LABEL PREDICTION FRAMEWORK FOR SEMI-SUPERVISED CROSS-MODAL RETRIEVAL

Devraj Mandal[†] *Pramod Rao*[⋆] *Soma Biswas*[†]

[†] Indian Institute of Science, Bengaluru, India.
[⋆]Saarland University, Saarbrücken, Germany.

ABSTRACT

Cross-modal data matching refers to retrieval of data from one modality, when given a query from another modality. In general, supervised algorithms achieve better retrieval performance compared to their unsupervised counterpart, as they can learn better representative features by leveraging the available label information. However, this comes at the cost of requiring huge amount of labeled examples, which may not always be available. In this work, we propose a novel framework in a semi-supervised cross-modal retrieval setting, which can predict the labels of the unlabeled data using complementary information from different modalities. The proposed framework can be used as an add-on with any baseline cross-modal algorithm to give significant performance improvement, even in case of limited labeled data. Extensive evaluation using several baseline algorithms across three different datasets show the effectiveness of our label prediction framework.

Index Terms— Cross-modal retrieval, semi-supervised learning

1. INTRODUCTION

For the application of cross-modal retrieval, supervised algorithms [1] [2] [3] [4] generally outperform their unsupervised counterparts [5] [6] [7], but at the cost of additional label information. The performance also greatly depends upon the amount of labeled data. Since the task of labeling is often very expensive and time-consuming, designing deep based models to mitigate this shortcoming is very important. Semi-supervised (SS) learning treads the middle ground by considering a small subset of data as labeled, and the remaining as unlabeled and has been extensively studied in the context of image classification [8] [9] [10] [11] [12]. These approaches generally follow one of the three possible strategies namely (1) Pre-training the deep network using unlabeled examples followed by training it again with its labeled counterparts, (2) Using the unlabeled samples as a regularization term for structure preservation of the embedded features and (3) An iterative scheme in which label prediction and network parameter learning are done in an alternate fashion repeatedly. Recently, some semi-supervised cross-modal algorithms have been developed [13] [14] [15] [16] [17] [18], which aims to find out the optimal way to jointly use both the labeled and unlabeled data for getting better cross-modal retrieval performance.

In this work, given a set of labeled and unlabeled training data in a semi-supervised cross-modal setting, we propose a novel label prediction framework (LPF) to predict the labels for the unlabeled data. Utilizing the complementary information from both modalities as well as the original features, we filter out the data for which the predicted labels are potentially wrong and select only that portion whose predicted labels are probably correct to re-train the LPF. These two steps are repeated iteratively and with each iteration, more number of unlabeled examples and their predicted labels are added which helps to train the LPF network better. Finally, we use all the labeled and pseudo-labeled examples to train any supervised cross-modal algorithm. We perform extensive experiments to show the efficacy of our algorithm for different baselines and three datasets, even with limited labeled data. The main contributions of our work is as follows:

(1) We propose a novel label prediction framework for predicting labels of unlabeled data in a semi-supervised setting, which can then be fed to any supervised cross-modal algorithm.

(2) The proposed framework is effective even in case of limited amount of labeled data.

(3) Extensive experiments show the usefulness of the proposed framework using several baselines and three different datasets.

Next, we discuss the related work in literature. The proposed approach for different scenarios is discussed in Section 3. The results of experimental evaluation is reported in Section 4 and the paper ends with a conclusion.

2. RELATED WORK

Here we describe the relevant works in the semi-supervised (SS) setting, first for image classification task and then for cross-modal retrieval.

Semi-supervised image classification: As discussed in the introduction, three strategies are usually followed in literature for SS scenario. The first strategy is followed in [11], but its performance usually suffers since the second stage typically dominates and the model tends to forget what it has learned in the first stage. The second strategy is followed in the works of [9] [12] [19] [20]. Algorithms like in [8] [21] employ a hierarchical strategy in which the unlabeled examples are used for image reconstruction and the labeled examples are used for image classification. The work in [10] [22] follows the iterative approach of the third strategy. [10] uses entropy regularization, denoising auto-encoders and dropout in the network architecture to identify probable correctly labeled examples and utilizes it for further training of the image classification models. [22] also has the additional property of growing the network layers if the necessity arises with the accumulation of additional pseudo-labeled examples. Data augmentation techniques in the image domain can greatly boost image classification performance in the SS setting [23] [24]. This technique though very useful is

Pramod made equal contribution to this work

Acknowledgement: This work is partly supported through a research grant from SERB, Department of Science and Technology, Government of India.

difficult to implement for applications in the cross-modal setting.

Semi-supervised cross-modal retrieval: The problem of SS learning in the cross-modal setting is relativity less explored and has been addressed in [13] [14] [15] [16] [17] [18]. In [15] [14], multi-graph learning is used over the unlabeled data for structure preservation while learning the common embedding representations. The work in [13] designs a dragging technique with a linear regression model so that embedded features lie close to the correct class labels while pushing the irrelevant samples far apart. In [18], sparse representation of the different modality data for both the labeled and unlabeled samples are projected into a common domain defined by its class label information. A non-parametric Bayesian approach has been proposed in [16] to handle the SS situation. A novel approach in semi-supervised hashing using Generative Adversarial Network [17] has been used to model the distribution across the different modalities and a loss function has been suitably designed to select correct/similar data pairs from the unlabeled set in an adversarial fashion. Though [17] shows impressive performance, the amount of labeled data required is quite large. A recent work in [25] does label prediction by first generating its weakly supervised labels by comparison with its nearest neighbors and then finally training a network to predict its true label given the data sample and its weak annotation.

In this work, we propose a novel label prediction framework in a semi-supervised setting which predicts the labels of the unlabeled data, which can then be used to augment the labeled portion of the data and given as input to any baseline cross-modal algorithm.

3. PROPOSED METHOD

Here, we describe the proposed framework for the standard semi-supervised setting. Let the cross-modal data be represented as $\mathbf{X}_t \in \mathcal{R}^{d_t \times N}$ ($t \in \{1, 2\}$), where $t = 2$ is the number of modalities, N is the number of training samples and d_t is the feature dimension. Let the labels be denoted as $\mathbf{L} \in \mathcal{R}^{C \times N}$, where C is the number of classes, with each sample belonging to a single category. Consider that the input data $\mathbf{X}_t$ consists of (a) m labeled samples denoted by $\mathbf{X}_t^l \in \mathcal{R}^{d_t \times m}$ with its corresponding labels $\mathbf{L}^l \in \mathcal{R}^{C \times m}$ and (b) n unlabeled samples denoted by $\mathbf{X}_t^{ul} \in \mathcal{R}^{d_t \times n}$, with $m + n = N, m \leq n$. We consider both the labeled and unlabeled data to be paired.

Given this set of labeled and unlabeled data, we will now describe the Label Prediction Framework (LPF) which is trained to predict the labels of the unlabeled samples. For training the LPF, we subdivide the labeled portion of the training data $\mathbf{X}_t^l$ as $\mathbf{X}_t^{tr}, \mathbf{L}^{tr}$ and $\mathbf{X}_t^{val}, \mathbf{L}^{val}$ to form the training and validation sets. The proposed network architecture is shown in Fig. 1, which consists of encoders $\mathcal{E}_t$ and decoders $\mathcal{D}_t$ for both the modalities. In our implementation, both $\mathcal{E}_t$ and $\mathcal{D}_t$ consists of three fully connected (fc) layers (in mirror configuration) with ReLU and dropout between all the fc layers, except the final layer. The final layer of the encoder has two activation functions, namely (1) *Softmax* for predicting the labels and (2) *tanh* whose output is subsequently passed through the decoder. For input data x_{tj} (j^{th} sample from modality t) to the encoder $\mathcal{E}_t$, the output of the softmax is denoted as x_{tj}^s and that of the tanh layer is denoted as $x_{tj}^{\tanh}$. The encoded output $x_{tj}^{\tanh}$ is passed through $\mathcal{D}_t$ to get the reconstruction $\hat{x}_{tj}$. Now, we will describe the different losses (for each modality) used to train this network:

1. **Labeled data:** For the labeled portion of the data, we want the samples from same class to cluster together, which in turn will help in classification. We tap the output from the second fc layer in $\mathcal{E}_t$ and denote it as x_{tj}^f. We use the following two losses:
   Cross-entropy loss: We minimize the classification errors over the labeled examples by using cross-entropy loss $\mathcal{L}^{ce} = -\sum_{j=1}^{m} \log\left(e^{x_{tj}^s[i]} / \sum_{k=1}^{C} e^{x_{tj}^s[k]}\right)$. Here, i is the correct class index.
   Center loss [26]: We use this loss to minimize the distance of each sample with respect to its center representation as follows: $\mathcal{L}^c = \sum_{j=1}^{m} \sum_{k=1}^{C} \mathbf{1}_{\{l_j=k\}} ||x_{tj}^f - c_{t,k}||_2^2$. Here, $c_{t,k}$ denotes the k^{th} class center for t^{th} modality and $\mathbf{1}_{\{l_j=k\}}$ is the indicator variable which gets activated when label l_j of the sample x_{tj} is consistent with the correct center. This also ensures that the samples from the same class are clustered together. The centers $\{c_{t,k}\}_{k=1}^{C}$ are learned while training the network using [26]. We consider learning the centers $c_{t,k}$ from the second fc layer output as the final layer length is limited by the number of training categories, which is often small and hence the learned center representations might not be discriminative enough. The two losses are important [26], since $\mathcal{L}^c$ helps to make the classification using $\mathcal{L}^{ce}$ better by pushing the centers apart and making the features of each individual classes as clustered together as possible.
2. **Unlabeled data:** To make the label predictions of the unlabeled data less ambiguous, we utilize the **Entropy Regularization loss** [19] as $\mathcal{L}^{ent} = \sum_{j=m+1}^{N} -x_{tj}^s \log(x_{tj}^s)$. Since the unlabeled samples belong to one of the C categories, we want to make the softmax probability for a particular class as

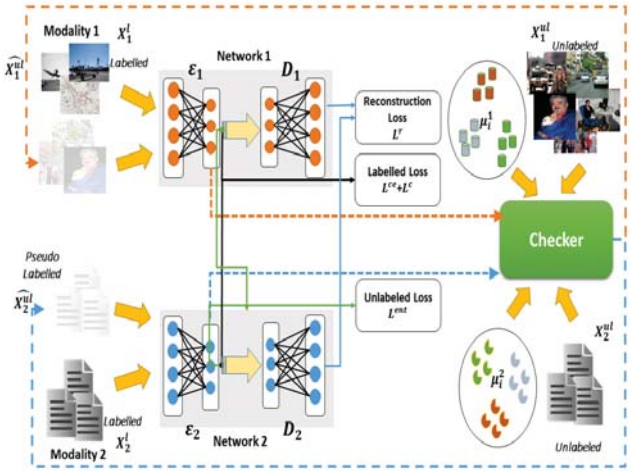

Fig. 1. Illustration of the proposed LPF. Initially, the labeled data $\{\mathbf{X}_t^l\}$ is used to learn the parameters of $\{\mathcal{E}_t, \mathcal{D}_t\}$ while minimizing $\mathcal{L}$. Next, we use the learned $\mathcal{E}_t$ to predict the labels for the unlabeled data $\mathbf{X}_t^{ul}$. The checker constructs the constraint set $\mathcal{C}$ using the original mean features μ^t and the performance of $\mathcal{E}_t$ on $\mathbf{X}_t^{val}$. Based on its decision, a subset of the unlabeled data $\hat{\mathbf{X}}_t^{ul}$ is selected and fed back to the network to fine-tune it further (the lightly highlighted elements denotes these pseudo-examples). Thus this model uses complimentary information from the original features as well as the paired information of the cross-modal data to select pseudo-examples judiciously for further fine-tuning of the network.

high as possible, which in turn is equivalent to minimizing the entropy of the prediction.

3. **Labeled and Unlabeled data:** To ensure that there is no loss of information in the encoder-decoder structure, for both the labeled and unlabeled samples, we use a **Reconstruction loss** at the decoder output given as $\mathcal{L}^r = \sum_{j=1}^{N} ||x_{tj} - \hat{x}_{tj}||_2^2$.

Thus, the total loss function for the entire network is given as: $\mathcal{L} = \alpha^{ce}\mathcal{L}^{ce} + \alpha^c\mathcal{L}^c + \alpha^{ent}\mathcal{L}^{ent} + \alpha^r\mathcal{L}^r$ where $\alpha^{ce}, \alpha^c, \alpha^{ent}, \alpha^r$ are the tunable hyper-parameters. Once the LPF network is trained, we can use it to predict the labels for the unlabeled samples (pseudo-labels).

Utilizing paired cross-modal information for verifying pseudo-labels: In this work, we leverage the complementary information available in the paired unlabeled data of the two modalities to verify if the label prediction given by LPF is reliable. For the j^{th} unlabeled data x_{tj}, if x^s_{tj} is the softmax output from $\mathcal{E}_t$, the predicted label is given by $l^{\mathcal{E}_t}_{tj} = \arg\max_i x^s_{tj}[i];\ 1 \leq i \leq C$. At the beginning of training, due to very limited amounts of data, these predictions are not reliable. This can be partially corrected by studying how close the original features are to their mean feature representations. Utilizing this fact, an alternate prediction on the unlabeled data can be made and both the predictions can be combined suitably to select the reliable predictions. Let us denote the mean features of each class for the t^{th} modality as $\{\mu^1_t, ..., \mu^C_t\}$. The means are computed using the original feature representation of the labeled data $\mathbf{X}^l_t$. The closest distance of the sample x_{tj} to this mean feature set is also a coarse prediction of the class it belongs to and is given by $l^{\mu}_{tj} = \arg\min_i ||x_{tj} - \mu^i_t||_2^2$. These four predictions can be computed for each data pair in $\mathbf{X}^{ul}_t$. Finally, the correctness of the label prediction is verified using a threshold τ. Specifically, a data pair (x_{1j}, x_{2j}) can be assumed to be correctly predicted if it satisfies the following conditions

$$\textbf{(1)}x^s_{1j}[i] \geq \tau, \quad \textbf{(2)}x^s_{2j}[i] \geq \tau, \quad \textbf{(3)}l^{\mu}_{1j} = l^{\mathcal{E}_1}_{1j}, \quad \textbf{(4)}l^{\mu}_{2j} = l^{\mathcal{E}_2}_{2j} \tag{1}$$

Let us term these set of constraints as $\mathcal{C}$. Here, we set $\tau = 0.9$ for all our experiments. We set a high value of τ to only consider the most confident predictions. This essentially implies that the confidence of the network's predictions must be more than τ each and it must match with the predictions made by the original features individually.

Since the two modalities have different features which may have different discriminative ability, we use a more relaxed condition to determine the correctness of the label prediction by taking either condition ((**1**) & (**3**)) or ((**2**) & (**4**)). The choice between the two conditions depends on the performance of $\mathcal{E}_t$ on the validation set $\mathbf{X}^{val}_t$. Let the accuracy of $\mathcal{E}_1, \mathcal{E}_2$ on $\mathbf{X}^{val}_1, \mathbf{X}^{val}_2$ by denoted as cf_1 and cf_2. Then the set $\mathcal{C}$ is determined as follows

$$\text{if, } cf_1 \geq cf_2, \quad \Rightarrow \mathcal{C} = \{\textbf{(1)}, \textbf{(3)}\} \qquad \text{otherwise, } \ \mathcal{C} = \{\textbf{(2)}, \textbf{(4)}\}$$

This approach has two-fold advantages, (1) It automatically selects the good features, thus there is no need for manual intervention and (2) automatic switching may occur which basically means that the label predictions will be driven in a complimentary fashion. In addition, since we are updating the constraint set $\mathcal{C}$ at each iteration to reflect the better classifier's performance, it is expected that increasingly more number of correctly labeled examples from $\mathbf{X}^{ul}_t$ gets selected. We thus get the set of unlabeled examples whose predictions are likely to be correct as $\hat{\mathbf{X}}^{ul}_t = \{x \mid x \in \mathbf{X}^{ul}_t \text{ and satisfies } \mathcal{C}\}$.

Algorithm 1 The Label Prediction Network

1: **Input** : $\mathbf{X}^{tr}_t, \mathbf{X}^{val}_t, \mathbf{X}^{ul}_t\ \{t = 1, 2\}, \mathbf{L}^{tr}, \mathbf{L}^{val}$.
2: **Output** : Data $\hat{\mathbf{X}}_t$ and their predicted labels $\hat{\mathbf{L}}$.
3: **Initialize** : Initialize the network parameters of $\mathcal{E}_t, \mathcal{D}_t$. Learn the mean feature sets μ^t.
4: Train $\mathcal{E}_t, \mathcal{D}_t$ using $(\mathbf{X}^{tr}_t, \mathbf{L}^{tr})$ by computing the loss $\mathcal{L}$ and back-propagating the error.
5: Continue until $|\hat{\mathbf{X}}^{ul}_t|$ does not change or until T iterations (whichever earlier):
6: Measure performance cf_t on validation set $\mathbf{X}^{val}_t, \mathbf{L}^{val}$ using $\mathcal{E}_t$.
7: Determine $l^{\mu}_{tj}, l^{\mathcal{E}^t}_{tj}$ for each sample in unlabeled set $\mathbf{X}^{ul}_t$.
8: Construct the new constraint set $\mathcal{C}$. Use this to determine $\hat{\mathbf{X}}^{ul}_t$.
9: Form $\hat{\mathbf{X}}_t$ as $[\mathbf{X}^{tr}_t\ \ \hat{\mathbf{X}}^{ul}_t]$ & $\hat{\mathbf{L}}$ as $[\mathbf{L}^{tr}\ \ \mathbf{L}^{pl}]$.
10: Fine-tune $\mathcal{E}_t, \mathcal{D}_t$ with a lower learning rate to update the network parameters.

Now, the expanded labeled set is given by $\hat{\mathbf{X}}_t = [\mathbf{X}^{tr}_t\ \ \hat{\mathbf{X}}^{ul}_t]$ with labels $\hat{\mathbf{L}} = [\mathbf{L}^{tr}\ \ \mathbf{L}^{pl}]$, where $\mathbf{L}^{pl}$ are the predicted labels. We use this data to further fine-tune our LPF network with a smaller learning rate. We repeat the label prediction and network fine-tuning iteratively until the cardinality of $\hat{\mathbf{X}}^{ul}_t$ saturates. Finally, we can feed $(\hat{\mathbf{X}}_t, \hat{\mathbf{L}})$ to any supervised cross-modal baseline algorithm for retrieval. Algorithm 1 gives the different steps of the LPF.

4. EXPERIMENTAL EVALUATION

We consider three standard single label datasets for evaluating the proposed approach. The UCI Digit data [27] contains different feature representations of handwritten numerals for ten categories i.e., (0-9) with 200 examples each. The train:test split is 1500:500. The features used for our experiments are the same as in [28]. The LabelMe data [29] contains image-text pairs from eight different categories. GIST features are considered for the image domain and Word frequency vector for the text domain [29]. We take 200 samples from each category in the training set and the rest of the samples in the testing set. Wiki data [30] contains 2,866 image-text pairs from 10 different categories. The images and texts are represented using 4096-d CNN descriptors and 100-d word vectors respectively. The train:test split considered is 2000:866 as in [14].

Mean Average Precision (MAP) is used as the evaluation metric for comparisons against the other cross-modal retrieval methods. We report average MAP@50 (average over Image-to-Text and Text-to-Image retrieval) for all our experiments [14].

Baseline Approaches: We consider a variety of baseline cross-modal algorithms like CCCA [1] GSSL [14] GsPH [3] LCMF [4] SCM [31] (SCM$_s$ and SCM$_o$ denotes the sequential and orthogonal versions) SePH [2] SMFH [32] ACMR [33] GrowBit [34] with which we integrate the proposed LPF. The last two are deep-learning based approaches and the others are classical non-deep approaches. We take the publicly available versions of the author's codes or re-implement them wherever necessary while running the baseline algorithms. We set the parameters of the baseline algorithms in accordance to strategies described in the individual papers. For the hashing based approaches, we used hash code of length 64.

Results for Semi-Supervised Protocol: Here, we report the results of the LPF module as an add-on with other baseline approaches for

Table 1. Average MAP@50 on UCI [27], Wiki [30] and LabelMe [29] datasets. Here, "f", "l" and "ss" denotes the three modes of operation. $^+$ indicates deep based algorithms. * indicates that GSSL is working in a semi-supervised mode in "b" and "c".

		UCI [27]			Wiki [30]			LabelMe [29]		
	ρ	10	30	50	10	30	50	10	30	50
CCCA [1]	f	0.667			0.419			0.639		
	l	0.634	0.648	0.657	0.314	0.362	0.381	0.573	0.620	0.628
	ss	0.639	0.655	0.657	0.379	0.398	0.400	0.611	0.624	0.640
GsPH [3]	f	0.853			0.473			0.820		
	l	0.779	0.821	0.833	0.359	0.426	0.451	0.717	0.784	0.794
	ss	0.800	0.833	0.842	0.426	0.449	0.467	0.746	0.792	0.792
LCMF [4]	f	0.847			0.484			0.827		
	l	0.774	0.819	0.834	0.354	0.422	0.447	0.719	0.790	0.799
	ss	0.809	0.830	0.843	0.425	0.451	0.470	0.758	0.799	0.801
SCM_s [31]	f	0.652			0.358			0.694		
	l	0.509	0.584	0.595	0.274	0.313	0.328	0.554	0.630	0.681
	ss	0.598	0.628	0.626	0.332	0.328	0.331	0.634	0.651	0.676
SCM_o [31]	f	0.437			0.273			0.475		
	l	0.364	0.403	0.402	0.214	0.236	0.239	0.382	0.423	0.453
	ss	0.401	0.411	0.430	0.234	0.243	0.240	0.420	0.432	0.437
SePH [2]	f	0.844			0.477			0.813		
	l	0.781	0.820	0.834	0.359	0.429	0.454	0.717	0.774	0.787
	ss	0.800	0.827	0.833	0.429	0.453	0.461	0.740	0.790	0.790
SMFH [32]	f	0.686			0.335			0.711		
	l	0.584	0.651	0.659	0.277	0.301	0.315	0.580	0.649	0.688
	ss	0.654	0.667	0.662	0.321	0.322	0.324	0.684	0.703	0.709
$ACMR^+$ [33]	f	0.776			0.444			0.828		
	l	0.543	0.694	0.721	0.319	0.396	0.411	0.642	0.765	0.801
	ss	0.751	0.757	0.768	0.421	0.440	0.436	0.767	0.797	0.823
$GrowBit^+$ [34]	f	0.812			0.465			0.833		
	l	0.558	0.773	0.784	0.279	0.390	0.419	0.654	0.792	0.812
	ss	0.785	0.795	0.802	0.409	0.445	0.448	0.756	0.796	0.816
$GSSL^*$ [14]	f	0.731			0.455			0.739		
	b	0.429	0.535	0.566	0.180	0.226	0.229	0.373	0.385	0.430
	c	0.589	0.588	0.589	0.254	0.257	0.247	0.411	0.398	0.424

the three datasets. We denote the results for each algorithm under three different modes, (1) 'f", denotes that the algorithm is working in supervised mode with no unlabeled data ; (2) "l", using only labeled portion of the data and (3) "ss", where the pseudo-labeled examples as predicted by LPF are provided in addition to the labeled data. We consider $\rho\%$ of the total training data as labeled, and the remaining as unlabeled and we report results for $\rho = \{10\%, 30\%, 50\%\}$. All experiments are repeated over 5 random labeled:unlabeled split and the average results are reported in Table 1. We make the following observations, (1) the result of "f" mode is the best as expected as it has access to all the labeled training data; (2) the results under "ss" mode is better than "l" mode which signifies that the proposed LPF is able to correctly predict the labels of the unlabeled set and pass it to the baseline algorithms; (3) the importance of LPF module is more when ρ is low, i.e. when the amount of labeled data is very limited, thus making the training more challenging; (4) LPF works equally well for non-deep and deep based algorithms. We observe similar pattern as we increase ρ from 50% to 90%, though the performance difference is less.

We conduct an additional experiment with the semi-supervised approach GSSL [14]. In Table 1, for GSSL, "b" implies that all labeled and unlabeled samples are provided to the algorithm, and "c" implies the case where it uses the labeled data, LPF predicted pseudo-labeled data and the remaining unlabeled data. In "b" and "c", GSSL is working as a semi-supervised algorithm. Though GSSL is designed to handle unlabeled data, the proposed LPF still gives significant improvement justifying its usefulness.

Now we show how the class prediction accuracies for the unlabeled data evolves with each epoch for the LabelMe dataset [29] with 10% labeled data in Fig. 2. The red and blue bars denote the per class accuracy over the unlabeled data of the two modalities (the higher the better). We also denote the validation accuracy over all the 8 categories in the LabelMe [29] dataset below each bar chart in Fig. 2. We observe that from left to right, the accuracy of both the networks improve. The phenomenon of automatic switching where the constraint set is changing to reflect the better updated network is shown with a star. Automatic switching helps the network to interplay among themselves for a better learning mechanism. Table 2 reports how many examples are being selected per epoch to be fed back for fine-tuning as the algorithm proceeds, and the accuracy of selection of the unlabeled examples for LabelMe [29] and UCI [27] data. We observe that as the algorithm proceeds, more number of examples gets selected and with good accuracy. The extra correct examples helps to give better results when baseline algorithms are run with these predicted labels.

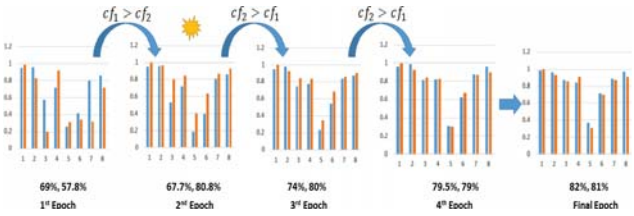

Fig. 2. Here, we show how the two networks are inter-playing among each other to help learn better. In each plot, from left to right, the red and blue bars (8 to denote each category) indicate per-class accuracy over the unlabeled data for the a single split of LabelMe [29] dataset with 10% provided labels. We observe that both the networks are learning better as the validation accuracy of $\{69\%, 57.8\%\}$ has increased to $\{82\%, 81\%\}$ at the end of the run. The star indicates the automatic switching phenomenon.

Table 2. This table reports the number of examples that are selected per epoch to be fed back to $\{\mathcal{E}_t, \mathcal{D}_t\}$ for further fine-tuning when having only 10% labeled data. We also report the prediction accuracy of the selected examples per epoch (the higher the better).

	1st Epoch	5th Epoch	9th Epoch
UCI	(730, 96.98%)	(772,96.63%)	(887, 96.50%)
LabelMe	(971, 95.7%)	(1025, 95.70%)	(1027, 95.7%)

Implementation Details: The $\mathcal{E}_t$ in LPF module has 3 fc layers of size $250 - 250 - C$ with $\mathcal{D}_t$ having the mirror architecture. We train $(\mathcal{E}_t, \mathcal{D}_t)$ using Stochastic Gradient Descent with learning rate between $lr = 10^{-2} - 10^{-3}$ for 200 epochs respectively with early stopping condition. LPF fine-tuning is done using a lower learning rate typically $lr = 10^{-4} - 10^{-5}$. For updating the centers, a learning rate of $lr' = 5lr$ is used. The hyper parameters for the loss are set as $\alpha^{ce} = 1, \alpha^r = 0.01, \alpha^c = 0.5$, and $\alpha^{ent} = 1$.

5. CONCLUSION AND FUTURE WORK

In this work, we propose a novel label prediction framework in semi-supervised setting which can act as an add-on to any cross-modal retrieval baseline algorithm to achieve better performance even in case of limited labeled data. Extensive experimental evaluation on three datasets and several non-deep and deep baseline approaches shows the usefulness of the proposed framework.

6. REFERENCES

[1] Nikhil Rasiwasia, Dhruv Mahajan, Vijay Mahadevan, and Gaurav Aggarwal, "Cluster canonical correlation analysis," in *AISTATS*, 2014, pp. 823–831.

[2] Z. Lin, G. Ding, M. Hu, and J. Wang, "Semantics-preserving hashing for cross-view retrieval," in *CVPR*, 2015, pp. 3864–3872.

[3] Devraj Mandal, Kunal N Chaudhury, and Soma Biswas, "Generalized semantic preserving hashing for n-label cross-modal retrieval," in *CVPR*, 2017, pp. 4076–4084.

[4] Devraj Mandai and Soma Biswas, "Label consistent matrix factorization based hashing for cross-modal retrieval," in *ICIP*, 2017, pp. 2901–2905.

[5] Harold Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, no. 3/4, pp. 321–377, 1936.

[6] David R Hardoon, Sandor Szedmak, and John Shawe-Taylor, "Canonical correlation analysis: An overview with application to learning methods," *Neural computation*, vol. 16, no. 12, pp. 2639–2664, 2004.

[7] G. Ding, Y. Guo, and J. Zhou, "Collective matrix factorization hashing for multimodal data," in *CVPR*, 2014, pp. 2083–2090.

[8] Antti Rasmus, Mathias Berglund, Mikko Honkala, Harri Valpola, and Tapani Raiko, "Semi-supervised learning with ladder networks," in *NIPS*, 2015, pp. 3546–3554.

[9] Elad Hoffer and Nir Ailon, "Semi-supervised deep learning by metric embedding," *arXiv preprint arXiv:1611.01449*, 2016.

[10] Dong-Hyun Lee, "Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks," in *ICML Workshop*, 2013.

[11] Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling, "Semi-supervised learning with deep generative models," in *NIPS*, 2014, pp. 3581–3589.

[12] Jason Weston, Frédéric Ratle, Hossein Mobahi, and Ronan Collobert, "Deep learning via semi-supervised embedding," in *Neural Networks: Tricks of the Trade*, pp. 639–655. Springer, 2012.

[13] Liang Zhang, Bingpeng Ma, Jianfeng He, Guorong Li, Qingming Huang, and Qi Tian, "Adaptively unified semi-supervised learning for cross-modal retrieval.," in *IJCAI*, 2017, pp. 3406–3412.

[14] Liang Zhang, Bingpeng Ma, Guorong Li, Qingming Huang, and Qi Tian, "Generalized semi-supervised and structured subspace learning for cross-modal retrieval," *IEEE Transactions on Multimedia*, vol. 20, no. 1, pp. 128–141, 2018.

[15] Jiale Wang, Guohui Li, Peng Pan, and Xiaosong Zhao, "Semi-supervised semantic factorization hashing for fast cross-modal retrieval," *Multimedia Tools and Applications*, vol. 76, no. 19, pp. 20197–20215, 2017.

[16] Behnam Gholami and Abolfazl Hajisami, "Probabilistic semi-supervised multi-modal hashing.," in *BMVC*, 2016.

[17] Jian Zhang, Yuxin Peng, and Mingkuan Yuan, "Sch-gan: Semi-supervised cross-modal hashing by generative adversarial network," *IEEE transactions on cybernetics*, vol. 50, no. 2, pp. 489–502, 2018.

[18] Xing Xu, Yang Yang, Atsushi Shimada, Rin-ichiro Taniguchi, and Li He, "Semi-supervised coupled dictionary learning for cross-modal retrieval in internet images and texts," in *ACM-MM*, 2015, pp. 847–850.

[19] Jost Tobias Springenberg, "Unsupervised and semi-supervised learning with categorical generative adversarial networks," *arXiv preprint arXiv:1511.06390*, 2015.

[20] Augustus Odena, "Semi-supervised learning with generative adversarial networks," *arXiv preprint arXiv:1606.01583*, 2016.

[21] J Zhao, M Mathieu, R Goroshin, and Y Lecun, "Stacked what-where auto-encoders. arxiv 2015," *arXiv preprint arXiv:1506.02351*, 2015.

[22] Guangcong Wang, Xiaohua Xie, Jianhuang Lai, and Jiaxuan Zhuo, "Deep growing learning," in *ICCV*, 2017, pp. 2812–2820.

[23] Samuli Laine and Timo Aila, "Temporal ensembling for semi-supervised learning," *arXiv preprint arXiv:1610.02242*, 2016.

[24] Antti Tarvainen and Harri Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *NIPS*, 2017, pp. 1195–1204.

[25] Devraj Mandal, Pramod Rao, and Soma Biswas, "Semi-supervised cross-modal retrieval with label prediction," *IEEE Transactions on Multimedia*, 2019.

[26] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao, "A discriminative feature learning approach for deep face recognition," in *ECCV*, 2016, pp. 499–515.

[27] R. He, M. Zhang, L. Wang, Y. Ji, and Q. Yin, "Cross-modal subspace learning via pairwise constraints," *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5543–5556, 2015.

[28] Hong Liu, Rongrong Ji, Yongjian Wu, Feiyue Huang, and Baochang Zhang, "Cross-modality binary code learning via fusion similarity hashing," in *CVPR*, 2017, pp. 7380–7388.

[29] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "Labelme: A database and web-based tool for image annotation," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 157–173, 2008.

[30] N. Rasiwasia, J. C. Pereira, E. Coviello, G. Doyle, G. Lanckriet, R. Levy, and N. Vasconcelos, "A new approach to cross-modal multimedia retrieval," in *ACM-MM*, 2010, pp. 251–260.

[31] D. Zhang and W. J. Li, "Large-scale supervised multimodal hashing with semantic correlation maximization.," in *AAAI*, 2014, pp. 2177–2183.

[32] J. Tang, K. Wang, and L. Shao, "Supervised matrix factorization hashing for cross-modal retrieval," *IEEE Transactions on Image Processing*, vol. 25, no. 7, pp. 3157–3166, 2016.

[33] Bokun Wang, Yang Yang, Xing Xu, Alan Hanjalic, and Heng Tao Shen, "Adversarial cross-modal retrieval," in *ACM-MM*, 2017, pp. 154–162.

[34] Devraj Mandal, Yashas Annadani, and Soma Biswas, "Growbit: Incremental hashing for cross-modal retrieval," in *ACCV*, 2018.