

# Supplementary Materials: Imbalanced Multi-instance Multi-label Learning via Coding Ensemble and Adaptive Thresholds

Xinyue Zhang

National University of Defense Technology  
Changsha, China  
zhang0331zxy@163.com

Yueying Liu

National University of Defense Technology  
Changsha, China  
liuyueyingnudu@163.com

Tingjin Luo\*

National University of Defense Technology  
Changsha, China  
tingjinluo@hotmail.com

Chenping Hou

National University of Defense Technology  
Changsha, China  
hcpnudu@hotmail.com

In supplementary materials, we mainly provide the appendix to explain some details that are not explicitly covered in the paper.

## A Notations

In the paper, matrices and vectors are written as boldface upper-case letters and italic boldface lowercase letters, respectively. For a matrix  $\mathbf{A} = [a_{ij}]$ , its  $i$ -th row and  $j$ -th column are denoted by  $\mathbf{A}(i, :)$  and  $\mathbf{A}(:, j)$ , respectively. Notations used in the paper are summarized in Table S1.

**Table S1: Notations.**

Notations	Description
$d_1$	Dimension of the representation of a bag
$d_2$	Dimension of the instance
$T_1$	Number of iterations for the Gaussian mixture model
$T_2$	Number of iterations for training the base classifier
$M$	Number of bags for training
$n_i$	Number of instances in the $i$ -th bag
$C$	Dimension of the label space
$N_I$	Number of all instances in training bags
$\mathbf{B}_i \subseteq \mathcal{X}$	Feature matrix of the $i$ -th bag
$\mathbf{x}_i^j \in \mathcal{X}$	The $j$ -th instance of the $i$ -th bag
$\mathbf{Y}_i \in \mathcal{Y}$	One-hot code of the $i$ -th bag
$y_i^l \in \mathcal{Y}$	The $l$ -th binary label of the $i$ -th bag
$z_i^l$	Code of the $l$ -th label
$\mathbf{Z}$	Encoding matrix
$\mathbf{t}^i$	Bag-label mismatching degrees vector of the $i$ -th bag
$U$	The constant upper bound of the parameter

## B Method

### B.1 Details of Efficient Bag Embedding Method

In MIML tasks, the  $i$ -th bag  $\mathbf{B}_i$  is formed by a collection of instances  $\{\mathbf{x}_i^1, \mathbf{x}_i^2, \dots, \mathbf{x}_i^{n_i}\}$ . As one can approximate with arbitrary precision any continuous distribution with the Gaussian mixture model (GMM) [8], we assume instances in  $\mathbf{B}_i$  are independently

and identically distributed and generated from GMM  $p$  consisting of  $G$  component with parameter  $\theta$ , i.e.,

$$p(\mathbf{x}_i^j | \theta) = \sum_{g=1}^G \alpha_g p_g(\mathbf{x}_i^j | \theta), \quad (\text{S1})$$

where  $\alpha_g \geq 0$  is the non-negative weight and satisfy the constraints  $\sum_{g=1}^G \alpha_g = 1$ , and  $p_g(\mathbf{x}_i^j | \theta)$  is the  $g$ -th Gaussian model. We denote the parameters of the  $g$ -component GMM by  $\theta = \{\alpha_g, \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g, g = 1, 2, \dots, G\}$ , where  $\boldsymbol{\mu}_g$  and  $\boldsymbol{\Sigma}_g$  are respectively the mean vector and covariance matrix of Gaussian  $g$ . To ensure that  $p(\mathbf{x}_i^j | \theta)$  is a valid distribution,  $\boldsymbol{\Sigma}_g$  is assumed to be diagonal, and diagonal entries form the vector  $\boldsymbol{\sigma}_g^2$  [10]. The above parameters can be estimated on training bags by maximum likelihood estimation (MLE).

Parameter  $\theta$  contains important statistics that provide the distribution characteristics of the instances. To contain as many of these statistics as possible in the bag representation and display as fully of these characteristics as possible,  $\mathbf{B}_i$  consisting of  $n_i$  instances  $\{\mathbf{x}_i^1, \mathbf{x}_i^2, \dots, \mathbf{x}_i^{n_i}\}$  is represented by the gradient of the log-likelihood of the GMM  $p$ ,

$$\begin{aligned} \mathbf{G}_\theta^{\mathbf{B}_i} &= \nabla_\theta \log p(\mathbf{B}_i | \theta) = \nabla_\theta \log p(\mathbf{x}_i^1, \mathbf{x}_i^2, \dots, \mathbf{x}_i^{n_i} | \theta) \\ &= \nabla_\theta \log \prod_{j=1}^{n_i} p(\mathbf{x}_i^j | \theta) = \sum_{j=1}^{n_i} \nabla_\theta \log p(\mathbf{x}_i^j | \theta), \end{aligned} \quad (\text{S2})$$

where gradients of  $\mathbf{G}_\theta^{\mathbf{B}_i}$  w.r.t. GMM model parameters  $\theta = \{\alpha_g, \boldsymbol{\mu}_g, \boldsymbol{\Sigma}_g, g = 1, 2, \dots, G\}$  are

$$\nabla_{\alpha_g} \log p(\mathbf{x}_i^j | \theta) = p(g | \mathbf{x}_i^j, \theta) - \alpha_g, \quad (\text{S3})$$

$$\nabla_{\boldsymbol{\mu}_g} \log p(\mathbf{x}_i^j | \theta) = p(g | \mathbf{x}_i^j, \theta) \left( \frac{\mathbf{x}_i^j - \boldsymbol{\mu}_g}{\boldsymbol{\sigma}_g^2} \right), \quad (\text{S4})$$

$$\nabla_{\boldsymbol{\sigma}_g} \log p(\mathbf{x}_i^j | \theta) = p(g | \mathbf{x}_i^j, \theta) \left[ \frac{(\mathbf{x}_i^j - \boldsymbol{\mu}_g)^2}{\boldsymbol{\sigma}_g^3} - \frac{1}{\boldsymbol{\sigma}_g} \right], \quad (\text{S5})$$

where  $p(g | \mathbf{x}_i^j, \theta)$  is the probability of  $\mathbf{x}_i^j$  being generated by the  $g$ -th Gaussian,

$$p(g | \mathbf{x}_i^j, \theta) = \frac{\alpha_g p_g(\mathbf{x}_i^j | \theta)}{\sum_{g=1}^G \alpha_g p_g(\mathbf{x}_i^j | \theta)}. \quad (\text{S6})$$

\*Corresponding author

**Algorithm 1:** Optimization of IMIMLC

---

**Input** : Training set  $\{(\mathbf{B}_1, \mathbf{Y}_1), \dots, (\mathbf{B}_M, \mathbf{Y}_M)\}$ .  
**Para** : Number of Gaussian models  $G$ ;  
Parameters in GMM  $\theta = \{\alpha_g, \mu_g, \Sigma_g\}$ ;  
Number of labels in subset  $k$ ;  
Size of the subtask  $d$ .  
**Output**: Classifiers  $h_j(\cdot)$  and  $\omega_l$ .

---

```

1 Estimate parameters  $\theta = \{\alpha_g, \mu_g, \Sigma_g\}$  by MLE.
2 for  $i = 1$  to  $M$  do
3   for  $j = 1$  to  $n_i$  do
4     Calculate  $p(g|x_i^j, \theta)$  by Eq. (S6).
5   end
6   for  $g = 1$  to  $G$  do
7     Calculate each part of  $\tilde{x}_i$  by Eq. (S7).
8   end
9   Normalize  $\tilde{x}_i$  according to Eq. (S8).
10 end
11 for  $i = 1$  to  $d$  do
12   Randomly and non-repetitively choose
13   subtask  $w_i$  from  $Q^k$ .
14   Encode  $Z^{(i)}$  using the OVA strategy for  $w_i$ .
15 end
16 Concatenate  $Z^{(i)}$  to obtain  $Z$ .
17 for  $j = 1$  to  $kd$  do
18   Train the base classifier  $h_j(\cdot)$  based on  $Z(j, :)$ .
19 end
20 for  $i = 1$  to  $M$  do
21    $z_i \leftarrow [h_1(\tilde{x}_i), h_2(\tilde{x}_i), \dots, h_{kd}(\tilde{x}_i)]$ 
22   Calculate the soft label set of  $\mathbf{B}_i$  by Eq. (3).
23 end
24 Train  $\omega_l (l = 1, 2, \dots, C)$  by Eq. (6) and Eq. (7).
```

---

Then, to ensure the stability and convergence of the algorithm and remove redundant information, we conduct whitening on  $G_\theta^{\mathbf{B}_i}$  using the fisher information matrix (FIM)  $\mathbf{F}_\theta = E_{\mathbf{x} \sim p}[\nabla_\theta \log p(\mathbf{B}_i|\theta) \nabla_\theta \log p(\mathbf{B}_i|\theta)^T]$ . By performing the Cholesky decomposition on FIM for  $\mathbf{L}_\theta$ , the bag representation  $G_\theta^{\mathbf{B}_i}$  in Eq. (S2) can be redefined as the fisher vector

$$\begin{aligned} \tilde{x}_i &= \mathbf{L}_\theta G_\theta^{\mathbf{B}_i} \\ &= [\tilde{x}_{\alpha_1}^{\mathbf{B}_i}, \dots, \tilde{x}_{\alpha_G}^{\mathbf{B}_i}, \tilde{x}_{\mu_1}^{\mathbf{B}_i}, \dots, \tilde{x}_{\mu_G}^{\mathbf{B}_i}, \tilde{x}_{\sigma_1}^{\mathbf{B}_i}, \dots, \tilde{x}_{\sigma_G}^{\mathbf{B}_i}], \end{aligned} \quad (\text{S7})$$

where for each  $g \in \{1, 2, \dots, G\}$ , each part of the fisher vector can be calculated as  $\tilde{x}_{\alpha_g}^{\mathbf{B}_i} = \frac{1}{\sqrt{\alpha_g}} \sum_{j=1}^{n_i} \nabla_{\alpha_g} \log p(x_i^j|\theta)$ ,  $\tilde{x}_{\mu_g}^{\mathbf{B}_i} = \frac{1}{\sqrt{\alpha_g}} \sum_{j=1}^{n_i} \nabla_{\mu_g} \log p(x_i^j|\theta)$  and  $\tilde{x}_{\sigma_g}^{\mathbf{B}_i} = \frac{1}{\sqrt{2\alpha_g\sigma_g}} \sum_{j=1}^{n_i} \nabla_{\sigma_g} \log p(x_i^j|\theta)$ , respectively.

Finally, similar with [5, 11, 12], we normalize  $\tilde{x}_i$  to reduce the dependence of the variance on the mean and remove that on the proportion of object-specific information by the following approach

$$\begin{cases} [\tilde{x}_i]_j \leftarrow \text{sign}([\tilde{x}_i]_j) \sqrt{|[\tilde{x}_i]_j|}, \\ \tilde{x}_i \leftarrow \frac{\tilde{x}_i}{\sqrt{\tilde{x}_i^T \tilde{x}_i}}. \end{cases} \quad (\text{S8})$$

**B.2 Algorithm**

The main procedure of our IMIMLC is summarized in Algorithm 1.

**B.3 Computational Complexity**

The computational complexity of IMIMLC can be analyzed in two parts: (1) bag vector representation based on instances, and (2) imbalanced learning based on the coding ensemble and adaptive thresholds. For the first one, the cost to fit  $N_I$  instances in training bags is  $O(N_I G d_2 T_1)$ . Subsequently, each bag will be mapped into a new vector, whose cost is  $O(N_I G d_2)$ . In short, the computational cost of the first part is  $O(N_I G d_2 (T_1 + 1))$ . In the second part, the algorithm needs to perform processes of “randomly selecting subtasks”, “training base classifiers”, “decoding”, and “adapting thresholds” in turn. We estimate their complexity as  $O(kd(T_2 N_s^3 + N_s d_1 + 3MC) + CN_s^3)$ , where  $N_s$  is the number of training bags that really matter to the process of adapting. However, because it is difficult to make a formal complexity analysis of other comparison algorithms, we empirically validated the efficiency and scalability of our proposed method in experiments.

**C Experiment Details****C.1 Datasets**

The details of related datasets in Table 1 are reported as follows:

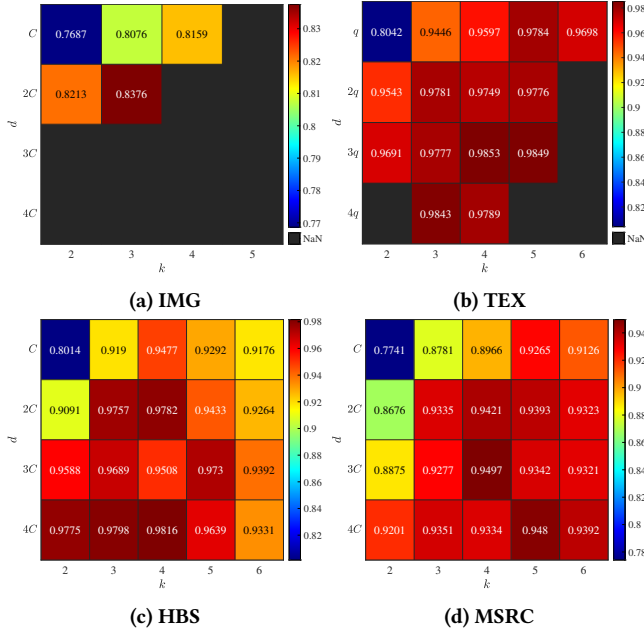
**MIML-image (IMG)** and **MIML-text (TEX)** were collected by Zhou et al. [13, 14] and are the most commonly used datasets in existing MIML tasks. IMG contains 2000 bags. Each one is extracted from a scene image by SBN [7] and may be associated with five labels: desert, mountains, sea, sunset, and trees. TEX is derived from the REUTERS-21578 dataset, and only the 7 most frequent categories are considered. Each document is extracted as a bag with instances using the method proposed in [1]. **HJA Bird Song (HBS)**<sup>1</sup> consists of audio recordings of bird songs at H. J. Andrews Experimental Forest using unattended microphones. The dataset contains 548 10-second audio recordings, each of which may be associated with 13 labels. **MSRC v2 (MSRC)**<sup>2</sup> is the latest version of the image dataset proposed by Microsoft Research Cambridge. Each image comes with its ground-truth segmentation, so histograms of gradients and colors can be extracted to form an instance for each region in segmentation. **Letter Carroll (LC)** was gathered by Fern et al. [2] and generated using the poem, whose name is “Jabberwocky”, from the UCI Letter Recognition dataset [4]. The black-and-white image of each word in this poem corresponds to a bag and displays some of the 26 letters in the English alphabet. **Isoform Gene Data (IGD)** collected by Luo et al. [6, 9] was generated from a total of 573 human RNA-seq runs of the ENCODE project [3]. It consists of 11,946 genes, each of which has a different number of isoforms, from one to fifteen. There are a total of 94 labels or functions in this dataset, as each gene may contain one or several labels or functions.

<sup>1</sup><https://paperswithcode.com/dataset/birdsong>

<sup>2</sup><https://www.microsoft.com/en-us/research/project/image-understanding/>

**Table S2: Comparisons of classification performance with different thresholds on the HBS, MSRC, and LC datasets.**

Datasets	Metrics	$\eta_p = 0.5$	$\eta_p = 0.6$	$\eta_p = 0.7$	$\eta_p = 0.8$	$\eta_p = 0.9$	$\eta_A$
HBS	HL ( $\downarrow$ )	.633 $\pm$ .003	.231 $\pm$ .002	.112 $\pm$ .001	.056 $\pm$ .001	.090 $\pm$ .001	<b>.025<math>\pm</math>.001</b>
	ACC ( $\uparrow$ )	.367 $\pm$ .003	.769 $\pm$ .002	.888 $\pm$ .001	.944 $\pm$ .001	.910 $\pm$ .001	<b>.975<math>\pm</math>.001</b>
	F1 ( $\uparrow$ )	.330 $\pm$ .001	.570 $\pm$ .003	.726 $\pm$ .005	.812 $\pm$ .008	.531 $\pm$ .005	<b>.842<math>\pm</math>.012</b>
MSRC	HL ( $\downarrow$ )	.817 $\pm$ .002	.394 $\pm$ .002	.176 $\pm$ .002	.088 $\pm$ .001	.070 $\pm$ .001	<b>.042<math>\pm</math>.001</b>
	ACC ( $\uparrow$ )	.183 $\pm$ .002	.606 $\pm$ .002	.824 $\pm$ .002	.912 $\pm$ .001	.930 $\pm$ .001	<b>.958<math>\pm</math>.001</b>
	F1 ( $\uparrow$ )	.194 $\pm$ .000	.316 $\pm$ .002	.498 $\pm$ .005	.602 $\pm$ .007	.504 $\pm$ .011	<b>.704<math>\pm</math>.007</b>
LC	HL ( $\downarrow$ )	.369 $\pm$ .002	.236 $\pm$ .003	.146 $\pm$ .003	.088 $\pm$ .002	.089 $\pm$ .002	<b>.044<math>\pm</math>.003</b>
	ACC ( $\uparrow$ )	.631 $\pm$ .002	.764 $\pm$ .003	.854 $\pm$ .003	.912 $\pm$ .002	.911 $\pm$ .002	<b>.956<math>\pm</math>.003</b>
	F1 ( $\uparrow$ )	.463 $\pm$ .008	.559 $\pm$ .010	.626 $\pm$ .010	.625 $\pm$ .013	.445 $\pm$ .005	<b>.777<math>\pm</math>.022</b>

**Figure S1: Sensitivity analysis with different  $k$  and  $d$ . Missing location is caused by the constraint  $d \leq \mathbb{C}_C^k$ .**

## C.2 Experiment Settings

All of the experiments were conducted on a machine with an i7-8650U CPU and 16.0 GB of RAM. In experiments, hyper-parameters of IMIMLC include label subset size  $k$  and selected subtask number  $d$  etc. Tuning ranges of hyper-parameters  $\lambda$  and  $\gamma$  in the kernel function were both from 0.0001 to 5. For other hyper-parameters of each MIML algorithm, tuning ranges are set the same as in the original references. The final optimal values were selected by incorporating grid search and cross-validation strategies.

## C.3 Ablation Study

In Table 4, we report classification performance with different thresholds on the TEX datasets. In this section, more results on the HBS, MSRC, and LC datasets are displayed.

## C.4 Hyper-parameter Sensitivity Analysis

In Fig. 4, we report AUCs based on different parameter combinations of the LC and IGD datasets. In this section, more results on other datasets are shown.

## References

- [1] Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann. 2002. Support Vector Machines for Multiple-Instance Learning. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, Vol. 15. MIT Press, Vancouver, Canada, 1–8.
- [2] Forrest Briggs, Xiaoli Z. Fern, and Raviv Raich. 2012. Rank-loss support instance machines for MIML instance annotation. In *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (KDD)*. Association for Computing Machinery, Beijing, China, 534–542.
- [3] ENCODE Project Consortium. 2012. An integrated encyclopedia of DNA elements in the human genome. *Nature* 489, 7414 (2012), 57–74.
- [4] Peter W. Frey and David J. Slate. 1991. Letter recognition using Holland-style adaptive classifiers. *Machine Learning* 6, 2 (1991), 161–182.
- [5] Hervé Jégou, Florent Perronnin, Matthijs Douze, Jorge Sánchez, Patrick Pérez, and Cordelia Schmid. 2012. Aggregating Local Image Descriptors into Compact Codes. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34, 9 (2012), 1704–1716.
- [6] Tingjin Luo, Weizhong Zhang, Shuang Qiu, Yang Yang, Dongyun Yi, Guangtao Wang, Jieping Ye, and Jie Wang. 2017. Functional annotation of human protein coding isoforms via non-convex multi-instance learning. In *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (KDD)*. Association for Computing Machinery, Halifax, Nova Scotia, Canada, 345–354.
- [7] Oded Maron and Aparna Lakshmi Ratan. 1998. Multiple-instance learning for natural scene classification. In *Proceedings of the International Conference on Machine Learning (ICML)*, Vol. 98. Association for Computing Machinery, Madison, WI, USA, 341–349.
- [8] J. Newton. 1986. Statistical analysis of finite mixture distributions. *Journal of the International Biometric Society* 42, 3 (1986), 679–680.
- [9] Bharat Panwar, Rajasree Menon, Ridvan Eksi, Hongdong Li, Gilbert S Omenn, and Yuanfang Guan. 2016. Genome-wide functional annotation of human protein-coding splice variants using multiple instance learning. *Journal of Proteome Research* 15, 6 (2016), 1747–1753.
- [10] Florent Perronnin and Christopher Dance. 2007. Fisher Kernels on Visual Vocabularies for Image Categorization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, Minneapolis, MN, USA, 1–8.
- [11] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. 2010. Improving the fisher kernel for large-scale image classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, Hersonissos, Heraklion, Crete, Greece, 143–156.
- [12] Jorge Sánchez, Florent Perronnin, Thomas Mensink, and Jakob Verbeek. 2013. Image Classification with the Fisher Vector: Theory and Practice. *International Journal of Computer Vision* 105, 3 (2013), 222–245.
- [13] Minling Zhang and Zhihua Zhou. 2008. M3MIML: A maximum margin method for multi-instance multi-label learning. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*. IEEE, Pisa, Italy, 688–697.
- [14] Zhili Zhang and Minling Zhang. 2006. Multi-instance Multi-label Learning with Application to Scene Classification. In *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS)*, Vol. 19. MIT Press, Vancouver, Canada, 1609–1616.