LABEL DISTRIBUTION LEARNING VIA LABEL CORRELATION GRID

Qimeng Guo, Liancheng Xu
Shandong Normal University
School of information science and Engineering

Zhuoran Zheng, Xiuyi Jia
Nanjing University of Science and Technology
School of computer science and engineering

ABSTRACT

Label distribution learning can characterize the polysemy of an instance through label distributions. However, some noise and uncertainty may be introduced into the label space when processing label distribution data due to artificial or environmental factors. To alleviate this problem, we propose a Label Correlation Grid (LCG) to model the uncertainty of label relationships. Specifically, we compute a covariance matrix for the label space in the training set to represent the relationships between labels, then model the information distribution (Gaussian distribution function) for each element in the covariance matrix to obtain an LCG. Finally, our network learns the LCG to accurately estimate the label distribution for each instance. In addition, we propose a label distribution projection algorithm as a regularization term in the model training process. Extensive experiments verify the effectiveness of our method on several real benchmarks.

Index Terms— label distribution learning, covariance matrix, label correlation grid (LCG)

1. INTRODUCTION

In contrast to the traditional single-label learning and multi-label learning, label distribution learning (LDL) [1] aims to characterize instance polysemy with labels of different description degrees. Specifically, LDL extends discrete labels into the form of a continuous probability distribution (description degree). The descriptive degree is deemed as the relevance of the label to the sample, and the sum of the descriptive degree of all labels is 1, showing the relationship between the instance and the sample. For example, a person’s facial expression [2-4] may involve multiple emotions, including happiness, sadness, surprise, etc. LDL can convey such sophisticated and ambiguous semantic information.

Since researchers built label distribution datasets by using manual annotation and label enhancement algorithms [5-8], the label space of the dataset is usually disturbed by the introduction of uncertainty and noise. On the one hand, the manually annotated datasets have a high degree of subjectivity and ambiguity. On the other hand, the inductive bias of the label enhancement algorithm cannot guarantee the accuracy of the label set. The uncertainty of the label set further affects the modeling of label relations. To clear this obstacle, we model the uncertainty of label relations to boost the performance of the LDL algorithm.

Inspired by the idea of modeling the implicit label distribution proposed by [9], our proposed model also implicitly characterizes the label relationship’s uncertainty. So far, we develop a novel approach to learning label distributions via a label correlation grid (LCG), which can model the relationships between labels and learn the implied uncertainty between pairs of labels. In particular, we first use an MLP (multi-layer perceptron) to extract the deep semantics of the input information in order to perform learning label distribution. Then, we use the predicted label distribution to compute a covariance matrix between the labels, with the help of which we generate an LCG. Each cell in the grid is generated by sampling the element values in the covariance matrix using a Gaussian distribution function. Note that this grid is constrained by the grid generated from the label space of the samples with the help of the $L_1$ norm.

In addition, since deep networks are prone to overfitting when learning tabular data, we propose a label distribution projection algorithm as a regularization term that performs on datasets with numerous labels. The label distribution projection algorithm aims to extract the depth semantics of the label space (in this paper, we use KPCA [10]), similar to the perceptual loss [11], which improves the generalization ability of the whole model. Compared to traditional LDL algorithms, our model can execute acceleration on GPU (milliseconds level speed on one inference), which is unparalleled competitive in terms of efficiency and accuracy. Extensive experiments (including the standard accuracy experiment and the noise immunity experiment) have confirmed the effectiveness of our approach. Our main contribution includes three folds: i) We propose a new method to learn label distributions via a label correlation grid that models the uncertainty of the label relationship. ii) We design a robust depth model that combines efficiency (ms level) and accuracy on a tabular dataset. iii) To alleviate the overfitting of the model, we design a label distribution projection algorithm on the dataset with a large number of labels.

2. RELATED WORK

Geng et al. [1] first proposed a new machine learning paradigm: LDL, which focuses on the relative importance of each label
3. PROPOSED METHOD

The architecture of our method is shown in Fig. 1. MLP learns the label distribution based on the deep semantics of the input features. Among them, we conduct three sub-learning targets, a key target is to learn the true label distribution by using $L_1$ loss, followed by learning the LCG and another one (label distribution projection loss) is to learn the deep semantics of the label distribution. Of these, the elements of the LCG come from the Gaussian distribution and the depth semantics of label distribution are from KPCA.

Notations. Given a particular example, the goal of our method is to learn the degree to which each label describes that instance. Input matrix $X \in \mathbb{R}^{m \times n}$, where $m$ is the number of instances and $n$ is the dimension of features. We define the $i$-th instance in the dataset as $x_i$. The label distribution is defined $Y \in \mathbb{R}^{m \times t}$, then the $j$-th label is defined $y_j$. For each instance $x_i$, we define its label distribution $D_i = \{d_{x_i}^1, d_{x_i}^2, ..., d_{x_i}^t\}$ where $d_{x_i}^j$ is the degree of description of the label $y_j$ for the sample $x_i$. The $d_{x_i}^j$ is constrained by $d_{x_i}^j \in [0, 1]$ and $\sum_{j=1}^t d_{x_i}^j = 1$.

Network Architecture. Early studies introduced neural networks into LDL, such as AA-BP [1]. We refer to this network architecture to regress a label distribution. Firstly, the input instance $x_i$ is fed into an MLP with two linear layers. In the first hidden layer, we used 256 neurons to learn the latent features. The number of linear layer neurons in the last layer is the number of labels. Note that the number of neurons is set to the power of 2 in favor of acceleration on the GPU shader. Two non-linear layers (sigmoid and softmax) are added after the hidden layer and the output layer, respectively. The MLP has the ability to learn the latent semantic and global information of the input examples and generates accurate label distribution values. The predicted label description degree $\tilde{d}_{x_i}^j$ of each instance $x_i$ is

$$
\tilde{d}_{x_i}^j = \frac{\exp(\text{MLP}(x_i))}{\sum_{k=1}^t \exp(\text{MLP}(x_k))}.
$$

We optimize the MLP by minimizing the $L_1$ distance between the predicted label distribution and the ground truth as follows:

$$
L_1 \left( d_i, \tilde{d}_i \right) = \sum_{i=1}^T \left\| d_i^y - \tilde{d}_i^y \right\|,
$$

where $\tilde{d}_i$ is the predicted label distribution and $d_i$ is the true label distribution, and $T$ denotes the number of labels. The following describes some key regularization terms.

Label Correlation Grid. Implicit representation [9] has nice properties due to its powerful expressiveness and for this reason, we develop the scheme of implicitly representing label relations. To robustly model label uncertainty, we build a tensor that can accommodate high-dimensional information. Summing up the above thesis, we propose an LCG $y_{grid} \in \mathbb{R}^{t \times t}$. $y_{grid}$ is initially composed of a covariance matrix computed from the label space, where each element is sampled by a Gaussian function to create a vector. The Gaussian
distribution has two hyperparameters and we use each element of the covariance matrix as the mean and the variance is set fixed at 0.5. We also try other values \( \{0.1, 0.2, \ldots, 0.9\} \) as the variance, but the results are not good as seen from the extensive experimental results. The LCG is shown as:

\[
\begin{bmatrix}
  y_1^1 & y_2^1 & \cdots & y_t^1 \\
  y_1^2 & y_2^2 & \cdots & y_t^2 \\
  \vdots & \vdots & \ddots & \vdots \\
  y_1^l & y_2^l & \cdots & y_t^l
\end{bmatrix} \rightarrow [q_1, q_2, \ldots, q_l]
\]

in which \( y_i^j = [q_1, q_2, \ldots, q_l] \) indicates the information between the \( i \)-th label and the \( j \)-th label. The final LCG is optimized by calculating the \( L_1 \) distance. A large number of papers use Gaussian priors for uncertainty estimation of unknown information. So we use the Gaussian process regression [19] to generate the description information.

**Label Distribution Projection Algorithm.** To boost the performance of the model, we propose a label distribution projection algorithm as a regularization term in the network training process. We use a sub-MLP to aggregate feature information into a lower dimension in the network. The sub-MLP consists of one linear layer with sigmoid and softmax as activation functions. For example, the 68-dimensional raw label distribution is mapped to a the 32-dimensional label distribution space by sub-MLP for the Human Gene dataset. The formulaic expression form is as follows:

\[
d_{p_{s_{l+}}} = \text{softmax}(\text{sub-MLP}(d_{p_{s_{l+}}})).
\]

For true low-dimensional label distributions, we use Kernel-based principal component analysis (KPCA) to project the raw true label distribution. KPCA draws on a nonlinear approach to extract the abstract semantics of the label distribution, and in addition, has the effect of reducing noise. The lower label distribution is calculated as \( d_{p_{s_{l+}}} \). We use \( L_1 \) to minimize the output of the network and the true low-dimensional label distribution.

**Loss Function.** The network parameters are optimized by combining \( L_1 \), projection loss \( L_{ldp} \) and label correlation grid loss \( L_{lclg} \). Combined with the content of the above, our target function is constructed as follows:

\[
\text{Loss} = \lambda_1 \times L_1 + \lambda_2 \times L_{ldp} + \lambda_3 \times L_{lclg},
\]

where \( \lambda_1, \lambda_2 \) and \( \lambda_3 \) are parameters. In the four label distribution datasets, we select \( \lambda_1, \lambda_2 \) and \( \lambda_3 \) to be 1, 0.1 and 0.05, respectively.

4. EXPERIMENTS

**Datasets.** To validate the efficiency of our model, we conduct experiments on four LDL datasets. Details of the datasets from Wang et al. [16] are presented in Table 1. The datasets involve different types, including biological experiment datasets, natural scene datasets, movie category datasets, and large-scale biomedical research datasets. The evaluation is based on six indicators proposed by [1], including Chebyshev distance (Chebyshev), Clark distance (Clark), Canberra distance (Canberra), KL divergence (K-L), Cosine similarity (Cosine), and Intersection similarity (Intersection).

**Comparative Algorithms.** We compared the results of our model with five LDL algorithms: DDH-LDL [20], BFGSLLD [1], LDL-LRR [17], LDSL [21] and LALOT [22]. Except for DDH-LDL, the indicators for all datasets of the models are from [17]. Our experiments in DDH-LDL are based on the tips in the code website corresponding to the article.

**Experimental Setting.** For our algorithm, we put the customized selection of parameters for each dataset in Table 2. The parameters that may be used include batch size, learning rate, and epoch. Since the label space of the three datasets Natural Scene, Movie is small, there is not enforced LDP loss function in the training phase. Here, 32 indicates that we squeeze the raw label dimension from 68 to 32 using KPCA, followed immediately by normalization using softmax. Besides, we use the PyTorch framework and AdamW optimizer to train and test a deep network on the GPU shader.

**Results and Discussion.** The experimental results for each dataset are summarized in Table 3, where we take the results of the 10 times 5-fold cross-validation. The experimental results are reported in the form of “mean±std”. ↑ indicates that the higher the value of the metric, the better. ↓ indicates that the smaller the value of the metric, the better.

Our algorithm performs optimally on six parameters across the four datasets, particularly on the metric Clark. Our approach also achieves competitive results in the remaining metrics. This is mainly due to the ability of the algorithm to expand label relationships and mine feature semantic information. Besides, we observe that algorithms that exploit label relations, like LDL-LRR and ours, generally outperform other algorithms, which demonstrates the importance of label relations in label distribution. Our method has a better performance compared to LDL-LRR by taking into account the

| Dataset         | Examples | Features | Labels |
|-----------------|----------|----------|--------|
| Human Gene      | 17892    | 36       | 68     |
| Natural Scene   | 2000     | 294      | 9      |
| Yeast-alpha     | 2465     | 24       | 18     |
| Movie           | 7755     | 1869     | 5      |

| Dataset         | Batch size | Learning rate | Epoch | LDP loss |
|-----------------|------------|---------------|-------|----------|
| Human Gene      | 500        | 0.0005        | 400   | 32       |
| Natural Scene   | 500        | 0.0005        | 300   | -        |
| Yeast-alpha     | 500        | 0.0005        | 400   | 16       |
| Movie           | 2000       | 0.0005        | 100   | -        |
uncertainty of the label relationship. Although our modeling has an expensive cost, a GPU acceleration can be enforced in the PyTorch to alleviate this problem. We evaluate the inference speed of our model on CPU (AMD 8-core processor) and GPU (RTX3080) as 22ms and 1ms, respectively.

**Ablation Studies.** We conduct ablation studies to demonstrate the effectiveness of LCG and label distribution projection algorithms on dataset Human gene. The results of simple ablation experiments are shown in Table 4. w/o grid is an algorithm for removing the LCG and w/o L_{dp} is defined when the loss term L_{dp} is removed. The variation of the data illustrates the effectiveness of the LCG and the label distribution projection algorithms. We conduct 10 times 5-fold cross-validation on the dataset of the ablation studies. The gray markers represent optimal performance.

**Noise Interference.** To verify the robustness of our algorithm against the noise and uncertainty in label space, we experimented on the dataset Movie with Gaussian noise. We synthesize a dataset on Movie by Gaussian function to enforce different degrees of noise, and the labels with noise are finally normalized by softmax. For label space on the dataset, the variance of the Gaussian distribution function is selected from (0, 1, 0.2, ..., 1).

It is widely believed that the higher the variance, the more dispersed the data, and the more noise. We conduct 10 times 5-fold cross-validation on the synthetic dataset. Our algorithm yield 0.1257±0.0071 (Chebyshev), 0.5713±0.0027 (Clark), 1.0854±0.0055 (Canberra), 0.1452±0.0033 (K-L), 0.9250±0.0092 (Cosine), 0.8203±0.0037 (Intersection) respectively on the data set (the variance is 1) with Gaussian noise has a variance of 1 and mean of 0. However, the accuracy of the label distribution predicted by other algorithms decreases by 50% on average, and our model is still very competitive.

5. CONCLUSION

Ambiguity and uncertainty of label space in LDL become major challenges, limiting the performance of LDL models. In this paper, we propose a new algorithm for label distribution by using an LCG. In standard evaluation experiments, our method achieves competitive results, especially on the metric of vector similarity rather than distance. Furthermore, our method has a striking performance on the mini-test of noise resistance, thanks to the gains from uncertainty modeling. Our model can execute GPU acceleration on processing LDL data, balancing accuracy and efficiency.
6. REFERENCES

[1] Xin Geng, “Label distribution learning,” *TKDE*, 2016.

[2] Zengqun Zhao, Qingshan Liu, and Feng Zhou, “Robust lightweight facial expression recognition network with label distribution training,” in *AAAI*, 2021.

[3] Shijing Si, Jianzong Wang, Junqing Peng, and Jing Xiao, “Towards speaker age estimation with label distribution learning,” in *ICASSP*, 2022.

[4] Jingying Chen, Chen Guo, Ruyi Xu, Kun Zhang, Zongkai Yang, and Honghai Liu, “Toward children’s empathy ability analysis: Joint facial expression recognition and intensity estimation using label distribution learning,” *TII*, 2022.

[5] Xingyu Zhao, Yuexuan An, Ning Xu, and Xin Geng, “Fusion label enhancement for multi-label learning,” in *IJCAI*, 2022.

[6] Qinghai Zheng, Jihua Zhu, Haoyu Tang, Xinyuan Liu, Zhongyu Li, and Huimin Lu, “Generalized label enhancement with sample correlations,” *TKDE*, 2020.

[7] Ning Xu, Jun Shu, Yun-Peng Liu, and Xin Geng, “Variational label enhancement,” in *ICML*, 2020.

[8] Ning Xu, Yun-Peng Liu, and Xin Geng, “Label enhancement for label distribution learning,” *TKDE*, 2021.

[9] Zhuoran Zheng and Xiuyi Jia, “Label distribution learning via implicit distribution representation,” *arXiv preprint arXiv:2209.13824*, 2022.

[10] Bernhard Schölkopf, Alexander Smola, and Klaus-Robert Müller, “Nonlinear component analysis as a kernel eigenvalue problem,” *Neural Computation*, 1998.

[11] Leon A Gatys, Alexander S Ecker, and Matthias Bethge, “Image style transfer using convolutional neural networks,” in *CVPR*, 2016.

[12] Deyu Zhou, Xuan Zhang, Yin Zhou, Quan Zhao, and Xin Geng, “Emotion distribution learning from texts,” in *EMNLP*, 2016.

[13] Ying Zhou, Hui Xue, and Xin Geng, “Emotion distribution recognition from facial expressions,” in *ACM MM*, 2015.

[14] Jing Wang and Xin Geng, “Classification with label distribution learning,” in *IJCAI*, 2019.

[15] Tingting Ren, Xiuyi Jia, Weiwei Li, and Shu Zhao, “Label distribution learning with label correlations via low-rank approximation,” in *IJCAI*, 2019.

[16] Jing Wang and Xin Geng, “Label distribution learning by exploiting label distribution manifold,” *TNNLS*, 2021.

[17] Xiuyi Jia, Xiaoxia Shen, Weiwei Li, Yunan Lu, and Jihua Zhu, “Label distribution learning by maintaining label ranking relation,” *TKDE*, 2021.

[18] Wenbin Qian, Yinsong Xiong, Jun Yang, and Wenhao Shu, “Feature selection for label distribution learning via feature similarity and label correlation,” *Information Sciences*, 2022.

[19] Andrew Gordon Wilson, David A Knowles, and Zoubin Ghahramani, “Gaussian process regression networks,” *arXiv preprint arXiv:1110.4411*, 2011.

[20] Zhen Zhang, Lei Zhu, Yaping Li, and Yang Xu, “Deep discrete hashing for label distribution learning,” *IEEE Signal Process. Lett.*, 2022.

[21] Tingting Ren, Xiuyi Jia, Weiwei Li, Lei Chen, and Zechao Li, “Label distribution learning with label-specific features,” in *IJCAI*, 2019.

[22] Peng Zhao and Zhi-Hua Zhou, “Label distribution learning by optimal transport,” in *AAAI*, 2018.