Learning from Small Data Set to Build Classification Model: A Survey

Nidhi H. Ruparel
K.K.W.I.E.E&R, Nashik,

Nitin M. Shahane
K.K.W.I.E.E&R, Nashik

Devyani P. Bhamare
S.R.E.S COE, Koparagaon

ABSTRACT

Classification is one of the important data mining techniques. Learning from a given data set to build a classification model becomes difficult when available sample size is small. How to extract more effective information from a small data set is thus of considerable interest. In this paper we provide a review of different classification methods which will help us build more amounts of data, so that classification performance is improved. We discuss different techniques which will work with small data set such as attribute construction, bootstrap method, incremental method and different diffusion functions. Different classification methods such as neural network, decision tree classifiers, Bayesian classifiers etc. are also discussed.

General Terms

Data Mining, Classification

Keywords

Data preprocessing, Small Data Set, Attribute Construction, SMO.

1. INTRODUCTION

Classification is one of the important data mining forms. As databases are rich with hidden information which can be used for intelligent decision making, Classification can be used to extract models describing important data classes. Such analysis can helps us to understand large amount of data. Classification predicts categorical (discrete, unordered) labels. For example, we can build a classification model to categorize bank loan applications as either safe or risky.

According to the computational learning theory, sample size in machine learning problems has a major effect on the learning performance. Faced with this issue, adding some artificial data to the system in order to accelerate acquiring learning stability and to increase learning accuracy is one effective approach.

To build a correct classification model sufficient amount of training data is required. But In the real world, there are many situations when organizations must work with small data sets. For example, with the pilot production of a new product in the early stages of a system, dealing with a small number of VIP customers, and some special cancers, such as bladder cancer for which there are only a few medical records. [1]

In the early time of a new system development, data on hand are not enough, hence, data characteristics such as data distribution, mean, and variance are unknown. As well as a decision is hard to make under the limit data condition.

A “small” data set is very much a relative and subjective concept that needs to be defined. In many multivariable classification or regression (e.g., estimation or forecasting) problems we have a training set $T_p = (x_i, t_i)$ of p pairs of input/output vector $x \in \Re_n$ and scalar target t, and the unfortunate circumstance that T_p is according to Vapnik:

"For estimating functions with VC dimension h, we consider the size p of data to be small if the ratio p/h is small (say $p/h < 20$)"

This paper is organized as follows. Section 2 describes the classification Techniques, Section 3 provides details of current data preprocessing methods which add data to overcome problem of small data set. Section 4 describes analysis of small data set. Finally, concluding remarks are given in Section 5.

2. CLASSIFICATION METHODS

2.1 Decision Trees:

Based on training data, a Decision Tree is built as a binary classification tree. Each internal node tests a feature to determine class which is labeled at leaf nodes. For new unlabeled instances, the prediction is made by a path from root to leaf node according to features properties of a new instance. The class of new instance is labeled when reaching the leaf node. To construct a tree, features in each node are selected from top to bottom by calculating the information gain of features, which reduces the entropy by separating instances.

2.2 Support Vector Machines:

In today’s machine learning applications, support vector machines (SVM) are considered a must try—it offers one of the most robust and accurate methods among all well-known algorithms. It has a sound theoretical foundation, requires only a dozen examples for training, and is insensitive to the number of dimensions. In addition, efficient methods for training SVM are also being developed at a fast pace.

In a two-class learning task, the aim of SVM is to find the best classification function to distinguish between members of the two classes in the training data. The metric for the concept of the “best” classification function can be realized geometrically. For a linearly separable dataset, a linear classification function corresponds to a separating hyper plane *f (x)* that passes through the middle of the two classes, separating the two. Once this function is determined, new data instance *xn* can be classified by simply testing the sign of the function *f (xn)*; *xn* belongs to the positive class if *f (xn)* > 0.

Because there are many such linear hyperplanes, what SVM additionally guarantee is that the best such function is found by maximizing the margin between the two classes. Intuitively, the margin is defined as the amount of space, or separation between the two classes as defined by the hyper plane. Geometrically, the margin corresponds to the shortest distance between the closest data points to a point on the hyper plane. Having this geometric definition allows us to explore how to maximize the margin, so that even though there are an infinite number of hyper planes, only a few qualify as the solution to SVM.

2.3 Bayesian Classification

The Bayesian Classification represents a supervised learning method as well as a statistical method for classification. It provides a useful perspective for understanding and evaluating many learning algorithms. It calculates explicit probabilities for hypothesis and it is robust to noise in input data. It is based on the Bayesian theorem. It is particularly suited when the dimensionality of the inputs is high. Parameter estimation for naive Bayes models uses the method of maximum likelihood. In spite oversimplified assumptions, it often performs better in many complex real world situations. It requires a small amount of training data to estimate the parameters [5]

2.4 Neural Network Classifier:

Neural networks have emerged as an important tool for classification. The recent vast research activities in neural classification have established that neural networks are a promising alternative to various conventional classification methods. The advantage of neural networks lies in the following theoretical aspects. First, neural networks are data driven self-adaptive methods in that they can adjust themselves to the data without any explicit specification of functional or distributional form for the underlying model. Second, they are universal functional approximators in that neural networks can approximate any function with arbitrary accuracy. Neural networks are nonlinear model-free method. That the outputs of neural networks are least square estimates of the Bayesian *a posteriori* probabilities [6]

3. TECHNIQUES FOR SMALL DATA SET CLASSIFICATION:

Many studies have been conducted for learning from data set, which are discussed in this section:

3.1 Diffusion-neural-network (DNN)

In this method the principle of information diffusion is combined with a traditional neural network, called a diffusion-neural-network (DNN), for functional learning according to the results of their numerical experiments, the DNN improved the accuracy of the Backpropagation Neural Network (BPN). The information diffusion approach partially fills the information gaps caused by data incompleteness via applying fuzzy theories to derive new samples, but the research does not provide clear indications for determining the diffusion functions and diffusion coefficients. Besides, the symmetric diffusion technique sometimes over simplifies a generation of new samples, which could cause over-estimation of the domain range. Either under estimating or over-estimating the ranges would lead to reduced accuracy. Therefore, in order to fully fill the information gaps, a technique called mega trend diffusion was substituted a sample set for diffusing samples one for one. Furthermore, a data trend estimation concept is combined with the mega diffusion technique to avoid over-estimating. This technique, which combines mega diffusion and data trend estimation, was called mega-trend-diffusion. Following mega-trend-diffusion, the production of virtual samples was proposed to improve the FMS scheduling accuracy. Unfortunately, in their research, the DNN is adopted to extract knowledge. The DNN has twice as many input factors as original ones, and this situation means the network has much more complex calculations than the ANN [7]

3.2 Generalized Trend Diffusion Modeling:

This method starts by considering the observations that are collected with an empty set, where the incoming data appear over time. The central location (CL) of data is described and trivially located as the procedure progresses, all the points in a data set appears to be in a batch. However, the occurrence sequence of data makes incremental learning unique and more informative than batch learning. Therefore a membership function is formulated to catch the possible changing of the estimated population at each step. When computing the value of the membership degree, a triangle type function is employed. The uni-modal membership functions assume that the true mean location is located where the peak is. Hence, bell-shaped or curve-shaped functions can also be employed if they are uni-modal. Nominal data should be transformed into the ordinal or numeric scale before applying GTD. GTD, is developed to extract information for predicting successive observations. When processing this approach, it generates shadow data employing the real data and the occurrence order of the observed data. Then, it quantifies the importance degree for both of the observed and shadow data by computing the membership function values based on fuzzy theories. Based on the technique, GTD can further systematically help the knowledge acquisition process to collect additional hidden data-related information, which the limited data set itself does not explicitly provide [8]

3.3 Bootstrap Method:

This method is used for pilot run modeling of manufacturing systems where initial data set is small. Using the limited data obtained from pilot runs to shorten the lead time to predict future production is considered in this method. Although, artificial neural networks are widely utilized to extract management knowledge from acquired data, sufficient training data is the fundamental assumption. Unfortunately, this is often not achievable for pilot runs because there are few data obtained during trial stages and theoretically this means that the knowledge obtained is fragile. The bootstrap implies re-sampling a given data set with replacement and is used for measuring the accuracy of statistical estimates. The bootstrap is applied to generate virtual samples in order to fulfill the data gaps. But, the bootstrap procedure is executed once for each input factor not to resample a job. With the help of this method the error rate can be significantly decreased if applied to a very small data set.

3.4 Mega Trend Diffusion Function

The method goes this way to extend the attribute information, collecting the data , building MTD functions, and computing the overlap area of the MTD functions, and then moves on to class-possibility attribute transformation, attribute construction, attribute merging, and finally SVM model building. After data collection, this method begins to build the triangular fuzzy membership function (called the megatrend diffusion function) for each class in every attribute, and then computes the overlap area of the fuzzy membership functions of each class. When the overlap area of membership functions is high, this means the class-possibility method cannot judge the classes very clearly, and the attribute will thus be analyzed by attribute construction. The attributes with low overlap area of membership functions will be examined by the class-possibility method, in which the class-possibility values are computed using fuzzy membership function called mega trend diffusion function. After constructing the attributes by class possibility and synthetic attributes, both the tables are merged to form a data set with high dimensions which can be applied to the classifier to build classification model.

4. Analysis of small Data Set

In this Section, an example of small data set is discussed and using Mega Trend diffusion Technique, how the small data set is converted to Extended Data set is shown.

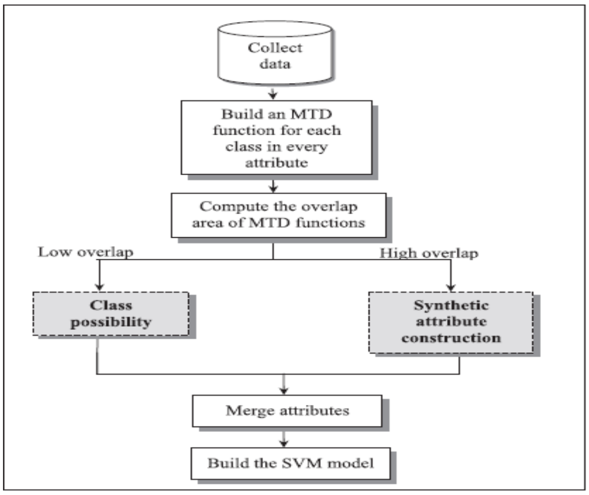
Collect data
Build an MTD function for each class in every attribute
Compute the overlap area of MTD functions
Low overlap
High overlap
Class possibility
Synthetic attribute construction
Merge attributes
Build the SVM model

Fig. 1 The Attribute construction approach [1]

As shown in the figure, the method collects data, Builds MTD function for each class in every attribute and then computes overlap area of MTD functions. Overlap area of MTD functions is calculated to decide whether to build class possibility values or synthetic attributes based on low overlap area or high overlap area respectively. After building the attributes, they are merged to apply it to classifier build model.

As Shown in the Table 1, the sample data set is small as it is having only five records in it. There are two attributes available in the data set and this data set contains two class labels as A and B. To extend the data set into higher dimensional data set, the attributes are partitioned into two classes. By using the following function, boundaries of the mega trend diffusion function are identified.

Table 1: Sample Data Set

Data	Attribute 1	Attribute 2	Class
X1	0.10	0.50	A
X2	0.60	0.40	A
X3	0.85	0.12	B
X4	0.35	0.50	B
X5	0.30	0.80	A

$a=U_{set} - skew_U \times \sqrt{(-2) \times S_X^2 / N_U \times \ln(f(t))}$

$b=U_{set} + skew_U \times \sqrt{(-2) \times S_X^2 / N_U \times \ln(f(t))}$

Where,

U_{set}= Core of the data set. (min+max)/2

$skew_L = N_L / (N_L + N_U)$,

$skew_U = N_U / (N_U + N_L)$,

$S^2_x = \Sigma^n_{i=1}(x_i - x)^2/(n-1)$,

$\ln(f(t)) = 10^{-20}$

N_L and N_U are the measure of skewness in the distribution of the data and are number of data points smaller and greater than U_{set} respectively.

Table 2: The Transformation Value of X for low overlap

	x1	M^A (x1)	M^B (x1)	x2	M^A (x2)	M^B (x2)	Class
X1	0.10	0.69	0.56	0.50	0.85	0.78	A
X2	0.60	0.69	1.00	0.40	0.70	0.90	A
X3	0.85	0.38	0.78	0.12	0.28	0.78	B
X4	0.35	1.00	0.78	0.50	0.85	0.78	B
X5	0.30	0.94	0.73	0.80	0.70	0.43	A

These values of a and b will be used to map triangular membership function, whose base is point a and b and height is 1.

As, there are two classes, for each attribute two triangles will be mapped and point of intersection of two will be found out to get the height of third triangle. Using following formula Overlap area of each attribute will be found out to get the average area. Then each attribute will be compared with the average value and based on that, the decision to build class possibility function or synthetic attribute construction will be taken.

$\text{Overlap Area}^i = \sqrt{\frac{\beta_o^i}{\beta_A^i} \cdot \frac{\beta_o^i}{\beta_B^i}}$

Average Overlap Area will be calculated by taking average of all the Overlap areas for all attributes (Overlap Area^i). Data sets are displayed in the following tables:

After preparing these two tables, attributes of both the data set are merged and are then applied to classifier. WEKA tool has been used as the machine learning tool for this experiment and the classifier used is SMO.

Table 3: Data Set that Contains Synthetic Attributes for high overlap

	a1	a2	a1*a2	a1/a2	a2/a1	Class
X1	0.10	0.50	0.05	0.2	5	A
X2	0.60	0.40	0.24	1.5	0.67	A
X3	0.85	0.12	0.10	7.08	0.14	B
X4	0.35	0.50	0.175	0.7	0.43	B
X5	0.30	0.80	0.24	0.375	2.67	A

The following table shows accuracy of sample data set with SMO classifier used in WEKA machine learning tool.

Table 4: Accuracies for sample data set with SMO

It can be seen from the table that classification accuracy has been increased after data set information is extended.

5. CONCLUSIONS

This paper presented survey of techniques for small data set learning to build classification model. We discussed a variety of classification strategies as well as different data pre-processing methods which will add data to data set which is small. After the data preprocessing, extended data set if applied to classifier, it learns well and accuracy of classification is increased.

Sr. No.	Data Set	Classification Accuracy (in terms of correctly classified instances)	Time to build model(in seconds)
1	Original Data set	60 %	0.2
2	Class Possibility Data Set	80%	0.02
3	Synthetic Attribute Data Set	80%	0.02
4	Merged Data Set	80%	0.03

6. REFERENCES

[1] Der-chiang Li and Chiao Wen Liu "Extending Attribute Information for Small Data Set Classification," IEEE Transactions On Knowledge And Data Engineering, Vol. 24, No. 3, March 2012.

[2] W.C. Li and C.W. Yeh, "A Non-Parametric Learning Algorithm for Small Manufacturing Data Sets," Expert Systems with Applications vol. 34, pp. 391-398, 2008.

[3] D.C. Li, C.S. Wu, T.I Tsai, and Y.S. Lina, "Using Mega-Trend-Diffusion and Artificial Samples in Small Data Set Learning for Early Flexible Manufacturing System Scheduling Knowledge," Computers and Operations Research, vol. 34, pp. 966-982, 2007.

[4] Kanthida Kusonmano, Michael Netzer, Bernhard Pfeifer, Christian Baumgartner, Klaus R. Liedl, and Armin Graber, "Evaluation of the Impact of Dataset Characteristics for Classification Problems in Biological Applications," World Academy of Science, Engineering and Technology 34 2009

[5]http://www.let.rug.nl/~tiedeman/ml05/03_bayesian_handout.pdf

[6]Guoqiang Peter Zhang "Neural Networks for Classification: A Survey," IEEE Transactions on systems, man, and cybernetics—part c: applications and reviews, vol. 30, no. 4, November 2000

[7] Chongfu Huang, Claudio Moraga, "diffusion-neural-network for learning from small samples," International Journal of Approximate Reasoning 35 (2004) 137–161

[8]Yao San Lin, Der Chiang Li, "The Generalized Trend Diffusion modeling algorithm for small data sets in the early stages of manufacturing systems"

[9]Tung-I Tsai, Der-Chiang Li , "Utilize bootstrap in small data set learning for pilot run modeling of manufacturing systems" Expert Systems with Applications 35 (2008) 1293–1300