The Impact of Feature Selection on the Accuracy of Naïve Bayes Classifier

Jasmina Novakovic

***Abstract* – In this paper is presented the impact of feature selection on the accuracy of naïve Bayes classifier. Six feature selection techniques have been used for feature selection, evaluated and compared using supervised learning algorithm naïve Bayes on eight real and three artificial benchmark data. Accuracy of the classifier is influenced by the choice of feature selection techniques. In our experiment, One-R improves naïve Bayes the most.**

Keywords: **classification accuracy, classification algorithm, feature selection, naïve Bayes.**

I. INTRODUCTION

Feature selection reduces the dimensionality of feature space, removes redundant, irrelevant, or noisy data. It brings the immediate effects for application: speeding up a data mining algorithm, improving the data quality and thereof the performance of data mining, and increasing the comprehensibility of the mining results.

Feature selection can be defined as a process that chooses a minimum subset of M features from the original set of N features, so that the feature space is optimally reduced according to a certain evaluation criterion. As the dimensionality of a domain expands, the number of feature N increases. Finding the best feature subset is usually intractable [1] and many problem related to feature selection have been shown to be NP-hard [2].

Researchers have studied various aspects of feature selection. Feature selection algorithms may be divided into filters [3], [4], wrappers [1] and embedded approaches [5]. Filters methods evaluate quality of selected features, independently from the classification algorithm, while wrapper methods require application of a classifier (which should be trained on a given feature subset) to evaluate this quality. Embedded methods perform feature selection during learning of optimal parameters (for example, neural network weights between the input and the hidden layer).

Jasmina Novakovic, Faculty of Public Administration, Megatrend University, Bulevar umetnosti 29, 11070 Novi Beograd, Srbija (telefon: 381-11-2092111, e-mail: jnovakovic@megatrend.edu,rs)

Some classification algorithms have inherited ability to focus on relevant features and ignore irrelevant ones. Decision trees are primary example of a class of such algorithms [6], [7], but also multi-layer perceptron (MLP) neural networks with strong regularization of the input layer may exclude the irrelevant features in an automatic way [8]. Such methods may also benefit from independent feature selection. On the other hand, some algorithms have no provisions for feature selection. The k-nearest neighbor algorithm is one family of such methods that classify novel examples by retrieving the nearest training example, strongly relaying on feature selection methods to remove noisy features.

II. FEATURE RANKING AND SELECTION

Diverse feature ranking and feature selection techniques have been proposed in the machine learning literature. The purpose of these techniques is to discard irrelevant or redundant features from a given feature vector.

In this paper, we consider evaluation of the practical usefulness of the following ranking methods:

- Information Gain (IG) attribute evaluation,
- Gain Ratio (GR) attribute evaluation,
- Symmetrical Uncertainty (SU) attribute evaluation,
- Relief-F (RF) attribute evaluation,
- One-R (OR) attribute evaluation,
- Chi-Squared (CS) attribute evaluation.

Entropy is a commonly used in the information theory measure, which characterizes the purity of an arbitrary collection of examples. It is in the foundation of the IG, GR, and SU attribute ranking methods. The entropy measure is considered as a measure of system's unpredictability. The entropy of Y is

$$H(Y) = -\sum_{y \in Y} p(y) \log_2(p(y)) \qquad (1)$$

where $p(y)$ is the marginal probability density function for the random variable Y. If the observed values of Y in the training data set S are partitioned according to the values of a second feature X, and the entropy of Y with respect to the partitions induced by X is less than the entropy of Y prior to partitioning, then there is a relationship between features Y and X. Then the entropy of Y after observing X is:

$$H(Y/X) = -\sum_{x \in X} p(x) \sum_{y \in Y} p(y/x) \log_2(p(y/x)) \qquad (2)$$

where $p(y|x)$ is the conditional probability of y given x.

A. Information Gain

Given the entropy as a criterion of impurity in a training set *S*, we can define a measure reflecting additional information about *Y* provided by *X* that represents the amount by which the entropy of *Y* decreases. This measure is known as IG. It is given by

$$IG = H(Y) - H(Y/X) = H(X) - H(X/Y) \quad (3)$$

IG is a symmetrical measure (refer to equation (3)). The information gained about *Y* after observing *X* is equal to the information gained about *X* after observing *Y*. A weakness of the IG criterion is that it is biased in favor of features with more values even when they are not more informative.

B. Gain Ratio

The Gain Ratio is the non-symmetrical measure that is introduced to compensate for the bias of the IG. GR is given by

$$GR = \frac{IG}{H(X)} \quad (4)$$

As equation (4) presents, when the variable *Y* has to be predicted we normalize the IG by dividing by the entropy of *X* and vice-versa. Due to this normalization, the GR values fall always in the range [0, 1]. A value of GR = 1 indicates that the knowledge of *X* completely predicts *Y*, and GR = 0 means that there is no relation between *Y* and *X*. In opposite to IG, the GR favors variables with fewer values.

C. Symmetrical Uncertainty

The Symmetrical Uncertainty criterion compensates for the inherent bias of IG by dividing it by the sum of the entropies of X and Y. It is given by

$$SU = 2\frac{IG}{H(Y) + H(X)} \quad (5)$$

SU takes values, which are normalized to the range [0, 1] because of the correction factor 2. A value of SU = 1 means that the knowledge of one feature completely predicts and the other SU = 0 indicates that *X* and *Y* are uncorrelated. Similarly to GR, the SU is biased toward features with fewer values.

D. Chi-Squared

Chi-squared attribute evaluation evaluates the worth of a feature by computing the value of the chi-squared statistic with respect to the class. The initial hypothesis H_0 is the assumption that the two features are unrelated, and it is tested by chi-squared formula:

$$\chi^2 = \sum_{i=1}^{r}\sum_{j=1}^{c}\frac{(O_{ij} - E_{ij})^2}{E_{ij}} \quad (6)$$

where Oij is an observed frequency and Eij is an expected (theoretical) frequency, asserted by the null hypothesis. The greater the value of $\chi 2$, the greater the evidence against the hypothesis H_0.

E. One-R

This attribute evaluation evaluates features individually by using the OneR classifier. OneR classifier ranks features according to error rate (on the training set). It treats all numerically valued features as continuous and uses a straightforward method to divide the range of values into several disjoint intervals. It handles missing values by treating "missing" as a legitimate value.

This is one of the most primitive schemes. It produces simple rules based on one feature only. Although it is a minimal form of classifier, it can be useful for determining a baseline performance as a benchmark for other learning schemes.

F. Relief-F

Relief-F attribute evaluation evaluates the worth of a feature by repeatedly sampling an instance and considering the value of the given feature for the nearest instance of the same and different class. This attribute evaluation assigns a weight to each feature based on the ability of the feature to distinguish among the classes, and then selects those features whose weights exceed a user-defined threshold as relevant features. The weight computation is based on the probability of the nearest neighbors from two different classes having different values for a feature and the probability of two nearest neighbors of the same class having the same value of the feature. The higher the difference between these two probabilities, the more significant is the feature. Inherently, the measure is defined for a two-class problem, which can be extended to handle multiple classes, by splitting the problem into a series of two-class problems.

III. NAÏVE BAYES

A supervised learning algorithm is adopted here to build model is naïve Bayes. This section gives a brief overview of this algorithm.

This classifier is based on the elementary Bayes' theorem. It can achieve relatively good performance on classification tasks [9]. Naïve Bayes classifier greatly simplifies learning by assuming that features are independent given the class variable. In simple terms, a naïve 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. In spite of their naive design and apparently over-simplified assumptions, naïve Bayes classifiers have worked quite well in many complex real-world situations. An advantage of the naïve Bayes classifier is that it requires a small amount of training data to estimate the parameters (means and variances of the variables) necessary for classification. Because independent variables are assumed, only the variances of the variables for each class need to be determined and not the entire covariance matrix.

More formally, this classifier is defined by discriminant functions:

$$f_i(X) = \prod_{j=1}^{N} P(x_j|c_i)P(c_i) \quad (7)$$

where $X = (x_1, x_2, \ldots, x_N)$ denotes a feature vector and c_j, $j = 1, 2, \ldots, N$, denote possible class labels.

The training phase for learning a classifier consists in estimating conditional probabilities $P(x_j|c_i)$ and prior probabilities $P(c_i)$. Here, $P(c_i)$ are estimated by counting

the training examples that fall into class *ci* and then dividing the resulting count by the size of the training set. Similarly, conditional probabilities are estimated by simply observing the frequency distribution of feature *xj* within the training subset that is labeled as class *ci*. To classify a class-unknown test vector, the posterior probability of each class is calculated, given the feature values present in the test vector; and the test vector is assigned to the class that is of the highest probability.

Despite the fact that the far-reaching independence assumptions are often inaccurate, the naïve Bayes classifier has several properties that make it surprisingly useful in practice. In particular, the decoupling of the class conditional feature distributions means that each distribution can be independently estimated as a one dimensional distribution. This in turn helps to alleviate problems stemming from the curse of dimensionality, such as the need for data sets that scale exponentially with the number of features. Like all probabilistic classifiers under the maximum a posteriori decision rule, it arrives at the correct classification as long as the correct class is more probable than any other class; hence class probabilities do not have to be estimated very well. In other words, the overall classifier is robust enough to ignore serious deficiencies in its underlying naive probability model.

IV. Experiment and results

Eight natural domains and three artificial Monk's domains [10] were used for evaluating different feature selection techniques on naïve Bayes, taken from the UCI repository of machine learning databases [11]. These domains were chosen because of (a) their predominance in the literature, and (b) the prevalence of nominal features, thus reducing the need to discretize feature values.

The characteristics of these domains are summarised on Table 1.

Table 1: Domain characteristics [12].

Domain	Instances	Features	%Missing	Accuracy
mu	8124	22	1.3	51.8
vote	435	16	5.3	61.4
cr	690	15	0.6	55.5
ly	148	18	0.0	54.7
pt	339	17	3.7	24.8
bc	286	9	0.3	70.3
au	226	69	2.0	25.2
sb	683	35	9.5	13.5
M1	432	6	0.0	50.0
M2	432	6	0.0	67.1
M3	432	6	0.0	52.8

On Table 1 data sets above the horizontal line are natural domains, those below are artificial. The default accuracy is the accuracy of always predicting the majority class on the whole data set. The % Missing column shows what percentage of the data set's entries (number of features × number of instances) have missing values.

Table 2 shows classification accuracy for four classification algorithms with full data sets mentioned above. Accuracy for three data sets are best with naïve Bayes, eight with C4.5 and one data set for IB and RBF.

The purpose of the experiments described in this section is to empirically test the claim that feature selection can improve the accuracy of naïve Bayes classification algorithm. Classification accuracy was estimated using ten-fold crossvalidation on each data set.

Table 2 Classification accuracy with full data sets

Data set	*Naive*	*C4.5*	*IB*	*RBF*
mu	95.8	100	100	98.5
vote	90.1	96.3	92.4	94.5
ly	83.1	77.0	81.1	80.4
cr	77.7	86.1	81.2	79.1
pt	50.1	39.8	33.6	-
bc	71.7	75.5	65.7	71
au	73.5	77.9	75.2	-
sb	93.0	91.5	90.0	-
M1	75	96.5	72.5	44.9
M2	66.4	67.1	55.3	67.1
M3	97.2	100	79.9	50.9

Table 3 shows the reliability of naïve Bayes classifier with feature selections using different selection techniques. In experiment we use the threshold 0.1 for all selection techniques. We set this threshold by which attributes can be discarded for all feature selection techniques.

Table 3 Classification accuracy with full data sets and feature selections

Set	full	IG	GR	SU	CS	OR	RF
mu	95.8	97.1	96.3	96.1	95.8	95.8	-
vote	90.1	89.9	89.9	89.9	90.1	90.1	91.7
ly	83.1	80.4	77.7	77.7	83.1	83.1	77.0
cr	77.7	76.9	74.8	74.8	77.7	77.7	84.9
pt	50.1	49.8	46.9	36.6	50.1	50.1	42.5
bc	71.7	70.3	70.3	70.3	71.7	71.7	66.8
au	73.5	73.9	73.0	67.7	73.0	73.5	68.1
sb	93.0	92.8	92.8	92.8	93.0	93.0	90.5
M1	75.0	75.0	75.0	75.0	75.0	75.0	75.0
M2	66.4	67.1	67.1	67.1	66.4	66.4	67.1
M3	97.2	97.2	97.2	97.2	97.2	97.2	97.2

The experiments presented in this article show that feature selection techniques ability to select useful features does carry over from artificial to natural domains.

IG maintains or improves the accuracy of naïve Bayes for five data sets and degrades its accuracy for six. GR maintains or improves accuracy for four data sets and degrades for seven. SU maintains or improves accuracy for four data sets and degrades for seven. CS maintains or improves accuracy for ten data sets and degrades for only one. OR maintains or improves accuracy for all data sets. RF maintains or improves accuracy for five data sets and degrades for five. RF has difficulty to calculate classification accuracy for one data set - mu.

Feature selection improves accuracy of Naïve Bayes for five data sets, maintains or degrades accuracy for six data sets. The accuracy of naïve Bayes with feature selections improves on mushroom data set (from 95.8% to maximum 97.7%), vote (from 90.1% to maximum 91.7%), credit (from 77.7% to maximum 84.9%), audiology (from 73.5% to maximum 73.9%) and M2 (from 66.4% to maximum 67.1%). Maximum improvement is 7.2%.

V. CONCLUSIONS

Feature selection may filter features leading to reduce dimensionality of the feature space. This is especially effective for classification methods that do not have any inherent feature selections build in, such as the nearest neighbor methods or some types of neural networks. Six feature selection techniques have been used for feature selection, evaluated and compared using naïve Bayes classifier on eight real and three artificial benchmark data. Accuracy of the classifier is influenced by the choice of feature selection techniques.

For different data sets and different feature selection techniques accuracy, may significantly differ. Evaluation of selecting features is fast. In our experiment, the best results we get with OR. OR maintains or improves accuracy for all data sets. The only way to be sure that the highest accuracy is obtained in practical problems requires testing a given classifier on a number of feature subsets, obtained from different feature selection techniques. The number of tests needed to find the best feature subset is very small comparing to the cost of wrapper approach for larger number of features.

There are many questions and issues that remain to be addressed and that we intend to investigate in future work. Some improvements of the selecting methods presented here are possible. The algorithms and data sets will be selected according to precise criteria: classify algorithms and several data sets, either real or artificial, with nominal, binary and continuous features.

These conclusions and recommendations will be tested on larger data sets using various classification algorithms in the near future.

REFERENCE

[1] R. Kohavi, and G.H. John, "Wrappers for feature Subset Selection", *Artificial Intelligence*, vol. 97, 1997, 273-324.

[2] A.L. Blum, and R.L. Rivest, "Training a 3-node neural networks is NP-complete", *Neural Networks*, 5:117–127, 1992.

[3] H. Almuallim, and T.G. Dietterich, "Learning with many irrelevant features", In: *Proc. AAAI-91*, Anaheim, CA, 1991, 547-552.

[4] K. Kira, and L.A. Rendell, "The feature selection problem: tradional methods and a new algorithm", In: *Proc. AAAI-92*, San Jose, CA, 1992, 122-126.

[5] A.I. Blum, and P. Langley, "Selection of relevant features and examples in machine learning", *Artificial Intelligence*, vol 97, 1997, 245-271.

[6] L. Breiman, J.H. Friedman, R.H. Olshen, and C.J. Stone, *Classification and Regression Trees*, Wadsworth and Brooks, Monterey, CA, 1984.

[7] J.R. Quinlan, *C4.5: Programs for machine learning*, San Mateo, Morgan Kaufman, 1993.

[8] W. Duch, R. Adamczak, and K. Grabczewski, "A new methodology of extraction, optimization and application of crisp and fuzzy logical rules", *IEEE Transactions on Neural Networks*, vol. 12, 2001, 277-306.

[9] P. Domingos, and M. Pazzani, "Feature selection and transduction for prediction of molecular bioactivity for drug design", *Machine Learning*, 29:103-130, 1997.

[10] S. B. Thrun, J. Bala, E. Bloedorn, I. Bratko, B. Cestnik, J. Cheng, K. De Jonng, S. Dzeroski, S.E. Fahlman, D. Fisher, R. Hamann, K. Kaufman, S. Keller, I. Kononenko, J. Kreuziger, R.S. Michalski, T. Mitchell, P. Pachowicz, Y. Reich, H. Vafaie, W. Van de Welde, W. Wenzel, J. Wnek, and J. Zhang, "The MONK's problems: A performance comparison of different learning algorithms", Technical Report CMU-CS-91-197, Carnegie Mellon University, 1991.

[11] C.J. Merz, and P.M. Murphy *UCI Repository of machine learning databases*, http://www.ics.uci.edu/˜mlearn/MLRepository.html, 1998.

[12] M. Hall, *Correlation-based Feature Selection for Machine Learning*, thesis, The University of Waikato, NewZealand, 1999.