MEMPREDIKSI STATUS BERLANGGANAN KLIEN BANK PADA KAMPANYE PEMASARAN LANGSUNG DENGAN MENGGUNAKAN METODE KLASIFIKASI DENGAN ALGORITMA C5.0

PREDICTING BANK CLIENT'S SUBSCRIPTION STATUS IN DIRECT MARKETING CAMPAIGN USING CLASSIFICATION METHOD WITH C5.0 ALGORITHM

Yuni Dwiyanti [1], Anisa Herdiani, S.T., M.T[2], Shinta Yulia Puspitasari, S.T.,M.T.[3]

[1,2,3]Prodi S1 Teknik Informatika, Fakultas Teknik, Universitas Telkom

[1]elviemuna@gmail,com, [2]anisaherdiani@gmail.com, [3]shinta1907@gmail.com

Abstrak

Pada suatu bank, proses pemasaran bisa dilakukan dengan menghubungi klien satu per satu via telepon. Terkadang, petugas perlu menghubungi klien lebih dari satu kali untuk memastikan apakah klien tersebut bersedia menggunakan produk yang ditawarkan [1]. Tentu hal ini sangat tidak efisien dan juga membutuhkan biaya yang tidak sedikit. Proses pemasaran yang tidak efisien ini disebabkan karena petugas tidak mengetahui karakteristik klien yang berpotensi untuk berlangganan deposito berjangka. Agar proses pemasaran lebih efisien, perlu dilakukan pengklasifikasian klien bank berdasarkan status berlangganan deposito berjangka klien pada kampanye pemasaran langsung. Pada tugas akhir ini, metode klasifikasi dengan algoritma C5.0 akan digunakan untuk mengolah dataset klien bank agar diperoleh suatu model klasifikasi. Selain itu, dataset yang akan digunakan dalam penelitian ini memiliki permasalahan *imbalanced* class, yang mana perbandingan antara kelas *yes*:*no* adalah sebesar 1:8. Teknik *Synthetic Minority Over-Sampling Technique* (SMOTE) akan diterapkan guna menangani permasalahan *imbalanced class* pada dataset mentah. Dari penelitian ini, Model dengan nilai performansi terbaik diperoleh setelah dilakukan penanganan terhadap permasalahan *imbalance class* dengan teknik SMOTE dengan persentase duplikasi kelas minoritas sebesar 700% atau perbandingan jumlah kelas antara *yes*:*no* adalah kurang lebih 1:1. Setelah itu, pembentukan model klasifikasi dengan algoritma C5.0 dilakukan dengan membagi sampel data berdasarkan atribut yang memiliki nilai *information gain* tertinggi. Nilai performansi terbaik dari model klasifikasi yang terbentuk adalah sebesar 91.3% untuk *accuracy*, 90.16% untuk *precision*, 93.18% untuk *recall*, dan 91.65% untuk *f-measure* dengan nilai *error rate* pada proses pembentukan model klasifikasi sebesar 4%.

Kata Kunci: Klasifikasi, Algoritma C5.0, *Imbalanced Class*, SMOTE.

Abstract

In a bank, the marketing process can be done by contacting clients one by one by the phone. Sometimes, officers need to contact clients more than once to ascertain whether the client is willing to use the products offered [1]. *Of course this is very inefficient and also requires a lot of money. This inefficient marketing process is due to officers not knowing the characteristics of clients who have the potential to subscribe to term deposits. In order for the marketing process to be more efficient, it is necessary to classify bank clients based on their subscription status of the term deposits on direct marketing campaigns. In this final task, the classification method with C5.0 algorithm will be used to process the client bank dataset in order to obtain a classification model. In addition, the dataset will be used in this study has an imbalanced class issue, in which the comparison between the yes: no classes is 1: 8. The Synthetic Minority Over-Sampling Technique (SMOTE) technique will be applied to solve the imbalanced class issues in the raw dataset. From this research, Model with best performance value is obtained after handling of imbalanced class issue with SMOTE technique with percentage duplication of minority class equal to 700% or comparison of class number between yes: no is about 1: 1. After that, the formation of the classification model with C5.0 algorithm is done by dividing the sample data based on the attribute that has the highest information gain value.Best performance values of the classification model were 91.3% for accuracy, 90.16% for precision, 93.18% for recall, and 91.65% for f-measure value with error rate value in the process of forming a classification model is 4%.*

***Keywords**: Classification, C5.0 Algorithm, Imbalanced Class, SMOTE.*

1. Pendahuluan

Dalam proses kampanye pemasaran langsung deposito berjangka yang dilakukan via telepon, petugas terkadang harus menghubungi seorang klien lebih dari satu kali untuk memastikan apakah klien tersebut bersedia berlangganan deposito berjangka [1]. Tentu hal yang dilakukan ini tidak efisien dan membutuhkan biaya yang tidak sedikit. Proses pemasaran yang tidak efisien ini disebabkan karena petugas tidak mengetahui karakteristik klien yang berpotensi untuk berlangganan deposito berjangka.

Pada tugas akhir ini, klasifikasi klien bank berdasarkan status deposito berjangka akan dilakukan dengan menggunakan algoritma C5.0. Menurut penelitian yang dilakukan oleh [2], menunjukkan bahwa algoritma C5.0 memiliki nilai akurasi senilai 99.6% yang mana lebih tinggi dibandingkan dengan algoritma *Classification and Regression Tree* (CART) senilai 94.8%. Selain itu, menurut penelitian yang dilakukan oleh [3], algoritma C5.0 memiliki nilai akurasi senilai 87.72% yang mana merupakan nilai paling tinggi dibandingkan dengan algoritma CART dan *Chi-Squared Automatic Interaction Detection* (CHAID) senilai 87,27% dan 87,15% walaupun tidak ada perbedaan nilai akurasi yang signifikan diantara ketiganya. Selain itu, dataset yang akan digunakan dalam penelitian ini memiliki permasalahan *imbalanced* class, yang mana perbandingan antara kelas *yes*:*no* adalah sebesar 1:8. Hal yang perlu dilakukan untuk menangani permasalahan *imbalanced class* tersebut adalah dengan menerapkan teknik *Synthetic Minority Over-Sampling Technique* (SMOTE) pada tahap *preprocessing*. Pengujian terhadap model yang terbentuk nantinya dilakukan dengan menghitung performansi dengan mempertimbangkan nilai *accuracy, precision, recall,* dan *f-measure*. Pengujian dilakukan untuk mengetahui apakah model tersebut cukup baik untuk menyelesaikan kasus ini.

2. Dasar Teori

2.1. Pemasaran Bank Langsung

Ada dua pendekatan utama bagi perusahaan untuk mempromosikan produk dan/atau jasa: melalui kampanye massal, menargetkan massal tanpa membeda-bedakan atau pemasaran langsung, menargetkan satu set khusus kontak. Saat ini, dalam dunia global yang kompetitif, respon positif terhadap kampanye massal biasanya sangat rendah, kurang dari 1%. Secara alternatif, pemasaran langsung berfokus pada target yang dapat diasumsikan akan berpotensi pada spesifik produk atau jasa, membuat kampanye menjadi lebih menarik tetapi juga efisien. Namun demikian, pemasaran langsung memiliki beberapa kelemahan, misalnya dapat memicu sikap negative terhadap bank karena intrusi privasi. Ada kebutuhan yang harus dipenuhi untuk meningkatkan efisiensi: kontak yang dilakukan harus lebih rendah, tapi jumlah angka keberhasilan pemasaran harus tetap terjaga [4].

2.2. *Data Mining*

Teknologi *data mining* merupakan salah satu pengaplikasian yang komprehensif dari hal yang berkaitan dengan teknologi dengan mengandalkan teknologi *database*, analisis statistik, kecerdasan buatan, dan telah menunjukkan nilai komersial yang besar dan secara bertahap diterapkan untuk profesi lainnya di bidang ritel, asuransi, telekomunikasi, dan industry penggunaan sumber daya [7]. *Data mining* juga didefinisikan sebagai proses mengekstraksi informasi dan pola yang bermanfaat dari kumpulan data yang sangat banyak. Proses ini juga dikenal sebagai proses penemuan pengetahuan, penambangan pengetahuan dari sekumpulan data, ekstraksi pengetahuan atau analisis data/pola. Tujuan dari teknik ini adalah untuk menemukan suatu pola yang sebelumnya tidak diketahui [8].

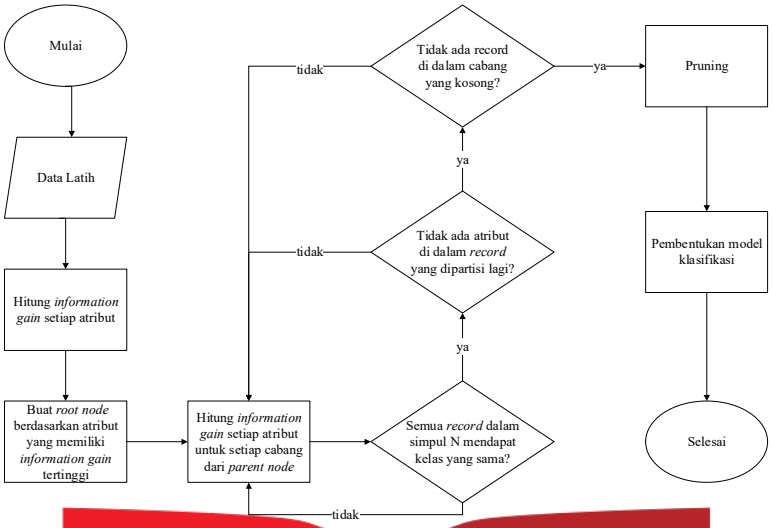
Data mining merupakan suatu komponen area yang menarik dari *machine learning* dan komputasi yang mampu beradaptasi. *Data mining* juga dapat didefinisikan sebagai analisis (besar) dari sekumpulan data pengamatan untuk menemukan hubungan yang tidak terduga dan menyimpulkan data dengan cara baru yang mana keduanya mudah dimengerti dan juga berguna bagi si pemilik data [7].

2.3. Klasifikasi

Klasifikasi adalah proses menemukan suatu model (atau fungsi) yang menggambarkan dan membedakan kelas data atau konsep, dengan tujuan agar model tersebut dapat digunakan untuk memprediksi kelas dari objek yang label kelasnya belum diketahui [9]. Klasifikasi digunakan secara luas pada *data mining* untuk mengelompokkan data ke dalam beragam kelas. Klasifikasi merupakan teknik data mining yang digunakan untuk memprediksi keanggotaan dari instansi data [10]. Klasifikasi juga didefinisikan sebagai proses generalisasi kumpulan data berdasar instansi yang berbeda-beda [11]. Klasifikasi terdiri dari dua proses yaitu *model construction* dan *model usage*. *Model construction* adalah proses mendeskripsikan sekumpulan kelas-kelas yang sebelumnya telah diketahui. Biasanya model yang terbentuk berupa *rule* klasifikasi, pohon keputusan, atau rumus matematis. Sedangkan *model usage* merupakan proses mengelompokkan data di masa mendatang atau yang belum diketahui [10].

2.4. C5.0

Algoritma C5.0 merupakan pengembangan dari algoritma C4.5 yang mana juga merupakan pengembangan dari algoritma ID3. Model C5.0 dapat membagi sampel data berdasarkan atribut yang memiliki nilai *information gain* tertinggi [9]. *Sample subset* yang diperoleh dari percabangan yang terbentuk akan dipecah lagi setelahnya. Prosesnya akan terus berlanjut sampai *sample subset* tidak dapat lagi dibagi dan biasanya menunjuk pada *field* lain. Pada akhirnya, *sample subset* yang tidak memiliki kontribusi yang besar bagi model akan ditolak. Proses pembentukan pohon keputusan dengan menggunakan algoritma C5.0 dapat dilihat pada Gambar 1.



Gambar 1. Flowchart Algoritma C5.0

Atribut dengan nilai *information gain* tertinggi akan terpilih sebagai *parent* bagi *node* selanjutnya. Algoritma ini membentuk pohon keputusan dengan cara pembagian dan menguasai sampel secara rekursif dari atas ke bawah. Algoritma ini dimulai dengan semua data yang dijadikan akar dari pohon keputusan sedangkan atribut yang dipilih akan menjadi pembagi bagi sampel tersebut. Pada pembentukan model klasifikasi, untuk mengklasifikasikan sampel yang digunakan maka diperlukan informasi dengan menggunakan formula 1. Kemudian lakukan perhitungan untuk mendapatkan informasi nilai subset dari suatu atribut A dengan menggunakan formula 2. Selanjutnya untuk mengetahui nilai *information gain* dari atribut A, maka digunakan formula 3 [12].

$$I(s_1, s_2, \dots, s_m) = -\sum_{i=1}^{m} p_1 \log_2(p_1) \quad (1)$$

$$E(A) = \sum_{j=1}^{y} \frac{s_{1j}+\cdots+s_{mj}}{s} I(s_1, s_2, \dots, s_m) \quad (2)$$

$$Gain(A) = I(s_1, s_2, \dots, s_m) - E(A) \quad (3)$$

Dimana:

S : jumlah data sampel.

s_i : jumlah sampel pada S dalam kelas *Ci.*

p_i : proporsi kelas dan diestimasikan dengan $\frac{s_i}{s}$.

$E(A)$: nilai subset dari atribut A.

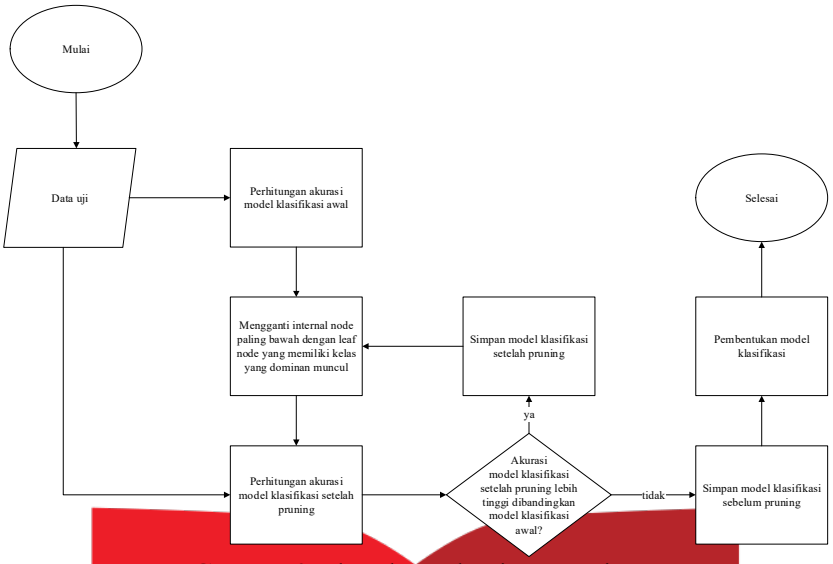
S_j : sampel pada S yang bernilai A_j.

S_{ij} : jumlah sampel pada kelas C_i dalam sebuah subset S_j.

$\frac{s_{1j}+\cdots+s_{mj}}{s}$: jumlah subset j yang dibagi dengan jumlah sampel pada S.

$Gain(A)$: nilai *information gain* dari atribut A.

Ulangi terus langkah diatas yaitu menghitung nilai tiap atribut berdasarkan nilai *information gain* yang tertinggi hingga semua *record* terpartisi. Proses dari *decision tree* ini akan berhenti jika semua *record* dalam simpul N mendapat kelas yang sama, tidak ada atribut di dalam *record* yang dipartisi lagi, dan tidak ada record di dalam cabang yang kosong [13].



Gambar 2. Flowchart Algoritma Pruning

Hasil pohon keputusan C5.0 dapat dipangkas atau terdapat *pruning* (pemangkasan) [12]. Lakukan *pruning* dengan mengganti node paling akhir sebelum *leaf* dengan kelas yang paling dominan dari node tersebut. Lakukan kembali perhitungan nilai akurasi. Jika nilai akurasi yang diperoleh setelah proses *pruning* lebih besar dari nilai akurasi sebelumnya, maka pohon keputusan akan berganti dengan pohon keputusan yang terbentuk setelah dilakukan *pruning.* Tetapi jika nilai akurasi yang diperoleh lebih kecil, maka tidak ada perubahan yang terjadi pada pohon keputusan. Algoritma dari proses *pruning* dapat dilihat pada Gambar 2.

2.5. Imbalanced Class dan Smote

Imbalanced class adalah sebuah masalah pada kasus klasifikasi yang mana kelas-kelas yang ada tidak direpresentasikan secara seimbang [14]. Terdapat beberapa teknik untuk mengatasi permasalah *imbalanced* class diantaranya adalah *Random Oversampling, Random Undersampling,* dan SMOTE (*Synthetic Minority Over-Sampling Technique*). Teknik *oversampling* dapat menyebabkan *overfitting* untuk membuat duplikat jumlah yang sama dengan sampel minoritas. Sementara itu, teknik *undersampling* dapat membuang sebagian besar potensi sampel yang berguna. Penggunaan teknik SMOTE menghasilkan hasil yang baik dan cara yang efektif untuk menangani ketidakseimbangan kelas yang mengalami *overfitting* pada teknik *oversampling* untuk memproses kelas minoritas [15]. Metode SMOTE bekerja dengan mencari *k nearest neighbors* (yaitu ketetanggaan data) untuk setiap data di kelas minoritas, setelah itu dibuatlah *synthetic data* sebanyak persentase duplikasi yang diinginkan antara data minoritas dan *k nearest neighbor* yang dipilih secara random [16].

2.6. Pengujian

Sebuah cara yang bersih dan tidak ambigu dalam merepresentasikan hasil prediksi dari sebuah *classifier* adalah dengan menggunakan sebuah *confusion matrix*. Untuk kasus *binary classification*, table *confusion matrix* terdiri dari 2 baris dan 2 kolom. Di bagian atas merupakan label kelas yang diobservasi dan di bagian sisi adalah label kelas yang diprediksi. Setiap sel berisi jumlah prediksi yang dihasikan oleh *classifier* yang sesuai dengan sel tersebut [17]. Tabel *confusion matrix* dapat dilihat pada Tabel 1

.

Tabel 1. *Confusion Matrix* [17]

		Predicted Value	
		Negatives	Positives
Actual Value	Negatives	TN	FP
	Positives	FN	TP

Setelah memperoleh sebuah model yang diyakini dapat membuat prediksi yang kuat, perlu dilakukan pengujian apakah model tersebut cukup baik untuk menyelesaikan kasus tersebut. Pengujian dengan mempertimbangkan nilai akurasi dari klasifikasi saja biasanya bukan merupakan informasi yang cukup untuk menyimpulkan performansi suatu *classifier*. Akurasi bukan merupakan pengukuran yang dapat digunakan ketika

bekerja dengan menggunakan dataset yang *imbalanced.* Terdapat beberapa pengukuran performansi yang diciptakan untuk *imbalanced class* diantaranya adalah *precision, recall,* dan *f-measure* [14]. *Precision*, *recall*, dan *f-measure* juga akan digunakan sebagai pengukuran performansi untuk mengevaluasi model yang dihasilkan untuk kasus *binary classification*. Tabel 2-2 menjelaskan fokus pengujian dari masing-masing pengukuran performansi. Nilai *accuracy* diperoleh dengan melakukan perhitungan menggunakan formula 2.4, nilai *precision* menggunakan formula 2.5, nilai *recall* menggunakan formula 2.6, dan nilai *f-measure* menggunakan formula 2.7 [16] [17] [18]:

$$Accuracy = \frac{tp+tn}{tp+fn+fp+t} \quad (4)$$

$$Precision = \frac{tp}{tp+fp} \quad (5)$$

$$Recall = \frac{tp}{tp+fn} \quad (6)$$

$$F\text{-}Measure = 2 \times \left(\frac{precision \times recall}{precision+reca}\right) \quad (7)$$

Tabel 2. Fokus Pengujian Pengukuran Performansi [19] [16]

Ukuran	Fokus Pengujian
Accuracy	Efektifitas keseluruhan dari sebuah *classifier*
Precision	Seberapa baik ketepatan model dapat memprediksi suatu kelas tertentu.
Recall	Seberapa besar *coverage* suatu model dalam memprediksi suatu kelas tertentu.
F-Measure	Menetukan hasil prediksi yang paling baik, merupakan kombinasi dari nilai *recall* dan *precision*.

3. Pembahasan

3.1. Perancangan Sistem

3.1.1. Pengumpulan Data

Data yang akan digunakan merupakan hasil dokumentasi marketing bank dari kampanye pemasaran langsung pada institusi perbankan di Portugal dari bulan Mei 2008 sampai November 2010 [4] [5] [6]. Produk yang dipasarkan pada kampanye pemasaran langsung ini berupa deposito berjangka (*term deposit*). Dataset ini berisi 4.521 record klien yang merepresentasikan kelas *yes* dan *no*.

3.1.2. Pembelajaran Karakteristik Data

Setelah diperoleh data yang dibutuhkan, tahap selanjutnya adalah mengamati karakteristik data. Hal-hal yang perlu diperhatikan adalah kualitas dan tipe atribut pada data. Misalnya, pada dataset yang dimiliki terdapat 4.521 *record* yang mana merepresentasikan kelas *yes* sebanyak 521 *record* dan kelas *no* sebanyak 4.000 *record* sehingga dapat dikatakan bahwa dataset memiliki karakteristik *imbalanced class* dengan rasio antara kelas *no* dan *yes* sebesar 1:8. Selain itu, pada dataset tidak ditemukan adanya *missing value* pada data. Setelah diketahui karakteristik dan tipe atribut pada data, barulah dapat ditentukan penanganan dan penyelesaian yang tepat yang nantinya akan dilakukan pada tahap *preprocessing*.

3.1.3. Preprocessing

Sebelum dataset diolah di dalam sistem untuk dibentuk pohon keputusannya, langkah sebelumnya adalah melakukan *preprocessing* terhadap data mentah. *Preprocessing* dilakukan di luar sistem dengan menggunakan *software* Weka 3.8.0. Pada tahap *preprocessing* ini dilakukan penanganan terhadap permasalahan *imbalanced class* pada dataset. Untuk menangani masalah *imbalanced class*, digunakan teknik SMOTE (*Synthetic Minority Over-Sampling Technique*) dengan jumlah *nearest neighbour*-nya adalah lima [21] [16].

Pembentukan data sintetis pada teknik SMOTE dengan berdasarkan jenis data terbagi menjadi 2 jenis, yakni pembentukan data sintetis untuk data bernilai numerik dan data bernilai kategoris [16].

Berikut tahapan teknik SMOTE dalam membentuk data sintetis yang bernilai numerik [16]:

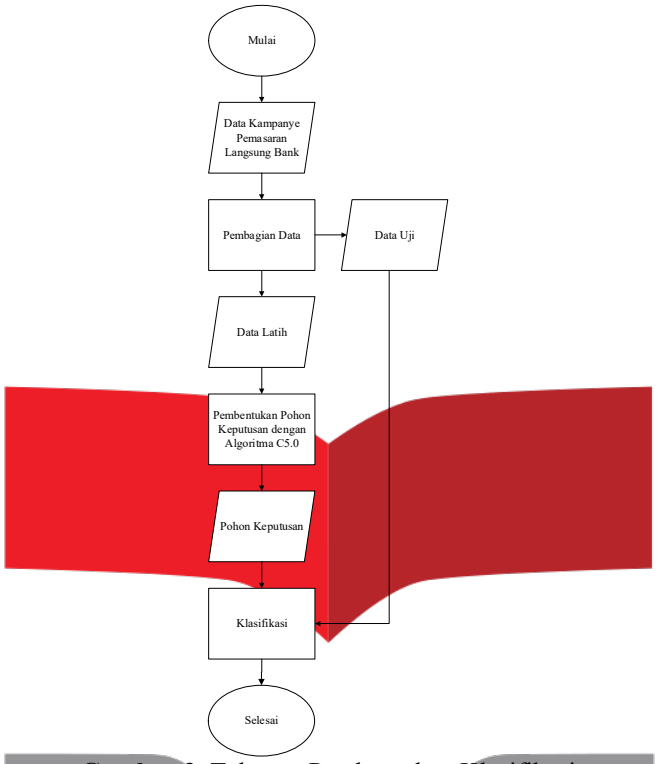
a. Hitung perbedaan nilai setiap atribut antara *minority sample* dengan salah satu dari *k nearest neighbor*-nya.
b. Pilih nilai secara random antara 0 sampai 1 untuk kemudian dikalikan dengan nilai yang diperoleh pada poin a.
c. Nilai sintetis baru diperoleh dengan menjumlahkan nilai *minority sample* dan nilai yang diperoleh pada poin b.

Berikut tahapan teknik SMOTE dalam membentuk data sintetis yang bernilai kategoris [16]:

a. Diambil voting antara minority sample dan k nearest neighbor-nya. Jika tidak ada majority class, maka pilihlah nilai atribut pada minority sample tersebut.
b. Nilai tersebut menjadi data sintetis baru.

3.1.4. Pembangununan Model Klasifikasi

Pembangunan model klasifikasi yang dilakukan untuk memprediksi status berlangganan terhadap klien bank diterapkan dengan berdasarkan algoritma C5.0. Tahapan dalam membangun model klasifikasi dapat dilihat pada Gambar .



Gambar 3. Tahapan Pembentukan Klasifikasi

a. Pembagian Data
Untuk membangun model klasifikasi, sebelumnya dilakukan pembagian datasett. Dataset dibagi ke dalam dua jenis, yaitu data latih dan data uji. Masing-masing persentase untuk data uji dan data latih adalah 80% dan 20% [13] [22] [21] [18] dan dipilih secara random. Data latih digunakan untuk membangun model klasifikasi, sedangkan data uji digunakan untuk mengevaluasi model klasifikasi yang terbentuk sebelumnya menggunakan data latih.

b. Pembentukan Pohon Keputusan dengan Algoritma C5.0
Data latih yang diperoleh dari pembagian data pada dataset akan digunakan untuk membentuk model klasifikasinya. Model klasifikasi yang dimaksud berupa pohon keputusan. Pohon keputusan akan dibentuk dengan menggunakan algoritma C5.0. Tahapan pembentukan pohon keputusan dengan menggunakan algoritma C5.0 telah dijelaskan pada sub bab 2.4.

c. Perhitungan Performansi
Setelah model klasifikasi berhasil terbentuk, selanjutnya akan dilakukan perhitungan nilai performansi dengan mempertimbangkan nilai *accuracy, precision, recall,* dan *f-measure*. Perhitungan nilai performansi dilakukan berdasarkan data uji yang sebelumnya diperoleh dari pembagian data pada dataset. Nilai performansi diperoleh dengan menggunakan persamaan yang sebelumnya telah dijelaskan pada subbab 2.7.

3.2. Skenario Pengujian

Berikut adalah skenario yang dilakukan dalam pengujian:

1. Skenario 1, pada skenario 1 dilakukan pengujian untuk mengetahui nilai performansi yang diperoleh dari pohon keputusan yang terbentuk dengan algoritma C5.0 dengan menggunakan dataset tanpa melalui tahap *preprocessing.* Nilai performansi dihitung dengan mempertimbangkan nilai *accuracy*, *precision*, *recall*, dan *f-measure.*
2. Skenario 2, pada skenario 2 dilakukan pengujian untuk mengetahui nilai performansi yang diperoleh dari pohon keputusan yang terbentuk dengan algoritma C5.0 dengan menggunakan dataset yang telah melalui tahap *preprocessing* dengan persentase duplikasi kelas minoritas sebesar 100%. Jumlah record yang merepresentasikan kelas *yes* sebanyak 1042 dan kelas *no* sebanyak 4000. Perbandingan antara kelas *yes*:*no* adalah 1:4. Nilai performansi dihitung dengan mempertimbangkan nilai *accuracy*, *precision*, *recall*, dan *f-measure.*
3. Skenario 3, pada skenario 3 dilakukan pengujian untuk mengetahui nilai performansi yang diperoleh dari pohon keputusan yang terbentuk dengan algoritma C5.0 dengan menggunakan dataset yang telah melalui tahap *preprocessing* dengan persentase duplikasi kelas minoritas sebesar 300%. Jumlah record yang merepresentasikan kelas *yes* sebanyak 2084 dan kelas *no* sebanyak 4000. Perbandingan antara kelas *yes*:*no*

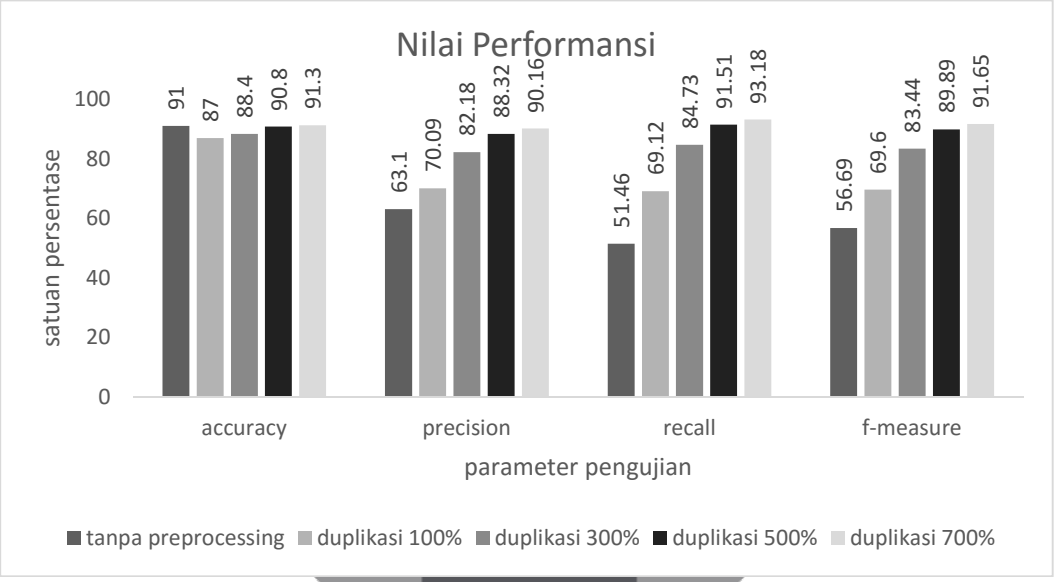
adalah 1:2. Nilai performansi dihitung dengan mempertimbangkan nilai *accuracy*, *precision*, *recall*, dan *f-measure.*

4. Skenario 4, pada skenario 4 dilakukan pengujian untuk mengetahui nilai performansi yang diperoleh dari pohon keputusan yang terbentuk dengan algoritma C5.0 dengan menggunakan dataset yang telah melalui tahap *preprocessing* dengan persentase duplikasi kelas minoritas sebesar 500%. Jumlah record yang merepresentasikan kelas *yes* sebanyak 3126 dan kelas *no* sebanyak 4000. Perbandingan antara kelas *yes*:*no* adalah 3:4. Nilai performansi dihitung dengan mempertimbangkan nilai *accuracy*, *precision*, *recall*, dan *f-measure.*
5. Skenario 5, pada skenario 5 dilakukan pengujian untuk mengetahui nilai performansi yang diperoleh dari pohon keputusan yang terbentuk dengan algoritma C5.0 dengan menggunakan dataset yang telah melalui tahap *preprocessing* dengan persentase duplikasi kelas minoritas sebesar 700%. Jumlah record yang merepresentasikan kelas *yes* sebanyak 4168 dan kelas *no* sebanyak 4000. Perbandingan antara kelas *yes*:*no* adalah 1:1. Nilai performansi dihitung dengan mempertimbangkan nilai *accuracy*, *precision*, *recall*, dan *f-measure.*

3.3. Pengujian

Karena pada dataset mentah terdapat permasalahan *imbalanced class* dengan perbandingan antara kelas *yes* dan *no* adalah sebesar 1:8, maka pada tahap *preprocessing* dilakukan penyelesaian terhadap permasalahan tersebut dengan melakukan duplikasi kelas minoritas menggunakan teknik SMOTE. Persentase duplikasi kelas minoritas yang diterapkan diantaranya adalah 100%, 300%, 500%, dan 700% [16].

Dalam menghitung nilai performansi dari model klasifikasi yang terbentuk dengan dataset yang memiliki permasalahan *imbalance class*, perlu dilakukan pengukuran nilai *accuracy, precision, recall*, dan *f-measure*. Nilai *precision*, *recall*, dan *f-measure* diperoleh berdasarkan prediksi terhadap kelas *yes*. Nilai *f-measure* menjadi pertimbangan untuk menentukan hasil prediksi yang paling baik [15]. Hasil perhitungan nilai performansi dari model klasifikasi yang terbentuk dapat dilihat pada Gambar 4.



Gambar 4. Grafik Nilai Performansi

Dari Gambar 4, nilai *f-measure* paling rendah adalah sebesar 56.69% yang terjadi saat pembentukan model klasifikasi dengan menggunakan dataset yang tidak melalui tahap *preprocessing*. Walaupun pada pembentukan model klasifikasi dengan dataset yang tidak melalui tahap *preprocessing* memiliki nilai akurasi yang cukup tinggi, namun model klasifikasi ini tidak dapat dikatakan paling baik diantara model klasifikasi lainnya karena nilai *f-measure* yang diperoleh merupakan nilai *f-measure* yang paling rendah diantara model klasifikasi lainnya. Nilai akurasi tinggi yang diperoleh pada proses pembentukan model klasifikasi dengan menggunakan dataset yang tidak melalui tahap *preprocessing* disebabkan karena baik data latih maupun data uji mayoritas terdiri dari kelas berlabel *no* (kelas mayoritas) sehingga model klasifikasi yang terbentuk dan nilai akurasi yang diperoleh sebagian besar hanya merefleksikan kelas berlabel *no* (kelas mayoritas). Sedangkan, ketika model klasifikasi diminta untuk memprediksi kelas berlabel *yes*, model akan melakukan kesalahan prediksi dengan jumlah yang cukup besar sehingga berdampak pada nilai *precision, recall,* dan *f-measure*. Pada proses pembentukan model klasifikasi dengan menggunakan dataset yang tidak melalui tahap *preprocessing*, dari 103 *record* untuk kelas *yes* yang harus diprediksi, model melakukan kesalahan prediksi sebanyak 50 *record*. Hal ini disebabkan karena pada saat proses pembentukan model klasifikasinya, nilai *error rate* dari model klasifikasi ini merupakan yang paling tinggi yaitu sebesar 8.2% sehingga akan mempengaruhi nilai performansi ketika model klasifikasi ini digunakan dalam mengklasifikasikan data uji. *Confusion matrix* dari model klasifikasi ini dapat dilihat pada Tabel 3. Sehingga dapat dikatakan bahwa model klasifikasi yang

terbentuk dengan menggunakan dataset yang tidak melalui tahap *preprocessing* kurang tepat untuk memprediksi kelas *yes* dan area *coverage* dari model tersebut paling kecil dibandingkan dengan model klasifikasi lainnya.

Tabel 3. Confusion Matrix Model Klasifikasi Tanpa Preprocessing

Actual	Predicted		Jumlah
	no	*yes*	
no	770	31	801
yes	50	53	103

Dari Gambar 4, juga dapat dilihat bahwa nilai *f-measure* tertinggi diperoleh ketika pembentukan model klasifikasi dengan menggunakan dataset dengan persentase duplikasi kelas minoritas sebesar 700% atau perbandingan antara kelas *yes*:*no* kurang lebih sebesar 1:1. Model klasifikasi ini memiliki nilai *f-measure* paling tinggi diantara model klasifikasi lainnya karena pada saat melakukan prediksi terhadap kelas berlabel *yes*¸ model klasifikasi ini melakukan kesalahan prediksi dengan jumlah yang paling kecil diantara model klasifikasi lainnya. Sehingga, jumlah kesalahan prediksi yang sedikit tersebut berpengaruh terhadap nilai *precision, recall*, dan *f-measure* dari model ini dan menyebabkan nilai yang diperoleh paling tinggi dibandingkan dengan model klasifikasi lainnya. Hal ini disebabkan karena pada saat proses pembentukan model klasifikasinya, nilai *error rate* dari model klasifikasi ini merupakan yang paling rendah yaitu sebesar 4% sehingga akan mempengaruhi nilai performansi ketika model klasifikasi ini digunakan dalam mengklasifikasikan data uji. *Confusion matrix* dari model klasifikasi yang terbentuk dengan menggunakan dataset dengan persentase duplikasi kelas minoritas sebesar 700% dapat dilihat pada Tabel 4.

Tabel 4. Confusion Matrix Model Klasifikasi Duplikasi 700%

Actual	Predicted		Jumlah
	no	*yes*	
no	713	85	798
yes	57	779	836

Dapat kita lihat dari gambar 4, peningkatan nilai performansi dari model klasifikasi dengan menggunakan dataset dengan persentase duplikasi 700% mengalami peningkatan yang tidak terlalu signifikan. Hal ini disebabkan karena semakin besar duplikasi kelas minoritas, maka data yang akan digunakan dalam membentuk model klasifikasi juga menjadi kurang bervariasi. Sehingga, peningkatan nilai antara model klasifikasi dengan menggunakan dataset dengan duplikasi 700% kurang signifikan. Berdasarkan pengujian yang dilakukan, nilai performansi tertinggi yang diperoleh adalah 91.3% untuk *accuracy*, 90.16% untuk *precision*, 93.18% untuk *recall*, dan 91.65% untuk *f-measure.* Nilai akurasi tersebut dapat dikatakan cukup tinggi namun belum mendekati 100%. Hal ini disebabkan pada proses pembentukannya, model klasifikasi ini masih memiliki nilai *error rate* sebesar 4% sehingga masih terdapat beberapa record pada data uji yang salah diprediksi. Nilai *error rate* pada proses pembentukan model klasifikasi diperoleh karena pada proses pembentukannya, terdapat *leaf node* yang dilabeli dengan kelas yang menjadi mayoritas dalam data latih yang ada sehingga untuk kelas yang menjadi minoritas dalam data latih akan diklasifikasikan ke dalam kelas yang salah. Misal, terdapat potongan model klasifikasi sebagai berikut:

```
contact in {telephone,unknown}:
:...duration <= 477.6331:
:  :...poutcome in {failure,other,unknown}:
:  :  :...month in {dec,feb,jan,jul,jun,may,nov}: no (1062/12)
```

Berdasarkan potongan model klasifikasi diatas, dari total 1074 sampel, 1062 diantaranya memiliki kelas no dan 12 sisanya memiliki kelas yes. Karena mayoritas di dalam data latih memiliki kelas *no*, maka leaf node akan dilabeli dengan kelas *no*.

4. Kesimpulan.

1. Klasifikasi status berlangganan klien bank pada kampanye pemasaran langsung menggunakan algoritma C5.0 diawali dengan tahap *preprocessing* untuk menangani permasalahan *imbalanced class* dengan menggunakan metode SMOTE. Setelah *preprocessing*, pembentukan model klasifikasi dengan algoritma C5.0 dilakukan dengan membagi sampel data berdasarkan atribut yang memiliki nilai *information gain* tertinggi.
2. Nilai performansi terbaik dari model klasifikasi yang terbentuk diperoleh ketika dilakukan penanganan terhadap permasalahan *imbalance class* dengan teknik SMOTE dengan persentase duplikasi kelas minoritas sebesar 700% atau perbandingan jumlah *record* kelas *yes*:*no* adalah sebesar 1:1.
3. Nilai performansi terbaik dari model klasifikasi yang terbentuk adalah sebesar 91.3% untuk *accuracy*, 90.16% untuk *precision*, 93.18% untuk *recall*, dan 91.65% untuk *f-measure.* Nilai akurasi tersebut dapat dikatakan cukup tinggi namun belum mendekati 100%. Hal ini disebabkan pada proses pembentukannya, model klasifikasi ini masih memiliki nilai *error rate* sebesar 4%.

Daftar Pustaka:

[1] U. M. L. Repository, "Bank Marketing Data Set," 14 February 2012. [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Bank+Marketing. [Accessed 19 May 2017].

[2] N. Patil, R. Lathi and V. Chitre, "Comparison of C5.0 & CART Classification algorithms using pruning technique," *International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181,* vol. 1, no. 4, pp. 1-5, 2012.

[3] Y. Yusuf W, "PERBANDINGAN PERFORMANSI ALGORITMA DECISION TREE C5.0, CART, DAN CHAID: KASUS PREDIKSI STATUS RESIKO KREDIT DI BANK X," *Seminar Nasional Aplikasi Teknologi Informasi 2007 (SNATI 2007),* pp. 59-62, 2007.

[4] S. Moro, R. M. S. Laureano and P. Cortez, "Using Data Mining for Bank Direct Marketing: An Application of the CRISP-DM Methodology," *European Simulation and Modelling Conference,* 2011.

[5] S. Abbas, "Deposit subscribe Prediction using Data Mining Techniques based Real Marketing Dataset," *International Journal of Computer Applications,* vol. 110, no. 3, pp. 1-7, 2015.

[6] Hairani, N. A. Setiawan and T. B. Adji, "METODE KLASIFIKASI DATA MINING DAN TEKNIK SAMPLING SMOTE MENANGANI CLASS IMBALANCE UNTUK SEGMENTASI COSTUMER PADA INDUSTRI PERBANKAN," in *Seminar Nasional Sains dan Teknologi*, Semarang, 2016.

[7] Z. Haiyang, "A Short Introduction to Data Mining and Its Application," *Management and Service Science (MASS),* pp. 1-4, 2011.

[8] B. M. Ramageri, "DATA MINING TECHNIQUES AND APPLICATIONS," *Indian Journal of Computer Science and Engineering,* vol. 1, no. 4, pp. 301-305.

[9] S. V. K. Kumar and P. Kiruthika, "An Overview of Classification Algorithm in Data Mining," *International Journal of Advanced Research in Computer and Communication Engineering,* vol. 4, no. 12, pp. 255-257, 2015.

[10] M. Gupta and N. Aggarwal, "CLASSIFICATION TECHNIQUES ANALYSIS," *National Conference on Computational Instrumentation,* pp. 128-131, 2010.

[11] S. Archana and K. Elangovan, "Survey of Classification Techniques in Data Mining," *International Journal of Computer Science and Mobile Applications,* vol. 2, no. 2, pp. 65-71, 2014.

[12] H. Munawaroh, B. K. K and Y. Kustiyahningsih, "PERBANDINGAN ALGORITMA ID3 DAN C5.0 DALAM INDENTIFIKASI PENJURUSAN SISWA SMA," *Jurnal Sarjana Teknik Informatika,* vol. 1, no. 1, pp. 1-12, 2013.

[13] A. Andriani, "SISTEM PENDUKUNG KEPUTUSAN BERBASIS DECISION TREE DALAM PEMBERIAN BEASISWA STUDI KASUS: AMIK "BSI YOGYAKARTA"," *Seminar Nasional Teknologi Informasi dan Komunikasi 2013,* pp. 163-168, 2013.

[14] J. Brownlee, "8 Tactics to Combat Imbalanced Classes in Your Machine Learning Dataset," Machine Learning Mastery, 19 August 2015. [Online]. Available: http://machinelearningmastery.com/tactics-to-combat-imbalanced-classes-in-your-machine-learning-dataset/. [Accessed 7 10 2016].

[15] S. A. Putri and R. S. Wahono, "Integrasi SMOTE dan Information Gain pada Naive Bayes untuk Prediksi Cacat Software," *Journal of Software Engineering,* vol. 1, no. 2, pp. 86-91, 2015.

[16] A. Z. K. Baizal, A. M. Bijaksana and A. S. Sastrawan, "ANALISIS PENGARUH METODE OVER SAMPLING DALAM CHURN PREDICTION UNTUK PERUSAHAAN TELEKOMUNIKASI," *Seminar Nasional Aplikasi Teknologi Informasi 2009,* pp. 61-66, 2009.

[17] J. Brownlee, "Classification Accuracy is Not Enough: More Performance Measures You Can Use," Machine Learning Mastery, 21 March 2014. [Online]. Available: http://machinelearningmastery.com/classification-accuracy-is-not-enough-more-performance-measures-you-can-use/. [Accessed 13 12 2016].

[18] B. Utami, "KLASIFIKASI PENENTUAN TIM UTAMA OLAHRAGA HOCKEY MENGGUNAKAN ALGORITMA C4.5 (Study Kasus : Hockey Kabupaten Kendal)," Jurusan Teknik Informatika FIK UDINUS, Semarang, 2015.

[19] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing and Management,* vol. 45, no. 4, pp. 427-437, 2009.

[20] S. Moro, P. Cortez and R. M. S. Laureano, "A Data Mining Approach for Bank Telemarketing Using the rminer Package and R Tool," Instituto Universitario de Lisboa, Portugal.

[21] D. Adiangga, "PERBANDINGAN MULTIVARIATE ADAPTIVE REGRESSION SPLINE (MARS) DAN POHON KLASIFIKASI C5.0 PADA DATA TIDAK SEIMBANG," Institut Pertanian Bogor, Bogor, 2015.

[22] F. Alhikmah, E. B. Setiawan and M. Imrona, "ANALISIS DAN IMPLEMENTASI ALGORITMA ID3 DAN CART PADA PENILAIAN KINERJA PEGAWAI," *Seminar Nasional Ilmu Komputasi & Teknik Informatika,* pp. 120-132, 2013.

[23] R. Pandya and J. Pandya, "C5.0 Algorithm to Improved Decision Tree with Feature Selection and Reduced Error Pruning," *International Journal of Computer Applications,* vol. 117, no. 16, pp. 18-21, 2015.

[24] I. Graha and Y. S. Nugroho, "DATA MINING JASA PENGIRIMAN TITIPAN KILAT DI PT CITRA VAN TITIPAN KILAT (TIKI) DENGAN METODE DECISION TREE," Program Sudi Teknik Informatika Fakultas Komunikasi dan Informatika Universitas Muhammadiyah Surakarta, Surakarta, 2014.

[25] A. S. Sastrwan, Z. A. Baizal and M. A. Bijaksana, "ANALISIS PENGARUH METODE COMBINE SAMPLING DALAM CHURN PREDICTION UNTUK PERUSAHAAN TELEKOMUNIKASI," *Seminar Nasional Informatika 2010,* pp. 14-22, 2010.

[26] F. Marbun, Z. A. Baizal and M. A. Bijaksana, "PERPADUAN COMBINED SAMPLING DAN ENSEMBLE OF SUPPORT VECTOR MACHINE (ENSVM) UNTUK MENANGANI KASUS CHURN PREDICTION PERUSAHAAN TELEKOMUNIKASI," *Jurnal Ilmiah Teknologi Informasi,* vol. 8, no. 2, pp. 43-48, 2010.