MIL at ImageCLEF 2013: Scalable System for Image Annotation

Masatoshi Hidaka, Naoyuki Gunji, and Tatsuya Harada

Machine Intelligence Lab., The University of Tokyo
{hidaka,gunji,harada}@mi.t.u-tokyo.ac.jp
http://www.mi.t.u-tokyo.ac.jp

Abstract. We give details of our methods in the ImageCLEF 2013 Scalable Concept Image Annotation task. For the textual feature, we propose a method for selecting text closely related to an image from its webpage. In addition, to consider the meaning of the concept, we propose to use WordNet for getting words related to the concept. For visual features, we use Fisher Vector (FV), which is regarded as an extension of the Bag-of-Visual-Words representation. We trained linear classifiers by Passive–Aggressive with Averaged Pairwise Loss (PAAPL), an online multilabel learning method based on Passive–Aggressive. Since PAAPL is computationally efficient and able to cope with multilabel data appropriately, it is suitable for this task. Results show that our annotation pipeline is simple but works well in this task.

Keywords: ImageCLEF, Textual Feature, WordNet, Annotation

1 Introduction

In ImageCLEF 2013 Scalable Concept Image Annotation, our task is multi-label image annotation [1]. The dataset is extracted from general webpages, so that the costs in collecting data are low [2]. However, collected images have no explicit labels. Therefore, we need to extract correct labels of corresponding images from webpages. As for the extraction of labels from websites, the simplest solution is that concept labels which exist in webpages are assigned to the images. However, this method often fails to get correct labels because it does not consider meanings of concepts. Furthermore texts of webpages are not necessarily related to the images. Therefore, we try some methods to get more accurate labels. To achieve it, we use information from WordNet [3] to get words related to the concepts. In addition, limitation of text extraction range is adopted to omit text not related to the images. For visual features, we adopt Fisher Vector (FV)[4], which is an improved method of Bag-of-Visual-Words (BoVW). We use linear classifiers for each concept because they are computationally efficient and suitable for large-scale data. When training classifiers, because labels assigned to the images are not ground-truth labels, they must be regarded as noisy. Therefore, we devote attention to robustness for noise in the training data. In order to train linear classifiers, we use PAAPL [5], an online multilabel learning method based on

Passive–Aggressive[6]. PAAPL shows faster convergence than PA and has the same feature of robustness to the noise as PA.

2 Feature Extraction

2.1 Visual Feature

As a visual feature, we use the Fisher Vector (FV). Because it can achieve a good classification performance with a linear classifier, it is often used for large-scale visual categorization. Indeed, in the ImageNet Large-Scale Visual Recognition Challenge 2012 (ILSVRC2012), four out of seven teams used FVs to represent images. We use four local descriptors: SIFT, C-SIFT, LBP, and GIST. Actually, GIST is usually used to describe a whole image, but we use it as a local descriptor. All local descriptors are reduced to 64 dimensions using Principal Component Analysis (PCA). Local descriptors are densely extracted from five scales of patches on a regular grid every six pixels and learn a Gaussian Mixture Model (GMM) with 256 components, which have a diagonal matrix as its covariance matrix. To use spatial information, we divide images into 1×1, 2×2, and 3×1 cells. Then FVs are calculated over each region as follows.

Let $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_N\}$ be a set of N local descriptors extracted from an image, and w_i, $\boldsymbol{\mu}_i$, $\boldsymbol{\Sigma}_i$ be the mixture weight, mean vector, covariance matrix of the i-th Gaussian, respectively. Then we difine,

$$\boldsymbol{u}_i = \frac{1}{N\sqrt{w_i}} \sum_{n=1}^{N} \gamma_n(i) \boldsymbol{\Sigma}_i^{-\frac{1}{2}} (\boldsymbol{x}_n - \boldsymbol{\mu}_i),$$

$$\boldsymbol{v}_i = \frac{1}{N\sqrt{2w_i}} \sum_{n=1}^{N} \gamma_n(i) \left[\boldsymbol{\Sigma}_i^{-1} \mathrm{diag}((\boldsymbol{x}_n - \boldsymbol{\mu}_i)(\boldsymbol{x}_n - \boldsymbol{\mu}_i)^T) - \mathbf{1} \right],$$

where $\mathbf{1}$ is a column vector whose components are all 1 and $\mathrm{diag}(X)$ for matrix X is a column vector which is composed of diagonal components of X. $\gamma_n(i)$ is the soft assignment of $\boldsymbol{x}_n$ to i-th Gaussian as

$$\gamma_n(i) = \frac{w_i u_i(x_n)}{\Sigma_{j=1}^{K} w_j u_j(x_n)},$$

where u_i is the i-th Gaussian, and it is also known as the posterior probability. The FV representation is therefore given as

$$\mathcal{G} = \left[\mathbf{u}_1^T \ \mathbf{v}_1^T \ldots \mathbf{u}_K^T \ \mathbf{v}_K^T \right]^T,$$

where K is the number of GMM components.

Following [4], we apply power normalization and L2 normalization to each of the extracted FVs. Power normalization is done by applying the function,

$$g(z) = sign(z)|z|^a,$$

to each component of FVs, where a is a parameter and is set to 1/2 in this work. After normalization, we concatenate them into a single vector. The dimension of our FVs is 262144.

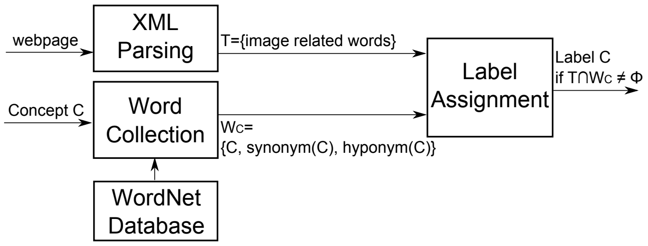

Fig. 1. Pipeline of label assignment.

2.2 Textual Feature

To assign correct labels to images, we take two steps. First we extract text closely related to an image from its webpage. Then if a concept word exists in the extracted text, the concept label is assigned to the image. The pipeline is presented in Fig. 1.

Text Extraction. To extract text closely related to an image, we consider three types of texts in the webpage: text around image, img tag attributes (src, alt, title), page title. First, we parse the xml file of the webpage and extract page title, text, img tag. Then we select some of them and split them into a set of single words T. For the text around the image, we consider the distance from the image (img tag position) because the entire webpage does not necessarily focuses on one image. Then we use max distance from an image as a parameter. We use words which are within the max distance. To normalize words, we singularize nouns.

Label Assignment. To assign labels to the image, first we collect words related to the each concept C given in the task. We denote a set of collected words by W_C. For W_C, collecting synonyms and hyponyms of C is considered.

$$W_C = \{C, \text{synonym}(C), \text{hyponym}(C)\}$$

For example, given a target concept "bird", we get

$$W_{bird} = \{\text{bird}, \text{parrot}, \text{pigeon}, ...\}.$$

For collecting synonyms and hyponyms, we use WordNet. To make implementation simpler, we use no compound words. Hyponyms are hierarchized. Therefore, we collect words of all depths recursively. Words which have multiple meanings are omitted. Determination is done by checking whether the word appears in multiple entries in WordNet.

Then if the extracted text T contains any of the concept-related words W_C (concept word, synonyms and hyponyms), we assign those concept labels C to the image. Consequently, we obtain a training dataset in which some images have multiple labels, and some images have no label.

3 Multilabel Annotation

In this section, we describe the method of training of the classifiers and annotating of the test images. We use linear classifier for each concept label considering the scalability. With linear classifier, the annotation for test images is performed by computing score of labels as product of the visual feature and the weight vector of labels, and assigning the top 5 scored labels.

To learn the models for each concept label from various images, requirements are not only compatibility of scalability for the data amount and accuracy for label estimation, but also noise tolerability.

For that reason, we use Passive–Aggressive with Averaged Pairwise Loss (PAAPL). PAAPL is based on Passive–Aggressive (PA) method, which is known to be tolerant to the noise in training samples.

First, we describe the model update rule of PA.

Given the t-th training sample, we denote the visual feature by $\boldsymbol{f}_t$, the set of concept labels assigned to the sample by Y_t, the set of concept labels not assigned to the sample by $\bar{Y}_t$, the current model (weight vector) corresponding to concept label C by $\boldsymbol{\mu}_t^C$. In our setting, the dimension of $\boldsymbol{f}_t$ is 262144 (Fisher Vector) + 1 (bias).

1. Fetch t-th training sample, compute scores for each label using current models.
2. Find a label $r_t \in Y_t$ associated with the sample and a label $s_t \in \bar{Y}_t$ not associated with the sample as follows.

$$r_t = \arg\min_{r \in Y_t} \boldsymbol{\mu}_t^r \cdot \boldsymbol{f}_t$$
$$s_t = \arg\max_{s \in \bar{Y}_t} \boldsymbol{\mu}_t^s \cdot \boldsymbol{f}_t$$

   Given these labels, compute the hinge-loss l from the current model. The hinge-loss l is given as

$$l(\boldsymbol{\mu}_t^{r_t}, \boldsymbol{\mu}_t^{s_t}; (\boldsymbol{f}_t, Y_t)) = \begin{cases} 0 & \boldsymbol{\mu}_t^{r_t} \cdot \boldsymbol{f}_t - \boldsymbol{\mu}_t^{s_t} \cdot \boldsymbol{f}_t > 1 \\ 1 - (\boldsymbol{\mu}_t^{r_t} \cdot \boldsymbol{f}_t - \boldsymbol{\mu}_t^{s_t} \cdot \boldsymbol{f}_t) & \text{otherwise} \end{cases}$$

3. Update models with the update rule below.

$$\boldsymbol{\mu}_{t+1}^{r_t} = \boldsymbol{\mu}_t^{r_t} + \frac{l}{2|\boldsymbol{f}_t|^2 + \frac{1}{D}} \boldsymbol{f}_t$$
$$\boldsymbol{\mu}_{t+1}^{s_t} = \boldsymbol{\mu}_t^{s_t} - \frac{l}{2|\boldsymbol{f}_t|^2 + \frac{1}{D}} \boldsymbol{f}_t$$

   D is a parameter which controls the sensitivity to label prediciton mistakes.

Then we describe the PAAPL method.

1. Fetch t-th training sample, compute scores for each label using current models.

2. For all combinations of label $r_t \in Y_t$ associated with the sample and label $s_t \in \bar{Y}_t$ not associated with the sample, compute the hinge-loss as PA.
3. For all combinations for which the hinge-loss is not 0, update the model corresponding to the update rule of PA.

In PA, only a pair of models is updated for one sample. In PAAPL, on the other hand, all pairs of models are updated for one sample, which reduces the number of training iterations and score computation process, which is time-consuming. Therefore, the models converge faster.

4 Results

Using the visual feature and the textual feature stated in the previous section, the image classifier was trained by PAAPL. The number of training iterations was 5.

First, we determined whether we should use synonyms and hyponyms of concept for assigning labels to the image. For extracting text from a webpage, we used 10 words of text around the image, img tag attributes and page title. The visual feature is provided BoVW representations of C-SIFT.As a result, we chose to use synonyms and hyponyms. The result is presented in Table 1.

Table 1. Comparison of the use of synonyms and hyponyms.

Synonym	Hyponym	MF-samples
-	-	0.234
✓	-	0.232
-	✓	0.261
✓	✓	**0.266**

Second, we conducted a grid search for the text extraction conditions on the length of words around the image should be considered, the necessity of using the img tag attributes and the page title. The visual feature is the same as in first step. Results show that using only img tag attributes was the best. The text far from the image decreased label assignment accuracy notably. The result is shown in Table 2. The number of images which have at least one label and the average number of labels assigned to one image was also shown in the result. Because of the property of PAAPL, only images which have at least one label are used for training. It is worth noting that in the best condition, the number of images used and the average number of labels are both lowest.

After this optimization, we tried a previous evaluation (of whether we should use synonyms and hyponyms) again, but the result was the same.

Finally, using the condition of the textual feature extraction stated above, we trained the weight vectors corresponding to each visual feature (Fisher Vector). It took 2 hr to learn for each visual feature. The final score of each test

Table 2. Comparison of the text extraction range.

Text around					
image	Img tag	Page			Average
(max distance)	attributes	title	MF-samples	Number of images	number of labels
10	-	-	0.254	113802	0.7
100	-	-	0.231	183545	2.5
1000	-	-	0.202	192210	5.2
-	✓	-	**0.276**	80009	0.4
10	✓	-	0.266	129050	0.8
100	✓	-	0.238	185471	2.5
1000	✓	-	0.213	193170	5.3
-	-	✓	0.246	92254	0.5
10	-	✓	0.255	134318	0.9
100	-	✓	0.229	185471	2.5
1000	-	✓	0.205	193497	5.3
-	✓	✓	0.260	111247	0.6
10	✓	✓	0.261	140448	0.9
100	✓	✓	0.230	186394	2.6
1000	✓	✓	0.207	193971	5.3

image is calculated by summing the scores of all the classifiers (C-SIFT+FV, GIST+FV, LBP+FV, and SIFT+FV). Final results are presented in Table 3. The evaluation with provided C-SIFT + Bag-of-Visual-Words is also shown in the table. Fisher vector exhibited much higher performance than Bag-of-Visual-Words. We performed learning and annotation for the test set with the top 5 ranked combinations.

According to the results presented from the task organizers, we have achieved the second score among all teams with our best run.

5 Conclusions

In this working note, our methods to annotate images in ImageCLEF 2013 Scalable Concept Image Annotation task are described, with particular emphasis on extracting labels for images from websites. Results show that, using concepts' synonyms and hyponyms from WordNet was useful and limiting text range of website was also shown to be important. For visual features, we applied Fisher Vector, a state-of-the-art coding method. Four local descriptors for FV were tried. The combination of C-SIFT, GIST and SIFT showed superior performance. Our textual and visual features are simple but we can achieve a good performance.

References

1. M. Villegas, R. Paredes, and B. Thomee. Overview of the ImageCLEF 2013 Scalable Concept Image Annotation Subtask. *CLEF 2013 working notes*, 2013.

Table 3. Results of score combinations.

C-SIFT	GIST	LBP	SIFT	MF-samples
✓	-	-	-	0.312
-	✓	-	-	0.324
-	-	✓	-	0.279
-	-	-	✓	0.311
✓	✓	-	-	0.338
✓	-	✓	-	0.321
✓	-	-	✓	0.336
-	✓	✓	-	0.331
-	✓	-	✓	0.340
-	-	✓	✓	0.317
✓	✓	✓	-	0.342
✓	✓	-	✓	**0.346**
✓	-	✓	✓	0.332
-	✓	✓	✓	0.339
✓	✓	✓	✓	0.343
Provided C-SIFT + BoVW				0.276

2. M. Villegas and R. Paredes. Image–Text Dataset Generation for Image Annotation and Retrieval. *CERI*, 2012.
3. C. Fellbaum. *WordNet: An Electronic Lexical Database.* MIT Press, 1998.
4. F. Perronnin, J. Sanchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. *European Conference on Computer Vision*, 2010.
5. Y. Ushiku, T. Harada, and Y. Kuniyoshi. Efficient image annotation for automatic sentence generation. *International Conference on Multimedia*, 2012.
6. K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer. Online Passive–Aggressive Algorithms. *The Journal of Machine Learning Research*, Vol. 7, pp. 551–585, 2006.