
DECISION TREE DAN ADABOOST PADA KLASIFIKASI PENERIMA PROGRAM BANTUAN SOSIAL

Oleh

Laila Qadrini¹⁾, Andi Seppewali²⁾, Asra Aina³⁾^{1,2}Program Studi Statistika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Sulawesi Barat, Majene³Program Studi Matematika, Universitas Sulawesi BaratEmail: ¹laila.qadrini@unsulbar.ac.id, ²andi.seppewali@unsulbar.ac.id,
³asra02aina@gmail.com**Abstrak**

Kemiskinan adalah masalah sosial yang masih belum terselesaikan di negara berkembang khususnya di Indonesia. Kemiskinan telah hadir dalam realitas kehidupan manusia dengan bentuk dan kondisi yang sangat memprihatinkan, Karena kemiskinan memang tidak bisa dihilangkan begitu saja. Dengan adanya permasalahan terhadap Negara berkembang terutama kemiskinan. Maka pemerintah membuat kebijakan-kebijakan atau program-program untuk memberantas masalah tersebut. Diantaranya adalah Bantuan langsung tunai atau biasa disebut BLT. Bantuan Langsung Tunai (BLT) dapat dipahami sebagai pemberian sejumlah uang (dana tunai) kepada masyarakat miskin setelah pemerintah memutuskan untuk menaikkan harga BBM dengan jalan mengurangi subsidi namun selisih dari subsidi itu diberikan kepada masyarakat miskin. Melihat dari program pemerintah tersebut, upaya pemberantasan kemiskinan di negara Indonesia ini cukup menarik simpati masyarakat. Hal ini menjadi salah satu objek yang menarik untuk diteliti dan dikaji lebih lanjut. Untuk menentukan klasifikasi tingkat penduduk miskin terdapat banyak metode yang dapat digunakan. Salah satunya yang digunakan pada penelitian ini yaitu *Decision Tree* dan *Adaboost*.

Kata Kunci: Klasifikasi, BLT, *Decision Tree*, *Adaboost***PENDAHULUAN**

Kemiskinan adalah masalah sosial yang masih belum terselesaikan di negara berkembang khususnya di Indonesia. Kemiskinan telah hadir dalam realitas kehidupan manusia dengan bentuk dan kondisi yang sangat memprihatinkan, Karena kemiskinan memang tidak bisa dihilangkan begitu saja. Dengan adanya permasalahan terhadap Negara berkembang terutama kemiskinan. Maka pemerintah membuat kebijakan-kebijakan atau program-program untuk memberantas masalah tersebut. Diantaranya adalah Bantuan langsung tunai atau biasa disebut BLT. Bantuan Langsung Tunai (BLT) dapat dipahami sebagai pemberian sejumlah uang (dana tunai) kepada masyarakat miskin setelah pemerintah memutuskan untuk menaikkan harga BBM

dengan jalan mengurangi subsidi namun selisih dari subsidi itu diberikan kepada masyarakat miskin. BLT merupakan implementasi dari Instruksi Presiden Nomor 3 Tahun 2008 tentang pelaksanaan program bantuan langsung tunai (BLT) untuk rumah tangga sasaran (RTS) dalam rangka kompensasi pengurangan subsidi BBM. Program BLT-RTS ini dalam pelaksanaannya harus langsung menyentuh dan memberikan manfaat langsung kepada masyarakat miskin (yang terkategori sebagai RTS), mendorong tanggung jawab social bersama dan dapat menumbuhkan kepercayaan masyarakat kepada pemerintah yang secara konsisten mesti benar-benar memperhatikan Rumah Tangga Sasaran yang pasti merasakan beban berat sebagai akibat dari kenaikan harga BBM. Program BLT dirancang sebagai pengganti kenaikan biaya hidup ketika terjadi

kenaikan harga BBM oleh karena itu, besaran BLT dihitung sebagai kenaikan biaya hidup penduduk miskin disebabkan kenaikan harga (inflasi) yang diakibatkan langsung maupun tidak langsung oleh kenaikan harga BBM. Melihat dari program pemerintah tersebut, upaya pemberantasan kemiskinan di negara Indonesia ini cukup menarik simpati masyarakat. Hal ini menjadi salah satu objek yang menarik untuk diteliti dan dikaji lebih lanjut. Menteri Desa, Pembangunan Daerah Tertinggal dan Transmigrasi (Mendes PDTT) Abdul Halim Iskandar mengatakan, penyaluran Bantuan Langsung Tunai (BLT) Dana Desa per 21 Juli 2020 telah mencapai Rp 10,83 triliun. Menurut dia, sebanyak 81 persen keluarga penerima manfaat (KPM) BLT Dana Desa merupakan penerima yang baru pertama kali mendapatkan bantuan dari pemerintah. Mereka masuk kategori keluarga miskin, namun tidak terdata untuk mendapatkan jaring pengaman sosial. (Kompas, Diakses 21 Juli 2020) sedangkan realita di lapangan tingkat kemiskinan di Indonesia tahun 2018 masih di angka 9,66%.

Kemiskinan bagi pemerintah Indonesia termasuk masalah yang sulit untuk diselesaikan karena kurangnya pemahaman berbagai pihak tentang penyebab kemiskinan itu sendiri, sehingga program penanggulangan kemiskinan tidak didasarkan pada penyebabnya yang berbeda-beda secara lokal. Upaya yang dilakukan pemerintah dalam mengatasi kemiskinan di Indonesia yaitu dengan program bantuan sosial meliputi BLT (Bantuan Langsung Tunai), PKH (Program Keluarga Harapan), Raskin (Beras Miskin), dan sebagainya. Berdasarkan Data dari Bappenas 2014, masalah kemiskinan saat ini disebabkan oleh beberapa faktor, antara lain: ketidaktepatan sasaran dalam penentuan penerimaan program, mekanisme pendampingan program belum optimal, koordinasi dan pelaksanaan program belum terintegrasi dan prioritas pendanaan untuk program perlindungan sosial yang masih terbatas. (Elly et al, 2020). Untuk menentukan

klasifikasi tingkat penduduk miskin terdapat banyak metode yang dapat digunakan. Salah satunya yaitu K-NN dan Gradient Boosted Trees. Berdasarkan penelitian sebelumnya yang dilakukan oleh Yunus, dkk (2019) tentang Data Mining untuk Memprediksi Hasil Produksi Buah Sawit pada PT Bumi Sawit Sukses menggunakan Metode K-Nearest Neighbor (K-NN) menghasilkan output dari Rapidminer dengan akurasi 85,15% (Yunus, Akbar, & Andri, 2019). Menurut penelitian sebelumnya yang dilakukan oleh Saikin dan Kusri mengenai Karakteristik Data Traveller didapatkan hasil kualifikasi metode K-NN dengan pengujian confusion matrix didapatkan nilai akurasi sebesar 84% (Saikin & Kusri, 2019). Berdasarkan penelitian terdahulu dari masing-masing metode yang digunakan menghasilkan akurasi yang rendah. Kombinasi antara Decision Tree dan Adaboost diperlukan untuk memperoleh nilai akurasi yang lebih tinggi.

LANDASAN TEORI

2.1 Decision Tree

Decision Tree adalah sebuah diagram alir yang berbentuk seperti struktur pohon yang mana setiap *internal node* menyatakan pengujian terhadap suatu atribut, setiap cabang menyatakan *output* dari pengujian tersebut dan *leaf node* menyatakan kelas-kelas atau distribusi kelas. *Node* yang paling atas disebut sebagai *root node* atau node akar. Sebuah *root node* akan memiliki beberapa *edge* keluar tetapi tidak memiliki *edge* masuk, internal node akan memiliki satu *edge* masuk dan beberapa *edge* keluar, sedangkan leaf node hanya akan memiliki satu *edge* masuk tanpa memiliki *edge* keluar. *Decision Tree* digunakan untuk mengklasifikasikan suatu sampel data yang belum diketahui kelasnya ke dalam kelas-kelas yang sudah ada. Algoritma - algoritma dalam *Decision Tree*. Ada banyak algoritma pada klasifikasi *Decision Tree* ini. Suatu algoritma biasanya dikembangkan untuk meningkatkan kinerja algoritma yang sudah ada. Penentuan algoritma yang terbaik dalam *Decision Tree*

tentunya tidak bisa ditentukan secara mutlak tetapi sangat tergantung dengan karakteristik training set-nya. Beberapa algoritma *Decision Tree* yang cukup populer antara lain : ID3, C4.5, dan CART.

2.2 Algoritma C4.5

Algoritma ini dikembangkan untuk memperbaiki algoritma ID3. Algoritma ini berbasiskan keputusan biner seperti yang terlihat dalam CLS. Jadi selain memiliki karakteristik seperti ID3, C4.5 juga memiliki beberapa karakteristik yang berbeda yang merupakan perbaikan dari karakteristik ID3. Berikut ini beberapa karakteristik C4.5 yang juga merupakan perbaikan terhadap ID3 :

- Dapat menangani atribut numerik
- Dapat menangani missing value
- Melakukan pruning untuk memperoleh model yang paling efisien
- Menggunakan kriteria gain ratio untuk menentukan jenis split yang terbaik.

2.3 Karakteristik *Decision Tree*

Berikut ini adalah beberapa karakteristik *Decision Tree* secara umum :

- *Decision Tree* merupakan suatu pendekatan nonparametrik untuk membangun model klasifikasi
- Teknik yang dikembangkan dalam membangun *Decision Tree* memungkinkan untuk membangun model secara cepat dari *training set* yang berukuran besar.
- *Decision Tree* dengan ukuran tree yang kecil relatif mudah untuk menginterpretasinya.
- *Decision Tree* memberikan gambaran yang ekspresif dalam pembelajaran fungsi nilai diskret.
- Algoritma *Decision Tree* cukup robbust terhadap munculnya *noise* terutama untuk metode yang dapat menangani masalah *overfitting*.
- Adanya atribut yang berlebihan tidak terlalu mengurangi akurasi *Decision Tree*.
- Karena sebagian algoritma *Decision Tree* menggunakan pendekatan top down, yaitu partisi dilakukan secara rekursif maka jumlah *record* menjadi lebih kecil. Pada *leaf node*, jumlah *record* mungkin akan terlalu kecil untuk dapat membuat keputusan secara

statistik tentang representasi kelas dari suatu node.

- Sebuah *subtree* dapat direplikasi beberapa kali dalam *Decision Tree* tetapi ini akan menyebabkan *Decision Tree* menjadi lebih kompleks dan lebih sulit untuk diinterpretasi. (Sibaroni, 2008:8).

Dalam proses pengambilan keputusan, *Decision Tree* digunakan sebagai alat visual dan analitis, untuk memprediksi nilai target. (Witten, 2011). Algoritma C4.5 adalah algoritma populer dalam klasifikasi data mining. Algoritma C4.5 menggunakan kriteria information gain untuk memilih atribut yang akan digunakan dalam pemisahan objek. Atribut yang mempunyai *information gain* paling tinggi dibandingkan dengan atribut pada data lain, maka atribut tersebutlah yang dipilih sebagai pemecahan. *Decision Tree* (Pohon Keputusan) sesuai digunakan untuk kasus-kasus dimana *outputnya* bernilai diskrit (Agustinus, 2012).

2.4 Adaptive Boosting (*Adaboost*)

Adaptive Boosting (Adaboost)

merupakan salah satu dari beberapa varian pada algoritma boosting (Chezian & Kumar, 2014). *Adaboost* merupakan *ensemble learning* yang sering digunakan pada algoritma boosting. *Boosting* bisa dikombinasikan dengan *classifier* algoritma yang lain untuk meningkatkan performa klasifikasi. Tentunya secara intuitif, penggabungan beberapa model akan membantu jika model tersebut berbeda satu sama lain. *Adaboost* dan variannya telah sukses diterapkan pada beberapa bidang (domain) karena dasar teorinya yang kuat, prediksi yang akurat, dan kesederhanaan yang besar. Langkah-langkah pada algoritma *Adaboost* adalah sebagai berikut.

- Input: Suatu kumpulan sample penelitian dengan label $\{(x_i, y_i), \dots, (x_N, y_N)\}$, suatu *component learn* algoritma, jumlah perputaran T.
- Initialize*: Bobot suatu sampel pelatihan $w_i^1 = 1/N$, untuk semua $i = 1, \dots, N$
- Do for* $t = 1, \dots, T$
- Gunakan *component learn* algoritma untuk melatih suatu komponen klasifikasi, pada

sample bobot pelatihan.

e. Hitung kesalahan pelatihnnya pada h_t : $\varepsilon_t = \sum_i^N w_i^t, y_i \neq h_t(x_i)$

f. Tetapkan bobot untuk component classifier

$$h_t = \alpha_t = \frac{1}{2} \ln \left(\frac{1 - \varepsilon_t}{\varepsilon_t} \right)$$

g. Update bobot sample pelatihan $w_i^{t+1} = \frac{w_i^t \exp\{-\alpha_t y_i h_t(x_i)\}}{C_t}$, $i = 1, \dots, N$ C_t adalah suatu konstanta normalisasi.

h. Output: $f(x) = \text{sign}(\sum_{t=1}^T \alpha_t h_t(x))$

2.5 Akurasi, Presisi, dan Recall

1) Akurasi, Presisi, dan Recall

Akurasi dapat didefinisikan sebagai tingkat kedekatan antara nilai prediksi dengan nilai aktual. Presisi menunjukkan tingkat ketepatan atau ketelitian dalam pengklasifikasian. Sedangkan recall berfungsi untuk mengukur proporsi positif aktual yang benar diidentifikasi. Untuk mengukur akurasi, presisi, dan recall biasanya digunakan confusion matrix. Confusion matrix adalah alat ukur berbentuk matrix yang digunakan untuk mendapatkan jumlah ketepatan klasifikasi terhadap kelas dengan algoritme yang dipakai. Berikut akan disajikan bentuk confusion matrix pada Tabel 1.

Tabel 1. Bentuk Confusion Matrix Dari Dua Kelas

Confusion Matrix		Nilai Sebenarnya	
		True	False
Nilai Prediksi	True	TP (True Positive) Correct result	FP (False Positive) Unexpected result
	False	FN (False Negative) Missing result	TN (True Negative) Correct absence of result

Pada Tabel 1 nilai TP (true positive) dan TN (true negative) menunjukkan tingkat ketepatan klasifikasi. Umumnya semakin tinggi nilai TP dan TN semakin baik pula tingkat klasifikasi

dari akurasi, presisi, dan recall. Jika label prediksi keluaran bernilai benar (true) dan nilai sebenarnya bernilai salah (false) disebut sebagai false positive (FP). Sedangkan jika prediksi label keluaran bernilai salah (false) dan nilai sebenarnya bernilai benar (true) maka hal ini disebut sebagai false negative (FN) (Han & Kember, 2013). Berikut formulasi untuk menghitung akurasi, presisi, dan recall pada pembentukan model klasifikasi ditunjukkan pada Persamaan (1), Persamaan (2), dan Persamaan (3).

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \tag{1}$$

$$Presisi = \frac{TP}{TP+FP} \times 100\% \tag{2}$$

$$Recall = \frac{TP}{TP+FN} \times 100\% \tag{3}$$

A. Kurva ROC

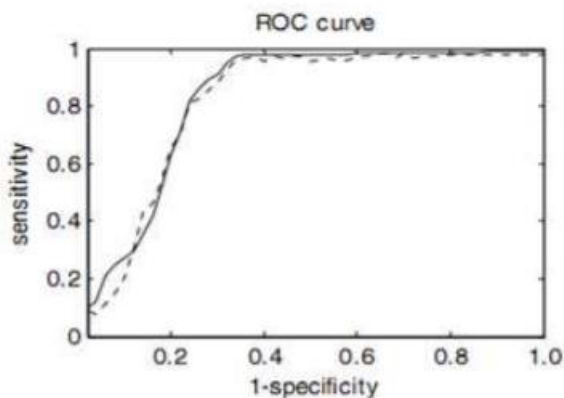
Kurva ROC (receiver operating characteristic) adalah salah satu alat ukur untuk menilai kemampuan sistem klasifikasi. Kurva ROC sering digunakan untuk mengevaluasi pengklasifikasian karena memiliki kemampuan evaluasi algoritma dengan cukup baik. Kurva ROC merupakan grafik perbandingan antara sensitivity (true positive rate (TPR)) yang diterjemahkan kedalam sumbu vertikal atau sumbu koordinat y dengan specificity (false positive rate (FPR)) yang diterjemahkan dalam bentuk kurva (Witten et al, 2011). Berikut formulasi dari sensitivity dan specificity dipaparkan pada Persamaan (4), dan Persamaan (5).

$$Sensitivity = \frac{TP}{TP+FN} \times 100\% \tag{4}$$

$$Specificity = \frac{TN}{TN+FP} \times 100\% \tag{5}$$

Kurva ROC dapat digunakan sebagai komparasi beberapa metode (classifier)

ataupun model classifier yang memiliki perbedaan parameter guna mendapatkan model yang paling baik. Berikut adalah contoh penerapan komparasi performansi dari dua classifier yang berbeda pada Gambar 1.



Gambar 1. Komparasi Classifier dengan Kurva ROC

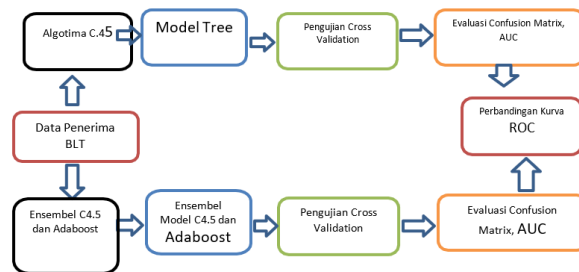
Pada Gambar 1 dapat dilihat bahwa terdapat dua buah classifier yang disimbolkan dengan garis putus-putus dan garis utuh. Jika pada Gambar 1 menunjukkan letak koordinat (0,1) hal tersebut mewakili *sensitivity* dan *specificity* sebesar 100%. Untuk menghitung dan memastikan *classifier* mana yang lebih unggul maka digunakan penghitungan AUC (*area under curve*). AUC (*area under curve*) adalah luas area dibawah kurva. Luas dari AUC selalu berada diantara nilai 0 hingga 1. AUC dihitung berdasarkan gabungan luas trapesium titik-titik (*sensitivity* dan *specificity*). Pada Gambar 1 memperlihatkan bahwa garis yang utuh memiliki area dibawah kurva yang lebih besar dibandingkan garis yang putus-putus, hal ini berarti bahwa tingkat performansi klasifikasi dari *classifier* yang dilambangkan dengan garis utuh lebih baik dibandingkan tingkat performansi klasifikasi dari *classifier* yang dilambangkan dengan garis putus-putus. Berikut adalah standar Tabel kategori pengklasifikasian berdasarkan nilai AUC pada Tabel 2

Tabel 2. Kategori Pengklasifikasian Berdasarkan Nilai AUC

Nilai AUC	Kategori Pengklasifikasian
.90 - 1.00	Excellent
.80 - 0.90	Good
.70 - 0.80	Fair
.60 - 0.70	Poor
.50 - 0.60	Fail

METODE PENELITIAN

Penelitian ini dilakukan untuk pengembangan evaluasi dan pemecahan masalah yaitu meningkatkan akurasi pada pelaksanaan program bantuan sosial agar tepat sasaran. Berikut ini adalah tahapan penelitian yang dilakukan:



Gambar 1. Metode Penelitian

HASIL DAN PEMBAHASAN

Sumber Data

Penelitian ini menggunakan data sekunder penduduk Desa yang memperoleh Bantuan Langsung Tunai di Kelurahan Banggae, Data yang digunakan dalam penelitian ini adalah data Tahun 2020. Variabel yang digunakan adalah numerik dan kategorik.

Variabel Penelitian

Pada penelitian ini menggunakan data penduduk desa yang diperoleh dari kantor kelurahan, sebagai berikut:

- Pendidikan: Tidak SD, SD, SLTP, SLTA, S1
- Luas Lantai: Kurang dari 13 m², Sama 13 m², Lebih dari 13 m²
- Jenis Lantai: Keramik, Semen, Tanah
- Jenis Dinding: Bambu, Tembok
- Jenis Penerangan: Non PLN, PLN
- Bahan Bakar Masak: Kayu, Gas

- g. Sumber Air Minum : Sumur, Ledeng
 h. Jenis Jamban: Umum, Bersama, Sendiri
 i. Pendapatan: 0 sampai 1,5 juta, 1,5 sampai 3 juta, lebih dari 3 juta
 j. Pekerjaan: Tidak Bekerja/Buruh, Petani, Pedagang, Wiraswasta, Pegawai

Pra-pemrosesan Data

Prapemrosesan data adalah mengeksplor dan memahami data yang akan diolah hingga data tersebut layak untuk melaju ketahap pemodelan, Hal ini merupakan langkah awal pada proses klasifikasi data. Penelitian ini menggunakan penggabungan dua metode yaitu metode *Decision Tree* dan *Adaboost*. Pada Seleksi Fitur digunakan untuk menyeleksi data yang rusak/tidak lengkap menggunakan fitur "Input Missing Value" dan "Rename Unused Value" dengan menggunakan metode *Decision Tree* sehingga didapatkan data set murni. Mentranformasikan data dari numerik ke nominal dan lakukan normalisasi menentukan bentuk data yang paling tepat. Selanjutnya membagi data *training* dan data *testing* dengan perbandingan 80%:20%. Pemrosesan data dilakukan dengan menggabungkan metode *Adaboost* dalam proses validasi sehingga diperoleh akurasi yang lebih tinggi dari pengolahan masing-masing metode. Hasil yang diperoleh dari masing-masing perhitungan masing-masing metode adalah sebagai berikut:

Tabel 2. Perhitungan Metode *Decision Tree*

<i>Accuracy:</i>	94.17%		
	True Tidak Miskin	True Miskin	<i>Class Precision</i>
Pred Tidak Miskin	98	7	93%
Pred Miskin	0	15	100%
<i>Class Recall</i>	100%	68%	

Tabel 3. Perhitungan Metode *Adaboost*

<i>Accuracy:</i>	95.00%		
	True Miskin	True Tidak Miskin	<i>Class Precision</i>
Pred Tidak Miskin	94	6	94%
Pred Miskin	0	20	100%
<i>Class Recall</i>	100%	77%	

Tabel 4. Perbandingan Akurasi

No	Metode Klasifikasi	Akurasi
1	<i>Decision Tree</i>	94.17%
2	<i>Adaboost</i>	95%

KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan dapat disimpulkan bahwa:

1. Metode klasifikasi *Decision tree* dan *Adaboost* sama sama memiliki hasil akurasi yang baik yaitu sebesar 94% dan 95%
2. Metode klasifikasi *Decision tree* dan *Adaboost* tepat digunakan untuk menentukan penerima bantuan sosial secara tepat

DAFTAR PUSTAKA

- [1] Chezian, D.R.M., & Kumar, K.S. 2014. Support Vector Machine and K-Nearest Neighbour Based Analysis for the Prediction of Hypothyroid. *International Journal of Pharma and Bio Sciences*. 5(4)(B) 447-453.
- [2] Elly Firasari, Umi Khultsum, Monikka Nur Winnarto Risnandar, Kombinasi K-NN dan *Gradient Boosted Trees* untuk Klasifikasi Penerima Program Bantuan Sosial Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK). Vol. 7, No. 6, Desember 2020, hlm. 1231-1236.
- [3] I.H. Witten, E. Frank, and M.A. Hall, "Data Mining Practical Machine Learning Tools and Techniques", Third Edition, Elsevier Publisher, USA, 2011.

-
- [4] J. Agustinus, “Sistem Deteksi Intrusi Jaringan dengan Metode Support Vector Machine”, M. Eng, Thesis, Jurusan Ilmu Komputer. FMIPA UGM, Yogyakarta, 2012.
- [5] J. Han, and M. Kamber, “Data mining: Concepts and Techniques”, Third Edition, Morgan Kaufmann Publishers, San Fransisco, 2013.
- [6] Kurniawan, Dios, Pengenalan Machine Learning dengan Python, PT. Elex Media Komputindo, Jakarta, 2020.
- [7] Saikin, & Kusri. 2019. Model Data Mining Untuk Karakteristik Data Traveller Pada Perusahaan Tour And Travel (Studi Kasus : Lombok Ceria Holiday). *Jurnal Manajemen Informatika & Sistem Informasi*, 2(2), 61–68.
- [8] Yunus, A., Akbar, M., & Andri. 2019. Data Mining Untuk Memeprediksi Hasil Produksi Nuah Sawit Pada Pt Bumi Sawit Saukses (Bss) Menggunakan Metode K-Nearest Neighbor. *Bina Darma Conference on Computer Science*, 198–207.

HALAMAN INI SENGAJA DIKOSONGKAN