OUTLIER DETECTION FROM POOLED DATA FOR IMAGE RETRIEVAL SYSTEM EVALUATION

Wei Xiong[1], S.H. Ong[2], Joo Hwee Lim[1], Qi Tian[1],
Changsheng Xu[1], Ning Zhang[2], and Kelvin Foong[3]

[1]Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore 119613
[2]Department of ECE, National University of Singapore, Singapore 117576
[3]Department of Preventive Dentistry, National University of Singapore, Singapore 119074

ABSTRACT

Widely used in the evaluation of retrieval systems, the pooling method collects top ranked images from submitted retrieval systems resulting in possibly a very large pool of images. Inevitably, the pool may contain outliers. Human experts then manually annotate the relevance of them to create a ground truth for evaluation. Studies show that this annotation is time-consuming, tedious and inconsistent. To reduce human workload, this paper introduces an automatic method to detect outliers. Different from traditional detection methods using unsupervised techniques only, we utilize both supervised and unsupervised techniques sequentially as both positive and negative examples are (partially) available in this context. Specifically, support vector machines (SVMs) and fuzzy c-means clustering are used to predict data relevance and “outlier-ness”. Performance improvements using our method after outlier removal have been validated on the medical image retrieval task in ImageCLEF 2004.

***Index Terms*—**Image classification, Image recognition, Pattern recognition, Pattern clustering, Pattern classification

1. INTRODUCTION

Widely used in the evaluation of image retrieval systems, the pooling method collects top ranked images from a number of submitted retrieval systems resulting in possibly a very large pool of images. Inevitably, the pool may contain outliers [1] due to performance limitations of the source systems. Human experts then manually annotate the relevance of them to create a ground truth for evaluation. Manual relevance annotation of a large number of samples is time-consuming, tedious and inconsistent. Müller et al. [2] studied the relevance assessments for the medical image retrieval tasks in ImageCLEF 2004 and ImageCLEF 2005 and found that, each expert needed about 1 hour and 3 hours to annotate images for each topic in ImageCLEF 2004 and ImageCLEF 2005 respectively. The difference in relevance judgments cannot be ignored either.

To reduce the workload of assessors, we can preprocess the pooled data, remove the possible outliers and highlight them for human’s special attention. We treat this setting as a task of outlier detection and removal [1,3] but embedded in the context of image retrieval and its results evaluation. Traditionally, outlier detection uses unsupervised learning techniques, based on analyses of distribution, depth, clustering, distances, densities, etc [1,3,4]. More recently, [4] introduces a supervised approach, support vector machine (SVM), to detect outliers based on user-given outlier examples and feedback. Active learning [5] also utilizes SVM and manual relevance feedback to learn query concepts in the context of retrieval. Thus outlier detection has strong connections to query concept learning.

Here we introduce an automatic outlier detection method to remove outliers and preprocess the pooled images. Different from existing methods, we utilize both supervised and unsupervised techniques sequentially. This is viable as both positive and negative data are (partially) available in this context when query topics are given and retrieval results of multiple systems for the designed topics are presented. SVMs are utilized to measure relevance. However, we do not use manual feedbacks as [3] and [5], although we can highlight those detected outliers for user’s special attention. Instead, we use the fuzzy c-means approach to cluster the relevance scores derived from SVMs. This enables us to detect and remove outliers automatically from the pooled data. With the outliers removed, the refined system can improve retrieval performance.

2. METHODOLOGY

2.1. SVM for Measure of Relevance

We employ SVMs to measure data relevance. Formally, given training samples $\{(\mathbf{x}_i, z_i)\}$, $\mathbf{x}_i \in \mathbb{R}^m$, $z_i \in \{-1,1\}$, $i = 1,,,N$, and subject to $\sum_{i=1}^{N} \alpha_i z_i = 0$ and $0 \le \alpha_i \le C$, the

1-4244-0728-1/07/$20.00 ©2007 IEEE

two-class SVM is to find the Lagrange multipliers α_i such that the objective function [6]

$$J(\alpha)=\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j z_i z_j k(\mathbf{x}_i,\mathbf{x}_j)-\sum_{i=1}^{N}\alpha_i \qquad (1)$$

is maximized. This work uses the RBF kernel $k\left(\mathbf{x}_i,\mathbf{x}_j\right)=\exp\left(-\left\|\mathbf{x}_i-\mathbf{x}_j\right\|/2\sigma^2\right)$. Solving (1), we obtain the optimal $\hat{\alpha}_i$ of α_i, $i=1,,,N$, and bias b from the normal to the decision boundary $f\left(\mathbf{x}\right)=0$ [6] where

$$f\left(\mathbf{x}\right)=\sum_{i=1}^{N}\hat{\alpha}_i z_i k\left(\mathbf{x},\mathbf{x}_i\right)+b \qquad (2)$$

is the relevance score of any $\mathbf{x}\in\mathbb{R}^m$, to the positive class. We used SVM^{Torch} [11] for this task with default parameters (where the learning iteration will stop at a precision of 0.01).

2.2. Fuzzy C-means for Measure of Outlier-ness

We have analyzed the distribution of the relevance scores $f\left(\mathbf{x}\right)$ of the pooled data for positive classes. Basically, they spread widely but exhibit roughly two modes with tails, one around 1, the other around -1, with the later indicating obvious non-relevance. This can be observed in **Fig. 1** where two typical distributions are shown. In the left sub-figure, no data have negative relevant scores. In the right, however, some data have negative relevant scores.

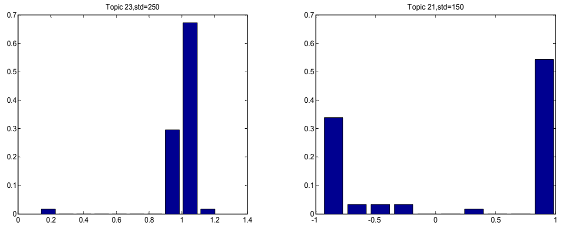

Fig.1. Two typical relevant score distributions. The horizontal axis is the relevance scores and the vertical axis is the probability.

To further classify data into outliers and non-outliers with such a wide spectrum of distributions, it is evident that we need to define a degree of outlier-ness. We introduce the fuzzy c-means clustering method [7], an unsupervised clustering algorithm which can specify a membership grade for each data point belonging to a cluster to some degree, with the cluster center(s) c determined. Let $g()$ refer to such a clustering analysis procedure, r the output, one has

$$r=g(f(\mathbf{x}),p), \qquad (3)$$

which uses as the inputs $f\left(\mathbf{x}\right)$ and a parameter p required by the fuzzy-c-means method. Finally the outlier-ness is given by

$$v=f\left(\mathbf{x}\right)-r\,. \qquad (4)$$

If $v<0$, then data $\mathbf{x}$ is an outlier. We determine parameter r as follows. We start from a guess of one cluster with $p=1$. In this case, the relevance scores are almost the same and $r=0$. Otherwise, we set $p=2$ and cluster the data into two groups. This results in two centers. Suppose c_1 is the smaller one and c_2 the larger. If $c_1<0$, then $r=(c_1+c_2)/2$, or else $r=0$.

2.3. The Proposed Algorithm

The image retrieval approach is summarized as follows.

a) Given a set of Q query topics and source retrieval systems, for each topic j, $j=1,,,Q$, construct a positive pool P_j^+ with N_j samples using the top ranked samples in the source systems with respective to topic j. Meanwhile, the top ranked samples, with respect to all other topics q, $q\neq j$, form another pool P_j^- of negative samples for j.

b) Train a two-class SVM using P_j^+ and P_j^- as training sets.

c) According to (2) find the algorithmic distances $f\left(\mathbf{x}_l\right)$ as relevance scores for each sample $\mathbf{x}_l$ in P_j^+, $l=1,,,N_j$. Rank $f\left(\mathbf{x}_l\right)$ for evaluations later.

d) Apply the fuzzy c-means algorithm to analyze these scores and calculate the reference parameter r_j according to (3).

e) Compute the outlier-ness v according to (4).

f) Remove outliers if $v<0$ and refine the positive pool P_j^+ to obtain a new pool $P_j^{+(s)}$.

g) Once all topics are processed, redo a) to f) if necessary. Otherwise just redo a) and c) to process all test data required.

3. EXPERIMENT DATA AND FEATURES

3.1. Datasets

The test dataset for the medical image retrieval task in ImageCLEF 2004 [8] is used here. It includes 26 image query topics (see **Fig.2**), and a database with 8725 medical images from various clinical routines and imaging modalities such as CT, X-ray, MRI, ultrasound, microscopy, PET, angiography, and even normal cameras. There are also drawings for teaching purpose. Using this common dataset enables us to compare our results with others.

Since we target to test outlier removal, we set up some artificial data to simulate the pooling evaluation routine. The initial training sets are manually selected from the test set only based on the 26 query examples by 5 non-medical undergraduate students. Since they are not domain experts, some outliers are chosen and there is relevance judgement variation among them. This procedure also happens in many training data collection applications. For each topic, each of them contributes about 12 images. After removing those duplicates, averagely about 51 images for each topic remain resulting in 1339 training samples in total for all 26 topics.

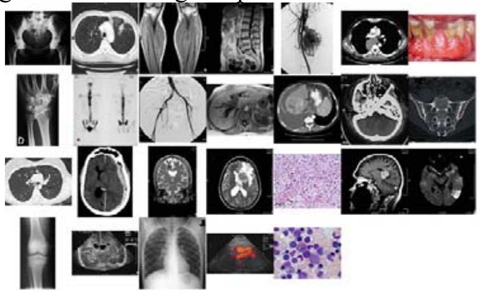

Fig.2. 26 query topic images in medical ImageCLEF 2004

3.2. Features

We employ two types of visual features, “icon” and “blob”. “Icon” is simply the thumbnail of an image. The size is 16-by-16. For color images, all color bands are averaged. Thus the feature is a 256-dimension vector at the pixel level. “Blobworld” was introduced in [9] to use colour, texture and shape features to represent images. Gaussian-mixture modelling and the expectation maximization algorithm are employed to model local regions and segment them. Here we construct the so-called “blob” feature [10] based on these local regions. A maximum of 10 local regions are used per image. “Blob” is a mixture representation of middle-level and high-level features including colour histograms of the image and its local objects. It also contains the shape parameters. They are 352-dimensional features [10].

3.3. Evaluation Measures

First, we define an indicator function $I(t,\mathbf{x}_l)$. $I(t,\mathbf{x}_l)=1$ if $\mathbf{x}_l$ is relevant to topic t. The number of relevant images in the top i returned ones is $u(t,i)=\sum_{l=1}^{i} I(t,\mathbf{x}_l)$. Suppose the ground truth is that there are $\hat{u}(t,N_t)$ images relevant to topic t in N_t testing images. For topic t, the non-interpolated average precision (AP) is defined by

$$AP(t,N_t)=\frac{1}{\hat{u}(t,N_t)}\sum_{i=1}^{N_t}\frac{u(t,i)}{i}I(t,\mathbf{x}_i). \qquad (5)$$

The set performance is evaluated by the mean average precision (MAP) over all the topics given:

$$a=MAP=\frac{1}{Q}\sum_{t=1}^{Q}AP(t,N_t). \qquad (6)$$

Here, $Q=26$ and $N_t=1000$, the same as those used for the medical image retrieval task in ImageCLEF 2004.

4. RESULTS AND DISCUSSION

It is evident that the reference parameter r for outlier detection can not be arbitrarily chosen. Algorithm performance depends on r, SVM parameters and the feature used. We have tested many sets of parameters. Due to space constraint, we only report $C=200$ in this work. Table 1 compares a few examples where different thresholds result in different performance in terms of MAP. In this table, $\sigma=150$ is used in the RBF kernel for SVM using “icon” feature. Before removing the outliers (corresponding to the case $r=--$ in Table 1), the MAP is $a=0.476$. After removing some outliers by using a threshold $r=0.001$, the performance improves to MAP=0.541 with an improvement by 13.66%. If we further increase the threshold to $r=0.5$, we get MAP=0.545 with an improvement by 0.74%. These results are already better than those systems submitted to the medical retrieval task in ImageCLEF 2004 [8,10]. For the “blob” feature, we choose $\sigma=10$. Before removing the outliers, MAP=0.329. Applying a threshold equal to $r=0.047$, MAP drops down to 0.279 with a decrease by -15.20%. This value is interesting because it is comparable to our published result (MAP=0.2618) presented in Table 2 of [10] for the same scheme “icon_SVM”. However, if we use a threshold equal to $r=0.5$, MAP jumps up to 0.308 with an improvement by 10.39% based on 0.279.

In general, the performance of the proposed algorithm depends on the value of the reference parameter r and it can be selected by using cross validation. However, we have observed that, in the above cases, $r=0.5$ gives improvements and this is used in our other experiments.

Table 1. Different thresholds result in different performance

Icon ($\sigma=150$)			Blob ($\sigma=10$)		
r --	0.001	0.5	r --	0.047	0.5

Icon ($\sigma=150$)				Blob ($\sigma=10$)			
r	--	0.001	0.5	r	--	0.047	0.5
a	0.476	0.541	0.545	a	0.329	0.279	0.308

In addition to features, the influence of thresholding also depends on the kernel parameters and the query topic. We have used our approach setting the thresholds and running the removal just once. The thresholds for both the “icon” feature and the “blob” range from 0 to 0.5 and the numbers of removed outliers vary from 0 to 38, with about 200 outliers in total from all training samples for some values of σ. Fig. 3 shows how parameter σ influences the system performance by comparing the APs before (marked by “+”) and after (marked by “o”) outlier removal. Each sub-figure corresponds to a query topic. In the top row, outlier removal improves AP for all σ presented for topics 16 and 18. In the bottom row, a mixture of improved and degraded performance is found. For topic 9 (bottom left), improvements only happen for larger σ . For topic 20 (bottom right), however, it is to the contrary. **Table 2** summarizes the system overall performance (MAP) before and after applying the outlier removal technique. It shows that, when σ is large for both features, the overall system performance improves.

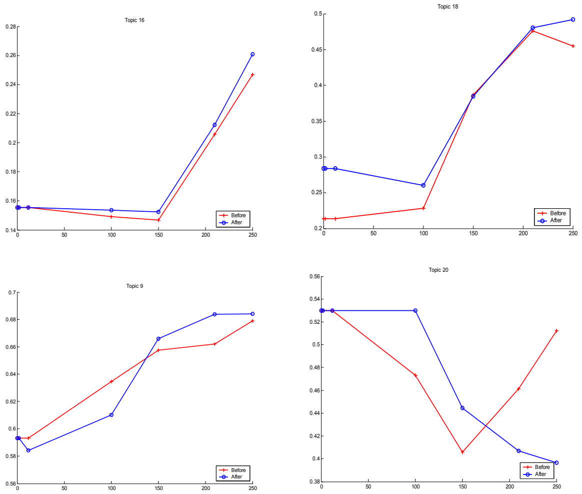

Fig.3. Kernel parameters σ influence the outlier removal performance. Each sub figure corresponds to a query topic. The horizontal axis is σ and the vertical axis is AP.

Table.2. MAP before and after the outlier removal (icon feature)

σ	MAP (“icon”) (before/after(change%))	MAP (“blob”) (before/after (change%))
0.1	0.3598/0.3551(-1.34)	0.3869/ 0.3792(-2.01)
2	0.3599/0.3551(-1.34)	0.3966/ 0.3810(-3.95)
30	0.3575/0.3634(+1.65)	--
50	--	0.2163/ 0.2151(-0.60)
100	0.4028/ 0.4088(+1.47)	0.2165/ 0.2134(-1.44)
150	0.4762/ 0.4877(+2.41)	0.2069/ 0.2120(+2.42)
250	--	0.1908/ 0.1953(+2.32)
400	--	0.1552/ 0.1618(+4.24)

5. CONCLUSION AND FUTURE WORK

We have studied the problem of outlier detection and removal in pooling data derived from multiple source sub-systems. Due to the limitations of sub-systems, it is inevitable that the pooled data contain outliers. Since manual annotating large sets of data are time-consuming and tedious for human, it is good to remove those outliers beforehand. Conventionally, outlier detection employs unsupervised learning techniques. This is largely due to the lack of training data. However, in the context of image retrieval by using the pooling method in the evaluation of multiple retrieval systems, multiple source data from these sub systems on different query topics are available. We thus can utilize both positive and negative training data for supervised learning. We first introduce SVM to classify the original data in the pooled set to obtain relevance scores. An unsupervised approach, fuzzy C-means, is then applied to the scores to define the outlier-ness and automatically remove outliers. We have tested our approach in the medical image retrieval benchmarking data for ImageCLEF 2004. Overall, we can improve the system performance.

However, the performance of outlier removal depends on kernel parameters and features used. Besides, here we set simple rules to find the reference parameter. In future, we plan to investigate optimized methods to define the outliers in retrieval context. We will study the performance gain with false alarms and compare our work with existing outlier detection methods quantitatively. We also intend to test our method on real pooling data.

6. ACKNOWLEDGEMENTS

The authors thank Bo Qiu for helpful discussions.

7. REFERENCES

[1] V. Barnett and T. Lewis. Outliers in statistical data, John Wiley and Sons, 1994.

[2] Müller H, Clough P, Hersh W, Geissbuhler A, Variations of relevance assessments for medical image retrieval, Workshop on Adaptive Multimedia Retrieval (AMR), Springer Lecture Notes in Computer Science (LNCS 3877), Geneva, Switzerland, 2006.

[3] Qing Song, Wenjie and Wenfang Xie. “Robust support vector machine with bullet hole image classification”. *IEEE Transactions on systems, man, and cybernetics*, vol. 32, no. 4, pp.440-448, Nov 2002.

[4] Cui Zhu, H. Kitagawa, S. Papadimitriou, and C.Faloutsos. “Example-based outlier detection with relevance feedback”. *DBSJ Letters*, vol. 3, no. 2, pp. 1-4, 2004.

[5] Simon Tong, Edward Y. Chang, “Support vector machine active learning for image retrieval”, *ACM Multimedia 2001*, pp. 107-118, 2001.

[6] G. Simon Haykin. *Neural Networks: A Comprehensive Foundation*, 2nd Ed, Prentice Hall, 1999.

[7] Bezdek, J.C., *Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, 1981.

[8] P. Clough, M. Sanderson and H. Müller, “The CLEF 2004 Cross Language Image Retrieval Track,” *Lecture Notes in Computer Science*, vol. 3491, pp. 579-613, 2005.

[9] C. Carson, S. Belongie, H. Greenspan, J. Malik, “Recognition of images in large databases using color and texture,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 8, pp. 1026-1038, 2002.

[10] Wei Xiong, Bo Qiu, Qi Tian, Changsheng Xu, Sim-Heng Ong, Kelvin Foong, Jean-Pierre Chevallet, “MultiPRE: a novel framework with multiple parallel retrieval engines for content-based medical image retrieval”. *ACM Multimedia 2005*, Singapore, pp. 1023-1032, 7-11, Nov 2005.

[11] Ronan Collobert and Samy Bengio. “SVMTorch: support vector machines for large-scale regression problems”. *Journal of Machine Learning Research*, v.1, pp.143-160, 2001.