Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 5, May 2014, pg.1033 – 1040

RESEARCH ARTICLE

Efficient High Dimensional Data Clustering Using Hubness Phenomenon

Miss. Sayali P. Barde[1]
Department of Computer Science& Engineering.
G.H.Raisony Academy College of Engineering& Technology, Nagpur, India
barde.sayali@gmail.com

Prof. Vikrant Chole[2]
Department of Computer Science& Engineering.
G.H.Raisony Academy College of Engineering& Technology, Nagpur, India.
vikrantchole@gmail.com

Prof. L. H. Patil[3]
Department of Computer Technlogy.
Priyadarshani Institute of Engineering & Technology, Nagpur, India.

Abstract: ***High dimensional data occur naturally in many domains and have presented great challenges for traditional data mining techniques. Traditional clustering algorithms become computational expensive when data set to be cluster is large. The curse of dimensionality refers to various phenomena that arise when analyzing data is high dimensional data that do not occur in low dimensional data. The common theme of this problem is that, when the dimensionality increases, the volume of the space increases so fast that the available data becomes sparse. So that in high dimensional data all objects appear to be dissimilar in many ways so it becomes difficult to cluster. A new aspect of curse of dimensionality referred to as hubness that affects the distribution of K-occurrences. In this paper, we proposed and implement hubness based clustering over the high dimensional data. More specifically , hubness i.e. the tendency of high dimensional data to contain points(hubs) that frequently occur in k-nearest neighbor list of other points where hubs can be used effectively as cluster prototypes.***

Keywords: Clustering, curse of dimensionality, hubs, k-nearest neighbour.

I. Introduction

Cluster analysis divides data into meaningful groups based on the objects and their relationships. Many important problems involve clustering large datasets. Cluster analysis or clustering is the task of assigning a set of objects into groups called clusters. Main task of clustering are explorative data mining, and a common technique for statistical data analysis used in many fields, including machine learning, pattern recognition, image analysis, information retrieval, and

bioinformatics. Over the years, various clustering algorithms have been proposed, which can be roughly divided into four groups: partitional, hierarchical, density-based, and subspace algorithms. High-dimensional data arise naturally in many domains, and have regularly presented a great challenge for traditional data-mining techniques, both in terms of effectiveness and efficiency. Clustering becomes difficult due to the increasing scarcity of such data, as well as the increasing difficulty in distinguishing distances between data points [1].

The *k*-nearest neighbor algorithm is one of the simplest pattern classification algorithms. It is based on a notion that instances which are judged to be similar in the feature space often share common properties in other attributes, one of them being the instance label itself.

The literature survey suggests many algorithm and techniques used for clustering. In centroid-based clustering, clusters are represented by a central vector, which may not necessarily be a member of the data set. When the number of clusters is fixed to k, k-means clustering gives a formal definition as an optimization problem: find the k cluster centers and assign the objects to the nearest cluster center, such that the squared distances from the cluster are minimized. In density-based clustering, clusters are defined as areas of higher density than the remainder of the data set. Objects in these sparse areas - that are required to separate clusters - are usually considered to be noise and border points.

In distribution based clustering, clusters can easily be defined as objects belonging most likely to the same distribution. A nice property of this approach is that this closely resembles the way artificial data sets are generated: by sampling random objects from a distribution. Connectivity based clustering, also known as hierarchical clustering, is based on the core idea of objects being more related to nearby objects than to objects farther away. These algorithms connect "objects" to form "clusters" based on their distance. A cluster can be described largely by the maximum distance needed to connect parts of the cluster.

For high dimensional data set different clustering algorithms are there as follows:
Subspace clustering is the task of detecting all clusters in all subspaces. This means that a point might be a member of multiple clusters, each existing in a different subspace. Subspaces can either be axis-parallel or affine. The term is often used synonymous with general clustering in high-dimensional data. Correlation clustering provides a method for clustering a set of objects into the optimum number of clusters without specifying that number in advance. Correlations among subsets of attributes result in different spatial shapes of clusters. Hence, the similarity between cluster objects is defined by taking into account the local correlation patterns.

II. Literature Survey

The following sections explain the survey of various papers. The phenomenon of *hubness* was further explored in natural property of many inherently high-dimensional data sets. Here we discuss about hubness phenomenon which occurred in high dimensional data and various methods that are used for clustering high dimensional data by using k-nearest neighbor algorithms.

The phenomenon of hubness was further explored where it was shown that hubness is a natural property of many inherently high-dimensional data sets. Not only do some very frequent points

emerge, but the entire distribution of k-occurrences exhibits very high skewness. In other words, most points occur very rarely in k-neighbor sets, less often than what would otherwise have been expected. We refer to these rarely occurring points as anti-hubs.

In k-NN algorithm, M. Radovanovic and A. Nanopulous [7] discusses that, the basic weighted k-nearest neighbor voting framework is retained. Each neighbor votes by its own label and the label weight is determined so as to minimize the influence of bad hubs on classification outcome. Even this simple weighting scheme was shown to often lead to significant improvements over the basic k-NN algorithm.

In [8] instead of observing only good and bad hubness, it is possible to take into account class-specific previous k-occurrences, i.e. class hubness. The h-FNN algorithm is based on this notion and it integrates class hubness information into a fuzzy k-nearest neighbor voting framework. It uses a threshold to distinguish between low hubness points (anti-hubs) and medium-to-high hubness points where inference based on class hubness is meaningful. Therefore, it requires a separate mechanism to deal with anti-hub.

In Naïve hubness Bayesian k-NN(NHBNN) [9], all k-occurrences are observed as random events. The class affiliation for a new instance is then inferred via a naïve Bayesian inference from the respective *k*-NN set. Experiments show that NHBNN compares favorably to different variants of the k-NN classifier, including probabilistic k-NN (PNN) which is often used as an underlying probabilistic framework for NN classification, signifying that NHBNN is a promising alternative framework for developing probabilistic-NN algorithms.

In [10] Hubness-information *k*-nearest neighbor (HIKNN) is a robust algorithm which uses both the information contained in an instance label and the information contained in its previous occurrences. This approach is based on an information-theoretic perspective, so that the vote of x is shifted more towards using class hubness if $N_k(x)$ is high and more towards the label of x if $N_k(x)$ is low. The algorithm also weights all the individual fuzzy votes based on their total occurrence frequencies, so that more weight is given to anti-hubs, since they are considered more local to the point of interest and, therefore, more important when trying to determine its label.

III. Proposed Method

K-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. The k-NN algorithm is among the simplest of all machine learning algorithms. In pattern recognition, the k-Nearest Neighbors algorithm (or k-NN for short) is a non-parametric method used for classification and regression. K-nearest neighbor algorithm (KNN) is part of supervised learning that has been used in many applications in the field of data mining, statistical pattern recognition and many others.KNN is a method for classifying objects based on closest training examples in the feature space.

An object is classified by a majority vote of its neighbors. K is always a positive integer. The neighbors are taken from a set of objects for which the correct classification is known. It is usual to use the Euclidean distance.

The algorithm on how to compute the K-nearest neighbors is given as follows:

1. Determine the parameter K = number of nearest neighbors beforehand. This value is all up to you.
2. Calculate the distance between the query-instance and all the training samples. You can use any distance algorithm.
3. Sort the distances for all the training samples and determine the nearest neighbor based on the K-th minimum distance.
4. Since this is supervised learning, get all the Categories of your training data for the sorted value which fall under K.
5. Use the majority of nearest neighbors as the prediction value.

After finding hubs in high dimensional data, we have to cluster those high dimensional data by applying clustering algorithm over that hubs .In this paper, we use hybrid approach for clustering which include hierarchical algorithm and DBSCAN algorithm.

Hierarchical clustering is an agglomerative (top down) clustering method. The idea of this method is to build a hierarchy of clusters, showing relations between the individual members and merging clusters of data based on similarity.

Clustering based on density (local cluster criterion), such as density-connected points or based on an explicitly constructed density function.

1. Arbitrary select a point *p*
2. Retrieve all points density-reachable from *p* wrt *Eps* and *MinPts*.
3. If *p* is a core point, a cluster is formed
4. If *p* is a border point, no points are density-reachable from *p* and DBSCAN visits the next point of the database.
5. Continue the process until all of the points have been processed**.**

IV. Experimental Setup

Data collection:
We tested our approach on high-dimensional data set. The data is collected from UCI data sets(archive.ics.uci.edu/ml/machinelearning-databases/heart-disease/heartdisease.names).
All attributes are numeric-valued. This database contains 78 attributes and 200 numbers of instances. The "goal" field refers to the presence of heart disease in the patient. We have experimented with random neighborhood size on heart disease high dimensional data. There is no known way of selecting the best k for finding neighbor sets, the problem being domain specific.

Preprocessing Techniques:
Raw data is highly susceptible to noise, missing values, and inconsistency. The quality of data affects the data mining results. In order to help improve the quality of the data and consequently, of the mining results raw data is preprocessed so as to improve the efficiency and ease of the mining process. Data preprocessing is one of the most critical steps in a data mining process

which deals with the preparation and transformation of the initial dataset. Data preprocessing methods are divided into following categories: Data cleaning, data Integration, data transformation, data Reduction.

V. Result Analysis

The proposed algorithms represent only one possible approach to using hubness for improving high dimensional data clustering. This shows that hubs can serve as good cluster center prototypes.

In this project, we have designed home page where high dimensional data file inserted and saved successfully. Home page contains an algorithm which is used for clustering.

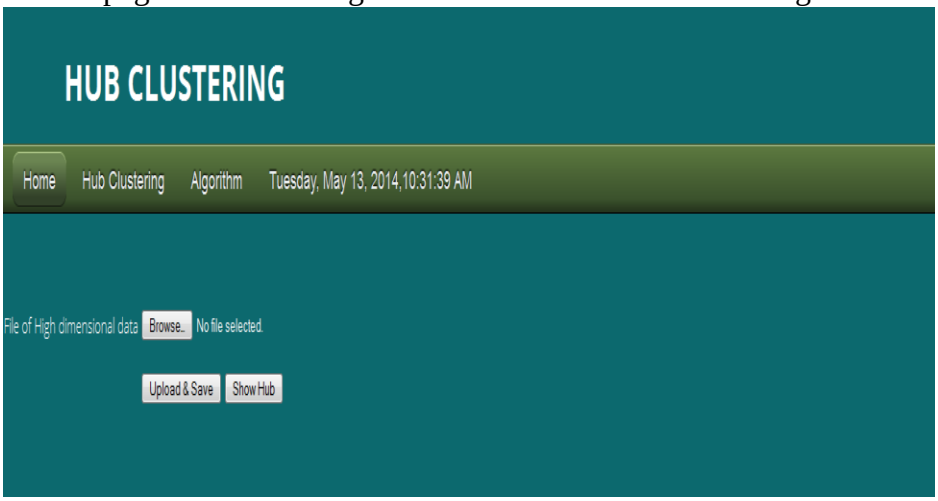

Fig 1: Home page of hubness based clustering

After uploading high dimensional data hubs are found out by k-nearest neighbor algorithm. According to those hubs clusters are formed. Those clusters are graphically shown in following modules.

In figure 2, X-axis plotted as number of doctors and Y-axis as city name. In figure 3, X-axis plotted as number of patients and Y-axis as doctors name.

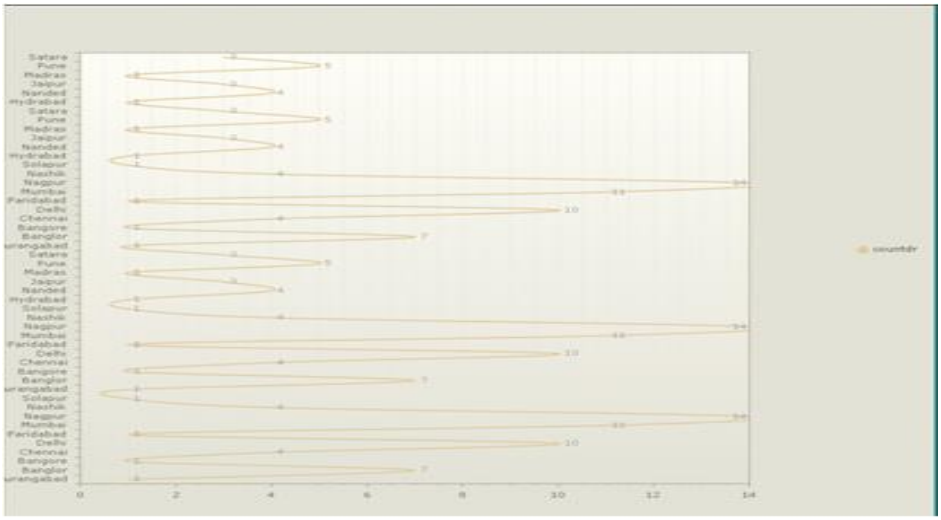

Fig. 2: Experimental results of clustering- cluster1

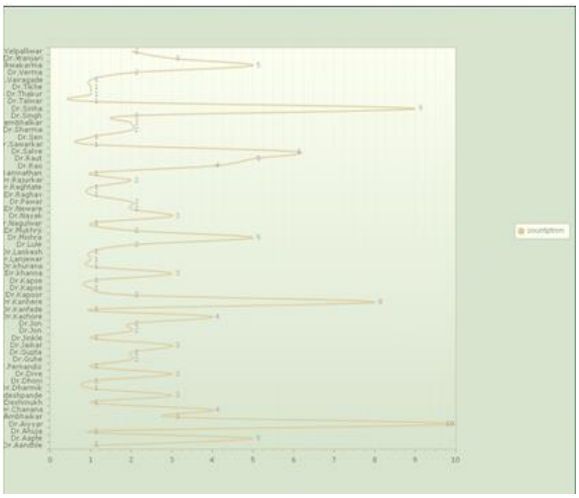

Fig. 3: Experimental results of clustering- cluster2

VI. Conclusion

By the result of proposed algorithm hubness phenomenon is useful for clustering high dimensional data. Data hubness, as a consequence of high inherent dimensionality, is a phenomenon of great importance for nearest-neighbor classification. K-nearest neighbor algorithm potentially helpful for finding hubs in high dimensional data. By applying hybrid approach (i.e DBSCAN and hierarchical) over the hubs clustering become efficient. It is accurate, deterministic, robust to noise, does not require the number of clusters as an input parameter, does not perform distance calculation and it is able to detect clusters.

References:

[1] Nenad Tomasev, Milos Radovanovic, Dunja Mladenic, and Mirjana Ivanovic The, "Role of Hubness in Clustering High-Dimensional Data", IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, REVISED JANUARY 2013.

[2] Nenad Tomašev, Dunja Mladenić Jožef "The influence of weighting the K- occurances on Hubness-aware Classification Methods", Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia.

[3] J. E. Keller, M. R. Gray, and J. A. Givens, "A fuzzy k-nearest-neighbor algorithm," in IEEE Transactions on Systems, Man and Cybernetics, 1985, pp. 580–585.

[4] D. Schnitzer, A. Flexer, M. Schedl, and G. Widmer, "Local and global scaling reduce hubs in space," Journal of Machine Learning Research, vol. 13, pp. 2871–2902, October 2012.

[5] D. Franc, ois, V. Wertz, and M. Verleysen, "The concentration of fractional distances," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 7, pp. 873–886, 2007.

[6] M. Radovanovi´c, A. Nanopoulos, and M. Ivanovic, Hubs in space: Popular nearest neighbors in high-dimensional data", Journal of Machine Learning Research.

[7] M. Radovanovic and A. Nanopulous,"Nearest neighbors in high-dimensional data: the emergence and influence of hubs", Proceedings of 26th International Conference on Machine Learning, Montreal, Canada, 2009.

[8] N. Tomasev and M. Radovanovic and D. Mladenić and M. Ivanovic, "Hubness-based fuzzy measures for high-dimensional k nearest - neighbor classification", In Proc. MDLM 2011, 7th International Conf. on Machine Learning and Data Mining. New York. 2011.

[9] N. Tomašev and M. Radovanović and D. Mladenić and M. Ivanović, "A Probabilistic approach to nearest-neighbor classification: Naive Hubness Bayesian kNN", In Proc. CIKM. 2011.

[10] N. Tomašev and D. Mladenić. Nearest neighbor voting in high dimensional data: learning from past occurrences. (under review) .

[11] Ming Zhang and Reda Alhajj "Novel Approach for Nearest Neighbor Search in High Dimensional Space"2008 4th International IEEE Conference "Intelligent Systems".

[12] Nenad Tomasev, Raluca Brehar, Dunja Mladenic and Sergiu Nedevschi ,"The Influence of Hubness on Nearest-Neighbor Methods in Object Recognition",978-1-4577-1481-8/11/$26.00 ©2011 IEEE.

[13] Nenad Tomašev, Dunja Mladenić, "EXPLORING THE HUBNESS-RELATED PROPERTIES OF OCEANOGRAPHIC SENSOR DATA", Jožef Stefan Institute, Jamova 39, 1000 Ljubljana, Slovenia.

[14]C. Ding and X. He, "K-nearest-neighbor consistency in data clustering: Incorporating local information into global optimization," in Proc. ACM. Symposium on Applied Computing (SAC), 2004, pp. 584–589.

[15] HansPeter Kriegel, Peer Kr¨oger, Arthur Zimek, "Detecting Clusters in ModeratetoHigh Dimensional Data: Subspace Clustering, Patternbased Clustering, and Correlation Clustering", VLDB '08, August 2430, 2008, Auckland, New Zealand.

[16] R. J. Durrant and A. Kab´an, "When is 'nearest neighbour' meaningful: A converse theorem and implications," *Journal of Complexity*, vol. 25, no. 4, pp. 385–397, 2009.

[17] Arthur Flexer, Dominik Schnitzel, Jan Schluter, "A MIREX META-ANALYSIS OF HUBNESS IN AUDIO MUSIC SIMILARITY", Austrian Research Institute for Artificial Intelligence (OFAI), Vienna, Austria.

[18] Eui□Hong _Sam_ Han, George Karypis, Vipin Kumar, Bamshad Mobasher, "Hypergraph Based Clustering in High Dimensional Data Department of Computer Science and Engineering Army HPC Research Center. University of Minnesota.

[19] Andrew McCallum, Kamal Nigam, Lyle H. Ungar, "Efficient Clustering of High Dimensional Data Sets with Application to Reference Matching".

[20] Anoop Kumar Jain, Prof. Satyam Maheswari," *Survey of Recent Clustering Techniques in Data Mining"* International Journal of Computer Science and Management Research Vol 1 Issue 1 Aug 2012 ISSN 2278-733X.

[21] GuiBin Hou,RuiXia Yao, Jiadong Ren, Changzhen Hu," Irregular Grid-Based Clustering over High-Dimensional Data Streams", Pervasive Computing Signal Processing and Applications(PCSPA),2012 First International Conference on 17-19 Sept. 2010, Pages: 783 – 786.

[22] P.-N. Tan, M. Steinbach, and V. Kumar, Introduction to Data Mining. Addison Wesley, 2005.

[23]A. Kab´an, "Non-parametric detection of meaningless distances in high dimensional data," Statistics and Computing, vol. 22, no. 2, pp. 375–385, 2012.

