Latent Structure Preserving Hashing

Ziyun Cai[1]
cziyun1@sheffield.ac.uk
Li Liu[2]
li2.liu@northumbria.ac.uk
Mengyang Yu[2]
m.y.yu@ieee.org
Ling Shao[2]
ling.shao@ieee.org

[1] Department of Electronic and Electrical Engineering
The University of Sheffield
Sheffield, S1 3JD, UK

[2] Computer Vision and Artificial Intelligence Group
Department of Computer Science and Digital Technologies
Northumbria University
Newcastle upon Tyne, NE1 8ST, UK

Aiming at efficient similarity search, hash functions are designed to embed high-dimensional feature descriptors to low-dimensional binary codes such that similar descriptors will lead to the binary codes with a short distance in the Hamming space. It is critical to effectively maintain the intrinsic structure and preserve the original information of data in a hashing algorithm. In this paper, we propose a novel hashing algorithm called Latent Structure Preserving Hashing (LSPH), with the target of finding a well-structured low-dimensional data representation from the original high-dimensional data through a novel objective function based on Non-negative Matrix Factorization (NMF) [2]. Via exploiting the probabilistic distribution of data, LSPH can automatically learn the latent information and successfully preserve the structure of high-dimensional data. After finding the low-dimensional representations, the hash functions can be acquired through multi-variable logistic regression. Experimental results on two large-scale datasets, i.e., **SIFT 1M** and **GIST 1M**, show that LSPH can significantly outperform the state-of-the-art hashing techniques. The outline of the proposed approach is depicted in Fig. 1.

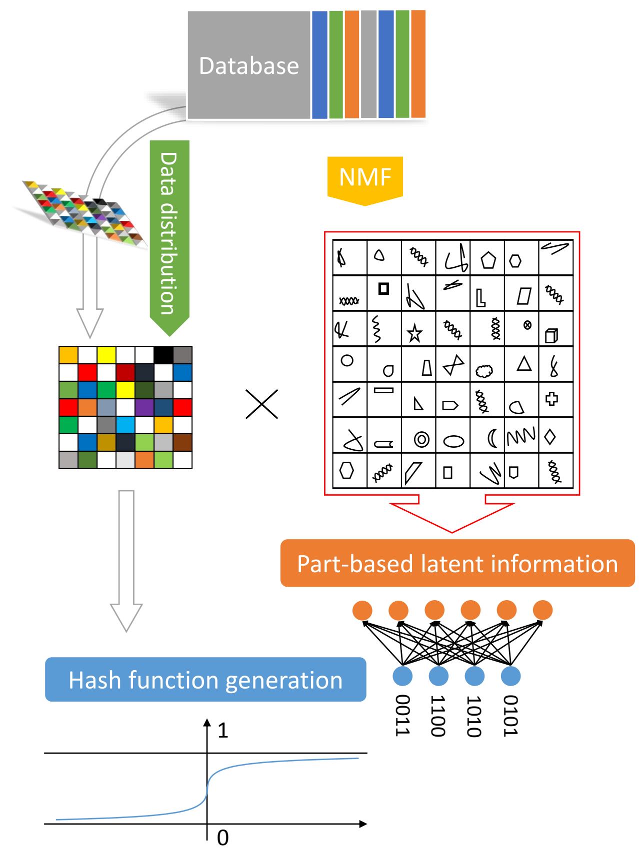

Figure 1: The outline of the proposed method. Part-based latent information is learned from NMF with the regularization of data distribution.

Theoretically, it is expected that the low-dimensional data V given by NMF can obtain locality structure from the high-dimensional data X. We propose to minimize the Kullback-Leibler divergence between the joint probability distribution in the high-dimensional space and the joint probability distribution in the low-dimensional space. Then through combining the data structure preserving part and the NMF technique, we can obtain the following new objective function:

$$O_f = \|X - UV\|^2 + \lambda KL(P\|Q), \tag{1}$$

where P is the joint probability distribution in the high-dimensional space, Q is the joint probability distribution in the low-dimensional space, $V \in \{0,1\}^{D\times N}$, $X, U, V \geqslant 0$, $U \in \mathbb{R}^{M\times D}$, $X \in \mathbb{R}^{M\times N}$, and λ controls the smoothness of the new representation.

Motivated by [3], we first relax the data $V \in \{0,1\}^{D\times N}$ to the range $V \in \mathbb{R}^{D\times N}$ for obtaining real values. The Lagrangian of our problem will be:

$$\mathcal{L} = \|X - UV\|^2 + \lambda KL(P\|Q) + tr(\Phi U^T) + tr(\Psi V^T), \tag{2}$$

where matrices Φ and Ψ are two Lagrangian multiplier matrices. By some algebraic deviations, we have the following update rules for any i, j:

$$V_{ij} \leftarrow \frac{(U^T X)_{ij} + 2\lambda \sum_{k=1}^{N} \frac{p_{jk}V_{ik} + q_{jk}V_{ij}}{1+\|\mathbf{v}_j - \mathbf{v}_k\|^2}}{(U^T UV)_{ij} + 2\lambda \sum_{k=1}^{N} \frac{p_{jk}V_{ij} + q_{jk}V_{ik}}{1+\|\mathbf{v}_j - \mathbf{v}_k\|^2}} V_{ij}, \quad U_{ij} \leftarrow \frac{(XV^T)_{ij}}{(UVV^T)_{ij}} U_{ij}. \tag{3}$$

The proof of convergence about U and V is similar to the ones in [1, 4].

Then we need to convert the low-dimensional real-valued representations from $V = [\mathbf{v}_1, \cdots, \mathbf{v}_N] \in \mathbb{R}^{D\times N}$ into binary codes via thresholding: if the d-th element in $\mathbf{v}_n$ is larger than the specified threshold, this real-value will be represented as 1; otherwise it will be 0, where $d = 1, \cdots, D$ and $n = 1, \cdots, N$. Therefore, with the vector-valued sigmoid function

$$h_\Theta(\mathbf{v}_n) = \left(\frac{1}{1 + e^{-(\Theta^T \mathbf{v}_n)_i}} \right)^T_{i=1,\cdots,D}$$

for the matrix $\Theta \in \mathbb{R}^{D\times D}$, our cost function for the corresponding regression matrix Θ can be defined as:

$$J(\Theta) = -\frac{1}{N}\left(\sum_{n=1}^{N} \left(\hat{\mathbf{v}}_n^T \mathbf{log}(h_\Theta(\mathbf{v}_n)) + (\mathbf{1} - \hat{\mathbf{v}}_n)^T \mathbf{log}(\mathbf{1} - h_\Theta(\mathbf{v}_n)) \right) + \delta\|\Theta\|^2 \right)$$

where $\mathbf{log}(\cdot)$ is the element-wise logarithm function and $\mathbf{1}$ is an $D \times 1$ all ones matrix. We use $\delta\|\Theta\|^2$ as the regularization term in logistic regression to avoid overfitting.

The updating equation is shown as follows:

$$\Theta^{(t+1)} = \Theta^{(t)} - \frac{\alpha}{N} \sum_{n=1}^{N} (h_{\Theta^{(t)}}(\mathbf{v}_n) - \hat{\mathbf{v}}_n)\mathbf{v}_n^T - \frac{\alpha\delta}{N}\Theta^{(t)}. \tag{4}$$

where α is the learning rate. Since h_Θ is a sigmoid function, the hash code for the new coming sample $X_{new} \in \mathbb{R}^{M\times 1}$ can be represented as:

$$\hat{V}_{new} = \lfloor h_\Theta(QX_{new}) \rceil, \tag{5}$$

where $\lfloor \cdot \rceil$ means the nearest integer function for each entry of h_Θ and $Q = (U^T U)^{-1} U^T$ which is the pseudoinverse of U.

[1] Deng Cai, Xiaofei He, Jiawei Han, and Thomas S Huang. Graph regularized nonnegative matrix factorization for data representation. *IEEE TPAMI*, 33(8):1548–1560, 2011.

[2] Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562, 2000.

[3] Yair Weiss, Antonio Torralba, and Rob Fergus. Spectral hashing. In *NIPS*, pages 1753–1760, 2009.

[4] Wenbin Zheng, Yuntao Qian, and Hong Tang. Dimensionality reduction with category information fusion and non-negative matrix factorization for text categorization. In *Artificial Intelligence and Computational Intelligence*, pages 505–512. 2011.