



# JEPIN

(Jurnal Edukasi dan Penelitian Informatika)

ISSN(e): 2548-9364 / ISSN(p) : 2460-0741

Vol. 6  
No. 3  
Desember  
2020

## Penerapan SMOTE untuk Mengatasi *Imbalance Class* dalam Klasifikasi *Television Advertisement Performance Rating* Menggunakan *Artificial Neural Network*

Edi Sutoyo<sup>#1</sup>, M. Asri Fadlurrahman<sup>#2</sup>

<sup>#</sup> Program Studi Sistem Informasi, Fakultas Rekayasa Industri, Telkom University

Jl. Telekomunikasi Jl. Terusan Buah Batu, Sukapura, Kec. Dayeuhkolot, Kota Bandung, Jawa Barat 40257

<sup>1</sup>edisutoyo@telkomuniversity.ac.id

<sup>2</sup>asrifadlurrahman@student.telkomuniversity.ac.id

**Abstrak**— Dalam data nyata, ada banyak situasi di mana jumlah instance di satu class jauh lebih sedikit daripada jumlah instance di class lain. Keadaan ini disebut sebagai masalah dataset tidak seimbang (*imbalance class*). Imbasnya kinerja klasifikasi biasanya menurun di beberapa aplikasi data mining. Pada penelitian ini, diidentifikasi bahwa dataset performansi rating iklan TV yang digunakan memiliki permasalahan *imbalance class* yang sangat besar dimana instance yang memiliki nilai rating tinggi, jauh lebih sedikit dibandingkan instance yang memiliki nilai rating kecil dan menengah. Sehingga diperlukan metode *over-sampling* untuk mengatasi permasalahan *imbalance class* tersebut. Metode yang dapat digunakan adalah *Synthetic Minority Over-sampling Technique (SMOTE)*. Untuk memvalidasi keefektifan model yang diusulkan, dilakukan dua skenario eksperimental yaitu: pertama algoritma ANN langsung digunakan untuk pemodelan tanpa mempertimbangkan ketidakseimbangan kelas, dan kedua dilakukan *over-sampling SMOTE* untuk meningkatkan jumlah dataset agar mencapai dataset yang seimbang. Hasil eksperimen menunjukkan bahwa performansi ANN+SMOTE mencapai akurasi sebesar 87.06% dibandingkan ANN yang hanya sebesar 86.35%. Penerapan Teknik SMOTE terbukti dapat mengatasi masalah ketidakseimbangan data dan mendapatkan hasil klasifikasi yang lebih baik.

**Kata kunci**— Artificial Neural Network, Classification, Data Mining, Imbalance Class, SMOTE

### I. PENDAHULUAN

Iklan adalah segala jenis presentasi non-pribadi dan penyiaran suatu ide/gagasan, barang, maupun jasa oleh penyedia sponsor tertentu yang membayar, melalui media komunikasi baik itu cetak ataupun elektronik [1]. Setiap perusahaan yang mengiklankan produk atau jasa yang mereka tawarkan pasti menginginkan keuntungan dari penjualan dimasa mendatang. Untuk mencapai hal tersebut, perusahaan harus bisa mengetahui efektivitas iklan mereka.

Salah satu parameter yang dapat digunakan untuk mengukur efektivitas iklan adalah dengan *rating* iklan.

*Rating* adalah suatu nilai persentase 0 sampai 100 dari estimasi pasar media yang dapat dijangkau oleh iklan yang disiarkan. *Rating* biasanya diukur dengan performa iklan di masa lampau ataupun dari survei yang dilakukan [2]. Besarnya nilai *rating* dapat mempengaruhi nilai jual produk yang diiklankan. Biaya pemasukan iklan diharapkan untuk dapat menutupi biaya produksi iklan yang besar. Oleh karena itu, penentuan *ads type* dan alokasi waktu yang tepat sangat berpengaruh terhadap kesuksesan iklan yang ditampilkan. Untuk mengetahui *ads type* maupun alokasi waktu pengiklanan yang tepat, dapat dilakukan prediksi *performance rating* suatu iklan dengan menggunakan teknik *data mining*.

*Data Mining* dapat digunakan untuk menyimpulkan suatu informasi dari sekumpulan data yang berjumlah besar. *Data mining* melibatkan ilmu statistika, matematika, dan kecerdasan untuk mengidentifikasi informasi ([3], [4]). Salah satu teknik yang bisa digunakan untuk mengidentifikasi informasi dengan *data mining* adalah klasifikasi. Klasifikasi dapat mengklasifikasikan setiap item pada suatu *dataset* ke dalam kelas-kelas atau kelompok yang sudah didefinisikan sebelumnya. Tujuannya adalah untuk memprediksikan pengelompokan setiap data yang ada pada *dataset* [5]. Klasifikasi telah banyak digunakan dan dikembangkan hasilnya berupa pembaruan teori maupun implementasi pada *real-world problems* ([6]–[11]). Pada penelitian ini, teknik klasifikasi akan digunakan untuk mengidentifikasi data iklan ke dalam beberapa jenis/kelas *rating*.

Berdasarkan Dokare dan Kant [12] dilakukan penelitian untuk menganalisis performa antara algoritma *Support Vector Machine (SVM)*, *k-Nearest Neighbour (k-NN)*, dan *Artificial Neural Network (ANN)*. Didapatkan hasil bahwa ANN merupakan metode yang lebih baik untuk digunakan dengan nilai akurasi maksimum 80%. Dewi [13]

melakukan penelitian perbandingan 5 algoritma data mining untuk prediksi keberhasilan pemasaran produk pada perbankan. Penelitian tersebut menunjukkan bahwa ANN menghasilkan akurasi tertinggi sebesar 89.71%. Penelitian lain yaitu Saritas dan Yasar [6] dilakukan analisa performa algoritma ANN dan *Naive Bayes* untuk klasifikasi mendapatkan hasil akurasi 86.95% untuk ANN.

Dari beragam algoritma klasifikasi *data mining*, salah satu metode klasifikasi yang sering digunakan adalah *Artificial Neural Network* (ANN) [13]. ANN merupakan suatu struktur komputasi yang terinspirasi oleh sistem jaringan saraf biologis. ANN terdiri atas beberapa proses yang sederhana tetapi saling berkoneksi yang disebut dengan neuron. ANN dapat mengkomputasikan data berstruktur melalui proses *learning* dan *training* [14]. Salah satu algoritma *learning* yang dapat digunakan untuk identifikasi klasifikasi menggunakan ANN adalah algoritma *Backpropagation*. Algoritma *Backpropagation* dapat memperkecil tingkat *error output* dengan cara mengubah nilai bobot-bobotnya dalam arah mundur (*backward*) sesuai dengan perbedaan *output* dengan target yang diinginkan [15].

*Dataset* yang digunakan pada penelitian ini diperoleh dari sebuah perusahaan yang bergerak pada periklanan. *Dataset* yang digunakan berisikan data historis *rating* iklan yang diiklankan pada perusahaan tersebut yang bergerak sebagai perusahaan agensi iklan. Dalam data nyata, ada banyak situasi di mana jumlah *instance* di satu *class* jauh lebih sedikit daripada jumlah *instance* di *class* lain. Fenomena ini disebut sebagai masalah *imbalance class*, yang berdampak menurunnya kinerja algoritma klasifikasi di beberapa aplikasi data mining termasuk pengenalan pola medis, manajemen telekomunikasi, bioinformatika, dan kategorisasi teks. Sebagian besar model konvensional menetapkan kelas mayoritas ke data dan mengabaikan kelas minoritas karena kemiringan data. Untuk mengatasi permasalahan *imbalance class*, salah satu metode yang digunakan adalah *sampling*. Metode *sampling* melakukan modifikasi terhadap distribusi data antar kelas mayoritas dan kelas minoritas pada dataset *training* untuk menyeimbangkan jumlah data tiap kelas ([16]–[18]).

Pada tahap *preprocessing* data, diidentifikasi bahwa dataset yang digunakan pada penelitian ini memiliki permasalahan *imbalance class* yang sangat besar dimana *instance* data yang memiliki nilai *rating* besar (>5) jauh lebih sedikit dibandingkan *instance* data yang memiliki nilai *rating* kecil/menengah (<5). Sehingga diperlukan metode *preprocessing* untuk mengatasi permasalahan *imbalance class* tersebut. Salah satu teknik *over-sampling* yang dapat digunakan adalah *Sythetic Minority Over-sampling Technique* (SMOTE). SMOTE dapat melakukan duplikasi data secara sintesis sehingga permasalahan distribusi data yang berbeda dapat diatasi [19]. Untuk memvalidasi keefektifan metode yang diusulkan, dilakukan dua skenario eksperimental: pertama, algoritma ANN langsung digunakan untuk pemodelan tanpa mempertimbangkan ketidakseimbangan kelas. Kemudian untuk skenario kedua, *over-sampling* SMOTE digunakan

untuk meningkatkan jumlah dataset untuk mencapai dataset yang seimbang [2].

Berdasarkan penelitian sebelumnya didapatkan bahwa ANN lebih baik digunakan daripada algoritma yang lainnya. Oleh karena itu, penelitian ini menggunakan ANN untuk melakukan prediksi *rating* performa iklan televisi. Dengan adanya model prediksi ini, diharapkan bisa mendapatkan hasil prediksi yang dapat dijadikan bahan pertimbangan dalam pengambilan keputusan pembuatan iklan pada periklanan tersebut.

## II. LANDASAN TEORI

### A. Data Mining

*Data mining* adalah proses pemecahan masalah dengan cara menganalisis data untuk menemukan pola atau informasi yang tersembunyi dalam sekumpulan data. Proses-proses ini harus berupa proses yang otomatis atau semi-otomatis. Pola atau informasi yang diperoleh dapat digunakan untuk menjelaskan data kemudian membuat prediksi berdasarkan pola atau informasi tersebut [20].

*Data mining* juga sering dikenali dengan istilah *Knowledge Discovery in Database* (KDD) yang merupakan langkah penting dalam penemuan pengetahuan. Langkah – langkah yang dilakukan didalam proses *data mining* adalah sebagai berikut [3] :

1. *Data Cleaning* (menghilangkan *noise* pada data dan data yang tidak konsisten).
2. *Data Integration* (proses untuk menggabungkan data dari beragam sumber data menjadi satu).
3. *Data Selection* (proses dimana dilakukan pemilihan data yang relevan untuk dilakukan analisisnya).
4. *Data Transformation* (proses mentransformasikan data kedalam bentuk yang siap sebagai masukan algoritma *data mining*).
5. *Data Mining* (proses utama dimana algoritma *data mining* diaplikasikan untuk menemukan pola pada data).
6. *Pattern Evaluation* (proses untuk mengidentifikasi pola yang menarik dari data).
7. *Knowledge Presentation* (proses dimana hasil *data mining* divisualisasikan untuk ditampilkan).

### B. Klasifikasi

Klasifikasi adalah salah satu fungsi dari data mining untuk mengelompokkan suatu item data kedalam kategori atau kelas-kelas yang sudah didefinisikan terlebih dahulu dengan tujuan untuk memprediksi secara akurat kelas/kategori yang sesuai untuk seluruh data yang ada pada dataset [5]. Pada klasifikasi, terdapat suatu variabel yang berfungsi sebagai label target. Model *data mining* memeriksa sekumpulan *record*, dimana tiap *record* menyimpan variabel label dari *record* tersebut, dan juga variabel *input* ataupun variabel prediktor [21].

Pertama, algoritma klasifikasi akan menguji *dataset training* yang berisikan variabel prediktor yang memiliki variabel label yang sudah diklasifikasi terlebih dahulu. Dengan demikian, algoritma akan “mempelajari”

hubungan antara kombinasi variabel prediktor dengan variabel label. Kemudian, algoritma akan diberikan *dataset testing* yang hanya berisikan variabel prediktor yang tidak memiliki variabel label. Algoritmanya kemudian melakukan klasifikasi terhadap *dataset testing* berdasarkan “pengalaman” algoritma saat mengklasifikasi *dataset training* [21].

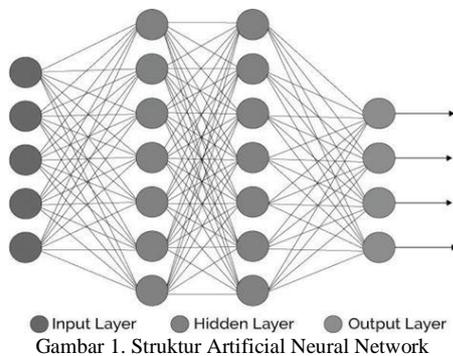
C. Artificial Neural Network (ANN)

ANN adalah suatu struktur komputasi yang didasarkan oleh sistem jaringan saraf biologis. ANN terdiri atas beberapa proses yang sederhana tetapi saling berkoneksi yang disebut dengan neuron. Proses ini terdiri dari pengumpulan data, analisis dan pemrosesan data, desain struktur jaringan, jumlah lapisan yang tersembunyi, inisialisasi, pelatihan jaringan, simulasi jaringan, penyesuaian bobot/bias, dan terakhir pengujian jaringan. ANN biasa digunakan untuk memproses kumpulan data yang besar, untuk dapat menyediakan informasi analisis yang dapat berguna untuk memprediksikan ataupun identifikasi klasifikasi suatu data [14].

Fungsi aktivasi merupakan fungsi yang digunakan pada jaringan saraf untuk mengaktifkan atau tidak mengaktifkan neuron. Karakteristik yang harus dimiliki oleh fungsi aktivasi jaringan perambatan balik antara lain harus kontinu, terdiferensial, dan tidak menurun secara monoton (*monotonically non-decreasing*) [22]. Fungsi aktivasi dibutuhkan oleh *hidden layer* pada ANN untuk membuat ANN menjadi non linear. Fungsi aktivasi dapat berupa fungsi linear, threshold, atau sigmoid. Fungsi sigmoid biasa digunakan karena fungsi ini mengombinasikan *nearly linear behavior*, *curvilinear behavior*, dan *nearly constant behavior* [21]. Fungsi aktivasi sigmoid dapat diketahui sebagai berikut:

$$y = \sigma(x) = \frac{1}{(1 + \exp^{-x})} \quad (1)$$

Setiap pola informasi *input* maupun *output* yang dimasukkan kedalam ANN akan diproses ke dalam neuron. Neuron tersebut terkumpul dalam 3 *layer/lapisan* seperti pada Gambar 1 berikut [14] :



1. Input layer, pada lapisan ini berisikan neuron - neuron yang digunakan untuk menerima input untuk proses *learning* dan *recognition* dari ANN.

2. Hidden Layer, pada lapisan ini berisikan neuron - neuron yang digunakan untuk mengganti kontribusi yang digunakan oleh neuron pada output layer.
3. Output Layer, pada lapisan ini berisikan neuron - neuron yang akan bereaksi terhadap data input yang diberikan.

*Backpropagation* merupakan algoritma pembelajaran yang terawasi dan biasanya digunakan oleh *perceptron* dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan neuron-neuron yang ada pada lapisan tersembunyinya. Algoritma *backpropagation* menggunakan *error output* untuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*). Untuk mendapatkan *error* ini, tahap perambatan maju (*feedforward*) harus dikerjakan terlebih dahulu [23], [24].

Komputasi *feed-forward* atau *forward pass* adalah tahap yang berupa dua proses. Proses pertama adalah untuk menghitung nilai *output* dari neuron pada *hidden layer*, dan proses kedua adalah untuk menghitung nilai *output* dari neuron pada *output layer*. Nilai *output* pada tiap neuron dapat dihitung dengan rumus sebagai berikut [25] :

$$y = \sigma \left( \sum_i^j W_{ij} X_{ij} \right) \quad (2)$$

Keterangan:

- $\sigma$  = fungsi sigmoid
- $W$  = nilai bobot antara neuron
- $X$  = nilai dari neuron *input*
- $y$  = nilai *output* prediksi ANN.

Langkah selanjutnya adalah menghitung nilai *error* dari nilai *output* pada neuron yang ada pada *output layer*. Penghitungan nilai *error* dapat menggunakan nilai *Mean Square Error* (MSE). MSE adalah nilai *error* rata-rata dari seluruh *record* (neuron pada *layer* penghitungan) pada ANN dirumuskan sebagai berikut [26]:

$$\delta_{ij} = \frac{\sum (t_{ij} - y_{ij})^2}{\text{jumlah record}} \quad (3)$$

Keterangan:

- $\delta_{ij}$  = *Error* pada neuron *ij*
- $t_{ij}$  = Nilai *Output* target pada neuron *ij*
- $y_{ij}$  = nilai *output* prediksi ANN pada neuron *ij*.

Setelah nilai *error* diketahui, nilai tersebut akan digunakan untuk *backward propagation* dan mengubah nilai bobot. Pertama, nilai *error* akan dipropagasikan dari *output layer* ke *hidden layer* sehingga nilai bobot antara setiap neuron pada *hidden layer* dan *output layer* akan di diubah terlebih dahulu. Sebelum nilai bobot dapat diubah, dilakukan penghitungan nilai koreksi bobotnya [26] :

$$\Delta W_{ij} = \alpha \delta_{ij} y_{ij} \quad (4)$$

Keterangan :

- $\Delta W_{ij}$  = nilai koreksi bobot
- $\alpha$  = nilai learning rate

Langkah selanjutnya adalah melakukan propagasi balik dari *hidden layer* ke *input layer* untuk memperoleh nilai

error sehingga dapat digunakan untuk mengubah nilai bobot antara setiap neuron pada *input layer* dan *hidden layer*. Hal yang paling utama untuk diingat adalah untuk tidak melakukan perubahan nilai bobot sebelum semua nilai *error* telah dihitung terlebih dahulu. Nilai bobot yang baru dapat diperoleh dengan menambahkan nilai bobot yang lama dengan nilai koreksi bobot [26] :

$$W_{ij}(new) = W_{ij}(old) + \Delta W_{ij} \quad (5)$$

Keempat tahapan ini kemudian dilakukan proses iterasi hingga ANN berhasil memenuhi *Stopping Criteria*, yaitu kondisi yang membuat algoritma akan berhenti melakukan proses *training* [21].

D. Imbalance Class

*Imbalance class* adalah kondisi distribusi antar kelas yang tidak seimbang pada suatu dataset, dimana salah satu kelasnya memiliki jumlah data yang sangat besar (kelas mayoritas) dibanding kelas lainnya (kelas minoritas) [27]. Perbedaan jumlah data yang besar antar kelas dapat mengakibatkan model klasifikasi sering tidak dapat memprediksikan kelas minoritas dengan tepat sehingga banyak data tes yang seharusnya berada pada kelas minoritas diprediksikan salah oleh model klasifikasi [28].

Untuk mengatasi permasalahan *imbalance class*, salah satu metode yang digunakan adalah *sampling*. Metode *sampling* melakukan modifikasi terhadap distribusi data antar kelas mayoritas dan kelas minoritas pada dataset *training* untuk menyeimbangkan jumlah data tiap kelas [29]. Salah satu metode *sampling* yang sering digunakan adalah *Syntetic Minority Over-sampling Technique* (SMOTE) [19].

E. Synthetic Minority Over-sampling Technique (SMOTE)

SMOTE adalah metode *over-sampling* dimana data pada kelas minoritas diperbanyak dengan menggunakan data sintetik yang berasal dari replikasi data pada kelas minoritas. *Over-sampling* pada SMOTE mengambil *instance* dari kelas minoritas lalu mencari *k-nearest neighbor* dari setiap *instance*, kemudian menghasilkan *instance* sintetik daripada mereplikasi *instance* kelas minoritas; oleh karena itu, dapat menghindari masalah *overfitting* yang berlebihan [19].

Algoritma yang bekerja pada SMOTE pertama akan mengambil nilai selisih antara vektor dari fitur pada kelas minoritas dan nilai *nearest neighbor* dari kelas minoritas lalu mengalikan nilai tersebut dengan angka acak antara 0 sampai 1. Selanjutnya, hasil kalkulasi tersebut ditambahkan dengan vektor fiturnya sehingga didapatkan hasil nilai vektor yang baru [30].

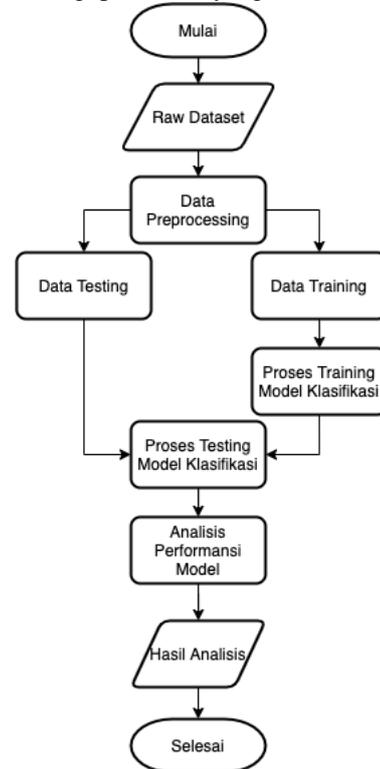
$$X_{new} = X_i + (\hat{X}_i - X_i) \times \delta \quad (6)$$

Keterangan :

- $X_i$  = vektor dari fitur pada kelas minoritas
- $\hat{X}_i$  = *k-nearest neighbors* untuk  $X_i$
- $\delta$  = angka acak antara 0 sampai 1

III. METODOLOGI PENELITIAN

Dalam pembuatan model klasifikasi, metode yang digunakan adalah ANN. Klasifikasi *tv rating* pada penelitian ini menggunakan data rating iklan yang diperoleh dari PT. XYZ yang bergerak dalam industri periklanan, dengan jumlah data sebanyak 22.218 *instance*. Pada Gambar 2 berikut memberikan gambaran umum tentang metodologi penelitian yang dilakukan.



Gambar 2. Metodologi penelitian

IV. HASIL DAN PEMBAHASAN

A. Pengumpulan Data

Pengumpulan data yang akan diolah pada penelitian ini berasal dari data yang diperoleh pada aplikasi *vislog* PT. XYZ, yang merupakan data performa iklan pada kategori *instant noodle* periode bulan Oktober tahun 2019 sampai dengan bulan Desember tahun 2019. Distribusi data yang digunakan pada penelitian ini dapat ditunjukkan pada Tabel 1.

TABEL I  
JUMLAH DATA PER-BULAN

Bulan	Jumlah Baris
Oktober	6.735
November	6.441
Desember	9.042
<b>Total</b>	<b>22.218</b>

Dari data yang diperoleh terdapat atribut yaitu tanggal, *channel*, program, produk, jenis iklan, waktu mulai, durasi,

biaya, dan TVR. Penjelasan dari atribut yang diperoleh akan dijelaskan pada Tabel 2.

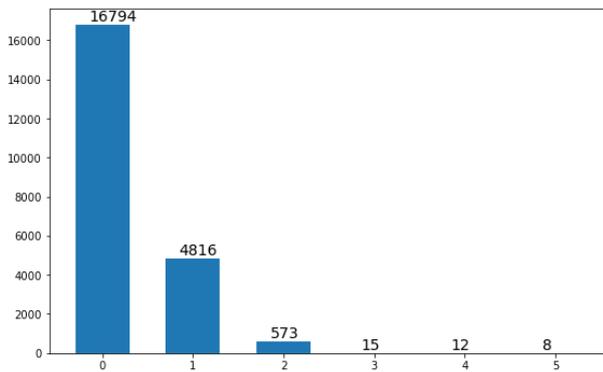
TABEL II  
PENJELASAN ATRIBUT DATA

No	Atribut	Penjelasan
1	Date	Tanggal penayangan dari iklan tersebut
2	Channel	Channel televisi yang menayangkan iklan tersebut
3	Program	Program televisi yang menayangkan iklan tersebut
4	Product	Nama produk yang diiklankan
5	Ads Type	Jenis iklan yang ditayangkan
6	Start Time	Waktu penayangan iklan dimulai
7	Duration	Durasi lama penayangan iklan
8	Cost	Biaya penayangan iklan
9	TVR	Rating performa iklan yang ditayangkan

Pada tahapan ini, dilakukan seleksi atribut yang akan digunakan pada penelitian ini. Metode *information gain* digunakan untuk menentukan keterkaitan antar atribut dengan label kelasnya. Atribut yang memiliki nilai *gain* di bawah 0.01 akan dihilangkan dan tidak akan digunakan pada penelitian ini. Dari hasil *information gain* yang ditampilkan pada Tabel 3, maka kolom atribut tidak akan digunakan dalam penelitian ini.

TABEL III  
HASIL INFORMATION GAIN

Atribut	Information Gain
Program	0,450532
Cost	0,326123
Start Time	0,226721
Channel	0.139214
Product	0.032163
Ads Type	0.021041
Duration	0.015359



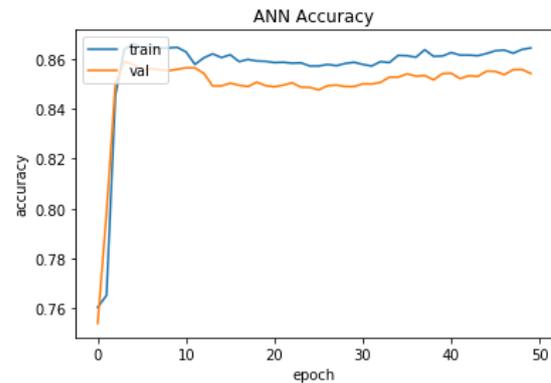
Gambar 3. Perbandingan jumlah instance setiap class

Pada Gambar 3, dapat dilihat bahwa terdapat sebaran distribusi data antar kelas yang sangat besar, dimana kelas TVR = 1 memiliki jumlah sampel sebanyak 13.483 data sedangkan kelas TVR = 5 hanya memiliki jumlah sampel sebanyak 11 data. Dengan menggunakan SMOTE, maka

sebaran distribusi data dapat menjadi seimbang karena pada kelas yang jumlah sampelnya lebih sedikit (kelas minoritas) akan diperbanyak jumlahnya dengan data sintetik yang dibuat oleh SMOTE.

Untuk menetapkan model klasifikasi dan mengujinya, setiap set data pertama-tama dibagi menjadi dua bagian, yaitu 80% digunakan untuk set pelatihan dan 20% digunakan untuk set pengujian. Set pelatihan digunakan untuk membangun model, sedangkan set pengujian digunakan untuk penilaian kinerja yang valid. Sebaran jumlah class dari dataset untuk masing-masing class sangat tidak seimbang, yaitu untuk class=0 berjumlah 16794 atau sebesar 75.587% dari total data, class=1 berjumlah 4816 atau sebesar 21.676%, kemudian untuk class=2 berjumlah 573 data atau menyumbang 2.579%, class=3 berjumlah 15 atau 0.068%, sedangkan untuk class=4 hanya berjumlah 12 data atau 0.054% dan untuk class=5 hanya terdapat 8 data atau hanya 0.036%.

Untuk memvalidasi keefektifan model yang diusulkan, dilakukan dua skenario eksperimental: pertama, algoritma ANN langsung digunakan untuk pemodelan tanpa mempertimbangkan ketidakseimbangan kelas. Kemudian untuk skenario kedua, *over-sampling* SMOTE [19] digunakan untuk meningkatkan jumlah dataset untuk mencapai dataset yang seimbang. Teknik *over-sampling* ini dipilih karena telah diterapkan secara luas pada masalah ketidakseimbangan kelas. Kedua skenario tersebut kemudian dievaluasi menggunakan metrik evaluasi, yaitu *accuracy*, *precision*, *recall* dan *F1-score*. Model ANN yang diusulkan diimplementasikan di Win-Python 3.6.6 menggunakan library Keras. Semua observasi dilakukan pada komputer dengan Sistem Operasi Windows 10, Processor 2.5 GHz Quad-Core Intel Core i7, dan menggunakan RAM sebesar 16 GB.

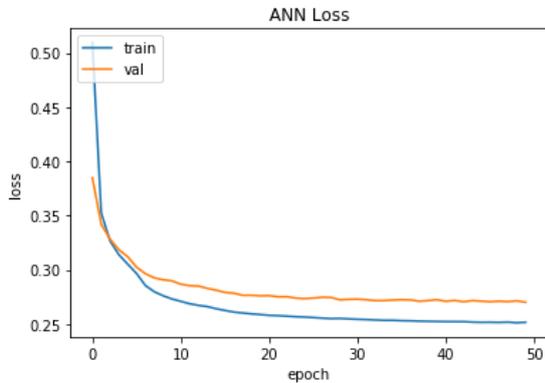


Gambar 4. Hasil Accuracy pada Model ANN

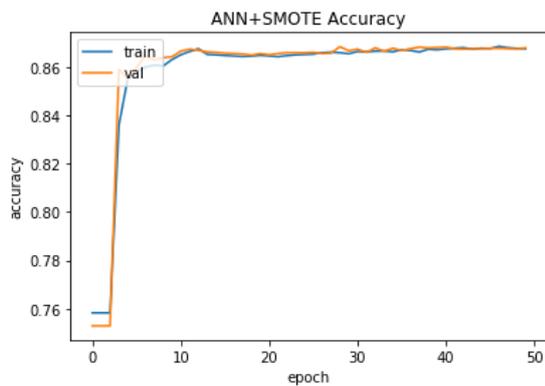
Hasil pada skenario pertama yang ditunjukkan pada Gambar 4 dan Gambar 5, menunjukkan bahwa model ANN untuk klasifikasi iklan rating TV cukup efektif dengan menghasilkan akurasi sebesar 86.35% dan loss sebesar 0.2630 pada proses pelatihan, sedangkan pada proses validasi menghasilkan akurasi sebesar 85.04 % dan loss sebesar 0.2873.

Pada skenario kedua yang ditampilkan pada Gambar 6 dan Gambar 7 menunjukkan bahwa model

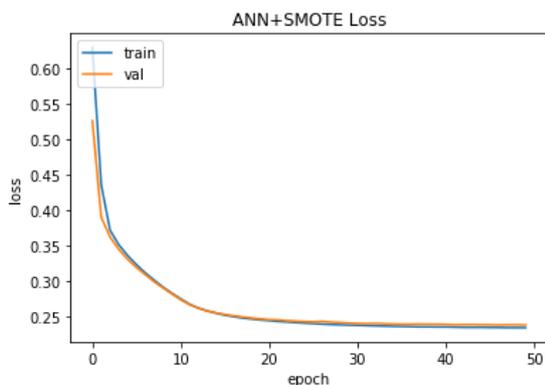
ANN+SMOTE untuk klasifikasi *rating* iklan TV mampu memberikan hasil yang lebih baik dibandingkan dengan model ANN. Model ANN+SMOTE mampu menghasilkan akurasi sebesar 87.08 % dan *loss* sebesar 0.2451 pada proses *training*. Sedangkan pada proses validasi, model ANN+SMOTE mampu menghasilkan akurasi sebesar 87.06% dan *loss* sebesar 0.2483. Sehingga, dengan menggunakan teknik SMOTE dapat meningkatkan akurasi sebesar 0.73% dan menurunkan *loss* sebesar 0.0179 pada proses pelatihan, serta dapat meningkatkan akurasi sebesar 2.2% dan menurunkan *loss* sebesar 0.039 pada proses validasi.



Gambar 5. Hasil *Loss* pada Model ANN



Gambar 6. Hasil *Accuracy* pada Model ANN+SMOTE



Gambar 7. Hasil *Loss* pada Model ANN+SMOTE

Pada skenario kedua menunjukkan bahwa teknik SMOTE selain meningkatkan akurasi dan menurunkan *loss* pada proses *training* maupun validasi, juga dapat mengurangi selisih jarak akurasi antara hasil *training* dengan hasil validasi yang hanya terdapat selisih 0.02% dan juga pada *loss* yang hanya terdapat selisih 0.0032.

TABEL IV  
HASIL EKSPERIMEN SECARA KESELURUHAN

Model Algoritma	Metriks Evaluasi			
	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
ANN	86.35	79.0	84.5	81.0
ANN+SMOTE	87.06	81.0	86.0	82.5

Secara keseluruhan, hasil eksperimen ditunjukkan pada Tabel 4. Tabel tersebut menunjukkan bahwa teknik gabungan yaitu ANN+SMOTE mempunyai kinerja lebih baik daripada teknik yang hanya menggunakan ANN ketika dievaluasi menggunakan metriks evaluasi yaitu *accuracy*, *precision*, *recall* dan *F1-Score*.

V. KESIMPULAN

Berdasarkan langkah penelitian yang telah dilakukan menunjukkan bahwa model ANN untuk klasifikasi performansi *rating* iklan TV cukup efektif dengan menghasilkan akurasi sebesar 86.35% dan *loss* sebesar 0.2630 pada proses pelatihan, sedangkan pada proses validasi menghasilkan akurasi sebesar 85.04% dan *loss* sebesar 0.2873. Sedangkan hasil eksperimen model ANN+SMOTE mampu memberikan hasil yang lebih baik dibandingkan dengan model ANN. Model ANN+SMOTE mampu menghasilkan akurasi sebesar 87.08% dan *loss* sebesar 0.2451 pada proses *training*. Sedangkan pada proses validasi, model ANN+SMOTE mampu menghasilkan akurasi sebesar 87.06% dan *loss* sebesar 0.2483. Sehingga dapat disimpulkan bahwa penggunaan teknik SMOTE mampu meningkatkan akurasi sebesar 0.73% dan menurunkan *loss* sebesar 0.0179 pada proses pelatihan, serta meningkatkan akurasi sebesar 2.2% dan menurunkan *loss* sebesar 0.039 pada proses validasi untuk *dataset* yang terdapat *imbalance class*.

REFERENSI

- [1] P. Kotler and K. L. Keller, *MarkKotler, P., & Keller, K. L.* 2016.
- [2] J. Pych, "Must Know Advertising Terms and Metrics | Bionic Advertising Systems," *Bionic*, 2016. .
- [3] J. Han, *Data Mining: Concepts and Techniques (The Morgan Kaufmann Series in Data Management Systems)*. 2011.
- [4] J. Han, M. Kamber, and J. Pei, *Data mining: Data mining Concepts and Techniques*, 3rd ed. Morgan Kaufmann, 2012.
- [5] G. Kesavaraj and S. Sukumaran, "A study on classification techniques in data mining," in *2013 4th International Conference on Computing, Communications and Networking Technologies, ICCCNT 2013*, 2013, doi: 10.1109/ICCCNT.2013.6726842.
- [6] M. M. Saritas and A. Yasar, "Performance Analysis of ANN and Naive Bayes Classification Algorithm for Data Classification," *Int. J. Intell. Syst. Appl. Eng.*, vol. 7, no. 2, pp. 88–91, Jun. 2019, doi: 10.18201/ijisae.2019252786.
- [7] E. Sutoyo, A. Almaarif, and others, "Educational Data Mining for Predicting Student Graduation Using the Naive Bayes

- Classifier Algorithm,” *J. RESTI (Rekayasa Sist. Dan Teknol. Informatika)*, vol. 4, no. 1, pp. 95–101, 2020.
- [8] E. Sutoyo, R. R. Saedudin, I. T. R. Yanto, and A. Apriani, “Application of adaptive neuro-fuzzy inference system and chicken swarm optimization for classifying river water quality,” in *Proceeding - 2017 5th International Conference on Electrical, Electronics and Information Engineering: Smart Innovations for Bridging Future Technologies, ICEEIE 2017*, 2018, vol. 2018-January, pp. 118–122, doi: 10.1109/ICEEIE.2017.8328774.
- [9] R. Rachmat and S. Suhartono, “Comparative Analysis of Single Exponential Smoothing and Holt’s Method for Quality of Hospital Services Forecasting in General Hospital,” *Bull. Comput. Sci. Electr. Eng.*, vol. 1, no. 2, pp. 80–86, Aug. 2020, doi: 10.25008/bcsee.v1i2.8.
- [10] H. Mudia, “Back Propagation Neural Network for Controlling Coupled Water Tank,” *Bull. Comput. Sci. Electr. Eng.*, vol. 1, no. 1, pp. 12–18, 2020, doi: 10.25008/bcsee.v1i1.4.
- [11] S. N. W. Ahmad, M. A. Ismail, E. Sutoyo, S. Kasim, and M. S. Mohamad, “Comparative Performance of Machine Learning Methods for Classification on Phishing Attack Detection,” *Int. J.*, vol. 9, no. 1.5, 2020.
- [12] I. Dokare and N. Kant, “Performance Analysis of SVM, k-NN and BPNN Classifiers for Motor Imagery,” *Int. J. Eng. Trends Technol.*, vol. 10, 2014.
- [13] S. Dewi, “KOMPARASI 5 METODE ALGORITMA KLASIFIKASI DATA MINING PADA PREDIKSI KEBERHASILAN PEMASARAN PRODUK LAYANAN PERBANKAN,” *None*, 2016.
- [14] A. El-Shahat, *Advanced Applications for Artificial Neural Networks*. 2018.
- [15] S. Haykin, *Neural Networks and Learning Machines*. 2008.
- [16] R. Barandela, R. M. Valdovinos, J. S. Sánchez, and F. J. Ferri, “The imbalanced training sample problem: Under or over sampling?,” in *Joint IAPR international workshops on statistical techniques in pattern recognition (SPR) and structural and syntactic pattern recognition (SSPR)*, 2004, pp. 806–814.
- [17] G. E. Batista, R. C. Prati, and M. C. Monard, “A study of the behavior of several methods for balancing machine learning training data,” *ACM SIGKDD Explor. Newsl.*, vol. 6, no. 1, pp. 20–29, 2004.
- [18] H. M. Nguyen, E. W. Cooper, and K. Kamei, “Borderline over-sampling for imbalanced data classification,” *Int. J. Knowl. Eng. Soft Data Paradig.*, vol. 3, no. 1, pp. 4–21, 2011.
- [19] N. V Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic Minority Over-sampling Technique,” *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, 2002.
- [20] I. H. Witten, E. Frank, and M. a Hall, *Data Mining: Practical Machine Learning Tools and Techniques (Google eBook)*. 2011.
- [21] D. T. Larose, *Discovering Knowledge in Data: An Introduction to Data Mining*. 2005.
- [22] E. B. Nababan and M. Zarlis, “Analisis Fungsi Aktivasi Sigmoid Biner Dan Sigmoid Bipolar Dalam Algoritma Backpropagation Pada Prediksi Kemampuan Siswa,” *J. Teknovasi*, vol. 02, no. 1, pp. 103–116, 2015.
- [23] D. Svozil, V. Kvasnicka, and J. Pospichal, “Introduction to multi-layer feed-forward neural networks,” *Chemom. Intell. Lab. Syst.*, vol. 39, no. 1, pp. 43–62, 1997.
- [24] R. Eldan and O. Shamir, “The power of depth for feedforward neural networks,” in *Conference on learning theory*, 2016, pp. 907–940.
- [25] M. Cilimkovic, “Neural Networks and Back Propagation Algorithm,” *Fett.Tu-Sofia.Bg*, pp. 3–7, 2010.
- [26] G. Dhaneswara and V. (Jurusan I. K. U. K. P. S. Moertini, “Jaringan Saraf Tiruan Propagasi Balik Untuk Klasifikasi Data,” *Integr. FMIPA Unpar*, 2004.
- [27] H. He and E. A. Garcia, “Learning from imbalanced data,” *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 9, pp. 1263–1284, Sep. 2009, doi: 10.1109/TKDE.2008.239.
- [28] Y. Sun, A. K. C. Wong, and M. S. Kamel, “Classification of imbalanced data: A review,” *Int. J. Pattern Recognit. Artif. Intell.*, vol. 23, no. 4, pp. 687–719, Jun. 2009, doi: 10.1142/S0218001409007326.
- [29] G. Nguyen, A. Bouzerdoum, and S. L. Phung, “Learning pattern classification tasks with imbalanced data sets,” *Fac. Informatics - Pap.*, Jan. 2009.
- [30] S. T. Jishan, R. I. Rashu, N. Haque, and R. M. Rahman, “Improving accuracy of students’ final grade prediction model using optimal equal width binning and synthetic minority over-sampling technique,” *Decis. Anal.*, vol. 2, no. 1, pp. 1–25, Dec. 2015, doi: 10.1186/s40165-014-0010-2.