SMOTE: METODE PENYEIMBANG KELAS PADA KLASIFIKASI DATA MINING

Amalia Anjani Arifiyanti, Eka Dyar Wahyuni
Program Studi Sistem Informasi, Fakultas Ilmu Komputer
Universitas Pembangunan Nasional "Veteran" Jawa Timur
Email : amalia_anjani.fik@upnjatim.ac.id

Abstrak. *Kasus dengan kelas observasi yang memiliki kemunculan jarang seperti penipuan dan penyakit cenderung data yang muncul tidak seimbang antara satu kelas dengan kelas lain. Metode sampling merupakan salah satu metode untuk menangani ketidakseimbangan ini. Salah satu metode sampling yang digunakan adalah oversampling dengan SMOTE. Dengan metode ini, kelas minoritas direplikasi sebanyak kelas mayoritas. Keseimbangan data pada semua kelas berdampak pada performa model klasifikasi. Pada penelitian ini, model klasifikasi yang dihasilkan oleh logistic linear, KNN, dan Naive Bayes menunjukkan bahwa metode SMOTE meningkatkan performa model klasifikasi, sedangkan decision tree tidak menunjukkan hasil yang berbeda baik sebelum oversampling maupun setelah oversampling.*

***Kata Kunci**: **Data Mining, Imbalanced Class, Klasifikasi, Oversampling, SMOTE**.*

Klasifikasi, yang merupakan salah satu permasalahan dalam data mining yang diselesaikan dengan mempergunakan metode *supervised learning* sangat bergantung pada data latih. Data latih itu sendiri, jumlah distribusi data untuk masing-masing kelas sangat jarang memiliki jumlah yang sama. Dalam kondisi nyata, sangat sering ditemui, jumlah dataset masing-masing kelas berbeda [1]. Kondisi ini disebut dengan *imbalanced data* (ketidakseimbangan data).

Imbalanced data (ketidakseimbangan data) adalah salah satu masalah utama yang muncul dalam deteksi anomali pada dataset yang bersifat *real time*. Dataset dianggap tidak seimbang jika salah satu kelasnya memiliki dominasi yang sangat besar dibandingkan dengan kelas lainnya [2]. Ketidakseimbangan data ini biasanya muncul pada kasus prediksi penipuan, spam, penyakit, teroris, dsb. Hal ini muncul karena kasus spam yang ingin diprediksi jumlah kemunculannya sedikit jika dibandingkan dengan kasus yang non spam (kemunculannya dominan). Karena kemunculan kelas minor ini sangat minim, classifier yang dibangun menjadi kurang terlatih dan karenanya memberikan prediksi yang tidak akurat. Bahkan pada beberapa kasus multi class classifier, ketidakseimbangan data ini menghasilkan representasi yang rendah dari suatu data dan akhirnya data ini cenderung diabaikan sama sekali [3]. Sebagian besar, algoritma pengklasifikasi cenderung secara implisit menganggap bahwa data yang diproses memiliki distribusi yang seimbang, karenanya pengklasifikasi standar lebih condong kearah data yang jumlah kelasnya dominan.

Secara umum, ada 2 cara untuk menangani dataset yang tidak seimbang, yaitu di tingkat algoritmik atau tingkat data [2], [4], [5]. Pendekatan pada tingkat algoritmik adalah ketika algoritma *machine learning* dimodifikasi agar dapat mengakomodasi ketidakseimbangan data. Algoritma yang umumnya dimodifikasi adalah C4.5, Naïve Bayes, Random Forest, Neural Network K-Means [6], dan sebagainya.

Pendekatan pada tingkat data melibatkan resampling untuk mengurangi ketidakseimbangan kelas. Dua teknik pengambilan sampel dasar yang digunakan pada tingkat data adalah r*andom oversampling* (ROS) dan *random undersampling* (RUS). ROS akan menduplikasi secara acak data dari kelas yang minoritas [4], [7]. ROS bisa menjadi pilihan yang baik ketika data yang dimiliki tidak banyak, tetapi mungkin menyebabkan *overfitting* karena metode ini membuat salinan/duplikat yang sama persis dari data yang berasal dari kelas minoritas [7]. Sementara itu, untuk memodifikasi distribusi kelas, RUS akan membuang data (yang berasal dari kelas yang bersifat mayoritas) secara acak. Kekurangan dari RUS adalah dapat menyebabkan *underfitting*, karena

menghapus informasi yang mungkin berharga [7].

Secara umum, metode *oversampling* memberikan hasil yang lebih baik daripada metode under sampling [8]. Salah satu metode modifikasi dari *oversampling* adalah *Synthetic Minority Oversampling Technique* (SMOTE). Teknik ini mirip dengan ROS, perbedaannya ada pada sampel yang dihasilkan, tidak diduplikat secara random dari sampel yang sudah ada, tetapi sampel tersebut dibuat dengan mempergunakan konsep *nearest neighbour*. Beberapa penelitian sudah mencoba mengimplementasikan metode SMOTE ini dan juga beberapa modifikasinya [9] seperti FSMOTE [5], SMOTE-TL-ENN, SMOTE RSB, Borderline-SMOTE dan Safe Level SMOTE [10].

SMOTE

Metode ini diusulkan pertama kali pada tahun 2002 oleh Chawla, dimana kelas minoritas di-*oversampling*-kan dengan membuat "data training sintetis". Data training sintetis tersebut dibuat berdasarkan k-*nearest neighbor*. Pembangkitan data training sintetis yang berskala numerik berbeda dengan kategorik. Data numerik diukur berdasarkan jarak kedekatannya dengan jarak Euclidean sedangkan data kategorik dengan memperhitungkan nilai modusnya [11].

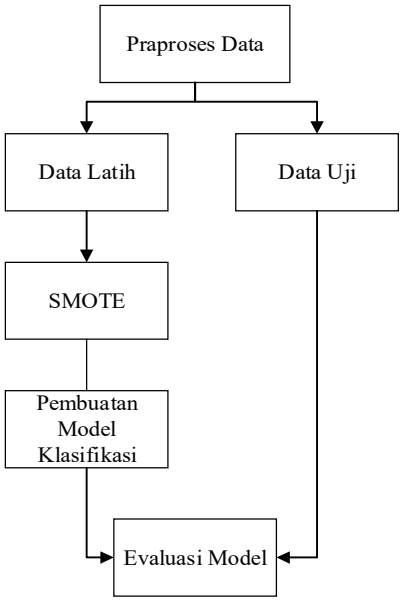
Pengukuran Evaluasi

Sebagian besar ukuran evaluasi untuk permasalahan klasifikasi mempergunakan *confussion matrix* [10], [11]. Dari *confussion matrix* ini, dapat diperoleh nilai akurasi, *precision* dan *recall*. Jika hanya mempertimbangkan nilai akurasi, model prediksi yang dihasilkan, tidak dapat digunakan di lingkungan produksi karena nilai ini sangat menyesatkan [13]. Karena akurasi bukanlah metrik terbaik untuk digunakan ketika mengevaluasi dataset yang tidak seimbang, metrik yang dapat memberikan wawasan yang lebih baik adalah *confussion matrix*, *precision* dan *recall* [10], [7].

Penelitian ini bertujuan untuk membuktikan apakah diperlukan teknik resampling jika dihadapkan pada situasi ketidakseimbangan data dengan cara menguji performa (*precision*, recall, F1 score, kurva ROC dan AUC) dari berbagai macam algoritma classifier.

I. Metodologi

Tahapan pada penelitian ini dapat dilihat pada gambar 1 berikut ini.



Gambar 1. Tahap Penelitian

Dataset dan Praproses Data

Dataset yang digunakan pada penelitian ini merupakan dataset yang dipublikasikan pada https://www.cs.purdue.edu/commugrate/data/credit_card/. DataminingContest2009.Task2.Train.Inputs.zip merupakan dataset untuk atribut prediksi dan DataminingContest2009.Task2.Train.Targets.zip untuk atribut kelas. Dataset tersebut merupakan data mengenai penipuan kartu kredit. Jumlah *instance*s dalam dataset berjumlah 1000, dengan total null berjumlah 1. Dataset memiliki 19 atribut yang digunakan sebagai atribut prediksi dan 1 atribut target klasifikasi. Dari 19 atribut prediksi, dua atribut diantaranya bersifat nominal. Pada atribut target klasifikasi terdapat dua kelas yaitu kelas 1 dan 0. Kelas 1 berarti transaksi penipuan dan 0 berarti bukan transaksi penipuan. Kelas 0 berjumlah 97346 dan kelas 1 berjumlah 2654. Proporsi kelas 0 dan kelas 1 adalah 97 : 3. Hal ini menunjukkan bahwa dataset tersebut tidak seimbang.

Dataset diproses terlebih dahulu sebelum dilakukan proses klasifikasi. Praproses pada penelitian ini dilakukan dengan cara menghapus *instance* yang

mengandung nilai Null, menghapus atribut yang tidak relevan dengan atribut target, dan terakhir melakukan encoding pada atribut yang bersifat nominal. Metode *encoding* menggunakan metode *one-hot encoding*. Hasil setelah praproses adalah kelas 0 berjumlah 97345 dan kelas 1 berjumlah 2654 dan jumlah atribut prediksi berjumlah 70 dengan 1 atribut target.

Pembagian Data

Data yang telah melalui tahap praproses data, kemudian dibagi menjadi dua dataset yaitu data latih dan data uji. Data latih digunakan untuk pembuatan model klasifikasi. Pada penelitian ini, model klasifikasi dibuat berdasarkan beberapa classifier yaitu Logistic Regression, K-Nearest Neighbor (KNN), Decision Tree, dan Gaussian Naive Bayes. Pembagian data mengunakan metode *hold-out* dengan proporsi data latih dan data uji adalah 70:30. Dari pembagian tersebut maka jumlah data latih dan data uji untuk masing-masing kelas dapat dilihat pada tabel 1.

Tabel 1 Proporsi Pembagian Data

	Data Latih (70%)	Data Uji (30%)
Kelas 0	68133	29212
Kelas 1	1866	788

Proses pembagian data dilakukan sebelum proses *oversampling* dengan metode SMOTE. Jika *oversampling* dilakukan sebelum pembagian data, maka data sintetis hasil penambahan data akan muncul pada data latih dan data uji. z

SMOTE

Oversampling digunakan pada data latih. Metode SMOTE digunakan untuk menambah data sintetis pada dataset kelas minor (dalam kasus ini adalah kelas 1), sehingga jumlah *instance*s pada kelas minoritas menjadi sama dengan jumlah *instance*s pada kelas mayoritas (dalam kasus ini adalah kelas 0). Hal ini dilakukan agar data pada kedua kelas seimbang. Data sintetis tersebut hanya ditambahkan pada data latih.

Evaluasi

Model klasifikasi yang telah dibangun kemudian dievaluasi dengan menggunakan metric pengukuran akurasi, *precision*, *recall*, dan *f-measure* serta ditambah dengan menggunakan metode pengukuran dengan menghitung AUC (*Area Under Curve*) pada ROC.

II. Hasil dan Pembahasan

Pembuatan model klasifikasi dilakukan dengan empat classifier yaitu logistic regression, KNN, decision tree, dan gaussian naive bayes. Hasil evaluasi dapat dilihat pada tabel 2 berikut ini.

Tabel 2 Evaluasi Model Klasifikasi

Jenis classifier	Akurasi	Precision	Recall	F1
Tanpa diseimbangkan				
Logistic regression	0.977	0.816	0.185	0.302
KNN	0.980	0.691	0.426	0.527
Decision tree	0.968	0.438	0.480	0.403
Naive Bayes	0.945	0.186	0.322	0.236
Diseimbangkan dengan SMOTE				
Logistic regression	0.748	0.070	0.703	0.128
KNN	0.872	0.124	0.640	0.207
Decision tree	0.962	0.395	0.475	0.338
Naive Bayes	0.829	0.080	0.525	0.139

Jika dilihat pada tabel hasil tersebut, model yang dihasilkan dari data yang tidak diseimbangkan memiliki nilai akurasi yang sangat tinggi dibandingkan dengan model yang dibangun dari data latih yang diseimbangkan dengan SMOTE. Akurasi yang tinggi tersebut namun dibarengi dengan nilai recall yang sangat rendah. Salah satu yang dapat dimaknai dalam hal ini adalah model yang dihasilkan tersebut mampu mengklasifikasikan kelas mayoritas dengan benar namun tidak mampu memprediksi kelas minoritas. Hal ini menunjukkan bahwa pada kasus kelas yang tidak seimbang, akurasi bukanlah alat pengukuran yang sesuai untuk menilai apakah model klasifikasi yang dihasilkan memiliki performa yang baik atau tidak untuk memprediksi kasus yang baru.

Hasil model pada jumlah kelas yang diseimbangkan dengan metode SMOTE, mengalami penurunan signifikan pada nilai akurasi kecuali pada classifier decision tree.

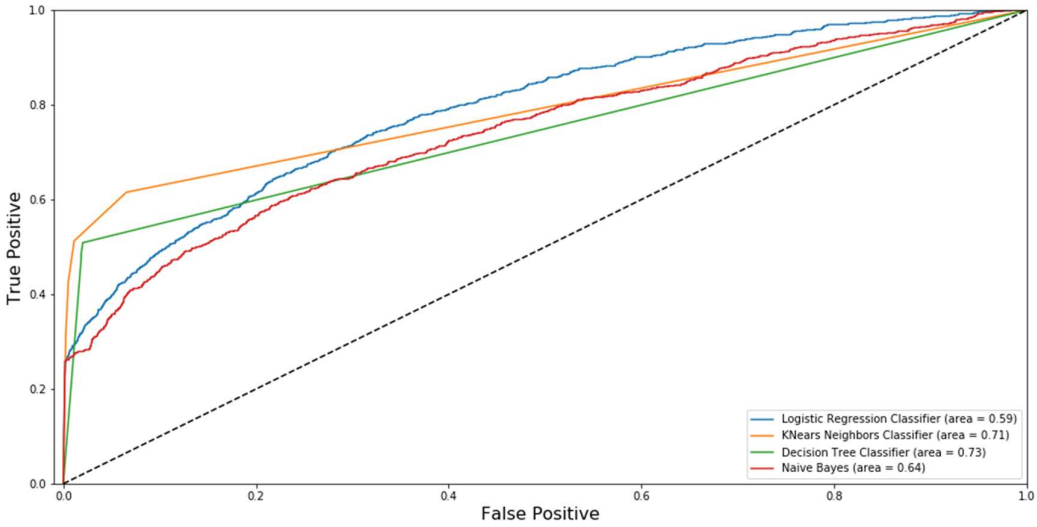
Penurunan signifikan ini juga terjadi pada nilai presisi akan tetapi nilai recall mengalami peningkatan cukup signifikan. Hal ini menunjukkan bahwa model tidak hanya condong terhadap kelas mayoritas, sedangkan pada model klasifikasi presisi dan recall sebaiknya memiliki keseimbangan yang ditunjukkan pada nilai f1. Jika dilihat pada hasil evaluasi performa tersebut maka dapat dinyatakan bahwa classifier decision tree memiliki performa yang stabil. Nilai akurasi tidak jauh berbeda pada data sebelum diseimbangkan dengan data yang telah diseimbangkan. Decision tree juga menghasilkan nilai precision dan recall yang cukup seimbang jika dibandingkan dengan model yang dihasilkan oleh classifier yang lain.

Evaluasi performa model klasifikasi juga dapat dinilai menggunakan metode kurva ROC dengan menghitung Area Under Curve (AUC). Metode kurva ROC ini menggambarkan true positive rate (TPR) dibandingkan dengan false positive rate (FPR). TPR merupakan sensitivity atau recall (*probability of detection*), sedangkan FPR dinyatakan sebagai *probability of false alarm*. Jika digambarkan dalam bentuk kurva ROC maka semakin mengarah ke pojok kiri atas semakin baik yaitu memiliki nilai TPR yang tinggi. Batas pada kurva ROC adalah 0.5 yang digambarkan sebagai garis diagonal. AUC adalah luas area di bawah curve ROC yang merupakan salah satu representasi dari kurva ROC. Model klasifikasi dinilai sempurna jika nilai AUC bernilai 1. Hasil model klasifikasi mengalami peningkatan AUC yang signifikan dari data yang belum diseimbangkan dengan data yang telah diseimbangkan. Nilai AUC pada model yang dihasilkan oleh logistic regression mengalami peningkatan paling tinggi, diikuti oleh KNN dan gaussian naive bayes. Hal ini berbeda dengan decision tree yang mengalami penurunan walaupun penurunan nilai AUC-nya tidak signifikan. Hasil AUC dari masing-masing classifier dapat dilihat pada tabel 3.

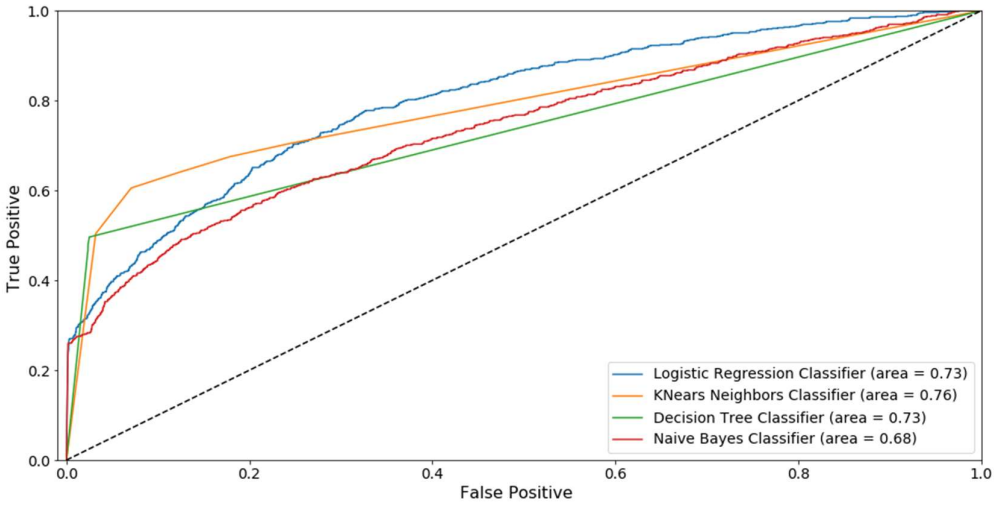
Berdasarkan beberapa model evaluasi performa model klasifikasi yang telah dilakukan baik pada data yang belum diseimbangkan maupun data yang telah diseimbangkan, dapat dinyatakan bahwa model klasifikasi yang dihasilkan berdasarkan data yang tidak seimbang bukanlah model klasifikasi yang baik. Untuk classifier logistic regression, KNN, dan gaussian naive bayes menghasilkan model klasifikasi yang lebih baik jika data diseimbangkan. Decision tree menghasilkan model klasifikasi yang tidak jauh berbeda baik pada saat data belum seimbang maupun data telah diseimbangkan. Decision tree menghasilkan model dengan performa yang baik walaupun data tidak seimbang. Hal ini terjadi karena decision tree mempelajari hirarki *if-else*, sehingga memaksa kedua kelas dipelajari dengan seimbang.

Tabel 3 Nilai AUC

Jenis classifier	AUC
Tanpa diseimbangkan	
Logistic regression	0.59
KNN	0.71
Decision tree	0.73
Naive Bayes	0.64
Diseimbangkan dengan SMOTE	
Logistic regression	0.73
KNN	0.76
Decision tree	0.72
Naive Bayes	0.68



Gambar 2 Kurva ROC Tanpa *Oversampling* dengan SMOTE



Gambar 3 Kurva ROC *Oversampling* dengan SMOTE

III. Kesimpulan

Ketidakseimbangan jumlah data pada masing-masing kelas dalam dataset yang digunakan untuk membuat model klasifikasi dapat menghasilkan model yang memiliki performa kurang baik karena model tersebut hanya mampu memprediksi kelas mayoritas dengan baik namun tidak dapat memprediksi kelas minoritas. Metode SMOTE dapat digunakan untuk menyeimbangkan data tersebut. Hasilnya adalah model klasifikasi yang telah diseimbangkan dengan metode SMOTE memiliki performa lebih baik dibandingkan jika tidak dilakukan penyeimbangan data. Hal tersebut berlaku untuk classifier logistic regression, KNN, dan naive bayes, akan tetapi decision tree tidak menunjukkan perubahan yang signifikan. Decision tree menghasilkan model yang memiliki performa baik pada kedua skenario yakni data tidak seimbang dan data yang telah diseimbangkan.

IV. Daftar Pustaka

[1] "How to handle Imbalanced Classification Problems in machine learning?" [Online]. Available: https://www.analyticsvidhya.com/blog/2017/03/imbalanced-classification-problem/. [Accessed: 23-Jan-2020].

[2] A. Somasundaram and U. S. Reddy, "Data Imbalance: Effects and Solutions for Classification of Large and Highly Imbalanced Data," Proc. 1st Int. Conf. Res. Eng. Comput. Technol. (ICRECT 2016), no. November, pp. 28–34, 2016.

[3] A. Fernández, V. López, M. Galar, M. J. Del Jesus, and F. Herrera, "Analysing the classification of imbalanced data-sets with multiple classes: Binarization techniques and ad-hoc approaches," Knowledge-Based Syst., vol. 42, pp. 97–110, Apr. 2013.

[4] B. W. Yap, K. A. Rani, H. A. Abd Rahman, S. Fong, Z. Khairudin, and N. N. Abdullah, "An application of *oversampling*, undersampling, bagging and boosting in handling imbalanced datasets," in Lecture Notes in Electrical Engineering, 2014, vol. 285 LNEE, pp. 13–22.

[5] A. Saifudin, S. Wahono, "Pendekatan Level Data untuk Menangani Ketidakseimbangan Kelas pada Prediksi Cacat Software", Journal of Software Engineering.", vol. 1, pp. 76-85, 2015.

[6] C. A. Sugianto, "Analisis Komparasi Algoritma Klasifikasi Untuk Menangani Data Tidak Seimbang Pada Data," Techno.COM, vol. 14, no. 4, pp. 336–342, 2015.

[7] "Dealing with Imbalanced Data - Towards Data Science." [Online]. Available: https://towardsdatascience.com/methods-for-dealing-with-imbalanced-data-5b761be45a18. [Accessed: 23-Jan-2020].

[8] R. Siringoringo, "Klasifikasi Data Tidak Seimbang Menggunakan Algoritma Smote dan K-Nearest Neighbor," vol. 3, no. 1, pp. 44–49, 2018.

[9] B. Santoso, H. Wijayanto, K. A. Notodiputro, and B. Sartono, "Synthetic *oversampling* Methods for Handling Class Imbalanced Problems : A Review," in IOP Conference Series: Earth and Environmental Science, vol. 58, no. 1, 2017.

[10] "Having an Imbalanced Dataset? Here Is How You Can Fix It." [Online]. Available: https://towardsdatascience.com/having-an-imbalanced-dataset-here-is-how-you-can-solve-it-1640568947eb. [Accessed: 23-Jan-2020].

[11] "Handling imbalanced datasets in machine learning - Towards Data Science." [Online]. Available: https://towardsdatascience.com/handling-imbalanced-datasets-in-machine-learning-7a0e84220f28. [Accessed: 23-Jan-2020