

# IJCIT

(Indonesian Journal on Computer and Information Technology)

Journal Homepage: <http://ejournal.bsi.ac.id/ejurnal/index.php/ijcit>

## Metoda *Distribution Based Balance* dan *Bagging C4.5* Untuk Prediksi Cacat *Software*

Nurul Ichsan

Sistem Informasi, Universitas Bina Sarana Informatika

nurul.nrc@bsi.ac.id

### ABSTRAK

Software berkualitas tinggi adalah software yang tidak ditemukan cacat (defect) baik selama proses pemeriksaan atau pengujian. Masalah utama dalam dataset software metrics adalah imbalance class yang membuat data menjadi tidak seimbang karena data yang cacat (kelas minoritas) jumlahnya lebih sedikit dibandingkan dengan data yang tidak cacat (kelas mayoritas), masalah ini dapat menurunkan kinerja klasifikasi. Pada Penelitian ini, untuk menangani masalah imbalance class dilakukan dengan integrasi *Distribution Based Balance* dan *Bagging* berbasis classifier C4.5 dan Naïve Bayes. Hasil penelitian menunjukkan bahwa model yang diusulkan mencapai akurasi dan AUC klasifikasi yang lebih tinggi. Rata-rata akurasi 93.84%, rata-rata nilai AUC 0.939 dengan nilai rata-rata peningkatan presentase AUC mencapai 0.34. Hasil kinerja Classifier C4.5 lebih baik dibandingkan Naïve Bayes dengan rata-rata akhir akurasi dari kinerja model klasifikasi 82.42% dan AUC 0.738 lebih baik dibandingkan kinerja algoritma pembanding Naïve Bayes dengan selisih akurasi 4.4% dan selisih AUC 0.023. Model yang diusulkan merupakan model terbaik dalam penelitian prediksi cacat software untuk menangani masalah imbalance class.

**Katakunci:** Software defect prediction, imbalance class, *Distribution Based Balance*, *Bagging*, C4.5

### ABSTRACTS

*Software high quality is software that is not found to be defective during the inspection or testing process. The main problem in the software metrics dataset is the imbalance class that makes the data become unbalanced because the data that is flawed (minority class) is less than the data that is not disabled (majority class), this problem can reduce classification performance. In this study, to deal with the problem of imbalance class is done by integrating *Distribution Based Balance* and *Bagging* based on C4.5 and Naïve Bayes classifiers. The results showed that the proposed model achieved a higher classification accuracy and AUC. The average accuracy is 93.84%, the average AUC value is 0.939 with an average increase in percentage of AUC reaching 0.34. Classifier C4.5 performance results are better than Naïve Bayes with the final average accuracy of the performance of the 82.42% classification model and 0.738 AUC is better than the performance of the Naïve Bayes comparison algorithm with a difference of 4.4% accuracy and AUC difference of 0.023. The proposed model is the best model in software defect prediction research to handle the problem of imbalance class..*

**Keywords:** *Software defect prediction, imbalance class, *Distribution Based Balance*, *Bagging*, C4.5*



## 1. PENDAHULUAN

*Software Defect Prediction* adalah salah satu topik penelitian paling penting dalam *software engineering* yang telah menarik banyak minat peneliti besar dari berbagai komunitas akademisi dan industri (Wu et al., 2018).

Sejumlah cacat yang ditemukan di akhir proyek secara sistematis menyebabkan penyelesaian proyek dapat melebihi jadwal yang sudah ditentukan (Lehtinen, Mäntylä, Vanhanen, Itkonen, & Lassenius, 2014). Secara tradisional, *software* berkualitas tinggi adalah *software* yang tidak ditemukan cacat selama pemeriksaan dan pengujian, serta dapat memberikan nilai kepada pengguna dan memenuhi harapan mereka (Laradji, Alshayeb, & Ghouti, 2014).

Masalah utama dalam dataset *software metrics* adalah *imbalance class* yang menjadikan data tidak seimbang karena data yang cacat (kelas minoritas) jumlahnya lebih sedikit dibandingkan dengan data yang tidak cacat (kelas mayoritas). Terdapat dua pendekatan yang dapat menangani *imbalance class* yaitu pendekatan level data (*sampling technique*) dan pendekatan level algoritma dengan teknik *ensemble learning* (Yap et al., 2014). Sedangkan klasifikasi adalah pendekatan yang paling populer untuk menangani *software defect prediction* (Lessmann, Member, Baesens, Mues, & Pietsch, 2008) dimana data penelitian yang didapat 77.46% dalam menyelesaikan masalah prediksi cacat *software* diselesaikan dengan menggunakan metode klasifikasi (Wahono, 2015).

*Software metrics* yang dikumpulkan selama pengembangan disimpan dalam bentuk dataset. 62 penelitian dari 208 penelitian dalam pengembangan model prediksi cacat *software* menggunakan dataset NASA (*National Aeronautics and Space Administration*) MDP (*Metrics Data Program*) repository sebagai *software metrics* (Hall et al., 2012).

Pada Penelitian ini untuk menangani masalah *imbalance class* dilakukan dengan menggunakan pendekatan level data (*sampling technique*) dan pendekatan level algoritma atau penggabungan dengan teknik *ensemble (ensemble learning)*. Pendekatan level data yang akan dilakukan dibagi dengan tiga pendekatan yang berbeda yaitu untuk teknik *oversampling* menggunakan SMOTE, teknik *hybrid sampling* penggabungan teknik *oversampling* dan

*undersampling* menggunakan *Class Balancer* dan *Distribution Based Balance*.

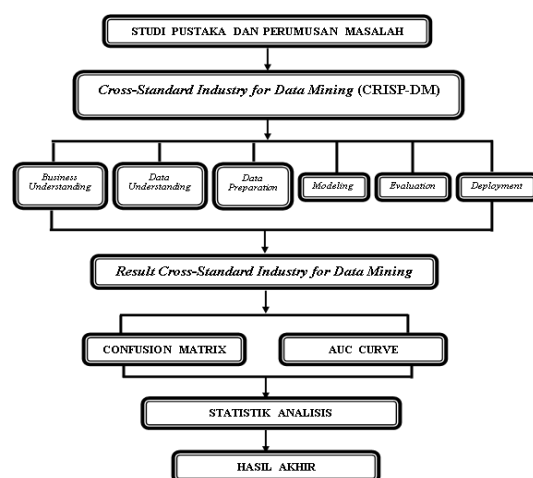
Dari ketiga metoda *sampling technique* yang ada, pendekatan level data *Distribution Based Balance* dipilih sebagai metoda utama yang diusulkan yang nantinya kinerja dari model ini akan dibandingkan dengan pendekatan level data SMOTE dan *Class Balancer*. Teknik *ensemble* yang digunakan adalah *Bagging*. Sedangkan algoritma klasifikasi yang digunakan dalam penelitian ini yaitu *classifier C4.5* dan satu *classifier* pembanding yaitu algoritma klasifikasi *Naive Bayes* untuk mengavaluasi dan membandingkan kinerja model *C4.5* secara keseluruhan dalam menangani *imbalance class*. Dataset yang digunakan dalam penelitian ini adalah dataset NASA MDP Repository sebagai *software metrics*.

Dengan melakukan penelitian tersebut diharapkan akan memperoleh nilai akurasi dan AUC (*Area Under ROC Curve*) yang signifikan dan dapat mengatasi permasalahan seperti yang dijelaskan sebelumnya yaitu mengenai ketidakseimbangan kelas (*imbalance class*) yang terdapat pada dataset *software metrics* dalam focus penelitian prediksi cacat *software*.

## 2. METODE PENELITIAN

Metode yang digunakan pada penelitian ini adalah eksperimen dengan menguji model yang diusulkan, evaluasi dan validasi. Penelitian ini bertujuan untuk mengurangi pengaruh ketidakseimbangan kelas pada model prediksi cacat *software*.

Tahapan-tahapan yang dilakukan dalam penelitian ini dapat dilihat pada Gambar 1.



Gambar 1. Tahapan Penelitian

Penelitian ini menggunakan pendekatan kuantitatif menggunakan data sekunder. Data yang digunakan dalam penelitian ini adalah dataset NASA (*National Aeronautics and Space Administration*) MDP (*Metrics Data Program*) repository. Dataset ini terbagi

menjadi 3 bagian yaitu dataset NASA MDP repository yang masih original, dataset yang sudah ditransformasi tahap pertama (D'), dan dataset yang sudah ditransformasi tahap kedua (D''). Sumber dataset NASA MDP repository ini diperoleh dari :

<https://github.com/klainfo/NASADefectDataset>

**Tabel 1.** Jumlah Data Penelitian

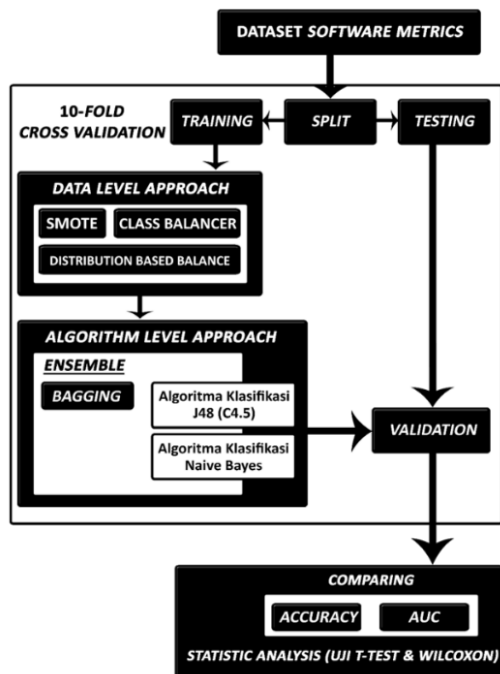
DATA NASA MDP REPOSITORY (D'')	12 DATASET											
	CM1	JM1	KC1	KC3	MC1	MC2	MW1	PC1	PC2	PC3	PC4	PC5
Attribut	38	22	22	40	39	40	38	38	37	38	38	39
Jumlah Modul	327	7720	1162	194	1952	124	250	679	722	1053	1270	1694
Modul Cacat	42	1612	294	36	36	44	25	55	16	130	176	458
Modul Tidak cacat	285	6108	868	158	1916	80	225	624	706	923	1094	1236
Presentase Cacat (%)	12.84	20.98	25.30	18.56	1.84	35.48	10	8.10	2.21	12.34	13.86	27.03

Metode penelitian yang digunakan pada eksperimen ini adalah model *Cross-Standard Industry for Data Mining* (CRISP-DM) yang terdiri dari 6 fase (Brown, 2014), yang dapat dilihat pada tabel 2.

**Tabel 2.** Penerapan *Cross-Standard Industry for Data Mining* (CRISP-DM)

No	<i>Cross-Standard Industry for Data Mining</i> (CRISP-DM)	Penerapan Dalam Penelitian
1	<i>Business Understanding</i>	Pada fase ini bisa disebut sebagai tahap pemahaman penelitian, menentukan tujuan proyek penelitian dalam perumusan mendefinisikan masalah <i>data mining</i> untuk penanganan ketidakseimbangan kelas pada <i>software defect prediction</i> .
2	<i>Data Understanding</i>	Pada fase ini penelitian lebih difokuskan pada tahap pemilihan dan pengumpulan data. Data yang akan digunakan merupakan data public yaitu 12 Dataset Nasa MDP Repository yang terdiri dari kelas (CM1, JM1, KC1, KC3, MC1, MC2, MW1, PC1 PC2, PC3, PC4, PC5).
3	<i>Data Preparation</i>	Fase pengolahan, Dataset NASA MDP yang akan digunakan dalam penelitian ini adalah yang telah ditransformasi (D''). Teknik validasi yang digunakan adalah <i>10-fold cross validation</i> , sehingga dataset dibagi menjadi 10, kemudian diambil satu bagian untuk dijadikan sebagai data uji dan yang lainnya dijadikan data latih.
4	<i>Modeling</i>	Pada tahap ini dilakukan pemrosesan <i>data training</i> . Model yang akan digunakan model algoritma klasifikasi yang di integrasikan dengan pendekatan level data dan <i>ensemble learning</i> untuk pengujiannya.
5	<i>Evaluation</i>	Pada tahap evaluasi, disebut tahap klasifikasi karena pada tahap ini akan ditentukan pengujian untuk akurasi. Tahap pengujiannya adalah melihat hasil akurasi dari berbagai model.
6	<i>Deployment</i>	-

Untuk mencari solusi masalah ketidakseimbangan kelas pada dataset software metrics, model yang diusulkan oleh penulis dalam penelitian ini terlihat pada gambar 2.

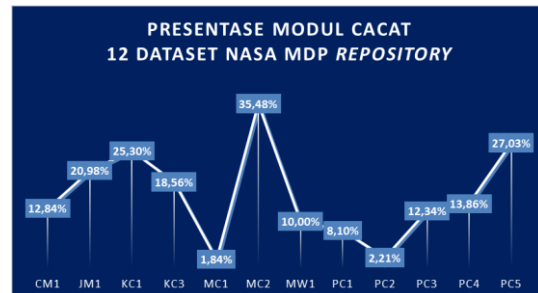


Gambar 2. Kerangka Kerja Model yang Diusulkan

### 3. HASIL DAN PEMBAHASAN

Eksperimen yang digunakan pada penelitian ini menggunakan desain eksperimen *Cross-Standard Industry for Data Mining* (CRISP-DM) seperti yang telah dijelaskan pada bab sebelumnya. Hasil dari penelitian ini bertujuan untuk mengurangi dan mengatasi permasalahan *imbalance class* pada model prediksi cacat *software*. Pada penelitian yang dilakukan, *tools* yang digunakan untuk pengujian algoritma adalah aplikasi WEKA (*Waikato Environment for Knowledge Analysis*), yang merupakan aplikasi *data mining open source* berbasis Java.

Spesifikasi jumlah data penelitian tersebut diambil dari 12 Dataset Nasa MDP *Repository* dengan memperhatikan jumlah atribut, modul, modul cacat (kelas minoritas), modul tidak cacat (kelas mayoritas) dan presentase modul cacat. Perbandingan jumlah presentase cacat sangat bervariasi dari 12 dataset *software metrics* yang akan diuji cobakan mulai dari 1.84% s/d 35.48% untuk tingkat presentase cacatnya.



Gambar 3. Presentase Modul Cacat

Dari hasil analisa Gambar 3. maka diperlukan model untuk menangani ketidakseimbangan kelas yang dapat menurunkan kinerja pengklasifikasi. Untuk menangani *imbalance class* teknik pengujiannya akan mencoba dengan model pengujian baru yaitu mengintegrasikan perbandingan kinerja model pendekatan level data (SMOTE, *Class Balancer*, *Distribution Based Balance*) dan pendekatan level algoritma (*ensemble*) *Bagging* berbasis *classifier* C4.5 dan Naïve Bayes untuk mencari kinerja model yang paling memberikan hasil yang terbaik dan signifikan baik untuk akurasi ataupun nilai AUC nya.

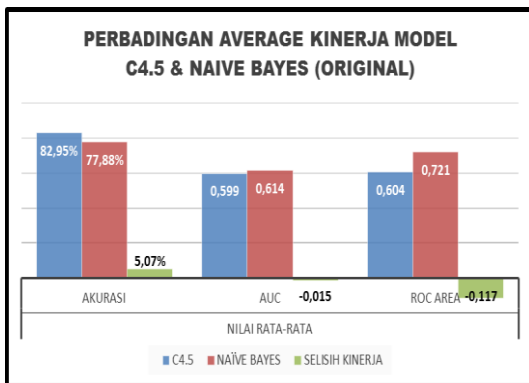
Sebelum diterapkan dengan berbagai model pengujian, terlebih dahulu dilakukan pengujiannya dengan kinerja asli dari 2 model *classifier* secara original yaitu C4.5 dan Naïve Bayes yang belum di integrasikan dengan model apapun dengan maksud dan tujuan untuk melihat sejauh mana perbedaan hasil kinerja model yang didapat sebelum dan sesudahnya dilakukan integrasi pengujian model yang diusulkan.

Hasil dari pengujian model C4.5 (Original) untuk akurasi, AUC, dan ROC Area terbaik terdapat pada dataset PC4.arff dengan nilai akurasi 86.93%, AUC 0.726 dan ROC Area 0.789 dengan tingkatan level klasifikasi berada pada level *Fair Classification*. Sedangkan hasil yang didapat dari *classifier* perbandingan yaitu Naïve Bayes (Original) menunjukkan hasil terbaik didapatkan pada dataset MW1.arff dengan nilai akurasi 81.60%, AUC 0.720 dan ROC Area 0.780 dengan tingkatan level klasifikasi "*Fair Classification*" dan rata-rata kinerja klasifikasi lebih baik dibandingkan dengan model C4.5 sebelum dilakukannya model integrasi. Jika mengacu pada Pedoman umum yang digunakan untuk klasifikasi terlihat pada tabel 3.

**Tabel 3.** Klasifikasi Akurasi

NO	NILAI AUC	KLASIFIKASI
1	0.90-1.00	<i>Excellent Classification</i>
2	0.80-0.90	<i>Good Classification</i>
3	0.70-0.80	<i>Fair Classification</i>
4	0.60-0.70	<i>Poor Classification</i>
5	0.50-0.60	<i>Failure</i>

Untuk melihat sejauh mana perbandingan kinerja rata-rata dari 2 *classifier* (C4.5 dan Naïve Bayes Original) bisa kita lihat pada Gambar 4.



**Gambar 4.** Perbandingan Kinerja Average Kinerja Model C4.5 & Naïve Bayes (Original)

Hasil yang didapat belum berarti menjadi sebuah kesimpulan bahwa model Naïve Bayes lebih baik dibandingkan model C4.5 yang diusulkan dalam penelitian ini, karena tahapan selanjutnya akan dilakukan pengujian model integrasi dengan pendekatan level data dan algoritma (*ensemble*). Sebelum menampilkan hasil yang didapat dari pengujian model integrasi *data and algorithm approach* menggunakan *classifier* C4.5 dan Naïve Bayes beserta hasil perbandingannya menggunakan

*tools* weka 3.8. Kinerja model yang diperoleh digunakan untuk membandingkan beberapa kinerja model yang diuji cobakan diantaranya terlihat pada tabel 4.

**Tabel 4.** Model Pengujian Algoritma

No	Model Pengujian Algoritma
1	<i>C4.5</i>
2	<i>Smote + Bagging + C4.5</i>
3	<i>Class Balancer + Bagging + C4.5</i>
4	<i>Distribution Based Balance + Bagging + C4.5</i>
5	<i>Naïve Bayes</i>
6	<i>Smote + Bagging + Naïve Bayes</i>
7	<i>Class Balancer + Bagging + Naïve Bayes</i>
8	<i>Distribution Based Balance + Bagging + Naïve Bayes</i>

Keterangan:

*Data Level Approach:*

SMOTE  
(*Synthetic Minority Over-sampling Technique*)  
*Class Balancer*  
*Distribution Based Balance*

*Algorithm Level Approach:*

*Bagging*

*Classification Algorithm:*

*C4.5 dan Naïve Bayes*

### 3.1 Original (C4.5 dan Naive Bayes)

Perbandingan hasil kinerja dan performa *classifier* antara C4.5 dan algoritma klasifikasi pembanding Naïve Bayes yang pertama yaitu pada model pengujian secara original tanpa integrasi. Hal ini tampak pada tabel 5.

**Tabel 5.** Perbandingan Kinerja & Performa C4.5 dan Naïve Bayes (Original)

Dataset	C4.5 (Ori)		Kinerja Klasifikasi	Naïve Bayes (Ori)		Kinerja Klasifikasi	Perbandingan		Kinerja Terbaik	
	Akurasi	AUC		Akurasi	AUC		Akurasi	AUC	C4.5	Naive Bayes
CM1	81.04%	0.536	Failure	81.35%	0.599	Failure	-0.31%	-0.063		
JM1	77.53%	0.582	Failure	78.91%	0.570	Failure	-1.38%	0.012		
KC1	74.18%	0.591	Failure	73.58%	0.599	Failure	0.60%	-0.008		
KC3	79.38%	0.616	Poor	78.87%	0.634	Poor	0.51%	-0.018		
MC1	97.90%	0.540	Failure	90.98%	0.600	Failure	6.92%	-0.06		
MC2	60.48%	0.576	Failure	70.16%	0.620	Poor	-9.68%	-0.044		NB
MW1	90.40%	0.662	Poor	81.60%	0.720	Fair	8.80%	-0.058		NB
PC1	91.46%	0.597	Failure	89.10%	0.667	Poor	2.36%	-0.07		NB
PC2	97.51%	0.499	Failure	90.30%	0.553	Failure	7.21%	-0.054		
PC3	84.71%	0.599	Failure	39.41%	0.602	Poor	45.30%	-0.003		NB
PC4	86.93%	0.726	Failure	85.35%	0.624	Poor	1.58%	0.102	C4.5	
PC5	73.91%	0.662	Poor	74.97%	0.582	Failure	-1.06%	0.08	C4.5	
Rata-rata	82.95%	0.599	Failure	77.88%	0.614	Poor	5.07%	-0.015		Naive Bayes

Rata-rata akurasi *Classifier C4.5* lebih baik dibandingkan Naïve Bayes dengan akurasi 82.95% dimana 5.07% lebih tinggi untuk peningkatan rata-ratanya dari 12 dataset yang ada dibandingkan Naïve Bayes, akan tetapi hasil ini tidak berpengaruh terhadap nilai rata-rata AUC. Untuk nilai AUC Naïve bayes lebih baik, rata-rata nilai AUC yang didapat 0.614 dan model ini berada pada level “*Poor Classification*”.

### 3.2 SMOTE (*Synthetic Minority Over-sampling Technique*)

Perbandingan pengujian kinerja model kedua yaitu pada *preprocessing* model SMOTE, yaitu membandingkan model SMOTE+Bagging+C4.5 dengan kinerja model pembanding yaitu SMOTE+Bagging+Naïve Bayes. Hasil perbandingannya dapat dilihat pada tabel 6.

**Tabel 6.** Perbandingan Kinerja & Performa SMOTE+Bagging+C4.5 dan SMOTE+Bagging+Naïve Bayes

Dataset	SMOTE+BG+C4.5		Kinerja Klasifikasi	SMOTE+BG+NB		Kinerja Klasifikasi	Perbandingan		Kinerja Terbaik	
	Akurasi	AUC		Akurasi	AUC		Akurasi	AUC	C4.5	NB
CM1	82.93%	0.705	<i>Fair</i>	74.25%	0.598	<i>Failure</i>	8.68%	0.107	<b>C4.5</b>	
JM1	80.72%	0.763	<i>Fair</i>	68.58%	0.572	<i>Failure</i>	12.14%	0.191	<b>C4.5</b>	
KC1	78.57%	0.768	<i>Fair</i>	65.80%	0.610	<i>Poor</i>	12.77%	0.158	<b>C4.5</b>	
KC3	83.04%	0.775	<i>Fair</i>	71.74%	0.643	<i>Poor</i>	11.30%	0.132	<b>C4.5</b>	
MC1	97.03%	0.604	<i>Poor</i>	89.34%	0.657	<i>Poor</i>	7.69%	-0.053		
MC2	79.17%	0.791	<i>Fair</i>	62.50%	0.638	<i>Poor</i>	16.67%	0.153	<b>C4.5</b>	
MW1	86.91%	0.733	<i>Fair</i>	<b>80.36%</b>	<b>0.748</b>	<i>Fair</i>	6.55%	-0.015		
PC1	89.92%	0.746	<i>Fair</i>	84.20%	0.664	<i>Poor</i>	5.72%	0.082	<b>C4.5</b>	
PC2	96.48%	0.609	<i>Poor</i>	89.43%	0.617	<i>Poor</i>	7.05%	-0.008		
PC3	85.04%	0.756	<i>Fair</i>	47.59%	0.632	<i>Poor</i>	37.45%	0.124	<b>C4.5</b>	
PC4	<b>88.45%</b>	<b>0.849</b>	<b>Good</b>	81.40%	0.693	<i>Poor</i>	7.05%	0.156	<b>C4.5</b>	
PC5	80.44%	0.797	<i>Fair</i>	64.36%	0.592	<i>Failure</i>	16.08%	0.205	<b>C4.5</b>	
Rata-rata	<b>85.73%</b>	<b>0.741</b>	<b>Fair</b>	<b>73.30%</b>	<b>0.639</b>	<b>Poor</b>	<b>12.43%</b>	<b>0.103</b>	<b>C4.5</b>	

Keterangan : BG=Bagging, NB=Naïve Bayes

Rata-rata akurasi *Classifier SMOTE+Bagging+C4.5* 85.73% dimana 12.43% lebih tinggi untuk peningkatan rata-ratanya dari 12 dataset yang ada dibandingkan model SMOTE+Bagging+Naïve Bayes, sedangkan untuk nilai rata-rata AUC SMOTE+Bagging+C4.5 masih lebih baik, rata-rata nilai AUC yang didapat 0.741 dimana 0.103 lebih tinggi untuk peningkatan rata-ratanya dari 12 dataset dan model ini berada pada level “*Fair Classification*” sedangkan model SMOTE+Bagging+Naïve berada pada level “*Poor Classification*” dengan

rata-rata AUC 0.639 dan rata-rata akurasi 73.30%.

### 3.3 Class Balancer

Perbandingan pengujian kinerja model ketiga yaitu pada *preprocessing* data *Class Balancer*, yaitu membandingkan model *Class Balancer+Bagging+C4.5* dengan kinerja model pembanding yaitu *Class Balancer+Bagging+Naïve Bayes*. Berikut adalah hasil perbandingannya.

**Tabel 7.** Perbandingan Kinerja & Performa *Class Balancer+Bagging+C4.5* dan *Class Balancer +Bagging+Naïve Bayes*

Dataset	CB+BG+C4.5		Kinerja Klasifikasi	CB+BG+NB		Kinerja Klasifikasi	Perbandingan		Kinerja Terbaik	
	Akurasi	AUC		Akurasi	AUC		Akurasi	AUC	C4.5	NB
CM1	58.84%	0.588	<i>Failure</i>	58.54%	0.585	<i>Failure</i>	0.30%	0.003		
JM1	63.42%	0.634	<i>Poor</i>	57.12%	0.571	<i>Failure</i>	6.30%	0.063	<b>C4.5</b>	
KC1	64.40%	0.644	<i>Poor</i>	60.43%	0.604	<i>Poor</i>	3.97%	0.04		
KC3	72.33%	0.723	<i>Fair</i>	62.80%	0.628	<i>Poor</i>	9.53%	0.095	<b>C4.5</b>	
MC1	60.41%	0.604	<i>Poor</i>	64.30%	0.643	<i>Poor</i>	-3.89%	-0.039		
MC2	67.95%	0.680	<i>Poor</i>	62.05%	0.620	<i>Poor</i>	5.90%	0.06		
MW1	65.78%	0.658	<i>Poor</i>	<b>70.89%</b>	<b>0.709</b>	<b>Fair</b>	-5.11%	-0.051		<b>NB</b>
PC1	72.76%	0.728	<i>Fair</i>	66.03%	0.660	<i>Poor</i>	6.73%	0.068	<b>C4.5</b>	
PC2	57.68%	0.577	<i>Failure</i>	57.05%	0.570	<i>Failure</i>	0.63%	0.007		
PC3	69.55%	0.695	<i>Poor</i>	63.46%	0.635	<i>Poor</i>	6.09%	0.06		
PC4	<b>83.35%</b>	<b>0.834</b>	<b>Good</b>	65.01%	0.650	<i>Poor</i>	18.34%	0.184	<b>C4.5</b>	
PC5	69.74%	0.697	<i>Poor</i>	58.58%	0.586	<i>Failure</i>	11.16%	0.111	<b>C4.5</b>	
Rata-rata	<b>67.18%</b>	<b>0.672</b>	<b>Poor</b>	<b>62.19%</b>	<b>0.622</b>	<b>Poor</b>	<b>5.00%</b>	<b>0.050</b>	<b>C4.5</b>	

Keterangan : CB=Class Balancer, BG=Bagging, NB=Naïve Bayes



Pada hasil perbandingan model ketiga ini berbeda dengan 2 hasil perbandingan sebelumnya dimana pada model ini hasil yang didapat *Classifier* C4.5 dan Naïve Bayes hampir merata dan tidak jauh berbeda, akan tetapi berdasarkan hasil analisa dari tabel diatas pada model pengujian ini model *Class Balancer+Bagging+C4.5* lebih baik dibandingkan dengan model pembanding yaitu *Class Balancer+Bagging+Naïve Bayes* dengan rata-rata nilai akurasi yang didapat 67.18% dan nilai rata-rata AUC 0.672.

Sedangkan untuk tingkatan level klasifikasi berada pada level yang sama yaitu “*Poor Classification*”

**3.4 Distribution Based Balance**

Perbandingan pengujian kinerja model ke-4 yaitu pada *preprocessing* model *Distribution Based Balance*, model *preprocessing* yang ini adalah yang penulis usulkan dalam penelitian ini. Pada model ini membandingkan model *Distribution Based Balance +Bagging+C4.5* dengan kinerja model pembanding yaitu *Distribution Based Balance +Bagging+Naïve Bayes*. Hasil perbandingannya dapat dilihat pada tabel 8.

**Tabel 8.** Perbandingan Kinerja & Performa *Distribution Based Balance + Bagging + C4.5* dan *Distribution Based Balance + Bagging + NB*

Dataset	DBB+BG+C4.5		Kinerja Klasifikasi	DBB+BG+NB		Kinerja Klasifikasi	Perbandingan		Kinerja Terbaik	
	Akurasi	AUC		Akurasi	AUC		Akurasi	AUC	C4.5	Naive Bayes
CM1	86.14%	0.861	Good	97.83%	0.978	Excellent	-11.69%	-0.117		NB
JM1	<b>98.69%</b>	<b>0.987</b>	<b>Excellent</b>	99.53%	0.995	Excellent	-0.84%	-0.008		
KC1	90.80%	0.908	Excellent	95.54%	0.955	Excellent	-4.74%	-0.047		
KC3	87.39%	0.874	Good	98.70%	0.987	Excellent	-11.31%	-0.113		NB
MC1	97.18%	0.972	Excellent	99.35%	0.993	Excellent	-2.17%	-0.021		
MC2	93.45%	0.935	Excellent	<b>100.00%</b>	<b>1.000</b>	<b>Excellent</b>	-6.55%	-0.065		
MW1	96.35%	0.964	Excellent	<b>100.00%</b>	<b>1.000</b>	<b>Excellent</b>	-3.65%	-0.036		
PC1	98.37%	0.984	Excellent	<b>100.00%</b>	<b>1.000</b>	<b>Excellent</b>	-1.63%	-0.016		
PC2	92.95%	0.93	Excellent	99.19%	0.992	Excellent	-6.24%	-0.062		
PC3	92.05%	0.920	Excellent	96.19%	0.962	Excellent	-4.14%	-0.042		
PC4	96.40%	0.964	Excellent	98.89%	0.989	Excellent	-2.49%	-0.025		
PC5	96.28%	0.963	Excellent	99.35%	0.993	Excellent	-3.07%	-0.03		
Rata-rata	<b>93.84%</b>	<b>0.939</b>	<b>Excellent</b>	<b>98.71%</b>	<b>0.987</b>	<b>Excellent</b>	<b>-4.88%</b>	<b>-0.049</b>		<b>NAÏVE BAYES</b>

Keterangan : DBB= *Distribution Based Balance*, BG=Bagging, NB=Naïve Bayes

Pada perbandingan pengujian model ke-4 ini hasil yang didapat kedua model *classifier* baik itu C4.5 ataupun Naïve Bayes sama-sama memberikan hasil yang terbaik dimana rata-rata nilai akurasinya berada pada 93.84% untuk C4.5 dan 98.71% untuk Naïve Bayes, sedangkan untuk AUC rata-rata C4.5 berada pada nilai AUC 0.939 sedangkan Naïve Bayes 0.987. Akan tetapi hasil tersebut belum menjadi hasil yang final bahwa naïve bayes lebih baik dibandingkan C4.5, pada tahap selanjutnya untuk hasil keseluruhan akan dilakukan evaluasi dari kedua model tersebut untuk melihat model mana yang lebih baik diantara C4.5 dan Naïve Bayes secara keseluruhan dari berbagai kinerja model pengujian yang telah dilakukan.

**3.5 Evaluasi Perbandingan Kinerja**

Hasil tabel 9 menunjukkan bahwa C4.5 *classifier* yang diusulkan penulis dalam penelitian ini memberikan hasil *classifier* terbaik

secara keseluruhan dari berbagai model pengujian dengan rata-rata akhir kinerja model klasifikasi dengan akurasi 82.42% dan AUC 0.738 lebih baik dibandingkan kinerja algoritma pembanding dengan selisih akurasi 4.4% dan selisih AUC 0.023.

**Tabel 9.** Perbandingan Evaluasi Kinerja C4.5 dan Naïve Bayes

Model	C4.5		Naive Bayes	
	Akurasi	AUC	Akurasi	AUC
ORI	82.59%	0.599	77.88%	0.614
SMOTE+BG	85.73%	0.741	73.30%	0.639
CB+BG	67.18%	0.672	62.19%	0.622
DBB+BG	93.84%	0.939	98.71%	0.987
Rata-rata	82.42%	0.738	78.02%	0.715

Keterangan:

ORI=Original

BG=Bagging

CB=Class Balancer

DBB=*Distribution Based Balance*

**3.6 Uji T-Test (Independent Sample T-Test)**

Hasil dari pengukuran kinerja, selanjutnya di analisis menggunakan uji t (*t-test*) untuk mengetahui model yang terbaik. Uji t dilakukan dengan membandingkan dua model dan mengukur *p-value*, jika *p-value* < nilai *alpha* (0.05), maka ada perbedaan yang signifikan antara dua model yang

dibandingkan. Sebaliknya, jika *p-value* > nilai *alpha*, maka tidak ada perbedaan yang signifikan.

Berikut adalah hasil perbandingan uji statistic dengan *independent sample t-test* menggunakan beberapa model yang diuji cobakan dari 12 dataset nasa MDP Repository.

**Tabel 10.** Hasil Perbandingan Uji Statistik Model C4.5 dan Naïve Bayes

NO	Model Pengujian	Mean	Sig. (2-tailed)	Keterangan
1	C4.5 (Original)	0.5988		
2	DBB+Bagging+C4.5	0.9385	0.000	Signifikan
3	SMOTE+Bagging+C4.5	0.7413	0.000	Signifikan
4	Class Balancer+Bagging+C4.5	0.6718	0.014	Signifikan
5	Naive Bayes (Original)	0.6142		
6	DBB+Bagging+Naive Bayes	0.9870	0.000	Signifikan
7	SMOTE+Bagging+Naive Bayes	0.6387	0.210	Tidak Signifikan
8	Class Balancer+Bagging+Naive Bayes	0.6218	0.670	Tidak Signifikan

Dari hasil Tabel 10. menunjukkan bahwa beberapa hasil yang didapat dari model C4.5 lebih baik dibandingkan model *Naïve Bayes*, dalam pengujian C4.5 semua model memberikan hasil yang signifikan, sedangkan pada model *naïve bayes* terdapat dua model yang tidak memberikan hasil yang signifikan,

sedangkan model yang diusulkan yaitu *Distribution Based Balance+Bagging+C4.5* berhasil memberikan hasil yang signifikan dengan rata-rata 0.9385 dan nilai *sig. (2-tailed)* 0.000 yang berarti nilai *sig. (2-tailed)* < dibandingkan nilai *alpha* (0.000 < 0.05).

**Tabel 11.** Uji t Model C4.5 dan Model *DBB+Bagging+C4.5*

		Group Statistics								
		Model	N	Mean	Std. Deviation	Std. Error Mean				
Hasil Pengujian Model	Model C4.5	12	.5988	.06230	.01798					
	Model DBB+BG+C4.5	12	.9385	.04153	.01199					
		Independent Sample Test								
		Levene's Test for Equality of Variances				t-tests for Equality of Means			95% Confidence Interval of The Difference	
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
Hasil Pengujian Model	Equal Variances Assumed	.719	.406	-15.716	22	.000	-.33967	.02161	-.38449	-.29484
	Equal Variances Not Assumed			-15.716	19.163	.000	-.33967	.02161	-.38488	-.29446

**3.7 Uji Wilcoxon**

Uji *Wilcoxon* dilakukan pada AUC dengan menggunakan metode statistik untuk menguji hipotesa pada Model C4.5 dengan model *Distribution Based Balance+Bagging+C4.5*, sedangkan untuk perbandingan hasil model C4.5 yang diusulkan dengan pendekatan C4.5 akan tetapi dengan level data yang berbeda akan dibandingkan dengan model

*SMOTE+Bagging+C4.5* dan *Class Balancer+Bagging+C4.5*. Dalam Uji *Wilcoxon* jika nilai *Asymp.Sig* <0.05, maka Hipotesis diterima sedangkan jika nilai *Asymp.Sig* >0.05, maka Hipotesis ditolak.

Hasil perbandingan uji statistic dengan uji *wilcoxon* menggunakan beberapa model yang diuji cobakan dari 12 dataset Nasa MDP Repository dapat dilihat pada tabel 12.



**Tabel 12.** Hasil Perbandingan Uji Statistik Model C4.5 dan Naïve Bayes

NO	Model Pengujian	Asymp. Sig. (2-tailed)	Keterangan
1	DBB+Bagging+C4.5	0.002	Signifikan
2	SMOTE+Bagging+C4.5	0.002	Signifikan
3	Class Balancer+Bagging+C4.5	0.003	Signifikan
4	DBB+Bagging+Naive Bayes	0.002	Signifikan
5	SMOTE+Bagging+Naive Bayes	0.006	Signifikan
6	Class Balancer+Bagging+Naive Bayes	0.328	Tidak Signifikan

Dari hasil tabel 12 menunjukkan bahwa beberapa hasil yang didapat dari model C4.5 lebih baik dibandingkan model *Naïve Bayes*, dalam pengujian C4.5 semua model memberikan hasil yang signifikan, sedangkan pada model *naïve bayes* terdapat satu model yang tidak memberikan hasil yang signifikan, Sedangkan model yang diusulkan yaitu

*Distribution Based Balance+Bagging+C4.5* berhasil memberikan hasil yang signifikan dengan nilai *Asymp.sig. (2-tailed)* 0.002 yang berarti nilai nilai *Asymp.Sig* model yang diusulkan < dibandingkan nilai nilai *Asymp.Sig* (0.002 < 0.05).

**Tabel 13.** Uji *Wicoxon* Model C4.5 dan Model *DBB+Bagging+C4.5*

		Ranks		
		N	Mean Rank	Sum of Ranks
DBB_BG_C4.5 - C4.5	Negative Ranks	0 <sup>a</sup>	.00	.00
	Positive Ranks	12 <sup>b</sup>	6.50	78.00
	Ties	0 <sup>c</sup>		
	Total	12		

- a. DBB\_BG\_C4.5 < C4.5
- b. DBB\_BG\_C4.5 > C4.5
- c. DBB\_BG\_C4.5 = C4.5

Test Statistics<sup>a</sup>

DBB_BG_C4.5 - C4.5	
Z	-3.059 <sup>b</sup>
Asymp. Sig. (2-tailed)	.002

- a. Wilcoxon Signed Ranks
- b. Based on Negative Ranks.

**4. KESIMPULAN**

Integrasi *Distribution Based Balance* dan Bagging diusulkan untuk meningkatkan kinerja *classifier* C4.5. Hasil dari beberapa percobaan integrasi model yang telah dilakukan dalam penelitian ini menghasilkan model yang diusulkan mencapai akurasi dan *Area Under ROC Curve (AUC)* klasifikasi yang lebih tinggi. Rata-rata nilai akurasi dari model *Distribution Based Balance+Bagging+C4.5* mencapai 93.84%, dan 0.939 untuk nilai *Area Under ROC Curve (AUC)*. Sedangkan grafik nilai peningkatan presentase AUC mencapai nilai 0.34.

Hasil berbeda terdapat pada model *classifier* pembanding yaitu *Distribution Based*

*Balance+Bagging+Naïve Bayes*. Rata-rata nilai akurasi dari model *Distribution Based Balance+Bagging+Naïve* berada pada nilai 98.71%, dan 0.998 untuk capaian rata-rata nilai *Area Under ROC Curve*. Sedangkan peningkatan presentase AUC lebih tinggi dari model yang diusulkan dengan capaian nilai 0.373. Kedua model ini jauh lebih baik diantara model *preprocessing* lainnya yaitu SMOTE dan *Class Balancer* yang diintegrasikan dengan Bagging berbasis *classifier* C4.5 dan Naïve Bayes.

Dari segi kinerja *Classifier* C4.5 lebih baik dibandingkan Naïve Bayes dengan rata-rata akhir kinerja model klasifikasi dengan akurasi 82.42% dan AUC 0.738 lebih baik dibandingkan kinerja algoritma pembanding Naïve Bayes

dengan selisih akurasi 4.4% dan selisih AUC 0.023. Oleh karena itu, dapat disimpulkan bahwa model yang diusulkan yaitu *Distribution Based Balance+Bagging+C4.5* merupakan model terbaik dalam penelitian prediksi cacat *software* untuk menangani *imbalance class*.

## 5. REFERENSI

- Chiş, M. (2008). Evolutionary Decision Trees and Software Metrics for Module Defects Identification. *Program*, 2(2), 25–29.
- Hall, T., Beecham, S., Bowes, D., Gray, D., & Counsell, S. (2012). A Systematic Literature Review on Fault Prediction Performance in Software Engineering. 38(6), 1276–1304.
- Laradji, I. H., Alshayeb, M., & Ghouti, L. (2014). Software defect prediction using ensemble learning on selected features. *INFORMATION AND SOFTWARE TECHNOLOGY*. <https://doi.org/10.1016/j.infsof.2014.07.005>
- Lehtinen, T. O. A., Mäntylä, M. V, Vanhanen, J., Itkonen, J., & Lassenius, C. (2014). Perceived causes of software project failures – An analysis of their relationships. *Information and Software Technology*, 56(6), 623–643. <https://doi.org/10.1016/j.infsof.2014.01.015>
- Lessmann, S., Member, S., Baesens, B., Mues, C., & Pietsch, S. (2008). *Benchmarking Classification Models for Software Defect Prediction : A Proposed Framework and Novel Findings*. 34(4), 485–496.
- Mantas, C. J., & Abellán, J. (2014). Expert Systems with Applications Credal-C4 . 5 : Decision tree based on imprecise probabilities to classify noisy data. *EXPERT SYSTEMS WITH APPLICATIONS*, 41(10), 4625–4637. <https://doi.org/10.1016/j.eswa.2014.01.017>
- Ming, C., Guoqing, W. U., Mengting, Y., & Hongyan, W. A. N. (2016). *Semi-supervised Software Defect Prediction Using Task-Driven Dictionary Learning* \*. 25(6). <https://doi.org/10.1049/cje.2016.08.034>
- Strate, J. D., & Laplante, P. A. (2013). A Literature Review of Research in Software Defect Reporting. 62(2), 444–454.
- Wahono, R. S. (2015). *A Systematic Literature Review of Software Defect Prediction : Research Trends , Datasets , Methods and Frameworks*. 1(1).
- Wu, F., Jing, X., Sun, Y., Sun, J., Huang, L., Cui, F., ... Prediction, A. S. D. (2018). *Cross-Project and Within-Project Semisupervised Software Defect Prediction : A Unified Approach*. 1–17.
- Yap, B. W., Rani, K. A., Aryani, H., Rahman, A., Fong, S., Khairudin, Z., & Abdullah, N. N. (2014). *An Application of Oversampling , Undersampling , Bagging and Boosting in Handling Imbalanced Datasets*. 13–23. <https://doi.org/10.1007/978-981-4585-18-7>