division methods, reduction methods,
images processing, nearest neighbor rule,
representative measure

Marcin RANISZEWSKI[1]

FAST REDUCTION OF LARGE DATASET FOR NEAREST NEIGHBOR CLASSIFIER

Accurate and fast classification of large data obtained from medical images is very important. Proper images (data) processing results to construct a classifier, which supports the work of doctors and can solve many medical problems. Unfortunately, Nearest Neighbor classifiers become inefficient and slow for large datasets. A dataset reduction is one of the most popular solution to this problem, but the large size of a dataset causes long time of a reduction phase for reduction algorithms. A simple method to overcome the large dataset reduction problem is a dataset division into smaller subsets. In this paper five different methods of large dataset division are considered. The received subsets are reduced by using an algorithm based on representative measure. The reduced subsets are combined to form the reduced dataset. The experiments were performed on a large (almost 82 000 samples) two-class dataset dating from ultrasound images of certain 3D objects found in a human body.

1. INTRODUCTION

Nowadays analyzing and processing huge amount of data in reasonable time is becoming more real. It is possible not only due to growing computing power of processors and memory sizes, but also because of more efficient algorithms, which can be successfully performed on slower hardware. And it is not about the development of entirely new algorithms for large data sets, but the adaptation and optimization of existing methods.

In pattern recognition we deal with the problem of data classification on the basis of training set, which is earlier available during the learning phase [1,2]. The training set consists of samples where each sample is described by quantitative or qualitative features and can be attributed to certain class (supervised learning) or not (unsupervised learning). During the learning phase the information contained in the training set is used to build the classifier (function which can predict class label for new sample) [1,2].

The tasks of pattern classification occur in the analysis of optical images in medicine [3], biology and optical quality control of industrial products. Recognizable objects are the pixels, and the features are, for example, the degrees of gray pixels from the surrounding pixels. Data processing can be automated using the methods of pattern classification. However, these methods should be adapted to operate on huge datasets [3].

In this article we consider supervised learning, samples with quantitative features and the Nearest Neighbor Rule (NN) as a classifier. NN is popular and widely-used method in pattern recognition [4]. New sample is classified to the class of its (in the sense of distance) nearest neighbor. The main drawback of NN is storing the entire training set in the process of classification. It can cause the lack of memory but, what is worse, seriously slow down the process of classification. One solution to this problem is the selection from the entire training set only that samples which are the most important in the terms of the classification. If the training set is significantly reduced as compared to its original size, the NN will work quickly and efficiently.

There are many reduction algorithms [5]. However, the time of reduction phase may be counted in days or even weeks for large training sets (datasets with thousands or millions samples). The reduction idea used in these algorithms is good and there is no need to change it. That what should be changed is

[1] Computer Engineering Department, Technical University of Lodz, Stefanowskiego 18/22, Lodz, Poland.

the size of the training set, on which the algorithm currently operates. Of course, the optimization may speed up the algorithm but, generally, effectiveness of optimization depends on the specific method of reduction and the implementation language.

The presented in this paper approach to reduction of a large dataset is very simple and can be used for every reduction method. The large training set is divided into smaller parts, before the start of reduction. Then each part of the divided set is reduced. The results are combined to form the reduced dataset.

In this article five different dataset division will be discussed and tested. Reduction algorithm based on representative measure [6] as the reduction method and Liver dataset [7] (almost 82 000 samples) as the training set will be used in these tests.

2. REDUCTION ALGORITHM BASED ON REPRESENTATIVE MEASURE

Reduction Algorithm based on Representative Measure (RARM) [6] creates reduced set from samples which actually have the highest representative measure. The representative measure (*rm*) of sample *x* is the number of samples (called voters) from the same class as *x*, which are less distanced to *x* than to the nearest neighbor from opposite class (fig. 1).

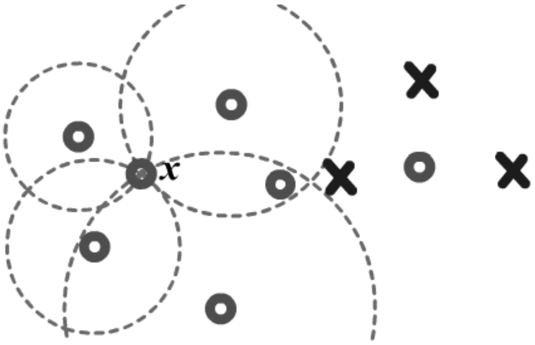

Fig. 1. The representative measure of sample *x* is equal to 4.

After adding one sample to a reduced set RARM recounts values of *rm* for the other samples from a training set (the voters are no longer taken into account) and then again selects a sample with the highest *rm*. The procedure is continued until the highest *rm* is less than or equal to the previously determined minimum of the parameter *rm*.

If two or more samples have the same value of *rm*, the priority value [8] is calculated:

$$priority(x) = \sum_{j} \frac{1}{d(x, x_j)^2}, \tag{1}$$

where:

x_j – is a voter of x

$d(x, x_j)$ – is a distance between samples x and x_j (if sample x has no voters, the priority is set to 0)

Then, the sample with the highest priority is added to the reduced set. To ensure that the denominator in the formula (1) will be different from zero, all overlapping samples are removed before the RARM start.

RARM returns simultaneously ten reduced sets corresponding to ten different minima of the parameter *rm* that equal to 9,8,…,0 respectively. The user can choose the reduced set, which satisfies his needs in the highest degree (we will evaluate the classification quality of reduced set using the cross validation method [9] and choose the set with the highest classification quality).

The algorithm acts in reasonable time for datasets up to 5000 samples like many other reduction procedures. For datasets with tens of thousands samples, there are the time-consuming problems, especially for methods like RARM, which uses additional tests to choose the best reduced set.

3. LIVER DATASET

In pattern classification, datasets containing more than 10000 items are considered to be large. Processing and classification of medical images is often associated with the operation on such datasets. In this article, the tests were performed on the Liver dataset, containing over 80000 items.

The Liver dataset was described in [7]: "Data set comes from ultrasound images that are sections of certain 3D objects found in a human body (liver). Pattern classification is used for segmentation of the images. The most usable information is contained in gray level distribution of the investigated pixel neighborhood. By an application of the orthogonal discrete wavelet transforms, 13 features were extracted. Only two classes of pixels were taken into account, i.e. class 1 that represents the objects (metastasis) of interest and class 2 that denotes the background (liver areas without metastasis)."

The Liver dataset contains 81968 samples (pixels), 10778 and 71190 from the first and the second class respectively.

4. LARGE TRAINING SET DIVISIONS

Many well-known reduction algorithms operate in acceptable time if the training sets contain up to 5000 - 10000 samples. The reduction phase also depends on the number of features describing each sample. For large datasets (datasets containing over 10000 samples) the methods become time-consuming or are not feasible because of the memory overhead. Simple solution to this problem is the division of the training set into smaller subsets which are then subject to reduction.

The division can be accomplished in many ways. In this paper five divisions are described and tested:

- simple division according to a sample order in a training set (SD – *Simple Division*) – a dataset is divided into specified number of smaller separable subsets of equal proportions between classes and approximately equal sizes; the selection order of samples is consistent with the sequence of samples in the dataset;
- random division (RD – *Random Division*) – a dataset is divided into specified number of smaller separable subsets of equal proportions between classes and approximately equal sizes; the selection order of samples is random;
- random division with repetitions (RDR – *Random Division with Repetitions*) – specified number of smaller inseparable subsets of equal proportions between classes and equal sizes are created (the size is equal to the number of samples in original dataset divided to the number of divisions); the selection order of samples is random and the sample repetitions are possible.
- division according to the values of *rm* of samples (RMD – *Representative Measure Division*) – a dataset is divided into specified number of smaller separable subsets of approximately equal sizes; the selection of samples is based on the decreasing arragemment of their *rm*: subsets are initially empty; the first sample is placed in the first subset, the next sample in the second subset; the sequential sample is placed in the third subset and so on; after placing a sample in the last subset, the sequential sample is put in the first subset and this procedure is continued until all samples will be placed in one of the subsets;
- division according to the values of *rm* of samples and class labels (RMDC – *Representative Measure Division of Classes*) – a dataset is divided into specified number of smaller separable subsets of equal proportions between the classes and approximately equal sizes; the selection order of samples is based on the class labels and the decreasing values of their *rm*: inside each class the samples are arranged in decreasing order of their *rm*; the subsets are initially empty; one sample from each class associated with the highest *rm* is placed in the first subset; next, in the similar manner, one sample from each class from the remaining set is placed to the second subset; this procedure is continued for the sequential subsets and then continued again for the

first subset and so on; finally, the samples from the larger class are distributed between all subsets in the same way as it took place in the case of RMD;

In RMD and RMDC division methods samples are initially sorted according to decreasing values of their *rm*. If two or more samples have the same value of *rm*, the priority value (1) is calculated and the samples are added to subsets according to their priority values. Distances of overlapping samples (equal to 0) are ignored in the calculation of the priority.

In each division we should determine the number of resulting subsets. The size of these subsets should be small enough that the reduction proceeds in an acceptable time.

In each of the proposed divisions, excluding RMD, the proportion between classes is maintained. Therefore, it is possible but unlikely that one or more of the RMD resulting subsets will contain only the samples from one class.

Also, in each of the proposed divisions, excluding RDR, the resulting subsets are separable. Therefore, as a result of RDR some samples can not be added to any of the subsets, while some can be added several times.

Exemplary divisions to three subsets of the training set from table 1 are presented in table 2.

Table 1. Exemplary training set.

no. of sample	class label	rm	priority
1	1	2	3,5
2	1	0	0,0
3	1	0	0,0
4	2	1	2,0
5	2	3	1,5
6	3	1	0,5
7	3	2	2,7
8	3	2	4,0
9	3	1	2,4
10	3	0	0,0

Table 2. Exemplary divisions to three subsets of the training set from table 1.

	no of samples in subsets		
division	I subset	II subset	III subset
SD	1,6	2,4,7,8	3,5,9,10
RD	2,8	1,5,9,7	3,4,6,10
RDR	7,4,1	3,6,2	10,4,5
RMD	5,7,6,10	8,9,3	1,4,2
RMDC	1,5,8,6	3,4,7,10	2,9

5. EXPERIMENTAL RESULTS

All algorithms were implemented in Java. Tests were conducted on Pentium Dual-Core CPU T4200 @ 2.00 Ghz with 4 GB RAM.

Liver dataset was divided into two separable parts: testing and training in 1:3 proportion. Training set (containing 61477 samples) was divided into 8 parts using five different division methods (section 4). Each part (containing approx. 7685 samples) was reduced by RARM (section 3) – see table 3. Finally, all reduced 8 parts from each division method were merged into one reduced set. Overlapping samples were removed – see table 4. Testing set was used to test the classification quality of the final reduced training sets (using NN) – see table 5.

Table 3. The number of samples in reduced subsets.

division	I subset	II subset	III subset	IV subset	V subset	VI subset	VII subset	VIII subset	sum
SD	317	55	14	12	20	25	23	31	497
RD	532	499	522	525	512	544	526	514	4174
RDR	495	520	518	511	521	524	526	517	4132
RMD	539	516	532	520	528	516	532	514	4197
RMDC	520	528	519	541	522	529	526	517	4202

Table 4. The number of samples in merged reduced sets before and after removal of overlapping samples.

division	before	after
SD	497	495
RD	4174	4160
RDR	4132	3732
RMD	4197	4186
RMDC	4202	4189

Table 5. Reduction levels and classification qualities of reduced sets from different division methods. Reduction level is a fraction of discarded samples counted after removal of overlapping samples. The first line ("-" in column "division") describes results of NN classification using complete training set.

division	class. qual.	red. level
-	98,42%	0,00%
SD	87,71%	99,19%
RD	98,09%	93,23%
RDR	97,96%	93,93%
RMD	98,01%	93,19%
RMDC	98,14%	93,19%

6. DISCUSSION AND CONCLUSSIONS

The results of Liver dataset reduction indicate no significant difference between RD, RDR, RMD and RMDC division methods. Dataset was reduced to a similar extent, and the classification quality obtained on the testing set is comparable (reduction levels approx. equal to 93% and classification qualities approx. equal to 98% - see table 5). So strong reduction decreased only slightly (from 98,42% to approx. 98,05%) the classification quality (compared to classification quality obtained on complete training set) – see table 5.

If we look at the results of SD division method, we find that in the Liver dataset samples are not independently arranged: the first subset was reduced to 317 samples, whereas the latter only to some tens of samples - see table 3. These high reductions can be explained by the fact that the SD subsets contain a large number of samples of the few class areas. Many important border samples are missed and the classification quality strongly decreased (to 87,71%). Of course, the reduction is strong and exceeds 99% - see table 5.

RMD and RMDC division methods use representative measure, which have to be counted for each sample. This is an additional cost which apparently does not produce visible benefits in the Liver dataset reduction.

RD and RDR division methods seem to be the best. Their implementation is simple and the results are very promising. However, both these procedures are fully random, which means that the results obtained are not repeatable.

The average duration of reduction phase (including the division methods) last approx. 23 hours, while duration of complete training set reduction would require about a month. Time gain achieved by dividing the set into smaller subsets is therefore more than 30 times.

RARM uses cross validation method (see section 3) to choose the best reduced set, what greatly extends the time of reduction. Durations of reduction phases of other reduction algorithms can be much shorter.

The described methods of dataset division may be helpful in reducing large datasets obtained, for example, from medical images. Reduction algorithms can then be used in their original implementation without using sophisticated optimization methods. Used methods of dataset division should be random, such as the RD and RDR. They are simple to implement and fast. The resulting reduction levels are satisfactory (RARM algorithm and the Liver dataset) and the NN classification qualities do not depart from the qualities obtained on the complete training set. Importantly, a reduction phase is much shorter then.

ACKNOWLEDGEMENTS

The author is a scholarship holder of project entitled "Innovative education without boundaries - integrated development of Technical University of Lodz - University management, innovative educational offer and an option to enhance the possibilities to take on new employees, also disabled ones" supported by European Social Fund.

BIBLIOGRAPHY

[1] DUDA R.O., HART P.E., STORK D.G., Pattern Classification – Second Edition, John Wiley & Sons, Inc, 2001.

[2] THEODORIDIS S., KOUTROUMBAS K., Pattern Recognition – Third Edition, Academic Press - Elsevier, 2006.

[3] MEYER-BAESE A., Pattern Recognition in Medical Imaging, Elsevier Academic Press - Elsevier, 2003.

[4] DASARATHY B.V., NN Pattern Classification Techniques, IEEE Computer Society Press, 1991.

[5] WILSON D.R., MARTINEZ T.R., Reduction techniques for instance-based learning algorithms, Machine Learning, Vol. 38, No. 3, 2000, pp. 257–286.

[6] RANISZEWSKI M., The Edited Nearest Neighbor Rule Based on the Reduced Reference Set and the Consistency Criterion, Biocybernetics and Biomedical Engineering, Vol. 30, No. 1, 2010, pp. 31-40.

[7] JÓŹWIK A., KIEŚ P., Reference set reduction for 1-NN rule based on finding mutually nearest and mutually furthest pairs of points, Advances in Soft Computing, Computer Recognition Systems, Springer-Verlag, Berlin-Heidelberg, 2005, pp. 195-202.

[8] XI X., KEOGH E. J., SHELTON C., WEI L., RATANAMAHATANA C. A., Fast Time Series Classification Using Numerosity Reduction, ICML, 2006, pp. 1033-1040.

[9] KOHAVI R., A study of cross-validation and bootstrap for accuracy estimation and model selection, Proc. 14th Int. Joint Conf. Artificial Intelligence, 1995, pp. 338–345.