ISSN (Print) : 2319-5940
ISSN (Online) : 2278-1021

International Journal of Advanced Research in Computer and Communication Engineering
Vol.2, Issue 12, December 2013

Prototype Selection Algorithms for kNN Classifier: A Survey

Shikha V. Gadodiya[1], **Manoj B. Chandak**[2]

M.Tech Student, CSE Department, Shri Ramdeobaba College of Engineering and Management, Nagpur, India [1]

Professor, CSE Department, Shri Ramdeobaba College of Engineering and Management, Nagpur, India [2]

Abstract: The k-Nearest Neighbor classifier is one of the most used and well-known techniques for performing recognition tasks but, it suffers from several drawbacks such as high storage requirements, low efficiency in classification response, and low noise tolerance. The most promising solution to overcome these drawbacks consists of reducing the data used for establishing a classification rule (training data) by means of selecting relevant prototypes. Prototype selection is a research field which has been active for more than four decades. As a result, a great number of methods tackling the prototype selection problem have been proposed yet. Different properties could be observed in the definition of these methods, but no formal categorization has been established yet. This paper provides a survey of the prototype selection method's categorization/taxonomy that could be considered relevant.

Keywords: k-NN classifier, prototype selection, data reduction, taxonomy.

I. Introduction

The Nearest Neighbor classifier is one of the most used and well-known nonparametric classifiers and widely used in Machine Learning and Data Mining (DM) tasks. k-NN is simple to implement still powerful and has been considered one of the top 10 methods in DM. k-NN belongs to a family of lazy learners that is it simply stores training data and waits until it is given a test data. Thus it suffers from several drawbacks: high storage requirements, low efficiency in classification response, and low noise tolerance. To overcome these weaknesses several solutions have been proposed and research is still going on. One promising solution is using prototype selection methods for classification, which belongs to a family of eager learners that is given a training set it first constructs a classification model before receiving new test data.

For classifying new prototypes a training set is used which provides information to the classifiers during the training stage. In practice, not all information in a training set is useful therefore it is possible to discard some irrelevant prototypes. This process is known as "prototype selection".

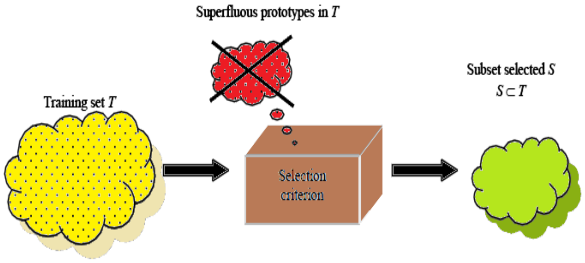

Fig. 1: Prototype Selection method

The advantage of Prototype Selection method is that it has the capacity to choose relevant examples without generating new artificial data. And dealing with large data sets is also possible with k-NN when Prototype Selection is applied to such large data.(using Stratification)

II. Prototype Selection Taxonomy

This section presents the taxonomy of PS methods and the criteria used for building it. Common properties in prototype selection method are: order of the search, type of selection, and evaluation of the search. These mentioned issues are involved in the definition of the taxonomy since they are exclusive to the operation of the PS algorithms.

A. Order of Search

When searching for a subset S of prototypes from training set TR, there are a various directions in which the search can be proceeded, which are as follows

i) Incremental Process:

An incremental search begins with an empty subset S, and incrementally adds each instance in TR to subset S if it fulfills some predefined criteria. If some instances are made available later, after training is complete, they can still continue to be added to S according to the same criteria without any additional efforts required. As a result it proves to be faster and requires less storage than non-incremental algorithms. The main disadvantage is that it have to make decisions based on little information and are therefore prone to errors until more information is available.

ii) Decremental Process:

The decremental search begins with subset S = TR (complete given data), and then decrementally searches for instances to be removed from S according to some predefined criteria. Again, the order of presentation is important, but unlike the incremental process, all of the training examples are available for examination at any time. One disadvantage of decremental algorithms is that it presents a higher computational cost than incremental algorithms. The learning stage must be done in an offline fashion. However, if the application of a decremental algorithm can result in greater storage reduction, then the extra computation during learning (which is done just once) can be well ignored at the benefit of computational savings during execution thereafter.

iii) Batch Process:
This involves deciding if each instance meets the removal criteria before removing any of them individually. Then, all those that do meet the criteria are removed at once. As with decremental algorithms, batch processing also suffers from increase in time complexity over incremental algorithms.

iv) Mixed Process:
A mixed search begins with a preselected subset S and can iteratively add or remove any instance which meets the predefined criteria. Its advantage is that it is easy to obtain good accuracy-suited subsets of instances. Note that these kinds of algorithms are closely related to the order-independent incremental approaches that does not allows removal of instances, but in this case, the instance removal from S is also allowed.

v) Fixed Process:
A fixed search is a subfamily of mixed algorithm in which the number of additions and removals remains constant that means, the number of final prototypes is determined at the beginning of the learning phase and is never changed.

B. *Type of Selection*

This factor is mainly conditioned by the type of search carried out by the PS algorithms, that means whether they seek to retain/remove the border points, central points, or some other set of points.

i) Condensation Approach:
The condensation techniques aims to retain the points which are closer to the decision boundaries and remove the interior points, because internal points do not affect the decision boundaries as much as the border points do, and thus can be removed with relatively little effect on classification. The idea behind these methods is to preserve the accuracy over the training set, but the generalization accuracy over the test set can be negatively affected which makes it slower. Nevertheless, the reduction capability of condensation methods is normally high due to the fact that there are fewer border points than internal points in most of the data.

ii) Edition Approach:
These kinds of techniques instead seek to remove the border points, that are more noisy or that do not agree with their neighbors. This removes close border points, leaving smoother decision boundaries behind. However, such algorithms do not remove internal points that do not necessarily contribute to the decision boundaries. The effect obtained is related to the improvement of generalization accuracy in test data, although the reduction rate obtained is lower since there are less border points as compared to internal points.

iii) Hybrid Approach:
Hybrid techniques try to find the smallest subset S which maintains or even increases the generalization accuracy in test data with proper reduction rate. To achieve this, it allows the removal of internal as well as border points based on criteria followed by the two previous approaches. The k-NN classifier is highly adaptable to these methods, thus obtaining greater improvements even with a very small subset of instances selected initially.

C. *Evaluation of Search*

k-NN is a simple technique and it can be used to direct the search of a PS algorithm. The objective pursued is to make a prediction on a non-definitive selection and to compare between selections. This characteristic influences the quality criterion and it can be divided into two subclasses:

i) Filter Class:
When the k-NN rule is used for partial data (that is initially just a part of data is made available) to determine the criteria of adding or removing and no leave-one-out validation scheme is used to obtain a good estimation of generalization accuracy then it comes under class of Filters. The fact of using subsets of the training data in each decision increments the efficiency of these methods, but the accuracy may not be enhanced.

ii) Wrapper Class:
When the k-NN rule is used for the complete training set made available at an instance with the leave-one-out validation scheme then it comes under class of Wrappers. The conjunction in the use of the two mentioned factors allows us to get a great generalization accuracy, which helps to obtain better accuracy over test data. However, in the techniques of this class each decision involves a complete computation of the k-NN rule over the training set and thus the learning phase can be computationally expensive.

D. *Main Properties needed for PS Methods*

When comparing PS methods, there are a number of properties that can be used to evaluate the relative strengths and the weaknesses of each algorithm. It includes:

i) Storage Reduction:
The main goal of the PS methods is to reduce storage requirements for the datasets. Further-more, another goal closely related to this is to speed up the classification.

Thus reduction in the number of stored instances will typically result a corresponding reduction in the time it takes to search through given test data and classify a new input vector.

ii) Noise Tolerance:

Two main problems that occur in the presence of noisy instances in training data are, first very few instances will be removed because many instances are needed to maintain the noisy decision boundaries. Second, the generalization accuracy can suffer, especially if noisy instances are retained instead of good instances. So the algorithm thus selected should be highly noise tolerant.

iii) Generalization Accuracy:

Generalization Accuracy of classification should be high. Thus a successful algorithm will often be able to significantly reduce the size of the training set without significantly reducing generalization accuracy.

iv) Time Requirements:

Generally, the learning process is done just once on a training set, so it does not seem to be that important evaluation property. However, if the learning phase is taking a long time then it can become impractical for real applications.

III. Prototype Selection Methods

Around 52 prototype selection methods have been proposed yet with 6 different families. The comparison network of these various methods is as shown in figure:

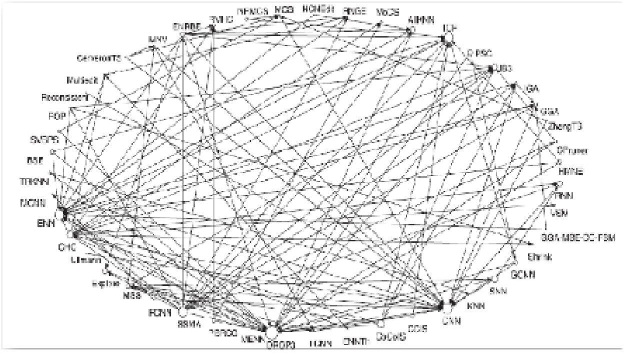

Fig. 2: Comparison Network of PS Methods

The taxonomy studied above can be used to categorize the PS methods proposed in the literature. The order of search, type of selection, and evaluation of the search may differ among PS methods and constitute a set of properties which are exclusive to the way of operating the PS methods.

- Condensation and Edition techniques display opposite behavior. IB3 was the first hybrid method which combines an edition stage with a condensation one. Since its proposal, there has been a significant effort in proposing new hybrid approaches, decreasing the proposals of condensation methods.

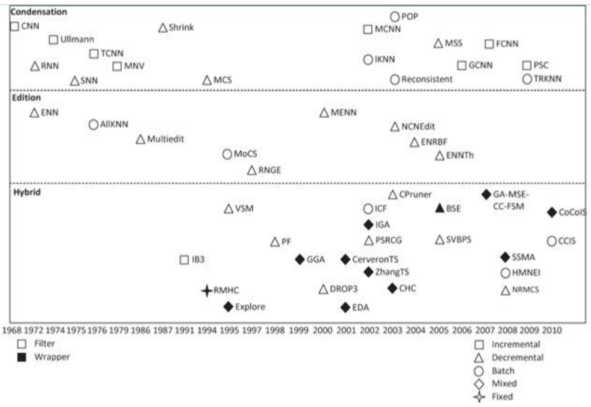

Fig. 3: Prototype Selection Methods Categorization

Fig. 3 depicts the categorization of PS methods. The figure allows us to point out the following interesting facts:

- Few edition methods have been proposed in comparison to the other two families. The main reason is that the first edition method, ENN, obtains good results in conjunction with k-NN and the edition approaches do not achieve high reduction rates, which is main goal of interest in PS. Incremental edition approaches have not been proposed because it is very important to know the complete set of data for identifying noisy instances.
- Recent efforts are being noted in proposing more condensation and hybrid approaches instead of edition approach. Both of them could be made in any direction search, but the mixed direction search is typical in hybrid methods and it is not presented in condensation methods.
- Wrapper evaluation searches are only found in hybrid approaches. This evaluation search is intended to optimize a selection, without thinking of computational costs. The resulting selection depends on the whole training set, whereas in edition and condensation approaches, the decision is made considering only local information.

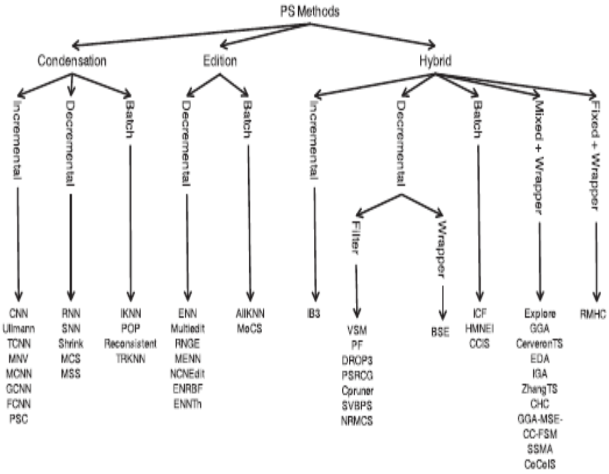

Fig. 4: Prototype Selection Taxonomy

Fig, 4 illustrates the categorization of prototype selection methods according to the taxonomy based on this order:

type of selection, direction of search, and evaluation of the search. It allows us to distinguish among families of methods and to estimate the size of each one.

IV. Comparison of Methods

Based on the classification shown in above fig. 4, we present the comparative experimental results of few methods which work on only small or medium scale data set. In this comparison data sets used are numeric (*glass, iris, liver, wine*) and mixed (*Bridges, Echocardiogram, Hearth Cleveland*). Following table shows results of accuracy (*Acc*) obtained and storage (*Str*) space required for Original training Set (*Orig*.), *DROP5, GCNN* and *TS.*
where,

- DROP5-Decremental Reduction Optimization Procedure 5,
- GCNN-Generalized Condensed Nearest Neighbour approach,
- TS-Tabu Search approach.

TABLE I
Comparison of PS Methods

Dataset	**Orig.**		**DROP5**		**GCNN**		**TS**	
	Acc	**Str**	**Acc**	**Str**	**Acc**	**Str**	**Acc**	**Str**
Bridges	66.09	100	62.82	20.66	68.20	88.20	45.90	18.94
Glass	71.42	100	62.16	25.91	69.61	61.62	62.59	15.98
Iris	94.66	100	94.00	12.44	96.00	38.00	70.66	6.50
Liver	65.22	100	63.46	30.59	66.09	83.70	64.13	5.21
Wine	94.44	100	93.86	10.55	94.44	78.89	79.44	6.10
Zoo	93.33	100	95.56	18.77	95.55	26.17	88.88	14.12
Average	82.52	100	80.22	18.16	79.06	47.34	71.59	9.40

V. Conclusion

The present paper offers an exhaustive survey of Prototype Selection methods proposed in the literature. Basic and advanced properties, existing work, and related fields have been reviewed. Based on the main characteristics studied, we have proposed a taxonomy of Prototype Selection methods. Much advanced and future work is needed to be done in this field since many characteristics are still not completely studied. Also many researchers confuse Prototype Generation(PG) in Prototype Selection(PS). According to well known surveys, many advanced PS methods are going unnoticed.

References

[1] Garc_IAET AL:*"Prototype Selection for Nearest Neighbor Classification: Taxonomy and Empirical Study"*: IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 34, No. 3, March 2012.

[2] T.M. Cover and P.E. Hart, "*Nearest Neighbor Pattern Classification*," *IEEE Trans. Information Theory,* vol. 13, no. 1, pp. 21-27, Jan. 1967.

[3] "*PROTOTYPE SELECTION FOR NEAREST NEIGHBOR CLASSIFICATION: SURVEY OF METHODS:"* Salvador Garc´ıa, Joaqu´ın Derrac, Jos´e Ram´on Cano, and Francisco Herrera.

[4] K. Hattori and M. Takahashi, *"A new edited k-nearest neighbor rule in the pattern classification problem,"* Pattern Recognition, vol. 33,no. 3, pp. 521–528, 2000.

[5] Computación y Sistemas Vol. 13 No. 4, 2010, pp 449-462 ISSN 1405-5546: Arturo Olvera López.

[6] N. Jankowski and M. Grochowski, *"Comparison of Instances Selection Algorithms I. Algorithms Survey," Proc. Int'l Conf. Artificial Intelligence and Soft Computing,* pp. 598-603, 2004.

[7] C. Garcı´a-Osorio, A. de Haro-Garcı´a, and N. Garcı´a-Pedrajas, "Democratic Instance Selection: A Linear Complexity Instance Selection Algorithm Based on Classifier Ensemble Concepts,"*Artificial Intelligence,* vol. 174, nos. 5/6, pp. 410-441, 2010.

[8] C.-L. Chang, *"Finding Prototypes for Nearest Neighbor Classifiers," IEEE Trans. Computers,* vol. 23, no. 11, pp. 1179-1184, Nov.1974.