IJCSE International Journal of Computer Sciences and Engineering Open Access
Survey Paper Volume-5, Issue-1 E-ISSN: 2347-2693

Relaxed Constraints Formulation for Non-linear Distance Metric Learning in Hierarchical Clustering

Akash Mhetre[1*], V.S. Gaikwad[2]

[1*]Department of Computer Engineering, Rajshree Shahu School of Engineering and Research, JSPM NTC, Pune, INDIA
[2]Department of Computer Engineering, Rajshree Shahu School of Engineering and Research, JSPM NTC, Pune, INDIA

Available online at: www.ijcseonline.org

Received:25/12/2016 Revised: 08/01/2017 Accepted: 22/01/2017 Published: 31/01/2017

Abstract— **The process of Learning distance function over different objects is called as Metric learning. In various of data mining processes like clustering, nearest neighbours etc. is very important problem that relies on distance function. For many types of data, linear model is not very useful but most of metric learning methods assumes linear model of distance. In the recent nonlinear data demonstrated potentialpower of non-Mahalanobis distance function, particularly tree-based functions. This leads to a more robust learning algorithm. We compare our method to a number of state-of-the-art benchmarks on k-nearest neighbour classification, large-scale image retrieval and semi supervised clustering problems. Then we find that our algorithm yields results comparable to the state-of-the-art. A novel tree-based non-linear metric learning method can have information from both constrained and unconstrained points. And hierarchical nature of training can minimize the constraint satisfaction problem as it won't have to go through the constraint satisfaction process per object but per hierarchy. Combining the output of many of the resulting semi-random weak hierarchy metrics and by introducing randomness during hierarchy training, we can obtain a powerful and robust nonlinear metric model.**

Keywords- Similarity Measures, Clustering, , Image Retrieval, Classification, Data Mining,Constrained And Unconstrained Point

I. INTRODUCTION

Many elemental data mining problems nearest neighbour classification, retrieval, clustering is at their core dependent on the availability of an effective measure of pair wise distance. A wide range of methods have been proposed to address this learning problem, linear methods have primarily benefited from two advantages. Firstly, they are generally easier to optimize, allowing for faster learning. Second they allow the original. Data to be easily projected into the new metric space, Metric learning is process of learning distance function over different objects. It is very important problem in data mining as various processes like clustering, nearest neighbours etc. relies on distance function. Most of metric learning methods assumes linear model of distance. But for many types of data, linear model is not very useful.

A wide range of methods have been proposed to address the learning problem, but the field has traditionally been dominated by algorithms that assume a linear model of distance, specially Mahalanobis metrics [2]. Linear methods have primarily benefited from two advantages. First, they are generally easier to optimize, allowing for faster learning and in many cases a globally optimal solution to the proposed problem[6] Second, they allow the original data to be easily projected into the new metric space, meaning the metric can be used in conjunction with other methods that operate only on an explicit feature representation. This methodology provides two significant contributions: first, unlike previous tree-based nonlinear metrics, it is semi-supervised, and incorporate information from both constrained and unconstrained points into the learning algorithm. This is an important advantage in many problem settings, particularly when scaling to larger datasets where only a small proportion of the full pair wise constraint set can realistically be collected or used in training. Second, the iterative, hierarchical nature of the training process allows to relax the constraint satisfaction problem. Rather than attempting to satisfy every available constraint simultaneously, at each hierarchy node optimize an appropriate constraint subset to focus on, leaving others to be addressed lower in the tree. By selecting constraints in this way, we can avoid situations where attempting to satisfy incoherent constraints [11], and thereby better model hierarchical data structures.

II. RELATED WORK

D. M. Johnson, C. Xiong and J. J. Corso proposed obtains more powerful metric model with the help of iterative hierarchical variant of semi supervised max-margin clustering [1]. A. Bellet, A. Habrard, and M. Sebban proposed that Recent trends and extensions, such as semi-supervised metric learning, metric learning for histogram data and the derivation of Generalization guarantees, are also covered [2]. J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S.

Dhillon, proposed that this method can handle a wide variety of constraints and can optionally incorporate a prior on the distance function. Also it is fast and scalable [3]. C.Shen, J. Kim, L. Wang, and A. van den Hengel, proposed that one of the primary difficulties in learning such a metric is to ensure that the Mahalanobis matrix remains positive semi definite. Semi definite programming is sometimes used to enforce this constraint, but does not scale well [4]. J. Blitzer, K. Q. Weinberger, and L. K. Saul, prosed the metric is trained with the goal that the k-nearest neighbours always belong to the same class while examples from different classes are separated by a large margin.[5]. Y.Ying and P. Li proposed that the framework not only provides new insights into metric learning but also opens new avenues to the design of efficient metric learning algorithms. Indeed, first-order algorithms are developed for DML-eig and LMNN which only need the computation of the largest eigenvector of a matrix per iteration [6]. R. Chatpatanasiri, T. Korsrilabutr, P. Tangchanachaianan, and B. Kijsirikul proposed that developing a new framework of kernelizing Mahalanobis distance learners. The new KPCA trick framework offers several practical advantages over the classical kernel trick framework [7]. S. Chopra, R. Hadsell, and Y. LeCun, proposed that a function that maps input patterns into a target space such that the norm in the target space approximates the "semantic" distance in the input space. The method is applied to a face verification task [8]. A. Frome, Y. Singer, and J. Malik, proposed that a distance function for each training image as a combination of elementary distances between patch-based visual features [9]. K. Q. Weinberger and L. K. Saul proposed that extended the original framework for LMNN classification in several important ways: by describing a solver that scales well to larger data sets, by integrating metric ball trees into the training and testing procedures [10].

III. EXISTING SYSTEM

The Mahalanobis distance is a parameter to calculate the distance between a point P and a distribution D, introduced by P. C. Mahalanobis in. It is a multi-dimensional generalization of the idea of measuring how many standard deviation away P is from the mean of D. The distance is zero if P is at the mean of D, and increases as P moves away from the mean: along each principal component axis, it calculates the number of standard deviations from P to the mean of D. If each of these axis is rescaled to have unit variance, then Mahalanobis distance corresponds to standard Euclidean distance in the transformed space. Mahalanobis distance is unit-less and scale-invariant, and takes into account the correlations of the data set.The full hierarchy forest distance is effectively the mean of a number of weak distance functions Ht, each corresponding to one hierarchy in the forest. These distance functions, in turn, are representations of the structure of the individual hierarchies—moreover the apart of two instances fall within a hierarchy, the greater the distance between them.

Learning a Mahalanobis distance is equivalent to learning a linear map, the in ability to learn a non-linear transformation is one important limitation of the learners .As their search in Mahalanobis distance learning has just recently begun, several issues are left open such as

(1) some efficient learners doesn't have non-linear extensions,(2)the kernel trick, a standard non-linearization method , is not fully automatic in the sense that new mathematical formulas were to be derived and codes have to be implemented; this is not convenient to non-experts, and (3) the problem of how to select an efficient kernel function has been left untouched in the previous works; in previous works, the best kernel function is achieved via a brute-force method such as cross validation.

System Overview

A. System Model

In existing system novel tree-based non-linear metric learning method is proposed. This method can have information from both constrained and unconstrained points. It also has hierarchical nature of training can minimize the constraint satisfaction problem as it won't have to go through the constraint satisfaction process per object but per hierarchy.

B. Algorithms for Existing System

Algorithm 1: HFD Learning

HFD is conceptually distinct from random forests in that the individual components of the forest represent cluster hierarchies rather than decision trees. HFD also differs from the common form of random forest in that it doesn't do bootstrap sampling on its training points and its splitting functions are linear combinations rather than single-feature thresholds.

Algorithm1: HFD Learning

t=0
Step I:
for t < T do
Step II:
function BuildTree(t,l,X^0,L_0)
w_{tl}←LearnSplit (X_{tl}, L_{tl})
Divide Xtl among c_L and c_R using,
$P_{tl}(x)= w_{tl}^T [\ x_j{}^K{}_{tl}\ 1]$
Send x to left if $P_{tl}(x)<=0$
Send x to right if $P_{tl}(x)>0$
if$|X_{tcL}|$ > minimize then
Use X_{tcL} todetermineL_{tcL}
BuildTree(t, c_L, X_{tcL} , L_{tcL})

end if
if$|X_{tcR}|$ > minsize then
Use X_{tcR} to determine L_{tcR}
BuildTree (t, c_R, X_{tcR} , L_{tcR})
end if
end function
Step III:
Function LearnSplit(X_{tl}, L_{tl})
Select split features K_{tl} and build $X^K{}_{tl}$ via
$K_{tl}{}^K = \{ [x_j{}^{Ktl} \ 1] \mid x_j € X_{tl} € \}$
if $L^c$is not null then
Use CCCP to solve w_{tl}
else
Use block-coord for solving w_{tl}
end if
return w_{tl}
end function

Algorithm 2: HFD Inference:

Metric inference on learned HFD structures is straight forward. We feed two points to the metric and track their progress down each tree. At each node, compute associated binary linear discriminates (for root info).

Algorithm 2: HFD Inference
Step I:
function INFERDISTANCE(a, b)
t = 0
D = 0
for t < T do
D = D + TREEDISTANCE (t, 0, a, b)
t = t + 1
end for
return D/T
end function
Step II:
function TREEDISTANCE (t, l, a, b)
Retrieve split features K_{tl} and build a^{Ktl} and b^{Ktl}
Apply following formulae to a^{Ktl} and b^{Ktl} to get $S_{tl}(a)$ and $S_{tl}(b)$
$P_{tl}(x)= w_{tlT} [xj^{Ktl} \ 1]$
Send x to left if $P_{tl}(x)<=0$
Send x to right if $P_{tl}(x)>0$
if $S_{tl}(a) = S_{tl}(b)$ then
return TreeDistance(t,$S_{tl}(a)$,a,b)
else
return output of
$H_t(a,b)= \{ 0pt(a,b).|H_{tl}(a,b)| / N$

Algorithm 3: Fast Approximate HFD Nearest Neighbors:

Comparing to a euclidean or even Mahalanobis distance. This is worsened, for many applications, by the unavailability of traditional fast approximate nearest-neighbor methods, which require an explicit representation of the data in the metric space in order to function. We address the latter problem by introducing our own fast approximate nearest-neighbor process, which takes advantage of the tree-based structure of the metric to greatly reduce the number of pair-wise distance computations needed to compute a set of Nearest-neighbors for a query point x.

Algorithm 3: Fast Approximate HFD Nearest Neighbors
T=0
Step I:
For t < T do
O=O ∪ [TREENEIGHBOURS (t;0;a)]
t=t+1
end for
Step II:
for x belongs to final candidate neighbor set do
INFREDISTANCE (x; a)
end for
Step III:
return the k points in final candidate neighbor set with the smallest distance
function TREENEIGHBOURS (t; l; a)
Step IV:
if l is a leaf node then
$O_{tl} = k_O$ points sampled from l
if $|X_{tl}|$ < ko then
Sample from parent node(s) as needed
end if
return O_{tl}
else
Retrieve split features Ktl and build a Ktl
Apply (6) to a^{Ktl} to get S_{tl} (a)
return TREENEIGHBORS (t, S_{tl}(a), a)
end if
end function

IV. PROBLEM STATEMENT

To implement a Novel tree based Non-linear metric method with improved performance.

V. PROPOSED SYSTEM

A novel semi-supervised nonlinear distance metric learning procedure based on forests of cluster hierarchies constructed via an iterative max margin clustering procedure. A novel relaxed constraint formulation for max-margin clustering which improves the performance of the method in hierarchical problem settings. A novel in-metric approximate nearest-neighbour retrieval algorithm for our method that greatly decreases retrieval times for large data with little reduction in accuracy.

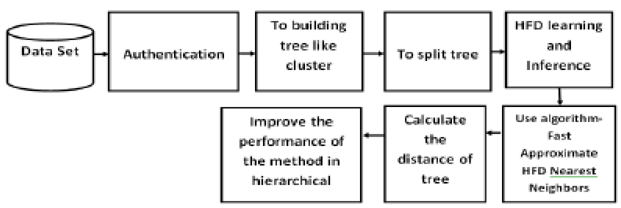

Fig1: Proposed System Architecture Diagram

VI. CONCLUSION

A novel semi-supervised nonlinear distance metric learning procedure based on forests of cluster hierarchies. These forests of cluster hierarchies constructed via an iterative max-margin clustering procedure. This paper presents a semi-supervised metric learning method based on forest of cluster hierarchies. The novel relaxed constraint formulation for max-margin clustering improves the performance of the method in hierarchical problem settings. The proposed algorithm can improve the performance. The algorithm shows that it can compete with currently implemented methods on small as well as large scale datasets. The nearest-neighbor retrieval algorithm that reduces the retrieving time on large dataset with small compromise with accuracy.

REFERENCES

[1] D. M. Johnson, C. Xiong and J. J. Corso, "Semi-Supervised Nonlinear Distance Metric Learning via Forests of Max-Margin Cluster Hierarchies," vol. 28, no. 4, pp. 1035-1046, April 1 2016.

[2] A. Bellet, A. Habrard, and M. Sebban, “A survey on metric learning for feature vectors and structured data,” arXiv preprint arXiv:1306.6709, 2013.

[3] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, “Information theoretic metric learning,” in Proc. 24th Int. Conf. Mach. Learn., 2007, pp. 209–216.

[4] C. Shen, J. Kim, L. Wang, and A. van den Hengel, “Positive semidefinite metric learning with boosting,” in Proc. Adv. Neural Inf. Process. Syst., 2009, pp. 1651–1660.

[5] J. Blitzer, K. Q. Weinberger, and L. K. Saul, “Distance metric learning for large margin nearest neighbor classification,” in Proc. Adv. Neural Inf. Process. Syst., 2005, pp. 1473–1480.

[6] Y. Ying and P. Li, “Distance metric learning with eigen value optimization,” J. Mach. Learn. Res., vol. 13, pp. 1–26, 2012.

[7] R. Chatpatanasiri, T. Korsrilabutr, P. Tangchanachaianan, and B. Kijsirikul, “A new kernelization framework for Mahalanobis distance learning algorithms,” Neurocomputing, vol. 73, no. 10, pp. 1570–1579, 2010.

[8] S. Chopra, R. Hadsell, and Y. LeCun, “Learning a similarity metric discriminatively, with application to face verification,” in Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recog.,2005, pp. 539–546.

[9] A. Frome, Y. Singer, and J. Malik, “Image retrieval and classification using local distance functions,” in Proc. Adv. Neural Inf. Process. Syst., 2006, pp. 417–424.

[10] K. Q. Weinberger and L. K. Saul, “Fast solvers and efficient implementations for distance metric learning,” in Proc. 25th Int. Conf. Mach. Learn., 2008, pp. 1160–1167.

Author’s Profile

Mr. Akash N. Mhetre, received the Bachelor of Engineering Degree in Computer Science & Engineering form, Dr, D Y Patil College of Engineering and technology,Kolhapur, India in year 2015. He is currently pursuing Master of Engineering from Rajshree Shahu School of Engineering and Research, JSPM NTC, Pune, India. His main research work focuses on Data Mining.

Mr. Vilas S. Gaikwad, received the BE Degree in Computer Science & Engineering from the Dr.BAMU Aurangabad, the M.Tech degree in Computer Science & Engineering from Walchand College of Engineering(An autonomous Institute), Sangli. He is currently working as Assistant Professor in the Department of Computer Engineering, Rajshree Shahu School of Engineering and Research, JSPM NTC, Pune, India. His research area includes Image Processing and Computer Network.

