International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 22 (2017) pp. 12693-12700

Defect Classification Using Naïve Bayes Classification

Dr. S.Veni
Professor and Head: Department of Computer Science, Karpagam Academy of Higher Education
Karpagam University, Coimbatore, Tamil Nadu, India.

Aparna Srinivasan
Research Scholar: Department of Computer Science, Karpagam Academy of Higher Eduction,
Karpagam University, Coimbatore, Tamil Nadu, India.

Abstract

In software development life cycle (SDLC) identifying defects and classifying them as Major, Moderate and Minor is important to find the impact on effort, schedule and cost. Classification is a data mining task to assign a data item to a predefined set of classes. The performance of classifier depends upon the type of defects identified in various stages by the class label of severity in the training data. In this research work, Naive Bayes classification is used to predict the class label of "severity" tuple. The data tuples are described by the attributes of Defect, Phase Detected, Phase Attributed, Impact and Weight.

Keywords: Data mining; Naïve Bayes Classification; Defect phases; Confusion matrix; Higher posterior probability

INTRODUCTION

The derived model is based on the analysis of training data. The classification of data is a two-step process: 1) learning and 2) classification used to predict class labels for training data.

In probability theory, Naïve Bayes classifier checks the condition rules and classified the data in the learning phase and checks if classification holds good in the testing phase [8] [6]. Bayesian reasoning is applied in decision making that deals with probability inference which is used to gather the knowledge of prior events by predicting events through rules [9, 10, 11, 12, 13, 14, 15].

This research work focuses on classifying severity of minor, moderate and major defects based on the Variables of Defect, Phase detected, Phase attributed, Impact and Weight. It enables organizations to improve the quality of management decision making by ensuring that reliable and secure information and data is available throughout the Software life cycle.

DATA FOR RESEARCH

Objective

The main idea is to integrate the information given in a set of predictors into the Naive Rule to obtain more accurate classifications. The probability of a record belonging to a certain class of severity can be evaluated for predicting the defect "severity".

Naive Bayes works only with predictors that are categorical.

- Numerical predictors must be binned and converted to categorical variables before the Naive Bayes Classifier can use.

DATA COLLECTION

In this research work, evaluation can be carried out on different stages of project development such as Requirements, Design, Build and Testing as shown in Table 1.

Defects classified based on below parameters are assigned a weight based on Phase Detected, Phase Attributed and Severity. Parameters values for Phase Detected, Phase Attributed and Severity are provided below.

- Phase Detected can be Requirements, Design Review, Code Review and Unit Testing, System Testing, User Acceptance Testing, Implementation and Post Implementation Testing and
- Phase Attributed can be the Requirements, Design, Coding and Implementation
- Severity can be the Minor, Moderate and Major.

Research work focusses on Naïve Bayes classification techniques to predict the Severity of defects that arise in various stages of SDLC. Here 115 instance of training data of defects are used to predict the model.

METHODOLOGY

Naive Bayes Classifier

In Classification process, the derived model is to predict the class of objects whose class label is unknown [8, 11]. The derived model is based on the analysis of asset of training data. This paper discusses about the accurate prediction of defect severity as shown in Table 3 that occurs in various phases of software development life cycle based on Naive Bayes Classification technique [7].

The probabilities are descriptive and are then used to predict the class membership for a target tuple [1, 2, 3, 4, 5]. The advantage of Naive Bayes classifier is that requires fewer amounts of training data to estimate the parameters necessary for classification. To classify the target value, the constraint and probabilities are estimated by the rule theorem of equation (1). It assumes the variables of training data correspond to the values as shown in the below Table 2.

Table 1: Class Label Training tuples for the Software development Life cycle Project defects

S.I No.	Defect	Phase Detected	Phase Attributed	Impact	Weight	Severity
1	Ambiguous Requirements	Requirements	Requirements	10%	1	Minor
2	Inadequate Requirements	Requirements	Requirements	10%	1	Minor
3	Incorrect Requirements	Requirements	Requirements	10%	1	Minor
4	Missing Requirements	Requirements	Requirements	10%	1	Minor
5	Ambiguous Design	Design Review	Design	10%	1	Minor
6	Boundary Conditions Neglected in Design	Design Review	Design	10%	1	Minor
7	Data error	Design Review	Design	10%	1	Minor
8	Database Error	Design Review	Design	10%	1	Minor
9	Incorrect Design	Design Review	Design	10%	1	Minor
10	Inadequate Design	Design Review	Design	10%	1	Minor
11	Sub optimal Design	Design Review	Design	10%	1	Minor
12	Message Error	Design Review	Design	10%	1	Minor
13	Missing Design	Design Review	Design	10%	1	Minor
14	Test Plan / Cases Error	Design Review	Design	10%	1	Minor
15	Ambiguous Requirements	Design Review	Requirements	25%	2.5	Moderate
16	Inadequate Requirements	Design Review	Requirements	25%	2.5	Moderate
17	Incorrect Requirements	Design Review	Requirements	25%	2.5	Moderate
18	Missing Requirements	Design Review	Requirements	25%	2.5	Moderate
19	Computational Error	Code Review & Unit Testing	Coding	10%	1	Minor
20	Boundary Conditions Neglected in Code	Code Review & Unit Testing	Coding	10%	1	Minor
21	Interface Error	Code Review & Unit Testing	Coding	10%	1	Minor
22	Logic Error	Code Review & Unit Testing	Coding	10%	1	Moderate
23	Navigation Error	Code Review & Unit Testing	Coding	10%	1	Minor
24	Performance Error	Code Review & Unit Testing	Coding	10%	1	Moderate
25	Sequencing / Timing Error	Code Review & Unit Testing	Coding	10%	1	Moderate

Table.2: Domain variable declaration of training data of 115 project defect

Requirement Defects	**Possible values**
Ambiguous Requirements	R1
Inadequate Requirements	R2
Incorrect Requirements	R3
Missing Requirements	R4
Design Defects	
Ambiguous Design	D1
Boundary Conditions Neglected in Design	D2
Data error	D3
Database Error	D4
Incorrect Design	D5
Inadequate Design	D6
Sub optimal Design	D7
Message Error	D8
Missing Design	D9
Test Plan / Cases Error	D10
Code Review & Unit Testing Defects	
Computational Error	CU1
Boundary Conditions Neglected in Code	CU2
Interface Error	CU3
Logic Error	CU4
Navigation Error	CU5
Performance Error	CU6
Sequencing / Timing Error	CU7
Missing / Inadequate Standards in Code	CU8
Typographical Error	CU9
Variable Declaration Error	CU10
Implementation Defects	
Deployment Error	DE
Phase Attributed	
Requirements	PA1
Design	PA2
Coding	PA3
Implementation	PA4
Phase Detected	
Requirements	PD1
Design Review	PD2
Code Review & Unit testing	PD3
System Testing	PD4
User Acceptance Testing	PD5
Implementation & Post Implementation Testing	PD6

Table 3: Range of Severity

Severity
Major
Moderate
Minor

By Bayes' theorem, the classic for which P (Ci |X) represents maximum posterior hypothesis [6].

$$P(C_i|X) = P(X|C_i)\,P(C_i) / P(X) \qquad (1)$$

The classic for which P (C_i |X) is maximized is called the maximum posteriori hypothesis. It can easily estimate the probabilities $P(x_1|C_i)\times P(x_2|C_i)\times\cdots\times P(xn|C_i)$ from the training tuples by the following relationship.

$$P(X|C_i) = \prod_{k=1}^{n} P(x_k|C_i) \qquad (2)$$

The attributes are conditionally independent to one another by given the class label of the tuple which predicts the data in tuple where X belongs to the class Ci.

$$P(C_i|X) > P(C_j|X) \quad \text{for } 1 \le j \le m, j \ne i \qquad (3)$$

The probabilities are descriptive and are then used to predict the class membership for a target tuple. The advantage of Naive Bayes classifier is that is requires an amount of training data to estimate the parameters necessary for classification. To classifying the target value, the constraint and probabilities are estimated by the rule theorem of equation (1). The classic for which P (C_i |X) is maximized is called the maximum posterior hypothesis. It can easily estimate the probabilities $P(x_1|C_i)\times P(x_2|C_i)\times\cdots\times P(xn|C_i)$ from the training tuple equation.

In this research work, Naive Bayes classification algorithm can be used to predict the class label of "severity" tuple. The data tuples are describe by the attributes of Defect, Phase Detect, Phase Attribute, Impact and weight. The class label attribute "severity" has three distinct values namely {Major, Moderate, Minor}

Let C1 correspond to the class severity = Major, C2 correspond to the class severity = Moderate and C3 correspond to the class severity = Minor. Assume that the tuples are to be classified as given below in Table 4.

Using rule based classification, If – Then rules can be used to classify the weight and impact of defect attributes. The rule base of weight and impact using IF- THEN is expressed in the form given below:

If weight = 1 and weight < 2.5and impact =10 and impact < 25 Then Factor = Low

If weight >= 2.5 and weight <=5and impact >= 25 and impact <= 50 Then Factor = Medium

If weight > 5 and weight = 10 and impact > 50 and impact =100 Then Factor = High

Table 4: Classify X from the tuples to maximize $P(X|C_i)\ P(C_i)$

Classification of tuples "X"						Then
Estimation	Defect	Phase Detected	Phase Attributed	Impact	Weight	Severity
X1	R1	PD1	PA1	10%	1	Minor
X2	R2	PD1	PA1	10%	1	Minor
X3	R3	PD1	PA1	10%	1	Minor
X4	R4	PD1	PA1	10%	1	Minor
X5	D1	PD2	PA2	10%	1	Minor
X6	D2	PD2	PA2	10%	1	Minor
X7	D3	PD2	PA2	10%	1	Minor
X8	D4	PD2	PA2	10%	1	Minor
X9	D5	PD2	PA2	10%	1	Minor
X10	D6	PD2	PA2	10%	1	Minor
X11	D7	PD2	PA2	10%	1	Minor
X12	D8	PD2	PA2	10%	1	Minor
X13	D9	PD2	PA2	10%	1	Minor
X14	D10	PD2	PA2	10%	1	Minor
X15	R1	PD2	PA1	25%	2.5	Moderate
X16	R2	PD2	PA1	25%	2.5	Moderate
X17	R3	PD2	PA1	25%	2.5	Moderate
X18	R4	PD2	PA1	25%	2.5	Moderate
X19	CU1	PD3	PA3	10%	1	Minor
X20	CU2	PD3	PA3	10%	1	Minor
X21	CU3	PD3	PA3	10%	1	Minor
X22	CU4	PD3	PA3	25%	2.5	Moderate
X23	CU5	PD3	PA3	10%	1	Minor
X24	CU6	PD3	PA3	25%	2.5	Moderate
X25	CU7	PD3	PA3	25%	2.5	Moderate
X26	CU8	PD3	PA3	10%	1	Minor
X27	CU9	PD3	PA3	10%	1	Minor
X28	CU10	PD3	PA3	10%	1	Minor
X29	D1	PD3	PA2	25%	2.5	Moderate
X30	D2	PD3	PA2	25%	2.5	Moderate
X31	D3	PD3	PA2	25%	2.5	Moderate
X32	D4	PD3	PA2	25%	2.5	Moderate
X33	D5	PD3	PA2	25%	2.5	Moderate
X34	D6	PD3	PA2	25%	2.5	Moderate
X35	D7	PD3	PA2	25%	2.5	Moderate
X36	D8	PD3	PA2	25%	2.5	Moderate
X37	D9	PD3	PA2	25%	2.5	Moderate
X38	D10	PD3	PA2	25%	2.5	Moderate
X39	R1	PD3	PA1	50%	5	Moderate
X40	R2	PD3	PA1	50%	5	Moderate
X41	R3	PD3	PA1	50%	5	Moderate
X42	R4	PD3	PA1	50%	5	Moderate
X43	CU1	PD4	PA3	10%	1	Minor
X44	CU2	PD4	PA3	10%	1	Minor
X45	CU3	PD4	PA3	10%	1	Minor
X46	CU4	PD4	PA3	25%	2.5	Moderate
X47	CU5	PD4	PA3	25%	2.5	Moderate
X48	CU6	PD4	PA3	25%	2.5	Moderate
X49	CU7	PD4	PA3	25%	2.5	Moderate
X50	CU8	PD4	PA3	10%	1	Minor
X51	CU9	PD4	PA3	10%	1	Minor
X52	CU10	PD4	PA3	25%	2.5	Moderate
X53	D1	PD4	PA2	60%	6	Major
X54	D2	PD4	PA2	60%	6	Major
X55	D3	PD4	PA2	60%	6	Major
X56	D4	PD4	PA2	60%	6	Major
X57	D5	PD4	PA2	60%	6	Major
X58	D6	PD4	PA2	60%	6	Major
X59	D7	PD4	PA2	60%	6	Major
X60	D8	PD4	PA2	60%	6	Major
X61	D9	PD4	PA2	60%	6	Major
X62	D10	PD4	PA2	60%	6	Major
X63	R1	PD4	PA1	60%	6	Major
X64	R2	PD4	PA1	60%	6	Major
X65	R3	PD4	PA1	60%	6	Major
X66	R4	PD4	PA1	60%	6	Major
X67	CU1	PD5	PA3	60%	6	Major
X68	CU2	PD5	PA3	40%	4	Minor
X69	CU3	PD5	PA3	50%	5	Moderate
X70	CU4	PD5	PA3	60%	6	Major
X71	CU5	PD5	PA3	50%	5	Moderate
X72	CU6	PD5	PA3	60%	6	Major
X73	CU7	PD5	PA3	60%	6	Major
X74	CU8	PD5	PA3	40%	4	Minor
X75	CU9	PD5	PA3	40%	4	Minor
X76	CU10	PD5	PA3	50%	5	Moderate
X77	D1	PD5	PA2	75%	7.5	Major
X78	D2	PD5	PA2	75%	7.5	Major
X79	D3	PD5	PA2	75%	7.5	Major
X80	D4	PD5	PA2	75%	7.5	Major
X81	D5	PD5	PA2	75%	7.5	Major
X82	D6	PD5	PA2	75%	7.5	Major
X83	D7	PD5	PA2	75%	7.5	Major
X84	D8	PD5	PA2	75%	7.5	Major
X85	D9	PD5	PA2	75%	7.5	Major
X86	D10	PD5	PA2	75%	7.5	Major
X87	R1	PD5	PA1	80%	8	Major
X88	R2	PD5	PA1	80%	8	Major
X89	R3	PD5	PA1	80%	8	Major
X90	R4	PD5	PA1	80%	8	Major
X91	CU1	PD6	PA3	75%	7.5	Major
X92	CU2	PD6	PA3	75%	7.5	Major
X93	CU3	PD6	PA3	50%	5	Moderate
X94	CU4	PD6	PA3	75%	7.5	Major
X95	CU5	PD6	PA3	50%	5	Moderate
X96	CU6	PD6	PA3	75%	7.5	Major
X97	CU7	PD6	PA3	75%	7.5	Major
X98	CU8	PD6	PA3	75%	7.5	Minor
X99	CU9	PD6	PA3	75%	7.5	Minor
X100	CU10	PD6	PA3	75%	7.5	Major
X101	D1	PD6	PA2	100%	10	Major
X102	D2	PD6	PA2	100%	10	Major
X103	D3	PD6	PA2	100%	10	Major
X104	D4	PD6	PA2	100%	10	Major
X105	D5	PD6	PA2	100%	10	Major
X106	D6	PD6	PA2	100%	10	Major
X107	D7	PD6	PA2	100%	10	Major
X108	D8	PD6	PA2	100%	10	Major
X109	D9	PD6	PA2	100%	10	Major
X110	D10	PD6	PA2	100%	10	Major
X111	R2	PD6	PA1	100%	10	Major
X112	R3	PD6	PA1	100%	10	Major
X113	R4	PD6	PA1	100%	10	Major
X114	DE	PD6	PA4	20%	2	Minor
X115	R1	PD6	PA1	100%	10	Major

In classification task, depending upon the Bayes theorem, it can estimate the probability of membership in each class given a certain set of predictor variables. This type of probability is called "conditional probability". For instance, a conditional probability of event A gives the event B (denoted by P(A|B)) which represents the chances of event A occurring only under the scenario that event B occurs.

In this research, to maximize $P(X_j|C_i)P(C_i)$, for i=1,2,3 $P(C_i)$, The prior probability of each class, can be computed based on the training tuples. To classify a record, to compute its chance of belonging to each of the classes by computing $P(X_1,\ldots,X_p|C_i)$ for each class *i,* then classify the record to the class that has the highest probability.

To compute the probability of class label "severity" solution is given below:

$$P(\text{severity=Major}) = \frac{52}{115} = 0.4521$$

$$P(\text{severity=Moderate}) = \frac{31}{115} = 0.2695$$

$$P(\text{severity=Minor}) = \frac{32}{115} = 0.2782$$

The terms on the right are estimated from frequency counts in the training data, with the estimate of $P(X_j|C_i)$ being equal to the number of occurrences of the value X_j in class C_i is divided by the total number of records in that class.

Computation of $P(X_j|C_i)$, for i=1, 2...n. and conditional probability is given below,

P(Defect=R1| Severity=Minor)= 0.031

P(Defect=R1| Severity=Moderate)= 0.064

P(Defect=R1| Severity=Major)= 0.058

P(Defect=R2| Severity=Minor)= 0.031

P(Defect=R2| Severity=Moderate)= 0.064

P(Defect=R2| Severity=Major)= 0.058

P(Defect=R3| Severity=Minor)= 0.031

P(Defect=R3| Severity=Moderate)= 0.064

P(Defect=R3| Severity=Major)= 0.058

P(Defect=R4| Severity=Minor)= 0.031

P(Defect=R4| Severity=Moderate)= 0.064

P(Defect=R4| Severity=Major)= 0.058

P(Defect=D1| Severity=Minor)= 0.031

P(Defect=D1| Severity=Moderate)= 0.032

P(Defect=D1| Severity=Major)= 0.058

P(Defect=D2| Severity=Minor)= 0.031

P(Defect=D2| Severity=Moderate)= 0.032

P(Defect=D2| Severity=Major)= 0.058

P(Defect=D3| Severity=Minor)= 0.031

P(Defect=D3| Severity=Moderate)= 0.032

P(Defect=D3| Severity=Major)= 0.058

P(Defect=D4| Severity=Minor)= 0.031

P(Defect=D4| Severity=Moderate)= 0.032

P(Defect=D4| Severity=Major)= 0.058

P(Defect=D5| Severity=Minor)= 0.031

P(Defect=D5| Severity=Moderate)= 0.032

P(Defect=D5| Severity=Major)= 0.058

P(Defect=D6| Severity=Minor)= 0.031

P(Defect=D6| Severity=Moderate)= 0.032

P(Defect=D6| Severity=Major)= 0.058

P(Defect=D7| Severity=Minor)= 0.031

P(Defect=D7| Severity=Moderate)= 0.032

P(Defect=D7| Severity=Major)= 0.058

P(Defect=D8| Severity=Minor)= 0.031

P(Defect=D8| Severity=Moderate)= 0.032

P(Defect=D8| Severity=Major)= 0.058

P(Defect=D9| Severity=Minor)= 0.031

P(Defect=D9| Severity=Moderate)= 0.032

P(Defect=D9| Severity=Major)= 0.058

P(Defect=D10| Severity=Minor)= 0.031

P(Defect=D10| Severity=Moderate)= 0.032

P(Defect=D10| Severity=Major)= 0.058

P(Defect=CU1| Severity=Minor)= 0.062

P(Defect= CU1| Severity=Moderate)= 0.000

P(Defect= CU1| Severity=Major)= 0.038

P(Defect=CU2| Severity=Minor)= 0.093

P(Defect= CU2| Severity=Moderate)= 0.000

P(Defect= CU2| Severity=Major)= 0.019

P(Defect=CU3| Severity=Minor)= 0.062

P(Defect= CU3| Severity=Moderate)= 0.064

P(Defect= CU3| Severity=Major)= 0.000

P(Defect=CU4| Severity=Minor)= 0.000

P(Defect= CU4| Severity=Moderate)= 0.064

P(Defect= CU4| Severity=Major)= 0.038

P(Defect=CU5| Severity=Minor)= 0.031

P(Defect= CU5| Severity=Moderate)= 0.096

P(Defect= CU5| Severity=Major)= 0.000

P(Defect=CU6| Severity=Minor)= 0.000

P(Defect= CU6| Severity=Moderate)= 0.064

P(Defect= CU6| Severity=Major)= 0.038

P(Defect=CU7| Severity=Minor)= 0.000

P(Defect= CU7| Severity=Moderate)= 0.064

P(Defect= CU7| Severity=Major)= 0.038

P(Defect=CU8| Severity=Minor)= 0.124

P(Defect= CU8| Severity=Moderate)= 0.000

P(Defect= CU8| Severity=Major)= 0.000

P(Defect=CU9| Severity=Minor)= 0.124

P(Defect= CU9| Severity=Moderate)= 0.000

P(Defect= CU9| Severity=Major)= 0.000

P(Defect=CU10| Severity=Minor)= 0.031

P(Defect= CU10| Severity=Moderate)= 0.064

P(Defect= CU10| Severity=Major)= 0.019

P(Defect=DE| Severity=Minor)= 0.031

P(Defect= DE| Severity=Moderate)= 0.000

P(Defect= DE| Severity=Major)= 0.000

P(Phase Detected=PD1| Severity=Minor)= 0.125

P(Phase Detected=PD1| Severity=Moderate)= 0.000

P(Phase Detected=PD1| Severity=Major)= 0.000

P(Phase Detected=PD1| Severity=Minor)= 0.125

P(Phase Detected=PD1| Severity=Moderate)= 0.000

P(Phase Detected=PD1| Severity=Major)= 0.000

P(Phase Detected=PD2| Severity=Minor)= 0.312

P(Phase Detected=PD2| Severity=Moderate)= 0.129

P(Phase Detected=PD2| Severity=Major)= 0.000

P(Phase Detected=PD3| Severity=Minor)= 0.218

P(Phase Detected=PD3| Severity=Moderate)= 0.547

P(Phase Detected=PD3| Severity=Major)= 0.000

P(Phase Detected=PD4| Severity=Minor)= 0.156

P(Phase Detected=PD4| Severity=Moderate)= 0.161

P(Phase Detected=PD4| Severity=Major)= 0.269

P(Phase Detected=PD5| Severity=Minor)= 0.094

P(Phase Detected=PD5| Severity=Moderate)= 0.097

P(Phase Detected=PD5| Severity=Major)= 0.346

P(Phase Detected=PD6| Severity=Minor)= 0.063

P(Phase Detected=PD6| Severity=Moderate)= 0.065

P(Phase Detected=PD6| Severity=Major)= 0.365

P(Phase Attributed = PA1| Severity=Minor)= 0.125

P(Phase Attributed = PA1| Severity=Moderate)= 0.258

P(Phase Attributed = PA1| Severity=Major)= 0.231

P(Phase Attributed = PA2| Severity=Minor)= 0.312

P(Phase Attributed = PA2| Severity=Moderate)= 0.322

P(Phase Attributed = PA2| Severity=Major)= 0.577

P(Phase Attributed = PA3 | Severity=Minor)= 0.513

P(Phase Attributed = PA3| Severity=Moderate)= 0.419

P(Phase Attributed = PA3| Severity=Major)= 0.192

P(Phase Attributed = PA4 | Severity=Minor)= 0.013

P(Phase Attributed = PA4| Severity=Moderate)= 0.000

P(Phase Attributed = PA4| Severity=Major)= 0.000

P(Impact=Low | Severity=Minor)= 0.8

P(Impact=Low | Severity=Moderate)= 0.67647059

P(Impact=Low | Severity=Major)= 0.01818182

P(Impact=Medium | Severity=Minor)= 0.11428571

P(Impact= Medium | Severity=Moderate)= 0.23529412

P(Impact=Medium | Severity=Major)= 0.27272727

P(Impact=High | Severity=Minor)= 0.08571429

P(Impact= High | Severity=Moderate)= 0.08823529

P(Impact= High | Severity=Major)= 0.70909091

P(Weight=Low | Severity=Minor)= 0.82857143

P(Weight=Low | Severity=Moderate)= 0.67647059

P(Weight=Low | Severity=Major)= 0.05454545

P(Weight=Medium | Severity=Minor)= 0.08571429

P(Weight= Medium | Severity=Moderate)= 0.23529412

P(Weight=Medium | Severity=Major)= 0.25454545

P(Weight=High | Severity=Minor)= 0.08571429

P(Weight= High | Severity=Moderate)= 0.08823529

P(Weight= High | Severity=Major)= 0.69090909

Using this probability, to obtain P(X| Severity= Minor), P(X| Severity= Moderate) and P(X| Severity= Major) as refer from

table 4, the Naive Bayes generation process can be measured from the implementation of correctly classified instance and various measure also computed as shown in the table 5.

Precision is a measure of accuracy provided a specific class has been retrieved from predicting. It is defined by Precision = diagonal element/sum of relevant column

$$\text{Precision} = tp / (tp + fp) \quad (4)$$

where tp and fp are the numbers of true positive and false positive predictions of **p** for the considered class when the actual value is n as show in table V.

$$\text{F-measures} = 2*\text{precision}*\text{recall} / (\text{precision} + \text{recall}) \quad (5)$$

Where precision can be seen as a measure of exactness or quality, recall is a measure of completeness or quantity. Recall is nothing but the true positive rate for the class.

A Receiver Operating Characteristic (ROC) curve is a two-dimensional depiction of classifier performance. A common method is to calculate the area under the ROC curve, abbreviated AUC. Value of AUC will always be between 0 and 1. No realistic classifier should have an AUC less than 0.5.

Table 5: Various measures conducted on Naïve Bayes Classification generation process

S. No	Various Measures	Percentage
1	Correctly Classified Instances	92.1739%
2	Incorrectly Classified Instances	7.8261%
3	Kappa Statistic	0.8782
4	Mean absolute error	0.143
5	Root mean absolute error	0.248
6	Relative absolute error	33.1995%
7	Root Relative absolute error	53.4633%
8	Total Number of Instances	115

Table 6: Detailed Accuracy By Predicting SeverityUsing Naïve Bayes Algorithm

TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area AUC	Class
0.875	0.000	1.000	0.875	0.933	0.957	Minor
0.935	0.048	0.879	0.935	0.906	0.935	Moderate
0.942	0.079	0.907	0.942	0.925	0.964	Major

Table 7: Average Weighted Rate of Classification

Weighted Average	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area AUC
Naïve Bayes	0.922	0.049	0.925	0.922	0.922	0.954

Tables 6 and 7 represent accuracy and weighted average of true positive rate, false positive rate, precision, F-measure and ROC Area.

CONCLUSION

In this research work, it can be concluded that the performance of Naïve Bayes classification can be used to classify defects in the software development life cycle. By classifying, it is estimated that the probability of membership in each class gives a certain set of predictor variables. With respect to severity, defects are classified as major, moderate and minor. The classification is proved to be realistic based on AUC.

REFERENCES

[1] Araken M Santos, et.al., "A Comparative Analysis of Classification Methods to Multi-label Tasks in Different Application Domains", International Journal of Computer Information Systems and Industrial Management Applications

[2] Hua Wang e.al., "Multi-label Classification: Inconsistency, Ambiguity and Class Balanced KNN Classification".

[3] WEKA, University of Waikato, New Zealand, http://www.cs.waikato.ac.nz/ml/weka/

[4] A Survey of Educational Data mining and Research Trends, " Rajni Jindal, Malaya Dutta Borah", International journal of Database Management Systems.

[5] Z. N. Khan, "Scholastic achievement of higher secondary students in science stream", Journal of Social Sciences, Vol. 1, No. 2, pp. 84-87, 2005.

[6] J. Han and M. Kamber, "Data Mining: Concepts and Techniques," Morgan Kaufmann, 2000.

[7] Waiyamai,K "Improving Quality Graduate Student by Data Mining". Departement of Computer engineering. Faculty of Engineering. Kasetsart University, Bangkok Thailand. 2003

[8] Luan, J. "Chapter 2: Data Mining and Its Application in Higher Education. Knowledge Management – Building a Competitive Advantage in Higher Education." Serban, A. & Luan, J. (eds.) Jossey-Bass. 2002.

[9] Xinhua Zhang, et.al, "Bayesian Online Learning for Multi-label and Multi-variate Performance Measures", International Conference on Artificial Intelligence and Statistics.

[10] Ogor Emmanuel. N, " Student Academic Performance: Monitoring and Evaluation Using Data Mining Techniques". Fourth Congress of Electronics, Robotics and Automotive Mechanics. 2007. I EEE Computer Society.

[11] P.V.Praveen Sundar, "A Comparative Study for Predicting Student's Academic Performance Using Bayesian Network Classifiers", IOSR Journal of Engineering (IOSRJEN) e-ISSN: 2250-3021, p-ISSN: 2278-8719 Vol. 3, Issue 2 (Feb. 2013), ||V1|| PP 37-42.

[12] Syed Tahir Hijazi1 and S.M.M. Raza Naqvi, "Factors Affecting Students Performance: A Case of Private Colleges", Bangladesh e- Journal of Sociology, Volume 3. Number 1. January 2006.

[13] Mohammed M. Abu Tair, Alaa M. El-Halees, "Mining Educational Data to Improve Students' Performance: A Case Study", International Journal of Information and Communication Technology Research, Volume 2 No. 2, February 2012 ISSN 2223-4985.

[14] Md. Hedayetul Islam Shovon, Mahfuza Haque, "Prediction of Student Academic Performance by an Application of K-Means Clustering Algorithm", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 7, July 2012 ISSN: 2277 128X.

[15] Mohammed M. Abu Tair, Alaa M. El-Halees, "Mining Educational Data to Improve Students' Performance: A Case Study", International Journal of Information and Communication Technology Research, Volume 2 No. 2, February 2012 ISSN 2223-4985.