Journal of Computer Science

Original Research Paper

Prototype-Based Sample Selection for Active Hashing

Cheong Hee Park

Department of Computer Science and Engineering, Chungnam National University, Daejeon, Korea

Article history
Received: 02-09-2014
Revised: 26-10-2015
Accepted: 29-10-2015

E-mail: cheonghee@cnu.ac.kr

Abstract: Several hashing-based methods for Approximate Nearest Neighbors (ANN) search in a large data set have been proposed recently. In particular, semi-supervised hashing utilizes semantic similarity given for a small fraction of pairwise data samples and active hashing aims to improve the performance for ANN search by relying on an expert for the labeling of the most informative points. In this study, we present an active hashing method by prototype-based sample selection. Knowing semantic similarities between cluster prototypes can help extracting relations among the points in the corresponding clusters. For expert labeling, we select prototypes from clusters which do not contain any data points with labeled information so that all areas can be covered effectively. Experimental results demonstrate that the proposed active hashing method improves the performance for ANN search.

Keywords: Active Hashing, Approximate Nearest Neighbors (ANN) Search, Hierarchical Clustering, Prototype-Based Sample Selection, Semi-Supervised Hashing

Introduction

As a huge size of data collection becomes easier to obtain, efficient methods for nearest neighbors search are needed in various areas such as data mining and pattern recognition (Shakhnarovich *et al*., 2006). However, most of algorithms for exact nearest neighbors search usually require high memory and time complexity (Indyk and Motwani, 1998; Muja and Lowe, 2009). Fortunately, returning Approximate Nearest Neighbors (ANN) is acceptable in many applications and several hashing based methods for ANN search have been proposed recently. In hashing-based methods, by mapping data points to k-bit binary codes, nearest neighbors are searched in a binary embedding space. Locality Sensitive Hashing (LSH) is one of the primitive hashing based ANN search methods, which uses random projections followed by a random threshold (Gionis *et al*., 1999). Unlike data-independent hashing in LSH, several data dependent hashing methods including Spectral Hashing (SH) (Weiss *et al*., 2008) and Binary Reconstructive Embedding (BRE) (Kulis *et al*., 2009) learn hash functions from training data so that similar data points in the original space are mapped to near points in the binary embedding space.

Semi-supervised hashing utilizes semantic similarity which is given in terms of two categories of relations for a fraction of pairwise data samples: Must-link and cannot-link (Wang *et al*., 2012; Mu *et al*., 2010). Must-link relation means that two data samples are neighbors or have the same class label. Similarly, cannot-link implies that two samples are far away or have different class labels. By enforcing hash functions to reduce the empirical errors on semantic relations, semi-supervised hashing can improve hashing performance (Wang *et al*., 2012). In (Zhen and Yeung, 2013), a method for active hashing was proposed, where the most informative points are chosen for active labeling by an expert. The authors measured the informativeness of data points by using a distance from the thresholding boundary. Also a batch of data points can be selected by searching for a data subset which minimizes both data certainty and pairwise similarities.

In this study, we present an active hashing method by prototype-based sample selection. Assuming clusters and their cluster prototypes are found, it is well known that prototypes can be used to find nearest neighbors efficiently (Tan *et al*., 2014). If two cluster prototypes are far apart, the points in the corresponding clusters cannot be nearest neighbors of each other. A prototype can be considered as a representative of the points belonging to the cluster, since points are relatively close to the prototype of their cluster. Hence, knowing semantic similarities between cluster prototypes should help extracting valuable information about the points in the corresponding clusters. For expert labeling, we select prototypes from the clusters which do not contain any data points with labeled information so that all areas can be covered.

Science Publications

In section II, some hashing methods are reviewed. In Section III, an active hashing method by prototype-based sample selection is proposed. In section IV, experimental results are given. Discussions follow in section V.

Related Works

Locally Sensitive Hashing (LSH)

The key idea of Locality Sensitive Hashing (LSH) is to hash the points using several hash functions so that the probability of collision is much higher for points which are close to each other than for those which are far apart (Gionis *et al*., 1999). LSH assumes that data points follow p-stable distribution such as Gaussian distribution and hash functions are constructed by random projections which are sampled randomly from *p*-stable distribution. However, LSH using random projections requires long binary codes to achieve high precision resulting in row recall.

Semi-Supervised Hashing (SSH)

Usually the similarity of data points in the original data space is measured by a metric distance. Semi-supervised hashing (Wang *et al*., 2012) utilizes high-level semantic similarity for hash function learning. Semantic similarity is given as relations of two types: must-link relation *M* and cannot-link relation *C*. A pair (x_i, x_i), ∈M means two data samples are neighbors in a metric space and (x_i, x_i)∈C implies that two samples are far away in a metric space. An objective function for linear projection based hash functions $H = \{h_l(x) = sign(w_l^T x + b_l) 1 \le l \le k\}$ which reduce the empirical errors on semantic information and maximize the variance of hash values is described as:

$$\max imize \sum_l \{ \sum_{(x_i, x_j) \in M} h_l(x_i) h_l(x_j) \sqrt{a^2 + b^2} - \sum_{(x_i, x_j) \in C} h_l(x_i) h_l(x_j) + \eta\, var_x [h_l(x)] \} \tag{1}$$

Assuming the data points have zero mean, Equation 1 is reformulated as a problem to find a projection matrix W = [w_1,..,w_k] satisfying:

$$\begin{aligned} & argmax_w J(W) \\ & = argmax_w \frac{1}{2} trace\{W^T (X_l S X_l^T + aXX^T) W\} \end{aligned} \tag{2}$$

X = [x_1,..,x_n] is a data matrix and X_l is a sub-matrix of *X* where *l* data points are associated with at least one of categories *M* and *C* and *S* is defined as:

$$S_{i,j} = \begin{matrix} 1 : (x_i, x_j) \in M \\ -1 : (x_i, x_j) \in C \\ 0 : otherwise \end{matrix} \tag{3}$$

The k eigenvectors {φ_1,...,φ_k} corresponding to the largest eigenvalues of $X_l S X_l^T + aXX^T$ solve Equation 2, giving orthogonal projection vectors. The original data space is projected by using the eigenvectors and thresholding by the mean in the projected space gives a binary embedding space where nearest neighbor search by the hamming distance can be performed efficiently. As an alternative approach, sequential projection learning can be applied where projection vectors are learned sequentially by updating the pairwise label matrix iteratively and trying to correct errors made by the previous one.

Active Hashing

Active learning has been mainly researched under the goal to improve classification performance (Settles, 2009). When the number of labeled data samples is very small, by acquiring the exact class labels of selected unlabeled data samples from an expert, active learning enlarges the size of a labeled data set on which a new classifier is trained. This process is repeated by moving selected unlabeled data samples to a labeled data set (Li and Sethi, 2006; Freund *et al*., 1997). In order to select the data sample that will be the most helpful for classification, active learning selects either the most difficult data samples for the current classifier (Freund *et al*., 1997) or the most informative data sample that maximizes some expected gain in classification (Zhu and Lafferty, 2003).

When semantic similarity is given only for a small portion of pairwise data points, active hashing pursues active learning in hashing (Zhen and Yeung, 2013). The most informative points are chosen for semantic relationship labeling by an expert and labeled data points are added to X_l in semi-supervised hashing. Assuming that the data points have zero mean, the certainty of a data point with respect to hash functions $\{h_i(x) = sign(w_i^T x) | i = 1,...,k\}$ is measured by:

$$f(H, x) = \|W^T x\|_2 \tag{4}$$

where W = [w_1,...,w_k]. Small value in f(H,x) implies less certainty or more informativeness of x, in other words, more valuable information can be obtained by active labeling.

A batch of informative data points is selected by searching for a data subset which minimizes both data certainty and pairwise similarities such as:

$$\begin{aligned} & min_\mu \mu^T \widehat{f} + \frac{\beta}{m} \mu^T K \mu \\ & such\, that\, \mu \in \{0,1\}^{|Y|}, \mu^T 1 = m \end{aligned} \tag{5}$$

Here $\hat{f}$ is a vector of normalized certainty values of the points in a candidate set *Y*, *μ* is an indicator vector for selection of the points, *m* is the number of points to be selected and *K* is a positive semi-definite similarity matrix defined on *Y*. Equation 5 is solved by relaxing the condition $\mu \in \{0,1\}^{|y|}$ to the continuous constraint $0 \le \mu \le 1$ and using quadratic optimization programming.

Active Hashing By Prototype Based Sample Section

A prototype is a representative of the members in a cluster, meaning that knowing characteristics of a prototype gives information about other members in the cluster. Cluster prototypes which can be found by an efficient clustering algorithm can be optimal candidates for sampling to reflect the data distribution. Figure 1 illustrates prototype-based sample selection. Three data points associated with two relations M and C are drawn with filled circles in Fig. 1a. For which area is given insufficient information and which points could give valuable information if they are selected for manual labeling? A dendrogram by hierarchical clustering which has seven clusters as leaves is shown in Fig. 1b. When we want to select four samples for active labeling, as shown in Fig. 1c, four cluster prototypes marked with the notation x can be selected. Four prototypes are selected from four clusters which do not contain any labeled data point within their clusters.

In order to select *m* prototypes whose clusters do not contain any labeled data point, we need a clustering algorithm which allows more than *m* clusters to be flexibly examined. For that reason, we use an agglomerative hierarchical clustering method based on Ward linkage (Ward, 1963). Ward linkage computes the distance between clusters as the increase in the squared error that results when two clusters are merged. Then the distance between two clusters *r* and *s* is equivalent to the following distance measure:

$$d(r,s)=\sqrt{\frac{2n_r n_s}{n_r+n_s}}\left\|\bar{x}_r-\bar{x}_s\right\|_2 \qquad (6)$$

where $\bar{x}_r$ and $\bar{x}_s$ are the centroids of clusters *r* and *s* and n_r and n_s are the number of elements in clusters *r* and *s*. After performing hierarchical clustering, *m* clusters which do not include any labeled data points are searched. Starting from the top in the dendrogram, we go downwards until the cutting line in the dendrogram is found which produces m clusters containing no labeled data points. Then cluster prototypes are selected from *m* clusters. However, since the centroid is not the real data point in a cluster, we set a cluster prototype as the data point which is the closest to the cluster centroid.

Generally, hierarchical clustering requires $O(n^3)$ time complexity where *n* is the number of data points and it can be reduced to $O(n^2 \log n)$ with the help of efficient data structures. Since hashing is considered as a useful technique for a large size of database, it is not recommended to perform hierarchical clustering with all the data points. Instead, a candidate set which is composed of a small size of data points is constructed and clustering is performed over the candidate set together with a given set of labeled data points.

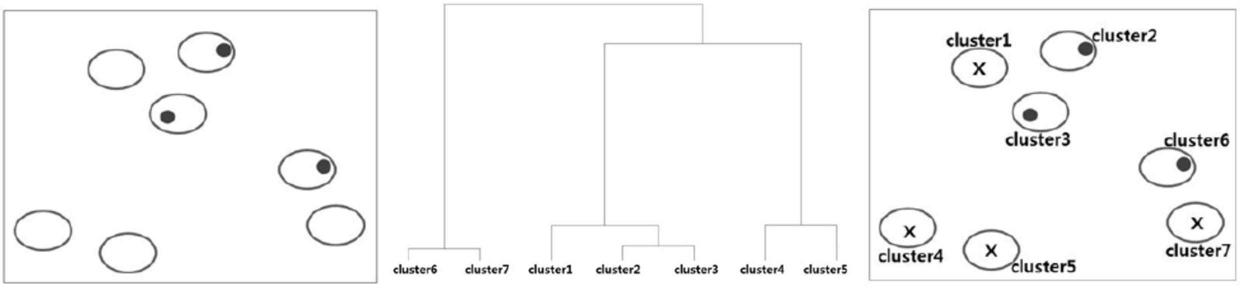
The proposed active hashing algorithm is summarized in Table 1. Given a data set X and a subset X_l of X where the data samples in X_l are associated with *M* and *C*, projective vectors $\Phi = \{w_1,...,w_k\}$ are learned by a semi-supervised hashing method. For active sample selection, we perform hierarchical clustering based on Ward linkage over a candidate set Y and X_l in the projected space by $\Phi = \{w_1,...,w_k\}$. Find the cutting line in the dendrogram which produces m clusters containing no data points of X_l and select cluster prototypes in those clusters. Note that cluster prototype is found in the original space although clustering is done in the projected space.

Active hashing and active learning have different goals and various hash function learning algorithms and classifier modeling algorithms are used (Refer to Fig. 2). The prototype-based sample selection approach does not depend on specific learning algorithm and it is applicable for both active hashing and active learning. The application results for active learning in classification can be found in (Woo and Park, 2013).

Table 1. The proposed active hashing method

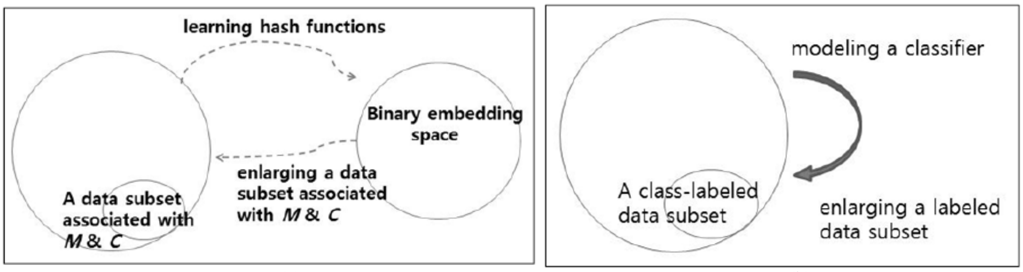
```
Algorithm 1. An active hashing algorithm
X: A set of data points
$X_l \subset X$: A set of data points associated with $M$ and $C$
  for t = 1, ···,
 Learn projective vectors $\Phi = \{w_1,\ldots,w_k\}$ by the
   semi-supervised hashing method in section 2.2
 Perform ANN search for a query in the binary
   embedding space obtained by the projection by
   $\Phi$ and mean thresholding
 if the budget is allowed for active hashing
    Make a candidate set Y by randomly
      sampling from $X$-$X_l$
    Call algorithm 2 and acquire manual labeling
      for the selected data points
      and add them to $X_l$
  end if
 end for
```

```
 - Algorithm 2. A prototype-based sample selection method
 Input: $X_l$, Y, $\Phi$
          m: The number of data points to be selected
             for active labeling
Perform hierarchical clustering based on Ward linkage over
$\Phi(Y \cup X_l)$
Find the cutting line in the dendrogram which produces m
clusters containing no data points of $X_l$
Find cluster prototypes in m clusters and return them
```



(a) (b) (c)

Fig. 1. Illustration of prototype-based sample selection


learning hash functions
Binary embedding space
enlarging a data subset associated with M & C
A data subset associated with M & C
modeling a classifier
A class-labeled data subset
enlarging a labeled data subset

(a) (b)

Fig. 2. Active hashing vs. active learning (a) active hashing (b) active learning in classification

Experimental Results for Active Hashing

We used two data sets which are publicly available. One is the MNIST data set (MNIST, 1998) which consists of 70,000 images of handwritten digits. They are grey level images of digits from 0 to 9 of size 28×28. Each image is transformed to a 784-dimensional vector using the intensity values of images. The data set is split into a training set of 69000 images and a test set of remaining 1000 images. From the training set, 100 points are sampled for an initial set X_l. The matrix S of Equation 3 is defined with class label information: 1 for the data points in the same class and -1 for the data points in the different class. Using a training set, hash functions are found and all the data are mapped to binary codes. For hash function learning, we used the se sequential projection learning approach in (Wang *et al*., 2012) as in active hashing of (Zhen and Yeung, 2013). Data points in the test set are used as a query and 500 points with the smallest hamming distance from the query point are retrieved from the training set. Precision and recall are computed for each query point and then averaged for all the query points. Since the data set is fully annotated, the ground truth semantic neighbors are defined based on the associated digit labels.

Our proposed method is compared with the active hashing method in (Zhen and Yeung, 2013) and active hashing using random sampling. All the parameters for the active hashing method in (Zhen and Yeung, 2013) are set as used in (Zhen and Yeung, 2013). In the proposed method, 5000 points are randomly sampled as a candidate set for clustering. At each call to a sample selection method, m = 100 data points are selected for active labelling. Random splitting to training and test sets is repeated ten times and the average precisions and recalls are reported in Fig. 3. The numbers in x axis denote the iteration of sample selection from 1 to 20. The active hashing method in (Zhen and Yeung, 2013) was denoted as “AH by Zhen”. The graphs in the left column compare the average precisions and the graphs in the right column show the average recalls. The results using $k = 24, 32, 48$ hash bits are displayed at each raw.

The other data set is 20 newsgroup data set which contains about 20000 articles from 20 newsgroups. The preprocessing was performed using the Rainbow software package. Stemming and stop-list were used and the words were removed that occur less than 50 times. Using TFIDF representation and normalization process, we have 19944 document vectors in 6165 dimensional space. The data set is split into a training set of 19000 images and a test set of remaining 1000 images. For manageable data size, Principal Component Analysis (PCA) was performed as a preprocessing step, reducing dimension to 1000. From the training set, 50 points are sampled for initial labeled set and 50 points are selected for manual labeling at each iteration. Other experimental setting is same as in the first experiment. Figure 4 shows the performance of the compared methods using the average precisions and recalls. Most of the cases, the proposed method shows superior or comparable performance to other methods.

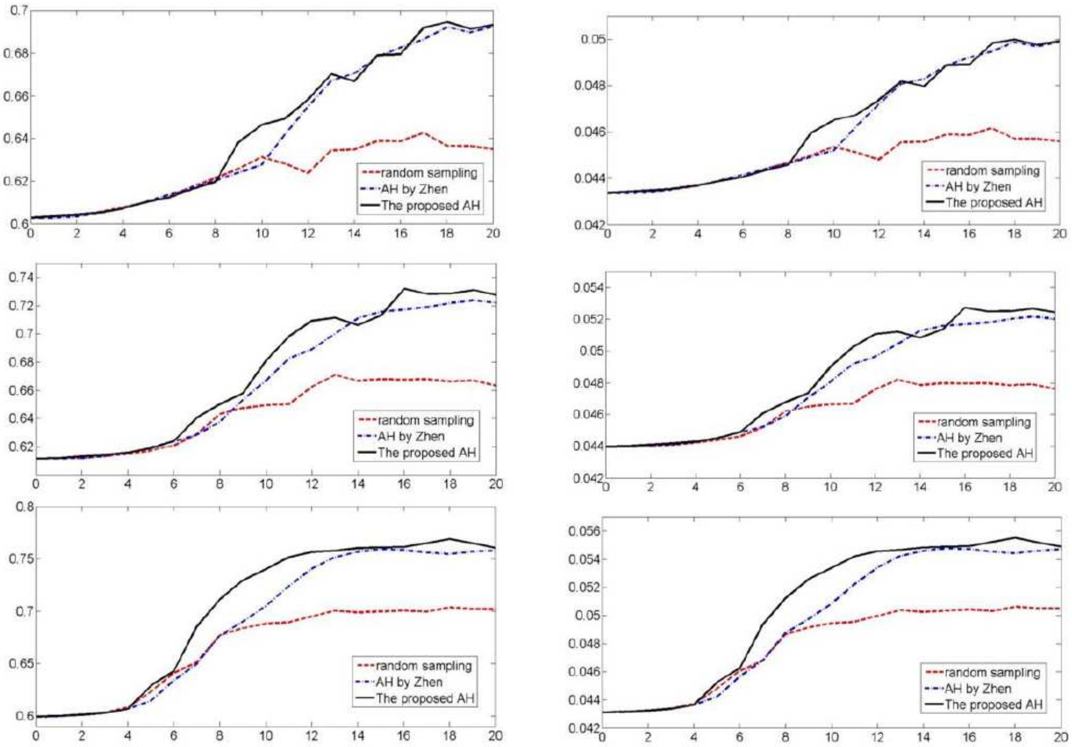


Fig. 3. Performance comparison using MNIST data (left column: precision, right column: recall, the first row: 24 bits embedding, the second row: 32 bits embedding, the third row: 48 bits embedding)

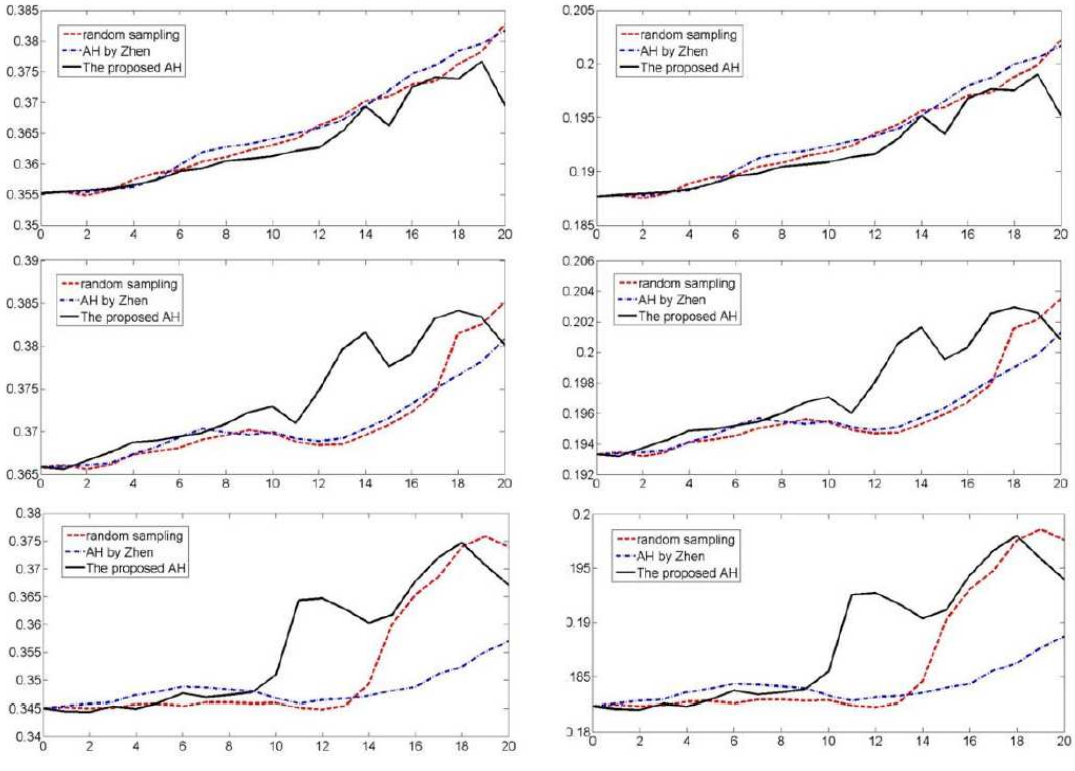


Fig. 4. Performance comparison using 20 newsgroup data (left column: precision, right column: recall, the first row: 24 bits embedding, the second row: 32 bits embedding, the third row: 48 bits embedding)

Conclusion

We proposed a prototype-based sample selection method for active hashing. A prototype is a representative of the members in a cluster and knowing similarity relationship between prototypes can give valuable information about members in the cluster. By selecting prototypes whose cluster do not contain any labeled data points, active hashing can get improved performance.

Acknowledgment

This study was supported by research fund of Chungnam National University in 2014.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

Freund, E.S.Y., H.S. Seung and N. Tishby, 1997. Selective sampling using the query by committee algorithm. Machine Learning, 28: 133-168. DOI: 10.1023/A:1007330508534

Gionis, A., P. Indyk and R. Motwani, 1999. Similarity search in high dimensions via hashing. Proceedings of the 25th International Conference on Very Large Data Bases, pp: 518-529. DOI: 10.1109/2.410146

Indyk, P. and R. Motwani, 1998. Approximate nearest neighbors: Towards removing the curse of dimensionality. Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing, DOI: 10.1145/276698.276876

Kulis, B., T. Darrell and R. Motwani, 2009. Learning to hash with binary reconstructive embeddings. Advances Neural Information Processing Syst., 20: 1042-1050.

Li, M. and I. Sethi, 2006. Confidence-based active learning. IEEE Tran. Patt. Analysis Machine Intelligence, 28: 1251-1261. DOI: 10.1109/TPAMI.2006.156

MNIST, 1998. The MNIST database of handwritten digits.

Mu, Y., J. Shen and S. Yan, 2010. Weakly-supervised hashing in kernel space. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Jun. 13-18, IEEE Xplore Press, San Francisco, pp: 3344-3351. DOI: 10.1109/CVPR.2010.5540024

Muja, M. and D. Lowe, 2009. Fast approximate nearest neighbors with automatic algorithm configuration. Proceedings of International Conference on Computer Vision Theory and Applications, (ICC' 09).

Settles, B., 2009. Active learning literature survey, computer science technical report 1648, university of wisconsin-madison.

Shakhnarovich, G., T. Darrell and P. Indyk, 2006. Nearest-neighbor methods in learning and vision. MIT Press.

Tan, P., M. Steinbach and V. Kumar, 2014. Introduction to Data Mining. 1st Edn., Pearson Education, Harlow, ISBN-10: 1292026154, pp: 736.

Ward, J., 1963. Hierarchical grouping to optimize an objective function. J. Am. Statistical Association, 48: 236-244. DOI: 10.1080/01621459.1963.10500845

Weiss, Y., A. Torralba and R. Fergus. 2008. Spectral hashing. Adv. Neural Inf. Proc. Syst., 21: 1753-1760.

Woo, H. and C.H. Park, 2013. Active learning based on Hierarchical clustering. Trans. Software Data Eng., 2: 705-712. DOI: 10.3745/KTSDE.2013.2.10.705

Wang, J., S. Kumar and S.F. Chang, 2012. Semi-supervised hashing for large-scale search. IEEE Trans. Pattern Analysis Machine Intelligence, 34: 2393-2406.

Zhen, Y. and D.Y. Yeung, 2013. Active hashing and its application to image and text retrieval. Data Mining Knowledge Discovery, 26: 255-274. DOI: 10.1007/s10618-012-0249-y

Zhu, Z.G.X. and J. Lafferty, 2003. Semi-supervised learning using Gaussian fields and harmonic functions. Proceedings of International Conference on Machine Learning, (ICM' 03). pp: 912-919. DOI: 10.1.1.14.4312