PENGARUH *INCOMPLETE DATA* TERHADAP AKURASI *VOTING FEATURE INTERVALS*-5 (VFI5)

Atik Pawestri Sulistyo[1], Aziz Kustiyo [1] , Agus Buono [2]
[1] Departemen Ilmu Komputer, Fakultas Matematika dan IPA
Institut Pertanian Bogor

ABSTRAK

Permasalahan mengenai data hilang merupakan masalah umum yang terjadi pada lingkungan medis. Data hilang dapat disebabkan karena beberapa hal yaitu salah memasukkan data, datanya tidak valid dan peralatan yang digunakan untuk mengambil data tidak berfungsi dengan baik.

Voting Feature Intervals merupakan algoritma klasifikasi yang dikembangkan oleh Gülşen Demiroz dan H. Altay Güvenir pada tahun 1997. Algoritma ini dapat mengatasi data hilang dengan mengabaikan data hilang tersebut. Pada penelitian ini dilakukan penerapan algoritma Voting Feature Intervals-5 (VFI5) sebagai algoritma klasifikasi pada kasus data hilang. Data yang digunakan adalah data ordinal (data Dermatology) dan data interval (data Ionosphere). Untuk mengatasi data hilang digunakan tiga metode yaitu mengabaikan data hilang, menghapus satu baris data hilang dan mengganti data hilang dengan mean atau modus.

Pengaruh data hilang terhadap tingkat akurasi adalah tingkat akurasi algoritma VFI5 mengalami penurunan dengan semakin banyaknya jumlah data yang hilang dan semakin banyaknya jumlah feature yang memiliki data hilang. Rata-rata tingkat akurasi data ordinal tertinggi sebesar 93.81% dan rata-rata tingkat akurasi data interval tertinggi sebesar 79.89%. Hasil penelitian menunjukkan rata-rata tingkat akurasi yang tertinggi dicapai ketika data hilang diatasi dengan mengganti data hilang dengan mean atau modus.

Kata Kunci*: incomplete data, voting feature intervals.*

PENDAHULUAN

Latar belakang

Permasalahan mengenai data hilang merupakan masalah umum yang terjadi pada lingkungan medis. Data hilang dapat disebabkan karena beberapa hal yaitu salah memasukkan data, datanya tidak valid dan peralatan yang digunakan untuk mengambil data tidak berfungsi dengan baik (Shyu, Chen dan Chang 2005).

Data hilang dapat menyebabkan berbagai masalah. Jumlah data hilang yang semakin banyak akan mempengaruhi tingkat akurasi *classifier* yang dihasilkan menggunakan algoritma VFI5 atau menyebabkan kesalahan klasifikasi menjadi semakin banyak.

Penelitian mengenai data hilang pernah dilakukan sebelumnya dengan menggunakan BP-ANN oleh Markey dan Patel pada tahun 2004. Berdasarkan penelitian tersebut dapat diketahui bahwa pengaruh data hilang pada data pengujian lebih tinggi daripada data pelatihan (Markey 2004).

Algoritma klasifikasi VFI5 merupakan suatu algoritma yang merepresentasikan deskripsi sebuah konsep oleh sekumpulan interval nilai-nilai *feature* atau atribut. Algoritma VFI5 memiliki tingkat akurasi yang lebih tinggi bila dibandingkan dengan algoritma *nearest-neighbor*. Kedua algoritma ini telah diuji dengan menambahkan *feature* yang tidak relevan. Ketika *feature* tidak relevan ditambahkan, akurasi dari algoritma VFI5 memperlihatkan jumlah pengurangan akurasi yang sangat kecil (Guvenir 1998).

Penerapan algoritma VFI5 sebagai algoritma klasifikasi diharapkan dapat mengatasi data yang tidak lengkap tersebut.

Tujuan

Tujuan dari penelitian ini adalah untuk mengetahui pengaruh data tidak lengkap terhadap akurasi *classifier* yang dihasilkan menggunakan algoritma klasifikasi VFI5.

Ruang lingkup

Ruang lingkup dari penelitian ini yaitu:

1. Bobot (*weight*) setiap *feature* pada semua data adalah seragam.

2. Data yang digunakan adalah data interval (*IonosphereData)* dan data ordinal (*DermatologyData)*.
3. Metode yang digunakan untuk mengatasi data tidak lengkap adalah dengan mengabaikan data tidak lengkap tersebut, menghapus satu baris data tidak lengkap dan mengganti data tidak lengkap dengan *mean* atau *modus.*

Manfaat

Penelitian ini diharapkan dapat memberikan informasi mengenai akurasi *classifier* yang dihasilkan menggunakan algoritma klasifikasi *voting feature intervals* pada pengklasifikasian data yang memiliki data tidak lengkap.

TINJAUAN PUSTAKA

K-Fold Cross Validation

Sebelum digunakan, sebuah sistem berbasis komputer harus dievaluasi dalam berbagai aspek. Di antara aspek-aspek tersebut, validasi kerja bisa menjadi yang paling penting.

Cross validation dan *bootstrapping* merupakan metode untuk memperkirakan error generalisasi berdasarkan "*resampling*" (Weiss and Kulikowski, 1991; Efron and Tibshirani, 1993; Hjorth, 1994; Plutowski, Sakata and White, 1994; Shao and Tu, 1995 diacu dalam Sarle 2004).

Dalam *K-Fold Cross Validation*, himpunan contoh dibagi ke dalam k himpunan bagian secara acak. Pengulangan dilakukan sebanyak k kali dan pada setiap ulangan disisakan satu *subset* untuk pengujian dan *subset-subset* lainnya untuk pelatihan

Pada metode tersebut, data awal dibagi menjadi k subset atau '*fold*' yang saling bebas secara acak, yaitu $S_1,S_2,...S_k$, dengan ukuran setiap subset kira-kira sama. Pelatihan dan pengujian dilakukan sebanyak k kali. Pada iterasi ke-i, subset S_i diperlakukan sebagai data pengujian dan subset lainnya diperlakukan sebagai data pelatihan. Pada iterasi pertama $S_2,...S_k$ menjadi data pelatihan dan S_1 menjadi data pengujian, pada iterasi kedua $S_1,S_3,...S_k$ menjadi data pelatihan dan S_2 menjadi data pengujian, dan seterusnya.

Algoritma *Voting Feature Intervals* (VFI5)

Voting Feature Intervals adalah salah satu algoritma yang digunakan dalam pengklasifikasian data. Algoritma tersebut dikembangkan oleh Gülşen Demiroz dan H. Altay Güvenir pada tahun 1997 (Demiroz dan Guvenir 1997).

Algoritma klasifikasi VFI5 merepresentasikan deskripsi sebuah konsep oleh sekumpulan interval nilai-nilai *feature* atau atribut. Pengklasifikasian *instance* baru berdasarkan *voting* pada klasifikasi yang dibuat oleh nilai tiap-tiap *feature* secara terpisah. VFI5 merupakan algoritma klasifikasi yang bersifat *non-incremental* dan *supervised* (Demiroz dan Guvenir 1997). Algortima VFI5 membuat interval yang berupa *range* atau *point interval* untuk setiap *feature*. *Point interval* terdiri atas seluruh *end point* secara berturut-turut. *Range interval* terdiri atas nilai-nilai antara 2 *end point* yang berdekatan namun tidak termasuk kedua *end point* tersebut.

Keunggulan algoritma VFI5 adalah algoritma ini cukup kokoh (*robust*) terhadap *feature* yang tidak relevan namun mampu memberikan hasil yang baik pada *real-world datasets* yang ada. VFI5 mampu menghilangkan pengaruh yang kurang menguntungkan dari *feature* yang tidak relevan dengan mekanisme *voting*-nya (Guvenir 1998).

Algoritma VFI5 terdiri dari 2 tahap yaitu

1 Pelatihan

Tahap pertama dari proses pelatihan adalah menemukan *end points* setiap *feature* f pada kelas data c. *End points* untuk *feature linear* adalah nilai minimum dan maksimum dari suatu *feature*. Sedangkan *end points* untuk *feature* nominal adalah semua nilai yang berbeda yang ada pada *feature* kelas yang sedang diamati. *End points* untuk setiap *feature* f akan dimasukkan ke dalam *array EndPoints[f]*. Jika *feature* adalah *feature* linier maka akan dibentuk dua interval yaitu *point interval* yang terdiri dari semua nilai *end point* yang diperoleh dan *range interval* yang terdiri dari nilai-nilai di antara dua *end point* yang berdekatan dan tidak termasuk *end points* tersebut. Jika *feature* adalah *feature* nominal maka akan dibentuk *point interval* saja.

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

Selanjutnya, jumlah *instance* pelatihan setiap kelas c dengan *feature* f untuk setiap interval dihitung dan direpresentasikan sebagai *interval_count [f, i, c]*. Untuk setiap *instance* pelatihan, dicari interval i dimana nilai *feature* f dari *instance* pelatihan e (e_f) tersebut jatuh. Jika interval i adalah *point interval* dan nilai e_f sama dengan batas bawah interval tersebut (sama dengan batas atas *point interval*) maka jumlah kelas *instance* pada interval i ditambah dengan 1. Jika interval i adalah *range interval* dan nilai e_f jatuh pada interval tersebut maka jumlah kelas instance e_f pada interval i ditambah 0.5. Hasil proses tersebut merupakan jumlah *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.

2 Klasifikasi

Tahap klasifikasi diawali dengan inisialisasi *vote* untuk setiap kelas dengan nilai nol. Untuk setiap *feature* f, dicari interval i dimana e_f jatuh, dengan e_f adalah nilai *feature* f untuk *instance* tes e. Jika nilai e_f tidak diketahui (hilang) maka *feature* tersebut tidak diikutsertakan dalam proses klasifikasi. Oleh karena itu, *feature* yang memiliki nilai tidak diketahui diabaikan.

Jika nilai e_f diketahui maka interval tersebut ditemukan. Interval tersebut dapat menyimpan *instance* pelatihan dalam beberapa kelas. Kelas-kelas dalam sebuah interval direpresentasikan dengan *vote* kelas-kelas tersebut pada interval tersebut. Untuk setiap kelas c, *feature* f memberikan *vote* yang sama dengan *interval_vote[f, i, c]*. *Interval_vote [f, i, c]* merupakan *vote feature* f yang diberikan untuk kelas c.

Setiap *feature* f mengumpulkan *vote*-nya ke dalam *vector* <$vote_{f,1}$,….,$vote_{f,k}$> kemudian dijumlahkan untuk mendapatkan total *vote vector* <$vote_1$,....,$vote_k$>. Kelas dengan jumlah *vote* paling tinggi akan diprediksi sebagai kelas dari *instance* tes e.

Pseudocode algoritma pelatihan dan klasifikasi VFI5 disajikan pada Gambar 1 dan Gambar 2.

```
train(Training Set);
begin
    for each feature f
        for each class c
            EndPoints[f] = EndPoints[f] U find_end_points(TrainingSet, f, c);
        sort(EndPoints[f]);
        If f is linear
            for each end point p in EndPoints[f]
                form a poin interval from end point p
                form a range interval between p and the next endpoint ≠ p
        else /*f is nominal*/
            each distinct point in EndPoints[f] forms a point interval
        for each interval I on feature dimension f
            for each class c
                interval_count[f, I, c] = 0
        count_instances(f, TrainingSet);
        for each interval I on feature dimension f
            for each class c
                interval_vote[f, I, c] = interval_count[f, I, c]/class_count[c]
                normalize interval_vote[f, i, c]
                /*such that ∑c interval_vote[f, I, c] = 1*/
end
```

Gambar 1 Algortima pelatihan VFI5

```
classify(e); /*e:example to be classified*/
begin
        for each class c
                vote[c] = 0

        for each feature f
                for each class c
                        feature_vote[f, c] = 0 /*vote of feature f for class c*/

                if ef value is known
                        i=find_interval(f, ef)

                        for each class c
                                feature_vote[f, c] = interval_vote[f, I, c]
                                vote[c] = vote[c] + feature_vote[f, c] * weight[f];

        return the class c with highest vote[c];
end
```

Gambar 2 Algoritma klasifikasi VFI5

Incomplete Data

Ada beberapa metode untuk mengatasi data tidak lengkap. Cara yang paling mudah untuk mengatasi data tidak lengkap adalah dengan menghapus satu baris data yang tidak lengkap. Teknik ini terkadang menyebabkan hilangnya informasi yang potensial. Pendekatan yang kedua adalah dengan mengganti semua data hilang dengan rataannya (Ennett 2001).

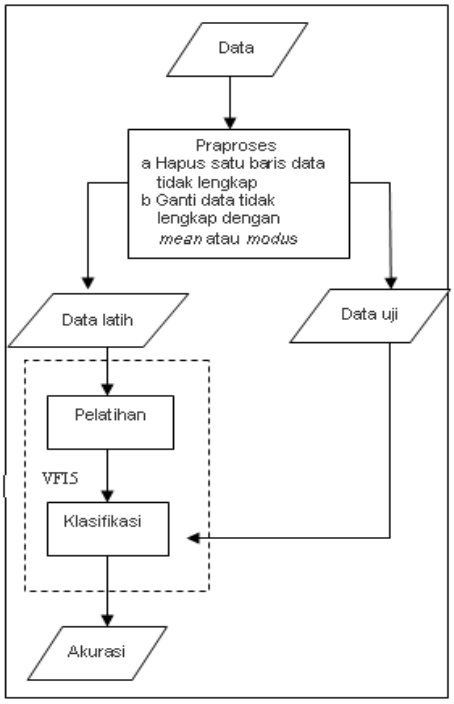
Suatu data terdiri dari nilai nominal dan nilai numerik. Salah satu teknik untuk mengatasi data hilang pada nilai nominal adalah mengganti data hilang dengan *modus* sedangkan untuk nilai numerik adalah mengganti data hilang dengan *mean* (Shyu, Chen dan Chang 2005).

METODE PENELITIAN

Ada beberapa tahapan proses yang dilakukan untuk mengetahui pengaruh data hilang terhadap kinerja algoritma VFI5. Tahapan-tahapan tersebut disajikan pada Gambar 3.

Data

Data yang digunakan dalam penelitian ini adalah data yang diambil dari *UCI Repository Of Machine Learning Databases,* dari ics.uci.edu. Data tersebut adalah data *Dermatology* sebagai data ordinal dan data *Ionosphere* sebagai data interval. Data *Dermatology* terdiri atas enam kelas, 366 *instances* dan 34 *attributes*. Jumlah data hilangnya sebanyak 8 *instances* dan atribut yang memiliki data hilang hanya atribut ke-34 yaitu atribut umur. Oleh karena itu, data *dermatology* yang digunakan ada 358 *instances*. Data *Ionosphere* terdiri atas dua kelas, 351 *instances* dan 34 *attributes*. Data tersebut merupakan data lengkap atau tidak memiliki data hilang.



Gambar 3 Tahapan proses klasifikasi data.

Tabel 1 Spesifikasi data yang akan digunakan

Data	Atribut Ordinal	Atribut Interval
Dermatology	33	1
Ionosphere	0	34

Praproses

Pada tahap ini data dihilangkan secara acak dengan persentase data hilangnya yaitu 2%, 5%, 10% dan 20%. Persentase data hilang tersebut merupakan persentase dari jumlah keseluruhan data. Sedangkan persentase untuk *feature* yang memiliki data hilang adalah 25%-75% dan 50%-50%. Nilai 25%-75% berarti terdapat 25% *feature* yang memiliki data hilang dan 75% *feature* lainnya lengkap. Nilai 50%-50% berarti terdapat 50% *feature* yang memiliki data hilang dan 50% *feature* lainnya lengkap.

Metode yang digunakan untuk mengatasi data hilang yaitu mengabaikan data hilang tersebut, menghapus satu baris data yang memiliki data hilang dan mengganti semua data hilang dengan *mean* untuk data interval dan *modus* untuk data ordinal.

Data Latih dan Data Uji

Pada tahapan ini dilakukan proses *3-fold cross validation* yaitu membagi data menjadi 3 bagian. Pembagian data tersebut dilakukan secara acak dengan mempertahankan perbandingan jumlah *instance* setiap kelas. Data tersebut akan digunakan sebagai data latih dan data uji.

Algoritma VFI5

Pada penelitian ini digunakan algoritma VFI5 dengan bobot setiap *feature* diasumsikan seragam yaitu satu. Tahapan ini terdiri atas dua proses yaitu pelatihan dan prediksi (klasifikasi) kelas *instance* baru.

Pada tahap pelatihan, input dari algoritma klasifikasi VFI5 adalah data yang telah dibagi-bagi menjadi beberapa *subset*. Selanjutnya akan dibentuk 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 itu dilakukan penghitungan jumlah *instance* setiap kelas yang berada pada setiap interval tersebut.

Pada tahap klasifikasi, setiap nilai *feature* dari suatu *instance* baru, diperiksa letak interval dari 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 akan menjadi kelas prediksi *instance* baru tersebut.

Menghitung tingkat akurasi

Pada tahapan ini dilakukan proses penghitungan tingkat akurasi. Tingkat akurasi diperoleh dengan perhitungan:

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

Spesifikasi aplikasi

Aplikasi ini dirancang dan dibangun dengan perangkat keras dan perangkat lunak sebagai berikut:

Perangkat keras

a *Processor* Intel Pentium 4
b Memori 512 MB
c *Harddisk* 40 GB
d *Mouse* dan *keyboard*

Perangkat lunak

a Windows XP sebagai Sistem Operasi
b Matlab 7.0.1

HASIL DAN PEMBAHASAN

Data yang digunakan pada penelitian ini adalah data *Ionosphere* (data interval) dan data *Dermatology* (data ordinal). Persentase *feature* yang memiliki data hilang adalah 25%-75% dan 50%-50%.

Tabel 2 Jumlah *feature* yang memiliki data hilang pada persentase 25%-75% dan 50%-50%

Data	25%-75%	50%-50%
Ionosphere	8 *feature*	17 *feature*
Dermatology	8 *feature*	17 *feature*

Berdasarkan Tabel 2 dapat dilihat bahwa jumlah *feature* yang memiliki data hilang untuk data *Ionosphere* dan data *Dermatology* pada persentase 25%-75% adalah 8 *feature,* sedangkan pada persentase 50%-50% jumlah *feature* yang memiliki data hilang adalah 17 *feature.* Jumlah *instances* yang memiliki data hilang, secara lengkap dapat dilihat pada Lampiran 1.

Akurasi *classifier* yang dibuat menggunakan Data Interval

Data interval yang digunakan pada penelitian ini adalah data *Ionosphere*. Hasil

sampling pada data *Ionosphere* dapat dilihat pada Lampiran 2.

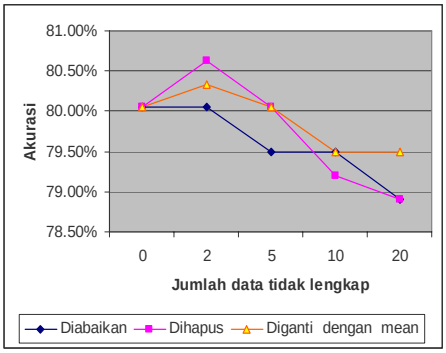
Tabel 3 Akurasi algoritma VFI5 untuk persentase 25% - 75 % (25% *feature* memiliki data hilang) pada data interval

Jumlah data tidak lengkap	Diabaikan	Dihapus	Diganti dengan *mean*
0%	80.06 %	80.06 %	80.06 %
2%	80.06 %	80.63 %	80.34 %
5%	79.49 %	80.06 %	80.06%
10%	79.49 %	79.20 %	79.49 %
20%	78.91 %	78.91 %	79.49 %
Rata-rata	79.60 %	79.77 %	79.89 %

Pengaruh data hilang pada data interval adalah tingkat akurasinya cenderung semakin menurun dengan semakin banyaknya jumlah data yang hilang dan semakin banyaknya jumlah *feature* yang memiliki data hilang.

Tingkat akurasi data interval ketika datanya lengkap adalah 80.06%. Berdasarkan Tabel 3 dapat dilihat bahwa tingkat akurasi ketika data hilang diabaikan cenderung menurun, meskipun saat data hilangnya 2% tingkat akurasinya sama dengan ketika datanya lengkap. Tingkat akurasi ketika data hilangnya dihapus satu baris dan diganti juga cenderung mengalami penurunan.

Dalam penelitian ini digunakan tiga metode untuk mengatasi data hilang. Pada metode yang pertama yaitu diabaikan, dapat dilihat bahwa tingkat akurasinya mengalami penurunan ketika jumlah data hilangnya semakin bertambah (Tabel 3). Tingkat akurasi mencapai 78.91% ketika data hilangnya 20%. Hal ini disebabkan karena *feature-feature* yang memiliki data hilang tidak memberikan *vote*-nya (memberikan *vote* nol). Pada metode yang kedua yaitu dihapus satu baris, tingkat akurasinya mengalami penurunan ketika persentase data hilangnya semakin besar. Tingkat akurasinya mencapai 78.91%. Hal ini disebabkan karena dengan menghapus satu baris, maka jumlah *instances* data akan semakin berkurang sehingga interval yang dibuat juga berbeda dengan data aslinya. Perbandingan jumlah *instances*nya dapat dilihat pada Lampiran 2. Pada metode yang ketiga yaitu diganti dengan *mean,* tingkat akurasinya mengalami penurunan. Hal ini disebabkan karena dengan diganti *mean,* suatu *feature* memberikan nilai *vote* yang lebih kecil daripada ketika datanya lengkap. Grafik tingkat akurasi *classifier* yang dibuat menggunakan data interval terhadap jumlah data hilang untuk 25%-75% dapat dilihat pada Gambar 4.

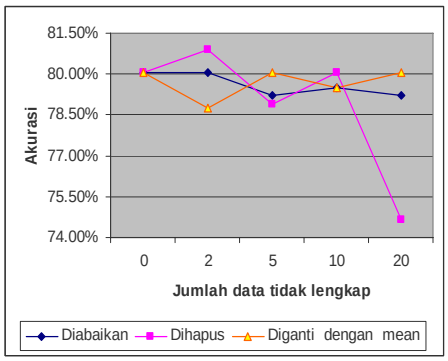


Gambar 4 Tingkat akurasi *classifier* yang dibuat menggunakan VFI5 pada data interval terhadap jumlah data hilang untuk 25%-75%

Tabel 4 Akurasi algoritma VFI5 untuk persentase 50% - 50 % (50% *feature* memiliki data hilang) pada data interval

Jumlah data tidak lengkap	Diabaikan	Dihapus	Diganti dengan *mean*
0%	80.06 %	80.06 %	80.06 %
2%	80.06 %	80.91 %	78.77 %
5%	79.20 %	78.91 %	80.06 %
10%	79.49 %	80.06 %	79.48 %
20%	79.21 %	74.65 %	80.06 %
Rata-rata	79.60 %	78.92 %	79.69 %

Berdasarkan Tabel 4 dapat dilihat bahwa dengan menggunakan metode yang pertama yaitu diabaikan, tingkat akurasinya mengalami penurunan ketika jumlah data hilangnya semakin bertambah. Tingkat akurasi mencapai 79.21% ketika data hilangnya 20%. Pada metode yang kedua yaitu dihapus satu baris, tingkat akurasinya cenderung mengalami penurunan meskipun ketika jumlah data hilangnya 10%, tingkat akurasinya mengalami peningkatan. Pada metode yang ketiga yaitu diganti dengan *mean,* tingkat akurasinya mengalami penurunan dan peningkatan. Tingkat akurasi dengan metode tersebut masih kurang stabil. Grafik tingkat akurasi *classifier* yang dibuat menggunakan data interval terhadap jumlah data hilang untuk 50%-50% dapat dilihat pada Gambar 5.



Gambar 5 Tingkat akurasi *classifier* yang dibuat menggunakan VFI5 pada data interval terhadap jumlah data hilang untuk 50%-50%

Pada data interval tingkat akurasi tertinggi adalah 80.91% dan tingkat akurasi terendah adalah 74.65%. Rata-rata tingkat akurasi tertinggi dicapai dengan metode mengganti data hilang dengan *mean.*

Tingkat akurasi dengan persentase *feature* yang memiliki data hilang 25%-75% dan 50%-50% cenderung mengalami penurunan ketika data hilangnya diabaikan, dihapus dan diganti dengan *mean*. Tingkat akurasi pada data interval (data *ionosphere*) secara lengkap dapat dilihat pada Lampiran 3.

Akurasi *classifier* yang dibuat menggunakan Data Ordinal

Data ordinal yang digunakan adalah data *Dermatology*. Hasil *sampling* pada data *Dermatology* dapat dilihat pada Lampiran 4.

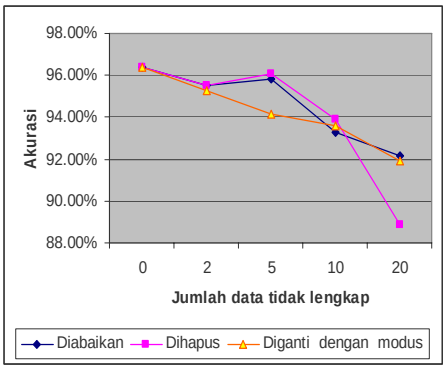
Data *Dermatology* terdiri atas enam kelas, tetapi hanya dua kelas yang sering mengalami kesalahan klasifikasi yaitu kelas 2 dan kelas 4.

Tabel 5 Akurasi algoritma VFI5 untuk persentase 25% - 75 % (25% *feature* memiliki data hilang) pada data ordinal

Jumlah data tidak lengkap	Diabaikan	Dihapus	Diganti dengan modus
0%	96.38 %	96.38%	96.38%
2%	95.54 %	95.54%	95.26%
5%	95.82 %	96.09%	94.15%
10%	93.31 %	93.88%	93.59%
20%	92.19 %	88.85%	91.90%
Rata-rata	94.65 %	94.15%	94.26%

Pengaruh data hilang pada data ordinal adalah tingkat akurasinya cenderung semakin menurun dengan semakin banyaknya jumlah data yang hilang dan semakin banyaknya jumlah *feature* yang memiliki data hilang.

Tingkat akurasi data ordinal adalah 96.38% ketika datanya lengkap. Berdasarkan Tabel 5 dapat dilihat bahwa ketika data hilang diabaikan, dihapus dan diganti dengan *modus,* tingkat akurasinya lebih kecil daripada ketika datanya lengkap. Tingkat akurasi dengan ketiga metode tersebut semakin menurun dengan semakin banyaknya jumlah data yang hilang.



Gambar 6 Tingkat akurasi *classifier* yang dibuat menggunakan VFI5 pada data ordinal terhadap jumlah data hilang untuk 25%-75%

Pada metode yang pertama yaitu diabaikan, tingkat akurasinya mengalami penurunan (Tabel 5). Akan tetapi, ketika persentase data hilangnya 5%, tingkat akurasinya mengalami kenaikan. Tingkat akurasinya mencapai 95.82%. Pada metode yang kedua yaitu menghapus satu baris data hilang, tingkat akurasinya juga cenderung mengalami penurunan. Tingkat akurasinya mencapai 88.85% ketika persentase data hilangnya 20%. Tingkat akurasi tersebut merupakan tingkat akurasi terendah pada data ordinal. Pada metode ketiga yaitu diganti dengan *modus*, tingkat akurasinya cenderung mengalami penurunan. Grafik tingkat akurasi data ordinal terhadap jumlah data hilang untuk persentase 25%-75% dapat dilihat pada Gambar 6

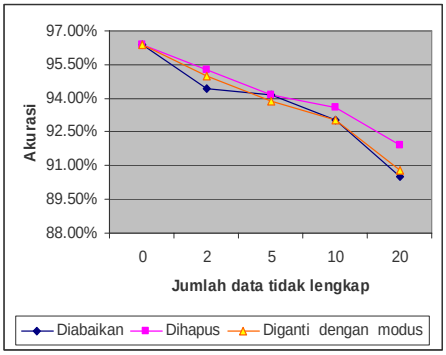
Tabel 6 Akurasi algoritma VFI5 untuk persentase 50% - 50 % (50% *feature* memiliki data hilang) pada data ordinal

Jumlah data tidak lengkap	Diabaikan	Dihapus	Diganti dengan modus
0%	96.38 %	96.38%	96.38%
2%	94.43 %	95.26%	94.99%
5%	94.14 %	94.15%	93.87%
10%	93.03 %	93.59%	93.03%
20%	90.52 %	91.90%	90.80%
Rata-rata	93.70 %	94.26%	93.81%

Berdasarkan Tabel 6 dapat dilihat bahwa tingkat akurasi ketika data hilang diabaikan mengalami penurunan karena *feature-feature* yang memiliki data hilang tidak memberikan *vote*-nya (memberikan *vote* nol). Hal ini menyebabkan kesalahan klasifikasi menjadi semakin banyak. Pada metode kedua yaitu menghapus satu baris data hilang, tingkat akurasinya mengalami kenaikan dan juga penurunan. Peningkatan atau penurunan tingkat akurasi disebabkan karena penempatan data hilangnya kurang tepat. Tingkat akurasi dengan metode tersebut masih kurang stabil. Pada metode ketiga yaitu diganti dengan *modus*, tingkat akurasinya cenderung mengalami penurunan. Grafik tingkat akurasi *classifier* yang dibuat menggunakan data ordinal terhadap jumlah data hilang untuk persentase 50%-50% dapat dilihat pada Gambar 7.

Tingkat akurasi tertinggi yang dicapai pada data ordinal adalah 96.38% yaitu ketika datanya lengkap sedangkan tingkat akurasi terendah adalah 88.85%. Untuk persentase 25%-75%, rata-rata tingkat akurasi tertinggi dicapai dengan metode diabaikan sedangkan pada persentase 50%-50% rata-rata tingkat akurasi tertinggi dicapai dengan metode diganti.

Tingkat akurasi dengan persentase *feature* yang memiliki data hilang 25%-75% dan 50%-50% cenderung mengalami penurunan ketika data hilangnya diabaikan, dihapus dan diganti dengan *modus*. Tingkat akurasi pada data ordinal (data *dermatology*) secara lengkap dapat dilihat pada Lampiran 5.



Gambar 7 Tingkat akurasi *classifier* yang dibuat menggunakan VFI5 pada data ordinal terhadap jumlah data hilang untuk 25%-75%

Perbandingan akurasi *classifier* yang dibuat menggunakan Data Ordinal dan Data Interval

Pada data ordinal dan interval, ketika data hilang diabaikan, dihapus dan diganti dengan *mean* atau *modus*, tingkat akurasinya cenderung mengalami penurunan. Rata-rata tingkat akurasi tertinggi dicapai dengan mengganti data hilang dengan *mean* atau *modus* untuk mengatasi data hilang. Perbandingan akurasi antara data ordinal dan data interval secara lengkap dapat dilihat pada Lampiran 6.

KESIMPULAN DAN SARAN

Kesimpulan

Pada data interval terjadi penurunan tingkat akurasi dengan semakin banyaknya jumlah data yang hilang. Tingkat akurasi tertinggi adalah 80.91% dan tingkat akurasi terendah adalah 74.65%.

Pada data ordinal terjadi penurunan tingkat akurasi dengan semakin banyaknya jumlah data yang hilang. Tingkat akurasi tertinggi adalah 96.38% dan tingkat akurasi terendah adalah 88.85%.

Rata-rata tingkat akurasi tertinggi dari algoritma tersebut dicapai dengan mengganti data hilang dengan *mean* atau *modus* untuk mengatasi data hilang. Untuk data ordinal rata-rata tingkat akurasi mencapai 93.81% sedangkan data interval rata-rata tingkat akurasi yang dicapai sebesar 79.89%.

Algoritma VFI5 mampu mengatasi data hilang dengan mengabaikan data hilang tersebut, tetapi tingkat akurasi algoritma tersebut mengalami penurunan dengan

semakin banyaknya jumlah data yang hilang. Tingkat akurasi pada data ordinal ketika jumlah data hilangnya 20% menurun sebanyak 4.19% pada persentase 25%-75% dan 5.86% pada persentase 50%-50%. Tingkat akurasi pada data interval menurun sebanyak 1.15% pada persentase 25%-75% dan 0.85% pada persentase 50%-50%.

Saran

Klasifikasi data hilang menggunakan algoritma VFI5 dapat dikembangkan dengan mengolah data yang atributnya adalah atribut nominal atau atributnya merupakan gabungan dari atribut nominal dan atribut interval.

Penelitian ini masih menggunakan bobot *feature* yang seragam. Hal ini dapat dikembangkan lebih lanjut dengan menggunakan bobot yang berbeda untuk setiap *feature*.

DAFTAR PUSTAKA

Demiröz G dan Güvenir HA. 1997. *Classification by Voting Feature Intervals.* http://www.cs.ucf.edu/~ecl/papers/demiros97classification.pdf. [November 2006].

Ennett CM, Frize M, Walker CR. 2001. *Influence of Missing Values on Artificial Neural Network Performance*. Amsterdam : IOS Press.

Guvenir HA. 1998. *A Classification Learning Algorithm Robust to Irrelevant Features.* http://www.cs.bilkent.edu.tr /tech-reports/1998/BU-CEIS-9810.ps.gz [November 2006].

Markey MK, Patel A. 2004. *Impact of Missing Data in Training Artificial Neural Network for Computer-Aided Diagnosis.* Computers in Biology and Medicine.

Sarle W. 2004. *What are cross-validation and bootstrapping?*. http://www.faqs.org/faqs/ai-faq/neural-nets/part3/section-12.html. [November 2006].

Shyu Mei-Ling, Chen Shyu-Ching, Chang LiWu. 2005. *Handling Missing Values Via Decomposition of the Conditioned Set*. Department of Electrical and Computer Engineering, University of Miami.