CLASSIFICATION USING NAÏVE BAYES- A SURVEY

Neha Sharma
Department of Computer Science, Gurukul Institute of Technology, Kota, Rajasthan
'neha549sharma@gmail.com

Abstract— Classification, particularly Text Classification, is a supervised learning approach categorizing into various categories, the available training set of correctly identified observations analyzed into a set of features. There are many phases involved in classification. The main classification phase involves the use of classification algorithms or classifiers. Among the various classifiers, the Naïve Bayes Classifier belongs to a family of probabilistic classifiers based on the Bayes theorem and the independence assumption criteria and is of great use in the fields of document classification and disease prediction. This paper discusses the classification process using the Naïve Bayes Classifier and surveys the recent developments in the same particularly in domains of health care and semantic analysis.

Keywords— Classification, Text Classification, Bayesian Classifier, Naïve Bayes, Semantic Analysis, Health Care

I. INTRODUCTION

This In machine learning, classification can be taken as an instance of supervised learning, a problem to identify that an observation belongs to which set of categories based on the training dataset containing instances or observations with a known category membership. Likewise, Text Classification aims to classify the given text in a number of categories. The process of text classification involves five main phases; documentation, preprocessing, indexing, feature selection, classification algorithms, performance measure. The Documentation phase collects the documents available in different formats. The preprocessing step converts these documents into a clear word format and breaks it into tokens with tokens further conflated to their root form through the process of tokenization and stemming respectively. The indexing phase is also a kind of preprocessing step aiming at easy handling of data by converting the full text version to a document vector. The most important of all phases is the feature selection phase involving selection of relevant features/words in the whole document based on the importance of those words in the underlying classification task. This is done though various effective feature selection methods. The document is then ready for the classification phase where the classification algorithms, specifically called “Classifiers” are used. The last step measures the performance of the classification task based on the classifier and the feature selection method used. There have been proposed many classifiers namely Naïve Bayes[1], Rochhio’s algorithm[2], Support Vector Machines[3], K-Nearest Neighbor[4], Maximum Entropy classifier[5] etc.

The Naïve Bayes classifiers are based on the Bayes Theorem [6] and are therefore sometimes referred to as Bayesian classifiers. Bayesian classifiers work on the idea that a class can possibly predict the features of the members of that class because of the commonality of the feature values in the class. If a class is known to an agent, prediction of feature values is easy. However, if the class itself is not known and the only source of knowledge are some of the values of the features, then the Bayes rule can be applied to predict the corresponding class. A learning agent can therefore build a probabilistic model of the available features and use the same for predicting the classification. Classification using Bayesian Classifiers can also be summarized in terms of latent variables, i.e. the probabilistic variables that are not observed. In such a case, classification is a latent variable, probabilistically related to the observed variables, thereby making the resultant classification an inference in the probabilistic model. The simplest of all the related classifiers proposed is the Naïve Bayes Classifier.

Naïve Bayes classifiers work on an independent assumption that the input features participating in the classification are totally independent. This independence assumption can be best explained through a network consisting of the features as the nodes, the target variable with no parents and the
classification as the only parent of each input feature. As long as the underlying assumption of independence is true, the Naïve Bayes classifier works very well. The independence here refers to the assumption that the underlying class should be a good predictor of the features, the features that are independent given the class. The other advantages of the classifier include simplicity and easy implementation which make it a promising method to be tried for a new classification problem. However, the underlying assumption is not always practical as per the real world scenario. The Naïve Bayes is a linear classifier and the assumption is also not practical always. For the cases where the classification problem presented is non-linear and strongly violates the independence assumption, the performance of the Naïve Bayes is a question.

II. BACKGROUND

A. Document Models

The essential criterion for the success of any classifier is proper emphasis on the type of data and the type problem. Based on the distributions according to which the values of each class are distributed, there are three variants of the Naïve Bayes Classifier; the Gaussian Naïve Bayes, Multinomial Naïve Bayes and Bernoulli Naïve Bayes. The assumptions on the distribution of features are termed as the event models or the document models of the Naïve Bayes Classifier. In a document model, the document is represented as bag of words where the bag further represents set allowing repeating words. Each document is represented as a matrix with the feature vectors corresponding to the word types. This frequency matrix tells the number of times a particular term has appeared in the document The dimension of the feature vector is |V| for |V| word types in a vocabulary V. Out of the three document models, Bernoulli and the multinomial document models work best when the features in the document to be classified are discrete, that is not binary. Assumption of a distribution and therefore a model is necessary for the estimation of parameters for the distribution of features. The three document models are discussed in brief below.

For the continuous values, the distribution is assumed to follow the Gaussian distribution. The **Gaussian Naïve Bayes model** can be understood through the following example. Let the training set consist of a continuous attribute x, then the steps would be segmenting the data by class followed by computation of mean and variance of x for each class. The probability distribution of some value given the class as input can then be computed as

\[
p(x = v|c) = \frac{1}{\sqrt{2\pi \sigma_c^2}} e^{-\frac{(v-\mu_c)^2}{2\sigma_c^2}}
\]

Where,

- \(\mu_c\) is the mean of the values in x associated with class c
- \(\sigma_c^2\) is the variance of the values in x associated with class c
- \(p(x = v|c)\) is the probability distribution with v plunged into the equation of normal distribution parameterized by the mean and the variance.

Another way of handling continuous data is discretizing them through the process of binning to convert them into Bernoulli distributed features. To apply Naïve Bayes for classification, discrete values sometimes become a necessary criterion though the process of discretization itself throws away the discriminative information.

A binary vector representing a point in the space of words is used to represent the **Bernoulli Naïve Bayes Model**. The inputs are described through the features which are binary variables or the independent Booleans. The model is widely used for the purpose of document classification using binary term occurrence features instead of the term frequencies. The advantages of the Bernoulli document model include its efficiency in classifying short texts and giving the model information regarding the absence of terms unlike the other similar models. For a Boolean \(x_i\) representing whether the \(i^{th}\) term has occurred in the vocabulary or not, the likelihood of the corresponding document in a given class C is given using the formula

\[
P(x|C) = \prod_{i=1}^{n} p_i^{x_i} (1 - p_i)^{1-x_i}
\]

Where,
\( p_i \) is the probability that the class C will generate the word on the \( i^{th} \) dimension.

The multinomial Naïve Bayes model represents the feature vectors as frequencies of the generation of event through a multinomial \( (p_1, ..., p_n) \) with the probability \( p_i \) showing that the event \( i \) has occurred. For counting the number of times an event \( i \) has occurred in a document at a particular instance, a feature vector, \( x_i = \{x_1, ..., x_n\} \) is used which is now a histogram. The likelihood of observing this histogram in a class C can be calculated as

\[
P(x|C) = \frac{(\sum_i x_i)!}{\prod_i x_i!} p_i^{x_i}
\]

B. Working Methodology of the Naïve Bayes Classifier

Classification with Naïve Bayes initiates with taking the text documents as word counts. For measuring the relative degree of association between the class-word pairs, the classifier makes a log linear decision rule that assigns an independent parameter to each class-word pair. The two steps of the classifier include

- Calculation of class conditional probability
- Calculation of classification or posterior probability

For every term \( t_i \) and class \( c_j \), the class conditional probability \( \hat{P}(t_i|c_j) \) considering only one training set is given as follows:

\[
\hat{P}(t_i|c_j) = \frac{1 + \text{number of times } t_i \text{ appears in a document from class } c_j}{d + \text{number of words in all documents from class } c_j}
\]

\[
\hat{P}(t_i|c_j) = \frac{\sum tf(t_i, d \in c_j) + \alpha}{\sum N_{d \in c_j} + \alpha \cdot M}
\]

Where,

\[
\sum tf(t_i, d \in c_j): \text{ The sum of raw term frequencies of word } t_i \text{ from all documents in the training sample belonging to class } c_j
\]

\( \alpha \): An additive smoothing parameter

\[
\sum N_{d \in c_j}: \text{ The sum of all term frequencies in the training dataset for class } c_j
\]

\( M \): The number of features / terms.

Once the conditional probability is calculated for each term and class, the trained classifier is able to predict the class of any upcoming new document.

Let the document to be queried query is \( d \), with feature vectors represented by term frequencies. The posterior probability of a document belonging to any class \( c_j \) is the product of individual class-conditional probabilities of all terms contained in the query document.

\[
P(d|c_j) = \hat{P}(t_1|c_j) \hat{P}(t_2|c_j) \ldots \hat{P}(t_M|c_j)
\]

\[
= \prod_{i=1}^M \hat{P}(t_i|c_j)^{tf(t_i,d)}
\]

After the calculation of both the probabilities, the maximum probability towards a class \( c_k \) that indicates that query document \( d \) belongs to class \( c_k \) is given by

\[
k = \arg \max_j P(d|c_j)
\]

III. Naïve Bayes and Health Care Data

Borkar and Deshmukh [7] proposed using Naïve Bayes classifier for detection of Swine Flu disease. The process starts with finding probability for each attribute of Swine flu against all output. The probabilities of each attribute are then multiplied. Selecting the maximum probability from all the probabilities, the attributes belong to the class variable with maximum value. The promising results of the proposed scheme can be used for investigating further the Swine flu disease in patients using Information technology.

Patil [8] worked in the direction of diagnosing whether a patient with his given information regarding age, sex, blood pressure, blood sugar, chest pain, ECG reports etc can have a heart disease later in life or not. The experiments involve taking the parameters of the medical tests as inputs. The proposal is effective enough in being used by nurses and medical students for training purposes. The data mining technique used is Naïve Bayes Classification for the development of Decision Support System in Heart Disease Prediction System (HDPS). The performance of the proposal is further improved using a smoothing operation. The implementation of HDPS is done through a MATLAB application able to detect and extract hidden knowledge related to heart diseases from a historical heart disease database.
Kharya et al [9] proposed detecting in patients the chances of having Breast Cancer later in life. Severity in Breast Cancer is necessary seeing it becoming the second most cause of death among women. A Graphical User Interface (GUI) is designed for entering the patient’s record for the prediction. The records are mined through the data repository. Naïve Bayes classifier, being simple and efficient is chosen for the prediction. The results obtained by the Naïve Bayes classifier are accurate, have low computational effort and fast. Implementation of the proposal is done through Java and the training of data is done using datasets from UCI Machine Repository [10]. Another advantage of the proposed system is that the system expands according to the dataset used.

Stephanie J. Hickey [11] proposed using Naïve Bayes Classifier for public health domain combined with greedy feature selection. The input was a public health dataset and the objective behind the proposal was to identify one or several attributes that best predict a selected target attribute without the need for searching the input space exhaustively. The proposal achieved its goal with increase in accuracy of classification. The target attributes were related to diagnosis or procedure codes.

Ambica et al [12] proposed using Naïve Bayes for an efficient decision support system for Diabetes disease. The proposed classification system was divided into two steps. The first step includes analysis of how optimal the dataset is and accordingly extraction of the optimal feature set from the training data is done. The second step forms the new dataset as the optimal training dataset and the proposed classification scheme is now applied on the optimal feature set. The mismatched and unavailable features from the training and testing datasets are ignored and the dataset attributes are used for the calculation of posterior probability. The proposed procedure therefore shows elimination of unavailable features and document wise filtering.

**IV. NAÏVE BAYES AND SEMANTIC ANALYSIS**

Gamollo et al [13] proposed sentiment analysis on Spanish tweets. The proposal was more inclined towards detecting sentiments on Twitter, a very popular microblogging service consisting of millions of tweets throughout the day from celebrities and the media and people following these celebrities. The notion of considering polarity for the classification is used. A total of six sentiment categories are detected in the proposed system with a resulting accuracy of 67%. Apart from the positive and negative tweets, polarity levels have been extended to strong, average, weak or neutral tweets by setting thresholds for experimental purposes. For detecting whether the analyzed text has some polarity or not, searching of polarity words is done within the same.

Gamallo and Garcia [14] implemented two Naïve Bayes classifiers-Baseline and Binary for the task of Sentiment Analysis on English tweets. English tweets are considered difficult to classify because of them based on the uncertain human subjectivity and being too small to be linguistically analyzed. The SemEval-2014 classifies a given message into positive, negative or neutral sentiments. For the message containing both the sentiments, majority of the two sentiments is considered. The work of the proposed Baseline Naïve Bayes Classifier is to classify from the training corpus, the positive, negative and neutral categories thereby introducing no modifications in the classifier. The Binary classifier uses the polarity lexicon and further simplifies the corpus, categorizing the tweets as negative or positive. In other words, a Boolean classifier was trained to distinguish between the polarity-constrained tweets. For the neutral tweets, if at least one word was found in the polarity lexicon, then the tweet was considered to be of some degree of that polarity. If no such word was found, then the tweet was considered neutral. The binary classifier used gave more than 80% precision in the trained corpus with the positive and negative categories. Their proposed scheme is being used by a company specializing in Natural Language Technology, Cilenis S. L. for four languages; English, Spanish, Portuguese and Galician.

Another similar classification was done on SemEval-2015, Task 10, Subtask B by Talbot et al [15]. The authors designed a sentiment classification system employing a supervised text classification approach based on constraints. The
two major contributions include enhancement of quality related to lexical information by introduction of various preprocessing steps for tweeting data and use of Naïve Bayes Classifier for the detection of tweet sentiments with the training data provided by the task organizers. The positive and negative words in the external human-generated lists are often used for the classification. A F-Score of 59.26 is observed by their proposal on an official test dataset.

V. NAÏVE BAYES AND OTHER DOMAINS

Kim et al [16] modified the Naïve Bayes classifier for an e-catalogue classification of online business transactions. The increasing product information and its heterogeneous nature have correspondingly increased the complexity in the classification task. The authors addressed this problem by extension of the Naïve Bayes classifier for utilizing the structural characteristics of the e-catalogs. Use of such appropriate characteristics for the purpose of classification improves the accuracy of classification. Instead of taking each word position as an attribute as done in the conventional Naïve Bayes classifier for text classification, sets of attribute value pairs are considered for classification as per the underlying objective. Every individual attribute is assigned a weight according to the importance they hold in the classification. Before the weights are assigned, normalization of the attributes is done. The accuracy is further improved by the use of ‘category name’ attribute that is available in the training set. The proposed classification is observed more accurate even in the real world scenario consisting of noisy attributes.

Salperwyck et al [17] used Naïve Bayes for data streams. The already proposed methods of improving the Naïve Bayes by the use of a variable selection method or by weighting of the explanatory variables are used for off-line learning. Plus they needed to store all the data available in memory or requiring reading each example more than once. These limitations were addressed by the authors’ method based on a graphical model where scholastic estimation was used for the computation of weights to be imposed on the input variables. The proposed method was incremental and the results when compared with the conventional Naïve Bayes classifier were better.

Rabenoro et al [18] focused on the application papers with expert knowledge, both simple and with low level parameter scores and build an interpretable classifier based on this knowledge and a training set. However, the estimation of scores requires other parameters and thresholds which are seldom available through the experts. Combination of these scores for achieving rates acceptable for classification is another limitation. The authors propose selecting features using the filter mRMR approach with the idea to keep only the useful features for consideration from the large set of redundant binary features obtained from the data under study. The advantages from the proposed solution include easy finding of required parameters and thresholds with the reasonable features. The decision by the interpretable Naïve Bayes classifier is also reduced.

Jing et al [19] aimed at improving the conventional Naïve Bayes classifier and proposed a Semantic Naïve Bayes classifier (SNBC) through the incorporation of document level semantic information. The semantic information is captured from each document by the proposed semantic feature extraction and modeling algorithms. The semantic feature extraction is further done in two steps. In the first step, each word in the document is transformed into a semantic vector through Log-Bilinear document modeling (LBDM). In the second step, the formed word vectors’ space can extract the semantic feature sets for each document through the use of Principal Component Analysis (PCA). The semantic features of the training document are then used for the semantic modeling. This semantic model is then combined with the conventional Naïve Bayes classifier in the testing phase of the proposed SNBC for document classification. The proposed SNBC shows outperformed results when compared to the original Naïve Bayes Classifier.

VI. CONCLUSION

Classification is the supervised form of data learning. For the purpose of classifying data, particularly texts, classification algorithms, called
classifiers are used in the final phase after the phases of documentation, pre-processing, indexing, and feature selection. Among the several classifiers proposed till date, Naïve Bayes is the most simplistic probabilistic classifier for the purpose of text classification. It is based on the assumption that given the context of the class, all attributes of the text are independent to each other. Its simplicity and easy implementation makes it very popular. However, it does lack in the aspect of accuracy. Therefore, all the research works using Naïve Bayes aim to improve its accuracy. This survey gives details about the Naïve Bayes Classifier, its models, work methodology and the recent developments using Naïve Bayes in different domains for classification.

**REFERENCES**


