

IMPLEMENTASI DATA MINING UNTUK KLASIFIKASI NASABAH KREDIT BANK "X" MENGGUNAKAN CLASSIFICATION RULE

Hendra Marcos^{1,2)}, Indriana Hidayah¹⁾

¹⁾Jurusan Teknik Elektro Dan Teknik Informatika, Universitas Gadjah Mada

²⁾Sistem Informasi STMIK AMIKOM Purwokerto

Jl Letjend Pol. Sumarto Watumas, Purwonegoro – Purwokerto

Email : hendra.marcos@mail.ugm.ac.id¹⁾, indriana.h@ugm.ac.id²⁾

Abstrak

Perkembangan sistem informasi dalam berbagai bidang usaha perbankan sangat diperlukan dalam rangka memberikan kenyamanan, keamanan, dan kemudahan dalam pengambilan keputusan khususnya pada bidang perkreditan. Tujuan paper ini adalah memberikan kontribusi untuk mempercepat pengambilan keputusan tersebut. Teknik data mining digunakan untuk memberikan model supaya pihak bank cepat dalam membuat keputusan kepada nasabah yang berhak diberi kredit atau ditolak. Dengan beberapa algoritme klasifikasi yang dicobakan terhadap data training diberikan model klasifikasi algoritme C4.5 yang mempunyai nilai akurasi yang tertinggi. Setelah diimplementasikan kepada data tes, didapatkan keputusan nasabah yang ditolak dan diterima kreditnya.

Kata kunci: Data Mining, Klasifikasi, Algoritme, Kredit.

1. Pendahuluan

Bank memiliki peranan yang sangat penting dalam menggerakkan roda perekonomian nasional. Sebagaimana umumnya negara berkembang, sumber pembiayaan dunia usaha di Indonesia masih didominasi oleh penyaluran kredit perbankan yang diharapkan dapat mendorong pertumbuhan ekonomi [1].

Untuk menentukan dengan cepat dan mengurangi resiko terjadinya “kredit macet” dalam pemberian kredit, diperlukan analisis terhadap pola data *training* dari nasabah yang sudah ada untuk diekstrak pengetahuannya berupa pohon keputusan atau aturan yang mudah dimengerti, sehingga pihak bank mudah untuk menetapkan kredit yang ditolak atau kredit yang diterima aplikasinya berdasar data-data yang sudah diolah tersebut.

Data mining adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstrasi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai *database* besar [2]. Teknik *data mining* merupakan suatu proses pendukung pengambil keputusan dimana kita mencari pola informasi dalam data [3]. Pencarian ini dapat dilakukan oleh pengguna, misalnya dengan menggunakan *query* (dalam kasus ini

sangat sulit dilakukan) atau dapat dibantu dengan suatu aplikasi yang secara otomatis mencari pola informasi pada basis data. Pencarian ini disebut *discovery*. *Discovery* adalah proses pencarian dalam basis data untuk menemukan pola yang tersembunyi tanpa ide yang didapatkan sebelumnya atau hipotesa tentang pola yang ada. Dengan kata lain aplikasi mengambil inisiatif untuk menemukan pola dalam data tanpa pengguna berpikir mengenai pertanyaan yang relevan terlebih dulu.

Beberapa penelitian dengan data set yang sama sudah dilakukan, diantaranya menggunakan *hybrid SVM* [4]. Pada penelitian *hybrid SVM* yang dilakukan dengan cara pemilihan parameter yang optimal dan pemilihan fitur terlebih dahulu dapat menambah akurasi hasil klasifikasi. Dari hasil penelitian *hybrid SVM* ini didapatkan nilai akurasi 88,70%, hanya saja model klasifikasi yang dihasilkan tidak spesifik memperlihatkan model yang mudah dipahami dan diketahui informasinya.

Dalam penelitian ini akan dilakukan pengujian dari data *training* terhadap beberapa algoritme klasifikasi. Data set berupa data nasabah yang berjumlah 416 nasabah. Dari data 274 nasabah, akan ditentukan apakah diterima atau ditolak permohonan kreditnya berdasarkan dari data-data yang sudah ada, dicocokkan dengan pola/model yang sudah dibentuk dari data *training* berupa *rule* atau *decision tree*.

2. Landasan Teori

2.1. Defenisi dan Fungsi Data Mining

Pengertian *Data Mining* atau *Knowledge Discovery in Database* (KDD) menurut William J. Frawley, dkk [6] *Data Mining* atau *Knowledge Discovery in Database* (KDD) adalah penyaringan data secara implisit dimana sebelumnya tidak diketahui terdapatnya informasi yang potensial. *Data mining* menganalisis data menggunakan *tool* untuk menemukan pola dan aturan dalam himpunan data. Perangkat lunak bertugas untuk menemukan pola dengan mengidentifikasi aturan dan fitur pada data. *Tool Data mining* diharapkan mampu mengenal pola ini dalam data dengan input minimal dari user.

2.2. Metode Klasifikasi

Pada penelitian ini metode klasifikasi *Data Mining* yang digunakan adalah *classification rule* dan *decision tree*.

Ada beberapa algoritme klasifikasi yang berdasarkan *rule*, diantaranya PART (Projective Adaptive Resonance Theory), 1R (oneR), MODLEM, dll [8]. Kemudian untuk algoritme klasifikasi dengan *decision tree* diantaranya adalah ADTree, J48 (C4.5), REPTree, dll. Pada penelitian ini diusulkan menggunakan J48 dan PART.

PART (Projective Adaptive Resonance Theory)

Algoritme PART dikembangkan oleh Cao dan Wu [5][7], telah terbukti sangat efektif dalam mengelompokkan data set dalam ruang dimensi tinggi. Algoritma PART didasarkan pada asumsi bahwa model persamaan PART (dengan skala besar dan luar biasa terganggu terhadap sistem persamaan diferensial yang digabungkan dengan mekanisme reset) memiliki kinerja komputasi yang cukup teratur. Membangkitkan *unrestricted decision list* menggunakan prosedur *separate-and-conquer*. Membuat *partial decision tree* untuk menghasilkan *rule*. Menggunakan prosedur C4.5 untuk membangun *tree*. Menghindari *global optimization* yang digunakan pada C4.5 dan RIPPER.

J48

J48 merupakan implementasi algoritma C4.5 (berbasis Java) pada Weka [8]. Algoritme C4.5 berbasis *Decision Tree* (pohon keputusan). Dalam algoritma C4.5 digunakan information gain untuk memilih atribut yang akan digunakan untuk pemisahan obyek [7]. Atribut yang mempunyai information gain paling tinggi dibanding atribut yang lain relatif terhadap set y dalam suatu data, dipilih untuk melakukan pemecahan. Pada algoritma ini, pemilihan atribut mana yang akan menempati suatu simpul dilakukan dengan melakukan perhitungan entropi informasi (information entropy) dan mencari nilai yang paling minimum. Pemilihan atribut pada algoritma ini berdasarkan pada asumsi bahwa kompleksitas yang dimiliki oleh pohon keputusan sangat berkaitan erat dengan jumlah informasi yang diberikan oleh nilai-nilai atributnya. Dengan kata lain, teknik heuristik berbasis informasi ini memilih atribut yang memberikan perolehan informasi terbesar (highest information gain) dalam menghasilkan subpohon (subtree) untuk mengklasifikasikan sampel. Proses C4.5, pilih atribut sebagai akar (root), buat cabang untuk tiap nilai, bagi kasus ke dalam cabang, ulangi proses untuk tiap cabang hingga semua cabang memiliki kelas yang sama. Atribut akar dipilih berdasar nilai *Gain* tertinggi (membutuhkan nilai perhitungan Entropy).

3. Metodologi Penelitian

Dataset yang disediakan berisi data aplikasi persetujuan kredit dari Bank “X” untuk menentukan persetujuan kredit. Jumlah data training 416 nasabah dan data pengujian 274 nasabah yang belum ada label, artinya 274 nasabah menunggu apakah kreditnya diterima atau ditolak. Jumlah atribut yaitu 15 atribut dan 1 atribut klas. *Tools* yang digunakan untuk pengujian hasil klasifikasi adalah Weka versi 3.7.10, yang didapatkan gratis *opensource* [8].

Tabel 1. Data Training

Nasabah	X1	X2	.	.	.	X14	X15	Class
1	1	30,83	.	.	.	202	0	SETUJU
2	2	58,67	.	.	.	43	560	SETUJU
3	2	24,5	.	.	.	280	824	SETUJU
4	1	27,83	.	.	.	100	3	SETUJU
5	1	20,17	.	.	.	120	0	SETUJU
6	1	32,08	.	.	.	360	0	SETUJU
.
.
.
412	1	17,25	.	.	.	160	40	TOLAK
413	2	17,67	.	.	.	86	0	TOLAK
414	2	?	.	.	.	?	5200	TOLAK
415	1	16,5	.	.	.	132	0	TOLAK
416	2	27,33	.	.	.	340	1	TOLAK

Keterangan :

- Atribut :
 - X1 = 1,2 (nominal)
 - X2 = kontinu
 - X3 = kontinu
 - X4 = 1,2,3,4 (nominal)
 - X5 = 1,2,3 (nominal)
 - X6 = 1,2,3,4,5,6,7,8,9,10,11,12,13,14 (nominal)
 - X7 = 1,2,3,4,5,6,7,8,9 (nominal)
 - X8 = kontinu
 - X9 = 1,2 (nominal)
 - X10= 1,2 (nominal)
 - X11=kontinu
 - X12=1,2 (nominal)
 - X13=1,2,3 (nominal)
 - X14=kontinu
 - X15=kontinu
 - X16=Klas (Setuju,Tolak)
- Data *training* 416 obyek/*instance* data
- Atribut *Class* Setuju = 185, Tolak = 231
- Ada *missing* atribut :
 - Persentase masing-masing dari jumlah data keseluruhan
 - X1 = 3 (1%) X9 = 0
 - X2 = 8 (2%) X10 = 0
 - X3 = 0 X11 = 0
 - X4 = 2 (1%) X12 = 0
 - X5 = 2 (1%) X13 = 0
 - X6 = 2 (1%) X14 = 7 (1%)
 - X7 = 2 (1%) X15 = 0
 - X8 = 0 X16 = 0

Tabel 2. Data Testing (belum ada class)

Nasabah	X1	X2	.	.	.	X14	X15	Class
417	1	33,5	.	.	.	253	857	?
418	1	27,67	.	.	.	487	500	?
419	2	58,42	.	.	.	0	6700	?
420	2	20,67	.	.	.	220	2503	?
.
.
.
686	1	21,08	.	.	.	260	0	?
687	2	22,67	.	.	.	200	394	?
688	2	25,25	.	.	.	200	1	?
689	1	17,92	.	.	.	280	750	?
690	1	35	.	.	.	0	0	?

Langkah-Langkah Metodologi

a) *Preprocessing Data*

Preprocessing dilakukan untuk data set adalah normalisasi nilai atribut yang mempunyai pesebaran yang besar diantaranya adalah untuk atribut yang bernilai numerik X2, X3, X8, X11, X14, dan X15 dengan menggunakan rumus normalisasi data.

$$\bar{x} = \frac{x' - x_{min}}{x_{max} - x_{min}}$$

\bar{x} = data ternormalisasi

x' = data yang akan dinormalisasi

Tabel 3. Data Training (yang ternormalisasi)

Nasabah	X1	X2	.	.	.	X14	X15	Class
1	1	0.28	.	.	.	0.11	0	SETUJU
2	2	0.78	.	.	.	0.03	0.02	SETUJU
3	2	0.17	.	.	.	0.14	0.02	SETUJU
4	1	0.23	.	.	.	0.05	0.01	SETUJU
5	1	0.09	.	.	.	0.06	0	SETUJU
6	1	0.3	.	.	.	0.18	0	SETUJU
.
.
.
412	1	0.04	.	.	.	0.08	0.01	TOLAK
413	2	0.05	.	.	.	0.05	0	TOLAK
414	2	?	.	.	.	?	0.11	TOLAK
415	1	0.03	.	.	.	0.07	0	TOLAK
416	2	0.22	.	.	.	0.17	0.01	TOLAK

Tabel 4. Data Testing (belum ada class) ternormalisasi

Nasabah	X1	X2	.	.	.	X14	X15	Class
417	1	0.49	.	.	.	0.49	0.86	?
418	1	0.33	.	.	.	0.94	0.5	?
419	2	1.17	.	.	.	0	6.7	?
420	2	0.14	.	.	.	0.43	2.51	?
.
.
.
686	1	0.15	.	.	.	0.5	0	?
687	2	0.19	.	.	.	0.39	0.4	?
688	2	0.26	.	.	.	0.39	0.01	?
689	1	0.06	.	.	.	0.54	0.75	?
690	1	0.53	.	.	.	0	0	?

b) *Missing Value*

Dalam dataset yang digunakan untuk uji coba pada makalah ini terdapat *Missing Values* pada atribut X1(1%), X2(2%), X4,X5,X6,X7(0,1%) dan X14(1%) dari keseluruhan jumlah data. Setelah missing value ditangani dengan *ReplaceMissingValue* (diisi dengan mean dan modus) dari data yang ada pada tiap-tiap atribut, ternyata *missing value* tidak berpengaruh terhadap hasil akhir klasifikasi, sehingga pada akhirnya diabaikan saja.

c) Merubah format data dari excel ke format .csv atau .arff

Format Data Training .arff (Weka Classifier)

@relation Train

```
@attribute X1 {1,2}
@attribute X2 numeric
@attribute X3 numeric
@attribute X4 {1,2,3,4}
@attribute X5 {1,2,3}
@attribute X6 {1,2,3,4,5,6,7,8,9,10,11,12,13,14}
@attribute X7 {1,2,3,4,5,6,7,8,9}
@attribute X8 numeric
@attribute X9 {1,2}
@attribute X10 {1,2}
@attribute X11 numeric
@attribute X12 {1,2}
@attribute X13 {1,2,3}
@attribute X14 numeric
@attribute X15 numeric
@attribute X16 {SETUJU,TOLAK}
```

@data

```
1,0.28,0,1,1,10,1,0.05,1,1,0.02,2,1,0.11,0,SETUJU
2,0.78,0.16,1,1,9,2,0.11,1,1,0.09,2,1,0.03,0.02,SETUJU
2,0.17,0.02,1,1,9,2,0.06,1,2,0,2,1,0.14,0.02,SETUJU
1,0.23,0.06,1,1,10,1,0.14,1,1,0.08,1,1,0.05,0.01,SETUJU
1,0.09,0.21,1,1,10,1,0.06,1,2,0,2,3,0.06,0,SETUJU
.
.
.
2,0.05,0,2,2,5,8,0,2,2,0,2,1,0.05,0,TOLAK
2,?,0.41,1,1,14,8,0,2,2,0,2,1,?,0.11,TOLAK
1,0.03,0.01,1,1,1,1,0.01,2,2,0,2,1,0.07,0,TOLAK
2,0.22,0.06,1,1,14,8,0,2,2,0,2,1,0.17,0.01,TOLAK
```

Format Data Testing .arff (Weka Classifier)

@relation Test

```
@attribute X1 {1,2}
@attribute X2 numeric
@attribute X3 numeric
@attribute X4 {1,2,3,4}
@attribute X5 {1,2,3}
@attribute X6 {1,2,3,4,5,6,7,8,9,10,11,12,13,14}
@attribute X7 {1,2,3,4,5,6,7,8,9}
@attribute X8 numeric
@attribute X9 {1,2}
@attribute X10 {1,2}
@attribute X11 numeric
@attribute X12 {1,2}
@attribute X13 {1,2,3}
@attribute X14 numeric
@attribute X15 numeric
@attribute X16 {SETUJU,TOLAK}
```

@data

```
1,0.49,0.13,1,1,11,2,0.55,1,1,0.34,1,1,0.49,0.86,?
1,0.33,0.99,1,1,10,1,0.7,1,2,0,1,1,0.94,0.5,?
2,1.17,1.5,1,1,4,3,1.21,1,1,1.09,2,1,0.6,7,?
2,0.14,0.14,1,1,9,1,0.26,1,1,0.42,2,1,0.43,2.51,?
.
.
.
2,0.19,0.06,1,1,1,1,0.25,2,1,0.17,1,1,0.39,0.4,?
2,0.26,0.97,2,2,14,8,0.25,2,1,0.09,1,1,0.39,0.01,?
1,0.06,0.02,1,1,13,1,0.01,2,2,0,2,1,0.54,0.75,?
1,0.53,0.25,1,1,1,2,1,2,2,0,1,1,0,0,?
```

d) *Attribute Selection / Attribute evaluator*

Attribute evaluator yang digunakan adalah *CfsSubseteval* (Correlation-based feature selector). *Cfs* mengevaluasi nilai subset dari atribut dengan

mempertimbangkan kemampuan prediktif individu masing-masing fitur bersama dengan tingkat redundansi antar fitur/atribut. Perlu adanya langkah seleksi atribut untuk mengurangi atribut yang banyak. *Search Method* yang digunakan yaitu *BestFirst Search*. *BestFirst Search* merupakan algoritme pencarian yang berdasarkan pada optimasi nilai terbaik.

e) *k-fold cross validation*

Menggunakan *Test options fold cross validation* (fold=10) dimana data *training* dan *testing* dibagi 10 data set untuk *training* dan *testing*.

4. Hasil dan Analisis

a. *Attribute Selection*

Seleksi atribut diperlukan untuk mempermudah pengambilan keputusan dalam menetapkan kredit diterima atau ditolak. Dengan seleksi atribut didapatkan atribut yang mempunyai nilai korelasi yang tinggi terhadap *output* klas.

```

=== Attribute Selection on all input data ===

Search Method:
  Best first.
  Start set: no attributes
  Search direction: forward
  Stale search after 5 node expansions
  Total number of subsets evaluated: 78
  Merit of best subset found: 0.465

Attribute Subset Evaluator (supervised, Class (nominal): 16 Class):
  CFS Subset Evaluator
  Including locally predictive attributes

Selected attributes: 4,7,8,9,11,14,15 : 7
  X4
  X7
  X8
  X9
  X11
  X14
  X15
    
```

Gambar 1. Hasil seleksi atribut

Didapatkan 7 atribut masing-masing yaitu X4, X7, X8, X9, X11, X14, dan X15.

b. Membandingkan beberapa hasil klasifikasi

Algoritme J48 (C4.5)

```

Correctly Classified Instances 371      89.1827 %
Incorrectly Classified Instances 45      10.8173 %
Kappa statistic 0.7797
Mean absolute error 0.1399
Root mean squared error 0.2903
Relative absolute error 28.3268 %
Root relative squared error 58.4176 %
Coverage of cases (0.95 level) 96.3942 %
Mean rel. region size (0.95 level) 62.9808 %
Total Number of Instances 416

=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
Weighted Avg.  0,849  0,074  0,902  0,849  0,875  0,781  0,930  0,918  SETUJU
                0,926  0,151  0,884  0,926  0,905  0,781  0,930  0,910  TOLAK

=== Confusion Matrix ===
  a  b  <-- classified as
157 28 | a = SETUJU
17 214 | b = TOLAK
    
```

Gambar 2. Hasil klasifikasi algoritme J48

Algoritme MLP

```

Correctly Classified Instances 357      85.8173 %
Incorrectly Classified Instances 59      14.1827 %
Kappa statistic 0.7142
Mean absolute error 0.1685
Root mean squared error 0.3285
Relative absolute error 34.1211 %
Root relative squared error 66.1081 %
Coverage of cases (0.95 level) 96.875 %
Mean rel. region size (0.95 level) 67.5481 %
Total Number of Instances 416

=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
Weighted Avg.  0,858  0,140  0,860  0,858  0,858  0,715  0,920  0,911  TOLAK

=== Confusion Matrix ===
  a  b  <-- classified as
160 25 | a = SETUJU
34 197 | b = TOLAK
    
```

Gambar 3. Hasil klasifikasi dengan algoritme MLP

Algoritme PART

```

Correctly Classified Instances 366      87.9808 %
Incorrectly Classified Instances 50      12.0192 %
Kappa statistic 0.7548
Mean absolute error 0.1499
Root mean squared error 0.3111
Relative absolute error 30.3488 %
Root relative squared error 62.5923 %
Coverage of cases (0.95 level) 94.2308 %
Mean rel. region size (0.95 level) 63.4615 %
Total Number of Instances 416

=== Detailed Accuracy By Class ===

      TP Rate  FP Rate  Precision  Recall  F-Measure  MCC      ROC Area  PRC Area  Class
Weighted Avg.  0,827  0,078  0,895  0,827  0,860  0,757  0,903  0,875  SETUJU
                0,922  0,173  0,869  0,922  0,895  0,757  0,903  0,873  TOLAK

=== Confusion Matrix ===
  a  b  <-- classified as
153 32 | a = SETUJU
18 213 | b = TOLAK
    
```

Gambar 4. Hasil klasifikasi dengan algoritme PART

c. Model yang dibentuk dari masing-masing algoritme klasifikasi

Algoritme J48 (C4.5)

```

X9 = 1
| X15 <= 0.01
| | X11 <= 0.05
| | | X7 = 1
| | | | X14 <= 0.05
| | | | | X4 = 1: SETUJU (18.0/1.0)
| | | | | X4 = 2
| | | | | | X14 <= 0.02: TOLAK (2.0)
| | | | | | | X14 > 0.02: SETUJU (3.0/1.0)
| | | | | | | X4 = 3: SETUJU (0.0)
| | | | | | | X4 = 4: SETUJU (0.0)
| | | | | | | X14 > 0.05
| | | | | | | | X15 <= 0
| | | | | | | | | X11 <= 0: TOLAK (27.0/8.0)
| | | | | | | | | | X11 > 0: SETUJU (11.0/4.0)
| | | | | | | | | | | X15 > 0
| | | | | | | | | | | | X14 <= 0.09
| | | | | | | | | | | | | X14 <= 0.08: SETUJU (3.0/1.0)
| | | | | | | | | | | | | | X14 > 0.08: TOLAK (3.0)
| | | | | | | | | | | | | | | X14 > 0.09: SETUJU (4.0)
| | | | | | | | | | | | | | | | X7 = 2
| | | | | | | | | | | | | | | | | X4 = 1: SETUJU (19.0)
| | | | | | | | | | | | | | | | | X4 = 2
| | | | | | | | | | | | | | | | | | X8 <= 0.16: SETUJU (4.0/1.0)
| | | | | | | | | | | | | | | | | | X8 > 0.16: TOLAK (4.0)
| | | | | | | | | | | | | | | | | | X4 = 3: SETUJU (0.0)
| | | | | | | | | | | | | | | | | | X4 = 4: SETUJU (0.0)
| | | | | | | | | | | | | | | | | | X7 = 3
| | | | | | | | | | | | | | | | | | | X14 <= 0.08: SETUJU (4.4/0.4)
| | | | | | | | | | | | | | | | | | | | X14 > 0.08: TOLAK (6.6)
| | | | | | | | | | | | | | | | | | | | X7 = 4: TOLAK (1.0)
| | | | | | | | | | | | | | | | | | | | X7 = 5: SETUJU (0.0)
| | | | | | | | | | | | | | | | | | | | X7 = 6: SETUJU (1.0)
| | | | | | | | | | | | | | | | | | | | X7 = 7: SETUJU (0.0)
| | | | | | | | | | | | | | | | | | | | X7 = 8: TOLAK (3.0)
| | | | | | | | | | | | | | | | | | | | X7 = 9: SETUJU (0.0)
| | | | | | | | | | | | | | | | | | | | | X11 > 0.05: SETUJU (43.0/3.0)
| | | | | | | | | | | | | | | | | | | | | X15 > 0.01: SETUJU (70.0)
X9 = 2: TOLAK (189.0/8.0)
    
```

Gambar 5. Model *decision tree* dari algoritme J48

Algoritme MLP

```

Sigma Node 0
Inputs      Weights
Threshold   2.7999622700290265
Node 2      2.313078246694319
Node 3      -6.310417675464461
Node 4      -1.4143338398051313
Node 5      2.4926222817505597
Node 6      -0.7442189166621282
Node 7      3.4815622601433316
Node 8      1.5454953526341477
Node 9      -4.027647957586576
Node 10     4.869019516633043
Node 11     -0.8102747138866057

Sigma Node 1
Inputs      Weights
Threshold   -2.800298208815744
Node 2      -2.204276322167545
Node 3      6.3124409477215035
Node 4      1.4186449360430387
Node 5      -2.522705213708379
Node 6      0.7308222630942942
Node 7      -3.4837589932728465
Node 8      -1.5159905264232467
Node 9      4.027025645197592
Node 10     -4.8696369368539765
Node 11     0.8129494474426172

.
.
.

Sigma Node 10
Inputs      Weights
Threshold   0.23497213655660673
Attrib X4=1 3.6231743827970497
Attrib X4=2 -3.5664020719309386
Attrib X4=3 -0.011647800850181876
Attrib X4=4 -0.004423119313556753
Attrib X7=1 -1.5543974811767853
Attrib X7=2 3.627423456339276
Attrib X7=3 -0.46411810486154237
Attrib X7=4 -0.9954176502493239
Attrib X7=5 -0.1870758399326045
Attrib X7=6 -1.0720387697041858
Attrib X7=7 -0.22659393488896926
Attrib X7=8 -0.13936162803630173
Attrib X7=9 -0.2611298878207511
Attrib X8   4.32387023991562
Attrib X9   -0.8823682493871072
Attrib X11  4.06071751296257
Attrib X14  -4.906927305182534
Attrib X15  7.024909945759453

Sigma Node 11
Inputs      Weights
Threshold   -0.6865427689036571
Attrib X4=1 -0.424147190846483
Attrib X4=2 0.3996772381005719
Attrib X4=3 -0.01895951495702623
Attrib X4=4 -0.04544266931414564
Attrib X7=1 -1.3433395728163093
Attrib X7=2 -0.9118705156465979
Attrib X7=3 0.5628896277633314
Attrib X7=4 0.7343259630773978
Attrib X7=5 0.6707870656238003
Attrib X7=6 2.082084028333327
Attrib X7=7 0.74276436027254
Attrib X7=8 1.6595348031743595
Attrib X7=9 0.7325351793862048
Attrib X8   -2.3287824126338132
Attrib X9   1.443611939585936
Attrib X11  -1.7596997025744874
Attrib X14  1.0540154974129159
Attrib X15  -1.267111898726718
    
```

Gambar 6. Model yang dibangun dari algoritme MLP

Algoritme PART

```

X9 = 2: TOLAK (189.0/8.0)
X15 > 0.01: SETUJU (70.0)
X11 > 0.05: SETUJU (43.0/3.0)

X7 = 2 AND
X4 = 1: SETUJU (19.0)

X7 = 3 AND
X14 > 0.08: TOLAK (6.6)

X7 = 3: SETUJU (4.4/0.4)

X7 = 2 AND
X8 <= 0.16: SETUJU (4.0/1.0)

X7 = 1 AND
X14 <= 0.05 AND
X4 = 1: SETUJU (18.0/1.0)

X7 = 1 AND
X15 <= 0 AND
X11 <= 0: TOLAK (30.0/9.0)

X7 = 1 AND
X15 <= 0: SETUJU (11.0/4.0)

X7 = 1 AND
X14 > 0.09: SETUJU (4.0)

X7 = 2: TOLAK (4.0)

X7 = 1 AND
X14 <= 0.08 AND
X8 > 0.05: TOLAK (3.0/1.0)

X7 = 1 AND
X14 > 0.08: TOLAK (3.0)

X8 <= 0.02: TOLAK (4.0)
    
```

Gambar 7. Model yang dibangun oleh algoritme PART

Perbandingan hasil klasifikasi dari ketiga algoritme J48, MLP dan PART dapat dilihat pada Tabel.

Tabel 5. Tanpa *attribute selection*

C4.5	Multilayer Perceptron	PART
85,09%	84,13%	87,50%

Tabel 6. Menggunakan *attribute selection*

C4.5	Multilayer Perceptron	PART
89,18%	85,82%	87,98%

Hasil klasifikasi menunjukkan bahwa ada perbedaan hasil akurasi antara dilakukannya seleksi atribut dan tidak ada seleksi atribut. Tanpa seleksi atribut untuk algoritme PART mendapatkan akurasi yang paling tinggi yaitu 87,50% diikuti dengan algoritme C4.5 sebesar 85,09% dan terakhir algoritme MLP sebesar 84,13%. Setelah itu dilakukan seleksi atribut sehingga mendapatkan akurasi yang cukup meningkat yaitu masing-masing algoritme PART 87,98% , C4.5 89,18% dan MLP 85,82%. Dimana dengan seleksi atribut algoritme yang akurasinya tertinggi adalah C4.5 (J48), hal ini dikarenakan C4.5 menggunakan *decision tree* untuk membangun model klasifikasinya.

Dengan mempertimbangkan hasil akurasi dan model yang dibentuk, maka pada kasus data set persetujuan pengajuan kredit dapat menggunakan hasil klasifikasi dan model dari algoritme C4.5 atau J48.

5. Implementasi Hasil

Untuk memprediksi data baru yang belum mempunyai klas, dimana data baru adalah pengajuan kredit baru yang harus diambil keputusannya TOLAK atau SETUJU. Untuk itu digunakan model *tree* dari C4.5 untuk melakukan prediksi terhadap data yang baru masuk tersebut. Model Tree C4.5.

=== Model information ===

```

Filename: J48.model
Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2
Relation: Train-weka.filters.supervised.attribute.AttributeSelection-Eweka.attributeSelection.CfsSubsetEval -P 1 -E 1-Sweka.attributeSelection.BestFirst -D 1 -N 5
Attributes: 8
          X4
          X7
          X8
          X9
          X11
          X14
          X15
          Class
    
```

=== Classifier model ===

J48 pruned tree

```

-----
X9 = 1
| X15 <= 0.01
| | X11 <= 0.05
| | | X7 = 1
| | | | X14 <= 0.05
| | | | | X4 = 1: SETUJU (18.0/1.0)
| | | | | X4 = 2
| | | | | X14 <= 0.02: TOLAK (2.0)
| | | | | X14 > 0.02: SETUJU (3.0/1.0)
| | | | | X4 = 3: SETUJU (0.0)
| | | | | X4 = 4: SETUJU (0.0)
| | | | X14 > 0.05
| | | | | X15 <= 0
| | | | | X11 <= 0: TOLAK (27.0/8.0)
| | | | | X11 > 0: SETUJU (11.0/4.0)
| | | | X15 > 0
| | | | | X14 <= 0.09
| | | | | X14 <= 0.08: SETUJU (3.0/1.0)
| | | | | X14 > 0.08: TOLAK (3.0)
| | | | | X14 > 0.09: SETUJU (4.0)
| | | X7 = 2
| | | | X4 = 1: SETUJU (19.0)
| | | | X4 = 2
| | | | | X8 <= 0.16: SETUJU (4.0/1.0)
| | | | | X8 > 0.16: TOLAK (4.0)
| | | | X4 = 3: SETUJU (0.0)
| | | | X4 = 4: SETUJU (0.0)
| | | X7 = 3
| | | | X14 <= 0.08: SETUJU (4.4/0.4)
| | | | X14 > 0.08: TOLAK (6.6)
| | | X7 = 4: TOLAK (1.0)
| | | X7 = 5: SETUJU (0.0)
| | | X7 = 6: SETUJU (1.0)
| | | X7 = 7: SETUJU (0.0)
| | | X7 = 8: TOLAK (3.0)
| | | X7 = 9: SETUJU (0.0)
| | X11 > 0.05: SETUJU (43.0/3.0)
| X15 > 0.01: SETUJU (70.0)
X9 = 2: TOLAK (189.0/8.0)
    
```

Number of Leaves : 26

Size of the tree : 40

=== Re-evaluation on test set ===

User supplied test set

Relation: Test

Instances: unknown (yet). Reading incrementally

Attributes: 8

=== Predictions on user test set ===

inst#	actual	predicted	error	prediction
1	1:?	1:SETUJU	0.93	
2	1:?	1:SETUJU	1	
3	1:?	1:SETUJU	1	
4	1:?	1:SETUJU	1	
.				
.				
.				
269	1:?	2:TOLAK	0.958	
270	1:?	2:TOLAK	0.958	
271	1:?	2:TOLAK	0.958	
272	1:?	2:TOLAK	0.958	
273	1:?	2:TOLAK	0.958	
274	1:?	2:TOLAK	0.958	

Dari 274 data berdasarkan data pengajuan kredit baru didapatkan keputusan SETUJU sebanyak 113 obyek dan TOLAK sebanyak 161 obyek.

6. Kesimpulan

Teknik *data mining* digunakan untuk menggali pengetahuan dari data *training* untuk diambil informasinya guna pengambilan keputusan. Pada data set Bank "X" didapatkan pola pengetahuan dari *decision tree* yang mudah dipahami, yang memudahkan untuk pengambilan keputusan apakah kredit dari nasabah diterima atau ditolak. Dengan mereduksi atribut yang diperlukan didapatkan klasifikasi yang mempunyai akurasi yang tinggi, yaitu menggunakan algoritme J48 (C4.5) sebesar 89,18%. Dari data pengujian yang diberikan berupa data nasabah sejumlah 274 nasabah yang akan ditentukan, didapatkan 113 nasabah disetujui kreditnya dan 161 nasabah ditolak.

Daftar Pustaka

- [1] Rafika Lihani, Ngadiman, Nurhasan Hamidi, "Analisis Manajemen Kredit Guna Meminimalkan Risiko Kredit (Studi pada PD BPR BKK Tasikmadu Karanganyar)," *Jupe UNS, Vol 1, No. 3 Hal 1 s/d 11, Juli, 2013*.
- [2] R. Agrawal, J. Gehrke, D. Gunopilos, and P. Raghavan, "Automatic subspace clustering of high dimensional data for data mining applications," in SIGMOD '98, 1998, pp. 94-105.
- [3] Han, J., and M. Kamber, "Data mining: Concepts and techniques. San Francisco" Morgan Kaufmann. 2001.
- [4] E.N. Jannah, R.W. Herlambang, M.R.Pangow, " Penentuan Persetujuan Pemberian Kredit Bank Menggunakan Metode Hybrid SVM ", *Data Mining Contest – Gemastik 2011, ITS Surabaya, Oktober 2011*.
- [5] Y. Cao and J. Wu, "Dynamics of Projective Adaptive Resonance Theory Model: The Foundation of PART Algorithm", *IEEE Transactions On Neural Networks, vol. XX, no. XX, 2004*.
- [6] William J. Frawley, Gregory Piatetsky-Shapiro, and Christopher J. Matheus, "Knowledge Discovery in Databases : An Overview", *AI Magazine, Volume 13, Number 3, 1992*
- [7] W. Nor Haizan W. Mohamed, Mohd Najib Mohd Salleh, Abdul Halim Omar, "A Comparative Study of Reduced Error Pruning Method in Decision Tree Algorithms", *IEEE International Conference on Control System, Computing and Engineering, 23 - Penang, Malaysia , 25 Nov. 2012*
- [8] <http://www.cs.waikato.ac.nz/ml/weka/documentation.html> (diakses terakhir 18 Nopember 2013)s

Biodata Penulis

Hendra Marcos, S.T., memperoleh gelar Sarjana Teknik (S.T.), Jurusan Teknik Elektro, lulus tahun 2002. Sedang menempuh studi S2 Program Pasca Sarjana, Program Studi Teknik Elektro dan Teknologi Informatika Universitas Gajah Mada Yogyakarta, masuk tahun 2012. Saat ini menjadi Dosen di STMIK AMIKOM Purwokerto.

Indriana Hidayah, S.T., M.T., memperoleh gelar Sarjana Teknik (S.T.), Jurusan Teknik Informatika Universitas Gajah Mada Yogyakarta, lulus tahun 2000.

Memperoleh gelar Magister Teknik (M.T.) Program Pasca Sarjana Magister Teknik Informatika Universitas Gajah Mada Yogyakarta, lulus tahun 2010. Saat ini menjadi Dosen UGM Yogyakarta.