

Klasifikasi Resiko Kehamilan Menggunakan *Ensemble Learning* berbasis *Classification Tree*

Muhamad Arief Hidayat*

* Fakultas Ilmu Komputer, Universitas Jember

*arief.hidayat@unej.ac.id

ABSTRACT

In health science there is a technique to determine the level of risk of pregnancy, namely the Poedji Rochyati score technique. In this evaluation technique, the level of pregnancy risk is calculated from the values of 22 parameters obtained from pregnant women. Under certain conditions, some parameter values are unknown. This causes the level of risk of pregnancy can not be calculated. For that we need a way to predict pregnancy risk status in cases of incomplete attribute values. There are several studies that try to overcome this problem. The research "classification of pregnancy risk using cost sensitive learning" [5] applies cost sensitive learning to the process of classifying the level of pregnancy risk. In this study, the best classification accuracy achieved was 73% and the best value was 77.9%. To increase the accuracy and recall of predicting pregnancy risk status, in this study several improvements were proposed. 1) Using ensemble learning based on classification tree 2) using the SVMAttributeEvaluator evaluator to optimize the feature subset selection stage. In the trials conducted using the classification tree-based ensemble learning method and the SVMAttributeEvaluator at the feature subset selection stage, the best value for accuracy was up to 76% and the best value for recall was up to 89.5%

Keyword: classification, Poedji Rochyati score technique, feature subset selection, classification tree, ensemble learning

1. Pendahuluan

Kehamilan merupakan suatu proses yang harus dilalui oleh wanita untuk melakukan regenerasi. Pada proses kehamilan, kesehatan ibu dan janin harus dijaga agar kehamilan dan persalinan berjalan dengan baik dan aman untuk ibu dan janinnya. Namun tidak semua kehamilan berjalan dengan lancar. Ada kehamilan yang beresiko bagi ibu maupun janin yang dikandungnya [8]. Terdapat beberapa faktor yang menyebabkan terjadinya kehamilan beresiko. Dalam ilmu kesehatan ada teknik untuk mengevaluasi kehamilan untuk mengetahui tingkat resiko kehamilan yaitu teknik skor Poedji Rochyati [12]. Pada teknik evaluasi ini, tingkat resiko kehamilan terbagi menjadi tiga yaitu : kehamilan resiko rendah (KRR), kehamilan resiko tinggi (KRT), dan kehamilan resiko sangat tinggi (KRST). Status tersebut dihitung dari nilai 22 parameter yang didapatkan dari ibu hamil. Dengan mengetahui nilai totalnya, tingkat resiko yang dialami oleh seorang ibu hamil bisa diketahui.

Teknik skor Poedji Rochyati ini tidak bisa bekerja dengan baik jika data atribut yang dimiliki kurang lengkap. Terdapat beberapa atribut yang pada kondisi tertentu nilai sulit diketahui oleh bidan. beberapa atribut yang nilainya sulit diketahui antara lain : anemia, payah jantung, diabetes, lebih bulan, dan kejang. Pada kasus di mana ada satu atau beberapa atribut tidak diketahui nilainya, maka teknik skor Poedji Rochyati tidak bisa digunakan untuk menentukan status resiko kehamilan.

Untuk itu diperlukan cara untuk melakukan prediksi status resiko kehamilan pada kasus nilai atribut tidak lengkap. Terdapat penelitian yang menggunakan teknik klasifikasi untuk memprediksi status resiko kehamilan Poedji Rochyati menggunakan metode Naïve Bayes [3]. Pada penelitian ini nilai akurasi yang didapat adalah 56,967% dengan tingkat *error* sebesar 43,043%. Atribut yang digunakan adalah atribut pada perhitungan skor Poedji Rochyati. Pada penelitian ini nilai akurasi yang didapat relatif rendah. Selain itu pada penelitian ini pengukuran kualitas *classifier* tidak menggunakan metrik *recall* yang lebih sesuai untuk kasus klasifikasi dengan resiko *misclassification* yang tidak simetris. Pada penelitian ini juga tidak menggunakan *Feature Subset Selection* untuk memilih kombinasi atribut optimal untuk klasifikasi

Penelitian tersebut dilanjutkan pada "klasifikasi resiko kehamilan menggunakan *cost sensitive learning*" [5]. Penelitian ini meningkatkan penelitian sebelumnya dengan menerapkan *cost sensitive learning* pada proses klasifikasi. Tujuan dari penerapan *cost sensitive learning* adalah untuk meningkatkan nilai *recall* dari klasifikasi. Meningkatnya nilai *recall* merupakan indikator kualitas *classifier* dalam aspek menghindari *misclassification* yang kerugiannya fatal (memprediksi KRT dan KRST sebagai KRR). Selain itu, penelitian ini menggunakan *Feature Subset Selection* dengan evaluator information gain dan *search method* ranker untuk meningkatkan hasil klasifikasi.

Penelitian ini berhasil meningkatkan akurasi klasifikasi dari penelitian sebelumnya. Pada penelitian sebelumnya nilai akurasinya adalah 56,967% dan tingkat *error* sebesar 43,043%. Pada klasifikasi resiko kehamilan menggunakan *cost sensitive learning*, nilai akurasinya meningkat menjadi rata-rata 72,04%. Meskipun ada peningkatan yang signifikan, nilai akurasi masih perlu ditingkatkan lagi. Hal ini disebabkan karena kasus prediksi kehamilan merupakan kasus yang jika terjadi *misclassification* tertentu (memprediksi KRT dan KRST sebagai KRR), kerugiannya terhitung besar bagi ibu hamil. Yaitu beresiko terhadap keselamatan ibu hamil dan bayinya.

Selain itu, pada klasifikasi resiko kehamilan menggunakan *cost sensitive learning*, eksperimen menunjukkan tidak ada perbedaan signifikan antara akurasi klasifikasi menggunakan *Feature Subset Selection* dengan tanpa *Feature Subset Selection*. Uji coba menggunakan *Feature Subset Selection*, nilai akurasi yang didapat adalah 72,04%. Sedangkan jika tanpa menggunakan *feature subset* tingkat akurasinya mencapai 71,7%. Nilai akurasinya memiliki selisih yang terlalu kecil. Hal ini bisa disebabkan karena ketidaksesuaian evaluator dan *search method* yang digunakan dengan karakter dataset. Penelitian ini juga menghasilkan nilai *recall* 76,92% jika menggunakan *Feature Subset Selection*. Sedangkan jika tanpa *Feature Subset Selection*, nilai *recall*-nya 77,9%. Hal ini merupakan indikasi tambahan bahwa pemilihan evaluator dan *search method* yang digunakan pada tahap *Feature Subset Selection* kurang sesuai dengan data yang digunakan.

Dari uraian di atas dapat disimpulkan bahwa pada klasifikasi resiko kehamilan masih terdapat beberapa permasalahan. Masalah pertama yaitu bagaimana cara meningkatkan akurasi klasifikasi. Akurasi yang dicapai pada penelitian sebelumnya yang bernilai 72,04%. Nilai ini masih terlalu rendah untuk kasus prediksi resiko kehamilan yang memiliki kerugian besar jika terjadi *misklasifikasi*. Masalah kedua adalah bagaimana meningkatkan nilai *recall* pada klasifikasi resiko kehamilan. Nilai *recall* yang dicapai pada penelitian sebelumnya dengan nilai 76,92% perlu ditingkatkan untuk meningkatkan persentase identifikasi ibu hamil dengan KRT dan KRST.

Untuk menyelesaikan permasalahan tersebut pada penelitian ini diajukan beberapa perbaikan sebagai berikut.

1. Menggunakan *Ensemble Learning*

Pada penelitian-penelitian sebelumnya, hanya digunakan satu *classifier* saja. Pada penelitian ini digunakan teknik *Ensemble Learning* yaitu menggunakan banyak *classifier* sekaligus untuk melakukan klasifikasi. Hasil penelitian tentang *ensemble learning* bisa dilihat di [1, 4, 10, 11]. Hasil akhir klasifikasi ditentukan dengan cara voting oleh semua *classifier* yang dibuat. Dengan menggunakan banyak *classifier* sekaligus, diharapkan nilai akurasi dan *recall* klasifikasi akan naik. *Classifier* dasar yang digunakan dalam *Ensemble Learning* di penelitian ini adalah *Classification Tree* menggunakan algoritma J48. Hasil penelitian tentang algoritma J48 bisa dilihat di [2, 9]

2. Mencari kombinasi evaluator dan *search method* yang memberi hasil optimal pada tahap *Feature Subset Selection*

Pada penelitian sebelumnya di tahap *Feature Subset Selection* digunakan evaluator *information gain* dan *search method ranker*. Hasil yang didapat ternyata tidak menunjukkan peningkatan signifikan pada akurasi. Bahkan nilai *recall* pada eksperimen malah turun. Hal ini mengindikasikan bahwa evaluator dan *search method* yang digunakan kurang sesuai. Untuk itu perlu dicari kombinasi evaluator dan *search method* yang bisa menghasilkan nilai akurasi dan *recall* optimal. Pada penelitian ini, evaluator yang digunakan adalah SVMAttributeEvaluator yang berbasis Support Vector Machine [6, 7]

2. Metode Penelitian

2.1. Resiko Kehamilan

Kehamilan merupakan tahap yang dibutuhkan untuk melakukan regenerasi pada manusia. Untuk mendapatkan keturunan yang berkualitas, selama dalam proses kehamilan banyak faktor yang harus dijaga agar ibu hamil dan anak yang dikandung sehat. Sayangnya tidak semua kehamilan berjalan lancar. Dalam kondisi tertentu, kehamilan bisa beresiko untuk ibu dan anak yang dikandung. Dalam kondisi kehamilan yang beresiko, maka dibutuhkan penanganan khusus agar ibu hamil dan anak tetap sehat.

Karena itu penting untuk mengetahui apakah kehamilan seorang ibu merupakan kehamilan yang beresiko atau tidak. Di Indonesia untuk mengetahui apakah kehamilan seorang ibu beresiko atau tidak digunakan teknik pengukuran skor Poedji Rochyati [12]. Teknik pengukuran ini mengukur resiko kehamilan menggunakan empat faktor resiko dengan beberapa atribut. Masing-masing atribut memiliki skor yang diisi sesuai kondisi ibu hamil. Nilai masing-masing atribut kemudian dijumlah. Total skor yang didapat menentukan resiko kehamilan seorang ibu. Terdapat tiga jenis resiko kehamilan: kehamilan resiko rendah (KRR), kehamilan resiko tinggi (KRT), dan kehamilan resiko sangat tinggi (KRST).

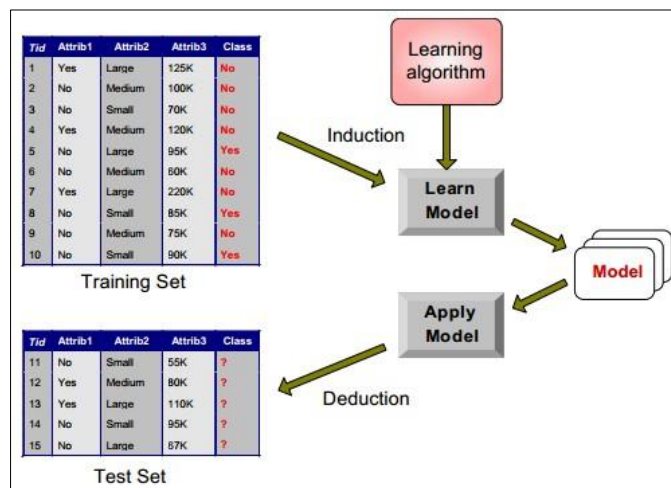
1. Kehamilan Resiko Rendah (KRR)

- Pada kehamilan resiko rendah, tidak ada faktor resiko pada ibu hamil dan bayi pada kandungan. Proses persalinan bisa dilakukan normal dengan kemungkinan besar ibu dan bayi yang dilahirkan dalam keadaan sehat
2. Kehamilan Resiko tinggi (KRT)
 Pada kehamilan resiko tinggi, ada satu faktor resiko pada ibu hamil dan bayi pada kandungan. Hal ini menimbulkan dampak bagi proses persalinan berupa kondisi gawat namun tidak darurat
 3. Kehamilan resiko sangat tinggi (KRST)
 Pada kehamilan resiko sangat tinggi, ada beberapa faktor resiko pada ibu hamil dan bayi pada kandungan. Hal ini menimbulkan pada proses persalinan berupa kondisi gawat dan darurat. Penanganan kelahiran pada kasus KRST harus dilakukan di rumah sakit di bawah penanganan dokter.

2.2 Klasifikasi

Manusia memiliki kemampuan menentukan jenis sebuah obyek berdasarkan ciri-cirinya. Misal jika melihat seekor ayam, manusia bisa mengetahui jenis ayam tersebut : apakah ayam kampung, ayam cemani ataukah ayam potong. Penentuan jenis ayam dilakukan dengan didasarkan pada atribut (ciri – ciri) ayam yang akan ditentukan jenisnya. Berdasar ciri-ciri seekor ayam, orang yang sebelumnya telah mengenal pola ketiga jenis ayam bisa menentukan jenis (*class*) ayam tersebut. Patut digarisbawahi di sini bahwa orang yang bisa menentukan jenis ayam adalah orang yang sebelumnya telah memiliki pengetahuan ciri-ciri dari beberapa jenis ayam. Sedangkan orang yang sebelumnya belum memiliki pengetahuan tentang ciri – ciri jenis ayam tidak bisa memprediksi dengan akurasi tinggi.

Klasifikasi memiliki cara kerja yang mirip dengan cara manusia mengenali jenis sebuah obyek. Asumsikan terdapat sekumpulan data yang disebut *data training* . Setiap data memiliki beberapa atribut dan sebuah *class*. Maka klasifikasi merupakan pencarian model *class* data sebagai fungsi dari nilai-nilai atributnya berdasar pengamatan terhadap *data training* . Dengan demikian jika ada data yang tidak diketahui kelasnya, maka dengan model yang telah ditemukan dapat dilakukan prediksi nilai kelas data seakurat mungkin [14].



Gambar 1 Tahap tahap klasifikasi [14]

Pada gambar 1 ditunjukkan tahap-tahap klasifikasi. Tahap pertama adalah membuat *data training* . *Data training* merupakan sekumpulan data di mana setiap data memiliki atribut dan *class*. *Data training* berfungsi sebagai sumber data untuk dipelajari oleh learning algorithm. Dari pengamatan terhadap *data training* , *learning algorithm* akan membuat sebuah model yang bisa membedakan antara satu *class* dan *class* lainnya (learn model). Model tersebut merupakan hasil analisa *class* data sebagai fungsi dari nilai atribut-atributnya pada *data training* . Model tersebut dapat digunakan untuk menentukan hubungan antara nilai atribut sebuah *class* dengan nilai *class*nya. Dengan demikian, setelah terbentuk model maka kita bisa memprediksi *class* sebuah object berdasar nilai atribut yang dimiliki

Untuk mengetahui kualitas sebuah model dalam melakukan prediksi *class* maka harus diajukan pengujian. Pada gambar 1, pengujian dilakukan pada proses *apply model* pada *test set*. *Test set* merupakan sekumpulan data di mana setiap data memiliki atribut dan *class* yang disembunyikan. Model yang telah dibuat pada saat learn model akan diuji akurasi dengan memprediksi *class* dari data-data pada data test. Kebenaran nilai prediksi dicek menggunakan *class* data test yang sebelumnya disembunyikan. Makin tinggi persentase nilai kebenaran dari proses prediksi maka model yang digunakan makin berkualitas

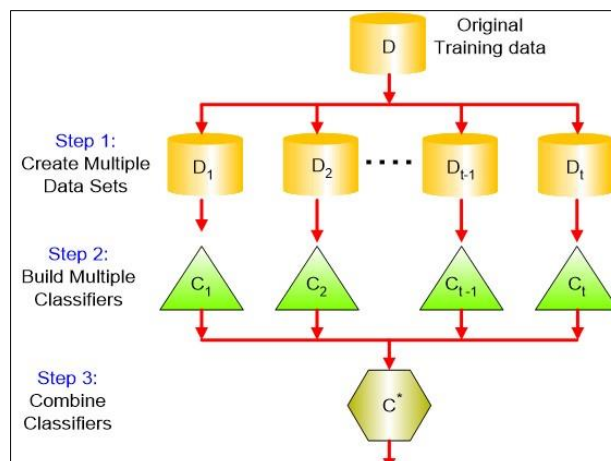
2.3 Feature Subset Selection

Feature Subset Selection merupakan pemilihan atribut secara otomatis menggunakan algoritma tertentu untuk mendapatkan atribut yang relevan dengan permasalahan pemodelan prediktif yang sedang dihadapi [13]. Dalam melakukan proses learning pada klasifikasi digunakan banyak atribut. Tidak semua atribut relevan. Bila ada atribut tidak relevan yang digunakan, maka hasilnya malah bisa menurunkan kualitas model klasifikasi. Karena itu diperlukan pemilihan atribut yang relevan dengan proses klasifikasi sehingga nilai akurasi dan metrik lain bisa meningkat

Beberapa metode *Feature Subset Selection*

1. **Metode filter**
Metode filter menerapkan ukuran statistik untuk memberikan skor untuk masing-masing atribut atau fitur. Fitur yang memiliki skor baik dipilih untuk disimpan dan digunakan pada tahap klasifikasi. Fitur yang memiliki skor kurang tidak digunakan pada tahap klasifikasi
2. **Metode wrapper**
Metode *wrapper* mempertimbangkan pemilihan seperangkat fitur sebagai masalah pencarian, di mana kombinasi atribut yang berbeda dibuat, dievaluasi dan dibandingkan dengan kombinasi lainnya [15]. Model prediksi digunakan untuk mengevaluasi kombinasi fitur dan menetapkan kombinasi atribut yang digunakan berdasarkan akurasi model.
3. **Embedded method**
Metode ini memilih atribut yang bisa memberi kontribusi positif terhadap metrik kualitas model ketika proses *training*

2.4 Ensemble Learning



Gambar 2 tahap tahap *Ensemble Learning* [14]

Ensemble Learning merupakan teknik klasifikasi yang menggunakan banyak *classifier*. Pada klasifikasi biasa, *classifier* yang digunakan hanya satu buah. Multi *classifier* tersebut yang digunakan untuk melakukan prediksi *class* pada object yang belum diketahui *class*nya. Proses prediksi dilaksanakan menggunakan mekanisme voting, di mana obyek akan dilabeli dengan *class* yang memiliki suara terbanyak.

Pada gambar 2 ditunjukkan tahap – tahap untuk membangun *Ensemble Learning*. pembuatan *classifier* pada *Ensemble Learning* terdiri atas tiga tahap

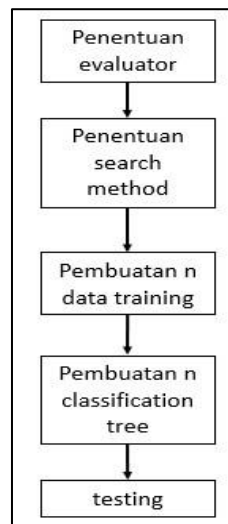
1. **Pembuatan dataset**
Pada tahap ini dibuat *data training* sebanyak *t* buah. Nilai *T* sesuai dengan banyaknya *classifier* yang akan dibuat. Sebuah *data training* dari *t data training* digunakan untuk menghasilkan satu *Classification Tree*. Cara pembuatan sebuah *data training* yang menjadi anggota *t data training* adalah sebagai berikut : dataset disampling dengan ukuran tertentu secara acak dengan menggunakan metode sampling with replacement
2. **Pembuatan multi Classification Tree**
Pada tahap ini dibuat *Classification Tree* sebanyak *t*. Masing – masing *Classification Tree* dibuat dari satu *data training* yang merupakan anggota *t data training* yang telah dibuat di tahap sebelumnya. Pada akhir tahap ini dihasilkan *t Classification Tree* yang memiliki struktur berbeda. Hal ini disebabkan karena pada saat pembuatan *data training* , proses sampling dilakukan secara acak

sehingga masing-masing *data training* pada *t data training* yang dihasilkan isinya berbeda-beda. Akibatnya *Classification Tree* yang dihasilkan oleh *data training* juga tidak sama

3. Testing

Pada saat testing, data test diuji menggunakan *n Classification Tree* yang telah dibuat di tahap sebelumnya. Karena *classifiernya* banyak, bisa jadi hasil prediksi dari setiap *classifier* akan berbeda. Untuk menentukan keputusan akhir, maka *classifier-classifier* tersebut melakukan voting untuk menentukan *class* dari data testing. Setelah melakukan testing terhadap data test menggunakan teknik testing tertentu, maka kualitas *classifier* yang ditunjukkan oleh nilai akurasi dan *recall* akan dapat diketahui. Makin tinggi nilai akurasi dan *recall* maka makin bagus juga kualitas *classifier*

2.5 Jalan penelitian



Gambar 3 Diagram tahap-tahap penelitian

Tahap – tahap penelitian klasifikasi resiko kehamilan menggunakan *Ensemble Learning* berbasis *Classification Tree* seperti yang ditunjukkan di gambar 3

1. Penentuan evaluator

Pada tahap ini ditentukan evaluator yang akan digunakan. Evaluator berfungsi sebagai metrik untuk menilai kontribusi atribut pada perubahan nilai akurasi dan *recall* klasifikasi. Pada penelitian ini *attribute evaluator* yang digunakan adalah SVMattributeeval. SVMattributeeval merupakan *attribute evaluator* yang didasarkan pada *Support Vector Machine*. Pada penelitian sebelumnya [5] digunakan *attribute evaluator information gain*.

2. Penentuan search method

Pada tahap ini dicari kombinasi dari atribut yang akan digunakan dalam proses klasifikasi. Penentuan kombinasi mana yang akan dipilih dinilai menggunakan evaluator SVMattributeeval. Pada penelitian ini *search method* yang digunakan adalah *ranker*. Ranker digunakan untuk memperingkat atribut berdasar hasil SVMattributeeval.

3. Pembuatan n data training

Pada tahap ini dibuat *data training* sebanyak *n* buah. *N* merupakan banyaknya *classifier* yang akan dibuat. Sebuah *data training* dari *n data training* digunakan untuk menghasilkan satu *Classification Tree*. Pada uji coba yang dilakukan, nilai *n* yang digunakan adalah 15, 25 dan 35

4. Pembuatan n Classification Tree

Pada tahap ini dibuat *Classification Tree* sebanyak *n*. Masing – masing *Classification Tree* dibuat dari satu *data training* yang merupakan anggota *n data training* yang telah dibuat di tahap sebelumnya. Pada akhir tahap ini dihasilkan *n Classification Tree* yang memiliki struktur berbeda. Pada uji coba yang dilakukan nilai *n* yang digunakan adalah 15, 25 dan 35 buah *tree*. Setiap *tree* dibuat menggunakan algoritma J48 [2, 9]. *Ensemble classification* diaplikasikan dengan menggunakan teknik bagging. Atribut yang dipilih adalah 10 atribut yang paling signifikan yang didapat dari tahap *Feature Subset Selection*

5. Testing

Pada saat testing, data test diuji menggunakan *t Classification Tree* yang telah dibuat di tahap sebelumnya. Karena *classifiernya* banyak, bisa jadi hasil prediksi dari setiap *classifier* akan berbeda. Untuk menentukan keputusan akhir, maka *classifier-classifier* tersebut melakukan voting untuk

menentukan *class* dari data testing. Setelah melakukan testing terhadap data test menggunakan teknik testing tertentu, maka kualitas *classifier* yang ditunjukkan oleh nilai akurasi dan *recall* akan dapat diketahui. Makin tinggi nilai akurasi dan *recall* maka makin bagus juga kualitas *classifier*

3. Hasil dan Analisa

3.1. pembuatan Dataset

Pada tahap ini dilakukan penentuan atribut apa saja yang digunakan untuk proses klasifikasi. Pada penelitian ini atribut yang digunakan adalah atribut penentuan resiko kehamilan seperti pada [12] yang didasarkan pada perhitungan skor Poedji Rochyati. Dari atribut – atribut skor Poedji Rochyati yang telah disebutkan, terdapat beberapa atribut yang nilainya kemungkinan tidak diketahui secara dini pada proses kehamilan. Ada juga atribut yang tidak relevan dengan proses klasifikasi yang jika digunakan tidak mengakibatkan akurasi atau *recall* klasifikasi naik. Ada atribut yang jika digunakan malah menurunkan akurasi dan *recall* klasifikasi. Selain terlalu banyak atribut yang digunakan bisa menurunkan akurasi maupun *recall* klasifikasi (*curse of dimensionality*), efek lain dari penggunaan atribut yang terlalu banyak adalah peningkatan waktu yang dibutuhkan untuk melakukan training klasifikasi

Pada penelitian ini proses *Feature Subset Selection* dilakukan menggunakan *attribute evaluator* SVMattributeeval dengan search method *ranker*. SVMattributeeval merupakan *attribute evaluator* yang didasarkan pada Support Vector Machine. *Ranker* digunakan untuk memperingkat atribut berdasar hasil SVMattributeeval. Pada penelitian sebelumnya digunakan *attribute evaluator information gain* [5]. Dengan penggunaan SVMattributeeval diharapkan nilai akurasi dan *recall* dari penelitian akan naik.

Dengan menggunakan teknik ini dengan bantuan *software* weka ini didapatkan peringkat atribut sebagai berikut. Pada penelitian ini hanya diambil 10 atribut paling berpengaruh pada proses klasifikasi. Atribut tersebut antara lain usia, tinggi badan, pendarahan, jumlah anak, kembar dua, infus transfusi, sesar, jarak kehamilan, gagal kehamilan, bengkak muka

3.2. Training dan Testing

Metode uji coba yang digunakan adalah metode *10 fold cross validation*. Pada metode uji coba ini dataset dibagi menjadi 10 partisi. Uji coba dilakukan sebanyak 10 kali. Pada setiap uji coba satu partisi digunakan sebagai data test dan 9 partisi digunakan sebagai data training . Sehingga setelah 10 kali uji coba, semua partisi mendapat giliran sebagai data test. Hasil dari *10 fold cross validation* adalah 10 nilai akurasi dan *recall*

Skenario uji coba yang dilakukan antara lain

1. Uji coba dengan 10 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 15
2. Uji coba dengan 10 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 25
3. Uji coba dengan 10 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 35
4. Uji coba dengan 9 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 15
5. Uji coba dengan 9 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 25
6. Uji coba dengan 9 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 35
7. Uji coba dengan 8 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 15
8. Uji coba dengan 8 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 25
9. Uji coba dengan 8 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 35
10. Uji coba dengan 7 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 15
11. Uji coba dengan 7 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 25
12. Uji coba dengan 7 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 35

Uji coba dengan 10 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 15

Uji coba skenario ini dilakukan dengan 10 atribut teratas hasil *Feature Subset Selection* dengan menggunakan evaluator SVMattributeeval dengan search method *ranker*. Jumlah *tree* yang digunakan adalah 15. Setiap *tree* dibuat menggunakan algoritma J48. Uji coba menggunakan *software* weka didapatkan hasil akurasi 74,78% dan *recall* 79,6% seperti yang ditunjukkan pada tabel 1

Tabel 1 Confusion matrix hasil uji coba menggunakan 10 atribut teratas dan jumlah *tree* 15

	Diprediksi sebagai KRR	Diprediksi sebagai KBR
Class asli KRR	20 (<i>true negative</i>)	39 (<i>false negative</i>)
Class asli KBR	19 (<i>false positive</i>)	152 (<i>true positive</i>)

Uji coba dengan 10 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 25

Uji coba skenario ini dilakukan dengan 10 atribut teratas hasil *Feature Subset Selection* dengan menggunakan evaluator SVMAttributeeval dengan *search method ranker*. Jumlah *tree* yang digunakan adalah 25. Setiap *tree* dibuat menggunakan algoritma J48. Uji coba menggunakan *software weka* didapatkan hasil akurasi 76% dan *recall* 80,5% seperti yang ditunjukkan pada tabel 2

Tabel 2 Confusion matrix hasil uji coba menggunakan 10 atribut teratas dan jumlah *tree* 25

	Diprediksi sebagai KRR	Diprediksi sebagai KBR
Class asli KRR	22 (<i>true negative</i>)	37 (<i>false negative</i>)
Class asli KBR	18 (<i>false positive</i>)	153 (<i>true positive</i>)

Uji coba dengan 10 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 35

Uji coba skenario ini dilakukan dengan 10 atribut teratas hasil *Feature Subset Selection* dengan menggunakan evaluator SVMAttributeeval dengan *search method ranker*. Jumlah *tree* yang digunakan adalah 35. Setiap *tree* dibuat menggunakan algoritma J48. Uji coba menggunakan *software weka* didapatkan hasil akurasi 74,7% dan *recall* 89,5% seperti yang ditunjukkan pada tabel 3

Tabel 3 Confusion matrix hasil uji coba menggunakan 10 atribut teratas dan jumlah *tree* 35

	Diprediksi sebagai KRR	Diprediksi sebagai KBR
Class asli KRR	19 (<i>true negative</i>)	40 (<i>false negative</i>)
Class asli KBR	18 (<i>false positive</i>)	153 (<i>true positive</i>)

Uji coba dengan 9 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 15

Uji coba skenario ini dilakukan dengan 9 atribut teratas hasil *Feature Subset Selection* dengan menggunakan evaluator SVMAttributeeval dengan *search method ranker*. Jumlah *tree* yang digunakan adalah 15. Setiap *tree* dibuat menggunakan algoritma J48. Uji coba menggunakan *software weka* didapatkan hasil akurasi 72,6% dan *recall* 85,4% seperti yang ditunjukkan pada tabel 4

Tabel 4 Confusion matrix hasil uji coba menggunakan 9 atribut teratas dan jumlah *tree* 15

	Diprediksi sebagai KRR	Diprediksi sebagai KBR
Class asli KRR	21 (<i>true negative</i>)	38 (<i>false negative</i>)
Class asli KBR	25 (<i>false positive</i>)	146 (<i>true positive</i>)

Uji coba dengan 9 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 25

Uji coba skenario ini dilakukan dengan 9 atribut teratas hasil *Feature Subset Selection* dengan menggunakan evaluator SVMAttributeeval dengan *search method ranker*. Jumlah *tree* yang digunakan adalah 25. Setiap *tree* dibuat menggunakan algoritma J48. Uji coba menggunakan *software weka* didapatkan hasil akurasi 73,9% dan *recall* 87,1% seperti yang ditunjukkan pada tabel 5

Tabel 5 Confusion matrix hasil uji coba menggunakan 9 atribut teratas dan jumlah *tree* 25

	Diprediksi sebagai KRR	Diprediksi sebagai KBR
Class asli KRR	21 (<i>true negative</i>)	38 (<i>false negative</i>)
Class asli KBR	22 (<i>false positive</i>)	149 (<i>true positive</i>)

Uji coba dengan 9 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 35

Uji coba skenario ini dilakukan dengan 9 atribut teratas hasil *Feature Subset Selection* dengan menggunakan evaluator SVMAttributeeval dengan *search method ranker*. Jumlah *tree* yang digunakan adalah 35. Setiap *tree* dibuat menggunakan algoritma J48. Uji coba menggunakan *software weka* didapatkan hasil akurasi 72,1% dan *recall* 85,5% seperti yang ditunjukkan pada tabel 6

Tabel 6 Confusion matrix hasil uji coba menggunakan 9 atribut teratas dan jumlah *tree* 35

	Diprediksi sebagai KRR	Diprediksi sebagai KBR
Class asli KRR	20 (<i>true negative</i>)	39 (<i>false negative</i>)
Class asli KBR	25 (<i>false positive</i>)	146 (<i>true positive</i>)

Uji coba dengan 8 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 15

Uji coba skenario ini dilakukan dengan 8 atribut teratas hasil *Feature Subset Selection* dengan menggunakan evaluator SVMAttributeeval dengan *search method ranker*. Jumlah *tree* yang digunakan adalah

15. Setiap *tree* dibuat menggunakan algoritma J48. Uji coba menggunakan *software* weka didapatkan hasil akurasi 73,4% dan *recall* 86,5% seperti yang ditunjukkan pada tabel 7

Tabel 7 Confusion matrix hasil uji coba menggunakan 8 atribut teratas dan jumlah *tree* 15

	Diprediksi sebagai KRR	Diprediksi sebagai KBR
Class asli KRR	22 (<i>true negative</i>)	37 (<i>false negative</i>)
Class asli KBR	24 (<i>false positive</i>)	147 (<i>true positive</i>)

Uji coba dengan 8 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 25

Uji coba skenario ini dilakukan dengan 8 atribut teratas hasil *Feature Subset Selection* dengan menggunakan evaluator SVMAttributeeval dengan *search method ranker*. Jumlah *tree* yang digunakan adalah 25. Setiap *tree* dibuat menggunakan algoritma J48. Uji coba menggunakan *software* weka didapatkan hasil akurasi 72,1% dan *recall* 85,4% seperti yang ditunjukkan pada tabel 8

Tabel 8 Confusion matrix hasil uji coba menggunakan 8 atribut teratas dan jumlah *tree* 15

	Diprediksi sebagai KRR	Diprediksi sebagai KBR
Class asli KRR	20 (<i>true negative</i>)	39 (<i>false negative</i>)
Class asli KBR	25 (<i>false positive</i>)	146 (<i>true positive</i>)

Uji coba dengan 8 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 35

Uji coba skenario ini dilakukan dengan 8 atribut teratas hasil *Feature Subset Selection* dengan menggunakan evaluator SVMAttributeeval dengan *search method ranker*. Jumlah *tree* yang digunakan adalah 35. Setiap *tree* dibuat menggunakan algoritma J48. Uji coba menggunakan *software* weka didapatkan hasil akurasi 73% dan *recall* 86,5% seperti yang ditunjukkan pada tabel 9

Tabel 9 Confusion matrix hasil uji coba menggunakan 8 atribut teratas dan jumlah *tree* 35

	Diprediksi sebagai KRR	Diprediksi sebagai KBR
Class asli KRR	20 (<i>true negative</i>)	39 (<i>false negative</i>)
Class asli KBR	23 (<i>false positive</i>)	148 (<i>true positive</i>)

Uji coba dengan 7 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 15

Uji coba skenario ini dilakukan dengan 7 atribut teratas hasil *Feature Subset Selection* dengan menggunakan evaluator SVMAttributeeval dengan *search method ranker*. Jumlah *tree* yang digunakan adalah 15. Setiap *tree* dibuat menggunakan algoritma J48. Uji coba menggunakan *software* weka didapatkan hasil akurasi 73,4% dan *recall* 88,3% seperti yang ditunjukkan pada tabel 10

Tabel 10 Confusion matrix hasil uji coba menggunakan 7 atribut teratas dan jumlah *tree* 15

	Diprediksi sebagai KRR	Diprediksi sebagai KBR
Class asli KRR	18 (<i>true negative</i>)	51 (<i>false negative</i>)
Class asli KBR	20 (<i>false positive</i>)	151 (<i>true positive</i>)

Uji coba dengan 7 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 15

Uji coba skenario ini dilakukan dengan 7 atribut teratas hasil *Feature Subset Selection* dengan menggunakan evaluator SVMAttributeeval dengan *search method ranker*. Jumlah *tree* yang digunakan adalah 25. Setiap *tree* dibuat menggunakan algoritma J48. Uji coba menggunakan *software* weka didapatkan hasil akurasi 73,9% dan *recall* 87,1% seperti yang ditunjukkan pada tabel 11

Tabel 11 Confusion matrix hasil uji coba menggunakan 7 atribut teratas dan jumlah *tree* 125

	Diprediksi sebagai KRR	Diprediksi sebagai KBR
Class asli KRR	21 (<i>true negative</i>)	38 (<i>false negative</i>)
Class asli KBR	22 (<i>false positive</i>)	149 (<i>true positive</i>)

Uji coba dengan 7 atribut teratas tahap *Feature Subset Selection* dengan jumlah *tree* 35

Uji coba skenario ini dilakukan dengan 7 atribut teratas hasil *Feature Subset Selection* dengan menggunakan evaluator SVMAttributeeval dengan *search method ranker*. Jumlah *tree* yang digunakan adalah 35. Setiap *tree* dibuat menggunakan algoritma J48. Uji coba menggunakan *software* weka didapatkan hasil akurasi 72,1% dan *recall* 86,5% seperti yang ditunjukkan pada tabel 12

Tabel 12 Confusion matrix hasil uji coba menggunakan 7 atribut teratas dan jumlah *tree* 135

	Diprediksi sebagai KRR	Diprediksi sebagai KBR
--	------------------------	------------------------

Class asli KRR	18 (<i>true negative</i>)	41 (<i>false negative</i>)
Class asli KBR	23 (<i>false positive</i>)	148 (<i>true positive</i>)

3.3. Analisa Hasil Uji Coba

Dari hasil uji coba C1 sampai C12 disusun tabel 13 yang berisi ringkasan nilai akurasi uji klasifikasi menggunakan *Feature Subset Selection SVMAttributeEval* dan ensemble *tree learning*. Dari tabel 13 dapat disimpulkan bahwa jika klasifikasi dilakukan menggunakan semakin sedikit atribut (mulai dari 10 sampai 7 atribut) maka terdapat tren nilai akurasi menurun. Jumlah atribut 10 merupakan jumlah optimal untuk akurasi pada beberapa variasi jumlah yang diuji coba. Terdapat kenaikan tajam nilai akurasi mulai penggunaan atribut ke 10. Namun pada rentang kurang dari 10 atribut, tren penurunan tersebut tidak terlalu signifikan.

Selain itu pada tabel 13 dapat dilihat hubungan antara jumlah *tree* yang digunakan dengan akurasi. Pada penelitian ini digunakan 15, 25, dan 35 *tree*. Dari tabel 13 bisa dilihat bahwa tidak ada perubahan signifikan pada akurasi. Sehingga tidak bisa disimpulkan jumlah *tree* berpengaruh ke nilai akurasi.

Dari hasil uji coba C1 sampai C12 juga disusun tabel 14 yang berisi ringkasan nilai *recall* uji klasifikasi menggunakan *Feature Subset Selection SVMAttributeEval* dan ensemble *tree learning*. Dari tabel 14 dapat disimpulkan bahwa jika klasifikasi dilakukan menggunakan semakin sedikit atribut (mulai dari 10 sampai 7 atribut) maka terdapat tren nilai akurasi naik. Hal ini dapat dilihat pada penggunaan 15 *tree* dan 25 *tree*. Sedang pada 35 *tree* justru nilai 10 atribut menghasilkan *recall* terbaik. Namun pada jumlah atribut yang kurang dari 10 tidak memberikan perbedaan signifikan untuk *tree* 35.

Selain itu pada tabel 14 dapat dilihat hubungan antara jumlah *tree* yang digunakan dengan *recall*. Dari tabel 14 bisa dilihat bahwa tidak ada perubahan signifikan dari *recall* dengan variasi jumlah *tree* 15, 25 dan 35. Sehingga tidak bisa disimpulkan jumlah *tree* berpengaruh ke nilai *recall*. Namun khusus pada kasus 10 atribut, kenaikan jumlah *tree* yang digunakan secara signifikan akan meningkatkan nilai *recall*.

Tabel 13 Nilai akurasi klasifikasi menggunakan *Feature Subset Selection* dan *ensemble tree learning*

Jumlah fitur/jumlah <i>tree</i>	15 <i>tree</i>	25 <i>tree</i>	35 <i>tree</i>
10 atribut	74,78 %	76 %	74,7 %
9 atribut	72,6 %	73,9 %	72,1 %
8 atribut	73,4 %	72,1 %	73 %
7 atribut	74,4 %	73,9 %	72,1 %

Tabel 4 Nilai *recall* klasifikasi menggunakan *Feature Subset Selection* dan *ensemble tree learning*

Jumlah fitur/jumlah <i>tree</i>	15 <i>tree</i>	25 <i>tree</i>	35 <i>tree</i>
10 atribut	79,6 %	80,5 %	89,5 %
9 atribut	85,4%	87,1 %	85,5 %
8 atribut	86 %	85,4 %	86,5 %
7 atribut	88,3 %	87,1 %	86,5 %

Bila dibandingkan dengan penelitian sebelumnya, yaitu klasifikasi menggunakan *cost sensitive learning* [5], metode penggunaan *Ensemble Learning* lebih unggul baik pada segi nilai akurasi terbaik maupun *recall* terbaik. Hasil terbaik dari *cost sensitive learning* adalah 73 % dengan menggunakan 6 atribut dan *Feature Subset Selection* menggunakan *attribute evaluator* information gain. Sedang metode *Ensemble Learning* menghasilkan akurasi terbaik 76% menggunakan 10 atribut dari *Feature Subset Selection* menggunakan *attribute evaluator SVMAttributeEval*.

Perbandingan nilai *recall* metode *cost sensitive learning* dengan *Ensemble Learning* juga menunjukkan kenaikan yang signifikan. Hasil terbaik dari *recall cost sensitive learning* adalah 77,9 % dengan menggunakan 6 atribut dan *Feature Subset Selection* menggunakan *attribute evaluator* information gain. Sedang metode *Ensemble Learning* menghasilkan *recall* terbaik 89,5% menggunakan 10 atribut dari *Feature Subset Selection* menggunakan *attribute evaluator SVMAttributeEval*.

4. Kesimpulan

Dari uji coba yang telah dilakukan dapat diambil kesimpulan sebagai berikut. Klasifikasi resiko kehamilan Poedji Rochyati menggunakan metode *Ensemble Learning* berbasis *Classification Tree* berhasil memperbaiki nilai akurasi penelitian sebelumnya yang berbasis *cost sensitive learning*. Untuk akurasi, metode *Ensemble Learning* menghasilkan nilai terbaik 76% dibanding metode *cost sensitive learning* yang menghasilkan nilai terbaik 73%. Untuk *recall*, metode *Ensemble Learning* menghasilkan nilai terbaik 89,5% dibanding metode *cost sensitive learning* yang menghasilkan nilai terbaik 77,9%.

Terdapat hubungan terbalik antara jumlah atribut yang digunakan dalam klasifikasi dengan nilai *recall*. Makin sedikit atribut yang digunakan, sampai nilai optimal tertentu, maka nilai *recall* makin naik. Tidak adanya korelasi antara jumlah atribut dan *tree* yang digunakan dengan nilai akurasi. Demikian juga antara jumlah *tree* yang digunakan dengan nilai *recall*.

Ucapan Terima Kasih

- Ucapan terima kasih disampaikan peneliti kepada
1. Fakultas Ilmu Komputer Universitas Jember
 2. Qomariatul Hasanah S.Kom. yang telah menyusun dataset yang digunakan di penelitian ini

Daftar Pustaka

- [1] Chen, CH., Tanaka, K., Kotera, M. et al. Comparison and improvement of the predictability and interpretability with ensemble learning models in QSPR applications. *J Cheminform* 12, 19, 2020
- [2] Cruz, A. and Tumibay, G. Predicting Tuberculosis Treatment Relapse: A Decision Tree Analysis of J48 for Data Mining. *Journal of Computer and Communications*, 7, 243-251, 2019
- [3] Hasanah, Qomariyatul, Andrianto, Anang, dan Hidayat, Muhamad Arief.. Sistem Informasi Posyandu Ibu Hamil Dengan Penerapan Klasifikasi Resiko Kehamilan Menggunakan Metode Naive Bayes. *Berkala Sainstek* 6(1):1. Mei 2018
- [4] Gomes, Heitor Murilo & Barddal, Jean Paul & Enembreck, Fabrício & Bifet, Albert. A Survey on Ensemble Learning for Data Stream Classification. *ACM Comput.* March 2017
- [5] Hidayat, Muhamad Arief. Klasifikasi resiko kehamilan menggunakan cost sensitive learning. LP2M Universitas Jember. 2017
- [6] Huang, Shujun et al. "Applications of Support Vector Machine (SVM) Learning in Cancer Genomics." *Cancer genomics & proteomics* vol. 15,1, 2018 : 41-51
- [7] Jair Cervantes, Farid Garcia-Lamont, Lisbeth Rodríguez-Mazahua, Asdrubal Lopez, A comprehensive survey on support vector machine classification: Applications, challenges and trends, *neurocomputing*, Volume 408, 2020, Pages 189-215,
- [8] Mariyona, Kartika. Komplikasi Dan Faktor Resiko Kehamilan Di Puskemas. *Jurnal Menara Medika*. Vol 1 No 2 Maret 2019
- [9] N.SaravanaN, Dr.V.Gayathri "Performance and Classification Evaluation of J48 Algorithm and Kendall's Based J48 Algorithm (KNJ48)". *International Journal of Computer Trends and Technology (IJCTT)* V59(2):73-80, May 2018
- [10] Nti, I.K., Adekoya, A.F. & Weyori, B.A. A comprehensive evaluation of ensemble learning for stock-market prediction. *J Big Data* 7, 20, 2020
- [11] Ranjit Panigrahi, Samarjeet Borah, Rank Allocation to J48 Group of Decision Tree Classifiers using Binary and Multiclass Intrusion Detection Datasets, *Procedia Computer Science*, Volume 132, 2018, Pages 323-332,
- [12] Rochyati, P. Skrining antenatal pada ibu hamil. Surabaya. *Airlangga Press*. 2003
- [13] Rostami, M., Berahmand, K. & Forouzandeh, S. A novel method of constrained feature selection by the measurement of pairwise constraints uncertainty. *J Big Data* 7, 83 (2020).
- [14] Tan, Pang-ning. Steinbach. Kumar , Introduction to Data Mining 1st Edition. *Pearson*. 2005
- [15] Yifei Mao, Yuansheng Yang, "A Wrapper Feature Subset Selection Method Based on Randomized Search and Multilayer Structure", *BioMed Research International*, vol. 2019, Article ID 9864213, 9 pages, 2019