Ensemble Multi-Instance Multi-Label Learning Approach for Video Annotation Task*

Xin-Shun Xu[1,2] xuxs@lamda.nju.edu.cn
Xiangyang Xue[3] xyxue@fudan.edu.cn
Zhi-Hua Zhou[1] zhouzh@lamda.nju.edu.cn

[1]National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210093, China
[2]School of Computer Science and Technology, Shandong University, Jinan 250101, China
[3]School of Computer Science, Fudan University, Shanghai 200433, China

ABSTRACT

Automatic video annotation is an important ingredient for video indexing, browsing, and retrieval. Traditional studies represent one video clip with a flat feature vector; however, video data usually has natural structure. Moreover, a video clip is generally relevant to multiple concepts. Indeed, the video annotation task is inherently a Multi-Instance Multi-Label (MIML) learning problem. In this paper, we propose the En-MIMLSVM approach for the video annotation task. It considers the class imbalance and long time training problems of most video annotation tasks. In addition, a temporally consistent weighted multi-instance kernel is developed to take into account both the temporal consistency in video data and the significance of instances of different levels in pyramid representation. The En-MIMLSVM is evaluated on TRECVID 2005 data set, and the results show that it outperforms several state-of-the-art methods.

Categories and Subject Descriptors

I.2.10 [**Artificial Intelligence**]: Vision and Scene Understanding—*video analysis*; I.2.6 [**Artificial Intelligence**]: Learning

General Terms

Algorithms, Design, Experimentation

Keywords

Video Annotation, Multi-Instance Multi-Label Learning, Ensemble Methods

1. INTRODUCTION

With the rapid development of storages devices, networks and compression techniques, large collections of digital videos are available publicly. How to effectively access large-scale video data, such as indexing, browsing and retrieval, has become a challenging task, and attracted more and more researchers. To tackle this issue, a common theme is to first annotate these videos with concepts which describe the information in the video content at semantic level, and then use these concepts to index or browse the video. Generally, it is labor-intensive and time-consuming to annotate large video data archives. Thus, annotating videos automatically (it is also called concept detection or high level feature extraction) has emerged as an active topic in the multimedia research community [12, 16].

A general approach is to first extract textual and/or visual features, and then train a model and use the trained model to predict concepts of new videos. Traditional methods represent each sample as a flat feature vector, and apply machine learning techniques such as support vector machine (SVM) [1, 21] and Hidden Markov Model [4] to construct classifiers which are then used for annotation. Recently, some researchers try to use structure-based representation, which represents each sample in a form beyond a flat feature vector, such as pyramid representation. In particular, multi-instance (MI) representation, which was originated from research on multi-instance learning [2], has attracted more and more attention. In MI, each sample is represented as a *bag* of instances, that is, feature vectors in the same feature space. In its standard form, it is assumed that a bag is positive if it contains at least one positive instance, and otherwise it is a negative bag. The relation between instances in each bag plays important roles [25]. MI has already been applied in many studies on video or image annotation tasks. For example, Naphade et al. [11] proposed a generalized multi-instance learning algorithm for video concept detection, Gu et al. [6] formulated the task as a multi-layer multi-instance learning problem, van de Sande et al. [18] represented each keyframe as a pyramid, and designed a kernel to measure the similarity between two keyframes.

It is important to notice that a keyframe usually contains more than one concept. For example, in TRECVID 2005 data set, according to LSCOM-Lite annotations [10], there are 39 concepts, and the largest number of concepts contained in one frame is 12. There are many studies trying to handle multi-labeled output. Generally, they try to annotate the video concepts separately. For example, one-vs-rest strategy can be used to train an SVM for each concept, and then the performance can be improved by techniques such as context-based concept fusion [12, 15]. The individual concepts and their correlations can also be treated simultaneously [14].

Recently, Zhou and Zhang [26] proposed the MIML (Multi-Instance Multi-Label learning) framework, where one object is represented by a bag of instances and the object is allowed to have multiple labels simultaneously. Such a framework is able to take the correlation among instances, correlation among labels, and correlation between instances and labels simultaneously, and provides a very rich representation and learning potential. It is evident that the video and image annotation tasks are naturally and inherently MIML tasks, and successful video and image annotation has al-

*Area Chair: Kiyoharu Aizawa. This research was supported by NSFC (61073097, 60970047, 61173068), 973 Program (2010CB327903), CPSF (20100470063), IIFSDU (2009TS033).

MM'11, November 28–December 1, 2011, Scottsdale, Arizona, USA.

ready been developed in the MIML framework [8, 22]. A number of MIML algorithms have been developed during the past few years. For example, the MIMLBoost and MIMLSVM algorithms [26] solve MIML problems by degeneration. The D-MIMLSVM algorithm which solves MIML problems in regularization framework considers the inherent class-imbalance in multiple labels [27]. The M3MIML algorithm is a fast large-margin MIML approach [23]. There are also MIML distance metric learning approach [7], MIML approach based on Dirichlet-Bernoulli Alignment [20], and recently a powerful degeneration approach SISL-MIML [13].

By training and combining multiple classifiers, ensemble methods [24] are state-of-the-art techniques achieving strong generalization ability. In this paper, we propose the En-MIMLSVM approach, an ensemble multi-instance multi-label learning algorithm based on MIMLSVM. Inspired by [9], this method first samples several subsets from majority class independently, then trains multiple classifiers using the subsets and minority class, and finally combines all classifiers for final decision. It is able to exploit the majority class examples ignored by under-sampling, and make the training speed faster due to the small size training set. Thus, it can deal with the class imbalance problem and the long learning time of SVM simultaneously. Experiments on TRECVID2005 data set show that the proposed approach outperforms a number of state-of-the-art methods.

The rest of this paper is organized as follows: In section 2, we introduce the En-MIMLSVM method. In section 3, we report on experiments. Finally, we conclude this paper in section 4.

2. THE EN-MIMLSVM APPROACH

2.1 Basic Approach

There is serious class imbalance in most video annotation tasks. For instance, there are only 106 positive samples of concept "Prisoner" among 61901 samples in TRECVID 2005 development data set. In addition, training an SVM is usually posed as a quadratic programming problem, and becomes challenging when there are a large number of samples due to its high computational and memory requirement.

To deal with the problems above, under-sampling is an efficient method. It uses a subset of majority class samples to train a classifier. Although the training set becomes balanced and the training process becomes faster, standard under-sampling often suffers from the loss of helpful information concealed in the ignored majority class samples. In [9], Liu et al. proposed an ensemble method; rather than randomly removing majority class samples, this method tries to remove samples which have already been learned well, and thus reduces the risk of neglecting samples containing important information. Inspired by [9], our En-MIMLSVM method first samples several subsets from majority class independently; then AdaBoost is used to train a number of weighted SVMs from the union of each subset and the minority class, and finally all classifiers are combined for final decision.

The pseudo-code for En-MIMLSVM is shown in Algorithm 1. Here, for each class, SVM is used as base learner of an ensemble. In this paper, we use kernel which can capture both the structure and temporal information in the videos. The kernel is detailed in the next section.

Algorithm 1 En-MIMLSVM

1. For each class(concept)Input: In the training set, select all the positive samples $\mathcal{P}$, and all the negative samples $\mathcal{N}$,$|\mathcal{P}| < |\mathcal{N}|$, the number of subsets T to sample from $\mathcal{N}$, and s_i, the number of iterations to train an AdaBoost ensemble H_i
2. $i \Leftarrow 0$
3. repeat
4. $\quad i \Leftarrow i + 1$
5. $\quad$ Randomly sample a subset $\mathcal{N}_i$ from $\mathcal{N}$, $|\mathcal{N}_i| = |\mathcal{P}|$.
6. $\quad$ Learn H_i using P and N_i. H_i is an AdaBoost ensemble with s_i weak classifier h_{ij} and corresponding weights α_{ij}.
   a. $j \Leftarrow 0$
   b. Input: a set of the weights of training samples:$w_{i1}^k \Leftarrow 1/(2 * |\mathcal{P}|), k \Leftarrow 1, ..., l = 2 * |\mathcal{P}|$
   c. repeat
   d. $\quad j \Leftarrow j + 1$
   e. $\quad$ Train an SVM component classifier, h_{ij} on the weighted training set.
   f. $\quad$ Calculate the training error of $h_{ij} : \varepsilon_{ij} \Leftarrow \sum w_i^j, y_i \neq h_{ij}(x_i)$.
   g. $\quad$ Set the weight of component classifier. $h_{ij} : \alpha_{ij} \Leftarrow \frac{1}{2} ln(\frac{1-\varepsilon_{ij}}{\varepsilon_{ij}})$
   h. $\quad$ Update the weights of training samples if $j \neq s_i$: $w_i^{j+1} \Leftarrow \frac{w_i^j exp(-\alpha_{ij} y_i h_{ij}(x_j))}{C_i}$ where C_i is a normalization constant, and $\sum w_i^{j+1} = 1$
   i. until $j = s_i$
7. $\quad$ Suppose the ensemble's threshold is θ_i, then the ensemble is: $H_i(x) = sgn(\sum_{j=1}^{s_i} \alpha_{ij} h_{ij}(x) - \theta_i)$
8. until $i = T$
9. Output: An ensemble:
   $H(x) = sgn(\sum_{i=1}^{T} \sum_{j=1}^{s_i} \alpha_{ij} h_{ij}(x) - \sum_{i=1}^{T} \theta_i)$

2.2 The Kernel Design

We represent each frame with pyramid style, and treat each patch as an instance. Several kernels have been designed for spatial pyramid representation [5, 18]; however, they usually compute the similarity between images/keyframes by using bin-by-bin method, and could not find the similarity between bins with different spatial positions. In addition, these kernels did not consider the property of temporal consistency in video data which has been shown very important in video annotation task [17, 19]. In order to leverage the spatial information in pyramid representation and the temporal consistency in video data, we design a new temporally consistent kernel.

Suppose that the bags of two keyframes are represented as:

$$P = \{p_1^1, p_2^1, ..., p_{m_1}^1, p_1^2, ..., p_{m_2}^2, ..., p_1^L, ..., p_{m_L}^L\}$$
$$Q = \{q_1^1, q_2^1, ..., q_{m_1}^1, q_1^2, ..., q_{m_2}^2, ..., q_1^L, ..., q_{m_L}^L\}$$

where m_i is the number of patches in level i in pyramid representation, L is the total number of levels, and p_j^i is the feature of the jth patch of level i, e.g., a histogram. Then the kernel for bags P and Q is defined as follows:

$$K_{WMI}(P, Q) = \sum_{i=1}^{L} w_i K_{MI}(P^i, Q^i) \tag{1}$$

where $P^i = \{p_1^i, p_2^i, ..., p_{m_i}^i\}$ and $Q^i = \{q_1^i, q_2^i, ..., q_{m_i}^i\}$, namely all the instances in level i, w_i the weight of the ith level instances which is defined according to the size of patches, and K_{MI} is a multi-instance kernel [3]. In this work, we use the following one:

$$K_{MI}(P^i, Q^i) = \frac{1}{(m_i)^2} \sum_{j=1}^{m_i} \sum_{k=1}^{m_i} k^\rho(p_j^i, q_k^i) \tag{2}$$

where ρ is a parameter, $k(.,.)$ is χ^2 kernel defined as:

$$k(p_j^i, q_k^i) = e^{-\frac{1}{A^i} D(p_j^i, q_k^i)} \tag{3}$$

where A^i is a scalar which normalizes the distances. In this paper, we set A^i to the average χ^2 distance between all elements of the ith level of all samples in the training set. $D(p_j^i, q_k^i)$ is the χ^2 distance, and can be defined as:

$$D(p_j^i, q_k^i) = \frac{1}{2} \sum_{t=1}^{d} \frac{(p_{jt}^i - q_{kt}^i)^2}{p_{jt}^i + q_{kt}^i} \tag{4}$$

where d is the dimension of feature vectors.

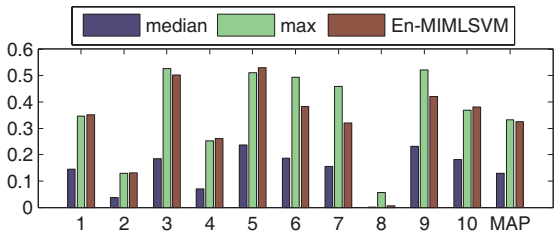

Figure 1: The results of En-MIMLSVM compared with median and max of TRECVID2005

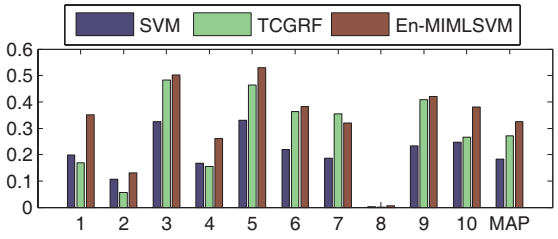

Figure 2: The results of En-MIMLSVM compared with SVM and TCGRF

It has been shown that adjacent video segments will be related to a same semantic concept [17, 19]. This intrinsic property of video data is usually called temporal consistency. To further incorporate the temporal consistency assumption into the kernel design, we propose the Temporally Consistent Weighted Multi-Instance Kernel (TCWMIK) for video annotation task. Here, an instance is represented as $p_j^i = (p_j^{iV}, p_j^{iT})$ in which p_j^{iV} is its visual feature vector and p_j^{iT} is its temporal information. Then the similarity between two instances can be calculated by

$$k_T(p_j^i, q_k^i) = e^{-\frac{\gamma_1}{A^t} D(p_j^{iV}, q_k^{iV}) - \gamma_2 \|p_j^{iT} - q_k^{iT}\|} \tag{5}$$

where the first part of the exponent measures the similarity of visual features between two instances, and $\|p_j^{iT} - q_k^{iT}\|$ calculates the temporal distance between two instances. Parameters γ_1 and γ_2 balance the contribution of these two kinds of features. Note that, given a sample P, all its instances contain the same temporal information. This means that, given two samples P and Q, the second part in Eq. 5 is a constant for any instances between them. Thus, Eq. 5 can be rewritten as:

$$k_T(p_j^i, q_k^i) = e^{-\gamma_2 T_{PQ}} e^{-\frac{\gamma_1}{A^t} D(p_j^{iV}, q_k^{iV})} \tag{6}$$

where $T_{PQ} = \|T_P - T_Q\|$, and T_P is the temporal information of sample P, e.g., the temporal order of the subshot in this paper. Substituting Eq. 6 into Eq. 2, then we get

$$K_{TMI}(P^i, Q^i) = \frac{e^{-\gamma_2 \rho T_{PQ}}}{(m_i)^2} \sum_{j=1}^{m_i} \sum_{k=1}^{m_i} e^{-\frac{\gamma_1 \rho}{A^t} D(p_j^{iV}, q_k^{iV})} \tag{7}$$

Considering the significance of instances of different levels in pyramid representation, we finally get the following kernel:

$$K(P, Q)_{TCWMI} = \sum_{i=1}^{L} w_i K_{TMI}(P^i, Q^i) \tag{8}$$

From the above, we can find that this kernel can utilize both spatial and temporal information contained in samples. Moreover, it is not a bin-by-bin method, and can capture the information between all pairs of patches of the same level in pyramid representation.

3. EXPERIMENTS

To evaluate the proposed method, we conduct experiments on the benchmark data set of TRECVID2005. It consists of 170 hours of TV news videos from 13 different programs in Englich, Arabic and Chinese. After automatic shot boundary detection, the development (DEV) set contains 43970 shots, and the evaluation (EVAL) set contains 45755 shots. Some shots are further segmented into sub-shots. Finally, DEV and EVAL contain 61901 and 64256 sub-shots, respectively. The annotation task of TRECVID 2005 is to detect the presence or absence of 10 predetermined benchmark concepts in each shot of the EVAL set. These concepts are: *walking_running*, *explosion_fire*, *maps*, *flag-US*, *building*, *waterscape_waterfront*, *mountain*, *prisoner*, *sports* and *car* which are numbered from 1 to 10 in this paper. For each concept, systems are required to return a ranked-list of up to 2000 shots. The performance is evaluated by *Average Precision*(AP) and *Mean Average Precision*(MAP), which are the official performance metric in TRECVID evaluations. Each frame is represented by pyramid scheme: $1\times1, 2\times2, 1\times3$ and 3×1. From each patch, SIFT, OpponentSIFT and *rg*SIFT descriptors are extracted by Harris-Laplace and dense sampling methods. Then, codebook is generated from random selected descriptors. Finally, each patch is represented as a histogram.

A series of experiments are conducted to compare En-MIMLSVM with several state-of-the-art approaches including SVM, TCGRF [17] , MIMLSVM [26]. In the experiments, for En-MIMLSVM, we set the number of subsets $T = 10$, and the rounds of AdaBoost $s_i = 10$. During training, if the number of samples in minority class for one concept is more than 2000, 2000 samples are selected to represent the minority class. Other parameters, as well as parameters of the comparison algorithms are all selected by 5-fold cross-validations on training set.

Figure 1 shows the results of En-MIMLSVM compared with the *median* (the average results of all the participants in TRECVID05) and *max* (The best results for each concept of all the participants in TRECVID2005). From this figure, we can find that the performance of En-MIMLSVM always outperforms the *median*. It outperforms the *max* on 5 concepts.

Figure 2 compares En-MIMLSVM with TCGRF and SVM. Note that, for SVM, the features of all patches of a sample are firstly averaged so that each sample is represented as a vector; then, an SVM with RBF kernel is constructed. We can find that En-MIMLSVM beats SVM on all concepts, and beats TCGRF on 9 concepts.

Figure 3 compares the performance of En-MIMLSVM and MIMLSVM [26]. We can find that En-MIMLSVM performs much better than MIMLSVM on all concepts, as we have expected since En-MIML-SVM is an ensemble method using an improved version of MIMLSVM as base learner. It is important to note that, En-MIMLSVM is about 5~8 times faster than MIMLSVM because each classifier in En-MIMLSVM is built on a small data set.

For a further study, we also implement a variant, En-MIMSVM-v, in which only the visual features are used. In other words, Eq. 3 is used for training and testing. The results are shown in Figure 4. From this figure, we can observe that En-MIMLSVM outperforms En-MIMLSVM-v on all concepts. In particular, En-MIMLSVM performs much better on *walking_running*, *flag-US*, *building*, and *sports* than En-MIMLSVM-v due to the fact that the temporal consistency in these concepts is stronger than others. This confirms the effectiveness of incorporating temporal consistency.

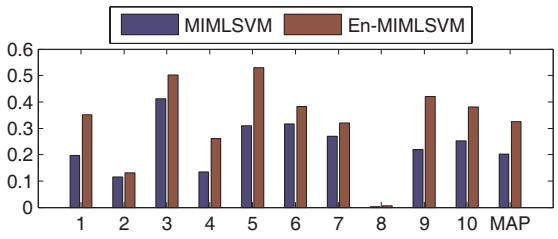

Figure 3: The results of En-MIMLSVM compared with MIMLSVM

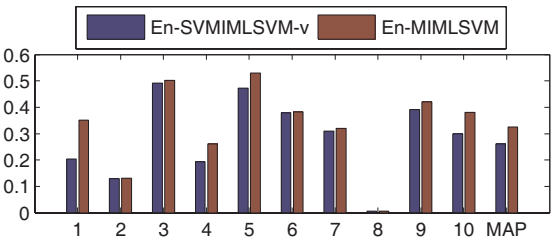

Figure 4: The results of En-MIMLSVM compared with En-MIMLSVM-v

4. CONCLUSION

In this paper, we propose the En-MIMLSVM approach for video annotation. This is an ensemble method developed in the MIML framework, and it is able to capture both the structure information and temporal information in the metadata. En-MIMLSVM is also endowed with the ability of dealing with class imbalance and improvement of learning efficiency. Experiments show that En-MIMLSVM outperforms a number of state-of-the-art methods. An interesting future work is to evaluate the proposed method on more concepts, such as the 39/374 LSCOM-lite concepts in TRECVID data.

5. REFERENCES

[1] A. Amir, M. Berg, and S.-F. Chang. IBM research TRECVID-2003 video retrieval system. In *TREVID*, 2003.

[2] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple-instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1–2):31–71, 1997.

[3] T. Gartner, P. A. Flach, A. Kowalczyk, and A. J. Smola. Multi-instance kernels. In *ICML*, 179–186, 2002.

[4] A. Ghoshal, P. Ircing, and S. Khudanpur. Hidden markov models for automatic annotation and content-based retrieval of images and video. In *SIGIR*, 544–551, 2005.

[5] K. Grauman and T. Darrell. The pyramid match kernel: Efficient learning with sets of features. *Journal of Machine Learning Research*, 8:725–760, 2007.

[6] Z. Gu, T. Mei, X.-S. Hua, J. Tang, and X. Wu. Multi-layer multi-instance learning for video concept detection. *IEEE Transactions on Multimedia*, 10(8):1605–1616, 2008.

[7] R. Jin, S. Wang, and Z.-H. Zhou. Learning a distance metric from multi-instance multi-label data. In *CVPR*, 896–902, 2009.

[8] Y.-X. Li, S. Ji, J. Ye, and Z.-H. Zhou. Drosophila gene expression pattern annotation through multi-instance multi-label learning. In *IJCAI*, 1445–1450, 2009.

[9] X.-Y. Liu, J. Wu, and Z.-H. Zhou. Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 39(2):539–550, 2009.

[10] M. Naphade, L. Kennedy, J. R. Kender, S. F. Chang, P. Over, and A. Hauptmann. A light scale concept ontology for multimedia understanding for TRECVID 2005. In IBM Research Technical Report, 2005.

[11] M. Naphade and J. Smith. A generalized multiple instance learning algorithm for large scale modeling of multimedia semantics. In *ICASSP*, 341–344, 2005.

[12] M. R. Naphade, I. Kozintsev, and T. Huang. Factor graph framework for semantic video indexing. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(1):40–52, 2002.

[13] N. Nguyen. A new SVM approach to multi-instance multi-label learning. In *ICDM*, 384–392, 2010.

[14] G.-J. Qi, X.-S. Hua, Y. Rui, J. Tang, T. Mei, M. Wang, and H.-J. Zhang. Correlative multilabel video annotation with temporal kernels. *ACM Transactions on Multimedia Computing, Communications and Applications*, 5(1):1–27, 2008.

[15] J. R. Smith, M. Naphade, and A. Natsev. Multimedia semantic indexing using model vectors. In *ICME*, 445–448, 2003.

[16] C. G. M. Snoek, M. Worring, J. C. van Gemert, J.-M. Geusebroek, and A. W. M. Smeulders. The challenge problem for automated detection of 101 semantic concepts in multimedia. In *MM*, 421–430, 2006.

[17] J. Tang, X.-S. Hua, T. Mei, G.-J. Qi, S. Li, and X. Wu. Temporally consistent gaussian random field for video semantic analysis. In *ICIP*, 525–528, 2007.

[18] K. E. van de Sande, T. Gevers, and C. G. Snoek. Evaluating color descriptors for object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1582–1596, 2010.

[19] J. Yang and A. G. Hauptmann. Exploring temporal consistency for video analysis and retrieval. In *MIR*, 33–42, 2006.

[20] S.-H. Yang, H. Zha, and B.-G. Hu. Dirichlet-bernoulli alignment: A generative model for multi-class multi-label multi-instance corpora. In *NIPS 23*, 2143–2150, 2009.

[21] J. Yuan, Z. Guo, and L. Lv et al. THU and ICRC at TRECVID. In *TRECVID*, 2007.

[22] Z.-J. Zha, X.-S. Hua, T. Mei, J. Wang, G.-J. Qi, and Z. Wang. Joint multi-label multi-instance learning for image classification. In *CVPR*, 1–8, 2008.

[23] M.-L. Zhang and Z.-H. Zhou. M3MIML: A maximum margin method for multi-instance multi-label learning. In *ICDM*, 688–697, 2008.

[24] Z.-H. Zhou. Ensemble learning. In S. Z. Li, editor, *Encyclopedia of Biometrics*, pages 270–273. Springer, 2009.

[25] Z.-H. Zhou, Y.-Y. Sun, and Y.-F. Li. Multi-instance learning by treating instances as non-i.i.d. samples. In *ICML*, 1249–1256, 2009.

[26] Z.-H. Zhou and M.-L. Zhang. Multi-instance multi-label learning with application to scene classification. In *NIPS 19*, 1609–1616, 2007.

[27] Z.-H. Zhou, M.-L. Zhang, S.-J. Huang, and Y.-F. Li. MIML: A framework for learning with ambiguous objects. *CoRR abs/0808.3231*, 2008.