LEARNING MULTI-LABEL AERIAL IMAGE CLASSIFICATION UNDER LABEL NOISE: A REGULARIZATION APPROACH USING WORD EMBEDDINGS

Yuansheng Hua[1,2], *Sylvain Lobry*[3], *Lichao Mou*[1,2], *Devis Tuia*[3], *Xiao Xiang Zhu*[1,2]

[1]Signal Processing in Earth Observation, Technical University of Munich (TUM), Munich, Germany
[2]Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), Wessling, Germany
[3]Laboratory of Geo-information Science and Remote Sensing, Wageningen University, Wageningen, the Netherlands

ABSTRACT

Training deep neural networks requires well-annotated datasets. However, real world datasets are often noisy, especially in a multi-label scenario, i.e. where each data point can be attributed to more than one class. To this end, we propose a regularization method to learn multi-label classification networks from noisy data. This regularization is based on the assumption that semantically close classes are more likely to appear together in a given image. Hereby, we encode label correlations with prior knowledge and regularize noisy network predictions using label correlations. To evaluate its effectiveness, we perform experiments on a mutli-label aerial image dataset contaminated with controlled levels of label noise. Results indicate that networks trained using the proposed method outperform those directly learned from noisy labels and that the benefits increase proportionally to the amount of noise present.

Index Terms— noisy labels, regularization, label correlations, multi-label classification, deep neural networks

1. INTRODUCTION

Recently, deep neural networks have obtained tremendous achievements in a variety of remote sensing tasks, such as land cover classification [1] and mapping [2, 3]. One of the key reasons for these successes is the increasing volume of available remote sensing datasets. However, considering the complexity and ambiguity of remote sensing image contents, it is not easy to annotate every image or pixel accurately. Besides, some data producers resort to crowdsourcing data or web search engines for the purpose of reducing the cost of annotation, which introduces label noise as well. As a consequence, the predictive performance of deep neural networks directly trained on such noisy labels might be limited. The problem is exacerbated when allowing more than one label per data point, i.e. multi-label scenarios.

Some methods have been proposed in the past few years to address the problem of learning with noisy labels.

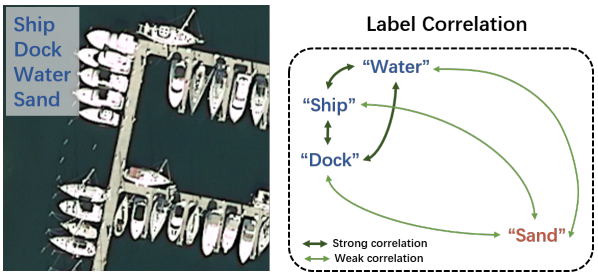


Fig. 1: Illustration of the label correlation regularization. To regularize predictions of a classifier learned from noisy data (left image), we map each label into the word embedding space, where representations/vectors of correlated labels are adjacent to each other. By measuring distances among these vectors, irrelevant labels (e.g., "sand") can be discarded using their word distances from the others being predicted.

Damodaran et al. [4] propose to improve the network robustness to noisy labels with adversarial virtual examples sampled using Wasserstein distances between classes. Sukhbaatar and Fergus [5] learn a noise transition matrix, which transforms clean predictions to noisy outputs used to compute the loss. Hu et al. [6] employ both clean and noisy data to supervise the learning of a multi-label classification network. However, most existing works either only focus on single-label classification [4, 5] or introduce clean training data [6], which is more often not available in real settings.

In this paper, we propose a new regularization method, namely label correlation regularization (LCR), to train multi-label classification models with noisy labels. Specifically, the proposed LCR aims at regularizing predictions of neural networks with label correlations extracted from prior knowledge similarly to the effect of a pairwise energy term in a Conditional Random Field (CRF). An intuitive explanation is that an unlikely prediction (e.g., *sand* and *ship* occur simultaneously) should be penalized, while a reasonable prediction (e.g., *pavement* and *car* appear together) get rewarded. However, the question then arises: *"How to define label correla-*

tions?" A common solution [7] is to take label co-occurrence patterns as the label correlation matrix, where frequently co-occurring labels are considered as highly related. Unfortunately, this approach is data-driven and therefore sensitive to the inherent label noise of the dataset used. To tackle this limitation, we model label correlations by measuring distances between prior label representations, e.g., word embeddings of labels as in [4]. Such a design has the advantage of being independent of the dataset label noise. Using this label-space distance, LCR pushes neural networks to make reasonable predictions according to the class semantics.

2. METHODOLOGY

2.1. Label correlation encoding with word embeddings

To encode label correlations, we use distances between prior label representations, i.e. word embeddings. A word embedding is a dense numerical representation/vector of a word, which is learned using NLP models, such as Word2Vec [8] and Glove [9]. These two models project words in an embedding space where similar concepts are close to each other.

To be more specific, we first use a pre-trained word embedding model to map each label to a high-dimensional vector, which is expected to contain label-relevant semantics. Afterwards, label correlations can be measured by calculating similarities/distances between corresponding word vectors. Formally, let L be the number of all candidate labels, i.e. possible classes, and $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ denote word embeddings of classes i and j, respectively. Label correlations (LC) can then be computed as follows:

$$\mathrm{LC}(i,j) = f(\boldsymbol{x}_i, \boldsymbol{x}_j), \tag{1}$$

where f denotes a similarity measure. In this work, we employ two measures of similarity: 1) a dot product:

$$f(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^T \boldsymbol{x}_j, \tag{2}$$

and 2) the Euclidean distance:

$$f(\boldsymbol{x}_i, \boldsymbol{x}_j) = e^{-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2}. \tag{3}$$

Applying Eq. 1 to all label pairs, a label correlation matrix can be built, where each element indicates the correlation between two labels. It is noteworthy that we normalize the correlation matrix between 0 and 1. Therefore, values approaching 1 suggest strong label correlations, while those close to 0 indicate weak relationships. The label correlation matrix is denoted as $\boldsymbol{A}$ in the following subsections.

2.2. Label Correlation Regularization (LCR)

Given a set of labeled images $\{\boldsymbol{I}_n, \boldsymbol{y}_n\}_{n=1,\dots N}$, the multi-label classification problem is considered as learning a classifier, e.g., a neural network, which can predict multiple labels $\hat{\boldsymbol{y}}_n$ of input images $\boldsymbol{I}_n$ with values 0 (absence of a class) and 1 (presence of a class). Here, $\boldsymbol{y}_n$ and $\hat{\boldsymbol{y}}_n$ are multi-hot vectors of L dimensions corresponding to the original noisy and predicted multiple labels of the n-th image $\boldsymbol{I}_n$, respectively. N represents the number of images. The learning problem is then defined as minimizing an empirical risk, e.g., binary cross-entropy:

$$\mathcal{L}_{BCE} = -\frac{1}{N}\sum_{n=1}^{N} \boldsymbol{y}_n \log(\hat{\boldsymbol{y}}_n) + (1-\boldsymbol{y}_n)\log(1-\hat{\boldsymbol{y}}_n). \tag{4}$$

Unfortunately, learning directly with the original noisy $\boldsymbol{y}_n$ leads to overfitting the label noise. To address this problem, we propose a regularization term $\mathcal{L}_{LCR}$ enforcing label correlations:

$$\mathcal{L}_{LCR} = \frac{1}{N \cdot N_p}\sum_{n=1}^{N}\sum_{i_p \in P}\sqrt{\sum_{j \in L,} j \neq i_p}(\hat{y}_{n,j} - \mathrm{LC}(i_p, j))^2, \tag{5}$$

where $\hat{y}_{n,j}$ indicates the j-th element of $\hat{\boldsymbol{y}}_n$, and P denotes the collection of positive predictions. N_p represents the number of positive predictions. With this regularization, a model is enforced to close gaps between predictions and label correlations. Specifically, for a label predicted as positive i_p, its highly related labels are favored, while uncommon labels co-occurrence is penalized. compute Eq. (5) in matrix form to make it more efficient:

1. A prediction $\hat{\boldsymbol{y}}_n$ (size $L \times 1$) is first binarized with a threshold of 0.5, and then replicated along the second axis to form a $L \times L$ mask, $\hat{\boldsymbol{M}}_n$. In this mask, elements at the i_p-th row are 1, while the others are 0.
2. The element-wise multiplication of $\hat{\boldsymbol{M}}_n$ and $\boldsymbol{A}$ is conducted to mask out label correlations with respect to negative predictions. That is to say, entries at rows with indexes belonging to $\neg P$ are all 0.
3. The matrix multiplication of $\hat{\boldsymbol{y}}_n$ and its transpose $\hat{\boldsymbol{y}}_n^T$ is performed. It is noteworthy that the i_p-th row of $\hat{\boldsymbol{Y}}_n$ is a replica of $\hat{\boldsymbol{y}}_n^T$, while the other rows are composed of only zeros.
4. The distance between the masked $\boldsymbol{A}$ and $\hat{\boldsymbol{Y}}_n$ is calculated, and $\mathcal{L}_{LCR}$ is then obtained by averaging it.

Accordingly, Eq. (5) can be rewritten as:

$$\mathcal{L}_{LCR} = \frac{1}{N}\sum_{n=1}^{N}\mathbb{D}(\hat{\boldsymbol{Y}}_n, \hat{\boldsymbol{M}}_n\boldsymbol{A}), \tag{6}$$

where $\mathbb{D}$ denotes the distance metric. Here we compute the distance by averaging the Euclidean distance between each row in $\hat{\boldsymbol{Y}}_n$ and $\hat{\boldsymbol{M}}_n\boldsymbol{A}$.

The final loss is given by:

$$\mathcal{L} = \mathcal{L}_{BCE} + \alpha\mathcal{L}_{LCR}, \tag{7}$$

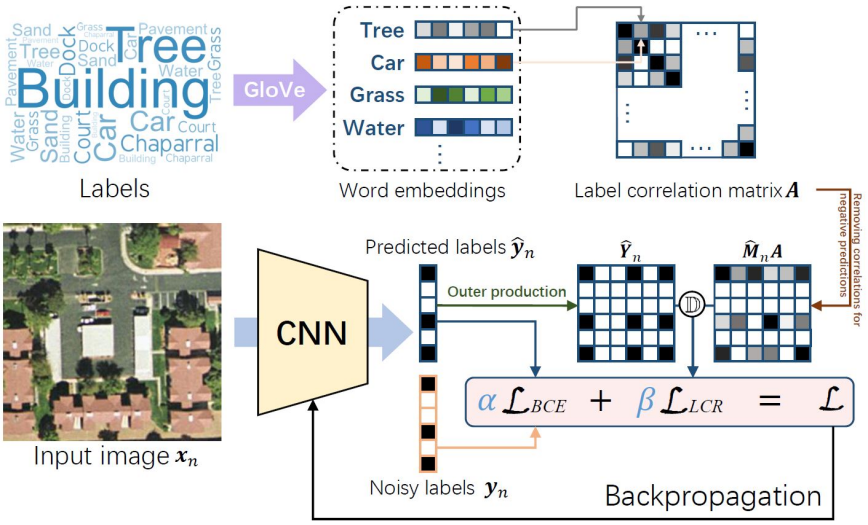


Fig. 2: The proposed regularization of predictions with label correlations encoded using word embeddings.

where α is a tradeoff parameter representing the strength of the prior $\mathcal{L}_{LCR}$. Figure 2 shows the workflow of the proposed LCR.

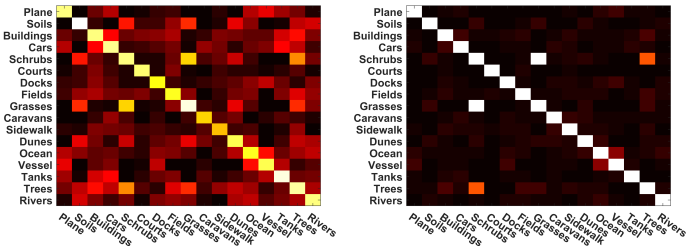
3. EXPERIMENTS AND DISCUSSION

To evaluate the performance of our proposed LCR, we experiment using ResNet-152 [10] as the backbone and the UCM multi-label dataset presented below.

3.1. Data description

UCM multi-label dataset [11] is a multi-label extension of the UCM dataset [12]: it is composed of 2100 aerial images cropped from aerial ortho imagery provided by the United States Geological Survey (USGS) National Map. The size of each image is 256×256 pixels, and the spatial resolution is 30 cm. Thanks to authors in [11], all images are reassigned with one or more labels according to primitive objects, and there are 17 newly defined object labels in total. We use 80% of the data for training.

Since we expect to assess the effectiveness of LCR on noisy labels, we simulate label noise in the training data. Specifically, we artificially corrupt up to 60% of the training samples with symmetric label noise. Considering the complexity of multiple labels, we inject noise to each label independently by flipping each element of $\boldsymbol{y}_n$ with a uniform probability of r, so called noise ratio. After flipping the labels, we learn the network with only 90% of these noisy training data and use the remaining noisy training samples to validate the network performance during the training phase, i.e. we do not use clean labels for model optimization.



(a) (b)

Fig. 3: Visualization of label correlation matrices measured with (a) Eq. (2) and (b) Eq. (3). White/black indicates 0/1.

3.2. Implementation details of LCR

To leverage word embeddings more efficiently, we adjust text descriptions of each label according to the following two rules. On one hand, countable nouns are changed from singular form to plural form, as objects often occur in the form of groups in an aerial image, such as trees and buildings. On the other hand, some labels are replaced with words, which are more in line with practical semantics in the dataset. For examples, `water` is modified as `rivers`. Afterwards, we adopt the 300-dim GloVe [9] pre-trained on the Wikipedia 2014 and Gigaword 5 dataset to map each label into the embedding space. Label correlation matrices built on Eq. (2) and (3) are visualized in Fig. 3.

3.3. Training details

We use a plain ResNet-152 network as a baseline, which is modified in a similar way as to the architecture of the multi-label classification network: On one hand, the number of units

Table 1: Numerical Results on UCM Multi-label Dataset (%)

r	0.0	0.1	0.2	0.3	0.4	0.5	0.6
plain	**80.03**	68.41	61.79	47.25	39.15	26.74	22.69
LCR†	76.17	**71.49**	**66.88**	**57.00**	**41.35**	32.32	30.76
LCR‡	74.98	70.56	65.29	53.79	40.60	**33.61**	**33.61**

plain represents training with only binary cross-entropy. LCR† and LCR‡ represent learning using both binary cross-entropy and LCR with $\boldsymbol{A}$ yielded from Eq. (2) and Eq. (3), respectively.

in the last fully-connected layer is reduced to L, e.g., 17 in our case. On the other hand, the last softmax layer is replaced with a sigmoid layer. In the training phase, we initialize layers before the last fully-connected layer with those in ResNet-152 pre-trained on ImageNet. The network loss is defined as $\mathcal{L}$, where α is set as 1. We select Adam with Nesterov momentum [13] as the optimizer, and its parameters are set as recommended [13]. The learning rate is initialized with the default value of $2e-03$ and decreased by a factor of 10, when the validation loss no longer decreases.

We implemented our model on TensorFlow-1.12.0 and trained for 100 epochs. The computational resource is an NVIDIA Tesla P100 GPU with a 16GB memory. The size of training batches is set as 32.

3.4. Discussion of the result

To compare the performance of networks learned with noisy data, we calculate the mean example-based F_1 score [14] as the evaluation metric and report numerical results in Table 1. We can see that the classification performance of ResNet-152 directly learned from noisy data (row: "plain") decreases drastically with an increasing noise ratio. Although performance decrease can be observed in the networks trained with LCR, their performances are more robust. Already at 10% of noise injected, LCR surpasses the original model and constantly outperforms it for increasing noise levels. Table 2 gives several examples with $r = 0.6$. As show in the first example, ResNet-152 with LCR regularization can make more correct predictions and fewer erroneous decisions compared to directly training on noisy labels. The last example shows a situation where both networks correctly predict four labels, while the later network makes fewer mistakes thanks to LCR regularization.

4. CONCLUSION AND OUTLOOK

We proposed a regularization method, LCR, to efficiently train a deep neural network with noisy labels. In addition to demonstrate the effectiveness of LCR, we also analyzed the influence of two label correlation measurements. For the future work, exploiting more efficient label correlation measurement can further benefit our regularization scheme.

Table 2: Prediction with LCR† on the UCM Multi-label Dataset. Red labels indicate false positives.

image	clean label	plain	LCR†
	bare soil, building, pavement, grass, and tree	airplane, bare soil, building, chaparral, sea, court, dock, sand, field, mobile home, ship, tank, water	bare soil, building, court, field, mobile home, tank, pavement, sand, ship, tree
	bare soil, court, grass, pavement, tree	airplane, bare soil, building, ship, chaparral, court, dock, sea, field, mobile home, sand, car, tank, tree, water	bare soil, chaparral, field, grass, sand, pavement, ship, tank, tree

5. ACKNOWLEDGEMENTS

supported by the China Scholarship Council, the Helmholtz Association (SiPEO, VH-NG-1018, www.sipeo.bgu.tum.de), and the European Research Council (ERC, No. ERC-2016-StG-714087, Acronym: *So2Sat*).

6. REFERENCES

[1] X. X. Zhu, D. Tuia, L. Mou, G. Xia, L. Zhang, F. Xu, and F. Fraundorfer, "Deep learning in remote sensing: A comprehensive review and list of resources," *GRSM*, vol. 5, no. 4, pp. 8–36, 2017.

[2] D. Marcos, M. Volpi, B. Kellenberger, and D. Tuia, "Land cover mapping at very high resolution with rotation equivariant CNNs: Towards small yet accurate models," *ISPRS JPRS*, vol. 145, pp. 96–107, 2018.

[3] L. Mou, Y. Hua, and X. X. Zhu, "A relation-augmented fully convolutional network for semantic segmentation in aerial scenes," in *CVPR*, 2019.

[4] B. Damodaran, K. Fatras, S. Lobry, R. Flamary, D. Tuia, and N. Courty, "Wasserstein adversarial regularization (WAR) for label noise," *arXiv:1904.03936*, 2019.

[5] S. Sukhbaatar and R. Fergus, "Learning from noisy labels with deep neural networks," in *ICLRW*, 2014.

[6] M. Hu, H. Han, S. Shan, and X. Chen, "Multi-label learning from noisy labels with non-linear feature transformation," in *ACCV*, 2018.

[7] Z. Chen, X. Wei, P. Wang, and Y. Guo, "Multi-label image recognition with graph convolutional networks," in *CVPR*, 2019.

[8] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv:1301.3781*, 2013.

[9] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *EMNLP*, 2014.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.

[11] B. Chaudhuri, B. Demir, Subhasis Chaudhuri, and Lorenzo Bruzzone, "Multilabel remote sensing image retrieval using a semisupervised graph-theoretic method," *TGRS*, vol. 56, no. 2, pp. 1144–1158, 2018.

[12] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *ACM SIGSPATIAL)*, 2010.

[13] Timothy Dozat, "Incorporating Nesterov momentum into Adam," http://cs229.stanford.edu/proj2015/054$_r$*eport.pdf*, *Online*.

[14] Y. Hua, L. Mou, and X. X. Zhu, "Recurrently exploring class-wise attention in a hybrid convolutional and bidirectional lstm network for multi-label aerial image classification," *ISPRS JPRS*, vol. 149, pp. 188–199, 2019.