# PEMILIHAN *CRITERIA SPLITTING* DALAM ALGORITMA *ITERATIVE DICHOTOMISER 3* (ID3) UNTUK PENENTUAN KUALITAS BERAS : STUDI KASUS PADA PERUM BULOG DIVRE LAMPUNG

**Yusuf Elmande[1], Prabowo Pudjo Widodo[2]**

**Magister Ilmu Komputer Program Pascasarjana Universitas Budi Luhur**
[1]elmande09@yahoo.co.id; [2]prabowo_pw@yahoo.com

**ABSTRAK**

*Beras merupakan bahan makanan pokok sebagian besar penduduk dunia, termasuk penduduk Indonesia. Bangsa Indonesia telah menjadi Bangsa yang terbesar mengkonsumsi beras di dunia yaitu 105 Kg/kapita/tahun. Data mining adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Dalam penelitian ini, penulis akan mengambil algoritma Iterative Dichotomiser 3 (ID3) untuk pemilihan Criteria Splittting dalam penentuan kualitas beras. Metode penelitian yang digunakan dalam eksperimen ini menggunakan model Cross-Standard Industry for Data Mining (CRISP-DM). Dengan demikian hasil yang diharapkan adalah untuk mengetahui Criteria Splitting mana pada Algoritma Interative Dichotomieser 3 (ID3) yang paling akurat dalam menentukan kualitas beras, dan ternyata criteria splitting Gain Ratio yang memiliki Decision Tree yang akurat.*

**Kata Kunci** : *Data Mining,* Klasifikasi*, Decision Tree, RapidMiner, Splitting,* ROC

## 1. Pendahuluan

Beras merupakan bahan makanan pokok sebagian besar penduduk dunia, termasuk penduduk Indonesia. Bangsa Indonesia telah menjadi Bangsa yang terbesar mengkonsumsi beras di dunia yaitu 105 Kg/kapita/tahun [1]. Tingginya konsumsi beras tersebut menurut pemerintah untuk selalu mengembangkan varietas padi yang lebih unggul dengan produktivitas tinggi. Konsumsi beras yang tinggi memicu terjadinya perdagangan bebas pada produk beras di Indonesia, sehingga pemerintah menerbitkan standar mutu beras giling agar beras yang diperdagangkan memenuhi standar. SNI beras giling berisi syarat mutu beras giling dengan lima tingkatan mutu yakni : mutu I, II, III, IV, dan V [2].

Di Indonesia terdapat sekitar 18 juta petani padi dan menyumbang 66% terhadap Produk Domestik Bruto (PDB) tanaman pangan. Selain itu usaha tani padi telah memberikan kesempatan kerja dan pendapatan bagi lebih 162 dari 21 juta rumah tangga dengan sumbangan pendapatan 25-35%. Oleh sebab itu, beras tetap menjadi komunitas straregis dalam perekonomian dan ketahanan pangan nasional, sehingga menjadi basis utama dalam revitalisasi pertanian kedepan [3]. Namun pemenuhan kebutuhan beras harus diiringi dengan peningkatan mutunya. Mutu beras secara umum dipengaruhi oleh empat faktor utama yaitu:

1) Sifat ginetik
2) Lingkungan dan kegiatan pra panen
3) Perlakuan pemanenan
4) Perlakuan pasca panen

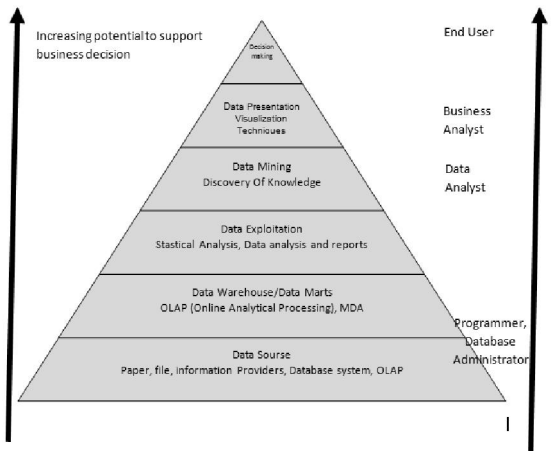
Di Indonesia mutu beras lebih dikenal berdasarkan cara pengolahan, seperti beras tumbuk atau beras giling, berdasarkan derajat sosoh seperti beras slip, berdasarkan asal daerah seperti beras Cianjur, dan

berdasarkan jenis atau kelompok varietas seperti beras IR [4].

Kualitas beras dapat ditentukan dengan berbagai macam metode, suatu teknologi yang dapat digunakan untuk mewujudkannya adalah *data mining. Data mining* adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik, metode atau algoritma dalam *data mining* sangat bervariasi. Salah satu metode yang umum digunakan adalah *decision tree. Decision tree* adalah struktur *flowchart* yang mempunyai *tree* (pohon), dimana setiap simpul internal menandakan suatu tes pada atribut, setiap cabang merepresentasikan hasil tes, dan simpul daun mempresentasikan kelas atau distribusi kelas. Alur pada *decision tree* ditelusuri dari simpul akar kesimpul daun yang memegang prediksi kelas untuk contoh tersebut.

## 2. Landasan Teori

Seberapa besar data dapat dimanfaatkan untuk mendukung proses pengambilan keputusan bisnis dalam sebuah perusahaan sangat ditentukan oleh teknologi pengolahan data yang digunakan seperti OLAP (*Online Analytical Processing)*, *data Warehause, dan data mining*. Perbedaan antara data mining dengan data warehouse dan OLAP secara singkat dapat dijawab dengan Gambar 1 di bawah ini:




Gambar 1. Hubungan Teknologi Basis data dan Data mining

Teknologi *data warehouse* digunakan untuk melakukan *OLAP*, sedangkan *data mining* digunakan untuk melakukan *information discovery* yang informasinya lebih ditujukan untuk seorang *Data Analyst* dan *Business Analys*. Dalam prakteknya, *data mining* juga dapat mengambil data dari *data warehouse*. Hanya saja aplikasi dari *data mining* lebih khusus dan spesifik dibandingkan *OLAP* mengingat basis data bukan satu-satunya bidang ilmu yang mempengaruhi *data mining.*

*Data mining* adalah suatu algoritma di dalam menggali informasi berharga yang terpendam atau tersembunyi pada suatu koleksi data (*database*) yang sangat besar sehingga ditemukan suatu pola yang menarik yang sebelumnya tidak diketahui. Analisa *data mining* berjalan pada data yang cenderung terus membesar dan teknik terbaik yang digunakan kemudian berorientasi kepada data berukuran sangat besar untuk mendapatkan kesimpulan dan keputusan paling layak. *Data mining* memiliki beberapa sebutan atau nama lain yaitu: *Knowledge discovery (mining) in databases* (KDD), ekstraksi pengetahuan *(knowledge extraction),* analisa data/pola, kecerdasan bisnis (*business intelligence*), dll.

Menurut Fayyad dalam bukunya berjudul "*Advances in Knowledge Discovery and Data Mining*", tahapan proses dalam *data mining* secara garis besar dimulai dari data sumber dan berakhir dengan adanya informasi yang dihasilkan dari beberapa tahapan [5], yaitu:

1) Seleksi Data
   Pemilihan (seleksi) data baru dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalian informasi dalam KDD dimulai. Data hasil seleksi yang akan digunakan untuk proses *data mining*, disimpan dalam suatu berkas, terpisah dari basis data operasional.
2) Pembersihan data (*Cleaning*)
   Sebelum proses *data mining* dapat dilaksanakan, perlu dilakukan proses pembersihan pada data yang menjadi fokus KDD. Proses pembersihan

mencakup antara lain membuang *duplikasi data*, memeriksa data yang *inkonsisten*, dan memperbaiki kesalahan pada data, seperti kesalahan cetak (*tipografi*).

3) Transformasi
Pada tahap transformasi data diubah ke dalam bentuk yang sesuai untuk di *mining*. Beberapa teknik *data mining* membutuhkan format data yang khusus sebelum bisa diaplikasikan. Sebagai contoh, beberapa teknik standar seperti analisis asosiasi dan klastering hanya bisa menerima input data kategorikal. Disini juga dilakukan pemilihan data yang diperlukan oleh teknik *data mining* yang dipakai.

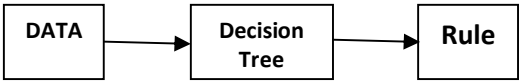
4) *Data mining*
*Data mining* adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik, metode, atau *algoritma* dalam *data mining* sangat bervariasi. Pemilihan metode atau algoritma yang tepat sangat bergantung pada tujuan dan proses KDD secara keseluruhan.

5) Interpretasi/Evaluasi
Pola informasi yang dihasilkan dari proses *data mining* perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak yang berkepentingan. Tahap ini merupakan bagian dari proses KDD yang disebut dengan *interpretation*. Tahap ini mencakup pemeriksaan apakah pola atau informasi yang ditemukan bertentangan dengan fakta atau hipotesa yang ada sebelumnya.

Pada pembahasan disini akan digunakan istilah *pola* dan *model*. Pola dapat diartikan sebagai instansiasi dari model, sebagai contoh $f(x) = 3x^2 + X$ adalah pola model $f(x) = ax^2 + bx$. *Data mining* melakukan "*pengepasan"* atau bisa dikatakan pencocokan model ke atau menentukan pola dari data yang diobservasi. Kebanyakan metodologi *data mining* didasarkan pada konsep mesin belajar, pengenalan atau pencocokan pola dan statistik: klasifikasi, pengelompokan *(clustering)*, pemodelan grafis dan yang lainnya.

**Pohon Keputusan *(Decision tree)***

Pohon keputusan adalah pohon yang ada dalam analisis pemecahan masalah, pemetaan mengenai alternatif-alternatif pemecahan masalah yang dapat diambil dari masalah [6]. Pohon Keputusan dapat juga dikatakan salah satu metode klasifikasi yang paling popular karena mudah untuk diinterpretasi oleh manusia. Konsep dasar *Decision Tree* adalah mengubah data menjadi pohon keputusan dan aturan-turan keputusan (*rule*).

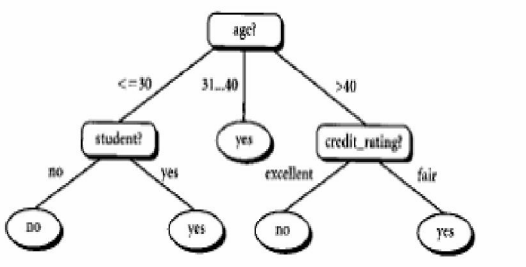



Gambar 2. Konsep *Decision Tree*

*Decision tree* sesuai digunakan untuk kasus-kasus dimana outputnya bernilai diskrit. Manfaat utama dari penggunaan pohon keputusan adalah kemampuannya untuk mem-*break down* proses pengambilan keputusan yang kompleks menjadi lebih simpel sehingga pengambil keputusan akan lebih menginterpretasikan solusi dari permasalahan. Pohon keputusan memadukan antara eksplorasi data dan pemodelan, sehingga sangat bagus sebagai langkah awal dalam proses pemodelan bahkan ketika dijadikan sebagai model akhir dari beberapa teknik lain. Pada umumnya beberapa ciri kasus berikut cocok untuk diterapkan *Decision Tree*:

1) Data/*example* dinyatakan dengan pasangan atribut dan nilainya. Misalnya atribut satu *example* adalah temperatur dan nilainya adalah dingin. Biasanya untuk satu *example* nilai dari satu atribut tidak terlalu banyak jenisnya. Dalam contoh atribut warna ada beberapa nilai yang mungkin yaitu hijau, kuning, merah. Sedang dalam atribut temperatur, nilainya bisa dingin, sedang atau panas. Tetapi untuk beberapa kasus bisa saja nilai temperatur berupa nilai numerik.

2) Label/*output* data biasanya bernilai diskrit. *Output* ini bisa bernilai ya atau tidak, sakit atau tidak sakit, diterima atau ditolak. Dalam beberapa kasus mungkin saja outputnya tidak hanya dua kelas. Tetapi penerapan *Decision Tree* lebih banyak kasus binari.
3) Data mempunyai *missing value*. Misalkan untuk beberapa *example*, nilai dari suatu atributnya tidak diketahui. Dalam keadaan seperti ini *Decision Tree* masih mampu memberi solusi yang baik. Membangun *tree* dimulai dengan data pada simpul akar (*root node*) kemudian pilih sebuah *atribut* dan formulasikan sebuah *logical test* pada *atribut* tersebut lakukan percabangan pada setiap hasil dari *test,* dan terus bergerak ke subset ke contoh yang memenuhi hasil dari simpul anak cabang (*internal node*) yang sesuai lakukan proses rekursif pada setiap simpul anak cabang. Ulangi hingga dahan-dahan dari *tree* memiliki contoh dari satu kelas tertentu. Contoh dari sebuah *decision tree* (Gambar 3).




Gambar 3. Model Pohon Keputusan [10]

### *Algoritma Klasifikasi Data mining*

Klasifikasi [7] adalah proses penemuan model (atau fungsi) yang menggambarkan dan membedakan kelas data atau konsep yang bertujuan agar bisa digunakan untuk memprediksi kelas dari obyek yang label kelasnya tidak diketahui. Klasifikasi data terdiri dari 2 langkah proses. Pertama adalah *learning* (*fase training*), dimana algoritma klasifikasi dibuat untuk menganalisa data *training* lalu direpresentasikan dalam bentuk *rule* klasifikasi. Proses kedua adalah klasifikasi, dimana data tes digunakan untuk memperkirakan akurasi dari *rule* klasifikasi [7].

Proses klasifikasi didasarkan pada empat komponen [8]:

*a. Kelas*
*Variabel dependen* yang berupa kategorikal yang merepresentasikan "*label"* yang terdapat pada obyek. 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.*

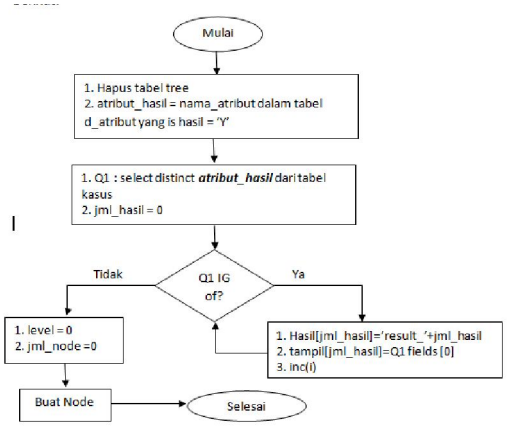
*d. Testing dataset*
Berisi data baru yang akan diklasifikasikan oleh model yang telah dibuat dan akurasi klasifikasi dievaluasi.

### Algoritma ID3

Algoritma ID3 atau *Iterative Dichotomiser 3* (ID3) merupakan sebuah metode yang digunakan untuk membangkitkan pohon keputusan. Algoritma pada metode ini menggunakan konsep dari *entropi informasi.* Secara ringkas, cara kerja Algoritma ID3 dapat digambarkan sebagai berikut [5]. Pemilihan atribut dengan menggunakan *Information Gain*

1. Pilih atribut dimana nilai *information gainnya* terbesar
2. Buat simpul yang berisi atribut tersebut
3. Proses perhitungan *information gain* akan terus dilaksanakan sampai semua data telah termasuk dalam kelas yang sama. Atribut yang telah dipilih tidak diikutkan lagi dalam perhitungan nilai *information gain.*

Dalam *flowchart* cara kerja Algoritma ID3 dapat pula digambarkan sebagai berikut:




Gambar 4. Algoritma Inisialisasi Pembentukan Node [5]

Pemilihan atribut pada ID3 dilakukan dengan properti statistik, yang disebut dengan *information gain. Gain* mengukur seberapa baik suatu atribut memisahkan *training example* ke dalam kelas target. Atribut dengan informasi tertinggi akan dipilih. Dengan tujuan untuk mendefinisikan *gain*, pertama-tama digunakanlah ide dari teori informasi yang disebut *entropi. Entropi* mengukur jumlah dari informasi yang ada pada atribut dengan rumus :

$$Entropy\ (S) = -P_+ \log_2 P_+ - P_- \log_2 P_-$$

Berdasarkan rumus di atas, $P_+$ adalah probabilitas sampel S yang mempunyai *class* positif. $P_+$ dihitung dengan membagi jumlah *sampel* positif ($S_+$) dengan jumlah *sampel* keseluruhan (S) sehingga $P_+ = \frac{S_+}{S}$.

$P_-$ adalah probabilitas *sampel* S yang mempunyai *class* negatif. $P_-$ dihitung dengan membagi jumlah *sampel* negatif ($S_-$) dengan jumlah *sampel* keseluruhan (S) sehingga $P_- = \frac{S_-}{S}$. Bagian daun dari sebuah *decision tree,* idealnya hanya terdiri dari data *e-mail Spam* dan *e-mail non-Spam.* Dengan kata lain bagian daun adalah sampel murni, jadi ketika membagi sebuah sampel, sisa sampel harus lebih murni dibandingkan simpul sebelumnya. Oleh karena itu nilai *entropy* harus dikurangi. Pada *algoritma ID3* pengurangan *entropy* disebut dengan *informasi gain.*

Pembagian *sampel S* terhadap atribut A dapat dihitung *information gain* dengan rumus:

$$Gain(S, A) = Entropy(S) - \sum_{v \in nilai(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

*Value A* adalah semua nilai yang mungkin dari atribut *A*, dan $S_v$ adalah *subset* dari *S* dimana *A* mempunyai nilai c. bagian pertama pada rumus adalah *entropy* total *S* dan bagian kedua adalah *entropy* sesudah dilakukan pemisahan data berdasarkan atribut *A*.

***Gain Ratio***

Untuk menghitung *gain ratio* diperlukan suatu *term Split Information. Split Information* dapat dihitung dengan formula sebagai berikut:

$$SplitInformation(S, A) = -\sum_{i=1}^{c} \frac{S_i}{S} \log_2 \frac{S_i}{S}$$

Dimana $S_1$ sampai $S_c$ adalah c subset yang dihasilkan dari pemecahan S dengan menggunakan atribut A yang mempunyai sebanyak c nilai.

Selanjutnya *gain ratio* dihitung dengan cara:

$$Gainratio(S, A) = \frac{Gain(S, A)}{SplitInformation(S, A)}$$

Sebuah cara yang jelas untuk meniadakan bias atau *"greediness" Informasi gain* adalah untuk memperhitungkan jumlah nilai dari atribut. Pendekatan ini yang dapat digunakan. Sebuah perhitungan baru ditingkatkan untuk atribut A melalui data S adalah:

$$SplitInfomation(A) = \sum_{i \in A} -\log_2 \frac{P_i}{N}$$

Persamaan di atas mengukur isi informasi untuk atribut A dengan melihat proporsi masing- masing $P_i$ contoh yang mengambil nilai i untuk atribut.

***Gini Index***

Jika kelas obyek dinyatakan dengan k, k-1, 2, …, C, dimana C adalah jumlah kelas untuk variabel/output dependent y, *index gini* untuk suatu cabang atau kotak A dihitung sebagai berikut:

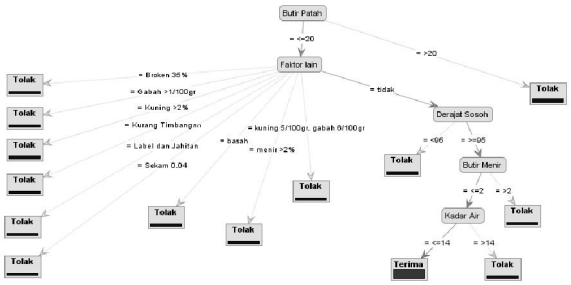
$$IG(A) = 1 - \sum_{k=1}^{c} p_k^2$$

Dimana $p_k$ adalah ratio observasi dalam kotak A yang masuk dalam kelas *k*. jika *IG(A)* = 0 berarti semua data dalam kotak A berasal dari kelas yang sama. Nilai *IG(A)* mencapai maksimum jika dalam kelas A proporsi data dari masing-masing kelas yang ada mencapai nilai yang sama.

## 3. Hasil Penelitian

***Information Gain* pada data *Training***

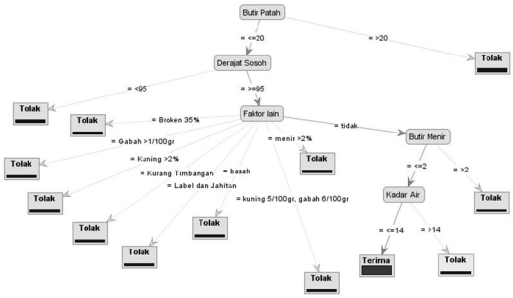
Gambar 5 adalah pohon keputusan akhir yang dihasilkan dari perhitungan *entropy* dan *gain* untuk seluruh atribut. Terlihat bahwa atribut butir patah menjadi simpul akar karena butir patah mempunyai nilai *gain* yang paling besar. Dari simpul akar splittingnya menjadi dua simpul sesuai dengan nilai yang dimilikinya. Kemudian pada gambar 5 juga terlihat, untuk cabang paling kanan, simpul 1.1 adalah faktor lain, karena atribut tersebut mempunyai nilai *gain* tertinggi, dibawah simpul 1.1, yaitu simpul 1.1.1 merupakan atribut derajat sosoh, atribut derajat sosoh merupakan atribut dengan nilai *gain* tertinggi,




Gambar 5. Pohon Keputusan Information Gain data training

***Gain Ratio pada data training***

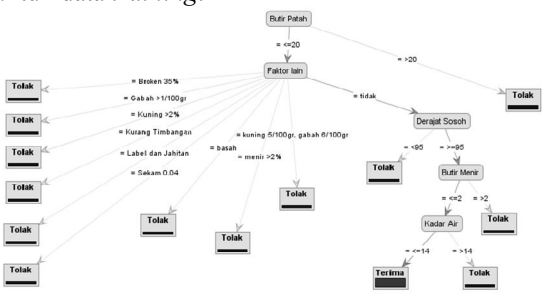
Gambar 6 adalah pohon keputusan akhir yang dihasilkan dari perhitungan *Split Information* dan *gain ratio* untuk seluruh atribut. Terlihat bahwa atribut butir patah menjadi simpul akar butir patah mempunyai nilai *gain ratio* yang paling besar. Dari simpul akar splittingnya menjadi dua split sesuai dengan nilai yang dimilikinya.




Gambar 6. Pohon Keputusan Gain Ratio data training

***Gini Index* pada data *Training***

Gambar 7 adalah pohon keputusan akhir yang dihasilkan dari perhitungan *nilai gini* untuk seluruh atribut. Terlihat bahwa atribut butir patah menjadi simpul akar, karena butir patah mempunyai nilai *gini* terkecil. Dari simpul akar *splittingnya* menjadi dua simpul sesuai dengan nilai yang dimilikinya. Kemudian pada gambar juga terlihat, untuk cabang paling kanan, simpul 1.1 adalah faktor lain, karena atribut tersebut juga mempunyai nilai *gini* terendah, di bawah simpul 1.1, yaitu simpul 1.1.1 merupakan atribut derajat sosoh, atribut derajat sosoh merupakan atribut dengan nilai *gini* terendah, dari hasil akhir yang di dapat untuk data *training*.

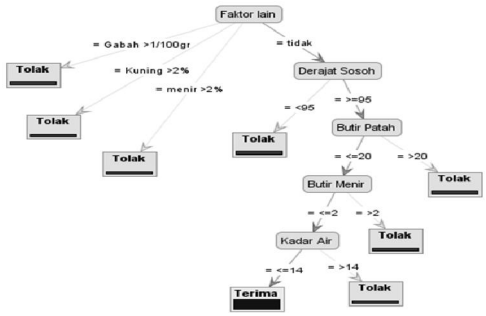



Gambar 7. Pohon Keputusan Gini Index data training

***Information gain pada data testing***

Gambar 8 adalah pohon keputusan akhir yang dihasilkan dari perhitungan *entropy*

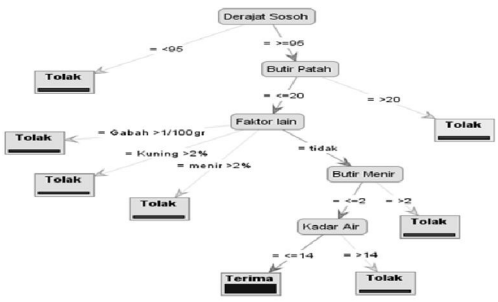
dan *gain* untuk seluruh atribut. Terlihat bahwa atribut faktor lain menjadi simpul akar karena faktor lain mempunyai nilai *gain* yang paling besar. Dari simpul akar splittingnya menjadi empat simpul sesuai dengan nilai yang dimilikinya. Kemudian pada gambar juga terlihat, untuk cabang paling kanan, simpul 1.1 adalah derajat sosoh, karena atribut tersebut mempunyai nilai *gain* tertinggi, di bawah simpul 1.1, yaitu simpul 1.1.1 merupakan atribut butir patah, atribut butir patah merupakan atribut dengan nilai *gain* tertinggi.




Gambar 8. Pohon Keputusan Information Gain data testing

***Gain Ratio pada data testing***

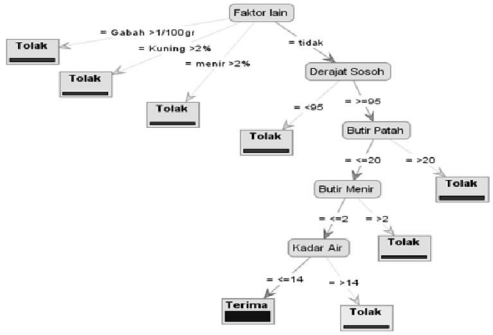
Gambar 9 adalah pohon keputusan akhir yang dihasilkan dari perhitungan *Split Information* dan *gain ratio* untuk seluruh atribut. Terlihat bahwa atribut derajat sosoh menjadi simpul akar karena derajat sosoh mempunyai nilai *gain ratio* yang paling besar. Dari simpul akar splittingnya menjadi dua split sesuai dengan nilai yang dimilikinya.




Gambar 9. Pohon Keputusan Gain Ratio data training

***Gini Index Pada data testing***

Gambar 10 adalah pohon keputusan akhir yang dihasilkan dari perhitungan *gini index* untuk seluruh atribut. Terlihat bahwa atribut faktor lain menjadi simpul akar karena faktor lain memiliki nilai *gini terkecil*. Dari simpul akar splittingnya menjadi empat split sesuai dengan nilai yang dimilikinya.




Gambar 10. Pohon Keputusan Gini Index data testing

## 4. Evaluasi dan Validasi

Untuk membuat model klasifikasi, bisa digunakan banyak metode. Dalam penelitian ini misalkan, metode yang digunakan, yaitu algoritma *ID3* dalam pemilihan *criteria splitting* pada *information gain*, *gain ratio dan gini index,* kemudian dilakukan komparasi ketiganya dan mengukur *criteria splitting* mana yang paling akurat. Metode klasifikasi *ID3* dalam *criteria splitting,* bisa dievaluasi berdasarkan beberapa kriteria seperti tingkat akurasi, kecepatan, kehandalan, skalabilitas, dan interpretabilitas [9].

## 5. Pengujian Model

Model yang telah dibentuk akan diuji tingkat akurasinya dengan memasukan data *testing/validasi* kedalam model. Untuk mengukur keakuratan model dengan baik, data uji seharusnya bukan data yang berasal dari data training [7]. Data uji diambil dari data *testing/validasi*. *Data testing* memiliki 147 sampel yang diambil dari data setelah proses *cleaning* sebesar 20%. Sampel akan diujikan ke dalam data training untuk mendapatkan hasil klasifikasi dari Algoritma ID3. Pada pengujian ini

ditambahkan pemilihan metode penyeleksian *criteria splitting* yaitu *information gain, gain ratio dan gini index* dengan tujuan untuk melihat akurasi dari masing-masing criteria dengan algoritma yang sama yaitu ID3.

### *1). Confusion Matrix*

Tabel 1 adalah perhitungan berdasarkan data *training,* setelah memasukan *data training* dan *data testing* pada Tabel 1 diketahui dari 590 *data training* dimana 344 diklasifikasikan *terima* sesuai dengan prediksi yang dilakukan dengan *criteria information gain*, lalu 11 data diprediksi terima tetapi ternyata ditolak, semua data *class ditolak* ternyata tidak ada satupun yang diterima, dan diprediksi 235 data *ditolak* sesuai dengan prediksi.

Tabel 1. Model *Confusion Matrix* Metode ID3 pada *criteria Information gain*

accuracy: 98.14%

| | true Tolak | true Terima | class precision |
|---|---|---|---|
| pred. Tolak | 235 | 0 | 100.00% |
| pred. Terima | 11 | 344 | 96.90% |
| class recall | 95.53% | 100.00% | |

Tabel 2 adalah *confusion matrix* untuk kriteria *Gain ratio*. Diketahui dari 590 data training, 344 diklasifikasikan diterima sesuai dengan prediksi yang dilakukan dengan metode ID3 pada *criteria Gain ratio*, lalu 10 data diprediksi *diterima* tetapi ternyata *ditolak*, 236 data *class ditolak* diprediksi sesuai, dan tidak ada data diprediksi *ditolak yang dapat diterima*.

Tabel 2. Model *Confusion Matrix* Metode ID3 pada *criteria gain ratio*

accuracy: 98.31%

| | true Tolak | true Terima | class precision |
|---|---|---|---|
| pred. Tolak | 236 | 0 | 100.00% |
| pred. Terima | 10 | 344 | 97.18% |
| class recall | 95.93% | 100.00% | |

Dengan metode ID3 pada *criteria gini index*, menghasilkan kondisi seperti pada Tabel 3, diketahui dari 590 data training, 344 diklasifikasikan diterima sesuai dengan prediksi yang dilakukan dengan metode ID3 pada *criteria giini index*, lalu 11 data diprediksi *diterima* tetapi ternyata *ditolak*, 235 data *class ditolak* diprediksi sesuai, dan tidak ada data diprediksi *di tolak yang dapat diterima*

Tabel 3. Model *Confusion Matrix* Metode ID3 pada *criteria gini index*

accuracy: 98.14%

| | true Tolak | true Terima | class precision |
|---|---|---|---|
| pred. Tolak | 235 | 0 | 100.00% |
| pred. Terima | 11 | 344 | 96.90% |
| class recall | 95.53% | 100.00% | |

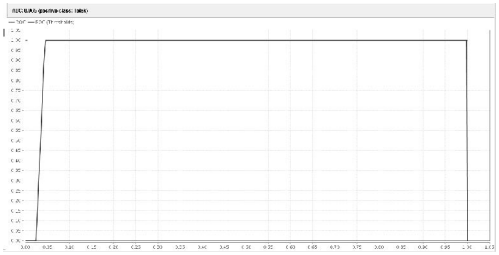
Dari tiga tabel *confusion matrix*, selanjutnya dilakukan perhitungan nilai *accuracy, precision sensitivity*, dan *recall*, dapat dilihat pada tabel 4

Tabel 4. Komparasi nilai *accuracy, precision sensitivity* dan *recall*

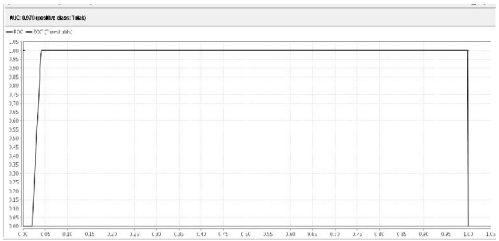
| | *Information Gain* | *Gain Ratio* | *Gini Index* |
|---|---|---|---|
| *Accuracy* | 98.14% | 98.30% | 98.14% |
| *Precision* | 95.53% | 95.93% | 95.53% |
| *Recall* | 100% | 100% | 100% |

### *2). Kurva ROC (Receiver Operating Characteristic)*

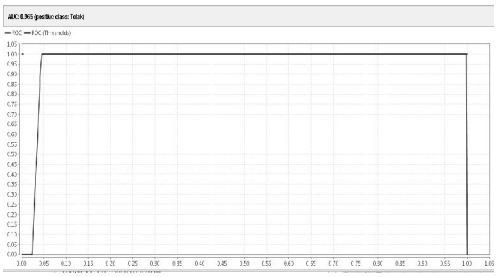
Dari hasil perhitungan divisualisasikan dengan menggunakan kurva *ROC* Perbandingan ketiga metode komparasi bisa dilihat pada Gambar 11, 12, dan 13 yang merupakan kurva ROC untuk *criteria Information gain*, *Gain ratio* dan *Gini Index*



Gambar 11. Kurva *ROC* dengan *criteria information Gain*



Gambar 12. Kurva *ROC* dengan *criteria information Gain*



Gambar 13. Kurva *ROC* dengan *criteria information Gain*

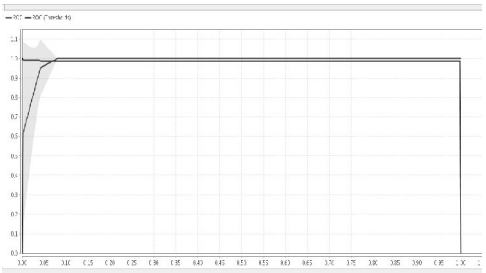
Perbandingan hasil perhitungan nilai AUC untuk kriteria *Information gain, Gain ratio dan Gini index* dapat dilihat pada Tabel 5.

Tabel 5. Komparasi Nilai *AUC*

| | ***Information gain*** | ***Gain ratio*** | ***Gini index*** |
|---|---|---|---|
| AUC | 0.965 | 0.970 | 0.965 |

Seperti yang telah dijelaskan sebelumnya, penulis membandingkan tiga criteria penyeleksian atribut yaitu *information gain, gain ratio* dan *gini index.* Seperti terlihat pada gambar 14 dengan menggunakan Rapid Miner hasil dari ketiga criteria tersebut menghasilkan grafik yang berbeda, namun demikian nilai akurasi hasil *confusion matrix* tetap sama.

Gambar 14. Kurva *ROC* dengan membandingkan tiga criteria; *Informattion_gain, Gain_ratio dan Gini_index* tanpa memasukkan data validasi/testing



Gambar 15. Kurva *ROC* dengan membandingkan tiga criteria; *Informattion_gain, Gain_ratio dan Gini_index* dengan memasukkan data validasi/testing

## 6. Analisis Hasil Komparasi

Model yang dihasilkan dengan kriteria *Information_gain, Gain_ratio* dan *Gini_index* diuji menggunakan metode *Confusion Matrix*, terlihat perbandingan nilai *accuracy, precision, sensitivity*, dan *recall* pada Tabel 4, untuk *criteria Gain ratio* memiliki nilai *accuracy, precision, sensitivity*, dan *recall* yang paling tinggi, diikuti dengan *criteria information gain* dan *gini index,* keduanya yang memiliki nilai yang sama.

Tabel 6. Komparasi Nilai A*ccuracy* dan *AUC*

| | ***Information gain*** | ***Gain ratio*** | ***Gini index*** |
|---|---|---|---|
| *Accuracy* | 98.14% | 98.30% | 98.14% |
| AUC | 0.965 | 0.970 | 0.965 |

Tabel 6 membandingkan *accuracy* dan *AUC* dari tiap *criteria*. Terlihat bahwa nilai *accuracy* criteria *gain ratio* paling tinggi begitu pula dengan nilai AUC-nya. Untuk *criteria Information gain dan Gini index* juga menunjukan nilai yang sesuai. Untuk klasifikasi *data mining*, nilai *AUC* dapat dibagi menjadi beberapa kelompok [8].

a. 0.90-1.00 = klasifikasi sangat baik
b. 0.80-0.90 = klasifikasi baik
c. 0.70-0.80 = klasifikasi cukup
d. 0.60-0.70 = klasifikasi buruk
e. 0.50-0.60 = klasifikasi salah

Pada pengelompokan nilai klasifikasi di atas dan berdasarkan Tabel 7 maka dapat disimpukan bahwa pengukuran kinerja ketiga *criteria Information gain, gain ratio dan gini index* pada algoritma ID3, termasuk klasifikasi sangat baik karena memiliki nilai *accuracy* antara 0.90-1.00.

## 7. Kesimpulan

Dalam penelitian ini dilakukan pembuatan model menggunakan algoritma ID3 menggunakan data kualitas beras yang diterima maupun ditolak pada Perum Bulog Divre Lampung. Model yang dihasilkan, di uji keakuratannya dengan cara mengambil data uji/validasi sebesar 20% dari data *cleaning* yang telah didapatkan dengan menggunakan *tools* Rapid Miner secara random dan sisanya 80% sebagai data training, lalu criteria splitting *Information gain, gain ratio* dan *gini index* dikomparasi untuk mengetahui kriteria mana yang paling baik dalam penentuan kualitas beras dengan memasukkan data uji kedalam data training. Untuk mengukur kinerja ketiga criteria tersebut digunakan metode pengujian *Confusion Matrix* dan Kurva *ROC*, diketahui bahwa dalam algoritma ID3 pada *splitting gain ratio* memiliki nilai *accuracy* dan AUC paling tinggi, diikuti oleh kedua *splitting* lainnya yang bernilai sama.

Sehingga dapat disimpulkan bahwa, metode ID3 pada *splitting gain ratio* dapat menghasilkan *decision tree* yang akurat untuk menentukan kualitas beras dan juga merupakan metode yang sangat baik dalam pengklasifikasian data dengan kasus data biner, dengan demikian algoritma ID3 pada *splitting gain ratio* juga dapat memberikan pemecahan untuk permasalahan penentuan kualitas beras yang dapat diterima Perum Bulog Divre Lampung.

## Daftar Pustaka


[1] Bayu Krisnamurthi, "*One Day No Rice"*, 2010, http://www.antaranews.com/berita/1287813032/one-day-no-rice-strategi-angkat-pangan-lokal (Diakses 9 Januari 2012).

[2] Badan Standarisasi Nasional, 1999, http://www.docstoc.com/docs/104996576/JENIS-MUTU-BERAS (Diakses 7 Januari 2012)

[3] Badan Litbang Pertanian, 2005, "*Prospek dan Arah Pengembangan Agribisnis Padi*", 2005, Departemen Pertanian, 49 hal.

[4] Damardjati D.S dan E. Y. Pirwani, 1991, Padi Buku 3. Penyunting Edi Soenarjo, D.S. dan Mahyudin Syam. Pusat Penelitian dan Pengembangan Tanaman Pangan Bogor.

[5] Kusrini & Luthfi, E. T. 2009. "*Algoritma Data Mining.* Yogyakarta: Andi Publishing.

[6] Feri, Sulianta, & Dominikus, Juju, 2010, "*Data Mining : Meramalkan Bisnis Perusahaan",* PT., Elex Media Komputindo, Jakarta.

[7] Han, J., & Kamber, M. 2006. "*Data Mining: Consepts, Models, and Techniques".* Fransisco: Morgan kauffman.

[8] Gorunescu, Florin, 2011. *Data Mining: Concepts, Models, and Techniques*. Verlag Berlin Heidelberg: Springer

[9] Carlo Vercellis, 2009, Business Intelligence: "*Data Mining and Optimization for Decision making"*.

[10] Pramudiono, I., Pengantar Data Minig : Menambang Permata Pengetahuan di gunung Data, http://www.ilmukomputer.com. (Diakses 17 Januari 2011)