

# Prediksi Mahasiswa Berisiko *Drop Out* (DO) dengan ADTree dan NNge

Andri<sup>1</sup>, Paulus<sup>2</sup>

STMIK Mikroskil, Jl. Thamrin No. 112, 124, 140, Telp. (061) 4573767, Fax. (061) 4567789

<sup>1</sup>Program Studi Teknik Informatika, STMIK Mikroskil, Medan

<sup>2</sup>Program Studi Sistem Informasi, STMIK Mikroskil, Medan

<sup>1</sup>[andri@mikroskil.ac.id](mailto:andri@mikroskil.ac.id), <sup>2</sup>[paulus@mikroskil.ac.id](mailto:paulus@mikroskil.ac.id)

## Abstrak

Dengan misi meningkatkan kualitas sumber daya manusia maka perguruan tinggi wajib meningkatkan kualitas lulusan dan juga menjaga agar jumlah mahasiswa *Drop Out* (DO) tidak terlalu tinggi. Selain kualitas lulusan, jumlah mahasiswa DO juga menjadi instrumen penilaian oleh Badan Akreditasi Nasional yaitu IAPT 3.0 pada indikator nomor 51. Oleh karena itu pengendalian terhadap jumlah mahasiswa DO perlu dilakukan. Berbagai upaya yang dilakukan oleh perguruan tinggi untuk meminimalkan jumlah DO umumnya belum memanfaatkan pola data historis untuk bisa dijadikan sebuah pengetahuan.

Penelitian ini menawarkan solusi berbasis data mining untuk memprediksi mahasiswa yang berisiko DO menggunakan algoritma ADTree dan NNge. Pengumpulan dataset dari sistem informasi akademik perguruan tinggi. Kemudian data diseleksi dan nilai atribut diubah ke dalam format tertentu. Teknik evaluasi menggunakan 10-fold cross-validation. Evaluasi keseluruhan atribut sebanyak 13 dan sejumlah atribut setelah diseleksi menggunakan metode CfsSubsetEval bawaan dari aplikasi WEKA. Hasil prediksi berupa apakah mahasiswa DO atau tidak.

Model yang dibangun dengan algoritma ADTree dan NNge mampu memprediksi kelas DO. Setelah jumlah atribut diseleksi dengan metode CfsSubsetEval dari 13 menjadi 2 (rata kehadiran dan IPK), maka dihasilkan tingkat keakuratan dengan algoritma ADTree mencapai 97.25% dan F-Measure sebesar 32.7% serta tingkat keakuratan dengan algoritma NNge mencapai 96.2% dan F-Measure sebesar 34.5%.

**Kata kunci**— ADTree, Drop Out, NNge, Prediksi

## Abstract

With a mission to improve the quality of human resources, universities are required to improve the quality of graduates and also keep the number of Drop Out (DO) students not too high. The number of DO students is also an instrument of assessment by the National Accreditation Board. Therefore, it is necessary to control the number of DO students. Various efforts made by universities generally have not utilized historical data to be used as knowledge.

This study predicts students who are at risk of dropping out using the NNge and ADTree algorithms. Dataset collection is selected and the attribute values are converted into a certain format. The evaluation technique used 10-fold cross-validation. Evaluation of all 13 attributes and a number of attributes after being selected using the built-in CfsSubsetEval method from the WEKA application. Prediction results in the form of whether students drop out or not.

The model built is able to predict DO class. After the number of attributes selected using the CfsSubsetEval method from 13 to 2, the resulting accuracy rate with the ADTree algorithm reached 97.25% and the F-Measure was 32.7% and the accuracy level with the NNge algorithm reached 96.2% and the F-Measure was 34.5%.

**Keywords**— ADTree, Drop Out, NNge, Prediction

## 1. PENDAHULUAN

Dengan misi meningkatkan kualitas sumber daya manusia, maka perguruan tinggi wajib meningkatkan kualitas lulusan. Selain kualitas lulusan, jumlah mahasiswa DO juga menjadi instrumen penilaian oleh Badan Akreditasi Nasional (IAPT 3.0) pada indikator nomor 51 yaitu persentase keberhasilan studi untuk setiap program. Standar penilaian untuk skor terbaik adalah persentase jumlah mahasiswa DO tidak melebihi dari 15%. Oleh karena itu perguruan tinggi dituntut untuk melakukan pengendalian terhadap jumlah mahasiswa DO.

Berbagai upaya yang umum dilakukan oleh perguruan tinggi untuk meminimalkan jumlah DO diantaranya melalui monitoring yang dilakukan oleh dosen penasehat akademik setiap semester berdasarkan absensi/Indeks Prestasi (IP) semester/nilai ujian, program studi memberikan peringatan lisan dan tertulis kepada mahasiswa dengan kualitas akademik yang rendah. Kemudian Andri, dkk (2019) juga membuat penelitian dengan membuat aplikasi untuk menentukan tingkat performansi akademik mahasiswa menggunakan logika *fuzzy* sehingga informasi mahasiswa yang memiliki kualitas akademik rendah dapat segera diperoleh [1]. Berbagai upaya yang dilakukan untuk meminimalkan jumlah mahasiswa DO tersebut masih memanfaatkan laporan dari transaksi sistem informasi akademik dan belum memanfaatkan pola data historis untuk bisa dijadikan sebuah pengetahuan atau prediksi.

Oleh karena itu penelitian ini menawarkan solusi berbasis *data mining* untuk memprediksi mahasiswa yang berisiko DO menggunakan algoritma ADTree dan *Non-Nested Generalized Exemplars* (NNge). Pengumpulan dataset dari sistem informasi akademik perguruan tinggi. Hasil prediksi berupa apakah mahasiswa DO atau tidak. Manfaat dari penelitian ini adalah mendapatkan informasi seberapa efektif model dapat digunakan untuk memprediksi mahasiswa berisiko DO dan sebagai referensi kepada penelitian lainnya yang memanfaatkan teknik *data mining*.

Untuk menyelesaikan rumusan masalah di atas, maka dilakukan studi literatur berupa ulasan dari penelitian terdahulu yaitu:

- a. Wong, M.L dan Senthil, S (2018) melakukan studi komparatif untuk mendapatkan algoritma seleksi atribut yang paling efisien dan efektif untuk memilih atribut terbaik dalam pengembangan model prediksi performansi akademik. Penelitian ini juga melakukan perbandingan atribut yang menghasilkan akurasi, *recall* dan *area under curve* (AUC) yang lebih baik. Algoritma kelompok *rule-based* yang digunakan adalah JRip, NNge, PART dan Ridor. Algoritma kelompok berbasis pohon yang digunakan adalah ADTree, *DecisionStump*, J48 dan LADTree. Perbandingan algoritma ini pada 3 parameter yaitu *accuracy*, *recall* dan *Area Under Curve* (AUC). Algoritma seleksi atribut yang digunakan adalah *CfsSubsetEval*, *ChiSquaredAttributeEval* dan *OneRAttributeEval*. *Tool* yang digunakan adalah WEKA. Hasil penelitian menunjukkan *CfsSubsetEval* merupakan algoritma seleksi atribut yang terbaik dan efisien. Berdasarkan komponen akurasi saja dan hasil seleksi atribut dengan *CfsSubsetEval*, NNge merupakan algoritma paling akurat dari kelompok aturan dan ADTree merupakan algoritma yang paling akurat dari kelompok pohon [2]. Berdasarkan ulasan ini, penelitian ini menggunakan *CfsSubsetEval* untuk seleksi atribut serta dua algoritma klasifikasi yaitu ADtree dan NNge yang terdapat dalam alat bantu WEKA.
- b. Ho, LC dan Shim, KJ (2018) mengembangkan metode untuk identifikasi dan prediksi mahasiswa berisiko DO sedini mungkin selama menggunakan *Learning Management System* (LMS) yang dapat digunakan oleh universitas terbuka untuk pembelajaran jarak jauh. *Dataset* yang digunakan mengandung data profil mahasiswa, aktivitas dan LMS seperti seberapa sering mahasiswa berinteraksi dengan kuliah online. *Tool* yang digunakan adalah SAS *Enterprise Miner* (EM), SAS JMP, Python 3.5. Hasil analisis menunjukkan bahwa interaksi mahasiswa dalam kuliah online dapat dijadikan alat prediksi mahasiswa berisiko DO [3]. Penelitian ini juga menggunakan studi kasus mahasiswa berisiko DO.
- c. Alqahtani, E.J. (2018) membuat klasifikasi penyakit parkinson menggunakan algoritma NNge. Penyakit parkinson memiliki sedikit gejala. Oleh karena itu penelitian ini menggunakan rekaman suara untuk menghasilkan sistem diagnosis penyakit parkinson. Tingkat keakuratan dari hasil

klasifikasi pada kasus ini adalah sebesar 96.3% [4]. Berdasarkan tingkat keakuratan ini, penelitian menggunakan algoritma NNge.

## 2. TINJAUAN PUSTAKA

### 2.1 Supervised dan Unsupervised Learning

*Data mining* mengenal 2 jenis model pembelajaran (*Learning*) yaitu:

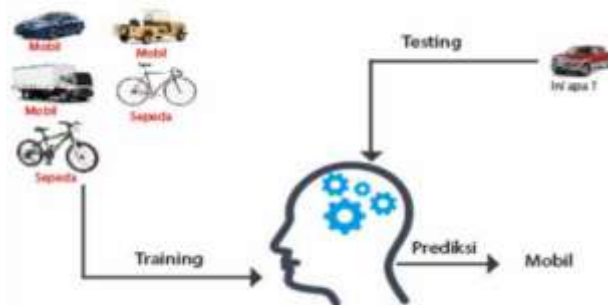
a. *Supervised Learning*

Pada model pembelajaran ini dikenal penggunaan istilah *training* dan *testing data*. *Training data* atau fase pelatihan berguna untuk pembentukan pola baru yang nantinya akan digunakan sebagai dasar dalam melakukan pengolahan data baru.

b. *Unsupervised Learning*

Pada model *unsupervised learning* berbeda dengan *supervised learning*. Model ini mengarah pada pembuatan *rules* untuk penemuan sebuah pola atau *knowledge* [5].

Semua teknik pada *data mining* dapat dikelompokkan pada 2 kategori ini berdasarkan pemodelan dari masing-masing teknik tersebut. Pada *supervised learning* terdapat istilah *data training* dan *testing* seperti yang terlihat pada Gambar 1. *Data training* adalah Data yang digunakan dalam proses *training* atau *learning* atau disebut juga dengan istilah *training set*. *Data testing* atau data uji adalah evaluasi terhadap performa algoritma tersebut. Pada proses *testing*, performa algoritma akan diuji menggunakan *testing set*. *Testing set* dan *training set* sebaiknya menggunakan data yang berbeda [5].



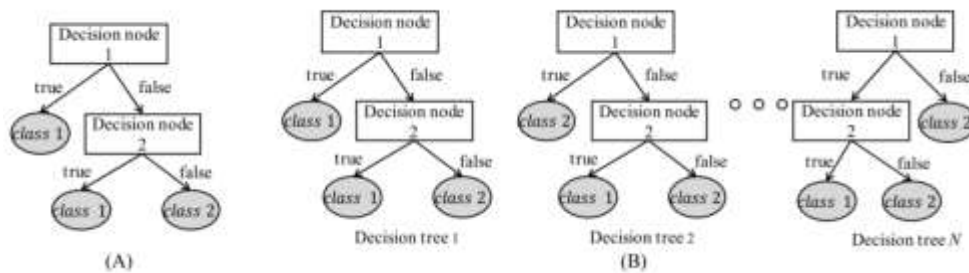
Gambar 1 Data Training dan Testing

Sumber: [5]

### 2.2 Algoritma Alternating Decision Tree (ADTree)

Pohon keputusan/*Decision Tree* (DT) adalah salah satu dari klasifikasi yang paling terkenal dalam aplikasi yang berhubungan dengan dunia nyata yang berhubungan dengan representasi pengetahuan, ketahanan, kemampuan untuk menangani *missing value* dan mudah diinterpretasikan. Algoritma pembelajaran digunakan untuk membentuk pohon keputusan secara otomatis. Sejumlah sampel yang telah benar digunakan sebagai data pelatihan. Pendekatan umum yang digunakan untuk menumbuhkan DT adalah melakukan *divide and conquer* secara rekursif terhadap data contoh untuk mendapatkan keluaran. Model DT dapat diinterpretasikan sebagai sejumlah aturan keputusan yang mengarah ke penemuan pengetahuan potensial. Sebagai contoh beberapa algoritma terkait dengan DT adalah ID3, C4.5, CART dan varian lain yang lebih baru [6].

Salah satu dari metode utama digunakan untuk meningkatkan performansi dari klasifikasi dari DT yaitu *boosting*. Algoritma *boosting* menghitung ulang distribusi data pelatihan secara iteratif setelah model prediksi lemah diperoleh. *Boosting* sering menghasilkan pohon yang besar, kompleks dan lebih sulit untuk diinterpretasikan seperti Gambar 2. Hal ini yang mendorong kelahiran dari ADTree [6].



Gambar 2 (A) Pohon Keputusan Klasik dan (B) Peningkatan Performansi Pohon Keputusan dengan Ukuran Pohon yang Lebih Besar  
Sumber: [6]

Pengenalan turunan dari *Decision Tree* (DT) yaitu ADTree diusulkan oleh Freund dan Mason pada tahun 1999. ADTree dapat merepresentasikan DT yang standar tetapi juga struktur yang lebih kompleks. ADTree terdiri dari dua tipe simpul yaitu simpul prediksi dan keputusan. Setiap simpul prediksi diisi dengan nilai nyata yang digunakan dalam pembuatan keputusan dimana semua nilai dari jalur yang dilintasi diklasifikasikan sebagai penjumlahan dan tanda dari hasil penjumlahan. Sebuah objek dapat mengikuti banyak jalur, bukan hanya jalur tunggal karena dalam ADTree, pada masing-masing simpul prediksi, semua simpul anak diuji dan menunjuk beberapa jalur yang berbeda [7].

Simpul keputusan ADTree diuraikan menjadi bentuk aturan khusus dimana prasyarat yang berhubungan dengan jalur dari akar sampai ke kondisi dan simpul keputusan merepresentasikan hasil uji dari simpul. Freund dan Mason menyatakan bahwa pendekatan mereka menunjukkan hasil yang kompetitif dengan pohon C5.0, tetapi biasanya lebih mudah menginterpretasikan pohon. Kelebihan penting lainnya adalah ADTree dapat memberikan pengukuran alami dari klasifikasi [7].

### 2.3 Algoritma Non-Nested Generalised Exemplars (NNge)

Algoritma ini diperkenalkan oleh Brent pada tahun 1995 yang menggabungkan contoh/model terbaik, membentuk *hyperrectangle* dalam ruang atribut yang merepresentasikan aturan *conjunctive* dengan *disjunction* internal. Algoritma ini membentuk generalisasi setiap waktu ada data baru yang dimasukkan ke basis data, dengan menggabungkannya ke tetangga terdekat dari kelas yang sama [8].

Algoritma melakukan pembelajaran dengan cara melakukan klasifikasi, kemudian generalisasi setiap data baru. Saat melakukan klasifikasi *record*, satu atau lebih *hyperrectangle* akan ditemukan dimana *record* tersebut adalah anggota dari kelas yang salah. Algoritma menghapus *record* ini sehingga tidak lagi menjadi member. Setelah diklasifikasikan, *record* baru digeneralisasi dengan menggabungkannya dengan contoh terdekat dari kelas yang sama, yang mungkin merupakan satu *record* atau *hyperrectangle* [8].

Generalisasi contoh menghasilkan performansi klasifikasi yang lebih baik dari *nearest neighbor* standar lainnya. NNge melakukan klasifikasi contoh baru dengan menentukan tetangga terdekat (*nearest neighbor*) di dalam basis data contoh/*hyperrectangle* menggunakan fungsi *Euclidean Distance*. Jarak  $D(T, H)$  antara *record* pelatihan  $T = (T_1, T_2, \dots, T_m)$  dan sebuah contoh atau *hyperrectangle*  $H$  diberikan sebagai berikut:

$$D(T, H) = W_H \sqrt{\sum_{i=1}^n \left[ W_i \frac{d(T_i - H_i)}{T_{i_{\max}} - T_{i_{\min}}} \right]^2} \quad (1)$$

Dimana  $T_i$  adalah nilai fitur dalam contoh,  $H_i$  adalah nilai fitur dalam *hyperrectangle*,  $W_H$  adalah bobot contoh dan  $W_i$  adalah bobot fitur,  $T_{i_{\min}}$  dan  $T_{i_{\max}}$  adalah jangkauan nilai di data pelatihan yang berkorespondensi dengan atribut  $i$  dari fitur numerik [8].

## 3. METODE PENELITIAN

### 3.1 Langkah-Langkah Penyelesaian

Langkah-langkah penyelesaian penelitian terdiri dari:

1. Persiapan  
Melakukan studi literatur dengan mencari referensi dari jurnal, buku dan website.
2. Pelatihan
  - a. Pengumpulan *Dataset*  
*Dataset* untuk pelatihan diambil dari sistem informasi akademik perguruan tinggi.
  - b. *Preprocessing*  
Data diseleksi terlebih dahulu yaitu memiliki 100% nilai atribut yang lengkap. Kemudian nilai atribut diubah ke dalam format tertentu.
  - b. Proses Pelatihan  
Data hasil *preprocessing* dilatih menggunakan algoritma ADTree dan NNge dengan *tools* WEKA 3.8.4.
3. Pengujian:
  - a. Pengumpulan *Dataset*  
*Dataset* hasil *preprocessing* juga digunakan untuk pengujian.
  - b. Proses Pengujian  
*Dataset* hasil *preprocessing* diolah menggunakan algoritma ADTree dan NNge dengan *tools* WEKA 3.8.4 untuk mendapatkan hasil prediksi. Teknik evaluasi pelatihan dan pengujian menggunakan *10-fold cross-validation* karena teknik ini menghasilkan model dengan bias yang lebih kecil dibandingkan teknik lainnya. Berbagai variasi pengujian yang dilakukan untuk melihat pengaruh terhadap hasil prediksi yaitu:
    - Jumlah iterasi *boosting* dari algoritma ADTree dengan semua atribut.
    - Jumlah iterasi *boosting* dari algoritma ADTree dengan seleksi atribut.
    - Jumlah percobaan generalisasi dari algoritma NNge dengan semua atribut.
    - Jumlah percobaan generalisasi dari algoritma NNge dengan seleksi atribut.
 Seleksi atribut menggunakan metode *CfsSubsetEval* bawaan dari aplikasi WEKA. Alasan pemakaian *CfsSubsetEval* karena merupakan metode yang paling baik dan efisien untuk memilih atribut yang penting [2].
  - c. Hasil Prediksi dan Evaluasi  
Hasil prediksi berupa apakah mahasiswa DO atau tidak. Selain itu dihitung akurasi, presisi, *recall* dan *F-measure* dari hasil prediksi kedua algoritma.

### 3.2 Data yang Digunakan

Data yang digunakan pada penelitian bersumber dari sistem informasi akademik perguruan tinggi yaitu:

1. Daftar mahasiswa yang telah lulus.
2. Daftar mahasiswa yang telah melewati masa studi maksimum namun belum lulus (sesuai dengan ketentuan SN-Dikti). Kelompok mahasiswa ini dapat berstatus akademik keluar atau mengundurkan diri.
3. Mahasiswa dari semua program studi.

Daftar mahasiswa baik yang telah lulus dan belum lulus tetapi sudah melewati masa studi maksimum dipisah menjadi dua buah *dataset* yang terlihat pada Tabel 1 dan Tabel 2 dengan rincian sebagai berikut:

Tabel 1 Informasi Dataset1

Nama Atribut	Tipe	Keterangan
Nim	String	
Kota Tempat Tinggal	String	
Kota Sewaktu SMA	String	
Nilai UN/Nem	Real	
Nilai USM (Ujian Saringan Masuk)	Char.	Berisi nilai: A, B, C, D dan X. X adalah status jalur beasiswa.

Frekuensi Ikut USM	Numerik.	Nilai 0 untuk jalur beasiswa. Nilai 1 s.d. 3 memperlihatkan frekuensi mengikuti USM
Tanggal Daftar Ulang	Date	
Program Studi	String	
Waktu Kuliah	String	Waktu kuliah terakhir mahasiswa: Pagi atau Sore.
IPK	Real	IPK terakhir. Jangkauan nilai 0 sampai 4
Total Record = 11.069		

Tabel 2 Informasi Dataset2

Nama Atribut	Tipe Data	Keterangan
Nim	String	
Persentase Absensi	Real	Jangkauan nilai 0 sampai 100%
Tugas	Real	Jangkauan nilai 0 sampai 100
UTS	Real	Jangkauan nilai 0 sampai 100
UAS	Real	Jangkauan nilai 0 sampai 100
Total Record = 116.673		

### 3.3 Alat-Alat Penelitian

- a. Perangkat Keras
 

Laptop Lenovo ThinkPad 13 2<sup>nd</sup> Gen Signature Edition dengan spesifikasi:

  - Processor: Intel® Core i7-7500 CPU @ 2.7GHz
  - RAM: 8 GB
  - Harddisk: 500 GB SSD
- b. Perangkat Lunak
  1. Microsoft Excel 2016 yang berisikan *dataset*.
  2. Microsoft SQL Server 2014 untuk mempercepat *query* atau pengolahan data.
  3. WEKA 3.8.4.

## 4. HASIL DAN PEMBAHASAN

### 4.1 Hasil

1. Pengumpulan *Dataset*

*Dataset* yang diperoleh dalam format *Microsoft Excel*.
2. *Preprocessing*
  - a. Menghapus *record* dari kedua *dataset* yang bernilai *null*/tidak sah dan memperbaiki format string menjadi numerik.
  - b. Mengelompokkan data di Dataset2 sehingga diperoleh rata-rata UTS, Tugas, UAS dan kehadiran untuk setiap mahasiswa.
  - c. Menggabungkan Dataset1 dan Dataset2 sehingga diperoleh 2396 *record* mahasiswa dimana jumlah mahasiswa dengan status lulus sebanyak 2314 *record* dan status DO sebanyak 82 *record*.
  - d. Mengubah atribut dengan nilai numerik menjadi format tertentu (Tabel 3 dan Tabel 4):

Tabel 3 Risiko DO Berdasarkan Nilai Atribut



Atribut	Tingkat Risiko DO Jika Nilai Atribut Baru Semakin Tinggi
kota tempat tinggal	Tinggi
kota tempat tinggal sewaktu SMU	Tinggi
NEM/Nilai UN	Rendah
frekuensi ikut USM	Tinggi
jangka waktu daftar ulang sebelum kuliah	Rendah
rata kehadiran	Rendah
rata tugas	Rendah
rata UTS	Rendah
rata UAS	Rendah
IPK	Rendah

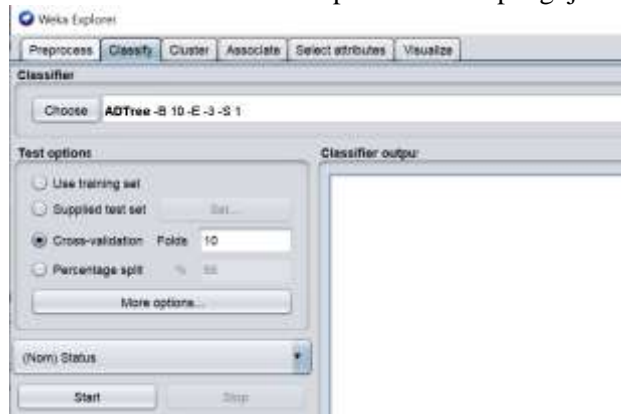
Tabel 4 Deskripsi Dataset

Atribut	Nilai Atribut Lama	Nilai Atribut Baru	Frekuensi	Persentase
kota tempat tinggal	kota Medan	1	1381	57.6
	luar Medan	2	1015	42.4
kota tempat tinggal sewaktu SMU	kota Medan	1	1328	55.4
	luar Medan	2	1068	44.6
NEM/nilai UN	$0 \leq \text{NEM} \leq 20$	1	223	9.3
	$20 < \text{NEM} \leq 40$	2	618	25.8
	$40 < \text{NEM} \leq 60$	3	1541	64.3
	$60 < \text{NEM} \leq 80$	4	3	0.1
	$80 < \text{NEM} \leq 100$	5	11	0.5
hasil USM	A	A	77	3.2
	B	B	327	13.6
	C	C	896	37.4
	D	D	504	21
	X	X	592	24.7
frekuensi ikut USM	Frekuensi = 0	1	670	28
	Frekuensi = 1	2	1649	68.8
	Frekuensi > 1	3	77	3.2
jangka waktu daftar ulang sebelum kuliah	$0 \leq \text{Jangka Waktu} \leq 7$	1	256	10.7
	$7 < \text{Jangka Waktu} \leq 115$	2	413	17.2
	$15 < \text{Jangka Waktu} \leq 30$	3	681	28.4
	$30 < \text{Jangka Waktu} \leq 60$	4	501	20.9
	$60 < \text{Jangka Waktu} \leq 172$	5	545	22.7
program studi	Prodi1	Prodi1	762	31.8
	Prodi2	Prodi2	882	36.8
	Prodi3	Prodi3	88	3.7
	Prodi4	Prodi4	297	12.4
	Prodi5	Prodi5	367	15.3
waktu kuliah	Pagi	Pagi	1392	58.1
	Sore	Sore	1004	41.9
rata kehadiran	Kehadiran < 75	1	190	7.9
	$75 \leq \text{Kehadiran} \leq 100$	2	2206	92.1

rata tugas	0 <= Tugas < 50	1	110	4.6
	50 <= Tugas < 75	2	778	32.5
	75 <= Tugas <= 100	3	1508	62.9
rata UTS	0 <= UTS < 50	1	402	16.8
	50 <= UTS < 75	2	1439	60.1
	75 <= UTS <= 100	3	555	23.2
rata UAS	0 <= UAS < 50	1	272	11.4
	50 <= UAS < 75	2	1442	60.2
	75 <= UAS <= 100	3	682	28.5
IPK	IPK < 2	1	16	0.7
	2 <= IPK < 2.75	2	195	8.1
	IPK >= 2.75	3	2185	91.2
status	Lulus	Lulus	2314	96.6
	DO	DO	82	3.4

### 3. Pelatihan dan Pengujian *Dataset*

Menggunakan aplikasi WEKA untuk melakukan pelatihan dan pengujian seperti Gambar 3.



Gambar 3 Pengaturan Ukuran Dataset Pelatihan dan Pengujian

## 4.2 Pembahasan

### 1. Pengujian metode ADTree dengan berbagai nilai iterasi *boosting*.

Gambar 4 menunjukkan pengaruh jumlah iterasi *boosting* terhadap akurasi, presisi, *recall* dan *F-Measure*. Seperti yang terlihat pada Tabel 5, jika semua atribut diikuti dalam pengujian dan semakin banyak iterasi yang dilakukan maka menghasilkan menghasilkan tingkat akurasi, presisi, *recall* dan *F-Measure* dari kelas DO yang tidak beraturan. Tingkat akurasi tertinggi dari kelas DO yaitu 97.33% diperoleh ketika jumlah iterasi = 20. Tetapi ketika jumlah iterasi = 25, dihasilkan *F-Measure* dari kelas DO terbaik (tertinggi) yaitu 54.4%.

Tabel 5 Pengujian ADTree dengan Semua Atribut

No	Jumlah Iterasi Boosting	Semua Atribut						
		Akurasi	Kelas DO			Kelas Lulus		
			Presisi	Recall	F-Measure	Presisi	Recall	F-Measure
1	5	96.66%	52.4%	26.8%	35.5%	97.5%	99.1%	98.3%
2	10	97.29%	68.1%	39%	49.6%	97.9%	99.4%	98.6%
3	15	97.25%	64.3%	43.8%	52.2%	98%	99.1%	98.6%
4	20	97.33%	65.5%	46.3%	54.3%	98.1%	99.1%	98.6%
5	25	97.2%	61.5%	48.8%	54.4%	98.2%	98.9%	98.6%
6	30	97.25%	63.8%	45.1%	52.9%	98.1%	99.1%	98.6%

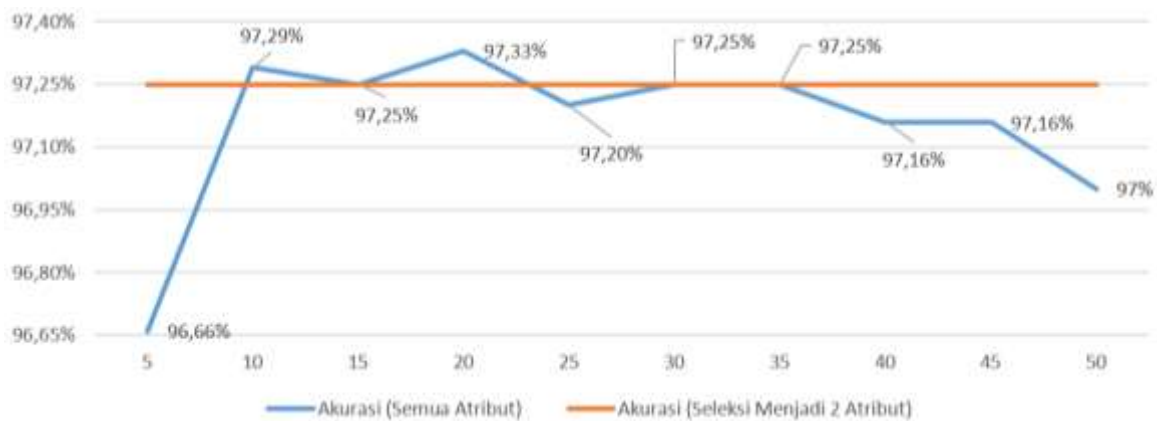


7	35	97.25%	64.3%	43.9%	52.2%	98%	99.1%	98.6%
8	40	97.16%	63%	41.5%	50%	98%	99.1%	98.5%
9	45	97.16%	63%	41.5%	50%	98%	99.1%	98.5%
10	50	97%	59.3%	39%	47.1%	97.9%	99%	98.5%
<b>Rata-Rata</b>		<b>97.16%</b>	<b>62.52%</b>	<b>42.57%</b>	<b>49.82%</b>	<b>97.97%</b>	<b>99.1%</b>	<b>98.54%</b>

Pengujian berikutnya melakukan seleksi atribut dengan metode *CfsSubsetEval*. Metode ini berfungsi untuk mengevaluasi nilai atribut dengan mempertimbangkan kemampuan prediksi dan tingkat redundansi. Seperti yang terlihat pada Tabel 6, setelah dilakukan seleksi, jumlah atribut dari 13 menjadi 2 buah yaitu rata kehadiran dan IPK. Jika hanya menggunakan 2 buah atribut hasil seleksi maka variasi dari banyaknya iterasi menghasilkan akurasi, presisi, *recall* dan *F-Measure* yang sama. Rata-rata tingkat akurasi dari kelas DO setelah melakukan seleksi atribut mengalami peningkatan dari 97.16% menjadi 97.25% atau sebesar 0.09%, tetapi rata-rata *F-Measure* mengalami penurunan dari 49.82% menjadi 32.7% atau sebesar 17,12%.

Tabel 6 Pengujian ADTree dengan Seleksi Atribut

No	Jumlah Iterasi Boosting	Seleksi Atribut dengan CfsSubsetEval						
		Akurasi	Kelas DO			Kelas Lulus		
			Presisi	Recall	F-Measure	Presisi	Recall	F-Measure
1	5	97.25%	100%	19.5%	32.7%	97.2%	100%	98.6%
2	10	97.25%	100%	19.5%	32.7%	97.2%	100%	98.6%
3	15	97.25%	100%	19.5%	32.7%	97.2%	100%	98.6%
4	20	97.25%	100%	19.5%	32.7%	97.2%	100%	98.6%
5	25	97.25%	100%	19.5%	32.7%	97.2%	100%	98.6%
6	30	97.25%	100%	19.5%	32.7%	97.2%	100%	98.6%
7	35	97.25%	100%	19.5%	32.7%	97.2%	100%	98.6%
8	40	97.25%	100%	19.5%	32.7%	97.2%	100%	98.6%
9	45	97.25%	100%	19.5%	32.7%	97.2%	100%	98.6%
10	50	97.25%	100%	19.5%	32.7%	97.2%	100%	98.6%
<b>Rata-Rata</b>		<b>97.25%</b>	<b>100%</b>	<b>19.5%</b>	<b>32.7%</b>	<b>97.2%</b>	<b>100%</b>	<b>98.6%</b>

Gambar 4. Tingkat Akurasi ADTree dengan Berbagai Nilai Iterasi *Boosting*

## 2. Pengujian metode NNge dengan berbagai jumlah percobaan generalisasi.

Gambar 5 menunjukkan pengaruh jumlah percobaan generalisasi terhadap akurasi, presisi, *recall* dan *F-Measure*. Seperti yang terlihat pada Tabel 7, jika semua atribut diikuti dalam pengujian dan semakin banyak percobaan generalisasi yang dilakukan maka menghasilkan tingkat akurasi, presisi, *recall* dan *F-Measure* dari kelas DO yang tidak beraturan. Tingkat akurasi tertinggi dari kelas DO yaitu 96.33% diperoleh ketika jumlah percobaan = 5. Tetapi ketika jumlah percobaan = 25, dihasilkan *F-Measure* dari kelas DO terbaik yaitu 43.5%.

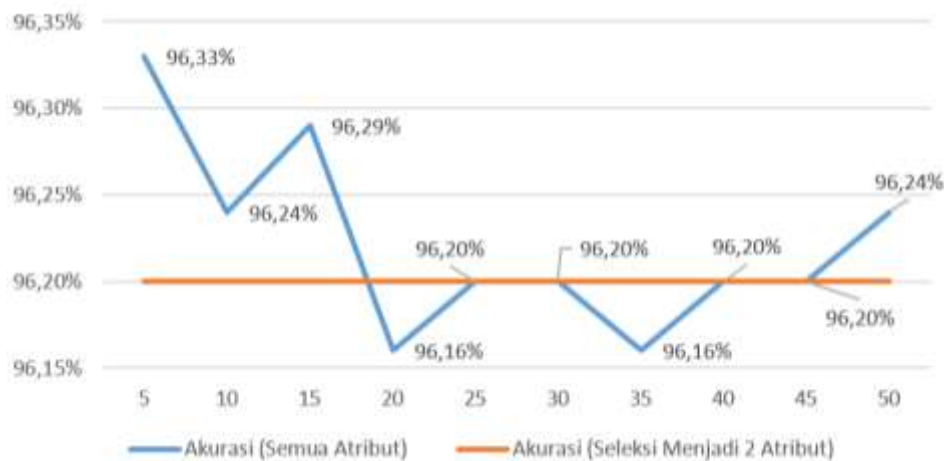
Tabel 7 Pengujian NNge dengan Semua Atribut

No	Jumlah Percobaan Generalisasi	Semua Atribut						
		Akurasi	Kelas DO			Kelas Lulus		
			Presisi	Recall	F-Measure	Presisi	Recall	F-Measure
1	5	96.33%	46.1%	42.7%	44.3%	98%	98.2%	98.1%
2	10	96.24%	44.7%	41.5%	43%	97.9%	98.2%	98.1%
3	15	96.29%	45.3%	41.5%	43.3%	97.9%	98.2%	98.1%
4	20	96.16%	43.6%	41.5%	42.5%	97.9%	98.1%	98%
5	25	96.2%	44.3%	42.7%	43.5%	98%	98.1%	98%
6	30	96.2%	44%	40.2%	42%	97.9%	98.2%	98%
7	35	96.16%	43.4%	40.2%	41.8%	97.9%	98.1%	98%
8	40	96.2%	44.2%	41.5%	42.8%	97.9%	98.1%	98%
9	45	96.2%	44.2%	41.5%	42.8%	97.9%	98.1%	98%
10	50	96.24%	44.7%	41.5%	43%	97.9%	98.2%	98.1%
<b>Rata-Rata</b>		<b>96.22%</b>	<b>44.45%</b>	<b>41.48%</b>	<b>42.9%</b>	<b>98%</b>	<b>98.15%</b>	<b>98.04%</b>

Pengujian berikutnya melakukan seleksi atribut dengan metode *CfsSubsetEval*. Seperti yang terlihat pada Tabel 8, jika hanya menggunakan 2 buah atribut hasil seleksi maka variasi dari jumlah percobaan menghasilkan akurasi, presisi, *recall* dan *F-Measure* yang sama. Rata-rata tingkat akurasi dari kelas DO setelah melakukan seleksi atribut mengalami penurunan dari 96.22% menjadi 96.2% atau sebesar 0.02%, rata-rata *F-Measure* juga mengalami penurunan dari 42.9% menjadi 34.5% atau sebesar 8,4%.

Tabel 8 Pengujian NNge dengan Seleksi Atribut

No	Jumlah Percobaan Generalisasi	Seleksi Atribut dengan CfsSubsetEval (Dua Atribut)						
		Akurasi	Kelas DO			Kelas Lulus		
			Presisi	Recall	F-Measure	Presisi	Recall	F-Measure
1	5	96.2%	42.1%	29.3%	34.5%	97.5%	98.6%	98%
2	10	96.2%	42.1%	29.3%	34.5%	97.5%	98.6%	98%
3	15	96.2%	42.1%	29.3%	34.5%	97.5%	98.6%	98%
4	20	96.2%	42.1%	29.3%	34.5%	97.5%	98.6%	98%
5	25	96.2%	42.1%	29.3%	34.5%	97.5%	98.6%	98%
6	30	96.2%	42.1%	29.3%	34.5%	97.5%	98.6%	98%
7	35	96.2%	42.1%	29.3%	34.5%	97.5%	98.6%	98%
8	40	96.2%	42.1%	29.3%	34.5%	97.5%	98.6%	98%
9	45	96.2%	42.1%	29.3%	34.5%	97.5%	98.6%	98%
10	50	96.2%	42.1%	29.3%	34.5%	97.5%	98.6%	98%
<b>Rata-Rata</b>		<b>96.2%</b>	<b>42.1%</b>	<b>29.3%</b>	<b>34.5%</b>	<b>97.5%</b>	<b>98.6%</b>	<b>98%</b>



Gambar 5. Tingkat Akurasi NNge dengan Berbagai Nilai Percobaan Generalisasi

## 5. KESIMPULAN

Beberapa kesimpulan yang dapat diambil setelah melakukan pengujian:

1. Model *data mining* yang dibangun dengan algoritma ADTree dan NNge mampu memprediksi kelas DO dari mahasiswa. Setelah jumlah atribut diseleksi dengan metode *CfsSubsetEval* dari 13 menjadi 2 (rata kehadiran dan IPK), maka dihasilkan tingkat keakuratan dengan algoritma ADTree mencapai 97.25% dan *F-Measure* sebesar 32.7% serta tingkat keakuratan dengan algoritma NNge mencapai 96.2% dan *F-Measure* sebesar 34.5%.
2. Tingkat keakuratan dari hasil prediksi dengan algoritma ADTree setelah atribut diseleksi (97.25%) adalah lebih baik daripada NNge (96.2%) tetapi nilai *F-Measure* dari algoritma NNge (34.5%) lebih baik daripada ADTree (32.7%) untuk kasus prediksi mahasiswa DO.

## 6. SARAN

Beberapa saran yang dapat diberikan untuk penelitian berikutnya yaitu:

1. Memperbaiki *dataset* sehingga dapat menghasilkan lebih banyak jumlah sampel yang sah.
2. Menguji *dataset* dengan algoritma *data mining* lain yang lebih baik.

## DAFTAR PUSTAKA

- [1] Andri, Paulus, Hanes, 2019, Penentuan kualitas akademik mahasiswa dengan logika fuzzy pada STMIK Mikroskil, *Jurnal SIFO Mikroskil*, vol 20, no 1, hal 19-32.
- [2] Wong, M.L, 2018, Applying Attribute Selection Algorithms in Academic Performance Prediction, *Springer: International Conference on Intelligent Data Communication Technologies and Internet of Things (ICICI)*, India, Agust 7-8.
- [3] Ho, L.C, Shim, K.J, 2018, Data Mining Approach to the Identification of At-Risk Students, *IEEE: International Conference on Big Data*, Seattle, Dec 10-13.
- [4] Alqahtani, E, Alshamrani, F.H, Syed, H.F, Olatunji S.O, 2018, Classification of Parkinson's Disease Using NNge Classification Algorithm, *IEEE: 21<sup>st</sup> Saudi Computer Society National Computer Conference (NCC)*, Riyadh, April 25-26.
- [5] Nengsih, W., 2019, *Implementasi Data Mining dalam Kasus Terapan*, Politeknik Caltex, Riau.
- [6] Ooi, MPL et al, 2017, Handbook of Neural Computation, *Science Direct*, hal 345-371.
- [7] Grabczewski, K, 2014, Meta-Learning in Decision Tree Induction, *Springer*, hal 96-97.
- [8] Panigrahi A, Patra M.R, 2015, Performance Evaluation of Rule Learning Classifier in Anomaly Based Intrusion Detection, *Springer: Proceedings of the International Conference on CIDM*, India, Dec 5-6.

