

## KOMPARASI KINERJA ALGORITMA C4.5, GRADIENT BOOSTING TREES, RANDOM FORESTS, DAN DEEP LEARNING PADA KASUS EDUCATIONAL DATA MINING

Siti Mutrofin <sup>\*1</sup>, M. Mughniy Machfud <sup>2</sup>, Diema Hernyka Satyareni <sup>3</sup>, Raden Venantius Hari Ginardi <sup>4</sup>, Chastine Fatichah <sup>5</sup>

<sup>1,2,3</sup> Sistem Informasi, Universitas Pesantren Tinggi Darul Ulum, <sup>4,5</sup> Teknik Informatika, Institut Teknologi Sepuluh Nopember

Email: <sup>1</sup> sitimutrofin@ft.unipdu.ac.id, <sup>2</sup> machfudmughniy03@gmail.com, <sup>3</sup> diemahernyka@ft.unipdu.ac.id, <sup>4</sup> hari.ginardi@gmail.com, <sup>5</sup> chastine@if.its.ac.id

\*Penulis Korespondensi

(Naskah masuk: 4 November 2019, diterima untuk diterbitkan: 25 November 2019)

### Abstrak

Penentuan jurusan di SMA Negeri 1 Jogoroto, Jombang, Jawa Timur menggunakan kurikulum 2013, di mana penentuan jurusan siswa tidak hanya melibatkan keinginan siswa, tes peminatan yang dilakukan siswa di SMA pada minggu pertama, tetapi juga dilengkapi dengan nilai siswa semasa di SMP (nilai rapor siswa, nilai Ujian Nasional, serta rekomendasi guru Bimbingan Konseling), rekomendasi orang tua siswa. Selama ini, sekolah menggunakan proses konvensional dalam menentukan jurusan, yaitu menggunakan Microsoft Excel, yang cenderung lama serta rawan akan kekeliruan dalam melakukan penghitungan. Penentuan jurusan ini dilakukan setiap awal ajaran baru pada siswa baru kelas X. Rata-rata setiap tahun, sekolah mengelola siswa sejumlah 290 dengan waktu dan sumber daya manusia yang terbatas. Pada penelitian ini, penggunaan algoritma ID3 tidak cocok karena data bertipe numerik, sedangkan ID3 hanya mampu menggunakan data bertipe nominal maupun polinomial, sehingga diganti algoritma C4.5. Namun, beberapa penelitian mengatakan algoritma C4.5 memiliki kinerja kurang bagus dibandingkan algoritma *Gradient Boosting Trees*, *Random Forests*, dan *Deep Learning*. Untuk itu, dilakukan perbandingan antara keempat metode tersebut untuk melihat keefektifannya dalam menentukan jurusan di SMA. Data yang digunakan pada penelitian ini adalah data penerimaan siswa baru tahun ajaran 2018/2019. Hasil dari penelitian ini menunjukkan jika atribut yang digunakan bertipe polinomial dengan *Deep Learning* memiliki kinerja paling unggul untuk semua algoritma jika menggunakan fungsi *activation ExpRectifier*. Sedangkan jika atributnya bertipe numerik, *Deep Learning* memiliki kinerja paling unggul untuk semua algoritma jika menggunakan fungsi Tanh untuk semua *random sampling*. Namun, *Deep Learning* memiliki kinerja paling buruk untuk semua algoritma jika menggunakan *loss Function* berupa absolut.

**Kata kunci:** C4.5, deep learning, gradient boosting trees, penentuan jurusan, random forest, SMA

## PERFORMANCE COMPARISON BETWEEN ALGORITHMS OF C4.5, GRADIENT BOOSTING TREES, RANDOM FORESTS, AND DEEP LEARNING IN EDUCATIONAL DATA MINING

### Abstract

In SMAN 1 Jombang, East Java, the process of determining the students' majors referred to the 2013 curriculum in which not only the students' own choices and specialization tests conducted in their first week of SMA were considered but also the student's SMP grades (a report card, UN scores, and counseling teacher's recommendation) and parents' recommendation. So far, the school had used Microsoft Excel which required a long time to do and was prone to calculation errors in the process of determination. The process was carried out, with limited time and human resources, at the beginning of a new academic year for grade X students, consisting of 290 students on average. In this present research, the use of ID3 algorithm was not suitable because of its numeric data type instead of nominal or polynomial data. Thus, the C4.5 algorithm was applied, instead. However, the performance of C4.5 algorithm was proved lower than the algorithms of Gradient Boosting Trees, Random Forests, and Deep Learning. Hence, a comparison of performance between them was done to see their effectiveness in the process. The data was the list of new students of the academic year 2018/2019. The results showed that if the attributes are polynomial, the Deep Learning algorithm had the best

performance when using the *ExpRectifier* activation function. When they were numeric, *Deep Learning* has the most superior performance when using the *Tanh* function. However, *Deep Learning* has the worst performance when using the loss function in the form of absolute.

**Keywords:** C4.5, deep learning, determining majors, gradient boosting trees, high school, random forest.

## 1. PENDAHULUAN

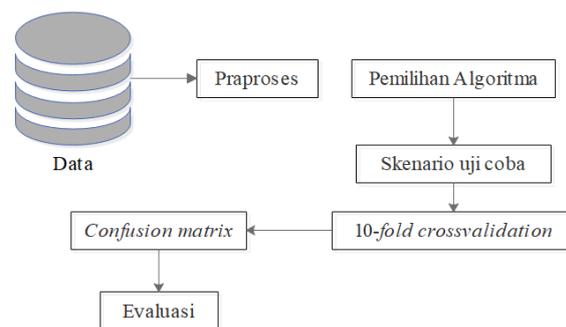
Penentuan jurusan di tingkat Sekolah Menengah Atas (SMA) Negeri 1 Jogoroto, Jombang, Jawa Timur saat ini berbasis Kurikulum 2013 (K13), yang mana pada kurikulum ini penentuan jurusan tidak hanya melibatkan keinginan siswa dan tes peminatan, tetapi juga dilengkapi dengan nilai rapor siswa selama masa pendidikan di tingkat Sekolah Menengah Pertama (SMP), nilai Ujian Nasional (UN), rekomendasi orang tua, serta rekomendasi guru Bimbingan Konseling (BK) samasa pendidikan di SMP.

Selama ini proses untuk menentukan jurusan masih bersifat konvensional menggunakan Microsoft Excel, yang mana penentuan seperti ini rawan akan kekeliruan dalam melakukan penghitungan, jika tidak benar-benar berhati-hati dan membutuhkan waktu yang lama. Sedangkan penentuan jurusan ini dilakukan setiap awal ajaran baru pada siswa baru kelas X, rata-rata setiap tahun mengelola siswa sejumlah 289 dengan waktu dan sumber daya manusia (SDM) yang terbatas.

Permasalahan ini telah menjadi perhatian banyak penelitian, diantaranya menerapkan *Educational Data Mining* (Peña-Ayala, 2014; Khan & Ghosh, 2018; Márquez- Vera, et al., 2016; Natek & Zwillig, 2014; Márquez-Vera, et al., 2013) berbasis klasifikasi untuk kasus yang sama yaitu penentuan jurusan di tingkat SMA dengan menggunakan algoritma kombinasi antara algoritma kNN dan SMART (Kustiyahningsih & Syafa'ah, 2015), ada pula yang menggunakan algoritma ID3 (Kristanto, 2014). Namun, tidak semua data cocok menggunakan algoritma tersebut dari sisi karakteristik atribut datanya. Pada penelitian ini data yang digunakan tidak cocok jika menggunakan algoritma ID3 karena data memiliki karakteristik tipe data berupa numerik, sedangkan pada ID3 hanya mampu menggunakan data bertipe nominal maupun polinomial, sebagai penggantinya adalah algoritma yang bisa digunakan adalah C4.5 (Mutrofin, et al., 2019). Namun, beberapa penelitian menyatakan algoritma C4.5 masih memiliki kinerja kurang bagus dibandingkan algoritma *Gradient Boosting Trees*, *Random Forests* (Brown & Mues, 2012), dan *Deep Learning* (Handhayani, et al., 2018) tetapi untuk kasus lain, bukan penentuan jurusan di SMA, untuk itu pada penelitian ini akan dilakukan perbandingan antara keempat metode tersebut untuk membuktikan pernyataan tersebut pada kasus ini.

## 2. METODE PENELITIAN

Metode penelitian yang diusulkan disajikan pada Gambar 1, adapun penjelasan dari Gambar 1 disajikan pada kalimat berikut ini. Tahapan pertama adalah pengumpulan data. Data yang digunakan pada penelitian ini didapatkan dari SMA Negeri 1 Jogoroto yang di bawah naungan Pemerintahan Kabupaten Jombang, Jawa Timur. Data asli berjumlah 291 siswa dari siswa baru tahun ajaran 2018/2019, karena penentuan jurusan pada Kurikulum 2013 ditentukan di tahun pertama pendidikan SMA. Data terdiri dari siswa yang terpilih pada jurusan IPA (Ilmu Pengetahuan Alam) sejumlah 124 siswa, IPS (Ilmu Pengetahuan Sosial) sejumlah 134 siswa, dan Bahasa sejumlah 33 siswa.



Gambar 1. Metode yang diusulkan

Namun pada siswa yang terpilih pada jurusan IPS dan Bahasa masing-masing ada satu siswa yang datanya terdapat *missing value*, sehingga dilakukan tahap kedua berupa praproses data, pada tahapan pertama dari praproses dilakukan reduksi data dengan cara menghapus data yang berupa *missing value*. Oleh karena itu, total data yang digunakan adalah sejumlah 289 yang terdiri dari siswa yang terpilih pada jurusan IPA sejumlah 124 siswa, IPS sejumlah 133 siswa, dan Bahasa sejumlah 32 siswa. Data asli memiliki 15 atribut yang terdiri dari nomor urut, jalur masuk, nama siswa, jenis kelamin siswa, nilai rapor semua rumpun mata pelajaran IPA, IPS dan Bahasa selama menempuh pendidikan di jenjang SMP (Sekolah Menengah Pertama) dari kelas VII hingga kelas IX, nilai rapor tertinggi dari 3 rumpun mata pelajaran (IPA, IPS dan Bahasa), nilai Ujian Nasional Matematika dan IPA (UN MIPA), jurusan berdasarkan pilihan siswa, jurusan berdasarkan pilihan orang tua, jurusan berdasarkan pilihan guru Bimbingan Konseling (BK) selama di SMP, nilai peminatan, nilai total (nilai rapor tertinggi, nilai UN MIPA, jurusan pilihan siswa, jurusan berdasarkan

pilihan orang tua, jurusan berdasarkan pilihan guru BK, dan nilai tes peminatan), dan terakhir nilai yang sudah diperingkat menjadi 3 kategory IPA (nilai total 502,5-385), IPS (nilai total 382,5-290) dan Bahasa (nilai total 288,5-212,5) berdasarkan nilai total. Praproses data yang kedua adalah memilih nilai rapor selama SMP yang tertinggi dari tiga rumpun mata pelajaran. Jika total nilai rapor yang tertinggi adalah rumpun IPA maka diberikan nilai 75 untuk nilai rapor, sedangkan jika tertinggi adalah IPS maka diberi nilai 50, dan jika nilai tertinggi adalah Bahasa maka diberi nilai 25.

Tabel 1. Atribut penentuan rekomendasi jurusan berdasarkan nilai rapor dari kelas X sampai kelas XII selama di SMP

Bahasa	MIPA	IPS	Nilai Rapor	Keterangan
1006	<b>1015</b>	1013	75	IPA
969	<b>972</b>	959	75	IPA
986	980	<b>990</b>	50	IPS
1083	1058	<b>1118</b>	50	IPS
819,7	<b>820,9</b>	812,7	75	IPA

Praproses ketiga adalah memilih atribut yang tepat untuk digunakan dalam proses klasifikasi penentuan jurusan pada tingkat Sekolah Menengah Atas (SMA), di mana atribut yang terpilih sejumlah 6 atribut yang terdiri dari nilai rapor tertinggi, nilai UN, pilihan siswa itu sendiri, pilihan orang tua siswa, rekomendasi guru BK, dan hasil tes peminatan di minggu pertama menjadi siswa di SMA Negeri 1 Jogoroto. Data asli sesungguhnya adalah berupa numerik, namun untuk semua nilai atribut kecuali nilai UN hanya terdiri dari 3 jenis, yaitu nilai 75 yang sudah pasti akan direkomendasikan pada jurusan IPA (Ilmu Pengetahuan Alam), 50 yang sudah pasti akan direkomendasikan pada jurusan IPS (Ilmu Pengetahuan Sosial), dan 25 yang sudah pasti akan direkomendasikan pada jurusan Bahasa. Oleh karena itu, pengujian nanti akan melibatkan pengujian ketika nilai atribut selain nilai UN, bertipe numerik juga bertipe polinomial, di mana 75 (IPA), 50 (IPS), dan 25 (Bahasa). Nilai rapor diambil dari nilai kelas X hingga kelas XII di masa pendidikan SMP, yang terdiri dari rumpun mata pelajaran Bahasa untuk jurusan Bahasa, rumpun mata pelajaran Matematika dan IPA untuk jurusan IPA, dan rumpun mata pelajaran IPS untuk jurusan IPS. Nilai tertinggi dari masing-masing 3 mata pelajaran tersebut yang akan terpilih sebagai rekomendasi jurusan berdasarkan nilai rapor. Contoh rekomendasi jurusan berdasarkan nilai rapor disajikan dalam Tabel 1.

Tahap ketiga adalah pemilihan algoritma, diantaranya adalah algoritma dari *Decision Tree* yang diwakili oleh C4.5 yang sudah terbukti digunakan pada penelitian sejenis dalam penentuan jurusan tetapi di tingkat perguruan tinggi (Swastina, 2013), karena atribut tidak hanya bertipe numerik tetapi juga polinomial. Selain C4.5 juga akan digunakan algoritma *Gradient Boosting Trees*, *Random Forests* (Brown & Mues, 2012) dan *Deep*

*Learning* (Handhayani, et al., 2018) karena sudah terbukti kehandalannya di bidang klasifikasi.

Tahap keempat adalah pengujian. Pengujian akan dilakukan menggunakan 2 skenario untuk rekomendasi berdasarkan nilai, yaitu skenario uji coba data bertipe numerik berupa integer karena bilangan bulat, seperti pada Tabel 1 kolom Nilai Rapor dan skenario uji coba 2 bertipe polinomial karena nilai 75 akan dikonversi menjadi IPA, 50 menjadi IPS dan 25 menjadi Bahasa, seperti pada Tabel 1 kolom Keterangan.

Tabel 2. Atribut penentuan rekomendasi jurusan berdasarkan pilihan siswa, orang tua siswa, guru BK, dan tes peminatan

Siswa	Orang tua	Guru BK	Tes Peminatan	Keterangan
75	75	75	75	IPA
75	75	75	75	IPA
50	50	50	50	IPS
50	50	50	50	IPS
25	25	25	25	Bahasa

Tabel 3. Atribut penentuan rekomendasi jurusan berdasarkan nilai UN SMP

UN SMP
Nilai UN
127,5
122,5
147,5
145,0
102,5

Tabel 4. Tipe data atribut yang digunakan

Atribut	Skenario Uji		Tipe Data	Keterangan
	Coba			
	1	2		
Nilai rapor, pilihan siswa, pilihan orang tua, rekomendasi guru BK, dan tes peminatan	75		Integer	75 (IPA), 50 (IPS), dan 25 (Bahasa)
Nilai UN		IPA IPS Bahasa	Polinomial	
			Real	

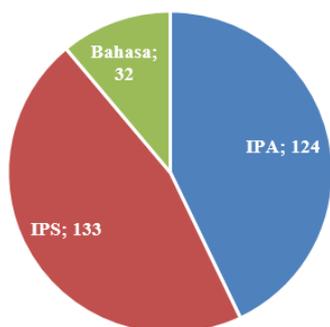
Tabel 5. Jumlah rekomendasi jurusan berdasarkan atributnya

Atribut	Jumlah		
	IPA	IPS	Bahasa
Rapor	52	53	184
Siswa	204	73	12
Orang tua	195	72	22
Guru BK	145	87	57
Tes Peminatan	28	26	235

Skenario uji coba 1 dan skenario uji coba 2 ini juga akan diterapkan untuk atribut jurusan berdasarkan pilihan siswa itu sendiri, jurusan berdasarkan pilihan orang tua siswa, jurusan berdasarkan rekomendasi guru BK selama di SMP, dan jurusan berdasarkan hasil tes peminatan karena memiliki kasus yang sama, lebih jelas untuk ilustrasi data tersebut akan disajikan pada Tabel 2.

Sedangkan untuk nilai UN memiliki tipe numerik berupa tipe *real* karena bernilai desimal, karena nilainya heterogen. Ilustrasi nilai UN dapat dilihat pada Tabel 3. Lebih lengkap terkait tipe data untuk semua atribut disajikan pada Tabel 4. Jumlah rekomendasi jurusan berdasarkan semua atribut kecuali UN diperlihatkan pada Tabel 5. Sedangkan

jumlah distribusi kelas ditunjukkan pada Gambar 2 yang memperlihatkan bahwa data pada penelitian ini memiliki karakteristik *imbalanced class distribution* (Márquez-Vera, et al., 2013), karena jumlah kelas tidak seimbang distribusinya (Li & Sun, 2012), kelas Bahasa paling sedikit dibandingkan dengan jumlah kelas IPA dan IPS. Bahasa hanya sejumlah 32 dari 289 atau 0,11%, IPA sejumlah 124 dari 289 atau 0,43%, dan IPS sejumlah 133 dari 289 atau 0,46%.



Gambar 2. Diagram *imbalanced class distribution*

Tahap kelima adalah validasi. Metode validasi yang digunakan pada penelitian ini seperti penelitian pada umumnya yang menggunakan *10-fold cross validation* (Bavan, et al., 2019).

Sedangkan untuk tahapan keenam adalah evaluasi kinerja algoritma. Evaluasi kinerja algoritma klasifikasi hanya menggunakan (Tu, et al., 2009) yang didapat dari *confusion matrix*, karena ketiga evaluasi tersebut sudah dapat menggambarkan kinerja algoritma klasifikasi (Hussain, et al., 2019). Skenario uji coba dilakukan sebanyak 150 kali, di mana skenario uji coba berdasarkan tipe numerik dan polinomial masing-masing sebesar 75 kali. 4 algoritma, masing-masing algoritma dilakukan uji coba berdasarkan 3 *sampling techniques* yaitu *linear sampling*, *shuffle sampling* dan *stratified sampling*. Untuk C4.5 dan *Random Forest* masing-masing menggunakan 4 parameter *criteria* yang terdiri dari *Gain Ratio*, *Information Gain*, *Gini Index* dan *Accuracy*. Sedangkan untuk *Deep Learning* masing-masing *random sampling* digunakan 4 fungsi aktivasi yang terdiri dari Tanh, Rectifier, Maxout dan ExpRectifier. Terakhir masing-masing fungsi aktivasi dilakukan pengujian berdasarkan *loss function* yang terdiri dari *Quadratic*, *CrossEntropy*, *Huber*, dan *Absolute*. Semua skenario pengujian ini digunakan untuk melihat pengaruh masing-masing random dan parameter terhadap data pada penelitian ini. *Software* yang digunakan pada penelitian ini adalah Rapid Miner Studio 9.3.001.

Tahap ketujuh adalah membandingkan hasil evaluasi masing-masing algoritma dari berbagai skenario uji coba, untuk mengetahui algoritma mana yang unggul dan yang lemah dalam permasalahan penelitian ini.

### 3. HASIL DAN PEMBAHASAN

Bagian ini akan menyajikan hasil uji coba yang telah dilakukan dalam penelitian ini dengan 150 kali uji coba dengan membandingkan berbagai macam algoritma dan parameter masing-masing algoritma.

#### 3.1. C4.5

Tabel 6. Hasil uji coba algoritma C4.5 dengan data bertipe integer

Random Sampling	Criterion	Accuracy (%)	Weighted Mean Recall (%)	Weighted Mean Presisi (%)
Linear	Gain Ratio	79,61	28,9	33,05
	Information Gain	73,74	26,59	32,46
	Gini Index	71,33	26,1	32,89
	Accuracy	64,27	23,69	29,34
Shuffled	Gain Ratio	92,04	88,85	88
	Information Gain	86,54	85,07	86,64
	Gini Index	89,64	87,05	87,99
	Accuracy	78,52	76,58	78,2
Stratified	Gain Ratio	89,64	85,83	86,7
	Information Gain	84,79	79,84	84,32
	Gini Index	89,63	85,79	89,55
	Accuracy	77,84	77,19	79,21

Tabel 7. Hasil uji coba algoritma C4.5 dengan data bertipe polinomial

Random Sampling	Criterion	Accuracy (%)	Weighted Mean Recall (%)	Weighted Mean Presisi (%)
Linear	Gain Ratio	76,13	27,73	29,7
	Information Gain	74,4	27,38	29,91
	Gini Index	72,34	26,68	29,89
	Accuracy	67,39	24,43	28,91
Shuffled	Gain Ratio	91,37	90	87,26
	Information Gain	86,88	84,06	84,57
	Gini Index	87,57	88	84,74
	Accuracy	74,74	74,7	75,66
Stratified	Gain Ratio	88,26	85,91	84,6
	Information Gain	87,23	83,15	83,53
	Gini Index	85,84	83,55	83,29
	Accuracy	75,07	74,22	75,05

Berdasarkan hasil 24 kali uji coba pada algoritma C4.5 dengan menggunakan 4 *criterion*, baik data bertipe integer maupun bertipe polinomial, C4.5 memiliki kinerja terbaik ketika menggunakan *shuffled sampling* dengan *criterion Gain ratio*. Namun, C4.5 memiliki kinerja terburuk ketika menggunakan *linear sampling* dengan *criterion accuracy*. Sedangkan untuk perbandingan antara tipe data, tipe data integer memiliki kinerja bagus ketika menggunakan *shuffled sampling*, tetapi data bertipe integer juga memiliki kinerja terburuk ketika menggunakan *linear sampling*, seperti yang ditampilkan pada Tabel 6 dan Tabel 7.

#### 3.2. Gradient Boosting Trees

Berdasarkan hasil 6 kali uji coba pada algoritma *Gradient Boosting Trees* dengan menggunakan *distribution multinomial* baik data bertipe integer maupun bertipe polinomial, *Gradient Boosting Trees* memiliki kinerja sama dengan C4.5, terbaik ketika menggunakan *shuffled sampling*. Namun, *Gradient Boosting Trees* memiliki kinerja terburuk ketika menggunakan *linear sampling*. Sedangkan untuk perbandingan antara tipe data,

*Gradient Boosting Trees* memiliki kinerja berbeda dengan C4.5 tipe data integer memiliki kinerja bagus ketika menggunakan *shuffled sampling*, tetapi data bertipe polinomial memiliki kinerja terburuk ketika menggunakan *linear sampling*, seperti yang ditampilkan pada Tabel 8 dan Tabel 9.

Tabel 8. Hasil uji coba algoritma *Gradient Boosting Trees* dengan data bertipe integer

<i>Random Sampling</i>	<i>Accuracy (%)</i>	<i>Weighted Mean Recall (%)</i>	<i>Weighted Mean Presisi (%)</i>
<i>Linear</i>	<b>74,82</b>	<b>26,93</b>	<b>28,94</b>
<i>Shuffled</i>	<b>91,71</b>	<b>91,6</b>	<b>89,75</b>
<i>Stratified</i>	89,99	89,47	88,15

Tabel 9. Hasil uji coba algoritma *Gradient Boosting Trees* dengan data bertipe polinomial

<i>Random Sampling</i>	<i>Accuracy (%)</i>	<i>Weighted Mean Recall (%)</i>	<i>Weighted Mean Presisi (%)</i>
<i>Linear</i>	<b>73,44</b>	<b>26,47</b>	<b>28,94</b>
<i>Shuffled</i>	<b>91,35</b>	<b>91,32</b>	<b>89,54</b>
<i>Stratified</i>	89,99	89,47	88,15

### 3.3. Random Forest

Tabel 10. Hasil uji coba algoritma *Random Forest* dengan data bertipe integer

<i>Random Sampling</i>	<i>Creterion</i>	<i>Accuracy (%)</i>	<i>Weighted Mean Recall (%)</i>	<i>Weighted Mean Presisi (%)</i>
<i>Linear</i>	<i>Gain Ratio</i>	<b>81,7</b>	<b>29,6</b>	<b>33,05</b>
	<i>Information Gain</i>	81,01	29,55	<b>33,05</b>
	<i>Gini Index</i>	81,35	29,45	32,91
<i>Shuffled</i>	<i>Accuracy</i>	<b>71,45</b>	<b>26,14</b>	<b>29,37</b>
	<i>Gain Ratio</i>	92,44	92,56	92,33
	<i>Information Gain</i>	<b>93,78</b>	<b>92,64</b>	<b>92,89</b>
<i>Stratified</i>	<i>Gini Index</i>	93,44	92,44	92,66
	<i>Accuracy</i>	<b>84,06</b>	<b>81,35</b>	<b>84,34</b>
	<i>Gain Ratio</i>	<b>92,76</b>	91,52	90,07
	<i>Information Gain</i>	92,75	<b>91,54</b>	<b>91,88</b>
	<i>Gini Index</i>	92,4	89,55	91,66
	<i>Accuracy</i>	<b>82</b>	<b>74,17</b>	<b>81,68</b>

Tabel 11. Hasil uji coba algoritma *Random Forest* dengan data bertipe polinomial

<i>Random Sampling</i>	<i>Creterion</i>	<i>Accuracy (%)</i>	<i>Weighted Mean Recall (%)</i>	<i>Weighted Mean Presisi (%)</i>
<i>Linear</i>	<i>Gain Ratio</i>	80,31	<b>31,28</b>	<b>34,23</b>
	<i>Information Gain</i>	<b>81,03</b>	29,6	<b>33,35</b>
	<i>Gini Index</i>	<b>79,99</b>	<b>29,27</b>	33,38
<i>Shuffled</i>	<i>Accuracy</i>	81,03	29,66	33,5
	<i>Gain Ratio</i>	<b>93,09</b>	94,36	<b>89,79</b>
	<i>Information Gain</i>	<b>94,13</b>	<b>95,14</b>	<b>92,95</b>
<i>Stratified</i>	<i>Gini Index</i>	<b>93,09</b>	<b>94,27</b>	90,08
	<i>Accuracy</i>	<b>93,09</b>	94,44	90,59
	<i>Gain Ratio</i>	92,75	<b>92,06</b>	<b>89,63</b>
	<i>Information Gain</i>	<b>93,44</b>	<b>93,42</b>	<b>91,07</b>
	<i>Gini Index</i>	<b>92,06</b>	92,42	90,19
	<i>Accuracy</i>	92,4	92,67	90,37

Berdasarkan hasil 24 kali uji coba pada algoritma *Random Forest* dengan menggunakan 4 *criterion*, baik data bertipe integer maupun bertipe polinomial, *Random Forest* memiliki kinerja sama seperti C4.5 dan *Gradient Boosting Trees* terbaik ketika menggunakan *shuffled sampling* dengan *criterion Gain ratio*. Namun, *Random Forest* memiliki kinerja terburuk ketika menggunakan

*linear sampling* dengan *criterion accuracy*. Sedangkan untuk perbandingan antara tipe data, tipe data polinomial *Random Forest* memiliki kinerja berbeda baik dengan C4.5 maupun *Gradient Boosting Trees*, di mana *Random Forest* bagus ketika menggunakan *shuffled sampling*, tetapi ketika data bertipe integer kinerja *Random Forest* menjadi buruk ketika menggunakan *linear sampling*, berbanding terbalik dengan *Gradient Boosting Trees*, seperti yang ditampilkan pada Tabel 10 dan Tabel 11.

### 3.4. Deep Learning

Tabel 12. Hasil uji coba algoritma *Deep Learning* dengan data bertipe integer

<i>Random Sampling</i>	<i>Activation</i>	<i>Loss Fuction</i>	<i>Accuracy (%)</i>	<i>Weighted Mean Recall (%)</i>	<i>Weighted Mean Presisi (%)</i>
<i>Linear</i>	Tanh	Quadratic	77,24	30,73	32,6
		CrossEntropy	78,28	<b>31,09</b>	<b>32,67</b>
		Huber	<b>76,55</b>	30,47	32,5
	Rectifier	Absolute	<b>80,69</b>	<b>28,98</b>	<b>31,45</b>
		Quadratic	80,69	31,66	32,88
		CrossEntropy	<b>85,52</b>	<b>33,59</b>	<b>33,83</b>
	Maxout	Huber	81,02	31,47	32,23
		Absolute	<b>57,25</b>	<b>24,01</b>	<b>29,09</b>
		Quadratic	79,98	<b>32,98</b>	<b>32,95</b>
	ExpRectifier	CrossEntropy	79,3	32,56	32,39
		Huber	<b>82,06</b>	<b>32,32</b>	<b>32,55</b>
		Absolute	<b>77,93</b>	32,5	<b>30,76</b>
<i>Shuffled</i>	Tanh	Quadratic	<b>78,28</b>	<b>30,76</b>	<b>32,17</b>
		CrossEntropy	<b>80,69</b>	<b>31,97</b>	<b>33,5</b>
		Huber	79,31	31,47	32,94
	Rectifier	Absolute	80	31,66	32,67
		Quadratic	<b>92,69</b>	93,21	93,86
		CrossEntropy	91,7	93,26	93,78
	Maxout	Huber	91,69	<b>93,71</b>	<b>94,21</b>
		Absolute	<b>83,71</b>	<b>75,37</b>	<b>78,37</b>
		Quadratic	92,72	89,39	93,83
	ExpRectifier	CrossEntropy	<b>93,45</b>	<b>91,2</b>	<b>94,26</b>
		Huber	91,71	88,71	92,81
		Absolute	<b>86,49</b>	<b>76,35</b>	<b>78,41</b>
<i>Stratified</i>	Tanh	Quadratic	<b>93,45</b>	<b>92,63</b>	<b>94,27</b>
		CrossEntropy	92,41	90,01	93,67
		Huber	89,98	88,55	87,67
Rectifier	Absolute	<b>87,86</b>	<b>75,92</b>	<b>75,15</b>	
	Quadratic	92,38	93,61	94,3	
	CrossEntropy	<b>93,08</b>	<b>94,61</b>	<b>94,52</b>	
Maxout	Huber	91,33	90,71	93,59	
	Absolute	<b>79,56</b>	<b>66,47</b>	<b>68,51</b>	
	Quadratic	90,64	92,06	92,57	
ExpRectifier	CrossEntropy	90,99	92,58	93,06	
	Huber	<b>91,33</b>	<b>92,81</b>	<b>94,09</b>	
	Absolute	<b>86,83</b>	<b>76,72</b>	<b>83,01</b>	
<i>Stratified</i>	Tanh	Quadratic	93,42	91,6	94,65
		CrossEntropy	<b>94,14</b>	<b>92,43</b>	<b>95,05</b>
		Huber	93,08	91,64	93,65
	Rectifier	Absolute	<b>90</b>	<b>76,75</b>	<b>80,86</b>
		Quadratic	<b>90,97</b>	<b>88,09</b>	<b>91,43</b>
		CrossEntropy	90,3	86,5	89,2
	Maxout	Huber	<b>89,27</b>	86,51	86,22
		Absolute	89,3	77,13	<b>82,66</b>
		Quadratic	90,64	92,04	91,88
	ExpRectifier	CrossEntropy	<b>90,99</b>	91,74	93,25
		Huber	<b>90,99</b>	<b>93,11</b>	<b>93,43</b>
		Absolute	<b>83,04</b>	<b>72,41</b>	<b>80,94</b>

Berdasarkan hasil 48 kali uji coba pada algoritma *Deep Learning* dengan menggunakan distribution multinomial baik data bertipe integer maupun bertipe polinomial, *Deep Learning* memiliki kinerja sama dengan terbaik ketika menggunakan *Stratified sampling*, dan *loss function* CrossEntropy. Namun, *Deep Learning* memiliki kinerja terburuk ketika menggunakan *linear sampling* dan *loss function absolute*. Sedangkan untuk perbandingan antara tipe data, *Deep Learning* memiliki kinerja terbaik pada tipe data integer dengan menggunakan *activation Rectifier*, tetapi data bertipe polinomial memiliki kinerja terburuk ketika menggunakan Tanh, seperti yang ditampilkan pada Tabel 12 dan Tabel 13. Mayoritas penggunaan *loss function* untuk data penelitian ini bagus ketika menggunakan CrossEntropy dan jelek ketika menggunakan

*absolute*. Untuk kinerja penggunaan atribut berupa polinomial memiliki kinerja lebih baik ketika menggunakan *stratified random*, *activation* Tahn, dan *loss function* CrossEntropy, tetapi tipe polinomial juga memiliki kinerja jelek ketika menggunakan *linear sampling*, *activation* Maxout, dan *loss function absolute* dibandingkan dengan bertipe interger tetapi juga memiliki kinerja terburuk, kebalikan dengan C4.5.

Tabel 13. Hasil uji coba algoritma *Deep Learning* dengan data bertipe polinomial

Random Sampling	Activation	Loss Fuction	Accuracy (%)	Weighted Mean Recall (%)	Weighted Mean Presisi (%)	
Linear	Tanh	Quadratic	79,66	32,81	<b>32,84</b>	
		CrossEntropy	<b>80,69</b>	33,18	<b>32,88</b>	
		Huber	<b>79,3</b>	<b>32,7</b>	32,86	
	Rectifier	Absolute	80,33	<b>33,19</b>	33,79	
		Quadratic	74,14	29,03	32,98	
		CrossEntropy	76,21	<b>30,06</b>	<b>36,48</b>	
	Maxout	Huber	<b>79,34</b>	<b>27,45</b>	<b>29,54</b>	
		Absolute	<b>80,34</b>	28,12	31,67	
		Quadratic	71,38	28,64	32,27	
	ExpRectifier	CrossEntropy	<b>77,24</b>	<b>29,79</b>	<b>32,94</b>	
		Huber	70,34	27,42	29,5	
		Absolute	<b>44,84</b>	<b>17,37</b>	<b>22,03</b>	
	Shuffled	Tanh	Quadratic	<b>79,66</b>	<b>32,85</b>	33,08
			CrossEntropy	80,34	<b>33,09</b>	32,92
			Huber	80	33,01	<b>33,38</b>
		Rectifier	Absolute	<b>87,93</b>	32,86	<b>32,08</b>
			Quadratic	91,7	92,22	87,84
			CrossEntropy	<b>93,42</b>	94,11	<b>90,01</b>
Maxout		Huber	<b>93,42</b>	<b>94,94</b>	89,96	
		Absolute	<b>86,12</b>	<b>65,88</b>	<b>61,59</b>	
		Quadratic	88,57	90,8	86,42	
ExpRectifier		CrossEntropy	<b>90,31</b>	<b>91,76</b>	<b>87,9</b>	
		Huber	84,42	83,71	80,21	
		Absolute	<b>80,6</b>	<b>72,1</b>	<b>68,46</b>	
Stratified		Tanh	Quadratic	87,19	87,94	85,22
			CrossEntropy	86,86	87,4	85,06
			Huber	<b>88,58</b>	<b>90,39</b>	<b>86,69</b>
		Rectifier	Absolute	<b>71,65</b>	<b>62,37</b>	<b>56,89</b>
			Quadratic	92,75	<b>94,54</b>	89,52
			CrossEntropy	<b>93,78</b>	93,88	<b>90,04</b>
	Maxout	Huber	93,08	93,77	89,82	
		Absolute	<b>87,87</b>	<b>65,85</b>	<b>59,66</b>	
		Quadratic	92,38	94,28	89,56	
	ExpRectifier	CrossEntropy	<b>94,82</b>	<b>96,08</b>	<b>92,43</b>	
		Huber	92,03	94,05	88,88	
		Absolute	<b>88,24</b>	<b>73,03</b>	<b>66,81</b>	
	Shuffled	Tanh	Quadratic	86,13	88,92	85,51
			CrossEntropy	<b>88,9</b>	<b>90,15</b>	<b>89,16</b>
			Huber	87,17	89,44	85,58
		Rectifier	Absolute	<b>66,38</b>	<b>52,3</b>	<b>46,08</b>
			Quadratic	86,49	87,48	86,92
			CrossEntropy	<b>87,17</b>	<b>89,1</b>	<b>87</b>
Maxout		Huber	86,15	88,91	86,81	
		Absolute	<b>73,34</b>	<b>62,46</b>	<b>57,98</b>	
		Quadratic	92,72	<b>94,58</b>	89,18	
ExpRectifier		CrossEntropy	<b>93,07</b>	92,77	<b>90,7</b>	
		Huber	91	92,42	87,25	
		Absolute	<b>85,46</b>	<b>64,17</b>	<b>57,99</b>	

Tabel 14. Perbandingan algoritma dengan data bertipe integer

Algoritma	Bagus-Buruk	Accuracy (%)	Weighted Mean Recall (%)	Weighted Mean Presisi (%)
C4.5	Linear	64,27	<b>23,69</b>	29,34
	Shuffled	92,04	88,85	88
Gradient Boosting Trees	Linear	74,82	26,93	<b>28,94</b>
	Shuffled	91,71	91,6	89,75
Random Forest	Linear	71,45	26,14	29,37
	Shuffled	94,13	<b>95,14</b>	92,95
Deep Learning	Linear	<b>57,25</b>	24,01	29,09
	Stratified	<b>94,14</b>	92,43	<b>95,05</b>

Berdasarkan semua uji coba didapatkan kesimpulan bahwa algoritma yang bagus untuk karakteristik data pada penelitian ini adalah algoritma *Deep Learning* dengan menggunakan *stratified sampling* disusul algoritma *Random Forest*. Namun, *Deep learning* bisa memiliki kinerja terburuk jika menggunakan *linear sampling* seperti yang disajikan pada Tabel 14. Ketiga algoritma kecuali *Deep Learning* sama-sama memiliki performa bagus jika menggunakan *shuffled sampling*

dikarena ketiga algoritma tersebut berbasis *tree*. Sedangkan semua *linear sampling* menghasilkan kinerja terburuk apapun algoritma dan parameternya, karena linear sampling menghasilkan *imbalanced class distribution* antara data latih dengan data uji, berbeda dengan *shuffled sampling* dan *stratified sampling* yang cenderung *balanced class distribution* antara data latih dengan data uji.

#### 4. KESIMPULAN DAN SARAN

Penelitian ini menghasilkan kesimpulan bahwa algoritma yang ungu adalah *Deep Learning* disusul oleh *Random Forest* dengan syarat menggunakan *stratified sampling*, tetapi *Deep Learning* juga bisa menghasilkan nilai terburuk dari pada algoritma lainnya jika menggunakan *Linear Sampling*. Mayoritas *linear sampling* menghasilkan kinerja terburuk. Sedangkan *shuffled sampling* memiliki kinerja bagus untuk algoritma berbasis *tree*, sedangkan *stratified sampling* cocok untuk algoritma *Deep Learning*.

Penerapan *cretarion Gain Ratio* pada C4.5 dan *Random Forest* memiliki kinerja bagus, berbading terbalik ketika menggunakan *cretarion accuracy*. Penerapan *loss function* CrossEntropy pada *Deep Learning* mayoritas memberikan hasil terbaik, sedangkan ketika menggunakan *loss function* memiliki hasil buruk. Kinerja bagus didasarkan pada hasil evaluasi *accuracy*, *weighted mean precision*, dan *weighted mean recall*. *Shuffled sampling* dan *stratified sampling* menggambarkan distribusi antar kelas seimbang baik pada data latih maupun data uji, begitu sebaliknya dengan *Linear Sampling*.

Penerapan data bertipe integer cocok untuk algoritma C4.5 sedangkan untuk *Gradient Boosting Trees*, *Random Forest*, dan *Deep Learning* cocok untuk tipe data polinomial asal tidak menggunakan *linear sampling*.

Berdasarkan hasil penelitian ini, hipotesa awal adalah benar, bahwa C4.5 lebih buruk kinerjanya dibandingkan ketiga algoritma lainnya. Ilustrasi terkait penjelasan ini dapat dilihat pada Tabel 14.

Penelitian selanjutnya diharapkan dapat menganalisis secara mendalam mengapa parameter-parameter tersebut bisa memberikan nilai bagus atau buruk.

#### UCAPAN TERIMA KASIH

Penghargaan dan terima kasih kepada Direktorat Riset dan Pengabdian Masyarakat, Dirjen Penguatan Riset dan Pengembangan, Kementerian Riset, Teknologi dan Pendidikan Tinggi, yang telah membiayai penelitian Kerjasama Antar Perguruan Tinggi 2019 dengan judul *Educational Data Mining Berbasis Klasifikasi untuk Menganalisis Mahasiswa Berpotensi Berhenti Kuliah pada Kasus Imbalanced Class Distribution*.

**DAFTAR PUSTAKA**

- BAVAN, L. et al., 2019. Adherence monitoring of rehabilitation exercise with inertial sensors: A clinical validation study. *Gait & Posture*, 70(May), pp. 211-217.
- BROWN, I. & MUES, C., 2012. An experimental comparison of classification algorithms for imbalanced credit. *Expert Systems with Applications*, 39(2012), p. Expert Systems with Applications.
- HANDHAYANI, T., HENDRYLI, J. & HIRYANTO, L., 2018. *Comparison of Shallow and Deep Learning Models for Classification of Lasem Batik Patterns*. Semarang, IEEE.
- HUSSAIN, R. G. et al., 2019. A performance comparison of machine learning classification approaches for robust activity of daily living recognition. *Artificial Intelligence Review*, 52(1), p. 357–379.
- KHAN, A. & GHOSH, S. K., 2018. Data mining based analysis to explore the effect of teaching on student performance. *Education and Information Technologies*, 23(4), p. 1677–1697.
- KRISTANTO, O., 2014. *Penerapan algoritma klasifikasi data mining ID3 untuk menentukan penjurusan siswa SMAN 6 Semarang*, Semarang: Universitas Dian Nuswantoro.
- KUSTIYAHNINGSIH, Y. & SYAFA'AH, N., 2015. Sistem pendukung keputusan untuk menentukan jurusan pada siswa sma menggunakan metode kNN dan SMART. *Jurnal Sistem Informasi Indonesia*, 1(1), pp. 19-28.
- LI, H. & SUN, J., 2012. Forecasting business failure: The use of nearest-neighbour support vectors and correcting imbalanced samples—Evidence from the Chinese hotel industry. *Tourism Management*, 33(3), pp. 622-634.
- MÁRQUEZ- VERA, C. et al., 2016. Early dropout prediction using data mining: a case study with high school students. *Expert Systems*, 33(1), pp. 107-124.
- MÁRQUEZ-VERA, C., CANO, A., ROMERO, C. & VENTURA, S., 2013. Predicting student failure at school using genetic programming and different data mining approaches with high dimensional and imbalanced data. *Applied Intelligence*, 38(3), p. 315–330.
- MUTROFIN, S. et al., 2019. *Detection of Potentially Students Drop Out of College in Case of Missing Value Using C4.5*. Bandung, IEEE.
- NATEK, S. & ZWILLING, M., 2014. Student data mining solution—knowledge management system related to higher education institutions. *Expert Systems with Applications*, 41(14), pp. 6400-6407.
- PEÑA-AYALA, A., 2014. Educational data mining: A survey and a data mining-based analysis of recent works. *Expert Systems with Applications*, 41(4), pp. 1432-1462.
- SWASTINA, L., 2013. Penerapan Algoritma C4.5 Untuk Penentuan Jurusan Mahasiswa. *Gema Aktualita*, 2(1), pp. 93-98.
- TU, M. C., SHIN, D. & SHIN, D., 2009. *Effective Diagnosis of Heart Disease through Bagging Approach*. Tianjin, China, IEEE.

*Halaman ini sengaja dikosongkan*