Automatic Image Annotation via Incorporating Naive Bayes with Particle Swarm Optimization

Mohamed Sami and Aboul Ella Hassanien
Faculty of Computers
and Information - Cairo University
Scientific Research Group
in Egypt (SRGE)
Cairo - Egypt
Email: mohamed.sami@egyptscience.net,
abo@egyptscience.net

Nashwa El-Bendary
Arab Academy for Science,
Technology, and Maritime Transport
Scientific Research Group
in Egypt (SRGE)
Cairo - Egypt
Email: nashwa.elbendary@ieee.org

Robert C. Berwick
Massachusetts Institute of
Technology (MIT)
USA
Email: berwick@csail.mit.edu

***Abstract*—This paper presents an automatic image annotation approach that integrates the Naive Bayes classifier with particle swarm optimization algorithm for classes probabilities weighting. The proposed hybrid approach refines the output of multi-class classification that is based on the usage of Naive Bayes classifier for automatically labeling images with a number of words. Each input image is segmented using the normalized cuts segmentation algorithm in order to create a descriptor for each segment. One Naive Bayes classifier is trained for all the classes. Particle swarm optimization algorithm is employed as a search strategy in order to identify an optimal weighting for classes probabilities from Naive Bayes classifier. The proposed approach has been applied on Corel5K benchmark dataset. Experimental results and comparative performance evaluation, for results obtained from the proposed approach and other related researches, demonstrate that the proposed approach outperforms the performance of other approaches, considering annotation accuracy, for the experimented dataset.**

I. Introduction

Recently, the rapid development of technology for camera devices, storage, transmission and the vast growth of internet users over the world have led to a huge increase of digital image libraries, especially on the web. Content based image retrieval (CBIR) [1] is a known technology to analyze and control these image resources. In CBIR, the user has to enter low level visual features, such as color, shape, or texture, leaving a semantic gap in the user query results. In addition to that, keyword search based is more user friendly than the visual features. So, the importance of automatic image annotation (AIA) techniques, which are technologies for assigning keywords in order to describe images context. AIA builds a bridge between the high level semantic and low level features, which is a considered as an approach to solve the semantic gap problem. Many contributions have been done in the AIA field. Mori et al [2] developed a co-occurrence model to establish the association between words and images. This model is for region labeling and it has involved four main steps. The first step is grid segmentation for the images, where each image was divided into equal rectangles.This type of segmentation has been chosen because it is fast and simple. The second step is feature extraction for region, the third step is clustering the features vectors using vector quantization, and the last step is creating a probability model that links each word to a given the cluster. On the other hand, Duygulu et al. [3] proposed a model based on machine translation. They treated AIA problem as learning lexicon. Moreover, Barnard and Forsyth [4] proposed a hierarchical model based on statistical clustering. They represented the words and the blobs as distribution over the hierarchy nodes. That model is a hierarchical combination of the asymmetric and symmetric clustering models. In their experiments, they clustered about 3000 Corel images into 64 clusters. Also, Jeon et al. [5] proposed AIA model for annotating and retrieving images. In that model, images are segmented and features are extracted from each region. They used the same segmentation algorithm and the same 33 features as in Duygulu et al. [3] and Lavrenko et al. [6]. However, for Jeon et al.'s work [5], they proposed a continuous relevance model (CRM) instead of the discrete one discussed lately. They assumed that each image region is represented by continuous valued feature vector. Furthermore, in [9], multiple-bernoulli relevance model was proposed to improve the previously proposed CRM and cross-media relevance (CMRM) models. Moreover, Shunle and Xiaoqiu [10] have proposed a AIA model based on multi instance learning. The proposed approach in this paper is based on the Naive Bayes classifier with particle swarm optimization algorithm for classes probabilites weighting. Many experiments have been done to benchmark Corel5k dataset and the results have been compared to previous related works. The rest of this paper is organized as follows. Section II introduces a brief description of Naive Bayes classifier and particle swarm optimization algorithm used in the proposed approach. Section III describes in details the proposed image annotation approach. Section IV presents experimental results. Finally, section V addresses conclusions and discusses future work.

II. Particle Swarm Optimization (PSO) and Naive Bayes Classifier : Preliminaries

Due to space limitations we provide only a brief explanation of the basic framework of particle swarm optimization algorithm and Naive Bayes classifier, along with some of the key definitions. A more comprehensive review can be found in sources such as [11]–[16].

A. Particle swarm optimization

The concept of particle swarms, although initially introduced for simulating human social behaviors, has become very popular these days as an efficient search and optimization technique. Particle swarm optimization (PSO) [11]–[13], does not require any gradient information of the function to be optimized. It uses only primitive mathematical operators and is conceptually very simple. PSO has attracted the attention of a lot of researchers resulting into a large number of variants of the basic algorithm as well as many parameter automation strategies. The canonical PSO model consists of a swarm of particles, which is initialized with a population of random candidate solutions. They move iteratively through the d-dimension problem space to search the new solutions, where the fitness f can be calculated as the certain qualities measure. Each particle has a position represented by a position-vector $\vec{x}_i$ (i is the index of the particle) and a velocity represented by a velocity-vector $\vec{v}_i$. Each particle remembers its own best position so far in a vector $\vec{x}_i^{\#}$ and its j-th dimensional value is $x_{ij}^{\#}$. The best position-vector among the swarm so far is then stored in a vector $\vec{x}^*$ and its j-th dimensional value is x_j^*. During the iteration time t, the update of the velocity from the previous velocity to the new velocity is determined, and the new position is then determined by the sum of the previous position and the new velocity, for more details refer to our published work in [7], [8].

B. Naive Bayes classifier

The naive Bayes classifier [18]–[20] is a simple probabilistic classifier based on the well known Bayes theorem with strong assumptions. This classifier is used in image level AIA first approach. This classifier is based on building a feature independent probability model. The naive Bayes classifier assumes that the presence (or absence) of a particular feature of a class is unrelated to the presence (or absence) of any other feature, given the class variable. One of the assumption that the features are independent for each class, for example a watermelon may be considered a watermelon if it is green, diameter larger than 15 cm and rounded. The Bayes classifier assumes that these features contribute for watermelon class independently even if it depends on each other. Parameter estimation for the naive Bayes classifier is done using maximum likelihood algorithm [17] where Bayes classifier is trained using supervised learning. One of the advantages of this classifier is that it doesn't need a large size of samples for good training. As this classifier is assuming the features are conditionally independent, its conditional model can be represented abstractly by :

$$P(Y|X_1, ...X_n) \tag{1}$$

where Y is the class labels and X_1 through X_n are feature variables. We can reformulate this model using Bayes' theorem which is :

$$Posterior = \frac{prior * likelihood}{evidence} \tag{2}$$

and mathematically it is :

$$P(Y|X_1, ...X_n) = \frac{P(Y) * P(X_1, ...X_n|Y)}{P(X_1, ...X_n)} \tag{3}$$

For the classifier parameters estimation you have to assume the distribution of features.

III. The proposed AIA approach

In this approach we assumed the features are continuous values and distributed according to Gaussian distribution per each class. The mean and the variances are estimated for each class using expectation maximization algorithm.

All the regions feature vectors belongs to images that are labeled with specific word(class) are collected as contributing for this word(class), After that a one bayes classifier is trained to classify between all the N words(classes). The problem appears here clearly which is there will be images repeated on more than one words(classes), and thus increase the noise in the data specially for the major words that are repeated frequently for example "sky". Figure 1 illustrates the naive bayes approach model .

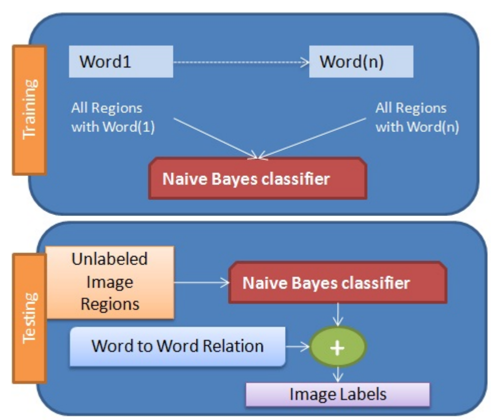


Fig. 1. bayes classifier model

In this approach the word to word relation is taken into consideration, for example the word "fish" most probably to appears with word "water" and the word "sun" most probably appear with word "sky". The word to word relation is established by counting the frequent words that appears with each word in the training dataset using the algorithm 1 [8]. Table I shows parameters used for algorithm 1.

The following is the testing algorithm 2 used to assign labels for unlabeled images. At this point we assume that naive Bayes

TABLE I
PARAMETERS FOR ALGORITHM 1

Parameter	Description
NumClasses	Total number of classes in the training dataset
NumImages	Total number of images in the training dataset
WordImages	Labels number per each image

Algorithm 1 Words correlations simple algorithm

for y=1 to NumClasses **do**
  for i=1 to NumClasses **do**
    for x=1 to NumImages **do**
      if i==y **then**
        $continue$
      end if
      Compute: $count$ = FindImages(WordImages(x)==i)
      Compute: $Correlations(y, i)$ = $count$
    end for
  end for
end for

classifier is already trained in addition to the correlation array. Table II shows parameters used for algorithm 2.

Algorithm 2 Annotation using bayes classifier algorithm

for x=1 to T **do**
  Compute: $RegionVector = GetRegionsForImage(x)$
  for y=1 to N **do**
    Compute: $ProbArray = BaysClassifier.classify(RegionVector(y))$
    Compute: $AllProbArray = AllProbArray * ProbArray$
  end for
  $ClassIndex$ =FindMax($AllProbArray$)
  $CorrletionLabels$=$WordCorrelations(ClassIndex)$
  $NewLabels$=$ClassIndex$+$CorrletionLabels$
end for

The $RegionVector$ is the features vector for the current image region, the $ProbArray$ variable includes the posterior probabilities for each class per image region, the $AllProbArray$ variable includes cumulative multiplication for each class using all posterior probabilities outputs per one image, $AllProbArray$ variable is sorted to find the class with the highest probability, this class and its most correlated ones will be the labels of non annotated image. In order to increase the accuracy of the AIA Bayes based model, we have applied the PSO algorithm to the Bayes classification part which tested on Corel5K dataset. In this case we didn't use PSO algorithm for features weighting or selection, but we used it for classes output probabilities weighting and the reason behind not using it as feature weighting is the high the computation cost needed for feature weighing on such big dataset, as in the feature selection based optimization we have to train the classifier using the weighted feature vector in the training phase and also use the generated weights in the testing phase and for weighting the test feature vectors to be classified by trained classifier, all this is very computational expensive specially in the cases with large datasets like Corel5k. For probabilities weighting we updated the Bayes AIA model algorithm and the following algorithm3 shows only the updated part.

TABLE II
PARAMETERS FOR ALGORITHM 2

Parameter	Description
T	Total number of test images
N	Total number of regions per image
W	Total number of test data
ProbArray	Array of classes probabilities per class
AllProbArray	Array of classes of conditional probabilities per class
CorrletionLabels	The word to word repeats
NewLabels	The output labels for unknown image

Algorithm 3 Fitness Bayes-PSO algorithm

for y=1 to N **do**
  Compute: $ProbArray = BaysClassifier.classify(RegionVector(y))$
  Compute: $AllProbArray = AllProbArray * ProbArray * ClassWeight$
end for

The algorithm section above is updated in the testing algorithm of the Bayes AIA model. The updated algorithm is used as fitness function for the PSO algorithm. The $ClassWeight$ variable is an array contains all the weights for N classes. The $ClassWeight$ is the optimized particles (weights) for the PSO algorithm, where the fitness value here is the average precision only.

IV. EXPERIMENTAL RESULTS

In our experiments, we used Corel5k [21] dataset. The Corel dataset consists of 5000 images from 50 Corel Stock Photo CDs. Each cd includes 100 images on the same topic and each image is also associated with 1-5 keywords. This dataset is divided into 4500 images for training and 500 images for testing. In the training dataset there are 371 words. We consider each word as a class, as previously explained in section III. Each Image is segmented using normalized cuts segmentation algorithm, then the region with size larger than a certain threshold is selected. Each image has a number of regions between 5 to 10. There are 42379 regions for all the training dataset. For each region, a 33 features vector is extracted and the regions are clustered into 500 clusters. These features include segment size, location, convexity, first moment, region color, and region average orientation energy. The dimension of each feature vector is 36. The size of testing data is 500 images and includes only 263 words.

In order to measure our experiments, we used the same measures applied in previous works on Corel5k benchmark dataset

TABLE III
THE ACCURACY RESULTS FOR OPTIMIZED BAYES AIA MODEL

Models	Average Precision	Average Recall	NumWords
NB	0.0951	0.1160	85
NB+PSO	0.1503	0.1474	97

[8]. These measures are well known in the field of automatic image annotation. the first measure is the precision, which is referred as the ratio of the counter of correct annotation in relation to all the times of annotation. The second measure is the recall, which is referred as the ratio of the times of correct annotation in relation to all the positive annotated samples. Equations (4) and (5) show the calculations of precision and recall measures, respectively.

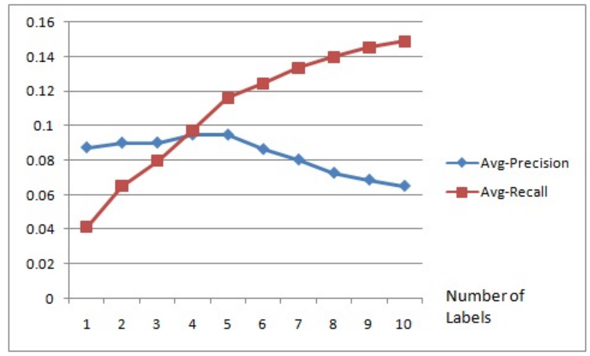
$$Precision = \frac{B}{A} \tag{4}$$

$$Recall = \frac{B}{C} \tag{5}$$

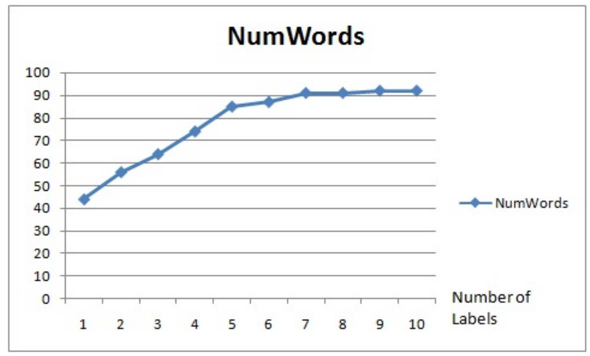
Where A is the number of images annotated by some keyword, B is the number of images annotated correctly, and C is the number of images annotated by some keyword in the whole dataset. Another measure is $NumWords$, which is statistics of the number of correctly annotated keywords that are used to correctly annotate at least one image. This statistical measure reflects the coverage of keywords in the different proposed methods. Figure 2 shows results for Bayes AIA model when annotation each image by number of output labels ranges from 1-10 words. There are two curves one represents the average precision and the other represents the average recall. The accuracy is when the number of labels equal 5 and in this case the average precision is equal to 0.095 and the average recall is equal to 0.116. We can notice that the change the in average precision values is not large compared with the results depicted in figure 2 (a) and (b). Table III presents the accuracy results for Optimized Bayes AIA model.

In this part we show the experiments results for apply the particle swarm optimization technique on the Naive Bayes AIA based model. For the Bayes based model we set the range of search in the particle swarm optimization technique to be from $0 - 1000$ and thus to increase the possibility to show the minor classes that have a small training data and most of times takes low probabilities values . For PSO configuration, we have used number of iterations equals to 400, number of particles equals to 200 and velocity step equals to 2.

Table III shows how the accuracy is improved after using PSO algorithm along with Bayes model. Table IV compare the Naive Bayes based approach proposed in this paper and previous traditional annotation models such as COM [2], TM [3], CMRM [5], CRM [6], MBRM [9], and MIL [10]. The proposed model is marked as $NB + PSO$ that stands for naive Bayes with Particle swarm optimization algorithm. The proposed model doesn't achieving the best accuracy but the improvements in the overall accuracy after using PSO



(a) Avg-Precision, Avg-Recall curve



(b) NumWords results curve

Fig. 2. Avg-Precision, Avg-Recall and NumWords results for Bayes based model

TABLE IV
THE PERFORMANCES OF VARIOUS ANNOTATION MODELS ON COREL5K VS RFC

Model	Average Precision	Average Recall	NumWords
COM	0.03	0.02	19
TM	0.06	0.04	49
CMRM	0.10	0.09	66
CRM	0.16	0.19	107
MBRM	0.24	0.25	122
MIL	0.20	0.22	124
NB+PSO	0.1503	0.1474	97

algorithm can to better results when using stronger classifiers.

We noticed that the high frequent classes in the training data like ”sky-class(3) or water-class(5)” will take small weighting values, and the minor classes will take large values, But this expectations is not obvious in the numbers above we can't find a specific pattern in it, the reason for that is the major classes in the training data is also a major classes in the testing data rather than the affect of the word correlations. Applying optimization algorithm on image level AIA models may take more than six month on a small scale hardware capabilities like the one used in out researches.

V. Conclusions and Future Work

In this paper, an automatic image annotation approach, based on Naive Bayes classifier and particle swarm optimization algorithm, has been proposed and tested. The proposed approach shows that applying PSO algorithm with Naive Bayes increased the average precision from 0.0951 to 0.1503. For the proposed Naive Bayes model, the error happens in the matter that there are no direct correspondence between the images regions and the classes in Corel5k dataset which is for example an image used for the class 'sky' may also used for the class 'tree'. There are minority classes that are represented in some cases with one image in the training Corel5k set, the case that leads to hard classifications for these classes. Creating one Naive Bayes classifier too wasn't powerful than the methods proposed in the latest related researches. For future work, testing with different numbers of clusters may has a noticeable impact on the overall performance. Also, applying features selections and weighting techniques or using different features than the ones generated in the Corel5k is another point of research. Moreover, changing the number of Naive Bayes classifiers should leads to new results.

References

[1] W. M. Arnold Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval at the end of the early years. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22:1349-1380, 2000.

[2] Y. Mori, H. Takahashi, and R. Oka, "Image-to-word transformation based on dividing and vector quantizing images with words. " in *Proc. the International Workshop on Multimedia Intelligent Storage and Retrieval Management*, Orlando, Florida, USA, 1999.

[3] P. Duygulu, K. Barnard, J. Freitas, and D. Forsyth, "Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. " in *Proc. 7th European Conference on Computer Vision*, Copenhagen, Danemark, 2002, pp. 97–112.

[4] K. Barnard and D. Forsyth, "Learning the semantics of words and pictures. " in *Proc. 8th IEEE International Conference on Computer Vision (ICCV 2001)*, Vancouver, British Columbia, Canada, 2001, vol. 2, pp. 408–415.

[5] J. Jeon, V. Lavrenko, and R. Manmatha, "Automatic image annotation and retrieval using cross-media relevance models. " in *Proc. 26th Annual Int ACM SIGIR2003 Conference on Research and Development in Information Retrieval*, Toronto, Canada, 2003, pp. 119–126.

[6] V. Lavrenko, R. Manmatha, and J. Jeon, "A model for learning the semantics of pictures. " in *Proc. 17th Conference on Neural Information Processing Systems*, MIT, Cambridge, USA, 2003, vol. 16, pp. 553–560.

[7] M. Sami, N. El-Bendary, A. E. Hassanien, K. Xiao, and A. A. Manaf, "Particle Swarm Optimization based Multi-Class Image Annotation. " in *Proc. 35th IEEE International Conference on Telecommunications and Signal Processing (TSP2012)*, July 3-4, Prague, Czech Republic, 2012, pp. 700-705.

[8] M. Sami, N. El-Bendary, A. E. Hassanien, and Berwick R. C., "Incorporating Random Forest Trees with Particle Swarm Optimization for Automatic Image Annotation. " in *Proc. Federated Conference on Computer Science and Information Systems (FedCSIS 2012)*, 9 - 12 September, Wrocaw, Poland, 2012, pp. 791–797.

[9] R. Manmatha, V. Lavrenko, and S. L. Feng, "Multiple bernoulli relevance models for image and video annotation. " in *Proc. IEEE Comput. Soc. Conf. on Computer Vision and Pattern Recognition*, Washington, DC, USA, 2004, pp. 1002–1009.

[10] S. Zhu and X. Tan, "A Novel Automatic Image Annotation Method Based on Multi-Instance Learning. " *Procedia Engineering*, vol. 15, pp. 3439–3444, 2011.

[11] J. Kennedy and R. Eberhart, "Particle Swarm Optimization. " in *Proc. IEEE International Conference on Neural Networks*, Perth, Australia, 1995, vol. 4, pp. 1942–1948.

[12] I. C. Trelea, "The Particle Swarm Optimization Algorithm: convergence analysis and parameter selection. " *Information Processing Letters*, vol. 85, no. 6, pp. 317–325, 2003.

[13] M. Clerc and J. Kennedy, "The particle swarm - explosion, stability, and convergence in a multidimensional complex space. " *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 58–73, 2003.

[14] P. Viswanath and T. H. Sarma, "An improvement to k-nearest neighbor classifier. " in *Proc. IEEE International Conference on Recent Advances in Intelligent Computational Systems (RAICS)*, Kerala, India, 2011, pp. 227-231.

[15] H. Qiang and J. Aibing, "Multiple Real-valued K nearest neighbor classifiers system by feature grouping. " in *Proc. IEEE International Conference on Systems Man and Cybernetics (SMC)*, Istanbul, Turkey, 2010, pp. 3922–3925.

[16] L. Lin, L. Qi, L. Jun-yong, and L. Chuan, "An improved particle swarm optimization algorithm. " in *Proc. IEEE International Conference on Granular Computing (GrC 2008)*, Hangzhou, China, 2008, pp. 486–490.

[17] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. 1st ed, New York: Chapman and Hall, 1984.

[18] P. Domingos and M. Pazzani, *On the Optimality of the Simple Bayesian Classifier under Zero-One Loss*. " *Mach. Learn.*, vol. 20, pp. 103–130, 1997.

[19] H. Zhang, *The Optimality of Naive Bayes*. " in *Proc.the 17th International FLAIRS conference*, Florida, USA, 2004, pp. 17-19.

[20] G. John and P. Langley , *Estimating Continuous Distributions in Bayesian Classifiers*. " *In Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*, pp. 338–345, 1995.

[21] Corel5k dataset website. [Online]. Available: $http://kobus.ca/research/data/eccv_2 002/$.