PENGGUNAAN METODE *BAGGING* DENGAN MENERAPKAN *DATA BALANCING* PADA *CHURN PREDICTION* UNTUK PERUSAHAAN TELEKOMUNIKASI

ZK. Abdurahman Baizal [1], Moch. Arif Bijaksana [2], Ina Rofi'atun Nasihati[3],
Telp (022)7564108 ext 2298 Fax (022)7565934
[1]Program Studi Ilmu Komputasi, Fakultas Sains Institut Teknologi Telkom, Bandung
[2,3]Program Studi Teknik Informatika, Fakultas Teknik Informatika Institut Teknologi Telkom, Bandung
Jl Telekomunikasi, Terusan Buah Batu, Bandung
E-mail : [1]zka@ittelkom.ac.id, [2]mab@ittelkom.ac.id, [3]ina_dhast3@yahoo.com,

ABSTRAK

Churn Prediction merupakan salah satu aplikasi data mining yang bertujuan untuk memprediksi para pelanggan yang berpotensial untuk churn. Churn Prediction merupakan salah satu kasus kelas imbalance dan churn merupakan kelas minor. Terdapat beberapa cara untuk mengatasi permasalahan imbalance class yang melekat pada kasus churn ini. Salah satu contohnya dengan cara melakukan balancing terhadap data training atau dengan cara menggunakan metode yang khusus dapat menyelesaikan permasalahan imbalance class ini. Analisis yang dilakukan pada penelitian ini adalah mengetahui apakah metode Bagging dan Lazy Bagging dapat dijadikan solusi dalam mengklasifikasikan data churn. Dalam mendukung penelitian ini, dibuat perangkat lunak yang mengimplementasikan metode Bagging, dan Lazy Bagging. Pengujian dilakukan dengan menggunakan data salah satu perusahaan telekomunikasidi Indonesia. Sebagai metode pembanding adalah Boosting Clementine 10.1 dan C5.0 Clementine 10.1. Analisis dilakukan dengan melakukan penghitungan akurasi model churn prediction yang dinyatakan dalam bentuk lift curve, top decile dan gini coefficient serta f-measure untuk penghitungan akurasi data yang imbalance. Dari analisa yang dilakukan, metode Bagging dapat memprediksikan data churn jika dilakukan balancing terlebih dahulu terhadap data training yang digunakan. Tetapi dari parameter lift curve, gini coefficient, ternyata Lazy Bagging menghasilkan nilai yang lebih baik untuk data yang sangat imbalance (tanpa balancing)

Kata kunci : bagging, lazy bagging, boosting , data imbalance, churn prediction, akurasi.

1. PENDAHULUAN

Perkembangan teknologi telekomunikasi yang semakin pesat mendorong berkembangnya perusahaan-perusahaan telekomunikasi selular seperti CDMA dan GSM. Semakin banyaknya perusahaan ini maka akan menyebabkan semakin maraknya persaingan diantara perusahaan telekomunikasi untuk menarik pelanggan sebanyak-banyaknya

Kasus *churn* merupakan permasalahan utama yang sering dihadapi oleh para perusahaan telekomunikasi karena akan berpengaruh terhadap *revenue* yang didapatkan oleh perusahaan tersebut. Oleh karena itu, perlu adanya suatu model prediksi yang akurat sehingga dapat memprediksi pelanggan yang akan *churn*.

Permasalahan yang ada pada data churn yaitu *imbalance class*, yang berarti adanya kelas mayor dan kelas minor (*rare event*). "*Churn* merupakan kelas minor karena biasanya untuk setiap bulannya Rata-rata churn pada suatu perusahaan telekomunikasi sekitar 1,8% dari seluruh pelanggan yang ada atau bahkan lebih sedikit lagi (Lemmens, 2006)". "*Ensemble method* merupakan metode yang membangun kumpulan *classifier* dari data *training* dan kemudian memprediksi kelas label pada data *testing* dengan ide menggabungkan prediksi dari beberapa *classifier* tersebut (Tan, Pang-Ning, 2005)". "Terdapat dua metoda pada *ensemble method* yaitu *Bagging* dan *Boosting* (Tan, Pang-Ning, 2005)". Oleh karena itu, pada penelitian ini dilakukan analisa *churn prediction* dengan menggunakan metode *Bagging*, dan *Lazy Bagging*. Sebagai metode pembanding yang juga merupakan *ensemble method* adalah Boosting Clementine 10.1 (memanfaatkan tool Clementaine 10.1). Metode pembanding tambahan adalah C5 Clementine 10.1 (memanfaatkan tool Clementaine 10.1)

Hasil akhir penelitian ini adalah *churn prediction* dengan melakukan penghitungan akurasi model *churn prediction* yang dinyatakan dalam bentuk *lift curve*, *gini coefficient* dan *top decile* lift sebagai evaluasi untuk kasus *churn.* Selain itu juga akan dilakukan perhitungan *f-measure* sebagai evaluasi untuk data *imbalance.*

2. *CHURN PREDICTION*

Churn prediction adalah salah satu aplikasi dari *task data mining* yang bertujuan untuk memprediksi pelanggan yang berpotensi untuk *churn*.

Dalam hal ini, pelanggan yang *churn* dapat dibagi menjadi dua kelompok utama (Rob, 2005), yaitu:

a. *Voluntary churners* / sukarela

Voluntary churners lebih sukar untuk ditentukan, sebab pada pelanggan jenis ini *churn* terjadi ketika seorang pelanggan membuat keputusan secara sadar untuk mengakhiri layanan yang digunakan.

b. *Involuntary churners* / tidak sukarela
Involuntary churners ini lebih mudah untuk diidentifikasi, seperti pelanggan yang menggunakan jasa ditarik/dicabut dengan sengaja oleh perusahaan tersebut dikarenakan adanya beberapa alasan.

Churn prediction merupakan salah satu kasus *imbalance class* pada kondisi datanya. Pada kasus *churn prediction*, terdapat dua kelas yaitu kelas loyal dan kelas *churn.*

3. *IMBALANCE PROBLEM*

"*Learning* dari dataset *imbalance*, dimana jumlah instan pada satu kelas (kelas mayor) jauh lebih banyak dibandingkan dengan kelas yang lain, merupakan suatu tantangan penting untuk komunitas mesin *learning* (Guo, 2004)".

Imbalance class merupakan suatu masalah atau tantangan karena biasanya mesin *learning* akan menghasilkan suatu akurasi prediksi yang baik terhadap kelas data latih yang banyak (kelas mayor), sedangkan untuk kelas data *training* yang sedikit (kelas minor) akan dihasilkan akurasi prediksi yang buruk.

4. *BAGGING*

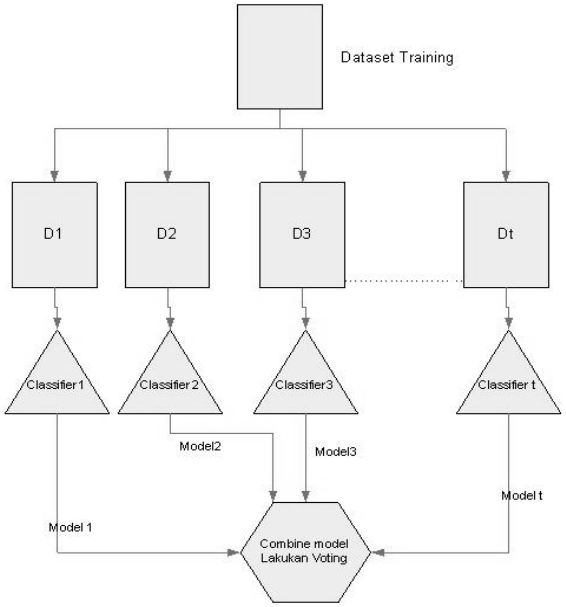
"Bagging merupakan metode yang dapat memperbaiki hasil dari algoritma klasifikasi machine learning (Breimann, 1994)". "Metode ini diformulasikan oleh Leo Breiman dan nama tersebut disimpulkan dari phrase "Bootstrap Aggregating" (Breimann, 1994)". Bagging merupakan salah satu metode yang berdasar pada *ensemble method*, oleh karena itu secara umum tahap-tahap pada metode Bagging dapat dilihat pada gambar yang telah dijelaskan tadi. Ada beberapa hal penting dalam metode ini yaitu :

a. Pendistribusian data (*bootstrap*) dibuat dengan menggunakan *sampling with replacement.*
b. Membangun classifier pada setiap *bootstrap sample.*

Cara kerja metode ini, diilustrasikan pada gambar 1.

5. *LAZY BAGGING*

"*Lazy Bagging* (LB) merupakan algoritma yang dapat digunakan untuk memprediksi data *imbalance* (Zhu, 2007)". Desain *Lazy Bagging* yaitu dengan cara membangun *bootstrap* berdasarkan pada karakteristik yang terdapat pada *instance* test. "Pertama, *Lazy Bagging* akan mencoba untuk menemukan tetangga terdekat untuk *test instance* sebanyak k dari training set *T*, dan menggunakan tetangga terdekat yang ditemukan tersebut untuk membangun *bootstrap bag* (Zhu, 2007)". "Berbeda dari *Bagging* biasa yang secara langsung melakukan sampling sebanyak *N* instan dari training set *T*, *Lazy Bagging* akan melakukan sampling secara *independent* dari keduanya yaitu dari kumpulan tetangga terdekat sebanyak *K* dan data training sebanyak *N-K* instan (Zhu, 2007)". *Lazy Bagging* merupakan *Lazy learning*, maka proses *learning* dilakukan dengan cara menunggu sampai terdapat instan test.



Gambar 1. Proses *Bagging*

Tabel 1. Kriteria Metode *Bagging* dan *Boosting*

No	Kriteria	*Bagging*	*Boosting*
1.	Berdasar pada metode *ensemble*	Ya	Ya
2.	Satu iterasi tidak berpengaruh terhadap iterasi lain (*independent*)	Tidak	Ya
3.	*Voting* dilakukan dengan cara *voting* kelas *mayoritas* yang dipilih , tidak dipengaruhi adanya nilai beta/alfa yang didapatkan dari setiap iterasi.	Ya	Tidak
4.	Adanya perhitungan bobot, nilai beta, dan *update* bobot yang dilakukan setiap iterasi	Tidak	Ya

6. *BOOSTING*

Metode *boosting* bekerja pada tiap iterasi yang dilakukan. Antara iterasi satu dengan iterasi yang selanjutnya ada keterkaitan. Setelah dilakukan pengklasifikasian pada tiap iterasi, akan adanya peng-*update*-an bobot pada setiap *record* data. Hal ini ditujukan untuk meningkatkan bobot pada record data yang salah diklasifikasikan pada iterasi sebelumnya dan mengurangi bobot pada record-record yang telah benar diklasifikasikan. Setelah itu akan dilakukan voting berdasarkan bobot yang didapatkan dari setiap iterasi. (lihat tabel 1).

7. C5

C5 adalah salah satu model decision tree. Di sini digunakan kriteria information gain untuk memilih atribut yang akan digunakan untuk pemisahan obyek. Atribut yang mempunyai information gain tertinggi dipilih untuk melakukan pemecahan. Algoritma ini adalah perbaikan dari C45 (Woolf, R.J., 2005). Beberapa perbaikan dalam C5 (Berry, Linoff, 2004) adalah dari sisi kecepatan, penghematn memori, decision tree yang lebih ramping, serta kemampuannya dalam mereduksi noise. Beberapa fungsionalitas tambahan di C5 adalah

1. *Variable misclassification costs*
2. *Case weight attribute*
3. Ada tambahan untuk menangani tipe data *dates*, *times*, *timestamps*, *ordered discrete attributes*, dan *case labels*
4. Dukungan untuk sampling and *cross-validation*

8. EVALUASI *IMBALANCE CLASS*

Setelah terbentuk matrik evaluasi, dapat dihitung beberapa parameter yang akan dijadikan ukuran sebagai evaluasi performansi *classifier* pada data *imbalance*. Parameter – parameter tersebut dapat dihitung dengan formula sebagai berikut.

$$Recall \quad (r) : \frac{TP}{TP+FN} \qquad (1)$$

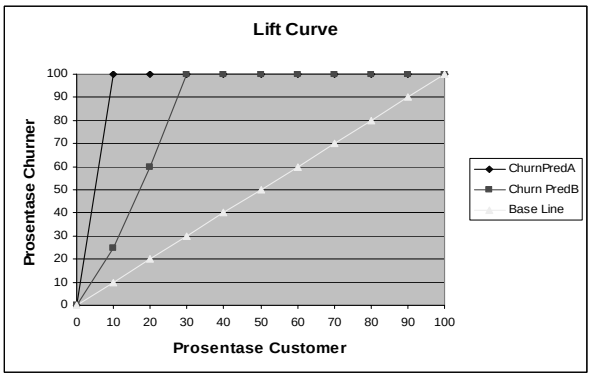
$$Precision \quad (p) : \frac{TP}{TP+FP} \qquad (2)$$

$$F\text{-}measure : \frac{2rp}{r+p} \qquad (3)$$

9. EVALUASI CHURN PREDICTION

9.1 Lift Curve

Lift curve adalah alat ukur yang biasa digunakan di dalam kasus *churn prediction* yang memetakan hasil prediksi dari model *classifier* ke dalam bentuk kurva seperti pada gambar 2.



Gambar 2. Contoh Lift Curve

9.2 *Top Decile* 10%

Top decile 10% merupakan akurasi yang lebih memfokuskan pada 10% *riskiest segment* yaitu fokus kepada sekumpulan customer sebanyak 10 % dari keseluruhan customer yang memiliki probabilitas *churn* yang paling tinggi. Sehingga dapat diketahui *customer* mana saja yang mempunyai kemungkinan untuk *churn* lebih besar dan suatu perusahaan dapat mengatur strategi untuk customer yang termasuk ke dalam kelompok *riskiest segment*, sehingga dapat dilakukan pencegahan prosentase *churner* yang lebih banyak lagi

$$TopDecile = \frac{\hat{\pi}10\%}{\hat{\pi}} \qquad (4)$$

Keterangan

$\hat{\pi}10\%$: prosentase churner yang berada pada riskiest segment

$\hat{\pi}$: prosentase churner pada keseluruhan customer

9.3 Gini Coefficient

Gini coefficient tidak hanya fokus pada kumpulan *customer* yang paling beresiko untuk churn. "Pengukurannya mempertimbangkan semua *score*, termasuk *customer* yang kemungkinan untuk churnnya kecil (Lemmens, 2006)". Perhitungan *gini coefficient* dapat dilakukan dengan formula (Lemmens, 2006):

$$Gini = \left(\frac{2}{n}\right)\sum_{i=1}^{n}\left(vi - \hat{v}i\right) \qquad (2)$$

Keterangan:

vi : prosentase churner yang nilai probabilitas churnnya sama atau lebih besar dari customer ke i.

$\hat{v}i$: prosentase customer yang nilai probabilitas churnnya sama atau lebih besar dari customer ke i.

n : jumlah customer.

10. ANALISIS DAN PENGUJIAN

Data yang digunakan adalah data pelanggan salah satu perusahaan telekomunikasi di Indonesia dengan komposisi sebagai berikut:

1. Data *training*, dengan jumlah record sebanyak 36265 *record* dan tingkat *imbalance* 0.77%.
2. Data *testing* dengan jumlah record sebanyak 12119 *record* dan tingkat *imbalance* 0.78%.

Metode *Bagging* yang digunakan adalah *Bagging* dan *Lazy Bagging.* Sebagai pembanding adalah metode *Boosting Clementine10*.1 serta C5 *Clementine10*.1.

Adapun Skenario Pengujian Sistem, dibagi menjadi 3 macam seperti terlihat pada tabel 2, dimana dilakukan proses *data balancing*, dengan cara menduplikasi data minor.

Tabel 2. Skenario Distribusi Data

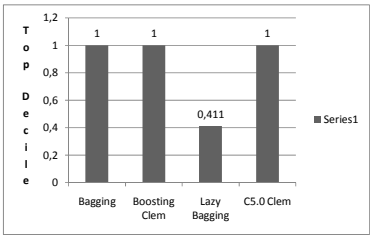
Skenario	Data Mayor	Data Minor	% *Imbalance*	Jumlah Data
1	35985	280	0.77%	36265
2	35985	14000	28%	49985
3	35985	30800	46.12%	66785

10.1 Analisis Akurasi Berdasarkan Parameter Evaluasi *Churn Prediction*

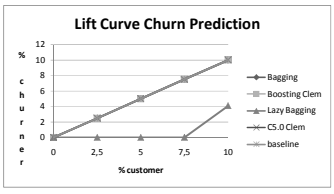
Parameter evaluasi churn prediction adalah *top decile* 10%, *lift curve* dan *gini coefficient*. Analisis dilakukan untuk ketiga skenario pada tabel 2.

10.1. 1. Pengujian dan Analisis Skenario 1

Dari hasil pengujian diperoleh nilai *top decile* yang ditunjukkan pada gambar 3.



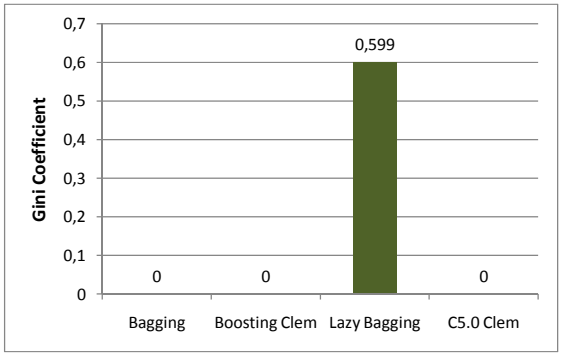
Gambar 3. *Top Decile* 10% Skenario 1



Gambar 4. Nilai *Lift Curve* Skenario 1

Bagging, *Boosting* Clementine dan C5.0 menghasilkan nilai *top decile* 1 karena semua *record* tidak ada yang diprediksikan *churn* dan mempunyai probabilitas *churn* yang sama untuk semua *record*. *Lazy Bagging* dapat menghasilkan nilai *top decile* yang berbeda dengan metode lain karena *rule* yang dibentuk beragam sehingga mengakibatkan probabilitas *churn* pun beragam.

Nilai *lift curve* dari hasil pengujian skenario 1 ditunjukkan pada gambar 4. Sama halnya seperti pada *top decile* 10%, *lift curve* untuk *Bagging*, *Boosting* dan C5.0 tidak terlihat karena *lift curve* yang dihasilkan sama dengan baseline (random).

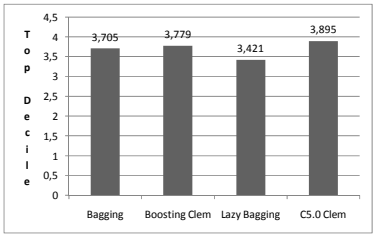


Gambar 5. *Gini Coefficient* Skenario 1

Nilai *Gini Coefficient* dari hasil pengujian skenario 1 ditunjukkan pada gambar 5. Nilai akurasi *gini coefficient* untuk *Bagging*, *Boosting* dan C5.0 menghasilkan nilai akurasi 0 dikarenakan membentuk *lift curve* yang sama dengan random *lift curve* (baseline) sehingga luasnya bernilai 0. Hal ini dikarenakan probabilitas churn yang dihasilkan sama untuk semua record. Nilai *gini coefficient* yang tertinggi yaitu *Lazy Bagging* karena rule yang dibentuk relatif lebih sedikit. Hal ini dikarenakan Lazy Bagging tidak menghasilkan data sintetik tapi menghasilkan bootstrap yang berisi tetangga terdekat data testing yang diambil dari data training itu sendiri.

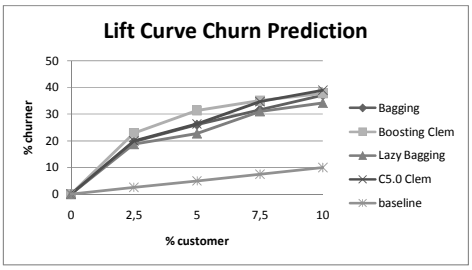
10.1. 2. Pengujian dan Analisis Skenario 2

Dari hasil pengujian diperoleh nilai *top decile* yang ditunjukkan pada gambar 6.



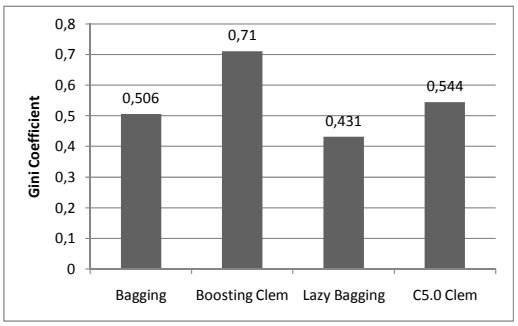
Gambar 6. Top Decile 10% Skenario 2

Nilai *top decile* yang tertinggi dihasilkan oleh C5.0 sebesar 3,895 sebagai peringkat pertama. Dalam hal ini *Bagging* menunjukkan kinerja yang lebih baik dibanding *Lazy Bagging*. Tetapi dibanding skenario 1, *Lazy Bagging* sudah lebih baik.



Gambar 7. Nilai *Lift Curve* Skenario 2

Nilai *lift curve* dari hasil pengujian skenario 2 ditunjukkan pada gambar 7. Sama halnya seperti pada *top decile*, *lift curve* yang tertinggi dihasilkan oleh C5.0 sebesar 38,95% *churner* pada 10% *customer* sebagai peringkat pertama. Sedangkan Bagging sedikit lebih baik daripada *Lazy Bagging*, walaupun keduanya masih lebih buruk daripada C5 dan *Boosting*.

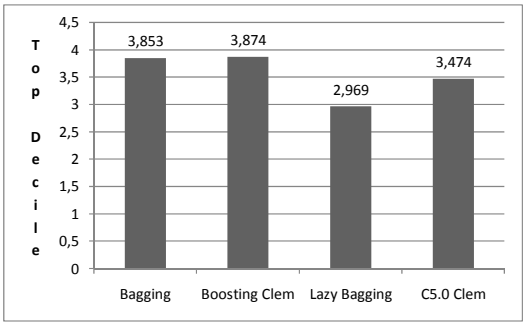


Gambar 8. Nilai *Gini Coefficient* Skenario 2

Nilai *Gini Coefficient* dari hasil pengujian skenario 2 ditunjukkan pada gambar 8. Nilai *gini coefficient* yang tertinggi yaitu pada saat menggunakan *Boosting* Clementine. Hal dikarenakan rule yang dibentuk oleh *Boosting* Clementine lebih sedikit keragamannya dibandingkan dengan metode lain. Di sini nilai *Gini* untuk Bagging sedikit lebih baik daripada *Lazy Bagging.*

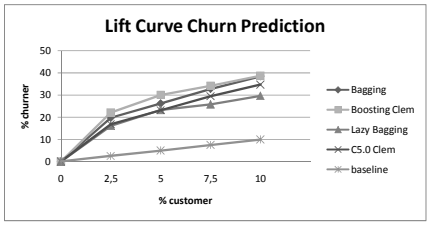
10.1.3. Pengujian dan Analisis Skenario 3

Dari hasil pengujian diperoleh nilai *top decile* yang ditunjukkan pada gambar 9.



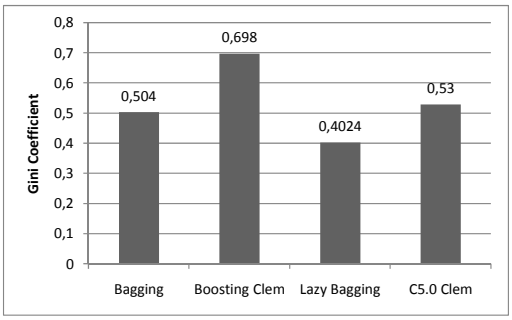
Gambar 9. Top Decile 10% Skenario 3

Boosting menempati peringkat pertama dengan menghasilkan nilai top decile 3,874. Bagging menempati peringkat kedua dengan nilai top decile 2,853. Sementara itu Lazy Bagging nilai Top decile paling kecil. Nilai *Top decile* untu Lazy Bagging tampak ada penurunan daripada skenario 2, ini data sintetik yang dibentuk lebih banyak sehingga keragaman probabilitas pun lebih banyak.



Gambar 10. Nilai *Lift Curve* Skenario 3

Nilai *lift curve* dari hasil pengujian skenario 2 ditunjukkan pada gambar 10. Boosting menempati peringkat pertama dengan menghasilkan lift curve yang terbaik dengan menebak 38,736% churner dari 10% customer. Bagging menempati peringkat kedua dengan menghasilkan lift curve 38,526% churner. Di sini *Lazy Bagging* ada peringkat terakhir, ini dikarenakan pada skenario 3 lebih banyak data sintetik yang dibentuk, dan ini justru melemahkan metode *Lazy Bagging*.



Gambar 11. Nilai *Gini Coefficient* Skenario 3

Nilai *Gini Coefficient* dari hasil pengujian skenario 2 ditunjukkan pada gambar 11. Sama halnya pada skenario 2, nilai *gini coefficient* yang tertinggi yaitu pada saat menggunakan Boosting Clementine yaitu sebesar 0,698. Hal dikarenakan rule yang dibentuk oleh *Boosting* Clementine lebih sedikit keragamannya dibandingkan dengan metode lain. *Lazy Bagging* mempunyai nilai yang paling kecil, dikarenakan pada skenario 3 lebih banyak data sintetik yang dibentuk, dan ini justru melemahkan metode *Lazy Bagging*

10.2 Analisis Akurasi Berdasarkan Parameter Evaluasi *Imbalance Class.*

Parameter evaluasi untuk imbalance class yang dipakai di sini adalah *f-measure*. Nilai f-measure dari hasil pengujian dari keetiga skenario diperlihatkan pada tabel 2

Tabel 2. Nilai *f-measure*

Metode	Skenario 1	Skenario 2	Skenario3
Bagging	0	0.095	0.092
LazyBagging	0	0.085	0.065
Boosting Clementine	0	0.138	0.122
C5.0	0	0.085	0.095

Pada skenario 1 nilai *F-measure* semuanya 0, ini berarti pada data yang sangat *imbalance*, keempat metode tidak dapat melakukan prediksi dengan baik. Sedangkan pada keadaan data yang sudah di-*balancing* , yaitu pada skenario 2 dan skenario 3 nilai *f-measure* yang tertinggi yaitu dihasilkan oleh *Boosting* Clementine, sementara itu Bagging dan Lazy Bagging juga mengalami peningkatan kemampuan.

11. KESIMPULAN

Metode *Bagging* dan Lazy Bagging dapat memprediksikan data *churn* jika dilakukan *balancing* terlebih dahulu terhadap data *training* yang digunakan. Tetapi metode Lazy Bagging lebih peka terhadap proses Balancing ini jika duplikasi data minor dilakukan lebih banyak, nilai penurunan kinerja metode ini lebih mencolok daripada Bagging.

Metode *Lazy Bagging* cenderung menghasilkan nilai akurasi *lift curve*, dan *gini coefficient* yang lebih baik dibandingkan metode lain pada *data imbalance* (tanpa proses *balancing*).

Lazy Bagging belum berhasil dalam mengatasi masalah *data imbalance*, karena terlihat pada akurasi nilai *f-measure* skenario 1, menghasilkan *f-measure* yang sangat kecil.

PUSTAKA

Berry, Michael J.A., Linoff, Gordon S., (2004), *Data Mining Techniques For Marketing, Sales, Customer Relationship Management, Second Edition*, John Wiley and Sons

Breiman,L:*Bagging* predictors., (1994) *Technical Report* 421, Departemen of Statistics, University of California at Berkeley, USA

Guo,H.,Viktor ,H.L., (2004) *Learning from Imbalanced Data Sets with Boosting and Data Generation: The Databoost-IM Approach*, ACM SIGKDD Explorations, 630-39.

Lemmens, Aurelie., Croux, Christophe., (2006).,*"Bagging and Boosting Classification Trees"*. Journal of Marketing Research, 43(2) 276-286.

Machova, Kristina., Barcak, Frantisek., Bedar, Peter., (2004), *A Bagging Method using Decision Trees in The Role of Base Classifiers*, Technical University, Slokavia.

Rob, Matison, (2005), *Telco Churn Management : The Golden Opportunity*, XIT Press, Illionis, USA

Tan, Pang-Ning., Steinbach, Michael and Kumar, Vipin., (2005) *Introduction to Data Mining.* Addison Wesley, USA

Woolf, R.J.(2005) *Data Mining using Matlab*. Faculty of Engineering & Surveying University of Southern Queensland

Zhu, Xingquan., (2007), *Lazy Bagging for Classifying Imbalance Data*, ICDM, Boca Raton.USA.