

Penerapan Case-Based Reasoning untuk Pendeteksian Penyakit Sapi dengan Algoritma Naïve Bayes

Clarissa E. Amos Pah¹, Derwin Rony Sina², Ahmad Z. Farhan³

^{1,2,3}Program Studi Ilmu Komputer, Fakultas Sains dan Teknik, Universitas Nusa Cendana, Kota Kupang, Nusa Tenggara Timur, Indonesia

Email : [1clarissaelfira@staf.undana.ac.id](mailto:clarissaelfira@staf.undana.ac.id), [2derwinsina@staf.undana.ac.id](mailto:derwinsina@staf.undana.ac.id), [3ahmadfarha000@gmail.com](mailto:ahmadfarha000@gmail.com)

Abstrak. Sapi adalah hewan ternak paling umum di Indonesia dengan jumlah 18.610.148 ekor pada tahun 2021. Seperti hewan ternak lainnya sapi dapat terserang berbagai macam penyakit. Berdasarkan data dari Pusat Kesehatan Hewan (Puskeswan) Kayu Putih, Nusa Tenggara Timur (NTT) di tahun 2022 penyakit yang biasanya dialami oleh sapi adalah Avitaminosis, Arthritis, Bloat, Malnutrisi, Hipocalsemia, Helminthiasis, Abses, Vulnus, dan Septicemia Epizootika. Penyakit-penyakit yang diderita sapi ini menjadi isu penting karena dapat membahayakan konsumen akibat tertular penyakit tersebut, khususnya di Kota Kupang yang menjadi produsen daging sapi terbanyak di NTT dengan 2.600.718 Kg pada tahun 2022. Adapun masalah lain yang dialami peternak sapi di Kota Kupang adalah kekurangan dokter hewan yang hanya berjumlah 11 orang dan harus menangani sapi dengan jumlah sebanyak 7.840 ekor pada tahun 2022. Oleh karena itu, salah satu solusi yang dapat diterapkan adalah pengembangan sebuah sistem pakar yang dapat memberikan diagnosis awal berdasarkan gejala yang diinput peternak. Sistem pakar yang dibangun pada penelitian ini adalah sistem berbasis kasus atau Case-Based Reasoning (CBR) dengan metode indexing Naïve Bayes yang dapat memudahkan peternak mendiagnosis penyakit ternak sapi. CBR merupakan teknik pencarian solusi berdasarkan pengalaman masa lalu untuk menyelesaikan masalah di masa mendatang. Adapun tahapan yang dilakukan pada CBR adalah Retrieve, Reuse, Revise, dan Retain. Hasil pengujian sistem Case-Based Reasoning menggunakan 8-fold cross validation pada 96 data didapat rata-rata similarity 99% dengan rata-rata waktu 0,071 detik. Sistem yang dibangun berbasis web untuk memudahkan pengaksesan secara dalam jaringan oleh pengguna.

Kata Kunci: Case-Based Reasoning (CBR), Penyakit Sapi, Naïve Bayes, K-Fold Cross Validation

Abstract. Cattle are the most common farm animal in Indonesia with 18,610,148 head in 2021. Like other farm animals, cattle can be affected by various diseases. Based on data from the Animal Health Center (Puskeswan) Kayu Putih, East Nusa Tenggara (NTT) in 2022, diseases that are usually experienced by cattle are Avitaminosis, Arthritis, Bloat, Malnutrition, Hypocalsemia, Helminthiasis, Abscess, Vulnus, and Septicemia Epizootika. These diseases have become an important issue because they can endanger consumers by infecting them, especially in Kupang City which is the largest beef producer in NTT with 2,600,718 Kg in 2022. Another problem experienced by cattle farmers in Kupang City is the shortage of veterinarians, with only 11 veterinarians, who must handle a total of 7,840 cattle in 2022. Therefore, one solution that can be applied is the development of an expert system that can provide an initial diagnosis based on the symptoms inputted by farmers. The expert system built in this research is a Case-Based Reasoning (CBR) system with the Naïve Bayes indexing method that can make it easier for farmers to diagnose their cattle diseases. CBR is a technique for finding solutions based on experience to solve problems in the future. The stages carried out in CBR are Retrieve, Reuse, Revise, and Retain. The results of testing the Case-Based Reasoning system using 8-fold cross validation on 96 data obtained an average similarity of 99% with an average time of 0.071 seconds. The system built is web-based to facilitate access to the network by users.

Keyword: Case-Based Reasoning (CBR), Cattle Disease, Naïve Bayes, K-Fold Cross Validation

PENDAHULUAN

Menurut data Peternakan Dalam Angka 2022, sapi adalah hewan ternak terbanyak kedua di Indonesia setelah unggas dengan jumlah 18.610.148 ekor pada tahun 2021 [1]. Berdasarkan data dari Badan Pusat Statistik (BPS) NTT, Kota Kupang menjadi produsen daging sapi terbanyak di NTT dengan 2.600.718 kg pada tahun 2022. Besarnya angka produksi ini seiring dengan manfaat yang akan didapatkan dari sapi seperti konsumsi daging, pemanfaatan kulit sapi, dan lainnya. Namun,



angka produksi ini dapat menurun akibat terpapar berbagai jenis penyakit sapi yang disebabkan oleh berbagai jenis bakteri, virus, parasit, dan cacing. Masalah penyakit pada ternak sapi menyebabkan penurunan produksi karena dapat membahayakan konsumen akibat tertular penyakit tersebut [2].

Berdasarkan data pada Pusat Kesehatan Hewan (Puskeswan) Kayu Putih, Kota Kupang di tahun 2022, penyakit yang biasanya diderita sapi adalah Avitaminosis, Arthritis, Bloat, Malnutrisi, Hipocalsemia, Helminthiasis, Abses, Vulnus, dan Septicemia Epizootika. Penyakit sapi bukan hal yang asing bagi para peternak, namun minimnya pengetahuan terhadap penyakit apa yang menyerang hewan ternaknya membuat para peternak tidak dapat langsung menangani penyakit yang dialami oleh hewan ternaknya, sehingga dibutuhkan pakar dalam hal ini dokter hewan untuk mendiagnosis dan menangani penyakit sapi. Namun, terbatasnya jumlah dokter hewan menyebabkan kesulitan berkonsultasi ketika dibutuhkan peternak. Berdasarkan wawancara dengan drh. Maria F. Vivi Jaman, Kota Kupang kekurangan dokter hewan yang dimana hanya berjumlah 11 orang dan harus menangani sapi dengan jumlah sebanyak 7.840 ekor pada tahun 2022.

Untuk mengatasi masalah yang telah dipaparkan, penulis menawarkan solusi untuk membangun sebuah sistem pakar berbasis kasus (CBR) untuk menyelesaikan suatu kasus baru dengan cara mengadaptasi atau mengingat solusi yang terdapat pada kasus sebelumnya yang mirip dengan kasus baru tersebut dan dapat menggantikan pakar untuk mempermudah peternak dalam mengidentifikasi penyakit sapi dan mendapatkan solusi yang diberikan oleh sistem tersebut [3]. CBR membutuhkan metode *indexing* untuk mengefisienkan waktu dan memori. pada penelitian ini metode *indexing* yang akan digunakan adalah Naïve Bayes yang merupakan metode statistik bayesian sederhana [4]. Sistem yang dibangun berbasis web sehingga dapat diakses langsung oleh para peternak.

METODOLOGI PENELITIAN

1. Tahapan Penelitian

Tahapan penelitian yang telah dilakukan untuk menyelesaikan penelitian ini adalah pengumpulan data, pengolahan data, perancangan sistem, dan pengujian sistem.

1. Pengumpulan data dilakukan dengan observasi dan wawancara dengan drh. Maria F. Vivi Jaman pada Puskeswan Kayu Putih untuk mendapatkan data penyakit sapi yang pernah didiagnosis pada tahun 2020 sampai 2023.
2. Pengolahan data dilakukan dengan penyuntingan untuk menghasilkan data awal penyakit, gejala, dan diagnosis (kedepannya akan disebut Basis Kasus) yang akan digunakan pada sistem CBR. Hasil penyuntingan data memperoleh 96 data diagnosis dengan melibatkan 9 penyakit dan 23 gejala. Sampel data data Penyakit, Gejala, dan Basis Kasus dapat dilihat pada Tabel 1, Tabel 2, dan Tabel 3.

Tabel 1. Penyakit Sapi

No	Kode	Penyakit
1	P01	<i>Avitaminosis</i>
2	P02	<i>Athritis</i>
3	P03	<i>Bloat</i>
4	P04	<i>Malnutrisi</i>
5	P05	<i>Hipocalsemia</i>
6	P06	<i>Helminthiasis</i>
7	P07	<i>Abses</i>
8	P08	<i>Vulnus</i>
9	P09	<i>Septicemia Epizootika</i>

Tabel 2. Gejala Penyakit Sapi

No.	Kode	Gejala
-----	------	--------



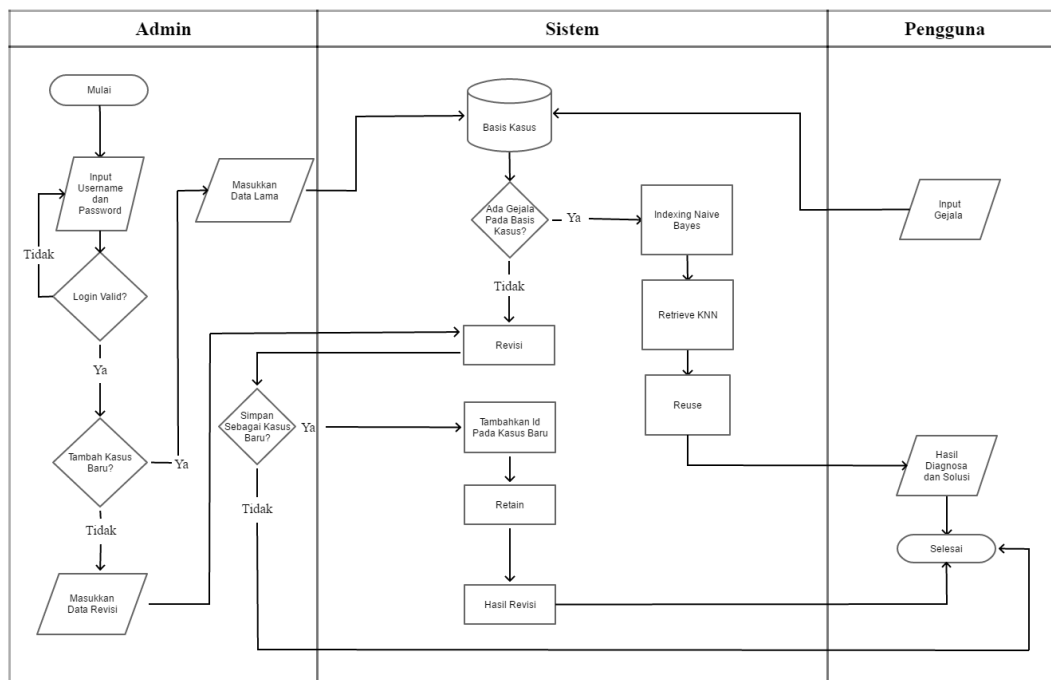
1.	G01	Nafsu Makan Menurun
2.	G02	Kelemahan Otot
3.	G03	Perut Kembung
4.	G04	Susah Buang Air Besar
5.	G05	Sulit Berjalan
6.	G06	Pembengkakan Pada Sendi Kaki
7.	G07	Kurus
...
23.	G23	Demam

Tabel 3. Basis Kasus

No.	Penyakit	Gejala
1	Avitaminosis	Nafsu makan menurun
		Kelemahan otot
2	Athritis	Kelemahan otot
		Sulit berjalan
		Pembengkakan pada sendi kaki
3	Bloat	Nafsu makan menurun
		Perut Kembung
		Susah buang air besar
...
96	vulnus	luka pada salah satu bagian tubuh

2. Perancangan Sistem

Sistem CBR yang dibangun digunakan oleh Peternak Sapi sebagai Pengguna dan Pakar/Dokter Penyakit Sapi sebagai Admin. *Flowchart* rancangan sistem terdapat pada



Gambar 1. Flowchart Perancangan Sistem

Gambar 1. Sistem yang dibangun menerapkan tahapan CBR, sistem akan mengambil kasus yang paling mirip (*retrieve*) pada saat kasus baru muncul. Tahap *retrieve* ini menggunakan

indexing dengan *Naïve Bayes* untuk mendapatkan beberapa kasus yang paling mirip. Selanjutnya dilakukan perhitungan *similarity* untuk melihat pertentase kemiripan kasus baru dengan kasus-kasus hasil klasifikasi *Naïve Bayes*. Kasus yang paling mirip tersebut akan digunakan kembali (*reuse*) untuk mendapatkan solusi yang tepat bagi kasus baru. Apabila tidak ada kasus yang mirip atau kemiripan kurang dari 80%, maka kasus baru ini akan dipertimbangkan dan disesuaikan (*revise*) untuk dimasukkan ke tabel basis kasus sehingga dapat menjadi pengetahuan baru yang dapat digunakan untuk kasus baru selanjutnya (*retain*).

3. Pengujian Sistem

Pengujian sistem dilakukan dengan menggunakan *K-Fold Cross Validation* dengan $K=8$. Masing-masing *fold* akan menghasilkan persentase *similarity* antara data uji dan data latih, yang kemudian dirata-ratakan. Semakin tinggi persentase *similarity*, maka basis kasus semakin akurat dalam memberi diagnosis.

3. Case-Based Reasoning (CBR)

CBR merupakan metode pemecahan masalah dengan menggunakan peran pengalaman sebelumnya. Pengalaman masa lalu disimpan di tempat yang disebut Basis Kasus. Salah satu aplikasi CBR untuk menyelesaikan masalahnya adalah bidang Kesehatan, yakni dengan mendiagnosis penyakit yang diderita pasien [5]. Awalnya, riwayat kasus (kumpulan gejala dan diagnosis yang diberikan yang sudah pernah terjadi) yang sudah divalidasi kebenarannya oleh Pakar/Dokter akan dikumpulkan dan dimasukkan ke dalam Basis Kasus, selanjutnya, riwayat ini akan digunakan untuk menjadi referensi diagnosis pada kasus baru mendatang. Kasus baru yang sudah divalidasi, akan dimenjadi riwayat dan menambah pengetahuan pada basis kasus. Semakin banyak variasi kasus, maka CBR akan memberikan solusi yang semakin akurat. Menurut [6], tahap-tahap penyelesaian masalah berbasis CBR adalah sebagai berikut:

- Pengambilan kasus-kasus yang sesuai dari sebuah basis kasus/memori (*retrieval*). Basis kasus membutuhkan *indexing* untuk mengelompokkan kasus-kasus sesuai kesamaan fitur-fiturnya.
- Pemilihan kasus-kasus terbaik sesuai dengan kasus baru yang diinput (*reuse*).
- Pemilihan solusi berdasarkan kasus yang terpilih sebagai penyelesaian (*reuse*).
- Evaluasi penyelesaian (*revise*) untuk menyerap kasus baru yang sudah disolusikan ke dalam basis kasus dengan validasi oleh pakar.
- Penyimpanan hasil *revise* atau penambahan kasus baru untuk digunakan pada basis kasus yang akan menjadi pengalaman untuk kasus baru ke depan.

4. Naïve Bayes

Metode *Naïve Bayes* berasal dari hukum perkalian yang melibatkan probabilitas kondisional, di mana dua peristiwa dikatakan memiliki hubungan bersyarat jika peristiwa yang satu memiliki syarat untuk peristiwa yang lain. Metode ini menghitung sekumpulan probabilitas dengan menjumlahkan frekuensi dan kombinasi nilai dari riwayat data yang diberikan. Teorema *Bayes* mengasumsikan semua nilai atribut (peristiwa) bersifat independen atau saling lepas [7]. *Naïve Bayes* didasarkan pada asumsi penyederhanaan bahwa nilai atribut secara kondisional saling bebas jika diberikan nilai output. Dengan kata lain, diberikan nilai output, probabilitas mengamati secara bersama adalah produk dari probabilitas individu [8]. Persamaan *Naïve Bayes* dapat dilihat pada Persamaan (1).

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)} = \frac{P(X|C_i) P(C_i)}{\sum_{i=1}^n P(X|C_i)P(C_i)} \quad (1)$$

Keterangan:

$P(C_i|X)$ = Peluang hipotesis C jika terdapat atribut X (probabilitas posterior)

$P(X|C_i)$ = Peluang atribut X terhadap terpilihnya hipotesis C

$P(C_i)$ = Peluang hipotesis C (probabilitas prior)

$P(X)$ = Peluang atribut X yang muncul (*evidence*)



5. Similarity

Metode kemiripan/*similarity* merepresentasikan nilai kedekatan didasarkan pada seberapa mirip kasus baru dan kasus lama. Beberapa kasus baru dan kasus lama akan dibandingkan, nilai kemiripan dalam penelitian ini akan sama dengan 1 jika atributnya sama persis, tetapi 0 jika tidak. *Similarity* dihitung dengan persamaan (2) [9].

$$SM_{pq} = \frac{\alpha (common)}{\alpha (common) + \beta (different)} \tag{2}$$

Keterangan:

SM = *Similarity Measure*

P = Kasus baru

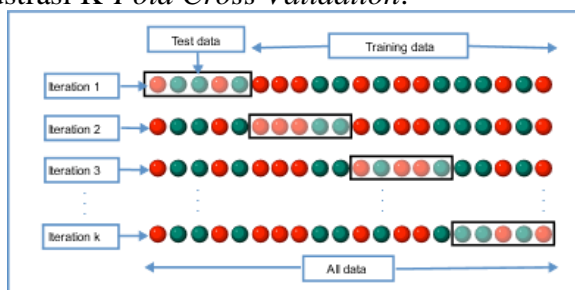
Q = Kasus yang tersimpan di basis kasus

$\alpha (common)$ = Fungsi jumlah atribut yang sama

$\beta (different)$ = Fungsi jumlah atribut yang berbeda

6. K-Fold Cross Validation

Pada metode pengujian ini, data dibagi menjadi beberapa bagian yang disebut *folds*. Banyaknya *folds* (*K*) ditentukan oleh penguji data. Semua data dibagi sama rata ke semua *fold* sehingga setiap *fold* memiliki data berjumlah sama atau mendekati sama jika ada sisa bagi. Saat pengujian pada salah satu *fold*, salah satu *fold* dipilih sebagai data uji, sedangkan sisa *k-1-fold* dijadikan data latih [10]. Gambar 2 menunjukkan ilustrasi *K-Fold Cross Validation*.



Gambar 2. Ilustrasi *K-Fold Cross Validation*

HASIL DAN PEMBAHASAN

Hasil penelitian penerapan *Case-Based Reasoning* untuk pendeteksian penyakit sapi diperoleh sebagai berikut:

1.1. Retrive

Untuk mengefisiensikan waktu dan memori dilakukan proses *indexing* menggunakan algoritma *Naïve Bayes* antara kasus lama dan kasus baru. Setelah mendapat hasil *indexing* akan dilakukan perhitungan *similarity* menggunakan persamaan (2). Sebagai contoh, terdapat kasus baru dengan gejala:

- Nafsu makan menurun
- Perut Kembung
- Susah buang air besar
- Bulu berdiri

Dari contoh kasus diatas akan dihitung menggunakan algoritma *Naïve Bayes* dengan persamaan (1) terhadap 96 kasus lama pada Basis Kasus dan menghasilkan *indexing* pada Tabel 4.

Tabel 4. Hasil *Indexing Naïve Bayes*

No	Gejala	Kelas Kasus
1	Nafsu makan menurun	Malnutrisi
	Perut Kembung	

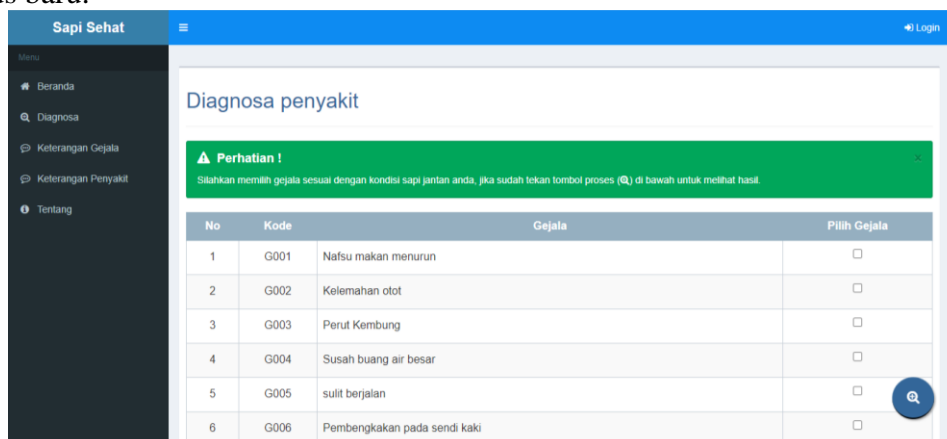
No	Gejala	Kelas Kasus
	Susah buang air besar	
2	Nafsu makan menurun	Malnutrisi
	Perut Kembung	
3	Nafsu makan menurun	Malnutrisi
	Susah buang air besar	
4	Perut Kembung	Malnutrisi
	Susah buang air besar	
5	Nafsu makan menurun	Malnutrisi
	Perut Kembung	
	Susah buang air besar	
	Bulu berdiri	
6	Perut Kembung	Malnutrisi
	Susah buang air besar	
	Bulu berdiri	
7	Nafsu makan menurun	Malnutrisi
	Kurus	
	Bulu berdiri	
8	Nafsu makan menurun	Malnutrisi
	Susah buang air besar	
	Bulu berdiri	

Setelah didapat hasil *indexing*, selanjutnya dilakukan penghitungan *similarity* menggunakan Persamaan (2) dan mendapatkan hasil seperti yang ditunjukkan pada Tabel 5.

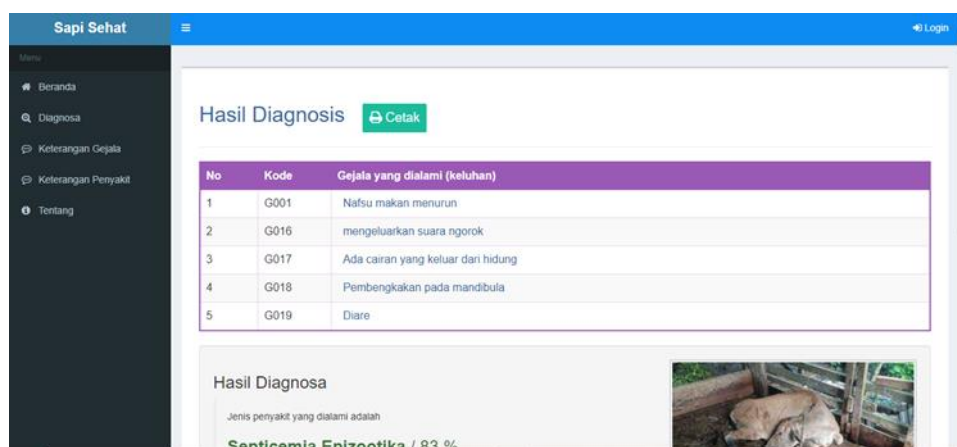
Tabel 5. Hasil Perhitungan *Similarity*

No	Gejala	Kelas Kasus	<i>Similarity</i>
1	Nafsu makan menurun	Malnutrisi	75%
	Perut Kembung		
	Susah buang air besar		
2	Nafsu makan menurun	Malnutrisi	50%
	Perut Kembung		
3	Nafsu makan menurun	Malnutrisi	50%
	Susah buang air besar		
4	Perut Kembung	Malnutrisi	50%
	Susah buang air besar		
5	Nafsu makan menurun	Malnutrisi	100%
	Perut Kembung		
	Susah buang air besar		
	Bulu berdiri		
6	Perut Kembung	Malnutrisi	75%
	Susah buang air besar		
	Bulu berdiri		
7	Nafsu makan menurun	Malnutrisi	75%
	Kurus		
	Bulu berdiri		
8	Nafsu makan menurun	Malnutrisi	75%
	Susah buang air besar		
	Bulu berdiri		

Dari hasil perhitungan *similarity*, kasus yang diinput memiliki nilai kemiripan tertinggi dengan kasus ke-5 dengan *similarity* 100%. Pada sistem yang telah dibangun, tampilan penginputan kasus baru dapat dilihat pada Gambar 3, di mana pengguna akan menginput gejala-gejala yang terjadi sebagai kasus baru.



Gambar 3. Tampilan Formulir Untuk Mendiagnosis Kasus Baru



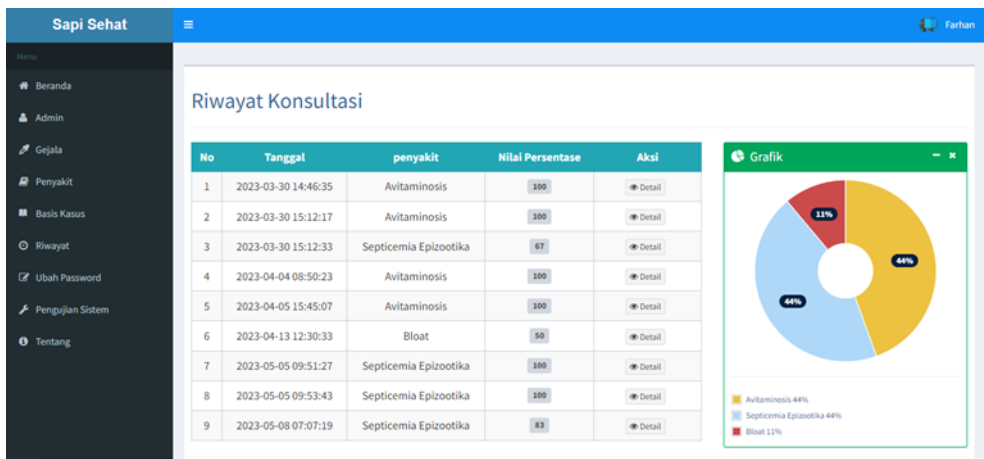
Gambar 4. Tampilan Hasil Diagnosis Untuk Kasus yang Telah Diinput Beserta Solusi Penanganannya.

1.2. Reuse

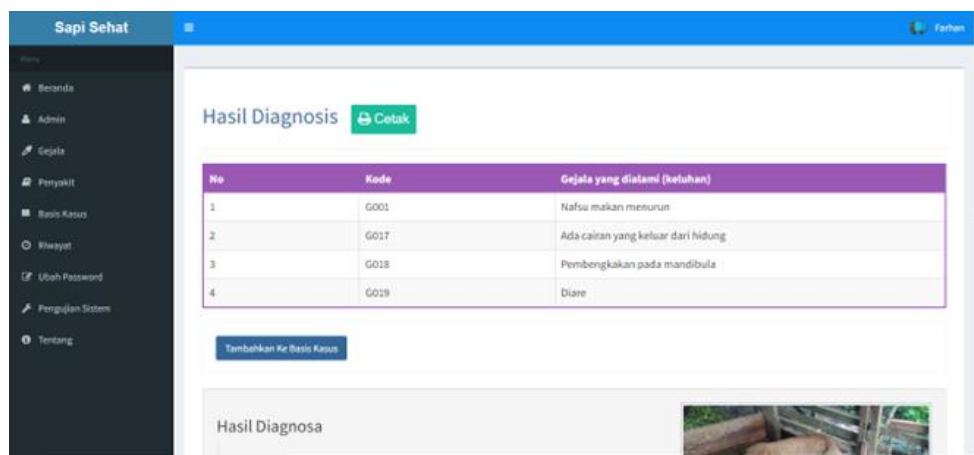
Setelah melewati tahap *indexing* serta perhitungan *similarity* didapatkan bahwa kasus baru memiliki tingkat *similarity* tertinggi pada kasus ke-5, maka dari itu solusi yang di berikan merupakan penanganan pada penyakit Malnutrisi. Contoh hasil reuse ditampilkan pada halaman Hasil Diagnosis yang terlihat pada Gambar 4.

1.3. Revise

Tahap *Revise* merupakan tahap menelaah kembali kasus dan solusi yang diberikan oleh sistem. Tahap ini dilakukan oleh Pakar/Dokter yang diset sebagai Admin pada sistem yang telah dibangun. Jika hasil yang diberikan terhadap kasus baru memiliki nilai *similarity* di bawah 80%, pakar akan melakukan peninjauan kembali agar kasus itu layak di masukan ke basis kasus. Gambar 5 menunjukkan riwayat konsultasi (diagnosis) dan *similarity*-nya. Gambar 6 merupakan tampilan detail konsultasi salah satu kasus yang memiliki nilai *similarity* di bawah 80% yang akan direvisi, terlihat tombol “Tambah ke Basis Kasus” untuk menambahkan kasus ini ke Basis Kasus. Saat tambah ke basis kasus, pakar akan melakukan revisi, lalu dilanjutkan penyimpanan kasus baru ke Basis Kasus.



Gambar 5. Tampilan Riwayat Konsultasi



Gambar 6. Detail Hasil Diagnosis Salah Satu Riwayat Konsultasi (Kasus)

1.4. Retain

Proses *retain* dilakukan jika kasus baru mendapatkan nilai *similarity* lebih besar sama dengan 80%, yang artinya solusi dan diagnosis penyakit telah dilakukan dengan tepat, maka hasil kasus tersebut akan ditambahkan ke basis kasus. Proses *retain* juga menyimpan kasus yang telah direvisi pakar pada tahap sebelumnya.

1.5. Hasil Pengujian Sistem

Pengujian dilakukan dengan menggunakan *K-Fold Cross Validation* dengan $K=8$. Proses pengujian dilakukan dengan memvalidasi data dengan membagi data tersebut menjadi kelompok data latih dan kelompok data uji, dengan tiap *fold* terdapat 12 data uji dan 84 data latih. Hasil pengujian dapat dilihat pada Tabel 6.

Tabel 6. Hasil Pengujian 8-Fold Cross Validation

<i>Fold</i>	<i>Similarity</i>	Rata-rata waktu per- <i>fold</i>
1	92%	0,081 detik
2	100%	0,08 detik
3	100%	0,074 detik
4	100%	0,062 detik
5	100%	0,07 detik
6	100%	0,072 detik
7	100%	0,074 detik
8	100%	0,06 detik

<i>Fold</i>	<i>Similarity</i>	<i>Rata-rata waktu per-fold</i>
Rata-rata	99%	0,071 detik

Berdasarkan pengujian di atas, maka dapat dilihat tingkat *similarity* sistem paling rendah adalah 92% dan paling tinggi 100% dengan rata-rata *similarity* system mencapai 99%. Dapat dilihat juga waktu pengujian sistem paling lama adalah 0,081 detik dan waktu paling cepat adalah 0,06 detik dengan rata-rata 0.071 detik waktu pengujian.

KESIMPULAN

Berdasarkan hasil penelitian sistem penerapan CBR untuk pendeteksian penyakit sapi dengan algoritma *Naïve Bayes*, maka dapat disimpulkan bahwa sistem penerapan CBR untuk pendeteksian penyakit sapi dengan algoritma *Naïve Bayes* mampu melakukan tahap Retrieve, Reuse, Revise, dan Retain pada kasus baru yang ditambahkan guna mendiagnosis penyakit yang dialami sapi sehingga memudahkan peternak agar tidak harus menemui pakar untuk mengetahui penyakit pada sapi. Sistem ini telah diuji dengan menggunakan *K-Fold Cross Validation* pada 96 dataset dan dibagi menjadi 84 data latih dan 12 data uji menjadi 8 *fold* dan memperoleh rata-rata hasil *similarity* 99% dan waktu pengujian rata-rata 0,071 detik. Sistem ini bermanfaat karena memudahkan peternak untuk dapat mengetahui penyakit sapi mereka lebih awal tanpa harus menemui pakar. Peternak juga dapat melakukan penanganan penyakit yang diderita sapi berdasarkan solusi penanganan yang diberikan oleh sistem.

DAFTAR PUSTAKA

- [1] "Peternakan Dalam Angka 2022," Direktorat Statistik Peternakan, Perikanan dan Kehutanan, 2022.
- [2] S. P. Faransyah, S. A. S. Mola and Y. Y. Nabuasa, "Implementasi Case Base Reasoning Menggunakan Metode Cosine Similarity Untuk Mendiagnosa Penyakit Pada Sapi," *J-Icon : Jurnal Komputer dan Informatika*, vol. 6, no. 2, pp. 47-52, 2018.
- [3] S. K. Pal and S. C. K. Shiu, *Foundations of Soft Case-Based Reasoning*, Canada: John Wiley & Sons, Inc, 2004C.
- [4] S. D. Jadhav and H. P. Channe, "Comparative Study of K-NN, Naive Bayes and Decision Tree Classification Techniques," *International Journal of Science and Research (IJSR)*, vol. 5, no. 1, pp. 1842-1845, 2013.
- [5] S. Mulyana and S. Hartati, "Tinjauan Singkat Perkembangan Case-Based Reasoning," in *Seminar Nasional Informatika 2009 UPN "Veteran" Yogyakarta*, Yogyakarta, 2009.
- [6] R. L. Mantaras, D. Mcsherry, D. Bridge, D. Leake, B. Smyth, S. Craw, B. Faltings, M. L. Maher, M. T. Cox, K. Forbus, M. Keane, A. Aamodt and I. Watson, "Retrieval, Reuse, Revision And Retention In Case-Based Reasoning," *The Knowledge Engineering Review*, vol. 20, no. 3, pp. 215-240, 2006.
- [7] T. R. Patil and M. S. Sherekar, "Performance Analysis of Naive Bayes and J48 Classification Algorithm for Data Classification," *International Journal of Computer Science and Applications*, vol. 6, no. 2, pp. 256-261, 2013.
- [8] M. Ridwan, H. Suyono and M. Sarosa, "Penerapan Data Mining Untuk Evaluasi Kinerja Akademik Mahasiswa Menggunakan Algoritma Naive Bayes Classifier," *Jurnal EECCIS*, vol. 7, no. 1, pp. 59-64, 2013.
- [9] A. Tversky, "Features of Similarity," *Psychological review*, vol. 84, no. 4, pp. 327-352, 1977.
- [10] P. Refaeilzadeh, L. Tang and H. Liu, "Cross-Validation," in *Encyclopedia of Database Systems*, New York, Springer US, 2009, pp. 532-538.

