

Implementasi Algoritma Genetika untuk Optimalisasi Random Forest Dalam Proses Klasifikasi Penerimaan Tenaga Kerja Baru: Studi Kasus PT.XYZ

Laras Binarwati, Imam Mukhlash, dan Soetrisno

Departemen Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Institut Teknologi

Sepuluh Nopember (ITS)

e-mail: imammukhlash@matematika.its.ac.id

Abstrak—Kualitas sumber daya manusia sangat penting bagi suatu perusahaan untuk mempertahankan keunggulan kompetitifnya agar mampu bersaing dengan perusahaan lainnya maupun untuk meningkatkan kualitas dari perusahaan itu sendiri. Oleh karena itu, menggali pola penerimaan tenaga kerja baru sangat diperlukan. Pada penelitian ini, metode random forest digunakan untuk menggali pola penerimaan tenaga kerja baru. Adapun Algoritma Genetika (GA) digunakan untuk mengoptimalkan akurasi berdasarkan pola yang didapat. Hasil pengujian program ini menunjukkan keakuratan pola yang dihasilkan oleh random forest yang dioptimalkan dengan algoritma genetika lebih tinggi dengan hasil keakuratan berkisar antara 91%-95% dibanding dengan hanya menggunakan random forest saja yang hanya berkisar 40%-95%.

Kata Kunci—Penerimaan Tenaga Kerja, Klasifikasi, *Random Forest*, Algoritma Genetika.

I. PENDAHULUAN

PENELITIAN mengenai perekrutan karyawan telah meningkat secara dramatis. Kualitas sumber daya manusia sangat penting bagi suatu perusahaan untuk mempertahankan keunggulan kompetitifnya agar mampu bersaing dengan perusahaan lainnya maupun untuk meningkatkan kualitas dari perusahaan itu sendiri. Banyak penelitian awal difokuskan pada penggunaan ulasan pekerjaan yang realistis (yaitu, memberikan para pelamar kerja dengan informasi yang akurat tentang posisi dalam organisasi), metode perekrutan tradisional (misalnya, iklan pada koran), dan karakteristik para pelamar (misalnya, perilaku mereka), maka dari itu rekrutmen personal dan seleksi secara langsung merupakan cara yang dianggap sangat mempengaruhi dari segi kualitas karyawan tersebut. Pemilihan calon karyawan yang tepat adalah salah satu faktor yang paling penting untuk keberhasilan keputusan strategis jangka panjang yang diambil di dalam suatu perusahaan.

Namun, karena sifat perubahan pengetahuan pekerja di suatu perusahaan, pekerjaan tidak dapat dengan mudah digambarkan terutama untuk pekerjaan di tingkat manajemen. Persyaratan kualitas personal di suatu perusahaan semakin ketat, sedangkan proses kerja di perusahaan itu menjadi beragam dan rumit. Dengan demikian, pendekatan konvensional dengan seleksi personal yang dikembangkan atas dasar

karakteristik pekerjaan statis tidak akan lagi cukup [1]. Dalam rangka untuk mencari orang yang tepat untuk melakukan hal yang benar untuk pekerjaan yang tepat, mengembangkan pendekatan pilihan yang efektif sangatlah penting. Salah satu hal penting dalam menentukan tindakan preventif ini dengan mengetahui pola yang telah terjadi sebelumnya. Salah satu cara yang dapat dilakukan untuk membuat pola tersebut dapat dilakukan dengan data mining.

C.-F. Chien dan L.-F. Chen telah melakukan penelitian mengenai penjelasan dan prediksi pola penerimaan tenaga kerja pada tahun 2008 [2]. Penelitian ini bertujuan untuk memprediksi dan menentukan dengan mengembangkan kerangka data mining berdasarkan pohon keputusan dan asosiasi rule untuk menghasilkan keputusan penerimaan tenaga kerja baru. Studi kasus yang digunakan dalam penelitian ini adalah sebuah perusahaan dibidang teknologi industri tingkat tinggi. Variabel yang dipakai sebanyak 4 macam yang berasal dari faktor internal karyawan dan sistem perusahaan itu sendiri. Hasil dari penelitian ini menunjukkan bahwa metode decision tree dengan classification Algorithm dapat mempermudah dan dapat membantu mengambil suatu keputusan untuk penerimaan tenaga kerja baru.

Pada penelitian lainnya membahas tentang pengaplikasian algoritma genetik (GA) untuk mengoptimalkan random forests pada data kelas rekayasa [3]. Hasil membuktikan keunggulan metode yang diusulkan dalam menaikkan akurasi. Kelengkapan training data set (contoh data pelatihan) memegang peran penting dalam menentukan kualitas pengklasifikasi pola yang dihasilkan. Penelitian ini menyarankan tiga komponen sistem untuk meningkatkan akurasi klasifikasi dalam random forests. Komponen pertama adalah class decomposition, kedua adalah dengan menggunakan dua parameter, yaitu jumlah tree dalam ensemble, dan jumlah sampel acak yang menonjol pada setiap node split di setiap tree. Ketiga adalah menggunakan algoritma genetika. Penelitian ini menggunakan 22 data set, dan hasilnya membuktikan effectiveness teknik pembelajaran mesin hybrid yang diusulkan dalam meningkatkan akurasi prediksi.

Data set untuk penelitian ini diambil dari perusahaan yang bergerak dibidang pelayaran shipping company, yang salah satu fungsinya sebagai Transportir pemuat Cargo bijih besi maupun batubara. Kualitas sumber daya manusia sangat penting bagi perusahaan untuk mempertahankan keunggulan

kompetitif dalam era sekarang ini. Namun, perusahaan sering merasa sulit untuk merekrut bakat yang tepat. Pada tugas akhir ini akan dibangun suatu perangkat lunak yang mampu menggali pola penerimaan tenaga kerja baru menggunakan random forests yang dioptimalkan dengan menggunakan algoritma genetika.

II. KAJIAN TEORI

A. Klasifikasi

Classification adalah proses penemuan model (fungsi) yang menggambarkan dan membedakan kelas data atau konsep yang bertujuan agar bisa digunakan untuk memprediksi kelas dari objek yang label kelasnya tidak diketahui [4].

Classification data terdiri dari 2 langkah proses. Pertama adalah learning (fase training), dimana algoritma classification dibuat untuk menganalisa data training lalu direpresentasikan dalam bentuk rule classification. Proses kedua adalah classification, dimana data tes digunakan untuk memperkirakan akurasi dari rule classification [2].

Proses classification didasarkan pada empat komponen [5]:

- a. Kelas. Variabel dependen yang berupa kategorikal yang merepresentasikan "label" yang terdapat pada objek. Contohnya: resiko penyakit jantung, resiko kredit, customer loyalty, jenis gempa.
- b. Predictor. Variabel independen yang direpresentasikan oleh karakteristik (atribut) data. Contohnya: merokok, minum alkohol, tekanan darah, tabungan, aset, gaji.
- c. Training dataset. Satu set data yang berisi nilai dari kedua komponen di atas yang digunakan untuk menentukan kelas yang cocok berdasarkan predictor.
- d. Testing dataset. Berisi data baru yang akan diklasifikasikan oleh model yang telah dibuat dan akurasi klasifikasi dievaluasi.

Berikut ini adalah algoritma classification dalam data mining yang paling populer yaitu adalah [5]:

- a. Decision/classification trees
- b. Bayesian classifiers/Naive Bayes classifiers
- c. Neural networks
- d. Statistical analysis
- e. Rough sets
- f. k-nearest neighbor classifier
- g. Rule-based methods
- h. Memory based reasoning

B. Random Forest

Metode random forest adalah pengembangan dari metode CART, yaitu dengan menerapkan metode bootstrap aggregating (bagging) dan random feature selection [6]. Dalam random forest, banyak pohon ditumbuhkan sehingga terbentuk hutan (forest), kemudian analisis dilakukan pada kumpulan pohon tersebut. Pada gugus data yang terdiri atas n amatan dan p peubah penjelas, random forest dilakukan dengan cara [6]:

1. lakukan penarikan contoh acak berukuran n dengan pemulihan pada gugus data. Tahapan ini merupakan tahapan bootstrap

2. dengan menggunakan contoh bootstrap, pohon dibangun sampai mencapai ukuran maksimum (tanpa pemangkasan). Pada setiap simpul, pemilihan pemilah dilakukan dengan memilih m peubah penjelas secara acak, dimana $m \ll p$. Pemilah terbaik dipilih dari m peubah penjelas tersebut. Tahapan ini adalah tahapan random feature selection

3. Ulangi langkah 1 dan 2 sebanyak k kali, sehingga terbentuk sebuah hutan yang terdiri atas k pohon

Respons suatu amatan diprediksi dengan menggabungkan (aggregating) hasil prediksi k pohon. Pada masalah klasifikasi dilakukan berdasarkan majority vote (suara terbanyak).

Error klasifikasi random forest diduga melalui error OOB yang diperoleh dengan cara [6]:

1. Lakukan prediksi terhadap setiap data OOB pada pohon yang bersesuaian. Data OOB (out of bag) adalah data yang tidak termuat dalam contoh bootstrap.
2. Rata-rata, setiap amatan gugus data asli akan menjadi data OOB sebanyak sekitar 36% dari banyak pohon. Oleh karena itu, pada langkah 1, masing-masing amatan gugus data asli mengalami prediksi sebanyak sekitar sepertiga kali dari banyaknya pohon. Jika a adalah sebuah amatan dari gugus data asli, maka hasil prediksi random forest terhadap a adalah gabungan dari hasil prediksi setiap kali a menjadi data OOB.
3. Error OOB dihitung dari proporsi misklasifikasi hasil prediksi random forest dari seluruh amatan gugus data asli.

C. Algoritma Genetika

Algoritma Genetika atau Genetic Algorithm (GA) dikenalkan oleh John Holland dalam menyelesaikan masalah optimasi. Algoritma Genetika mensimulasikan proses yang terjadi pada populasi alamiah yang merupakan hal yang penting dalam proses evolusi. Algoritma Genetika adalah metode pencarian yang meniru perumpamaan evolusi biologis alami untuk menentukan kromosom atau individu berkualitas tinggi dalam suatu kawasan berhingga potensial yang disebut populasi. Proses pemilihan individu dari suatu populasi dievaluasi berdasarkan fungsi fitness. Kromosom berwujud string tersebut merupakan calon pada setiap siklus operasi yang disebut generasi. Struktur umum pada Algoritma Genetika yaitu:

1. Representasi kromosom.
2. Evaluasi dengan menghitung fitness.
3. Proses crossover untuk mendapatkan individu baru.
4. Proses mutasi yang untuk meningkatkan variasi dalam populasi.
5. Proses seleksi untuk membentuk populasi baru.

Beberapa hal yang termasuk kelebihan dari Algoritma Genetika adalah sebagai berikut [7]:

1. Mengoptimalkan dengan variabel kontinu atau diskrit,
2. Tidak memerlukan informasi derivatif,
3. Bersamaan pencarian dari sebuah sampling yang luas pada permukaan biaya,
4. Berkaitan dengan sejumlah besar variabel,
5. Baik untuk komputer paralel,

6. Mengoptimalkan permukaan variabel dengan biaya yang sangat kompleks (GA bisa melompat dari minimum lokal),
7. Memberikan daftar variabel yang optimal, bukan hanya solusi tunggal,
8. Dapat menyandikan variabel sehingga optimasi dilakukan dengan mengkodekan variabel, dan
9. Bekerja dengan data numerik yang dihasilkan, data eksperimen, atau analitis fungsi.

Dalam tugas akhir ini, algoritma genetika dipergunakan untuk mengoptimalkan dan membandingkannya dengan hasil dari random forest.

III. HASIL DAN PENGUJIAN SISTEM

A. Pendefinisian Individu

Individu menyatakan satu solusi yang mungkin. Individu sendiri dapat dikatakan sebagai kromosom, yang merupakan kumpulan gen. Gen ini dapat berisi biner, integer, float, double, karakter, atau bahkan kombinatorial. Dalam penelitian kali ini, gen akan berisi nilai biner, yaitu nilai biner dari nilai pada setiap fitur. Fitur yang digunakan berjumlah 6 yaitu, posisi, pendidikan, usia numerik, tanggungan, status, dan keterangan. Satu individu terdiri dari 6 kromosom dan individu tersebut merupakan solusi dari permasalahan yang ada dalam penelitian kali ini

B. Inisialisasi Kromosom

Inisialisasi adalah tahap dimana Populasi Awal akan ditentukan. Karena kromosom bisa direpresentasikan dengan binary string dan kita memiliki 64 fitur (Ada 6 fitur dalam data sehingga banyak kemungkinan kromosom yang terjadi adalah $2^6 = 64$) maka untuk membuat inisialisasi awal kromosom kita dapat merandom angka dari 1-63 karena 0 akan menghilangkan semua fitur.

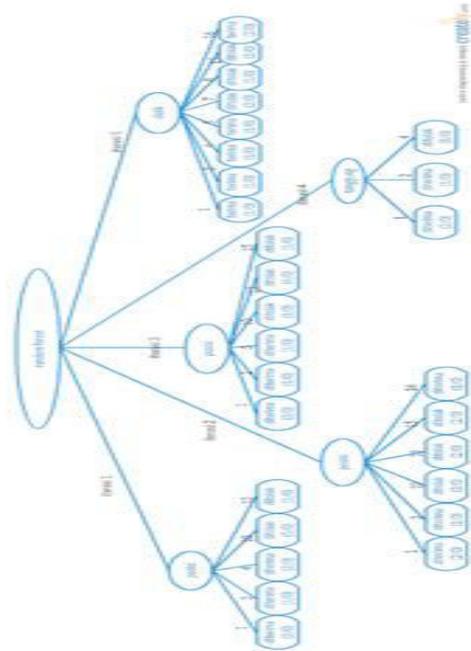
Pada pengujian kali ini, akan di inputkan jumlah insialisasi awal kromosom sebanyak 20. Hasil Input merupakan nilai random integer dari 1-63. Apabila hasil random menunjukkan angka 3 maka representasi kromosomnya: 0,0,0,0,1,1 sehingga fitur yang digunakan nomer 5 dan 6 (karena bernilai 1 atau true).

C. Akurasi Random Forest

Pada weka random forest memiliki parameter yaitu -U untuk unpruned tree. Inisialisasi RandomForest dengan new RandomForest. Kemudian build classifier (process learning) menggunakan data training. Untuk uji coba memakai Evaluation dengan input tree hasil training random forest serta data set training, kemudian menghitung nilai benar /jumlah data test untuk mendapatkan nilai akurasi.

Dalam pembentukan tree, random forest akan melakukan training pada sampel data. Pengambilan sample dengan cara sampling with replacement. Sebanyak sepertiga dari sample akan digunakan untuk menentukan data out of bag (OOB). Penentuan data out of bag (OOB) dilakukan untuk mengestimasi eror dan menentukan variable importance. Setelah seluruh tree terbentuk, makan proses klasifikasi akan berjalan. Setelah klasifikasi berjalan, maka akan dilanjutkan dengan inisialisasi. Inisialisasi dilakukan sebanyak jumlah

yang diinginkan oleh user dan akan di urutkan berdasarkan nilai akurasi tertinggi ke rendah.



Gambar 1. Struktur Random Forest.

Struktur random forest yang diperlihatkan pada gambar 3.1 adalah contoh 11 data acak dari keseluruhan total data yang berjumlah 126 dengan bagging sebanyak 5 iterasi dan persentase split sebanyak 70%.

Berikut ini adalah contoh data awal (Tabel 3.1) dan data hasil resample (Tabel 3.2) sebelum diolah dengan teknik random forest:

Tabel 1.
Data awal

Data ke	X1...	X6	Y
1	2	1	1
2	3	2	1
3	1	3	0
4	4	21	0
5	16	45	0
.			
.			
N	10	47	1

Tabel 2.
Data resample

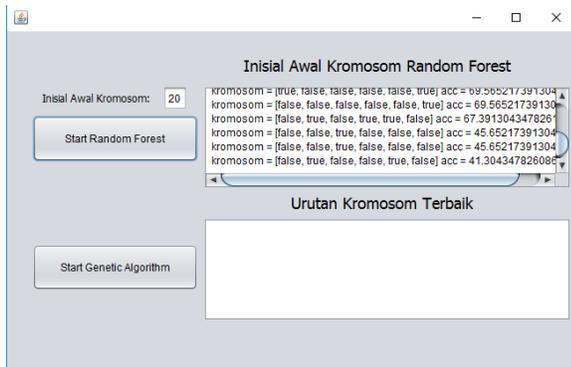
Data ke	X1	X3	X4	Y
1	3	34	4	0
2	27	27	2	0
3	1	40	4	0
4	1	27	2	1
5	1	38	1	1
6	15	41	2	0
7	11	42	3	1

Inti dari algoritma random forest adalah bagaimana mengambil data resample dan variable faktor secara acak. Pada contoh ini digunakan variabel faktor sebanyak 3 seperti yang terlihat pada Tabel 3.2 yaitu X1, X3, DAN X4.

Tabel 3.
Hasil Prediksi

Dat a ke	X1..	X	Y	Prediks i 1	Prediks i 2	Prediks i 3	Prediks i 4	Prediks i akhir
1	1	47	0	1	1	-	0	1
2	15	48	0	0	-	-	-	0
3	2	47	0	-	1	0	0	0

Tabel 3.3 diambil dengan aturan mayoritas semua pohon keputusan. Nilai akurasi didapatkan dari jumlah total prediksi benar dibagi dengan total keseluruhan data yaitu 2/3 dan menghasilkan akurasi sebesar 66,7%.



Gambar 2. Hasil Akurasi *Random Forest*.

Saat user memasukkan/menginput jumlah inisialiasasi kromosom awal sesuai gambar 3.2 maka akan ditampilkan jumlah kromosom sesuai input beserta nilai akurasinya. Berdasarkan running program tersebut diperoleh hasil sebagai berikut:

1. kromosom = [true, false, false, true, false, true] acc = 95.65217391304348
2. kromosom = [false, true, true, true, false, true] acc = 93.47826086956522
3. kromosom = [true, true, false, true, false, true] acc = 93.47826086956522
4. kromosom = [true, true, true, false, false, true] acc = 91.30434782608695
5. kromosom = [true, true, false, false, false, true] acc = 91.30434782608695
6. kromosom = [false, true, true, false, true, true] acc = 84.78260869565217
7. kromosom = [false, false, false, true, true, false] acc = 84.78260869565217
8. kromosom = [false, false, true, false, true, true] acc = 82.6086956521739
9. kromosom = [true, true, true, true, false, false] acc = 78.26086956521739
10. kromosom = [false, false, true, true, true, false] acc = 78.26086956521739
11. kromosom = [false, true, true, true, true, false] acc = 73.91304347826086
12. kromosom = [false, false, false, false, true, true] acc = 69.56521739130434
13. kromosom = [true, true, true, false, false, false] acc = 69.56521739130434
14. kromosom = [false, true, false, false, false, true] acc = 69.56521739130434

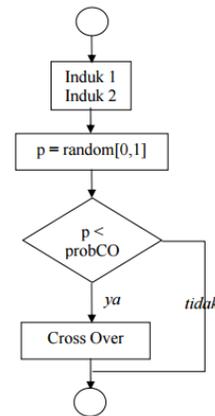
15. kromosom = [true, false, false, false, false, true] acc = 69.56521739130434
16. kromosom = [false, false, false, false, false, true] acc = 69.56521739130434
17. kromosom = [false, true, false, true, true, false] acc = 67.3913043478261
18. kromosom = [false, false, true, false, false, false] acc = 45.65217391304348
19. kromosom = [false, false, true, false, false, false] acc = 45.65217391304348
20. kromosom = [false, true, false, false, true, false] acc = 41.30434782608695

D. Akurasi Algoritma Genetika

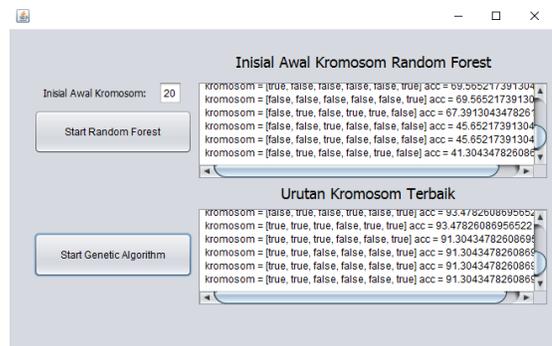
Pada algoritma genetika, kromosom hasil inialisasi akan diproses sesuai tahapan algoritma genetika yaitu crossover dan mutasi. Proses algoritma genetika menggunakan perulangan sebanyak 30 kali. Setiap perulangan akan dipilih 2 kromosom secara random dengan kriteria akurasi rank 10 besar.

Setelah terpilih kromosom 1 dan 2, maka akan di crossover dengan tipe one point crossover. Pada fungsi ini apabila iterasi kurang dari point maka nilai kromosom sama dengan nilai kromosom awal tetapi jika melebihi nilai point maka nilainya akan diganti dengan kromosom lainnya.

Setelah selesai crossover maka setiap kromosom akan dilakukan mutasi dengan random rate mutasi. Pada tahap ini setiap bit dalam kromosom akan dicek satu persatu dengan suatu nilai random 0 dan 1. apabila nilai random bernilai 0 maka nilai kromosom index tsb di negasikan (0→1,1→0) apabila 1 maka tetap.



Gambar 3. Alur Kerja *Crossover*.



Gambar 4. Akurasi Algoritma Genetika.

Berdasarkan running program pada gambar 3.3, diperoleh hasil sebagai berikut:

1. kromosom = [true, false, false, true, false, true] acc = 95.65217391304348
2. kromosom = [true, false, false, true, false, true] acc = 95.65217391304348
3. kromosom = [true, false, true, true, true, true] acc = 95.65217391304348
4. kromosom = [true, false, true, true, true, true] acc = 95.65217391304348
5. kromosom = [true, false, true, true, true, true] acc = 95.65217391304348
6. kromosom = [true, false, false, true, true, true] acc = 95.65217391304348
7. kromosom = [true, true, true, true, false, true] acc = 95.65217391304348
8. kromosom = [false, true, true, true, false, true] acc = 93.47826086956522
9. kromosom = [true, true, false, true, false, true] acc = 93.47826086956522
10. kromosom = [true, true, true, false, true, true] acc = 93.47826086956522
11. kromosom = [true, false, true, true, false, true] acc = 93.47826086956522
12. kromosom = [true, true, false, true, true, true] acc = 93.47826086956522
13. kromosom = [false, true, false, true, false, true] acc = 93.47826086956522
14. kromosom = [true, false, true, true, false, true] acc = 93.47826086956522
15. kromosom = [false, true, false, true, false, true] acc = 93.47826086956522
16. kromosom = [true, true, true, false, true, true] acc = 93.47826086956522
17. kromosom = [true, true, true, false, false, true] acc = 91.30434782608695
18. kromosom = [true, true, false, false, false, true] acc = 91.30434782608695
19. kromosom = [true, true, false, false, false, true] acc = 91.30434782608695
20. kromosom = [true, true, false, false, false, true] acc = 91.30434782608695

IV. KESIMPULAN

Berdasarkan analisis terhadap hasil pengujian yang telah dilakukan terhadap klasifikasi random forest dan klasifikasi random forest yang dioptimalkan dengan algoritma genetika dengan 126 data tenaga kerja, maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Hasil Perangkat lunak dapat melakukan pengujian algoritma sehingga didapatkan pola prediksi penerimaan tenaga kerja beserta angka keakuratannya.
2. Dari Hasil keakuratan pola yang dihasilkan oleh random forest yang dioptimalkan dengan algoritma genetika lebih tinggi dibanding dengan hanya menggunakan random forest saja. Dibuktikan pada hasil perhitungan akurasi random forest yang berkisar antara 40%-95%, sedangkan hasil perhitungan akurasi dengan pengoptimalan algoritma genetika menunjukkan hasil berkisar antara 91%-95%.
3. Didapatkan kromosom-kromoson terbaik hasil dari klasifikasi random forest yang dioptimalkan dengan algoritma genetika sebagai acuan untuk menerima tenaga kerja baru.

A. Saran

Berdasarkan hasil yang dicapai pada tugas akhir ini, ada beberapa hal yang penulis sarankan untuk pengembangan selanjutnya yaitu:

1. Data yang dipakai dalam perangkat lunak ini masih sedikit dan bervariasi. Dalam perhitungan data mining, sebaiknya data harus dalam jumlah yang banyak karena mempengaruhi pola dan hasil keakuratan yang didapatkan. Untuk penelitian selanjutnya dapat menggunakan data yang lebih banyak.
2. Pengembangan perangkat lunak ini dapat dilakukan dengan menambah menu input data baru demi manfaat yang lebih baik.
3. Berdasarkan pola yang dihasilkan maka diperlukan metode terkait pengambilan keputusan untuk penerimaan tenaga kerja selanjutnya

DAFTAR PUSTAKA

- [1] V. D. Lievens and Anderson, *Recent Trends and Challenges in Personnel Selection*. ABI/INFORM Global, 2002.
- [2] C. F. Chien and L. F. Chen., "Data mining to improve personnel selection and enhance human capital: A case study in high-technology industry," *Expert Syst. Appl.*, pp. 280–290, 2008.
- [3] E. Elyan and M. M. Gaber, "A Genetic Algorithm Approach to Optimising Random Forests Applied to Class Engineered Data," United Kingdom, 2016.
- [4] J. Han, K. Micheline, and J. Pei, *Data Mining Concept and Techniques*. USA: Morgan Kaufman Publisher, 2012.
- [5] F. Gorunescu, *Data Mining: Concepts, Models, and Techniqu*. Heidelberg: Springer, 2011.
- [6] L. Breiman, "Machine Learning," 2001.
- [7] R. L. Haupt and S. E. Haupt, *Practical Genetic Algorithms*, 2nd ed. John Wiley & Sons, 2004.