

DIAGNOSIS GANGGUAN SISTEM URINARI PADA ANJING DAN KUCING MENGGUNAKAN VFI 5

Dhany Nugraha Ramdhany¹, Aziz Kustiyo², Ekowati Handharyani³, dan Agus Buono⁴

^{1, 2, 4} Departemen Ilmu Komputer, FMIPA, IPB, Kampus IPB Darmaga

³ Departemen Klinik, Reproduksi dan Patologi, Fakultas Kedokteran Hewan, IPB
pudesha@yahoo.co.id

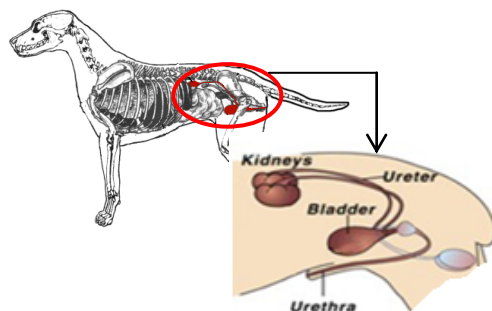
Abstrak

Sistem urinari hewan dapat dibagi menjadi 2 bagian yaitu sistem urinari bagian atas dan sistem urinari bagian bawah. Ginjal yang merupakan bagian dari sistem urinari memiliki 2 fungsi penting, yaitu filtrasi dan reabsorpsi. Dalam mendiagnosis penyakit yang diderita hewan pada sistem urinarnya terdapat beberapa kendala. Pada penelitian ini, dikembangkan model untuk mendiagnosis gangguan sistem urinari pada anjing dan kucing dengan menggunakan algoritma VFI 5 berdasarkan gejala klinis (terdapat 37 *feature*) dan pemeriksaan laboratorium (39 *feature*). Percobaan dilakukan baik pada *feature* gejala klinis dan juga pada *feature* pemeriksaan laboratorium. Hasil pengamatan yang dilakukan menunjukkan bahwa akurasi rata-rata sebesar 77,38% untuk percobaan dengan *feature* gejala klinis, dan 86,31% untuk percobaan dengan *feature* pemeriksaan laboratorium. Peningkatan ini mengindikasikan bahwa dalam mendiagnosis penyakit dalam sistem urinari, pemeriksaan laboratorium masih sangat dibutuhkan dalam menentukan hasil diagnosis suatu penyakit.

Kata kunci : *Voting Feature Interval 5 (VFI5), Renal Disease, Urinary Tract.*

1. Pendahuluan

Sistem urinari memiliki tiga fungsi, yaitu metabolisme, hormonal dan ekskresi. Sistem ini terdiri dari dua bagian, yaitu sistem urinari bagian atas dan bagian bawah. Sistem urinari bagian atas hanya terdiri dari ginjal sedangkan sistem urinari bagian bawah disusun oleh *ureter*, *vesica urinaria* (*gall bladder*) dan *urethra*, seperti diperlihatkan pada Gambar 1.



Gambar 1. Sistem Urinari Atas dan bawah pada Anjing

Pada sistem urinari, ginjal memiliki peranan yang sangat penting karena ia memiliki dua fungsi utama, yaitu filtrasi dan reabsorpsi. Selain itu, ginjal juga memiliki peranan penting dalam sistem sirkulasi darah. Ginjal turut berperan dalam proses pembentukan sel darah merah dan menjaga tekanan

darah [1].

Sama halnya pada manusia, hewan pun dapat mengalami gangguan pada sistem urinarnya. Gangguan tersebut dapat terjadi pada sistem urinari bagian bawah, bagian atas, maupun keduanya. Gangguan yang diderita baik oleh manusia maupun hewan, pada akhirnya dapat menyebabkan individu tersebut mengalami gagal ginjal, yaitu suatu keadaan tidak berfungsinya ginjal dengan baik, dan kondisi tersebut dapat menyebabkan kematian pada individu penderitanya.

Terdapat beberapa kendala dalam mendiagnosis gangguan sistem urinari pada hewan, antara lain: (1) hewan tidak dapat memberitahukan secara langsung apa keluhan yang dideritanya, dan (2) beberapa pemeriksaan yang dilakukan memerlukan biaya yang cukup besar sedangkan tidak semua pemilik hewan bersedia mengeluarkan dana yang cukup besar untuk pemeriksaan tersebut.

Kendala-kendala di atas dapat menyulitkan pemeriksaan dan penentuan diagnosis yang tepat apakah hewan tersebut terkena gangguan ginjal atau tidak. Padahal dibutuhkan diagnosis yang tepat untuk menentukan terapi yang sesuai. Hal ini dikarenakan terdapat beberapa obat yang bersifat meracuni ginjal. Jika hewan yang bermasalah pada ginjalnya tidak berhasil di diagnosis dengan tepat, maka terdapat kemungkinan terapi yang dilakukan mengharuskan hewan itu mengkonsumsi obat-obatan yang dapat meracuni ginjalnya. Jika hal itu terjadi saat ginjal hewan terganggu, maka kondisi ginjal terutama

fungsi dari ginjalnya akan semakin memburuk dan akan mempercepat proses terjadinya gagal ginjal.

Oleh karena itu diperlukan suatu cara/alat yang dapat membantu penentuan diagnosis, dalam kasus ini penyakit pada ginjal atau bukan, agar dapat ditentukan terapi yang tepat sehingga pada saat pengobatan tidak menimbulkan efek buruk pada ginjal.

Selanjutnya, *paper* ini disajikan dengan susunan sebagai berikut: Bagian 2 mengenai landasan teori yang terdiri dari sistem urinari, ginjal, *K-Fold Cross Validation*, dan *Voting Feature Intervals* (VFI5). Tahapan yang dilakukan pada penelitian ini disajikan pada bagian 3. Bagian 4 memberikan deskripsi mengenai hasil serta bahasannya. Bagian akhir *paper* ini menyajikan kesimpulan serta saran untuk penelitian selanjutnya.

2. Landasan Teori

2.1. Sistem Urinari

Sistem urinari adalah sistem organ dalam tubuh yang terdiri dari ginjal, *vesica urinaria*, *ureter* dan *urethra* [1,2]. Organ-organ tersebut berperan dalam produksi dan ekskresi urin. Organ utama dari sistem ini adalah ginjal yang memfiltrasi darah dan memproduksi urin sedangkan organ lainnya hanyalah struktur tambahan untuk menyimpan dan mengalirkan urin [2].

Sistem urinari memiliki tiga fungsi yaitu: metabolisme, hormonal dan ekskresi [3]. Sistem urinari bertanggung jawab dalam filtrasi kotoran dalam darah dan dalam produksi maupun sekresi urin.

2.1. Ginjal

Ginjal adalah organ yang terdapat pada daerah lumbal dan termasuk ke dalam bagian dari sistem urinari [1]. Fungsi dari ginjal adalah mem-*filter* darah, mengekskresikan urin dan mengatur konsentrasi hidrogen, sodium, potasium, fosfat dan ion-ion lain yang terdapat di dalam cairan ekstrasel.

2.2. Klasifikasi

Klasifikasi merupakan proses menemukan sekumpulan model (atau fungsi) yang menggambarkan dan membedakan konsep atau kelas-kelas data. Tujuannya adalah agar model tersebut dapat digunakan untuk memprediksi kelas dari suatu objek atau data yang label kelasnya tidak diketahui [4].

Klasifikasi terdiri atas dua tahap, yaitu pelatihan dan prediksi. Pada tahap pelatihan, dibentuk sebuah model domain permasalahan dari setiap *instance* yang ada. Penentuan model tersebut berdasarkan analisis pada sekumpulan data pelatihan, yaitu data yang label kelasnya sudah diketahui. Pada tahap prediksi, dilakukan prediksi kelas dari *instance* (kasus) baru dengan menggunakan model yang telah dibuat pada tahap pelatihan [5].

2.3. K-Fold Cross Validation

Validasi silang dan *bootstrapping* merupakan metode dalam memperkirakan generalisasi *error* berdasarkan *resampling* (Weis dan Kulikowski 1991, dalam [6]). Metode *k-fold cross validation* membagi sebuah himpunan contoh secara acak menjadi *k subset* yang saling bebas. Dilakukan pengulangan sebanyak *k*-kali untuk pelatihan dan pengujian. Pada setiap ulangan, disisakan satu *subset* untuk pengujian dan *subset* lainnya untuk pelatihan (Stone 1974, dalam [7]). Tingkat akurasi dihitung dengan membagi jumlah keseluruhan klasifikasi yang benar dengan jumlah semua *instance* pada data awal [4].

2.5. Voting Feature Intervals (VFI 5)

Algoritma klasifikasi *Voting Feature Intervals* (VFI 5) merepresentasikan deskripsi sebuah konsep oleh sekumpulan interval nilai-nilai *feature* atau atribut (Demiröz et al. 1997, [5] dan [8]). Pengklasifikasian *instance* baru berdasarkan voting pada klasifikasi yang dibuat oleh nilai tiap-tiap *feature* secara terpisah. VFI 5 merupakan algoritma klasifikasi yang bersifat non-incremental di mana semua *instance* pelatihan diproses secara bersamaan (Güvenir & Sirin 1996 dalam [8]).

Dari *instance-instance* pelatihan tersebut, Algoritma VFI 5 membuat interval untuk setiap *feature*. Interval-interval yang dibuat dapat berupa *range interval* atau *point interval*. *Point interval* terdiri atas seluruh *end point* secara berturut-turut. *Range interval* terdiri atas nilai-nilai antara dua *end point* yang berdekatan tetapi tidak termasuk kedua *end point* tersebut. Untuk setiap interval, *vote* setiap kelas pada interval tersebut disimpan. Dengan demikian, sebuah interval dapat merepresentasikan beberapa kelas dengan menyimpan *vote* setiap kelas, sehingga algoritma VFI dapat dikatakan sebagai *Multi-Class feature projection based algorithms*.

Keunggulan algoritma VFI 5 adalah algoritma ini cukup kokoh (*robust*) terhadap *feature* yang tidak relevan tetapi mampu memberikan hasil yang baik pada *real-world datasets* yang ada. VFI 5 mampu menghilangkan pengaruh yang kurang menguntungkan dari *feature* yang tidak relevan dengan mekanisme *voting*-nya, [5].

Algoritma Klasifikasi VFI 5 mampu melakukan klasifikasi lebih cepat dibandingkan dengan algoritma *Nearest Neighbor* dan *Decision Tree*, [9]. VFI 5 mampu menangani nilai *feature* yang tidak diketahui (hilang) dengan cara mengabaikan nilai *feature* tersebut yang ada pada data pelatihan dan data pengujian. Namun pada algoritma klasifikasi lain seperti *Nearest Neighbor* dan *Decision Tree*, nilai tersebut harus diganti (Quinlan 1993 dalam [9]). Algoritma VFI 5 terdiri atas dua tahap, yaitu pelatihan dan klasifikasi.

2.5.1. Pelatihan

Langkah pertama pada tahap *pelatihan* adalah menemukan *end point* setiap *feature f* dari setiap kelas *c*. *End point* untuk *feature* linier, yaitu *feature* yang nilainya memiliki urutan atau bisa dibandingkan tingkatannya, merupakan nilai minimum dan nilai maksimum *feature* tersebut. *End point* untuk *feature* nominal, yaitu *feature* yang nilainya tidak memiliki urutan dan tidak bisa dibandingkan tingkatannya, merupakan semua nilai yang berbeda yang ada pada *feature* kelas yang sedang diamati. Sebelum dibentuk interval, seluruh *end point* yang diperoleh untuk setiap *feature* linier diurutkan. Jika suatu *feature* merupakan *feature* linier maka akan dibentuk dua interval yaitu *point interval* dan *range interval*. Jika *feature* tersebut merupakan *feature* nominal maka hanya dibentuk *point interval*.

Batas bawah pada *range interval* (ujung paling kiri) adalah $-\infty$ sedangkan batas atas *range interval* (ujung paling kanan) adalah $+\infty$. Jumlah maksimum *end point* pada *feature* linier adalah $2k$ sedangkan jumlah maksimum intervalnya adalah $4k+1$, dengan k adalah jumlah kelas yang diamati.

Setelah itu, jumlah *instance* pelatihan setiap kelas *c* dengan *feature f* untuk setiap interval *i* dihitung dan direpresentasikan sebagai *interval_class_count[f,i,c]*. Untuk setiap *instance* pelatihan, dicari interval *i*, yaitu interval nilai *feature f* dari *instance* pelatihan *e* (e_f) tersebut berada. Jika interval *i* merupakan *point interval* dan e_f sama dengan batas bawah interval tersebut (sama dengan batas atas untuk *point interval*), jumlah kelas *instance* tersebut (e_f) pada interval *i* ditambah 1. Jika interval *i* merupakan *range interval* dan e_f jatuh pada interval tersebut maka jumlah kelas *instance* e_f pada interval *i* ditambah 1. Hasil proses ini merupakan *vote* kelas *c* pada interval *i*.

Untuk menghilangkan efek perbedaan distribusi setiap kelas, *vote* kelas *c* untuk *feature f* pada interval *i* dinormalisasi dengan cara membagi *vote* tersebut dengan jumlah *instance* kelas *c* yang direpresentasikan dengan *class_count[c]*. Hasil normalisasi ini dinotasikan sebagai *interval_class_vote[f,i,c]*. Kemudian nilai-nilai *interval_class_vote[f,i,c]* dinormalisasi sehingga jumlah *vote* beberapa kelas pada setiap *feature* sama dengan 1. Normalisasi ini bertujuan agar setiap *feature* memiliki kekuatan *voting* yang sama pada proses klasifikasi yang tidak dipengaruhi oleh ukurannya.

2.5.2. Prediksi (klasifikasi)

Proses klasifikasi diawali dengan inisialisasi *vote* setiap kelas dengan nilai nol. Untuk setiap *feature f*, dicari interval *i* di mana nilai e_f jatuh, dengan e_f merupakan nilai *feature f* dari *instance* tes *e*. Jika e_f tidak diketahui (hilang), *feature* tersebut tidak diikutsertakan dalam *voting* (memberikan *vote* nol untuk setiap kelas). Oleh karena itu, *feature* yang

memiliki nilai tidak diketahui diabaikan.

Jika e_f diketahui maka interval tersebut dapat ditemukan. Interval tersebut dapat menyimpan *instance* pelatihan dari beberapa kelas. Kelas-kelas dalam sebuah interval direpresentasikan oleh *vote* kelas-kelas tersebut pada interval itu. Untuk setiap kelas *c*, *feature f* memberikan *vote* yang sama dengan *interval_class_vote[f,i,c]*. Notasi tersebut merepresentasikan *vote feature f* yang diberikan untuk kelas *c*.

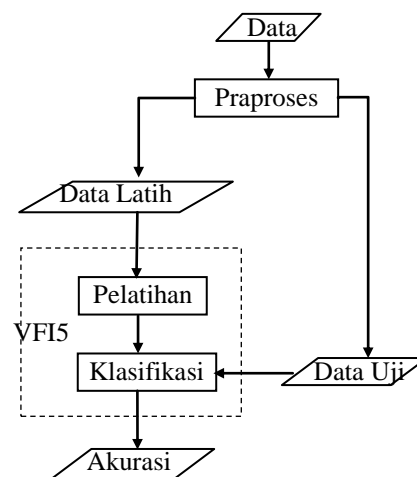
Setiap *feature f* mengumpulkan *vote-vote*-nya dalam sebuah vektor $\langle \text{feature_vote}[f,C_1], \dots, \text{feature_vote}[f,C_j], \dots, \text{feature_vote}[f,C_k] \rangle$, di mana *feature_vote[f,C_j]* merupakan *vote feature f* untuk kelas C_j dan k adalah jumlah kelas. Kemudian d vektor *vote*, di mana d merupakan jumlah *feature*, dijumlahkan untuk memperoleh total vektor *vote* $\langle \text{vote}[C_1], \dots, \text{vote}[C_k] \rangle$. Kelas dengan jumlah *vote* terbesar diprediksi sebagai kelas dari *instance* tes *e*. Algoritma detail dari VFI5 dapat dilihat pada [9].

3. Tahapan Proses Penelitian

Penelitian ini mempunyai beberapa tahapan proses untuk mengetahui akurasi yang diperoleh algoritma VFI 5 dalam pengklasifikasian penyakit yang menyerang ginjal dan non ginjal, seperti disajikan pada Gambar 2. Proses utama yang terjadi yaitu praproses, pelatihan untuk melihat domain permasalahan data dan prediksi (klasifikasi) untuk menduga kelas dari suatu *instance* baru.

3.1. Data

Semua data yang akan digunakan dalam penelitian ini adalah data hewan dalam hal ini anjing dan kucing yang menderita gangguan sistem urinari (ginjal dan non-ginjal) dari Rumah Sakit Hewan Institut Pertanian Bogor.



Gambar 2. Tahapan proses klasifikasi data.

3.2. Praproses

Sebelum digunakan dalam Algoritma VFI 5, *feature* pada setiap *instance* dipilih kembali agar diperoleh *feature-feature* yang bersesuaian dengan gejala pada gangguan sistem urinari. Proses ini akan menghasilkan 2 himpunan data yang sama tapi dengan jumlah *feature* yang berbeda untuk masing-masing himpunan.

3.3. Data Latih dan Data Uji

Seluruh data yang digunakan dibagi menjadi tiga *subset* dengan ukuran contoh yang lebih kurang sama berdasarkan metode *3-fold cross validation* yang digunakan. Pembagian data dilakukan secara acak dengan mempertahankan perbandingan jumlah *instance* setiap kelas. Ketiga *subset* yang dihasilkan akan digunakan sebagai data pelatihan dan data pengujian.

Sesuai dengan hasil praproses, maka data latih dan data uji pada penelitian ini akan digunakan dalam dua kali percobaan dengan jumlah *feature* yang berbeda. Pada percobaan pertama, *feature* pemeriksaan laboratorium akan dihilangkan sedangkan pada percobaan yang kedua *feature* pemeriksaan laboratorium akan digunakan. Hal ini dilakukan untuk membandingkan akurasi yang diperoleh dengan dan tanpa *feature* pemeriksaan laboratorium.

3.4. Algoritma VFI 5

Pada penelitian ini digunakan algoritma VFI 5 dengan bobot setiap *feature* diasumsikan seragam, yaitu satu. Bagian ini terdiri atas dua proses yaitu pelatihan dan prediksi (klasifikasi) kelas *instance* baru.

3.5. Pelatihan

Data yang telah dibagi menjadi beberapa *subset* kemudian digunakan sebagai input algoritma klasifikasi VFI 5. Pelatihan dilakukan dalam tiga kali iterasi. Pada iterasi pertama, *subset* yang digunakan sebagai data latih adalah *subset* kedua dan ketiga. Pada iterasi kedua, data latih terdiri dari *subset* pertama dan ketiga sedangkan pada iterasi ketiga, data latih terdiri dari *subset* pertama dan kedua.

Pada tahap pelatihan akan dibentuk interval-interval dari setiap *feature* yang ada. Jika *feature* tersebut adalah *feature* linier maka akan dibentuk dua buah interval, yaitu *point interval* dan *range interval*. Jika *feature* tersebut adalah *feature* nominal maka hanya akan dibentuk satu interval, yaitu *point interval*. Setelah semua interval terbentuk, langkah selanjutnya adalah menghitung jumlah *instance* setiap kelas yang berada pada setiap interval tersebut.

3.6. Klasifikasi

Pada tahap klasifikasi, setiap nilai *feature* dari suatu *instance* baru diperiksa letak interval nilai *feature* tersebut. Vote-vote setiap kelas untuk setiap

feature pada setiap interval yang bersesuaian diambil dan kemudian dijumlahkan. Kelas dengan nilai total vote tertinggi menjadi kelas prediksi *instance* baru tersebut. Sama seperti tahap pelatihan, tahap inipun dilakukan dalam tiga iterasi. Pada iterasi pertama, data uji yang digunakan untuk klasifikasi adalah data dari *subset* pertama. Pada iterasi kedua, data uji yang digunakan berasal dari *subset* kedua sedangkan pada iterasi ketiga, *subset* ketigalah yang digunakan sebagai data uji.

Proses pelatihan dan klasifikasi dilakukan selang-seling pada tiap iterasi. Pada iterasi pertama dilakukan pelatihan terlebih dahulu lalu dilanjutkan dengan klasifikasi, begitu seterusnya hingga tiga iterasi.

3.7. Akurasi

Hasil yang diamati pada penelitian ini adalah perubahan rata-rata tingkat akurasi algoritma VFI 5 dalam mengklasifikasikan data pengujian dengan menggunakan validasi silang, yaitu *3-fold cross validation*. Tingkat akurasi diperoleh dengan perhitungan :

$$\text{tingkat akurasi} = \frac{\sum \text{data uji benar diklasifikasi}}{\sum \text{total data uji}}$$

4. Hasil dan Analisis

Penelitian diawali dengan pengumpulan data rekam medik dari Rumah Sakit Hewan IPB, Bogor. Data yang diambil merupakan data hewan penderita gangguan sistem urinari dengan jenis hewan 2 yaitu anjing (*canine*) dan kucing (*feline*). Selanjutnya dilakukan tahap praproses terhadap data yang berhasil diperoleh. Tahap ini dilakukan dengan tujuan untuk mengurangi *feature* (gejala klinis dan hasil pemeriksaan laboratorium) yang diperoleh dalam tahap pengumpulan data sehingga didapatkan *feature-feature* yang relevan terhadap gangguan sistem urinari. Data rekam medik hasil praproses memiliki 39 *feature* yang terdiri dari 37 *feature* gejala klinis dan 2 *feature* pemeriksaan laboratorium. Pada Tabel 1 dapat terlihat spesifikasi data hasil praproses tanpa *feature* pemeriksaan laboratorium sedangkan pada Tabel 2 dapat terlihat spesifikasi data hasil praproses dengan *feature* pemeriksaan laboratorium.

Tabel 1. Spesifikasi data hasil praproses tanpa *feature* pemeriksaan laboratorium

Nama data	Jumlah <i>instance</i>	Jumlah <i>feature</i>	Jumlah kelas
Hasil pra-proses	44	37	2

Tabel 2. Spesifikasi data hasil praproses dengan *feature* pemeriksaan laboratorium

Nama data	Jumlah <i>instance</i>	Jumlah <i>feature</i>	Jumlah kelas
Hasil pra-proses	44	39	2

Feature-feature pada data rekam medik hasil praproses masing-masing dilambangkan berturut-turut dengan variabel F1, F2, F3, ..., F39.

Pembagian data latih dan data uji

Data keseluruhan sebanyak 44 *instances* hasil praproses terlebih dahulu dibagi menjadi tiga *subset* dengan jumlah *instance* yang hampir sama di tiap *subset* dan perbandingan ukuran kelas yang seimbang. Setiap *subset* yang dihasilkan lalu digunakan dalam metode validasi silang, yaitu metode *3-fold cross validation*. Tabel 3 menunjukkan proporsi jumlah *instance* pada setiap kelas untuk setiap *subset*.

Tabel 3. Proporsi jumlah *instance* setiap kelas pada masing-masing *subset*

Subset	Ginjal (G)	Non-Ginjal (N)
1	8	6
2	9	7
3	8	6

Dengan metode *3-fold cross validation*, maka akan dilakukan tiga kali iterasi dengan menggunakan dua *subset* sebagai data latih dan satu *subset* lain sebagai data uji. Data latih dan data uji pada masing-masing iterasi akan digunakan dalam dua kali percobaan dengan jumlah *feature* yang berbeda. Pada percobaan pertama, *feature* yang digunakan hanyalah *feature* gejala klinis tanpa *feature* pemeriksaan laboratorium. Pada percobaan kedua, *feature* yang digunakan merupakan gabungan dari *feature* gejala klinis dengan *feature* pemeriksaan laboratorium.

4.1 Percobaan pertama

Pada tahap ini, di setiap iterasinya digunakan data dengan jumlah *feature* 37 buah yang terdiri dari *feature* gejala klinis.

4.1.1. Iterasi pertama

Pada iterasi ini, proses pelatihan menghasilkan *vote* yang cukup berimbang pada beberapa *feature* untuk kedua kelas. Sebagai contoh, pada *feature* 5 (kiposis) *vote* yang diberikan untuk interval bernilai 0 (tidak terdapat gejala kiposis) hanya berbeda 0,05. Hal ini mempengaruhi kecilnya rasio peluang suatu individu uji diklasifikasikan sebagai kelas ginjal atau kelas non-ginjal.

Hasil proses klasifikasi pada iterasi ini dapat dilihat pada Tabel 4. Dari 8 *instances* kelas ginjal yang diuji, 7 *instances* benar diprediksi sebagai kelas ginjal dan sisanya 1 *instance* diprediksi sebagai kelas non-ginjal. Dengan demikian, akurasi untuk kelas ginjal pada iterasi pertama ini mencapai 87,50%. Untuk kelas non-ginjal, akurasi yang diperoleh sebesar 83,33% dengan jumlah *instances* yang benar diprediksi sebagai kelas non-ginjal sebanyak 5 *instances* dari 6 *instances* kelas non-ginjal yang diuji. Secara keseluruhan, jumlah *instances* yang benar diprediksi pada iterasi ini sebanyak 12 *instances* dari

14 *instances* yang diuji sehingga dihasilkan akurasi sebesar 85,71%.

Tabel 4. Hasil percobaan pertama pada iterasi pertama

Kelas	Diprediksi sebagai kelas		Akurasi
	G	N	
G	7	1	87.50%
N	1	5	83.33%
Akurasi iterasi pertama			85.71%

4.1.2. Iterasi kedua

Sama seperti pada iterasi pertama, *vote* yang dihasilkan dari proses pelatihan pada iterasi inipun cukup berimbang pada beberapa *feature* untuk kedua kelas bahkan terdapat *feature* yang memiliki nilai *vote* yang benar-benar seimbang di kedua kelas. Sebagai contoh, pada *feature* 30 (abdomen besar) *vote* yang diberikan di setiap interval sebesar 0,5 untuk masing-masing kelas. Hal ini mempengaruhi kecilnya rasio peluang suatu individu uji diklasifikasikan sebagai kelas ginjal atau kelas non-ginjal.

Hasil proses klasifikasi pada iterasi ini dapat dilihat pada Tabel 5. Dari 9 *instances* kelas ginjal yang diuji, 6 *instances* benar diprediksi sebagai kelas ginjal dan sisanya 3 *instances* diprediksi sebagai kelas non-ginjal. Dengan demikian, akurasi untuk kelas ginjal pada iterasi kedua ini mencapai 66,67%. Untuk kelas non-ginjal, akurasi yang diperoleh sebesar 85,71% dengan jumlah *instances* yang benar diprediksi sebagai kelas non-ginjal sebanyak 6 *instances* dari 7 *instances* kelas non-ginjal yang diuji. Secara keseluruhan, jumlah *instances* yang benar diprediksi pada iterasi ini sebanyak 12 *instances* dari 16 *instances* yang diuji sehingga dihasilkan akurasi sebesar 75%.

Sama seperti pada iterasi pertama dan kedua, *vote* yang dihasilkan dari proses pelatihan pada awal iterasi inipun cukup berimbang pada beberapa *feature* untuk kedua kelas bahkan terdapat setidaknya 4 *feature* yang memiliki nilai *vote* yang benar-benar seimbang di kedua kelas. Sebagai contoh, pada *feature* 11 (nafas aritmis) *vote* yang diberikan di setiap interval sebesar 0,5 untuk masing-masing kelas. Hal ini mempengaruhi kecilnya rasio peluang suatu individu uji diklasifikasikan sebagai kelas ginjal atau kelas non-ginjal.

Tabel 5. Hasil percobaan pertama pada iterasi kedua

Kelas	Diprediksi sebagai kelas		Akurasi
	G	N	
G	6	3	66.67%
N	1	6	85.71%
Akurasi iterasi kedua			75%

4.1.3. Iterasi ketiga

Hasil proses klasifikasi pada iterasi ini dapat dilihat pada Tabel 6. Dari 8 *instances* kelas ginjal yang diuji, 7 *instances* benar diprediksi sebagai kelas ginjal dan sisanya 1 *instances* diprediksi sebagai kelas non-ginjal. Dengan demikian, akurasi untuk kelas ginjal pada iterasi ketiga ini mencapai 87.50%. Untuk kelas non-ginjal, akurasi yang diperoleh hanya mencapai 50% dengan jumlah *instances* yang benar diprediksi sebagai kelas non-ginjal sebanyak 3 *instances* dari 6 *instances* kelas non-ginjal yang diuji. Secara keseluruhan, jumlah *instances* yang benar diprediksi pada iterasi ini sebanyak 10 *instances* dari 14 *instances* yang diuji sehingga dihasilkan akurasi sebesar 71,43%.

Pada percobaan pertama ini, digunakan data dengan jumlah *feature* 37 buah yang keseluruhannya merupakan gejala klinis yang mungkin diderita saat seekor hewan mengalami gangguan saluran urinari. Ke-37 buah *feature* tersebut dapat dibagi kedalam 7 bagian utama, yaitu bagian keadaan umum (terdiri dari 5 *feature*), mulut (2 *feature*), mukosa/selaput lendir (3 *feature*), nafas (5 *feature*), jantung (7 *feature*), ekskresi (7 *feature*), dan palpasi (8 *feature*). Dari ke-37 *feature* tersebut, hanya satu *feature* yang memiliki *range interval* yaitu *feature* suhu di bagian keadaan umum sedangkan ke-36 *feature* yang lainnya memiliki *point interval* (bernilai 1 untuk ditemukannya gejala klinis tertentu, dan bernilai 0 untuk tidak atau jika gejala yang ditemukan normal).

Tabel 6. Hasil percobaan pertama pada iterasi ketiga

Kelas	Diprediksi sebagai kelas		Akurasi
	G	N	
G	7	1	87.50%
N	3	3	50%
Akurasi iterasi ketiga			71.43%

4.1.4. Hasil pelatihan dan pengklasifikasian pada percobaan pertama

Hasil pelatihan dari ketiga iterasi yang dilakukan pada percobaan yang pertama ini memperlihatkan bahwa *vote* yang dihasilkan baik di iterasi pertama, kedua maupun ketiga cukup berimbang pada beberapa *feature* untuk kedua kelas. Hal ini akan mempengaruhi nilai total *vote* yang didapatkan *instances* uji pada suatu kelas. Pada akhirnya, akurasi dalam memprediksi kelas pada *instance* uji tersebut akan berkurang.

Hasil klasifikasi dari ketiga iterasi memperlihatkan kurang akuratnya prediksi yang diberikan dengan akurasi sebesar 85,71% pada iterasi pertama, 75% pada iterasi kedua, dan 71,43% pada iterasi ketiga. Secara keseluruhan, rata-rata akurasi yang dihasilkan algoritma VFI 5 pada percobaan pertama ini sebesar 77,38%.

4.2. Percobaan kedua

Pada tahap ini, di setiap iterasinya digunakan data dengan jumlah *feature* 39 buah yang terdiri dari 37 *feature* gejala klinis dan 2 *feature* pemeriksaan laboratorium. Pemeriksaan laboratorium yang digunakan merupakan pemeriksaan laboratorium yang berhubungan langsung dengan fungsi ginjal yaitu naik tidaknya nilai *ureum* dan *creatinine*. Anggota *subset* yang digunakan pada tahap ini tidak dipilih secara acak, melainkan menggunakan susunan anggota yang sama persis dengan *subset* pada percobaan pertama. Hal ini dilakukan agar dapat terlihat pengaruh dari penggunaan *feature* pemeriksaan laboratorium pada akurasi yang dihasilkan.

4.2.1. Iterasi pertama

Sama seperti pada iterasi pertama di percobaan pertama, proses pelatihan pada iterasi pertama di percobaan kedua ini pun menghasilkan *vote* yang cukup berimbang pada beberapa *feature* untuk kedua kelas. Akan tetapi, terlibatnya *feature* pemeriksaan laboratorium memberikan pengaruh yang cukup signifikan karena *feature* kenaikan nilai *creatinine* memberikan *vote* yang cukup tinggi (sebesar 0,74) untuk kelas non-ginjal saat *feature* tersebut bernilai 0 (nilai *creatinine* normal) sedangkan saat *feature* tersebut bernilai 1 (nilai *creatinine* meningkat) *vote* yang cukup tinggi (sebesar 1) diberikan pada kelas ginjal. Hal ini bersesuaian dengan fakta bahwa fungsi ginjal terganggu jika nilai *creatinine*-nya diatas batas normal.

Hasil proses klasifikasi pada iterasi ini dapat dilihat pada Tabel 7. Dari 8 *instances* kelas ginjal yang diuji, tidak ada satu *instance* pun yang diprediksi sebagai kelas non-ginjal, ini berarti akurasi untuk kelas ginjal pada iterasi pertama ini mencapai 100%. Nilai akurasi ini meningkat dari percobaan pertama, hal ini terjadi karena pada percobaan kedua ini, *instance* uji kedua pada *subset* pertama yaitu G14 benar diprediksi sebagai Ginjal sedangkan pada percobaan pertama G14 diprediksi sebagai non-ginjal. Untuk kelas non-ginjal, akurasi yang diperoleh sama dengan akurasi yang diperoleh pada iterasi pertama di percobaan pertama, yaitu sebesar 83,33% dengan jumlah *instances* yang benar diprediksi sebagai kelas non-ginjal sebanyak 5 *instances* dari 6 *instances* kelas non-ginjal yang diuji. Secara keseluruhan, jumlah *instances* yang benar diprediksi pada iterasi ini sebanyak 13 *instances* dari 14 *instances* yang diuji sehingga dihasilkan akurasi sebesar 92,86%.

4.2.2. Iterasi kedua

Sama seperti pada iterasi pertama di percobaan kedua, pada iterasi ini *feature* kenaikan nilai *creatinine* memberikan *vote* yang cukup tinggi (sebesar 0,75) untuk kelas non-ginjal saat *feature* tersebut bernilai 0 (nilai *creatinine* normal) sedangkan saat *feature* tersebut bernilai 1 (nilai

creatinine meningkat) *vote* yang cukup tinggi (sebesar 1) diberikan pada kelas ginjal. Selain itu, *vote* yang diberikan oleh *feature* kenaikan nilai *ureum* pun memberikan *vote* yang cukup tinggi (sebesar 0,80) untuk kelas non-ginjal saat *feature* tersebut bernilai 0 (nilai *ureum* normal).

Tabel 7. Hasil percobaan kedua pada iterasi pertama

Kelas	Diprediksi sebagai kelas		Akurasi
	G	N	
G	8	0	100%
N	1	5	83.33%
Akurasi iterasi pertama			92.86%

Hasil proses klasifikasi pada iterasi ini dapat dilihat pada Tabel 8. Dari 9 *instances* kelas ginjal yang diuji, 8 *instances* benar diprediksi sebagai kelas ginjal dan sisanya 1 *instances* diprediksi sebagai kelas non-ginjal. Dengan demikian, akurasi untuk kelas ginjal pada iterasi kedua ini mencapai 88,89% atau meningkat sebanyak 2 *instances* yang benar diprediksi. Kedua *instances* tersebut adalah G2 (*instance* uji ke-8) dan G1 (*instance* uji ke-9) yang berasal dari *subset* kedua yang dijadikan data uji. Untuk kelas non-ginjal, akurasi yang diperoleh sebesar 85,71% dengan jumlah *instances* yang benar diprediksi sebagai kelas non-ginjal sebanyak 6 *instances* dari 7 *instances* kelas non-ginjal yang diuji. Nilai akurasi ini sama dengan nilai akurasi yang diperoleh pada percobaan pertama, atau tidak mengalami peningkatan. Secara keseluruhan, jumlah *instances* yang benar diprediksi pada iterasi ini sebanyak 14 *instances* dari 16 *instances* yang diuji sehingga dihasilkan akurasi sebesar 87,5%.

Tabel 8. Hasil percobaan kedua pada iterasi kedua

Kelas	Diprediksi sebagai kelas		Akurasi
	G	N	
G	8	1	88.89%
N	1	6	85.71%
Akurasi iterasi ketiga			87.50%

4.2.3. Iterasi ketiga

Sama seperti pada iterasi kedua di percobaan kedua, pada iterasi ini *feature* kenaikan nilai *creatinine* dan *feature* kenaikan nilai *ureum* memberikan *vote* yang cukup tinggi. Untuk kelas non-ginjal *feature* kenaikan nilai *creatinine* memberikan *vote* sebesar 0,74 saat bernilai 0 (nilai *creatinine* normal) sedangkan *feature* kenaikan nilai *ureum* memberikan *vote* sebesar 0,82 saat bernilai 0 (nilai *ureum* normal). Untuk kelas ginjal, *feature* kenaikan nilai *creatinine* memberikan *vote* sebesar 1 saat bernilai 1 (nilai *creatinine* meningkat).

Tabel 9. Hasil percobaan kedua pada iterasi ketiga

Kelas	Diprediksi sebagai kelas		Akurasi
	G	N	
G	7	1	87.50%
N	2	4	66.67%
Akurasi iterasi ketiga			78.57%

Hasil proses klasifikasi pada iterasi ini dapat dilihat pada Tabel 9. Berbeda dengan kedua iterasi sebelumnya, pada iterasi ini, *instance* yang berhasil diperbaiki prediksinya justru berasal dari kelas non-ginjal. Dari 6 *instances* yang diuji, 4 *instances* benar diprediksi sebagai kelas non-ginjal, dan sisanya sebanyak 2 *instances* diprediksi sebagai kelas ginjal. Bila dibandingkan dengan hasil dari percobaan pertama, pada percobaan kedua ini terdapat satu *instance* uji yang pada percobaan pertama salah diprediksi, pada percobaan kedua ini justru benar diprediksi, *instance* tersebut adalah *instance* uji ke-12 dari *subset* ketiga yaitu *instance* N4. Dengan demikian akurasi yang diperoleh mencapai 66,67%. Untuk kelas ginjal, akurasi yang diperoleh tidak mengalami peningkatan dari percobaan pertama yaitu sebesar 87.50% dengan memprediksi benar 7 *instances* dari 8 *instances* yang diuji. Secara keseluruhan, jumlah *instances* yang benar diprediksi pada iterasi ini sebanyak 11 *instances* dari 14 *instances* yang diuji sehingga dihasilkan akurasi sebesar 78,57%.

4.2.4. Hasil pelatihan dan pengklasifikasian pada percobaan kedua

Pada percobaan yang kedua ini, digunakan data dengan jumlah *feature* 39 buah di mana 37 *feature* merupakan gejala klinis dan 2 lainnya merupakan pemeriksaan laboratorium. Pengelompokan 37 *feature* gejala klinis sama dengan pengelompokan saat percobaan pertama sedangkan 2 *feature* terakhir yang merupakan pemeriksaan laboratorium dibuatkan kelompok baru yaitu bagian fungsi ginjal. Dari ke-39 *feature* tersebut, hanya satu *feature* yang memiliki *range interval* yaitu *feature* suhu di bagian keadaan umum sedangkan ke-38 *feature* yang lainnya (termasuk *feature* pemeriksaan laboratorium) memiliki *point interval*. *Feature* pemeriksaan laboratorium dijadikan *point interval* karena pada prakteknya, pemeriksaan laboratorium memiliki batas normal sendiri untuk setiap jenis hewan sedangkan pada penelitian ini digunakan 2 jenis hewan yaitu kucing dan anjing. Oleh karena itu, agar nilai interval kedua jenis hewan itu sama, maka dibuatlah *point interval* yang bernilai 0 jika nilai *ureum/creatinine* normal dan 1 jika nilai *ureum/creatinine* meningkat.

Karena susunan anggota setiap *subset* pada percobaan kedua ini sama dengan yang di percobaan pertama, maka *vote* dan interval yang dihasilkannya pun sama, kecuali di *feature* pemeriksaan laboratorium. Pada *feature* pemeriksaan laboratorium, *vote* yang diberikan kepada kedua kelas memiliki

rasio yang cukup tinggi, hal ini terlihat terutama pada *feature* kenaikan nilai *creatinine*. Hal ini akan mempengaruhi nilai total *vote* yang didapatkan *instances* uji pada suatu kelas. Pada akhirnya, akurasi dalam memprediksi kelas pada *instance* uji tersebut akan meningkat jika dibandingkan dengan akurasi yang diperoleh pada percobaan pertama.

Hasil klasifikasi dari ketiga iterasi di percobaan kedua memperlihatkan peningkatan dibandingkan akurasi di percobaan pertama. Pada percobaan kedua, akurasi sebesar 92,86% dihasilkan pada iterasi pertama, akurasi sebesar 87,5% dihasilkan pada iterasi kedua, dan pada iterasi terakhir dihasilkan akurasi sebesar 78,57%. Secara keseluruhan, pada percobaan kedua ini algoritma VFI 5 dapat menghasilkan rata-rata akurasi sebesar 86,31%. Peningkatan akurasi ini diperoleh dengan memperbaiki prediksi 3 *instances* kelas ginjal (1 *instance* pada iterasi pertama dan 2 *instances* pada iterasi kedua) dan 1 *instance* kelas non-ginjal (pada iterasi ketiga) yang salah diprediksi saat percobaan pertama.

4.2.5. *Feature-feature* yang konsisten terhadap kelas tertentu

Pada setiap iterasi, dihasilkan selang-selang dengan nilai *vote* yang diberikan oleh setiap *feature* untuk masing-masing kelas.

Terdapat *feature-feature* yang konsisten memberikan nilai *vote* yang lebih besar kepada suatu kelas (baik kelas ginjal maupun kelas non-ginjal) pada setiap iterasi. *Feature-feature* tersebut adalah :

- Kelas ginjal
 - F2 = Muntah
 - F3 = Gemetar
 - F4 = Dehidrasi
 - F6 = *Ulcus* di mulut
 - F7 = Mulut bau *ureum*
 - F9 = Mukosa *icterus*
 - F10 = Mukosa *jaundice*
 - F13 = Frekuensi nafas cepat
 - F23 = Diare
 - F27 = Poliuria
 - F29 = Disuria
 - F31 = Palpasi abdomen sakit
 - F34 = Ginjal membesar
 - F35 = Ada batu di ginjal
 - F38 = *Ureum* naik
 - F39 = *Creatinine* naik
- Kelas non-ginjal
 - F8 = Mukosa *hiperemi*
 - F24 = *Konstipasi*
 - F32 = Abdomen tegang
 - F36 = Ada batu di *urethra/VU*

Selain itu terdapat pula *feature* yang di setiap iterasinya memberikan *vote* yang selalu seimbang untuk kedua kelas. *Feature* tersebut adalah F12 =

Frekuensi nafas lambat.

5 Kesimpulan

Penelitian yang telah dilakukan menghasilkan suatu model diagnosis gangguan sistem urinari yang diimplementasikan dalam suatu prototipe aplikasi diagnosis. Tingkat akurasi yang diperoleh pada proses klasifikasi tanpa *feature* pemeriksaan laboratorium lebih kecil daripada tingkat akurasi dari proses klasifikasi dengan *feature* pemeriksaan laboratorium. Hal ini semakin membuktikan bahwa dalam pendiagnosisan penyakit dalam sistem urinari, pemeriksaan laboratorium masih sangat dibutuhkan dalam menentukan hasil diagnosis suatu penyakit. Pada proses klasifikasi tanpa *feature* pemeriksaan laboratorium, tingkat akurasi rata-rata yang diperoleh sebesar 77,38%, sedangkan tingkat akurasi rata-rata yang diperoleh dari proses klasifikasi dengan *feature* pemeriksaan laboratorium sebesar 86,31%.

Untuk penelitian selanjutnya, maka pada diagnosis gangguan sistem urinari, teknik yang disajikan perlu dimodifikasi dengan memberikan bobot yang berbeda untuk tiap *feature* dan menambahkan *feature* pemeriksaan laboratorium yang lainnya. Selain itu dapat ditambahkan juga *threshold* untuk mengantisipasi kemungkinan seekor hewan terkena gangguan ginjal dan non-ginjal secara bersamaan.

REFERENSI

- [1] Blood DC dan Studdert VP., *Saunders Comprehensive Veterinary Dictionary* Second Edition. Philadelphia: WB Saunders Company, 1999.
- [2] S. Foster, "Kidney Disease: Causes, Signs, and Diagnosis". <http://www.peteducation.com/article.cfm?cls=2&cat=1634&articleid=350>, [diakses pada Agustus 2007], 2007
- [3] Aieolo SE. *et al.*, *The Merck Veterinary Manual Eight Edition*. USA: Merck & CO, Inc. White House Station, 2000.
- [4] Han J dan Kamber M. *Data Mining Concepts & Techniques*. USA : Academic Press, 2001.
- [5] Güvenir HA, Demiröz G, dan İltter N. "Learning Differential Diagnosis of Erythematous-Squamous Diseases using Voting Feature Intervals". *Artificial Intelligence in Medicine*, 13 (3), 1998, pp. 147-165.
- [6] Sarle W. *What are cross-validation and bootstrapping?*. <http://www.faqs.org/ai-faq/neural-nets/part3/section-12.html> [diakses pada Maret 2006], 2004.
- [7] Liming Fu, *Neural Network in Computers Intelligence*. Singapura: McGraw-Hill, 1994.
- [8] Güvenir HA dan Emeksiz N., *An Expert System for the Differential Diagnosis of Erythematous*

- Squamous Diseases*. Expert Systems with Applications, 18(1), 2000, pp. 43-49.
- [9] Demiröz G., *Non-Incremental Classification Learning Algorithms Based on Voting Feature Intervals*. <http://www.cs.bilkent.edu.tr/tech-reports/1997/BU-CEIS-9715.ps.gz>, [diakses 2007], 1997.