
**IMPLEMENTASI ALGORITMA C5.0 UNTUK DIAGNOSA
PENYAKIT LIMFADENOPATI**

Oleh:

Florida Nirma Sanny Damanik¹⁾, Kristian Telaumbanua²⁾, Denny Setya Hermawan³⁾

¹²³Program Studi Teknik Informatika – Universitas Mikroskil

Abstrak

Limfadenopati adalah kelainan dari Kelenjar Getah Bening dalam bentuk ukuran, jumlah, konsistensi, yang disebabkan penambahan sel-sel pertahanan tubuh yang berasal dari Kelenjar Getah Bening itu sendiri. Pada penelitian ini Algoritma C5.0 dengan model klasifikasi tree digunakan untuk diagnosa penyakit Limfadenopati. Penerapan Algoritma C5.0 ini menggunakan 21 data pasien dari Rumah Sakit Umum Haji Medan, yang terdiri dari 17 data latih dan 4 data uji. Pengguna dapat melakukan diagnosa kapan saja. Pengujian dari algoritma C5.0 dengan 4 data testing untuk menguji tingkat sistem klasifikasi diagnosa penyakit limfadenopati mendapatkan hasil akurasi 75%, presisi 50%, recall 100%.

Kata kunci : Limfadenopati, Decision Tree, C5.0

PENDAHULUAN

Latar belakang Penelitian

Algoritma C5.0 memiliki kemampuan untuk mengklasifikasikan sebagai pohon keputusan atau seperangkat aturan. Kekuatan algoritma ini dalam menangani nilai, serta lebih sedikit waktu diperlukan untuk mempelajarinya. C5.0 adalah classifier yang mengklasifikasikan data dalam waktu yang lebih singkat dibandingkan dengan classifier lain. Untuk menghasilkan pohon keputusan penggunaan memori minimum dan juga meningkatkan akurasi. C5.0 merupakan terusan dari algoritma C4.5, dimana C5.0 memperbaiki kelemahan dari C4.5 dari segi hasil klasifikasi, kesalahan klasifikasi, prediksi akurasi, waktu, tingkat kesalahan, dan pemakaian memori. Dengan munculnya Algoritma C5.0 ini, tingkat akurasi yang dimiliki lebih tinggi, lebih cepat untuk mengambil keputusan,

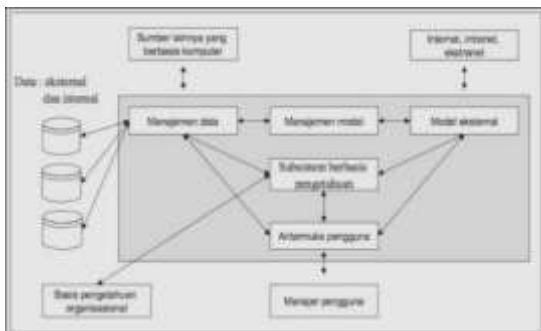
dan penggunaan memori yang jauh lebih rendah dari algoritma sebelumnya [1]. Atribut uji pada setiap node didalam tree dipilih dengan cara memproses ukuran information gain. Pemilihan parent node selanjutnya adalah dengan cara memilih atribut yang memiliki nilai information gain paling tinggi. Algoritma ini membentuk pohon keputusan dengan cara pembagian dan menguasai sampel secara rekursif dari atas sampai ke bawah. Algoritma ini diawali dengan menjadikan semua data menjadi akar dari pohon keputusan, sedangkan pembagi untuk sampel didapat dari atribut yang telah dipilih. Klasifikasi pada Algoritma C5.0 disajikan dalam bentuk pohon keputusan.

KAJIAN PUSTAKA

Sistem Pendukung Keputusan

Sistem Pendukung Keputusan (SPK) atau *Decision Support System* (DSS) adalah

suatu sistem yang memiliki kemampuan pemecahan masalah atau pun kemampuan pengkomunikasian untuk masalah yang memiliki kondisi semi struktur dan tak terstruktur. Sistem pendukung keputusan digunakan membantu pengambilan keputusan dalam situasi semi terstruktur dan situasi yang tidak terstruktur, dimana tak ada seorangpun yang tahu secara pasti bagaimana keputusan seharusnya dapat dibuat. [2]



Gambar 1. Arsitektur dari Sistem Pendukung Keputusan

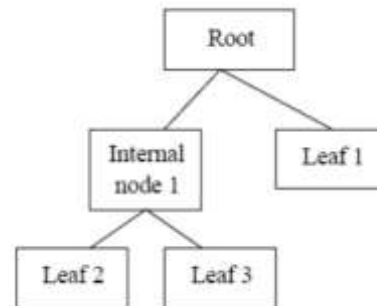
Data Mining

Data mining ilmu yang menggambarkan area-area kerja seperti teknologi basis data, pembelajaran mesin, statistik, pengenalan pola, pengambilan informasi, jaringan saraf tiruan, sistem berbasis pengetahuan, kecerdasan buatan, komputasi kinerja-tinggi, dan visualisasi data. Data mining mampu mengubah data yang luas dan jumlah yang besar menjadi informasi yang berguna dalam ilmu pengetahuan. Data mining adalah proses pengumpulan, pemakaian data historis yang menemukan keteraturan, pola dan hubungan dalam set data berukuran besar. Proses pencarian informasi yang

tidak diketahui sebelumnya dari sekumpulan data besar [3]

Decision Tree

Pohon keputusan merupakan sebuah struktur yang dapat digunakan untuk membagi kumpulan data yang besar menjadi himpunan-himpunan record yang lebih kecil dengan menerapkan serangkaian aturan keputusan, dengan masing-masing rangkaian pembagian, anggota himpunan hasil menjadi mirip satu dengan yang lain. Pohon keputusan merupakan salah satu metode klasifikasi dengan mengubah data menjadi pohon keputusan dan aturan-aturan keputusan (*rule*), yang dapat digunakan untuk memprediksi atau mengklarifikasi suatu kejadian.



Gambar 2. Pohon Keputusan

Pada pohon keputusan terdapat 3 jenis *Node* yaitu: [4]

- a. *Root Node*, merupakan *node* paling atas didalam sebuah pohon keputusan. Pada *root node* tidak terdapat inputan dan juga bisa tidak memiliki *output* atau pun juga memiliki *output* yang jumlahnya lebih dari satu.
- b. *Internal Node*, merupakan *node* percabangan didalam sebuah pohon keputusan. Pada *node* ini hanya

memiliki satu inputan dan memiliki *output* minimal dua.

- c. *Leaf Node*, merupakan *node* paling akhir didalam sebuah pohon keputusan. Pada *leaf node* hanya terdapat satu inputan dan tidak mempunyai *output* sama sekali.

Algoritma C5.0

Algoritma C5.0 adalah salah satu algoritma klasifikasi *Data mining* yang khususnya diterapkan pada teknik pohon keputusan (*decision tree*). C5.0 adalah penyempurnaan dari algoritma sebelumnya yang dibentuk oleh Ross Quinlan pada tahun 1987, yaitu ID3 dan C4.5. [5]. Strategi pengembangan *decision tree* dengan menggunakan algoritma C5.0 adalah sebagai berikut:

1. *Node* tunggal menggambarkan *tree* yang merepresentasikan sebuah *training set*.
2. *Node* berubah menjadi *leaf* dan dilabeli dengan kelas sampel, jika seluruh sampel berisi kelas yang sama.
3. Jika sampel seluruhnya tidak berisi kelas yang sama, maka algoritma yang menggunakan ukuran berbasis entropi (*information gain*) akan memilih atribut yang akan memisahkan *record* kedalam kelas-kelas individual. Atribut yang dipilih tersebut diubah menjadi atribut tes atau keputusan pada *node* tersebut.
4. Lalu cabang kemudian dikembangkan untuk tiap-tiap nilai yang diketahui dari atributtes, dan sampel dipartisi berdasarkan cabang tersebut.
5. Algoritma C5.0 menggunakan proses yang sama seperti sebelumnya dan dilakukan secara rekursif yang

kemudian akan membentuk sebuah pohon keputusan.

6. Rekursif berakhir hanya jika salah satu dari kondisi-kondisi berikut ini terpenuhi:
 - a. Seluruh sampel */record* pada *node* tertentu memiliki kelas yang sama.
 - b. Tidak ada atribut yang tersisa pada sampel/*record* yang dapat dipartisi lebih lanjut. Dalam kasus ini suara mayoritas digunakan. *Node* tersebut diubah menjadi *leaf node* dan dilabeli dengan kelas yang menjadi mayoritas dalam sampel/*record* yang ada.
 - c. Tidak ada sampel/*record* untuk cabang atributtes. *Leaf* terbentuk dengan mayoritas kelas sebagai label sampel/*record* tersebut.

Langkah kerja pembuatan *tree* pada algoritma C5.0 mirip dengan pembuatan *tree* pada algoritma C4.5. Kemiripan tersebut meliputi perhitungan *entropy* dan *gain*. Algoritma C5.0 berhenti sampai perhitungan *gain*, dengan menggunakan *gain* dan *entropy* yang telah ada. Adapun rumus untuk mencari nilai *entropy* adalah sebagai berikut:

Entropy (*S*) = –dengan: *S* = Himpunan Kasus; *m* = Jumlah Sampel; *Pi* = Proporsi kelas.

Untuk mendapatkan nilai atribut *A* tersebut maka digunakan rumus sebagai berikut [6] :

$$E(A) = \sum_{j=i}^y \frac{|S_{ij}|}{|S|} * Entropy(S_{ij})$$

dengan S_i merupakan himpunan kasus pada kategori ke- i , A adalah variabel yang digunakan, m adalah jumlah kategori pada variabel A , $|S_i|$ merupakan jumlah kasus pada kategori ke- i , dan $|S|$ merupakan jumlah kasus dalam S . Selanjutnya untuk mencari nilai *gain* digunakan persamaan berikut:

$$GAIN(S, A) = Entropy(S) - E(A)$$

Dengan *Gain* (S, A) merupakan nilai *gain* dari suatu variabel dan $\sum_{j=i}^y Entropy(S_j)$ merupakan jumlah nilai *entropy* dalam suatu variabel.

2.5. Confusion Matrix

Metode *confusion matrix* terdiri dari informasi mengenai klasifikasi yang diprediksi dengan benar oleh system klasifikasi. Evaluasi menggunakan metode *confusion matrix* menghasilkan nilai akurasi, presisi, dan *recall*. Akurasi adalah hasil perbandingan data yang terklasifikasi benar secara keseluruhan data. Presisi menunjukkan berapa banyak jumlah data kategori positif dan diklasifikasi secara benar, lalu dibagi dengan jumlah data yang diklasifikasi positif.

Sedangkan *recall* menggambarkan tingkat persentase data yang tergolong positif dan terklasifikasi dengan benar oleh sistem. Metode ini menggunakan *table matrix* seperti dibawah ini [7] :

Tabel 1. Model Confusion Matrix

		Observed	
		True	False
Predicted Class	True	True Positive (TP)	False Positive (FP)
	False	False Negative (FN)	True Negative (TN)

Untuk mendapatkan nilai akurasi maka digunakan rumus sebagai berikut :

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

Untuk mendapatkan nilai presisi maka digunakan rumus sebagai berikut:

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

Untuk mendapatkan nilai *recall* maka digunakan rumus sebagai berikut:

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

Keterangan:

- a. *True Positive* yaitu total data positif yang terklarifikasi secara benar oleh sistem.
- b. *True Negative* yaitu total data negatif yang terklarifikasi secara benar oleh sistem.
- c. *False Negative* yaitu total data negative tetapi terklarifikasi salah oleh sistem.
- d. *False Positive* yaitu total data positif tetapi terklarifikasi salah oleh sistem.

METODE PENELITIAN

Pengumpulan Data

Data-data yang digunakan pada penelitian ini yaitu data penyebab, gejala, dan lokasi penyakit *Limfadenopati*. Data ini didapatkan dari Dokter spesialis penyakit dalam dengan menanyakan langsung data tersebut melalui sesi tanya jawab / wawancara. Sejumlah 21 data yang dibagi menjadi data training dan data testing dengan perbandingan 80% dan 20%, sehingga diperoleh 17 data training dan 4 data testing. Data testing digunakan untuk mengukur tingkat akurasi dari hasil klasifikasi tersebut.

Tabel 2. *Dataset* Gejala Covid-19 (diwakilkan dengan 15 data)

	Batuk	Demam	Pusing	Menggigil	Benjolan	Tidak Selera makan
1	Tidak	Ya	ya	ya	Ya	Tidak
2	Ya	Ya	ya	ya	Tidak	Tidak
3	Ya	Tidak	tidak	tidak	Ya	Tidak
4	Ya	Tidak	tidak	tidak	Ya	Ya
5	Ya	Ya	tidak	ya	Tidak	Tidak
6	Tidak	Tidak	tidak	tidak	Ya	Tidak
7	Tidak	Ya	Ya	tidak	Ya	Ya
8	Ya	Tidak	Ya	ya	Ya	Tidak
9	Ya	Ya	Ya	tidak	Ya	Tidak
10	Tidak	Tidak	Ya	ya	Tidak	Ya
11	Ya	Ya	Tidak	tidak	Ya	Tidak
12	Tidak	Tidak	Ya	tidak	Ya	Tidak
13	Ya	Ya	Tidak	tidak	Ya	Tidak
14	Ya	Ya	Ya	ya	Ya	Tidak
15	Tidak	Tidak	Ya	tidak	Ya	Tidak

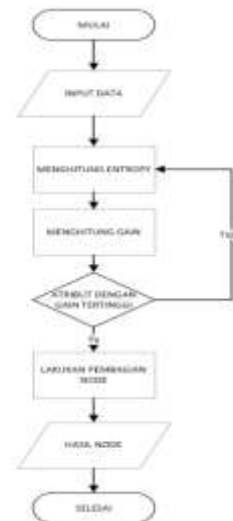
Proses Algoritma C5.0

Adapun tahapan dalam proses algoritma ini meliputi membaca *data set* dari *database*, kemudian proses algoritma C5.0 untuk melakukan diagnosis awal *Limfadenopati*. *Flowchart* dari analisis proses yaitu sebagai berikut.



Gambar 3. *Flowchart* Analisis Proses

Operasi algoritma C5.0 dilakukan dengan perhitungan menggunakan dataset yang sudah didapatkan. Untuk memudahkan penjelasan, proses algoritma C5.0 digambarkan kedalam *flowchart* sebagai berikut.



Gambar 4. *Flowchart* Algoritma C5.0

Berdasarkan gambar bagan tahapan penelitian di atas, dapat dijelaskan sebagai berikut:

1. Data akan di input kedalam sistem.

2. Data diolah untuk mendapatkan nilai *entropy* dari setiap atribut
3. Hasil dari *entropy* akan dilakukan perhitungan kembali untuk menemukan *gain*
4. Setelah nilai *gain* ditemukan, akan di cari *gain* tertinggi untuk menjadi *node*
5. Jika data masih dapat di hitungakan dilakukan pengulangan sampai data tidak bisa dihitung kembali dan *node* yang sebelumnya sudah didapatkan akan dilakukan pembentukan pohon keputusan.

Perhitungan Entropy

Entropy Total Keputusan

$$\left(-\frac{9}{17} * \log_2\left(\frac{9}{17}\right)\right) + \left(-\frac{8}{17} * \log_2\left(\frac{8}{17}\right)\right) = 0,997503$$

Hasil *entropy* untuk total keputusan adalah = 0,997503

Entropy Batuk (ya)

$$\left(-\frac{6}{11} * \log_2\left(\frac{6}{11}\right)\right) + \left(-\frac{5}{11} * \log_2\left(\frac{5}{11}\right)\right) = 0,994030$$

Hasil *Entropy* untuk gejala batuk (ya) adalah = 0,994030

Entropy Batuk (tidak)

$$\left(-\frac{3}{6} * \log_2\left(\frac{3}{6}\right)\right) + \left(-\frac{3}{6} * \log_2\left(\frac{3}{6}\right)\right) = 1$$

Hasil *entropy* untuk gejala batuk (tidak) adalah = 1

Entropy Demam (ya)

$$\left(-\frac{6}{10} * \log_2\left(\frac{6}{10}\right)\right) + \left(-\frac{4}{10} * \log_2\left(\frac{4}{10}\right)\right) = 0,970951$$

Hasil *entropy* untuk gejala demam (ya) adalah = 0,970951

Entropy Demam (tidak)

$$\left(-\frac{3}{7} * \log_2\left(\frac{3}{7}\right)\right) + \left(-\frac{4}{7} * \log_2\left(\frac{4}{7}\right)\right) = 0,985228$$

Hasil *entropy* untuk gejala demam (tidak) adalah = 0,985228

Entropy Pusing (ya)

$$\left(-\frac{5}{9} * \log_2\left(\frac{5}{9}\right)\right) + \left(-\frac{4}{9} * \log_2\left(\frac{4}{9}\right)\right) = 0,991076$$

Hasil *entropy* untukgejalapusing (ya)adalah = 0,991076

Entropy Pusing (Tidak)

$$\left(-\frac{5}{8} * \log_2\left(\frac{5}{8}\right)\right) + \left(-\frac{3}{8} * \log_2\left(\frac{3}{8}\right)\right) = 0,954434$$

Hasil *entropy*untukgejalapusing (tidak)adalah = 0,954434

Entropy Menggigil (ya)

$$\left(-\frac{1}{6} * \log_2\left(\frac{1}{6}\right)\right) + \left(-\frac{5}{6} * \log_2\left(\frac{5}{6}\right)\right) = 0,650022$$

Hasil *entropy* untuk gejala menggigil (ya) adalah = 0,650022

Entropy Menggigil (tidak)

$$\left(-\frac{8}{11} * \log_2\left(\frac{8}{11}\right)\right) + \left(-\frac{3}{11} * \log_2\left(\frac{3}{11}\right)\right) = 0,845351$$

Hasil *entropy* untuk gejala menggigil (tidak) adalah = 0,845351

Entropy Benjolan (ya)

$$\left(-\frac{9}{13} * \log_2\left(\frac{9}{13}\right)\right) + \left(-\frac{4}{13} * \log_2\left(\frac{4}{13}\right)\right) = 0,890492$$

Hasil *entropy* untuk gejala benjolan (ya) adalah = 0,890492

Entropy Benjolan (tidak)

$$\left(-\frac{0}{4} * \log_2\left(\frac{0}{4}\right)\right) + \left(-\frac{4}{4} * \log_2\left(\frac{4}{4}\right)\right) = 0$$

Hasil *entropy* untuk gejala benjolan (tidak) adalah = 0

Entropy Tidak Selera Makan (ya)

$$\left(-\frac{3}{5} * \log_2\left(\frac{3}{5}\right)\right) + \left(-\frac{2}{5} * \log_2\left(\frac{2}{5}\right)\right) = 0,970951$$

Hasil *entropy* untuk gejala tidak selera makan (ya) adalah = 0,970951

Entropy Tidak Selera Makan (tidak)

$$\left(-\frac{6}{12} * \log_2\left(\frac{6}{12}\right)\right) + \left(-\frac{6}{12} * \log_2\left(\frac{6}{12}\right)\right) = 1$$

Hasil *entropy* untuk gejala tidak selera makan (tidak) adalah = 1

Entropy Nyeri Pada Benjolan (ya)

$$\left(-\frac{6}{8} * \log_2\left(\frac{6}{8}\right)\right) + \left(-\frac{2}{8} * \log_2\left(\frac{2}{8}\right)\right) = 0,811278$$

Hasil *entropy* untuk gejala nyeri pada benjolan (ya) adalah = 0,811278

Entropy Nyeri Pada Benjolan (tidak)

$$\left(-\frac{3}{9} * \log_2\left(\frac{3}{9}\right)\right) + \left(-\frac{6}{9} * \log_2\left(\frac{6}{9}\right)\right) = 0,918296$$

Hasil *entropy* untuk gejala nyeri pada benjolan (tidak) adalah = 0,918296

Entropy Kelelahan (ya)

$$\left(-\frac{3}{7} * \log_2\left(\frac{3}{7}\right)\right) + \left(-\frac{4}{7} * \log_2\left(\frac{4}{7}\right)\right) = 0,985228$$

Hasil *entropy* untuk gejala kelelahan (ya) adalah = 0,985228

Entropy Kelelahan (tidak)

$$\left(-\frac{6}{10} * \log_2\left(\frac{6}{10}\right)\right) + \left(-\frac{4}{10} * \log_2\left(\frac{4}{10}\right)\right) = 0,970951$$

Hasil *entropy* untuk gejala kelelahan (tidak) adalah = 0,970951

Menghitung Jumlah Gain

Pada proses ini dilakukan pencarian nilai *gain*.

Gain (Total Gejala Batuk)

$$0,997503 - \left(\left(\frac{11}{17} * 0,994030\right) + \left(\frac{6}{17} * 1\right)\right) = 0,001365$$

Hasil *gain* untuk total gejala batuk adalah = 0,001365

Gain (Total Gejala Demam)

$$0,997503 - \left(\left(\frac{10}{17} * 0,970951\right) + \left(\frac{7}{17} * 0,985228\right)\right) = 0,020673$$

Hasil *gain* untuk total gejala demam adalah = 0,020673

Gain (Total Gejala Pusing)

$$0,997503 - \left(\left(\frac{9}{17} * 0,991076\right) + \left(\frac{8}{17} * 0,954434\right)\right) = 0,023670$$

Hasil *gain* untuk total gejala pusing adalah = 0,023670

Gain (Total Gejala Menggigil)

$$0,997503 - \left(\left(\frac{6}{17} * 0,650022\right) + \left(\frac{11}{17} * 0,845351\right)\right) = 0,221091$$

Hasil *gain* untuk total gejala menggigil adalah = 0,221091

Gain (Total Gejala Benjolan)

$$0,997503 - \left(\left(\frac{13}{17} * 0,890492\right) + \left(\frac{4}{17} * 0\right)\right) = 0,316538$$

Hasil *gain* untuk total gejala benjolan adalah = 0,316538

Gain (Total Gejala Tidak Selera Makan)

$$0,997503 - \left(\left(\frac{5}{17} * 0,970951\right) + \left(\frac{12}{17} * 1\right)\right) = 0,006046$$

Hasil *gain* untuk total gejala tidak selera makan adalah = 0,006046

Gain (Total Gejala Nyeri Pada Benjolan)

$$0,997503 - \left(\left(\frac{8}{17} * 0,811278\right) + \left(\frac{9}{17} * 0,9182296\right)\right) = 0,129568$$

Hasil *gain* untuk total gejala nyeri pada benjolan adalah = 0,129568

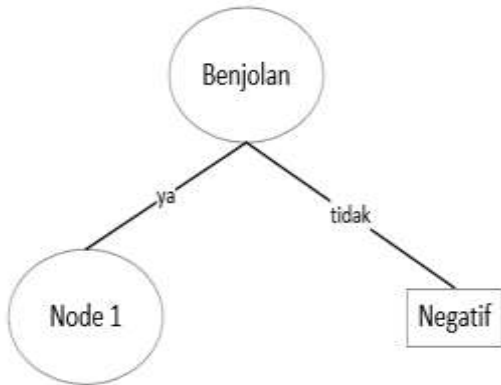
Gain (Total Gejala Kelelahan)

$$0,997503 - \left(\left(\frac{7}{17} * 0,985228\right) + \left(\frac{10}{17} * 0,970951\right)\right) = 0,020673$$

Hasil *gain* untuk total gejala tidak selera makan adalah = 0,020673

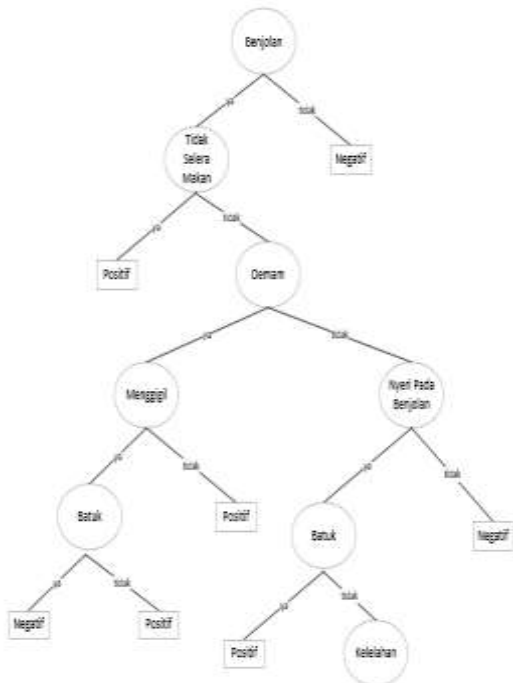
Pembentukan Pohon Keputusan

Setelah melakukan perhitungan didapatkan *gain* tertinggi pada gejala, kemudian *gain* yang tertinggi dijadikan sebagai *root*.



Gambar 5. Pembentukan Pohon Keputusan Diagnosa Limfadenopati

Perhitungan selanjutnya untuk mencari akar berikutnya sampai berakhir.



Gambar 6. Pembentukan Node Pohon Keputusan Diagnosa Limfadenopati

3. HASIL DAN PEMBAHASAN

Pengujian dari hasil klasifikasi diagnosa penyakit *Limfadenopati* dengan Algoritma C5.0 dapat menggunakan *confusion matrix*. Pengujian dengan *confusion matrix* menghasilkan nilai *accuracy*, *precision*, dan *recall*. Pengujian ini untuk melakukan verifikasi dengan membandingkan antara hasil yang ada pada data *training* dengan hasil sistem dengan 1 kali percobaan menggunakan 4 data *testing*.

Tabel 3 Data Testing

No	Batuk	Demam	Pusing	Menggigil	Benjolan	TSM
1	tidak	Ya	Ya	tidak	Ya	Tidak
2	ya	Tidak	Ya	tidak	Tidak	Ya
3	ya	Tidak	Ya	tidak	Ya	Tidak
4	ya	Ya	tidak	ya	Tidak	Tidak

Selanjutnya akan dihitung dengan *Confusion Matrix* untuk menemukan akurasi, presisi dan *recall*. Nilai *accuracy* merupakan persentase jumlah record data yang diklasifikasikan secara benar oleh sebuah algoritma. Nilai *precision* atau dikenal juga dengan nama *confidence* merupakan proporsi jumlah kasus yang diprediksi positif yang juga positif benar pada data yang sebenarnya. Sedangkan nilai dari *recall* atau *sensitivity* merupakan proporsi jumlah kasus positif yang sebenarnya yang diprediksi positif secara benar.

The screenshot shows a web interface titled "Uji Akurasi". It contains a table with 4 rows and 10 columns. The columns are labeled: No, Bahu, Denum, Faring, Tenggorokan, Tenggorokan, Tidak Selera Makan, Nyeri Pada Tenggorokan, Kelenjar, Batuk, and Hasil Diagnosis. The rows contain binary data (Ya/Tidak). Below the table, the following statistics are displayed: Nilai Akurasi : 75 %, Nilai Presisi : 50 %, and Nilai Recall : 100 %.

Gambar 6. Tampilan Pengujian *Confusion Matrix*

Hasil pengujian *confusion matrix* dapat dilihat pada tabel 4.

Tabel.4 Hasil tingkat akurasi, presisi dan recall dengan Confusion Matrix

	Prediksi	Aktual		Akurasi	Presisi	Recall
		True	False			
	True	1	1	75%	50%	100%
	False	0	2			

Dari tabel diatas, pengujian sistem telah dilakukan dengan *Confusion Matrix*. Dalam penelitian sistem ini telah didapatkan hasil nilai akurasi 75%, presisi 50% dan *recall* 100% pada algoritma C5.0.

KESIMPULAN

Berdasarkan pengujian yang telah dilakukan diperoleh kesimpulan bahwa algoritma C5.0 mampu melakukan

diagnosa awal penyakit *limfadenopati* dengan menggunakan pengujian *confusion matrix* dengan menggunakan data testing sebanyak 4 data dan mendapatkan hasil *accuracy* 75%, *precision* 50% dan *recall* 100%. Sehingga dengan hasil tersebut sistem dapat dikatakan berfungsi dengan baik.

DAFTAR PUSTAKA

[1] M. H. E. I. W. J. C. Pal, "Data Mining: Practical Machine Learning Tools and Techniques," vol. 4, 2016.

[2] T. Efraim and J. E. Aronson, Decision Support Systems and Intelligent Systems. 6th edition, Upper Saddle River, New Jersey: Prentice Hall, 2001.

[3] S. L. B. Ginting, W. Zarman and I. Hamidah, "Analisis Dan Penerapan Algoritma C4.5 Dalam Data Mining Untuk Memprediksi Masa Studi Mahasiswa Berdasarkan Data Nilai Akademik," *Prosiding Seminar Nasional Aplikasi Sains & Teknologi (SNAST)*, pp. 263-272, 2014.

[4] A. Sijabat, "PENERAPAN DATA MINING UNTUK PENGOLAHAN DATA SISWA DENGAN MENGGUNAKAN METODE DECISION TREE (STUDI KASUS : YAYASAN PERGURUAN KRISTEN ANDREAS)," *Informasi dan Teknologi Ilmiah (INTI)*, vol. 5, pp. 7-12, 2015.

[5] Y. Y. "Perbandingan Performasi Algoritma Decision Tree C5.0, CART, DAN CHAID : Kasus Prediksi Status Resiko Kredit di Bank X," *Seminar Nasional Aplikasi Teknologi Informasi 2007 (SNATI 2007)*, pp. B-61, 2007.

- [6] N. Patil, R. Lathi and V. Chitre,
"Comparison of C5.0 & CART
Classification algorithms using
pruning technique," *International
Journal of Engineering Research &
Technology (IJERT)*, Vols. Vol. 1. ISSN:
2278-0181, 2012.
- [7] B. P. Pratiwi, A. S. Handayani and S. ,
"Pengukuran Kinerja Sistem Kualitas
Udara Dengan Teknologi WSN
Menggunakan Confusion Matrix,"
vol. 6, pp. 66-75, 2020.