Application of Data Mining to Candidate Screening

Shrihari A. Hudli[1], Anand V. Hudli[2], Aditi A. Hudli[3]
[1],[3] Computer Science Department MS Ramaiah Institute of Technology, Bangalore
[2] ObjectOrb Technologies Bangalore,
[1] shrihari@hudli.com,[2]anand.hudli@objectorb.com,[3]aditi@hudli.com

Abstract— **Classification models are supervised learning methods used for predicting the value of a categorical target attribute. These models use a set of examples, called the training set, to learn to predict the target class of a future example whose class is unknown. The development of learning algorithms capable of learning from past experience is an important step in emulating inductive learning in humans. Classification finds application in many domains, of which selection of customers for a marketing campaign, fraud detection, diagnosis of diseases, image recognition, and spam e-mail filtering are just a few examples. In this paper, we present a comparative study of the application of the Naive Bayes and k-Nearest Neighbors (kNN) classification methods to the problem of screening candidates for a vacant position in an organization. The observable attributes of a candidate profile are first established. In the training phase, a training set of example profiles is used for learning the classification rules of the organization. In the test phase, the accuracy of the classification model is assessed by classifying example profiles not included in the training set, but for which the target class is already known. In the prediction phase, a profile whose target class is not known is classified. We present the results of the comparison of the Naive Bayes method with the kNN method. Keywords- Data Mining, Machine Learning, Classification Models, Naive Bayes Learning, k-Nearest Neighbor Algorithm, Candidate Screening**

I. INTRODUCTION

Classification models are supervised learning methods and are often used for predicting the value of a categorical target attribute [1]. For example, given a set of symptoms for a patient, a classifier predicts the disease that the patient is most likely suffering from. Starting from a set of past observations whose target class is known, classification models are used to generate a scheme by which the target class of future examples can be predicted.

Classification is an important topic in learning theory due to its theoretical implications and the large number of domains where it can be successfully applied. The development of algorithms capable of learning from past experience represents a fundamental step towards emulating the inductive capabilities of humans.

The opportunities presented by classification extends into many different application domains: the selection of target customers for a marketing campaign, fraud detection, image recognition, early diagnosis of diseases, text cataloguing, and spam email detection are just a few examples of real world problems that can be formulated in terms of the classification paradigm.

One such practical application of classification is the selection of candidates from a pool of resumes to be called for further testing and interviews. Typically, an organization that needs to recruit employees advertises the job openings in newspapers, radio, TV, Internet, and other media. In response to the advertisement, interested candidates send their resumes to the organization. The task of the Human Resources department is to screen the resumes based on set of criteria and shortlist the candidates to be called for testing and interviews.

The organization of the paper is as follows. In Section 2, we discuss the basics of two classification models, namely the Naive Bayes and k-Nearest Neighbors methods, which are two of the most popular data mining algorithms. In Section 3, we discuss the application of the two methods to the problem of screening candidates based on a set of attributes. In Section 4, we present the results of the comparison study. Section 5 contains conclusions and discussion of future work.

II. CLASSIFICATION PROBLEMS AND MODELS

In a classification problem, there is a dataset Ɗ consisting of m observations described in terms of n explanatory attributes and a target categorical attribute. The explanatory attributes are also called predictive variables. The target attribute is also called a class or label, and the observations are also termed examples or instances. The target variable for classification models takes a finite number of values. In particular, the case where the observations belong to one of only two classes is called a binary classification problem. The purpose of a classification model is to identify recurring relationships among observations which describe the examples belonging to the same class. These relationships are then converted into classification rules which can be used to predict the class for observations for which only the values of explanatory variables are known. The rules may be of different forms depending on the type of classification model used.

From a mathematical perspective, in a classification problem, m known examples are given, consisting of pairs of (xi,yi), i Є M, where xi is the vector of values taken by the

ISBN No. 978-1-4673-2048-1/12/$31.00©2012 IEEE

{v1,v2,…,vH} denotes the target class. Each component xij of the vector xi is treated as a realization of the random variable Xj representing an attribute aj in the dataset $\mathcal{D}$. In a binary classification problem, H may be denoted by H = {-1,1} without loss of generality.

Let $\mathcal{F}$ be a class of functions $f(\mathbf{x_i}) : \mathbb{R}^n \mapsto \mathcal{H}$ called the *hypotheses* that represent possible relationships between y_i and $\mathbf{x_i}$. A *classification problem* consists of defining an appropriate hypothesis space $\mathcal{F}$ and an algorithm A_F such that A_F identifies a function $f^* \in \mathcal{F}$ that can optimally describe the relationship between the predictive variables and the target class.

There are three components of a classification problem: a *generator* of observations, a *supervisor* of the examples according to an unknown probability distribution $P_X(\mathbf{x})$. For each vector x of examples, the supervisor returns the value of the target class according to a conditional distribution $P_{Y|X}(y|\mathbf{x})$ which is also unknown. A classification algorithm A_F, also called the *classifier* selects a function $f^* \in \mathcal{F}$ in the hypothesis space that minimizes a loss function.

The development of a classification model consists of three main phases [1].

Training phase: During the training phase, the classification algorithm is applied to the examples of the training set, which is a subset of the dataset $\mathcal{D}$. This subset consists of observations for which the target class is already known. This allows the classifier to derive classification rules that establish the correspondence between the target class y and each observation x.

Test phase: During the test phase, the rules generated in the training phase are used to classify observations of the dataset $\mathcal{D}$ that are not included in the training set and for which the target class is already known. The accuracy of the classification model is assessed by comparing the predicted class for each example in the test set with the actual class of the example. The training set and test set must be disjoint to avoid an overestimate of the model accuracy.

Prediction phase: In the prediction phase, the classifier is used to predict the class of an observation for which the target class is not known. This phase thus represents the use of the classification model to assign the target class to new observations in the future.

Figure 1 shows the logical flow of the learning process for a classification algorithm.

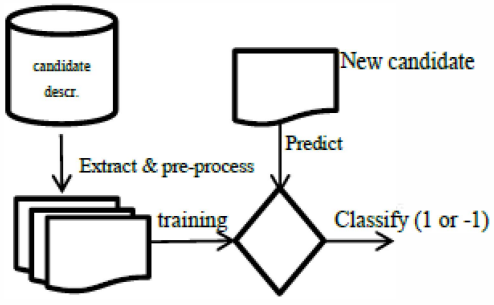

Figure 1 Various steps in predicting the target class

Classification models may be divided into four broad categories. Heuristic models make use of classification algorithms that are simple and intuitive. These include nearest neighbor methods and classification trees. Separation models divide the attribute space $\mathbb{R}n$ into H disjoint regions, {S1,S2,…,SH}, separating the observations based on the target class, so that observations in region SH are assigned to class yi = vH. In general, it is difficult to divide the observations exactly into a set of simple regions. Hence, a loss function is defined to take into account the points that are not classified correctly, and an optimization problem is solved to arrive at a subdivision into regions that minimizes the loss. Discriminant analysis, perceptron methods, neural networks, and support vector machines are some of the most popular separation methods. Regression models make an explicit assumption regarding the functional form of the conditional probabilities Py|x(y|x) which correspond to the assignment of the target class by the supervisor. Linear regression assumes a linear relationship exists between the dependent variable and the predictors. Logistic regression is an extension of linear regression to handle binary classification problems. In probabilistic models, a hypothesis is formulated regarding the functional form of the conditional probabilities Px|y(x|y) of the observations, given the target class, known as class-conditional probabilities. Next, based on the estimate of the prior probabilities Py(y) and Bayes' Theorem, the posterior probabilities of the target class Py|x(y|x) can be calculated. Naive Bayes and Bayesian Networks are popular families of probabilistic methods.

III. EVALUATION OF CLASSIFICATION MODELS

Alternative classification models for a classification problem can be evaluated using several criteria. In this paper, we use a method of confusion matrices to evaluate the two classification models, Naive Bayes and k-Nearest Neighbors, which are used in problem of screening candidates, as mentioned in Section 1. We assume that the values taken by the target class are {-1,1}. We can consider a 2 X 2 matrix whose rows correspond to the observed values and whose columns denote the values predicted by the classification model, as shown in Table 1 below.

		Predicted	
		-1	+1
Observed	-1	p (tn)	q (fp)
	+1	u (fn)	v (tp)

The accuracy of a classifier is given by:

$$Accuracy = \frac{p+v}{p+q+u+v} = \frac{tn+tp}{tn+fp+fn+tp}$$

The true positives rate, also known as *recall*, is defined as:

$$Recall = tp\ rate = \frac{v}{u+v} = \frac{tp}{fn+tp}$$

The false positives rate is defined as:

$$fp\ rate = \frac{q}{p+q} = \frac{fp}{tn+fp}$$

The precision is defined as:

$$Precision = \frac{v}{q+v} = \frac{tp}{fp+\ tp}$$

The F-measure is defined as:

$$F - measure = 2 * \frac{Precision * Recall}{Precision + Recall}$$

IV. TWO CLASSIFICATION MODELS

A. *Naive Bayes Classification Model*

The Bayesian model calculates the posterior probability of a specific target class Py|x(y|x), given an observation x, by means of Bayes' Theorem, using the prior probability of class y, P(y) and the *conditional probabilities* P(**x**|*y*), which are computed in the training phase. Consider an observation **x** whose class variable y may take *H* distinct values, $\{v_1, v_2, \ldots, v_H\}$. We can use Bayes' Theorem to calculate the posterior probability P(*y*|**x**), the probability that the observation **x** belongs to class *y*:

$$P(y|\mathbf{x}) = P(\mathbf{x}|y)P(y) / \sum_{i=1}^{H} P(\mathbf{x}|y)P(y) = P(\mathbf{x}|y)P(y) / P(\mathbf{x})$$

To classify an observation **x**, the Bayes' classifier applies the principle of *maximum a posteriori hypothesis* (MAP), which involves calculating the posterior probability P(*y*|**x**) for all classes *y* and assigning the observation **x** to the class which has the maximum value P(*y*|**x**). The prior probabilities P(*y*) can be estimated using the frequencies m_h with which each class appears in the dataset. ( ) m_h/m.

The sample estimate of the conditional probabilities P(**x**|*y*) cannot be obtained in practice due to the computational complexity and the huge number of sample observations that it would require. To overcome this difficulty, we use the *Naive Bayes classifier* which we describe below.

Naive Bayes classifiers are based on the assumption that the explanatory variables in the observation **x** are all conditionally independent for a given target class. This assumption allows us to express P(**x**|y) as:

$$P(\mathbf{x}|y) = P(x_1|y) * P(x_2|y) * \ldots * P(x_n|y) = \prod_{j=1}^{n} P(x_j|y)$$

The probabilities P(x_j|*y*) can be estimated using the examples from the training set. P(x_j=v|*y*) is calculated as the ratio of the number of instances of class *y* for which the attribute x_j takes the value v to the total number of instances of the class *y* in the dataset. Empirical comparisons showing the effectiveness of the Naive Bayes method are found in [2] and a comparative assessment is found in [3].

B. *k- Nearest Neighbors Classification Model*

The k-Nearest Neighbors classification model or the kNN model finds a group of k observations in the training set that are closest to the test example, and bases the assignment of the target class on the predominance of a particular class in this neighborhood. The kNN method is an example of a lazy learning technique. It waits until the query arrives to generalize beyond the training data. Although it is easy to understand and implement, it performs well in many situations. A well-known result by Cover and Hart [4] shows that the classification error of the nearest neighbor rule is bounded above by twice the optimal Bayes error under certain reasonable assumptions. The error of the general kNN method asymptotically approaches the Bayes error and can be used to approximate it.

Given a training dataset *D* and a test observation **x**, whose target class is unknown, the kNN algorithm computes the distance between **x** and all training examples to determine the list of its k nearest neighbors. It then assigns a class to **x** by taking the majority of the nearest neighbors. The value of k must be chosen carefully. If k is too small, the result can be sensitive to noise points. If k is too large, then the neighborhood may include too many points from other classes. The choice of the distance measure is another important consideration. Consider two observations **x** and **y** with n attributes. The Euclidean distance between the two points is given by:

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

The Manhattan distance is given by:

$$d(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n} |x_i - y_i|$$

V. APPLICATION TO CANDIDATE SCREENING

Candidate screening is an important step in many applications. For example, the human resources department of an organization may screen candidate resumes to arrive at a list of candidates who are to be called for a written test and interview. Companies often receive thousands of resumes for each job posting and dedicated screeners shortlist qualified candidates. Decision support tools can help in improving the efficiency of the shortlisting process [5]. Another application is the selection of customers from a list who are most likely to buy a certain product. In such cases, a candidate would have a set of attributes, such as qualifications, experience, proficiency in required skills, etc. The training set would consist of candidates with their attributes and the decision on whether each candidate is selected for the next round, which is the target class. The classifier learns the classification rules that establish the correspondence between the target class and each candidate. In the test phase, candidates for whom the class is known and who were not included in the training set are classified as selected or not by the classifier. This allows us to assess the accuracy of the classifier. In the prediction phase, the

classifier is used to classify one or more candidates whose target class is not known.

We now describe the application of the classification methods described above specifically to the problem of shortlisting of candidates from a pool of resumes. We have used seven attributes to describe a candidate. The first attribute is qualifications and it can take values 1 (low), 2 (medium), and 3 (high). The second attribute is experience and it can also take 3 values. The next five attributes are related to different skill sets that the candidate may possess. For example, Programming Skill may indicate the degree to which a candidate is skilled in the art of programming. Designing software using well known software design methodologies such as Object Oriented Design may comprise another skill set. Each skill attribute can take values from 1 to 5, where 1 indicates little or no proficiency in the skill set and 5 indicates the highest possible skill level. The target class value will be either a -1, which indicates the candidate is not selected, or a 1, which indicates the candidate is selected.

VI. RESULTS

Receiver Operating Characteristic (ROC) [6] curve charts allow the user to evaluate the accuracy of a classifier. An ROC chart is a plot with the false positive rate on the horizontal axis and the true positive rate on the vertical axis. The point (0,1) represents the ideal classifier which makes no error, since the false positive rate is 0 and the true positive is 1. The point (0,0) is the classifier that predicts the class -1 for all observations, while the point (1,1) is a classifier that predicts class 1 for all observations in the dataset, regardless of the values of the attributes.

Figure 2 shows the ROC curves for the Naive Bayes and kNN classifiers applied to the problem of candidate screening described in Section 3. The area under the ROC curve gives an indication of the accuracy of the classifier. The classifier which corresponds to the greatest area is preferable. In the case of candidate screening, it is seen that the Naive Bayes classifier, which is associated with the greater of the two areas, is preferable.

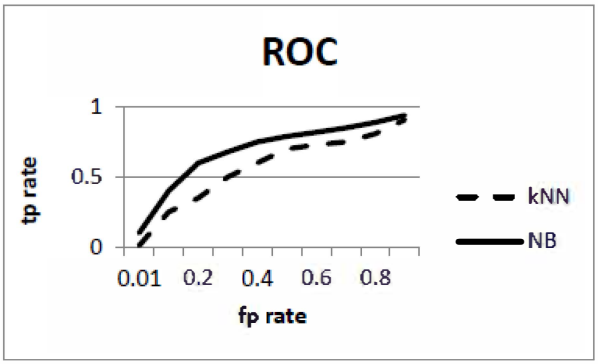

Figure 2 ROC curves for Naive Bayes and kNN classifiers

VII. CONCLUSION

We have presented two classification models for screening candidates, the Naive Bayes and k-Nearest Neighbors classifiers. It is found that the Naive Bayes classifier performs better than the kNN classifier. The use of such classifiers helps improve the efficiency of the screening process. Further research can be done on comparing other classifiers, such as Support Vector Machines (SVMs) and Neural Networks. The work presented here can also be extended to rank candidates and to assign candidates to the most suitable job opening in an organization.

REFERENCES

[1] C. Vercellis , Business Intelligence: Data Mining and Optimization For Decision Making, John Wiley & Sons Ltd., Chichester, UK, 2009.

[2] P. Domingos and M. Pazzani, "On the optimality of the simple Bayesian classifier under zero-one loss", *Machine Learning*, 29, pp. 103–130, 1997.

[3] A. Jamain and D. J. Hand, "Mining supervised classification performance studies: A meta-analytic investigation", *Journal of Classification*, 25, pp. 87–112, 2008.

[4] T. Cover and P. Hart , "Nearest neighbor pattern classification", *IEEE Transactions on Information Theory,* 13(1): pp. 21–27, 1967.

[5] A. Singh, et. al., "PROSPECT: A system for screening candidates for recruitment", in CIKM Proceedings of the *19th ACM Conference on Information and Knowledge Management*, New York, USA, 2010.

[6] A. P. Bradley , "The use of the area under the ROC curve in the evaluation of machines learning algorithm" *Pattern Recognition,* Vol.30, Issue 7, pp.1145-1159,1997