proposed an improvement to CI using Fuzzy K-Means (FKM) algorithm for decomposition.

The FKM algorithm works on the assumption that there are natural tendencies of cluster structure in the data and its goal is to uncover this latent structure [6, 7]. In contrast to crisp or hard clustering techniques, FKM algorithm allows the objects to partially belong to multiple clusters. FKM partitions a set of t dimensional vectors $\mathbf{X}=\{X_1,X_2....X_d\}$ into k clusters where $Xj=\{x_{j1},x_{j2},....,x_{jt}\}$ represents the jth sample for $j=1...d$. Every cluster is a fuzzy set. For the jth sample Xj and the i^{th} cluster center v_i, there is a membership degree u_{ij} indicating with what degree the sample Xj belongs to the cluster center v_i resulting in a fuzzy partition matrix $\mathbf{U}=(u_{ij})_{txk}$. The FKM algorithm is based on minimizing the heuristic global objective function $\mathbf{J}_{Fuzz}$ defined as where dij is the Euclidean distance between Xj to the cluster center vi defined as,

$$I_{fuzz} = \sum_{i=1}^{k} \sum_{j=1}^{d} u_{i,j}^{m} d_{ij}$$

The exponent b in equation 5 is called as fuzzifier parameter and determines the fuzziness of the clustering. In all our experiments we consider the value of b to be 1.05. For higher values of b the clustering becomes more fuzzifier.

III. CONCLUSION

In this paper, the problem of high dimensionality for text retrieval is discussed. DR techniques are used to improve the data representation by understanding the data in terms of concepts rather than words. The objective of this paper is to provide an overview of unsupervised DR techniques for text retrieval task.

REFERENCES

[1] Aswani Kumar, Ch. Srinivas, S.: "Latent semantic indexing using eigenvalue analysis for efficient information retrieval". International Journal of Applied Mathematics and Computer Science, Vol. 16, No. 4, pp. 551-558. (2006).

[2] Berry, M.W., Browne, M., Langville, A.N., Pauca, V.P., Plemmons, R.J.: "Algorithms and applications for approximate nonnegative matrix factorization". Computational Statistics and Data Analysis. Vol. 52, No.1, pp. 155-173. (2007).

[3] Cunningham, P. :Dimension reduction. Technical Report: UCD-CSI-2007-7. (2007).

[4] Deerwester, S., Dumais, S., Furnas, G., Landauer, T., Harshman, R.: "Indexing by latent semantic analysis", Journal of the American Society for Information Science. Vol. 41, No. 6, pp. 391-407. (1990)

[5] Dhillon, I.S., Modha, D.S.: "Concept decomposition for large sparse text data using clustering. Machine Learning", Vol. 42, No. 1, pp. 143-175. (2001)

[6] Dobsa, J., Dalbelo-Basic, B.: "Concept decomposition by fuzzy k-means algorithm. In Proceedings International conference on Web Intelligence" pp. 684-688. (2003)

[7] Doring, C., Lesot, M.J., Kruse, R.: "Data analysis with fuzzy clustering methods. Computational statistics and data analysis". Vol. 51, No. 1, 192-214. (2006).

[8] [Dunteman 1989] Dunteman, G. H. "Principal Component Analysis Sage Publications", 1989

[9] Foder, I.K.: "A survey of dimension reduction techniques". Technical report URL-ID- 148494, Center for applied scientific computing, Lawrence Livermore National Laboratory. (2002)

[10] Raoul Harel Selman Ercan Elgar de Groot Stijn van Schooten "Applying feature selection methods on fMRI data" March 29, 2014.

[11] "An Introduction to Feature Extraction" Isabelle Guyon and Andr´e Elisseeff, 2008

[12] [Jollife 2002] Jollife, "I. T. Principal Component Analysis Wiley", 2002

[13] Jennifer G. Dy and Carla E. Brodley "Feature Selection for Unsupervised Learning" Journal of Machine Learning Research 5, pp.845–889, (2004).

[14] Kolenda, T., Hansen, L.K., Sigurdsson, S.: "Independent components in text. Advances in Independent Component Analysis", Springer-Verlag. (2000)

[15] Lee, D.D., Seung, H.S.: "Algorithms for non-negative matrix factorization. Advances in Neural Information Processing Systems". Vol. 13, 556-562. (2001)

[16] Lee, D.D., Seung, H.S.: "Learning the parts of objects by non-negative matrix factorization. Nature". Vol. 401, 788-791. (1999)

[17] Noelia Sánchez-Maroño, Amparo Alonso-Betanzos, María Tombilla- Sanromán "Filter Methods for Feature Selection" – A Comparative Study **Special Issue Paper** in Computer Science - Research and Development Volume 4881 of the series Lecture Notes in Computer Science pp 178-187.

[18] [Pearson 1901] Pearson, K. "On Lines and Planes of Closest Fit to Systems of Points in Space. Philosophical Magazine", 1901, Vol.2, pp.559-572

[19] R. Kohavi and G. H. John. "Wrappers for feature subset selection". *Artificial Intelligence*, 97(1-2): 273–324, 1997.

[20] Skillicorn, D.B., McConnell, S.M., Soong, E.Y.: "Handbook of data mining using matrix decompositions". Queen's University, Canada. (2003).

[21] Thomas Navin Lal1 , Olivier Chapelle1 , Jason Weston2 , and Andr´e Elisseeff3, "Embedded Methods" Feature Extraction Volume 207 of the series Studies in Fuzziness and Soft Computing pp 137-165.

[22] [Wold 1987] Wold, S.; Esbensen, K. & Geladi, P. "Principal component analysis Chemometrics and Intelligent Laboratory Systems", 1987, 2, 37-5.

[23] Yan, J., Zhang, B., Liu, N., Yan, S., Cheng, Q., Fan, W., Yang, Q., Xi, W., Chen, Z.: "Effective and efficient dimensionality reduction for large-scale and streaming data preprocessing", IEEE Transaction on Knowledge and Data Engineering, Vol. 18, No. 3, 320-333. (2006).