Unsupervised Hashing with Contrastive Information Bottleneck

Presenter: Zijing Ou

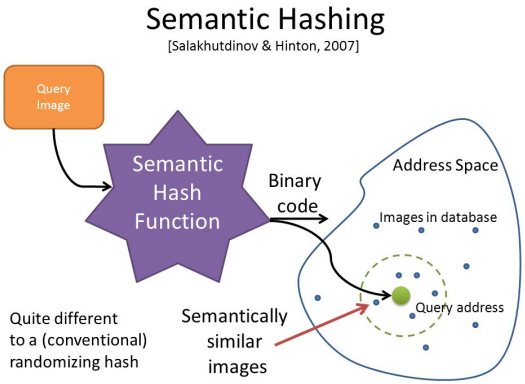
Join work with: Zexuan Qiu, Qinliang Su, Jianxin Yu, Changyou Chen

Sun Yat-sen University & University at Buffalo

June 27, 2021

Discrete Representation Learning

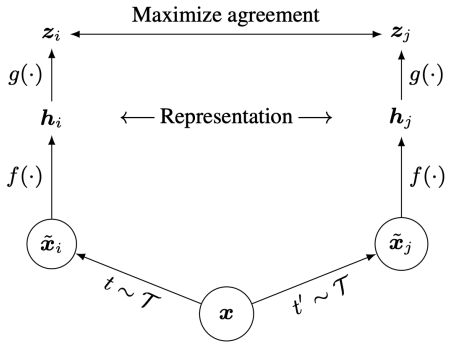
Unsupervised Hashing plays a crucial role in fast and accurate similarity search, thanks to the fast evaluation based on **Hamming distances**.



- Existing methods implicitly established on the paradigm of **auto-encoder**, which basically encourages the hash codes to **reconstruct the input image**, such that retaining as much information of original data as possible.
- This requirement may force the models spending lots of their effort on reconstructing the unuseful background information, while ignoring to preserve the discriminative semantic information.

Contrastive Learning

Contrastive learning (CL) has been recently shown that it is able to produce informatively semantic representations under unsupervised paradigms.



- Given an image $x^{(k)}$ with k denoting its index of a minibatch with size N, CL first transforms it into **two views** $v_1^{(k)}$ and $v_2^{(k)}$, and then feeds them into an encoder network $f_\theta(\cdot)$ to produce **continuous** representation

$$h_i^{(k)} = f_\theta(v_i^{(k)}), \quad i = 1 \text{ or } 2.$$

Contrastive Learning

- The CL framework further projects latent representation $h_i^{(k)}$ into to a new space via a **projection layer**

$$z_i^{(k)} = g_\phi(h_i^{(k)}).$$

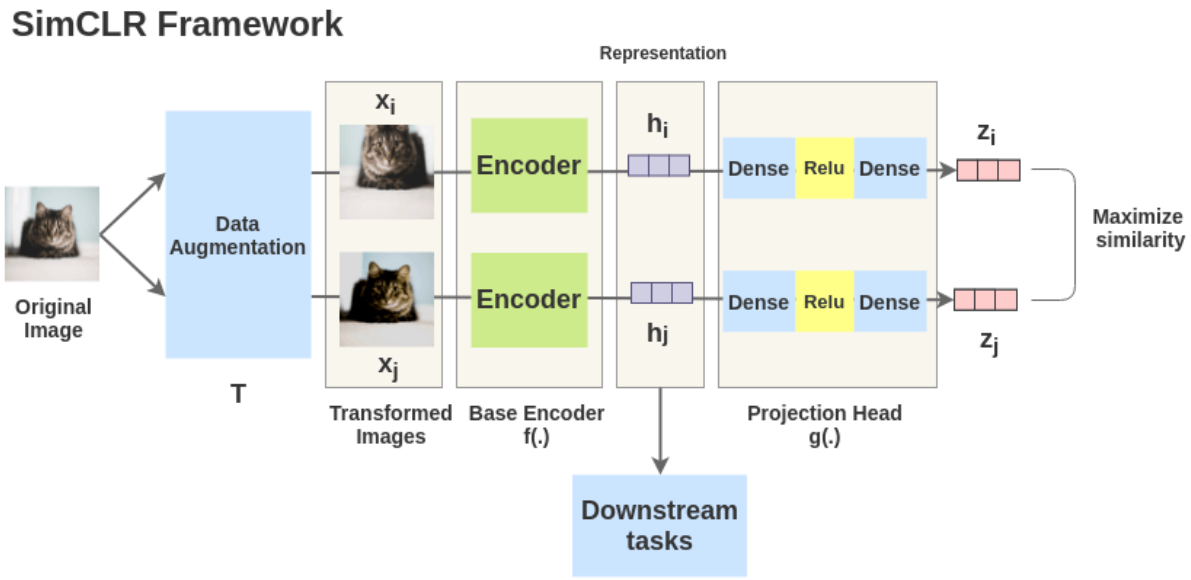
- The learning objective is minimizing the **contrastive loss** on the projected vectors $z_i^{(k)}$

$$L_{cl} = \frac{1}{N}\sum_{k=1}^{N}\left(\ell_1^{(k)} + \ell_2^{(k)}\right),$$

where

$$\ell_1^{(k)} \triangleq -\log \frac{e^{sim(z_1^{(k)}, z_2^{(k)})/\tau}}{e^{sim(z_1^{(k)}, z_2^{(k)})/\tau} + \sum_{i,n\neq k} e^{sim(z_1^{(k)}, z_i^{(n)})/\tau}}.$$

Contrastive Learning



After training, for a given image $x^{(k)}$, we can obtain its representation by feeding it into the encoder network

$$r^{(k)} = f_\theta(x^{(k)}).$$

Motivation & Contribution

- Success so far limited to
  - **Continuous** representations
  - The additional projection head will introduce **noisy fitting**.

- We present **CIBHash**: an hashing framework with **C**ontrastive **I**nformation **B**ottleneck
  - A new objective for **learning discrete** structured representations
  - We establish a connection between the proposed probabilistic hashing method and **mutual information**
  - The proposed contrastive-learning-based hashing method is further considered under the broader **information bottleneck** (IB) principle.
  - State-of-the-art performance on image hashing

Adapting Contrastive Learning to Hashing

- Given a view $v_i^{(k)}$ from the k-th image $x^{(k)}$, we first compute the probability

$$p_i^{(k)} = \sigma(f_\theta(v_i^{(k)})).$$

- Then, the binary codes are generated by sampling from the multivariate **Bernoulli distribution** as

$$b_i^{(k)} \sim Bernoulli(p_i^{(k)}).$$

- We can minimize the expected contrastive loss

$$\bar{L}_{cl} = \frac{1}{N}\sum_{k=1}^{N}\left(\bar{\ell}_1^{(k)} + \bar{\ell}_2^{(k)}\right),$$

where

$$\bar{\ell}_1^{(k)} = -\mathbb{E}\left[\log \frac{e^{sim(b_1^{(k)}, b_2^{(k)})/\tau}}{e^{sim(b_1^{(k)}, b_2^{(k)})/\tau} + \sum_{i, n\neq k} e^{sim(b_1^{(k)}, b_i^{(n)})/\tau}}\right].$$

Reformulating the Contrastive Loss

- We rewrite the minibatch of views $\{v_1^{(k)}, v_2^{(k)}\}_{k=1}^N$ as $\{v_i\}_{i=1}^{2N} \triangleq \mathcal{V}$ and binary codes $\{b_1^{(k)}, b_2^{(k)}\}_{k=1}^N$ as $\{b_i\}_{i=1}^{2N} \triangleq \mathcal{B}$.
- Randomly take one view (*e.g.*, v_k) for consideration. For the rest of $2N-1$ views $\{v_i\}_{i\neq k}$, we assign each of them a unique label from $\{1, 2, \cdots, 2N-1\}$. The target view v_k is assigned the label same as the view derived from the same image.
- Then, we can train a **instanced-based classifier** to maximize the predictive probability

$$\begin{aligned}\ell_{ce}(v_k, y_k) &= -\mathbb{E}_{p(\mathcal{B}|\mathcal{V})}\left[\log \frac{e^{sim(b_k, \mathcal{B}\setminus b_k(y_k))/\tau}}{\sum_{c\in\mathcal{B}\setminus b_k} e^{sim(b_k,c)/\tau}}\right] \\ &\triangleq -\mathbb{E}_{p(\mathcal{B}|\mathcal{V})}[\log q(y_k|b_k)].\end{aligned}$$

Note that $\ell_{ce}(v_k, y_k)$ is the same as predefined contrastive loss $\bar{\ell}_1^{(k)}$.

Connections to Mutual Information

- To extend to multiple views for training, we take the expectation over ℓ_{ce} and express the loss over all view-label pairs as

$$L_{ce} = -\mathbb{E}_{\mathbb{P}(v,y)}\mathbb{E}_{p(\mathcal{B}|\mathcal{V})}\left[\log q(y|b)\right].$$

- Without loss of generality, the loss L_{ce} can be written as

$$\begin{aligned} L_{ce} &= -\int \mathbb{P}(b,y)\log q(y|b)dydb. \\ &\geq -\int \mathbb{P}(b,y)\log \mathbb{P}(y|b)dydb = H(Y|B). \end{aligned}$$

- Using the $I(Y,B) = H(Y) - H(Y|B)$ and the fact that $H(Y)$ is a constant, we have

$$\min_{\theta} L_{ce} \Leftrightarrow \max_{\theta} I(Y,B).$$

The proposed model essentially maximizes the mutual information!

Improving under the IB Framework

- Now, it is natural to introduce the IB framework

$$R_{IB} = I(Y, B) - \beta I(B, V).$$

- From the non-negativeness of KL-divergence, it can be shown that

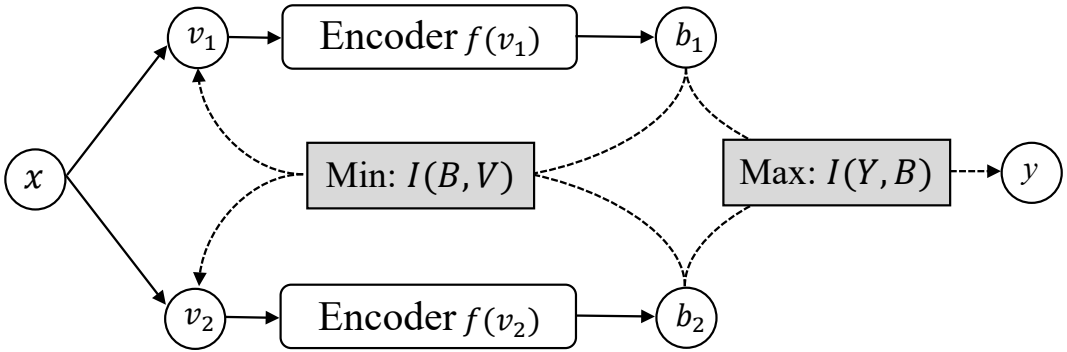
$$I(B, V) \leq \mathbb{E}_{\mathbb{P}(v)} \left[KL \left(\mathbb{P}(b|v) || q(b) \right) \right].$$

- the expression of $KL\left(\mathbb{P}(b|v)||q(b)\right)$ can be easily derived to be

$$\begin{aligned} KL(\mathbb{P}(b|v)||q(b)) = & \sum_{d=1}^{D} [\sigma(f_\theta(v))]_d \log \frac{[\sigma(f_\theta(v))]_d}{[\gamma]_d} \\ & + \sum_{i=1}^{D} (1 - [\sigma(f_\theta(v))]_d) \log \frac{1 - [\sigma(f_\theta(v))]_d}{1 - [\gamma]_d}. \end{aligned}$$

In our experiments, for a given view $v_1^{(k)}$, the value of γ is set by letting $q(b) = p(b|v_2^{(k)})$; and $q(b)$ for the view $v_2^{(k)}$ can be defined similarly.

Summary of Training and Inference



Training:

$$\max_{\theta} R_{IB} \Leftrightarrow \min_{\theta} L_{ce} + \beta \mathbb{E}_{\mathbb{P}(v)} \left[KL \left(\mathbb{P}(b|v) || q(b) \right) \right]$$

Inference:

$$b = \frac{\text{sign}(\sigma(f_\theta(x_i)) - 0.5) + 1}{2}$$

Unsupervised Image Hashing

- **Training Details:** the encoder network $f_\theta(\cdot)$ is constituted by a pre-trained VGG-16 network followed by an one-layer ReLU feedforward neural network
- **Task:** encode an image into a binary vector such that nearest neighbors (in Hamming distance) share same labels
  - Labels are only used for evaluation
- **Datasets:** CIFAR-10, NUS-WIDE and MSCOCO
- **Metrics:** MAP@1000 for CIFAR-10, MAP@5000 for NUS-WIDE and MSCOCO

Results

Table 1: MAP comparison with different state-of-the-art unsupervised hashing methods.

Method	Reference	CIFAR-10			NUS-WIDE			MSCOCO		
		16bits	32bits	64bits	16bits	32bits	64bits	16bits	32bits	64bits
DeepBit	CVPR16	0.194	0.249	0.277	0.392	0.403	0.429	0.407	0.419	0.430
SGH	ICML17	0.435	0.437	0.433	0.593	0.590	0.607	0.594	0.610	0.618
BGAN	AAAI18	0.525	0.531	0.562	0.684	0.714	0.730	0.645	0.682	0.707
BinGAN	NIPS18	0.476	0.512	0.520	0.654	0.709	0.713	0.651	0.673	0.696
GreedyHash	NIPS18	0.448	0.473	0.501	0.633	0.691	0.731	0.582	0.668	0.710
HashGAN	CVPR18	0.447	0.463	0.481	-	-	-	-	-	-
DistillHash	CVPR19	0.284	0.285	0.288	0.667	0.675	0.677	-	-	-
DVB	IJCV19	0.403	0.422	0.446	0.604	0.632	0.665	0.570	0.629	0.623
TBH	CVPR20	0.532	0.573	0.578	0.717	0.725	0.735	0.706	0.735	0.722
CIBHash	**Ours**	**0.590**	**0.622**	**0.641**	**0.790**	**0.807**	**0.815**	**0.737**	**0.760**	**0.775**

The proposed CIBHash model **outperforms** the current SOTA method by a substantial margin on all three datasets considered.

Results

Table 2: MAP comparison with variants of CIBHash.

Component Analysis		16bits	32bits	64bits
CIFAR-10	Naive CL	0.493	0.574	0.606
	CLHash	0.580	0.609	0.628
	CIBHash	**0.590**	**0.622**	**0.641**
MSCOCO	Naive CL	0.666	0.712	0.737
	CLHash	0.721	0.749	0.765
	CIBHash	**0.737**	**0.760**	**0.775**

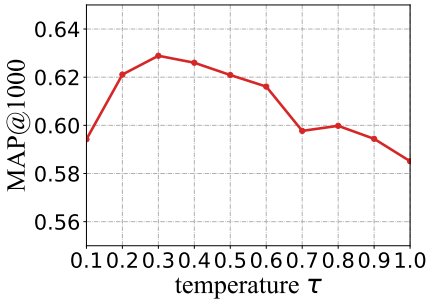


Figure 1: Model performance with various temperature τ in terms of 32bits on CIFAR-10.

- By dropping the projection head and enabling end-to-end training, CIBHash consistently outperform naive CL method;
- CIBHash achieve best performance thanks to the incorporation of broader IB framework.

Results

Table 3: MAP comparison with different IB-based methods.

Method	CIFAR-10		
	16bits	32bits	64bits
β-VAE	0.468	0.508	0.495
Multi-View β-VAE	0.465	0.492	0.522
CIBHash	**0.590**	**0.622**	**0.641**

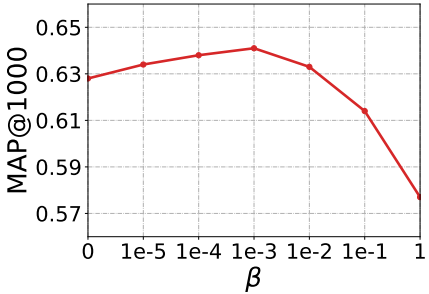


Figure 2: Model performance with various Lagrange multiplier β in terms of 32bits on CIFAR-10.

- The non-reconstruction-based method is better at extracting semantic information than the reconstruction-based methods.

Results

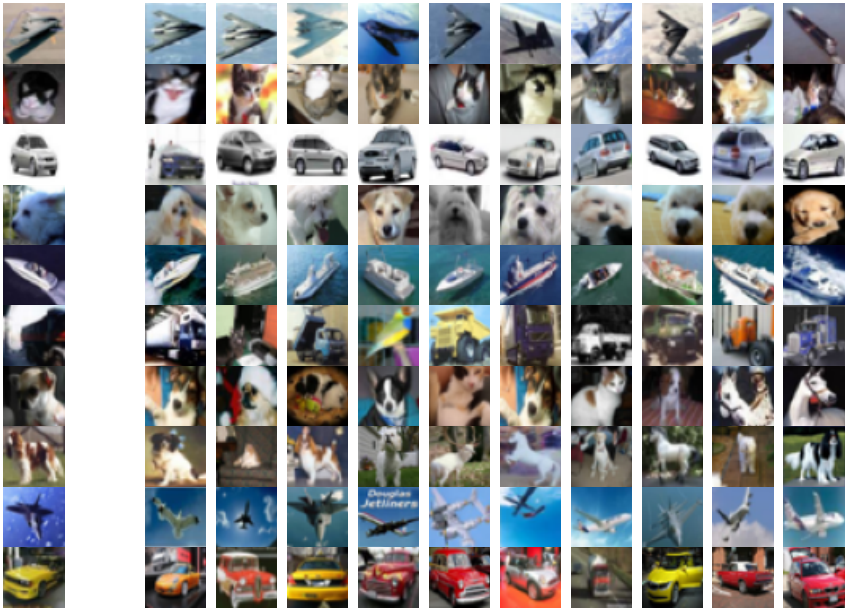


Figure 3: 64-bit retrieval results on CIFAR-10.

- CIBHash typically compresses images with shared labels into very similar binary codes.

Conclusions

- This paper presents a new paradigm towards non-reconstruction-based unsupervised hashing;
- We adapt the contrastive learning to probabilistic Bernoulli perspectives, thus enabling the end-to-end training;
- The connections between the proposed method and information bottleneck theory are established;
- Significant performance gains are observed with contrastive information bottleneck framework.