

PERPADUAN COMBINED SAMPLING DAN *ENSEMBLE OF SUPPORT VECTOR MACHINE* (ENSVM) UNTUK MENANGANI KASUS *CHURN PREDICTION* PERUSAHAAN TELEKOMUNIKASI

Fernandy Marbun¹⁾, Z.K. Abdurahman Baizal²⁾, Moch Arif Bijaksana³⁾

^{1,3}Program Studi Teknik Informatika Institut Teknologi Telkom, Bandung

²Program Studi Ilmu Komputasi Institut Teknologi Telkom, Bandung

Jl. Telekomunikasi Dayeuh Kolot, Bandung 40257

Telp : 0227564108- psw 2298, Fax : 0227565934

Email : ¹ndy_mrb@yahoo.com, ²zka@ittelkom.ac.id, ³mab@ittelkom.ac.id

Abstrak

Churn prediction adalah suatu cara untuk memprediksi pelanggan yang berpotensi untuk *churn*. *Data mining* khususnya klasifikasi tampaknya dapat menjadi salah satu alternatif solusi dalam membuat model *churn prediction* yang akurat. Namun hasil klasifikasi menjadi tidak akurat disebabkan karena data *churn* bersifat *imbalance*. Kelas data menjadi tidak stabil karena data akan lebih condong ke bagian data yang memiliki komposisi data yang lebih besar. Salah satu cara untuk menangani permasalahan ini adalah dengan memodifikasi dataset yang digunakan atau yang lebih dikenal dengan metode *resampling*. Teknik *resampling* ini meliputi *over-sampling*, *under-sampling*, dan *combined-sampling*. Metode *Ensemble of SVM* (EnSVM) diharapkan dapat meminimalisir kesalahan klasifikasi kelas mayor dan minor yang dihasilkan oleh classifier SVM tunggal. Dalam penelitian ini akan dicoba untuk memadukan *combined sampling* dan EnSVM untuk *churn prediction*. Pengujian dilakukan dengan membandingkan hasil klasifikasi *CombinedSampling-EnSVM* dengan *SMOTE-SVM* (perpaduan *oversampling-SVM*) dan *pure-SVM*. Hasil pengujian menunjukkan bahwa metode *CombinedSampling-EnSVM* secara umum hanya mampu menghasilkan performansi *Gini Index* yang lebih baik daripada metode *SMOTE-SVM* dan tanpa *resampling* (*pure-SVM*).

Kata kunci: *churn prediction*, *imbalance*, *combined sampling*, *ensemble*, *SVM*, *performansi*.

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

Churn prediction pada pelanggan telekomunikasi merupakan upaya memprediksi/mengklasifikasi pelanggan jasa telekomunikasi yang berhenti atau berpindah berlangganan dari suatu operator ke operator yang lain. 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)".

Untuk itulah maka dibangun suatu sistem *data mining* untuk memprediksi *churn* (*churn prediction*) yang dapat memprediksi (mengklasifikasi) apakah seorang pelanggan berpotensi melakukan *churn* atau tidak.

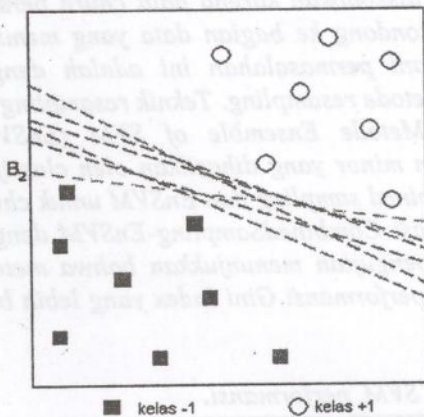
Untuk mengatasi permasalahan *imbalance* pada data *churn*, dapat digunakan metode *Synthetic Minority Over-sampling Technique* (SMOTE). Namun teknik ini masih memiliki kelemahan [17]. Untuk mengurangi kelemahan yang ada pada teknik ini, maka pada perangkat lunak *churn prediction* ini, proses *sampling* menggunakan teknik *Combined Sampling* yang diharapkan dapat meningkatkan akurasi kelas minor dan mayor [8].

Metode *Ensemble of SVM* (EnSVM) adalah kumpulan N buah SVM yang dilatih secara individu terhadap N *dataset* yang telah diproses sebelumnya (*preprocessing*) dan hasil prediksi akhir dihasilkan dengan menggabungkan hasil dari masing-masing SVM [9]. Dengan metode *Combined Sampling* dan *ensemble* tersebut diharapkan dapat meminimalisir kesalahan klasifikasi kelas mayor dan minor yang dihasilkan oleh *classifier* tunggal. Penelitian ini bertujuan untuk membangun suatu perangkat lunak

alternatif *churn prediction* yang dapat digunakan untuk memprediksi (mengklasifikasikan) pelanggan-pelanggan yang (berpotensi) melakukan *churn* pada kasus perusahaan telekomunikasi, serta mengimplementasikan teknik *Combined Sampling* pada tahap *sampling* dari metode EnSVM, serta mengimplementasikan pada perangkat lunak metode *pure-SVM*, *SMOTE-SVM*, dan *CombinedSampling-EnSVM*. Pada akhir penelitian, akan dilakukan pengujian dengan membandingkan dan menganalisis performansi terhadap metode *pure-SVM*, *SMOTE-SVM*, dan *CombinedSampling-EnSVM*.

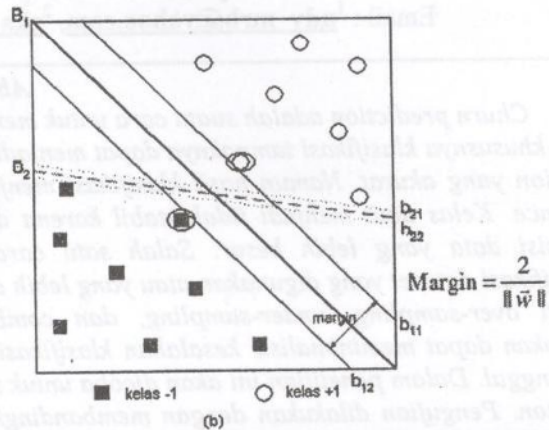
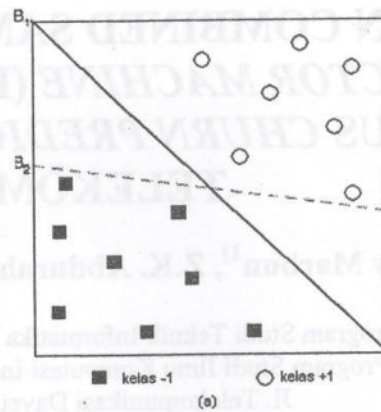
2. SUPPORT VECTOR MACHINE (SVM)

Konsep dasar SVM adalah bagaimana menemukan atau mencari fungsi pemisah atau *hyperplane* terbaik yang memisahkan dua kelas pada *input space* atau ruang input [2, 4,6,7,9,13,15].



Gambar 1. Ilustrasi pemisahan kelas dengan berbagai pilihan *hyperplane*

Problem klasifikasi dapat diterjemahkan dengan usaha menemukan *hyperplane* yang memisahkan antara kedua kelompok (+1 dan -1). Salah satu alternatif *hyperplane* adalah *hyperplane* B2, dan berbagai alternatif *hyperplane* yang lain diilustrasikan pada gambar 1. *Hyperplane* terbaik antara kedua kelas dapat ditemukan dengan mencari margin *hyperplane* terbesar. Tujuan dari usaha ini adalah untuk meningkatkan probabilitas pengelompokan secara benar pada data *testing*. Margin disini adalah jarak antara titik terdekat dari masing-masing kelas terhadap *hyperplane*. Dengan kata lain *task* SVM adalah bagaimana menemukan *hyperplane* yang terletak tepat ditengah-tengah diantara dua titik terdekat dari dua kelompok kelas yang berbeda tersebut



Gambar 2: Ilustrasi pemisahan kelas *hyperplane* terbaik dengan margin terbesar dan support vector-nya.

Pada gambar 2a diatas, margin *hyperplane* B1 lebih besar daripada margin *hyperplane* B2. Sehingga *maximum margin hyperplane* (MMH) [7] atau *optimal separating hyperplane* (OSM) [2] pada contoh kasus diatas adalah *hyperplane* B1. Titik-titik paling dekat ini disebut dengan *Support Vectors* (SV) seperti yang diilustrasikan dengan titik elips pada gambar 2-b.

Jika data dinotasikan sebagai $\vec{x} \in \mathcal{R}$, dan label dari masing-masing kelas dinotasikan dengan $y_i \in \{-1, +1\}$ untuk $i = 1, 2, \dots, l$, di mana l adalah banyaknya jumlah data. Dengan asumsi bahwa kedua kelas dapat terpisah secara sempurna oleh *hyperplane* dengan dimensi d , maka *hyperplane* dapat dinotasikan sebagai berikut [12]:

$$\vec{w} \cdot \vec{x} + b = 0 \quad (1)$$

Karena *hyperplane* membagi dua lokasi berdasarkan kelas masing-masing, maka untuk sampel \vec{x}_i yang termasuk kelas negatif (-1) dan positif (+1) masing-masing memenuhi pertidaksamaan [4,6,7,10,12,16]:

$$\vec{w} \cdot \vec{x}_i + b \leq -1 \quad (2)$$

$$\vec{w} \cdot \vec{x}_i + b \geq +1 \quad (3)$$

Sehingga kedua margin yang sama tersebut dapat dihitung dengan cara mengurangi persamaan (3) dengan (2) sebagai berikut [10]:

$$\bar{w} \cdot \bar{x}_i + b = +1 - \bar{w} \cdot \bar{x}_i + b = -1 \Rightarrow$$

$$\bar{w} \cdot (x_i - x_i) = 2 \Rightarrow \left(\frac{\bar{w}}{\|\bar{w}\|} (x_1 - x_2) \right) = \frac{2}{\|\bar{w}\|},$$

seperti diilustrasikan pada gambar 2-b. Dimana $\|\bar{w}\|$ adalah bentuk normal *Euclidean* dari \bar{w} yaitu $\sqrt{\bar{w} \cdot \bar{w}}$, jika $\bar{w} = \{w_1, w_2, \dots, w_n\}$ maka $\sqrt{\bar{w} \cdot \bar{w}} = \sqrt{w_1^2 + w_2^2 + \dots + w_n^2}$ [7].

Ini berarti jarak antara *hyperplane* dengan dua titik terdekat pada kelas yang berbeda adalah sama yaitu $\frac{1}{\|\bar{w}\|}$.

Oleh karena itu, untuk memperoleh OSM atau MMH dapat dilakukan dengan cara memaksimalkan jarak antara *hyperplane* dengan masing-masing SV pada kedua kelompok kelas yang berbeda tadi yaitu memaksimalkan $\frac{1}{\|\bar{w}\|}$. Memaksimalkan margin ini

ekivalen dengan meminimalkan *objective function* [2,4,7,6,10,11,12,16]:

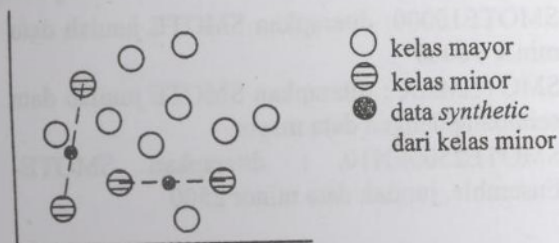
$$\min_w \tau(w) = \frac{1}{2} \|\bar{w}\|^2 \quad (4)$$

Dengan *constraint* yang harus dipenuhi sesuai dengan pertidaksamaan berikut :

$$y_i (\bar{x}_i \cdot \bar{w} + b) - 1 \geq 0, \forall i$$

3. SMOTE (SYNTHETIC MINORITY OVER-SAMPLING TECHNIQUE)

SMOTE melakukan sejumlah *over-sampling* terhadap masing-masing *instance* kelas minor dengan menghasilkan *instance* sintetis (*synthetic instances*) secara random di sepanjang garis yang menghubungkannya dengan salah satu dari *k-Nearest Neighbour* kelas minor. Tergantung pada jumlah *over-sampling* yang diinginkan, tetangga dari *k-Nearest Neighbour* dipilih secara random [1,3]. Misalkan, k dipilih 5 kemudian jumlah *over-sampling* (T) dipilih 300%, dan jumlah data (P), maka SMOTE pertama-tama memilih 3 dari 5 tetangga terdekat secara random. Kemudian membuat *instance* baru disepanjang garis yang menghubungkan *instance* tersebut dengan tetangganya secara random. Jadi, ada 3 *instance* sintetis baru untuk setiap P *instance* awal. Ilustrasi dari SMOTE dapat dilihat pada gambar 3.



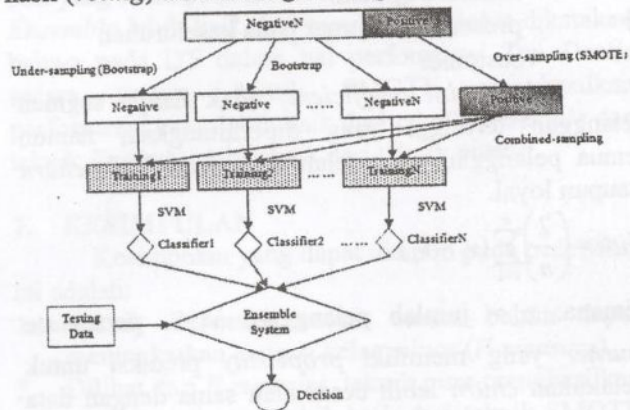
Gambar 34. Ilustrasi SMOTE

2.3 Combined Sampling

Teknik *Combined Sampling* disini dilakukan dengan cara sebagai berikut: pertama, melakukan *over-sampling* dengan menggunakan teknik SMOTE pada kelas minor hingga pada tingkat tertentu (T). Kedua, melakukan *under-sampling* yang masing-masing didapatkan dengan melakukan *under-sampling* kelas mayor dengan teknik *bootstrap* sampai jumlah/tingkat yang sama dengan kelas minor. Ketiga, menggabungkan hasil *over-sampling* dan *under-sampling* tersebut.

4. ENSEMBLE OF SVM (EnSVM)

Metode *Ensemble of SVM* (EnSVM) adalah kumpulan N buah SVM yang dilatih (*training*) secara individu terhadap N *dataset* yang telah diproses sebelumnya (*preprocessing*) dan hasil prediksi atau klasifikasi akhir dihasilkan dengan menggabungkan hasil (*voting*) dari masing-masing SVM.



Gambar 4. Ilustrasi EnSVM

Gambaran tentang EnSVM dapat dilihat pada gambar 4.

4. PARAMETER EVALUASI PERFORMANSI

4.1. Evaluasi performansi *imbalance class*:

Untuk mengukur performansi suatu *classifier* dalam *data mining* biasanya dengan menggunakan *confusion matrix* atau matriks evaluasi:

		Prediksi	
		Positive	Negative
Aktual	Positive	TP (True Positive)	FN (False Negative)
	Negative	FP (False Positive)	TN (True Negative)

Evaluasi performansi *imbalance class*:

$$recall(r) = \frac{TP}{TP + FN}$$

$$\text{precision}(p) = \frac{TP}{TP + FP}$$

$$\text{negative_accuracy} = \frac{TN}{TN + FP}$$

$$\text{positive_accuracy} = \frac{TP}{TP + FN}$$

$$g\text{-mean} = \sqrt{a^- \cdot a^+}$$

$$F\text{-measure} = \frac{2 \cdot r \cdot p}{r + p}$$

4.2. Evaluasi Performansi Churn Prediction :

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.

$$\text{TopDecile} = \frac{\hat{\pi}_{10\%}}{\hat{\pi}}$$

Keterangan

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

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

Dalam *gini coefficient*, tidak hanya segmen pelanggan tertinggi yang diperhitungkan, namun semua pelanggan yang telah diprediksi, baik *churn* ataupun loyal.

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

Dimana: n = jumlah pelanggan, v_i = persentase *churner* yang memiliki *propensity* prediksi untuk melakukan *churn* lebih besar atau sama dengan data *customer* ke- i , dan \hat{v}_i = persentase *customer* yang memiliki *propensity* prediksi untuk melakukan *churn* lebih besar atau sama dengan data *customer* ke- i .

5. DATASET YANG DIGUNAKAN DAN SKENARIO PENGUJIAN

5.1. Dataset

Penelitian ini mengambil studi kasus data (*dataset*) yaitu pelanggan *Data Telekomunikasi* (DT) yaitu data *sample* pelanggan di suatu perusahaan telekomunikasi di Indonesia (kasus *dataset* [5]). Data Telekomunikasi merupakan data *sample* pelanggan dari suatu perusahaan telekomunikasi di kota di Indonesia yang merupakan data *real* (data *active* dan data *churn*). *Dataset* memiliki 23 atribut (seperti yang terdapat Lampiran B-1) dimana satu *record* mewakili satu pelanggan, jumlah *record* atau *instance* 48.384 *record*, dimana 48.009 *record* berkelas *active* (non *churn*) dan 375 *record* berkelas *churn*. Ini berarti rasio *imbalance* data sekitar 128:1. Data *churn* pada *dataset* DT ini bersifat *voluntary churn*. Sebelum digunakan pada perangkat lunak, data ini telah mengalami

beberapa proses *preprocessing* dengan bantuan Clementine.

5.2. Skenario pengujian

Pengujian dan analisis pada perangkat lunak ini dilakukan pada kesepuluh versi dari dataset Masing-masing versi data *training* di-*train* dengan menggunakan ketiga metode yaitu *pure-SVM*, *SMOTE-SVM* dan *CombinedSampling-EnSVM* dengan suatu *input* parameter SVM yang sama. Parameter SVM yang digunakan pada setiap *training* di sini, antara lain Tipe Kernel = Gaussian RBF, $C = 10$, dan *Gamma* RBF = 0.1. Alasan pemilihan kernel RBF adalah karena kernel ini dianjurkan untuk dipakai sebagai pilihan utama [9]. Selain itu kernel RBF secara umum mampu bekerja dan memberikan hasil performansi yang lebih baik jika dibandingkan dengan kernel umum yang lain dengan mengatur *setting* parameter masukan yang tepat [13]. Sedangkan parameter $C = 10$ merupakan parameter yang terbaik digunakan pada permasalahan *dataset* yang besar, *imbalance* dan kelas *biner* [14]. Nilai $\gamma = 0.1$ dipilih untuk mendapatkan σ (*sigma* adalah nilai ukuran jari-jari "hyperball" untuk klasifikasi yang mengelilingi data [13]) yang lebih besar.

Pada skenario uji metode *SMOTE-SVM* dan *CombinedSampling-EnSVM*, *input* jumlah *over-sampling* (T) pada parameter *SMOTE* yang ingin ditambahkan pada masing-masing versi data *training* dilakukan secara bertahap mulai dari tingkat terkecil sampai tingkat terbesar (jumlah kelas mayor dan minor pada masing-masing versi data tersebut *balance*). Jadi, pada kedua metode ini, pengujian dilakukan beberapa kali (tepatnya lima tingkat *over-sampling*) hingga data *balance*. *Input* jumlah k yang digunakan adalah 5. Alasannya adalah agar data sintetis yang dihasilkan masih memiliki *similarity* yang tidak jauh berbeda dengan data minor acuannya. Selain itu, nilai tersebut merupakan nilai yang digunakan pada referensi asli *SMOTE* [3]. Sedangkan *input* jumlah N di-*set* selalu sama dengan 10 (maksimal) agar dapat mewakili distribusi seluruh data mayor/negatif yang ada pada setiap versi data *training*.

Beberapa macam data untuk pengujian adalah sebagai berikut :

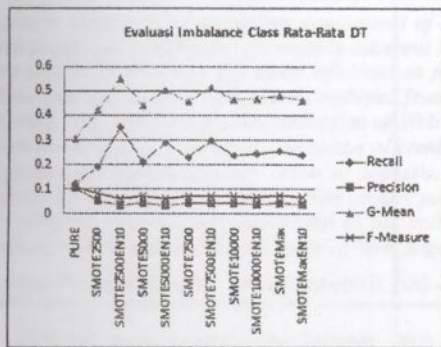
- PURE : tanpa dilakukan *sampling*
- SMOTE2500 : diterapkan *SMOTE* jumlah data minor 2500
- SMOTE5000: diterapkan *SMOTE* jumlah data minor 5000
- SMOTE7500: diterapkan *SMOTE* jumlah data minor 7500
- SMOTE10000: diterapkan *SMOTE* jumlah data minor 10000
- SMOTEMAX : diterapkan *SMOTE* jumlah data seimbang dengan data mayor
- SMOTE2500EN10 : diterapkan *SMOTE-Ensemble*, jumlah data minor 2500

- SMOTE5000EN10 : diterapkan SMOTE-Ensemble, jumlah data minor 5000
- SMOTE7500EN10 : diterapkan SMOTE-Ensemble, jumlah data minor 7500
- SMOTE10000EN10 : diterapkan SMOTE-Ensemble, jumlah data minor 10000
- SMOTEMAXEN10 : diterapkan SMOTE-Ensemble, jumlah data minor seimbang dengan Mayor

6. HASIL PENGUJIAN

6.1 Hasil Pengujian dan Analisis Imbalance Class

Evaluasi performansi imbalance class rata-rata dari kesepuluh hasil uji pada Data Telekomunikasi (DT) dapat dilihat pada gambar 5.



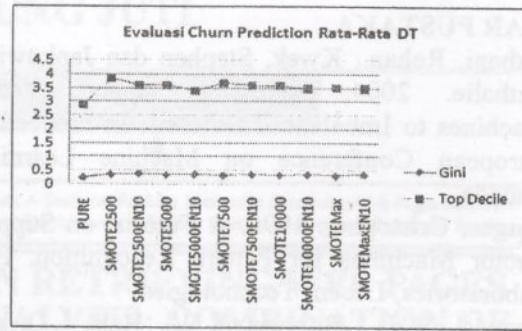
Gambar 5. Evaluasi imbalance class DT

Dari gambar 5 diatas, dapat dilihat bahwa nilai recall rata-rata yang terbaik (0.35) dihasilkan oleh teknik Ensemble yaitu SMOTE2500EN10. Hal ini bersesuaian dengan nilai recall yang terbaik dan terbanyak yang dihasilkan pada kesepuluh hasil uji seperti yang diuraikan sebelumnya. Karena nilai g-mean dipengaruhi oleh sensitivity/recall dan spesificity maka nilai g-mean rata-rata terbaik (0.55285) juga dihasilkan oleh teknik SMOTE2500EN10 juga.

Dari gambar 5 diatas, dapat dilihat juga bahwa nilai precision rata-rata terbaik (0.10462) dihasilkan oleh teknik pure. Hal ini disebabkan karena kebanyakan teknik pure pada kesepuluh versi selalu menghasilkan nilai precision terbaik, walaupun beberapa nilai terbaik dihasilkan oleh teknik SMOTE yaitu SMOTE2500 dan SMOTEMax. Karena nilai F-measure sangat dipengaruhi oleh precision, maka teknik yang menghasilkan precision rata-rata terbaik tersebut (pure) menghasilkan F-measure rata-rata terbaik (0.11084) juga.

6.2 Hasil Pengujian dan Analisis Churn Prediction

Hasil pengujian pada DT untuk evaluasi performansi churn prediction rata-rata dapat dilihat pada gambar 7.



Gambar 6. Evaluasi churn prediction DT

Karena pada tingkat over-sampling yang sama, seluruh teknik Ensemble memiliki nilai Gini Index rata-rata yang lebih baik dari teknik SMOTE dan seluruh teknik SMOTE lebih baik dari pure. Maka dapat dikatakan bahwa pada DT dalam hal performansi Gini Index secara umum, teknik Ensemble menghasilkan performansi yang lebih baik dari teknik SMOTE dan teknik SMOTE lebih baik dari teknik pure.

Sebaliknya pada tingkat over-sampling yang sama, seluruh teknik SMOTE memiliki nilai Top Decile rata-rata yang lebih baik dari Ensemble, dan seluruh teknik Ensemble lebih baik dari pure. Maka dapat dikatakan bahwa pada DT dalam hal performansi Top Decile secara umum, teknik SMOTE menghasilkan performansi yang lebih baik dari teknik Ensemble dan teknik Ensemble lebih baik dari teknik pure.

7. KESIMPULAN

Kesimpulan yang dapat diambil pada penelitian ini adalah:

1. Teknik Ensemble secara umum belum dapat meningkatkan akurasi kelas minor (F-measure).
2. Dilihat dari F-measure, teknik pure menghasilkan performansi yang lebih baik dari teknik SMOTE dan teknik SMOTE lebih baik dari teknik Ensemble.
3. Dalam hal precision dan F-measure, nilai precision umumnya sangat mempengaruhi nilai F-measure (dari nilai recall) pada kedua jenis data.
4. Dilihat dari Gini Index, teknik Ensemble menghasilkan performansi yang lebih baik dari teknik SMOTE dan teknik SMOTE lebih baik dari teknik pure. Dalam hal performansi Top Decile secara umum, teknik SMOTE menghasilkan performansi yang lebih baik dari teknik Ensemble dan teknik Ensemble lebih baik dari teknik pure.
5. Suatu teknik sampling yang menghasilkan performansi imbalance yang terbaik belum tentu menghasilkan performansi churn prediction yang terbaik pula.
6. Tingkat over-sampling SMOTE dan Ensemble yang semakin tinggi (balance) tidak menjamin performansi (ukuran evaluasi) semakin baik.

DAFTAR PUSTAKA

- [1] Akbani, Rehan., Kwek, Stephen dan Japkowicz, Nathalie. 2004. Applying Support Vector Machines to Imbalanced Datasets. In Proceeding European Conference on Machine Learning, Kanada
- [2] Burges, Cristopher. 1998. A Tutorial on Support Vector Machines for Pattern Recognition. Bell Laboratories, Lucent Technologies
- [3] Chawla, N.V., Bowyer, K.W., Hall, L.O. dan Kegelmeyer, W. P. 2002. SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research (JAIR)
- [4] Cristianini, Nello dan Shawe-Taylor, John. 2000. An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge University Press, UK.
- [5] Fahrudin, Tora. 2007. Analisis dan Implementasi Metode Databoost-IM (Studi Kasus Churn Prediction Mobile Telecommunication). Penelitian STT Telkom, Bandung
- [6] Gunn, R. Steve. 1998. *Support Vector Machines for Classsification and Regression*. Faculty of Engineering, School of Electronics and Computer Sciences
- [7] Han, Jiawei dan Kamber, Micheline. 2006. *Data Mining : Concepts and Techniques 2nd Edition*. University of Illinois, Urbana-Champaign..
- [8] Liu, Yang., An, Aijun dan Huang, Xiangji. 2006. *Boosting Prediction Accuracy on Imbalanced Datasets with SVM Ensembles*. Departement of Computer Science and Engineering, York University, Canada..
- [9] Olson, L. David dan Delen, Dursun. 2008. *Advanced Data Mining Technique*. Springer, USA..
- [10] Santosa, Budi. 2007. *Data Mining: Teknik Pemanfaatan Data untuk Keperluan Bisnis*. Graha Ilmu, Yogyakarta..
- [11] Satrio Nugroho, Anto. 2008. *Pengantar Support Vector Machine*. STT Telkom, Bandung..
- [12] Satrio Nugroho, Anto., Witarto, Arif Budi dan Handoko, Dwi. 2003. *Support Vector Machine: Teori dan Aplikasinya dalam Bioinformatika*. Ilmukomputer.com..
- [13] Setiawan, Eko Rufianto. 2007. *Prediksi Pelanggan 2G yang Berpotensi Beralih ke Layanan 3G pada Jaringan Telekomunikasi Bergerak dengan Metode Support Vector Machine (Studi Kasus PAKDD 2006 Kompetisi Data Mining)*. Penelitian STT Telkom, Bandung..
- [14] Shin, Hyunjung dan Cho, Sungzoon. 2004. *How To Deal with Large Dataset, Class Imbalance and Binary Output in SVM based Response Model*. Seoul National University. Seoul, Korea..
- [15] Tan, Pang-Ning., Steinbach, Michael dan Kumar, Vipin. 2005. *Introduction To Data Mining*. University of Minnesota, USA, Bandung..
- [16] Vapnik. 1998. *Statistical Learning Theory*. Springer, New York..
- [17] Wang, X. Benjamin dan Japkowicz, Nathalie. 2004. *Imbalanced Data Set Learning with Synthetic Examples..*