2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Dual Instance and Attribute Weighting for Naive Bayes Classification

Jia Wu, Shirui Pan, Zhihua Cai, Xingquan Zhu, and Chengqi Zhang

***Abstract*— Naive Bayes (NB) network is a popular classification technique for data mining and machine learning. Many methods exist to improve the performance of NB by overcoming its primary weakness—the assumption that attributes are conditionally independent given the class, using techniques such as backwards sequential elimination and lazy elimination. Some weighting technologies, including attribute weighting and instance weighting, have also been proposed to improve the accuracy of NB. In this paper, we propose a dual weighted model, namely DWNB, for NB classification. In DWNB, we firstly employ an instance similarity based method to weight each training instance. After that, we build an attribute weighted model based on the new training data, where the calculation of the probability value is based on the embedded instance weights. The dual instance and attribute weighting allows DWNB to tackle the conditional independence assumption for accurate classification. Experiments and comparisons on 36 benchmark data sets demonstrate that DWNB outperforms existing weighted NB algorithms.**

I. INTRODUCTION

BAYESIAN network (BN) is a popular machine learning algorithm used to predict the label of an instance by using generative models [1]. In a BN learning task, a Bayesian network classifier (BNC) from a given set of labeled training instances represented by a set of attribute variables is constructed to predict the distribution of a class variable. The Bayesian approach for classification is to assign the most probable class label to a test instance.

Given a training set $\mathcal{D} = \{\mathbf{x}_1, \cdots, \mathbf{x}_N\}$ with N instances, each of which containing n attribute values and a class label. We use $\mathbf{x}_i = \{x_{i,1}, \cdots x_{i,j}, \cdots x_{i,n}, y_i\}$ to denote the ith instance in the data set $\mathcal{D}$, with $x_{i,j}$ denoting the jth attribute value and y_i denoting the class label of the instance. The class space $\mathcal{Y} = \{c_1, \cdots, c_k, \cdots, c_L\}$ denotes the set of labels that each instance belongs to and c_k denotes the kth label of the class space. For ease of understanding, we use (x_i, y_i) as a shorthand to represent an instance and its class label, and use x_i as a shorthand of $\mathbf{x}_i$. We also use A_j to represent the jth attribute. Each attribute can be a discrete random variable (with a number of discrete values) or a continuous random variable. In this paper, we only focus on categorical (or nominal) attributes. For each attribute A_j, we use $a_j^\tau, \tau = 1, \cdots, |A_j|$ to denote the τth attribute value of A_j and $|A_j|$ denotes the total number of distinct values of A_j. For each instance $\mathbf{x}_i$, its value satisfies $x_{i,j} \in A_j$. For an instance (x_i, y_i) in the training set $\mathcal{D}$, its class label satisfies $y_i \in \mathcal{Y}$, whereas a test instance x_t only contains attribute values and its class label y_t needs to be predicted by the BNC model. For BNC classification, a set of training instances with class label are given in advance to train a BNC classifier, which can be mathematically formulated as:

$$c(x_t) = \underset{c_k \in \mathcal{Y}}{\arg\max}\, P(c_k) P(x_{t,1}, x_{t,2}, \cdots, x_{t,n} \,|c_k) \tag{1}$$

In Eq. (1), $P(c_k)$ denotes the probability of class c_k in the whole training set. $P(x_{t,1}, x_{t,2}, \cdots, x_{t,n} \,|c_k)$ denotes the joint distribution of x_t conditioned by the given class c_k. It is easy to estimate $P(c_k)$ but calculating $P(x_{t,1}, x_{t,2}, \cdots, x_{t,n} \,|c_k)$ is difficult. Unless the number of instances in training data is very large, we cannot obtain reliable estimations. In fact, learning an optimal BNC has been proved to be NP-hard [2]. Alternatively, assume that all attributes are independent, given the class variable, the probabilities of observing the conjunction is just the product of the provability of individual attributes. This is the core concept of Naive Bayes (simply NB [3]), a simplified yet highly practical Bayesian network classifier, as defined in Eq. (2).

$$c(x_t) = \underset{c_k \in \mathcal{Y}}{\arg\max}\, P(c_k) \prod_{j=1}^{n} P(x_{t,j} \,|c_k) \tag{2}$$

where $P(x_{t,j} \,|c_k)$ denotes the joint distribution of $x_{t,j}$ conditioned by the given class c_k.

The NB classifier is based on the so-called Bayesian theorem and is particularly suitable for high dimensional data [4]. This is mainly because that each attribute node in NB has the class node as its only parent and it does not involve any other attribute nodes. Training NB is also easy and highly efficient because it only needs to compile a table of class probability estimates and a table of conditional attribute-value probability estimates from the training examples [5]. As a result, despite of its simplicity, NB often demonstrates very efficient runtime performance and frequently outperforms other sophisticated classification models.

In practice, the attribute independence assumption of NB does impact on its classification performance when it is violated in a learning task. Researchers have proposed many methods to further improve the performance of NB [6],

Jia Wu is with the Centre for Quantum Computation & Intelligent Systems, FEIT, University of Technology Sydney, Australia, and is affiliated with the Department of Computer Science, China University of Geosciences Wuhan, China (jia.wu@student.uts.edu.au).

Zhihua Cai is with the Department of Computer Science, China University of Geosciences Wuhan, China (zhcai@cug.edu.cn).

Xingquan Zhu is with Dept. of Computer and Electrical Engineering & Computer Science, Florida Atlantic University, USA (xzhu3@fau.edu).

Shirui Pan and Chengqi Zhang are with the Centre for Quantum Computation & Intelligent Systems, FEIT, University of Technology Sydney, Australia ({shirui.pan@student.,chengqi.zhang@}uts.edu.au).

This work was supported by the National Scholarship for Building High Level Universities, China Scholarship Council (No. 201206410056).

[7], [8]. One major approach to mitigate NB's primary weakness (*i.e.,* the conditional independence assumption of the attributes) is to assign a weight value to each individual attribute. Because weight values enforce attributes to play different roles in classification, the resulting Weighted Naive Bayes (WNB) can help relax the conditional independence assumption and make NB more efficient for real-world applications [9]. In order to discover proper weight values for WNB to improve the resulting classification performance, researchers have proposed many useful methods to evaluate the importance of attributes. Examples include Gain Ratio [10], CFS (Correlation-based Feature Selection) attribute selection algorithm [11], Mutual Information [12], and ReliefF attribute ranking algorithm [13]. Moreover, Hall [14] proposed a new attribute weighting method, based on the degree of dependence on other attributes. Recently, we also proposed a self-adaptive attribute weighting approach base on artificial immune system [6].

Another practical issue of BNC learning is the high variance with respect to the limited number of training samples [15], where the estimated class distribution of a BNC classifier is inaccurate if the number of training instances is small. In this case, one can assign different weight values to instances for probability value estimation. One similar approach is instance clone, which adds some clone examples to the training set [16] to generate more training samples. For instance clone based approaches, the underlying sample distributions should be known in advance, which is, unfortunately, not the case in reality. Jiang [17] put forward an instance-cloned naive Bayes (ICNB), which produces an expanded training set by cloning some training instances based on their similarities to the test instance, and then a naive Bayes classifier is trained from the expanded training data set to classify test instances. In another instance weighting approach, Jiang [18] also proposed to use instance weighting to improve the performance of Averaged One-Dependence Estimators, which is another Bayesian model. For this type of weighting method, each training instance is eagerly weighted according to the similarity with the "model" of training data set.

The above attribute and instance weighting methods for NB have achieved good performance to solve domain specific problems. However, these two types of weighting are treated in a separated way without utilizing the relationship between attributes and instances. In this paper, we propose a novel model, namely DWNB, with a hybrid of instance and attribute weighting: (1) an instance similarity based method is used to weight instances in training data; and (2) apply attribute weighting approach to weight instances for further estimating probability distributions. In addition, we also analyze the performance of the proposed DWNB model compared with attribute weighted NB (AWNB) and instance weighted NB (IWNB). Experiments and comparisons, on 36 UCI benchmark data sets [19] demonstrate that the proposed DWNB, with a dual weighting strategy, consistently outperforms simple instance or attribute weighted NB algorithms.

Algorithm 1 DWNB (Dual Weighted Naive Bayes)

Input:

Training data set $\mathcal{D}$;

Output:

The target class label $c(\mathbf{x}_t)$ of a test instance $\mathbf{x}_t$;

$\mathbf{x}^* \leftarrow$ Finding the frequent instance from $\mathcal{D}$.
for all $\mathbf{x}_i$ in $\mathcal{D}$ **do**
  $s(\mathbf{x}_i, \mathbf{x}^*) \leftarrow$ Apply Eq. (3) to measure the similarity between the instance $\mathbf{x}_i$ and the frequent instance $\mathbf{x}^*$.
  $w'_i \leftarrow$ Apply 1+$s(\mathbf{x}_i, \mathbf{x}^*)$ to set the ith instance weight.
end for
$w_j \leftarrow$ Apply a tree based method to calculate the weight of jth attribute A_j on the instance weighted $\mathcal{D}$.
$P(c_k) \leftarrow$ Apply w'_i to calculate instance weighted class probability via Eq. (6).
$P(a_j^\tau|c_k) \leftarrow$ Apply w'_i to calculate instance weighted conditional probability via Eq. (7).
$H_{DWNB} \leftarrow$ Apply $w_j, P(c_k), P(a_j^\tau|c_k)$ to build the DWNB model.
$c(\mathbf{x}_t) \leftarrow$ Apply H_{DWNB} to instance $\mathbf{x}_t$ to predict the underlying class label.

II. DWNB: Dual Instance and Attribute Weighted Naive Bayes

In this section, we propose a dual instance and attribute weighting approach for weighted NB classification. In our design, the instance weighting and attribute weighting is in an irreversible order, because the attribute weighting is based on distributions estimated from the instance weighting strategy. In DWNB, we first employ an instance similarity based approach to weight each instance in the training data and then use a decision tree based attribute weighting method to build weighted naive Bayes models, in which the probabilities are calculated by embedding the instance weight information.

A. Instance Weighted Strategy

In this section, we first introduce some important notations and definitions, then propose our solutions.

DEFINITION 1 **Similarity between Instances**: Given two instances $\mathbf{x}_i$ and $\mathbf{x}_j$, the similarity, denoted by $s(\mathbf{x}_i, \mathbf{x}_j)$, between them is as

$$s(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{n} I(x_{i,k} = x_{j,k}) \tag{3}$$

where $I(\cdot)$ is 1, if the condition inside is true and 0 otherwise.

DEFINITION 2 **Frequent Instance**: A frequent instance is an artificial instance created by using the most frequent value of each attribute (*i.e.,* we check the frequency of each attribute value across the whole training set, and find the most frequent attribute value for each attribute to create a frequent instance.)

For each single instance $\mathbf{x}_i \in \mathcal{D}$, we compute its similarity with the frequent instance $\mathbf{x}^*$, $s(\mathbf{x}^*, \mathbf{x}_i)$. After that, the instance weight of $\mathbf{x}_i$, w'_i, will be set to $1 + s(\mathbf{x}^*, \mathbf{x}_i)$.

B. Attribute Weighting Strategy

Given a test instance x_t, which only contains attribute values, the class label y_t of x_t needs to be predicted by a DWNB model trained from weighted instances in $\mathcal{D}$, as defined in Eq. (4),

$$c(x_t) = \arg\max_{c_k \in \mathcal{Y}} P(c_k) \prod_{j=1}^{n} P(x_{t,j} \,|c_k)^{\,w_j} \tag{4}$$

where w_j denotes the weight of the jth attribute. In DWNB, we employ a weighting method proposed by Hall [14] that assigns weight values to the attributes, which strongly depend on other attributes. In order to estimate each attribute's dependence on other attributes, an unpruned decision tree is constructed from the training instances with a minimum $depth$ indicating the depth for testing the tree. The weight for the attribute A_j is set as

$$w_j = 1/\sqrt{d_{A_j}} \tag{5}$$

Where d_{A_j} is the minimum $depth$ at which the attribute A_j is tested in the tree. Attributes that do not appear in the tree receive a zero weight value.

C. Instance Weight Embedding Probability Estimation

The challenge for DWNB is to accurately calculate the priori probability $P(c_k)$ and the conditional probability $P(x_{t,j}|c_k)$, by taking the instance weights into consideration. In this paper, we propose an instance weighting embedded Laplace-estimate to solve the above problem as given in Eqs. (6) and (7).

$$P(c_k) = \frac{n_k + 1.0}{\mathrm{N_k} + L} \tag{6}$$

$$P(a_j^\tau|c_k) = \frac{n_k^{(j,\tau)} + 1.0}{n_k + |A_j|} \tag{7}$$

$$\begin{aligned} N_k &= \sum_{x_i \in \mathcal{D}} w'_i; \quad n_k = \sum\nolimits_{x_i \in \mathcal{D}; y_i = c_k} w'_i; \\ n_k^{(j,\tau)} &= \sum_{x_i \in \mathcal{D}; y_i = c_k; x_{i,j} = a_j^\tau} w'_i \end{aligned} \tag{8}$$

where $|A_j|$ is the number of distinct values of attribute A_j, and L is the number of classes in $\mathcal{D}$. w'_i denotes the weight of the ith instance.

Algorithm 1 lists detailed procedures of the proposed D-WNB. Given a training data set $\mathcal{D}$ with N instances, DWNB firstly finds the frequent instance $\mathbf{x}^*$ from the training data. For each single instance $\mathbf{x}_i \in \mathcal{D}$, we compute the similarity $s(\mathbf{x}^*, \mathbf{x}_i)$ by using Eq. (3), and then set the weight of $\mathbf{x}_i$, w'_i, to $1 + s(\mathbf{x}^*, \mathbf{x}_i)$. After that DWNB trains a weighted naive Bayes, in which a tree based approach is employed to obtain the attribute weight. Meanwhile, we propose an instance weighting method to calculate the underlying probabilities by using Eqs. (6) and (7).

III. Experiments

A. Experimental Conditions

We implemented the proposed method using WEKA [20] data mining tool and validate its performance on 36 benchmark data sets from the UCI data repository [19]. The data characteristics are described in Table I. Because naive Bayes classifier is designed for categorical attributes, in our experiments, we first replace all missing attribute values using unsupervised attribute filter $ReplaceMissingValues$ in WEKA. Then, we apply unsupervised filter $Discretize$ in WEKA to discretize numeric attributes into nominal attributes by using unsupervised 10-bin discretization. Apparently, if the number of values of an attribute is close to the number of examples in a data set, it will result in poor probability estimation. Thus, we use an unsupervised filter named $Remove$ in Weka to remove this type of attributes.

B. Baseline Methods

For comparison purposes, we use the following baseline algorithms in our experiments.

1. NB: The standard naive Bayes classifier with conditional attribute independence assumption [3].
2. $AWNB$: Attribute weighted naive Bayes with the weight of each attribute inversely proportional to the degree of correlation between the underlying attribute and all other attributes [14].

TABLE I
DETAILED INFORMATION OF THE BENCHMARK DATA SETS

Data set	Instances	Attributes	Classes	Missing	Numeric
anneal	898	39	6	Y	Y
anneal.ORIG	898	39	6	Y	Y
audiology	226	70	24	Y	N
autos	205	26	7	Y	Y
balance-scale	625	5	3	N	Y
breast-cancer	286	10	2	Y	N
breast-w	699	10	2	Y	N
colic	368	23	2	Y	Y
colic.ORIG	368	28	2	Y	Y
credit-a	690	16	2	Y	Y
credit-g	1000	21	2	N	Y
diabetes	768	9	2	N	Y
Glass	214	10	7	N	Y
heart-c	303	14	5	Y	Y
heart-h	294	14	5	Y	Y
heart-statlog	270	14	2	N	Y
hepatitis	155	20	2	Y	Y
hypothyroid	3772	30	4	Y	Y
ionosphere	351	35	2	N	Y
iris	150	5	3	N	Y
kr-vs-kp	3196	37	2	N	N
labor	57	17	2	Y	Y
letter	20000	17	26	N	Y
lymph	148	19	4	N	Y
mushroom	8124	23	2	Y	N
primary-tumor	339	18	21	Y	N
segment	2310	20	7	N	Y
sick	3772	30	2	Y	Y
sonar	208	61	2	N	Y
soybean	683	36	19	Y	N
splice	3190	62	3	N	N
vehicle	846	19	4	N	Y
vote	435	17	2	Y	N
vowel	990	14	11	N	Y
waveform-5000	5000	41	3	N	Y
zoo	101	18	7	N	Y

TABLE II

EXPERIMENTAL RESULTS FOR DWNB VS. AWNB, IWNB, NB, SBC, AND C4.4: CLASSIFICATION ACCURACY AND STANDARD DEVIATION.

Data Sets	DWNB	AWNB		IWNB		NB		SBC		C4.4	
anneal	91.15±2.08	89.73±1.94	•	97.07±1.50	○	94.32±2.23	○	94.03±2.37	○	99.06±0.91	○
anneal.ORIG	89.92±2.67	88.11±2.53	•	88.24±3.14		88.16±3.06		84.66±3.74	•	92.26±2.43	○
audiology	79.15±8.21	71.92±6.67	•	80.18±8.17		71.40±6.37	•	70.91±7.09	•	77.58±7.82	
autos	75.26±9.15	67.76±10.51	•	65.47±10.94	•	63.97±11.35	•	70.10±9.38		83.68±7.70	○
balance-scale	89.90±1.95	90.03±1.99		91.25±1.40	○	91.44±1.30	○	91.44±1.30	○	68.55±4.20	•
breast-cancer	72.17±7.74	72.39±7.47		72.18±8.18		72.94±7.71		73.25±7.60		68.29±7.84	
breast-w	97.50±1.76	97.34±1.81		97.47±1.75		97.30±1.75		97.30±1.75		94.02±2.87	•
colic	83.67±5.45	83.64±5.47		80.30±5.85	•	78.86±6.05	•	82.28±5.86		79.91±6.00	•
colic.ORIG	76.19±6.10	76.00±6.53		73.34±7.08		74.21±7.09		74.57±5.85		76.68±6.71	
credit-a	86.23±3.86	86.46±3.85		84.57±3.76		84.74±3.83		85.75±4.16		83.36±4.24	•
credit-g	76.79±3.70	76.14±3.62		75.99±3.93		75.93±3.87		72.43±3.61	•	68.97±3.95	•
diabetes	76.26±5.23	76.91±5.07		75.09±5.03		75.68±4.85		75.93±5.07		70.73±4.39	•
glass	60.16±9.42	57.44±9.37		58.90±10.12		57.69±10.07		56.22±10.36		54.64±8.56	
heart-c	82.32±6.48	83.57±6.04		82.92±6.98		83.44±6.27		84.20±6.37		75.32±7.75	•
heart-h	83.51±6.31	83.34±6.28		84.36±5.89		83.64±5.85		82.59±6.40		75.60±6.72	•
heart-statlog	83.26±5.43	84.04±5.90		83.19±5.56		83.78±5.41		84.19±6.07		76.67±7.94	•
hepatitis	84.28±8.12	83.35±8.24		85.50±9.63		84.06±9.91		83.60±9.77		79.83±9.98	
hypothyroid	93.63±0.51	93.58±0.50		92.69±0.75	•	92.79±0.73	•	93.52±0.48		92.39±0.75	•
ionosphere	92.03±3.86	92.00±4.06		88.75±5.16	•	90.86±4.33	•	90.89±4.72		85.75±5.81	•
iris	95.27±5.47	95.53±5.19		94.00±6.46		94.33±6.79		96.87±4.29		94.20±5.25	
kr-vs-kp	94.75±1.25	94.21±1.29	•	88.25±1.88	•	87.79±1.91	•	92.38±1.56	•	99.44±0.40	○
labor	89.17±11.99	88.10±12.66		94.23±10.25		96.70±7.27		84.97±12.91		84.00±15.37	
letter	67.63±2.01	65.99±2.08	•	67.08±1.92		65.80±2.04	•	67.32±2.22		72.02±2.09	○
lymph	81.90±9.09	82.20±10.01		85.15±8.77		85.97±8.88	○	82.72±9.39		76.55±10.76	
mushroom	99.54±0.61	97.82±1.10	•	96.97±1.58	•	93.58±2.03	•	98.12±1.14	•	99.75±0.36	
primary-tumor	48.14±5.32	45.54±5.39	•	46.82±6.34		47.20±6.02		45.78±6.84		40.65±6.50	•
segment	90.89±1.63	90.48±1.56		89.96±1.68	•	89.03±1.66	•	90.46±2.10		93.81±1.58	○
sick	97.30±0.84	96.94±0.92	•	97.07±0.82	•	96.78±0.91	•	96.81±0.89	•	97.83±0.76	○
sonar	75.33±9.00	75.36±8.81		76.96±9.55		76.35±9.94		73.54±9.45		68.74±8.79	•
soybean	94.30±2.09	92.85±2.90	•	93.95±2.69		92.20±3.23	•	91.00±3.31	•	92.56±2.74	
splice	96.12±1.01	96.14±1.03		95.57±1.11		95.42±1.14	•	95.84±1.03		92.16±1.40	•
vehicle	60.97±3.59	61.75±3.44		60.72±3.42		61.03±3.48		56.32±4.01	•	69.24±4.15	○
vote	94.83±3.01	94.83±3.01		90.32±3.86	•	90.21±3.95	•	94.46±2.81		95.21±3.11	
vowel	66.71±5.01	66.65±4.73		65.94±4.81		66.09±4.78		62.75±5.10	•	78.56±4.88	○
waveform-5000	80.24±3.04	79.56±3.00		80.24±2.83		79.80±2.97		79.71±2.97		68.65±5.07	•
zoo	93.61±7.18	90.65±7.29		96.05±5.60		94.37±6.79		91.51±7.68		93.41±7.28	
Mean	83.34±4.73	82.45±4.79		82.69±4.96		82.16±4.88		81.90±4.99		81.11±5.20	

○, •: Statistically significant upgradation and degradation, respectively.

3. $IWNB$: Instance weighted naive Bayes with the weighting method based on the instance similarity [18].
4. SBC: A bagged decision-tree based attribute selection filter for naive Bayes [21].
5. $C4.4$: A specially designed tree to improve C4.5 [22] performance on classification ranking.
6. **DWNB**: The proposed method, dual weighting naive Bayes, which calculates the probabilities through a instance weight embedding method.

In our experiments, we implemented DWNB, IWNB, AWNB, and SBC, and used implementation of NB and C4.4 in Weka framework. For the attribute weighting approaches, we used the package provided by Mark Hall [14]. C4.4 is J48 in Weka with Laplace correction and without pruning.

We compare the performance of the proposed DWNB with AWNB, IWNB, NB, SBC and C4.4. The attribute weighting method in AWNB and instance weighting strategy in IWNB are similar to the ones used in DWNB. For the proposed DWNB, we do not consider any other attribute and instance weighting approaches. This is due to the fact that DWNB intends to propose a general dual weighting model, which can be applied to any attribute or instance weighting approach. The purpose of our experiment is to demonstrate the effectiveness of the proposed dual weighting model.

In our experiments, the algorithms are evaluated in terms of classification accuracy, which is a well accepted evaluation metric in many real-world applications [23], [24], [25]. The accuracy performance of each method is calculated by the percentage of successful predictions on the test data sets via

TABLE III

TWO-TAILED t-TEST ON ACCURACY WITH 95% CONFIDENCE LEVEL.

	C4.4	SBC	NB	IWNB	AWNB
SBC	12/13/11				
NB	13/12/11	5/25/6			
IWNB	13/12/11	9/23/4	8/28/0		
AWNB	15/11/10	6/27/3	7/26/3	7/25/4	
DWNB	**15/12/9**	**9/25/2**	**12/21/3**	**9/25/2**	**10/26/0**

10 runs of 10-fold cross validation, with all algorithms being evaluated on the same training and testing data sets. Finally, we compare related algorithms via two-tailed t-test with a 95% confidence level. Based on the statistical theory, the difference is statistically significant only if the probability of significant difference is at least 95 percent, *i.e.,* the p-value for a t-test between two algorithms is less than 0.05.

C. High Performance of Dual Weighted DWNB

Table II reports the detailed results (the accuracy and standard deviation) of DWNB and other baseline algorithms, with the mean values and standard deviation on all data sets being summarized at the bottom. In the table, symbols $\circ$ and $\bullet$ represent statistically significant upgradation and degradation over DWNB with the p-value less than 0.05. Besides, Table III illustrates the compared results of two-tailed t-test, in which each entry $w/t/l$ means that the algorithm in the corresponding row wins in w data sets, ties in t data sets, and loses in l data sets on the 36 benchmark UCI data sets, compared to the algorithm in the corresponding column. Overall, the results can be summarized as follows:

1. Instance weighting IWNB significantly outperforms NB (8 wins and 0 losses), and the average classification accuracy on 36 benchmark UCI data sets for IWNB (82.69±4.96) is higher than NB (82.16±4.88).
2. Attribute weighting AWNB sightly outperforms NB with (7 wins and 3 losses), and the average classification accuracy on 36 benchmark UCI data sets for AWNB (82.45±4.79) is higher than than NB (82.16±4.88).
3. AWNB shows a little better performance compared with IWNB (7 wins and 4 losses). However, the average classification accuracy for AWNB (82.45±4.79) is lower than IWNB (82.69±4.96).
4. DWNB significantly outperforms IWNB (9 wins and 2 losses), also significantly outperforms AWNB (10 wins and 0 losses) on 36 benchmark UCI data sets.
5. DWNB greatly outperforms all other baselines with the superior average classification accuracy on 36 data sets (83.34±4.73).

IV. Conclusion and Future Work

We proposed a dual weighting strategy for weighted naive Bayes classification, which calculates the probability values by using a dual instance and attribute weighting approach. The experiments and comparisons on 36 benchmark UCI data sets with respect to the classification accuracy performance show that DWNB outperforms existing NB weighting models, such as instance weighted IWNB and attribute weighted AWNB. The proposed dual attribute and instance weighting can also be extended to other Bayesian networks.

References

[1] J. Hernández-González, I. n. Inza, and J. A. Lozano, "Learning bayesian network classifiers from label proportions," *Pattern Recogn.*, vol. 46, no. 12, pp. 3425–3440, 2013.

[2] D. M. Chickering, "Learning bayesian networks is np-complete," in *Learning from Data: Artificial Intelligence and Statistics V*. Springer-Verlag, 1996, pp. 121–130.

[3] N. Friedman, D. Geiger, and M. Goldszmidt, "Bayesian network classifiers," *Mach. Learn.*, vol. 29, no. 2-3, pp. 131–163, 1997.

[4] X. Wu, V. Kumar, J. Ross Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P. S. Yu, Z.-H. Zhou, M. Steinbach, D. J. Hand, and D. Steinberg, "Top 10 algorithms in data mining," *Knowl. Inf. Syst.*, vol. 14, no. 1, pp. 1–37, 2007.

[5] G. I. Webb, J. R. Boughton, F. Zheng, K. M. Ting, and H. Salem, "Learning by extrapolation from marginal to full-multivariate probability distributions: Decreasingly naive bayesian classification," *Mach. Learn.*, vol. 86, no. 2, pp. 233–272, 2012.

[6] J. Wu, Z. Cai, S. Zeng, and X. Zhu, "Artificial immune system for attribute weighted naive bayes classification," in *Proceedings of the International Joint Conference on Neural Networks*, Dallas, TX, USA, 2013, pp. 798–805.

[7] J. Wu, Z. Cai, and X. Zhu, "Self-adaptive probability estimation for naive bayes classification," in *Proceedings of the International Joint Conference on Neural Networks*. Dallas, TX, USA: IEEE Publishers, 2013, pp. 2303–2310.

[8] G. Webb, J. Boughton, F. Zheng, K. Ting, and H. Salem, "Learning by extrapolation from marginal to full-multivariate probability distributions: decreasingly naive bayesian classification," *Machine Learning*, vol. 86, no. 2, pp. 233–272, 2012.

[9] N. A. Zaidi, J. Cerquides, M. J. Carman, and G. I. Webb, "Alleviating naive bayes attribute independence assumption by attribute weighting," *Journal of Machine Learning Research*, vol. 14, pp. 1947–1988, 2013.

[10] H. Zhang and S. Sheng, "Learning weighted naive bayes with accurate ranking," in *Proc. of IEEE International Conference on Data Mining*, ser. ICDM, Washington, DC, USA, 2004, pp. 567–570.

[11] M. A. Hall, "Correlation-based feature selection for discrete and numeric class machine learning," in *Proceedings of the Seventeenth International Conference on Machine Learning*, ser. ICML '00, San Francisco, CA, USA, 2000, pp. 359–366.

[12] L. Jiang, H. Zhang, Z. Cai, and D. Wang, "Weighted average of one-dependence estimators?." *J. Exp. Theor. Artif. Intell.*, vol. 24, pp. 219–230, 2012.

[13] M. Robnik-Šikonja and I. Kononenko, "Theoretical and empirical analysis of relieff and rrelieff," *Mach. Learn.*, vol. 53, no. 1-2, pp. 23–69, 2003.

[14] M. Hall, "A decision tree-based attribute weighting filter for naive bayes," *Know.-Based Syst.*, vol. 20, no. 2, pp. 120–126, 2007.

[15] L. Jiang, H. Zhang, and Z. Cai, "A novel bayes model: Hidden naive bayes," *IEEE Trans. on Knowl. and Data Eng.*, vol. 21, no. 10, pp. 1361–1371, 2009.

[16] L. Jiang, D. Wang, H. Zhang, Z. Cai, and B. Huang, "Using instance cloning to improve naive bayes for ranking." *IJPRAI*, vol. 22, pp. 1121–1140, 2008.

[17] L. Jiang and H. Zhang, "Learning instance greedily cloning naive bayes for ranking," in *Proc. of IEEE International Conference on Data Mining*, ser. ICDM, DC, USA, 2005, pp. 202–209.

[18] L. Jiang, Z. Cai, and D. Wang, "Learning averaged one-dependence estimators by instance weighting." *Journal of Computational Information Systems*, vol. 4, pp. 2753–2760, 2008.

[19] K. Bache and M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml

[20] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd ed., ser. The Morgan Kaufmann Series in Data Management Systems. San Francisco, CA: Morgan Kaufmann Publishers, 2005. [Online]. Available: http://www.cs.waikato.ac.nz/ml/weka/

[21] P. Langley and S. Sage, "Induction of selective bayesian classifiers," in *Proceedings of the Tenth International Conference on Uncertainty in Artificial Intelligence*, ser. UAI'94. Morgan Kaufmann, 1994, pp. 339–406.

[22] J. R. Quinlan, *C4.5: programs for machine learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1993.

[23] J. Wu, X. Zhu, C. Zhang, and Z. Cai, "Multi-instance multi-graph dual embedding learning," in *Data Mining (ICDM), 2013 IEEE 13th International Conference on*, 2013, pp. 827–836.

[24] S. Pan, X. Zhu, C. Zhang, and P. Yu, "Graph stream classification using labeled and unlabeled graphs," in *Data Engineering (ICDE), 2013 IEEE 29th International Conference on*, 2013, pp. 398–409.

[25] J. Wu, X. Zhu, C. Zhang, and P. Yu, "Bag constrained structure pattern mining for multi-graph classification," *Knowledge and Data Engineering, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2014.