MODUL DATA MINING

UNSUPERVISED LEARNING
PERTEMUAN 10 (ONLINE)

Disusun Oleh
Syefira Salsabila

Kata *Mining* merupakan kiasan dari bahasa inggris, mine. Jika mine berarti menambang sumber daya yang tersembunyi di dalam tanah, maka Data Mining merupakan penggalian makna yang tersembunyi dari kumpulan data yang sangat besar. Karena itu *Data Mining s*ebenarnya memiliki akar yang panjang dari bidang ilmu seperti kecerdasan buatan (*artificial intelligent*), m*achine learning*, statistik dan basis Data.

Data Mining dibagi menjadi beberapa kelompok berdasarkan tugas yang dapat dilakukan, yaitu:

1. Classification
   Suatu teknik dengan melihat pada kelakuan dan atribut dari kelompok yang telah didefinisikan. Teknik ini dapat memberikan klasifikasi pada data baru dengan memanipulasi data yang ada yang telah diklasifikasi dan dengan menggunakan hasilnya untuk memberikan sejumlah aturan. Salah satu contoh yang mudah dan popular adalah dengan Decision tree yaitu salah satu metode klasifikasi yang paling populer karena mudah untuk diinterpretasi. Decision tree adalah model prediksi menggunakan struktur pohon atau struktur berhirarki.

2. Association
   Digunakan untuk mengenali kelakuan dari kejadian-kejadian khusus atau proses dimana hubungan asosiasi muncul pada setiap kejadian. Salah satu contohnya adalah Market Basket Analysis, yaitu salah sati metode asosiasi yang menganalisa kemungkinan pelanggan untuk membeli beberapa item secara bersamaan.

3. Clustering
   Digunakan untuk menganalisis pengelompokkan berbeda terhadap data, mirip dengan klasifikasi, namun pengelompokkan belum didefinisikan sebelum dijalankannya tool data mining. Biasanya menggunkan metode *neural network* atau statistik. Clustering membagi item menjadi kelompok-kelompok berdasarkan yang ditemukan tool data mining.

Metode pelatihan yaitu cara berlangsungnya pembelajaran dan pelatihan dalam menganalisis dataset untuk mendapatkan pola data tertentu. Metode pelatihan data mining memiliki 3 kelompok, seperti : *supervised learning*, *unsupervised learning*, dan *association learning*. 3 kelompok tersebut memiliki definisi, sebagai berikut.

1. *Supervised Learning*
   Kumpulan record dari inputan yang digunakan dan telah diketahui output , dengan kata lain variable yang menjadi target telah ditentukan dalam *dataset* yang sedang dianalisis. Sebagian besar algoritma dalam kelompok tersebut terdiri dari : klasifikasi, estimasi, dan prediksi. Algoritma yang digunakan akan melakukan *process* pembelajaran yang berdasarkan *value* dari *variable* sasaran yang telah terasosiasi dengan *value* pada varibel *predictor*.

2. *Unsupervised Learning*
   Pada metode tersebut data yang dianalisa diterapkan tanpa adanya guru serta pelatihan pada data lampau, dengan kata lain diartikan sebagai pencarian pola pada setiap atribut yang digunakan. Tidak termasuk penetapan atribut atau kelas pada sasaran. Contoh algoritma yang menerapkan metode *unsupervised learning* adalah Clustering.
3. *Association Learning*
   Berbeda dengan dua kelompok yang terdapat di atas, pada mode ini mempunyai tujuan untuk mencari atribut yang muncul pada transaksi yang sama. Algoritma asosiasi biasanya berfungsi untuk mencari dan menganalisa transaksi belanja dengan konsep mencari produk yang dibeli secara bersamaan dalam satu transaksi yang sama. Algoritma yang digunakan dalam kelompok asosiasi adalah Apriori.

Dalam dunia data mining atau data science sering kali kita mendengar *supervised* dan *unsupervised learning*. Secara garis besar terdapat 2 pendekatan untuk melakukan teknik - teknik data mining. ***Supervised learning*** **adalah sebuah pendekatan dimana sudah terdapat data yang dilatih, dan terdapat variable yang ditargetkan sehingga tujuan dari pendekatan ini adalah mengkelompokan suatu data ke data yang sudah ada,** lain halnya dengan unsupervised learning, ***unsupervised learning*** **tidak memiliki data latih, sehingga dari data yang ada, kita mengelompokan data tersebut menjadi 2 bagian atau 3 bagian dan seterusnya.**

S*upervised learning* bertujuan untuk menemukan pola baru dalam data dengan menghubungkan pola data yang sudah ada dengan data yang baru. Sedangkan pada *unsupervised learning*, data belum memiliki pola apapun, dan tujuan *unsupervised learning* untuk menemukan pola dalam sebuah data. Contoh Supervised Learning adalah ketika Anda memiliki sejumlah buku yang sudah dilabeli dengan kategori tertentu. Misalnya, kategori buku novel seperti Digital Fortress, Inferno, Deception Point. Kategori buku akademik, seperti Pengantar Teknologi Informasi, R in Action, Rekayasa Perangkat Lunak. Kategori biografi antara lain Anne Frank, Abraham Lincoln dan Mandela. Selanjutnya, ketika Anda membeli sejumlah buku baru, maka Anda harus mengindentifikasi isi dari buku tersebut, dan memasukannya dalam kategori. Ketika Anda membeli buku Logika fuzzy, Anda pasti akan memasukan buku tersebut ke dalam buku akademik.

Lain halnya dengan *Unsupervised Learning.* Anda tidak memiliki data yang dilatih sebelumnya. Anggaplah Anda belum pernah membeli buku sama sekali, namun dalam satu hari, Anda membeli banyak tumpukan buku dan ingin membaginya kedalam beberapa kategori agar nantinya mudah dicari. Anda akan mengidentifikasi buku buku mana yang mirip. Dalam hal ini, kita memilih pendekatan buku berdasarkan isinya. Misalnya anda memiliki buku Digital Fortress, Inferno, Deception Point, Pengantar Teknologi Informasi, R in Action, Rekayasa Perangkat Lunak, Anne Frank, Abraham Lincoln dan Mandela. Anda akan mengklasifikasikan buku Pengantar Teknologi Informasi, R in Action, Rekayasa Perangkat Lunak Anda ke dalam buku akademik karena keperluannya untuk kuliah.

Untuk melakukan hal itu Anda perlu algoritma yang mendukung untuk pengimplementasian dari metode tersebut.

Algoritma Supervised Learning:

a. Decision tree
b. Nearest - Neighbor Classifier
c. Naive Bayes Classifier
d. Artificial Neural Network
e. Support Vector Machine
f. Fuzzy K-Nearest Neighbor

Algoritma Unsupervised Learning

a. K-Means
b. Hierarchical Clustering
c. DBSCAN
d. Fuzzy C-Means
e. Self-Organizing Map

Kesimpulannya dari penjelasan di atas adalah jika anda memiliki data data sebelumnya dan memiliki variabel target yang akan diklasifikasikan, maka Anda dapat memakai metode *supervised learning.* Jika Anda ingin membagi data - data tersebut ke dalam beberapa kelompok maka Anda memakai metode u*nsupervised learning*

A. Decision Tree

Seperti diketahui bahwa manusia selalu menghadapi berbagai macam masalah di dalam kehidupannya sehari-hari. Masalah-masalah yang timbul dari berbagai macam bidang ini memiliki tingkat kesulitan dan kompleksitas yang sangat bervariasi, mulai dari masalah yang sangat sederhana dengan sedikit faktor-faktor terkait hingga masalah yang sangat rumit dengan banyak sekali faktor-faktor yang terkait, sehingga factor-faktor yang berkaitan dengan masalah tersebut perlu untuk diperhitungkan.

Seiring dengan perkembangan kemajuan pola pikir manusia, manusia mulai mengembangkan sebuah sistem yang dapat membantu manusia dalam menghadapi masalah-masalah yang timbul sehingga dapat menyelesaikannya dengan mudah. Pohon keputusan atau yang lebih dikenal dengan istilah *Decision Tree* ini merupakan implementasi dari sebuah sistem yang manusia kembangkan dalam mencari dan membuat keputusan untuk masalah-masalah tersebut dengan memperhitungkan berbagai macam faktor yang berkaitan di dalam lingkup masalah tersebut.

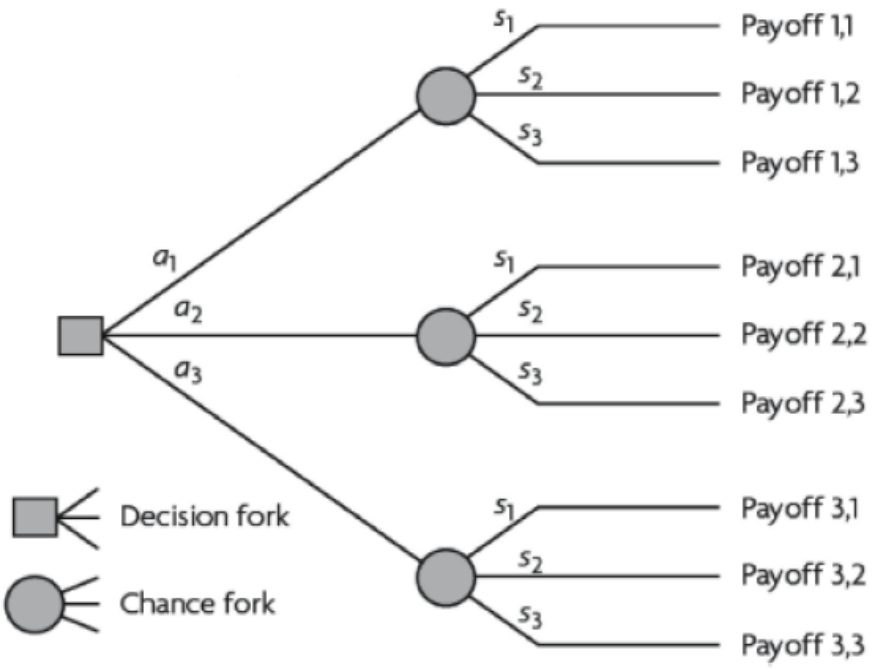
Dengan pohon keputusan, manusia dapat dengan mudah mengidentifikasi dan melihat hubungan antara faktor-faktor yang mempengaruhi suatu masalah sehingga dengan memperhitungkan faktor-faktor tersebut dapat dihasilkan penyelesaian terbaik

untuk masalah tersebut. Pohon keputusan ini juga dapat menganalisa nilai risiko dan nilai suatu informasi yang terdapat dalam suatu alternatif pemecahan masalah.

Pohon keputusan dalam analisis pemecahan masalah pengambilan keputusan merupakan pemetaan alternatif-alternatif pemecahan masalah yang dapat diambil dari masalah tersebut. Pohon keputusan juga memperlihatkan faktor-faktor kemungkinan yang dapat mempengaruhi alternative-alternatif keputusan tersebut, disertai dengan estimasi hasil akhir yang akan didapat bila kita mengambil alternatif keputusan tersebut.

Secara umum, pohon keputusan adalah suatu gambaran permodelan dari suatu persoalan yang terdiri dari serangkaian keputusan yang mengarah kepada solusi yang dihasilkan. Peranan pohon keputusan sebagai alat bantu dalam mengambil keputusan telah dikembangkan oleh manusia sejak perkembangan teori pohon yang dilandaskan pada teori graf. Seiring dengan perkembangannya, pohon keputusan kini telah banyak dimanfaatkan oleh manusia dalam berbagai macam sistem pengambilan keputusan.

Decision tree adalah struktur flowchart yang menyerupai tree (pohon), dimana setiap simpul internal menandakan suatu tes pada atribut, setiap cabang merepresentasikan hasil tes, dan simpul daun merepresentasikan kelas atau distribusi kelas. Alur pada decision tree di telusuri dari simpul akar ke simpul daun yang memegang prediksi.



Gambar 4.1 Bentuk Decision Tree Secara Umum

a) Algoritma c4.5

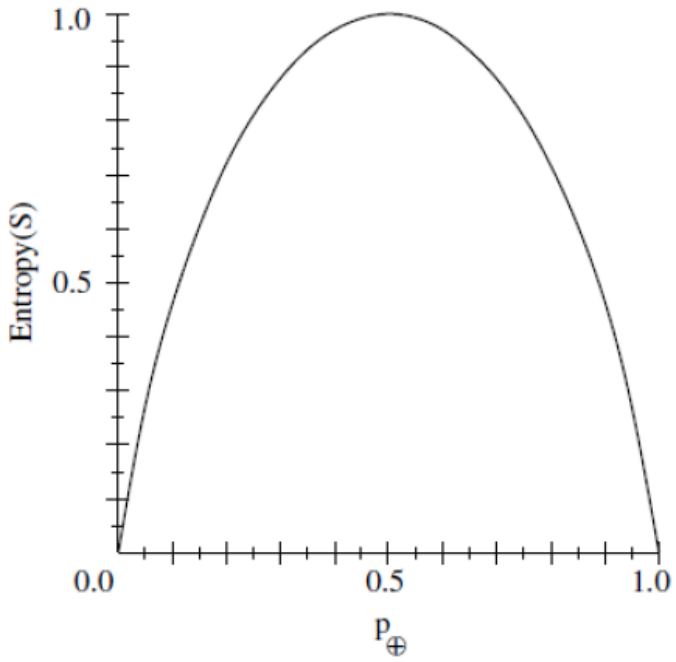
Pohon keputusan merupakan metode yang umum digunakan untuk melakukan klasifikasi pada data mining. Seperti yang telah dijelaskan sebelumnya, klasifikasi merupakan Suatu teknik menemukan kumpulan pola atau fungsi yang mendeskripsikan serta memisahkan kelas data yang satu dengan yang lainnya untuk menyatakan objek tersebut masuk pada kategori tertentu dengan melihat pada kelakuan dan atribut dari kelompok yang telah didefinisikan.

Metode ini popular karena mampu melakukan klasifikasi sekaligus menunjukkan hubungan antar atribut. Banyak algoritma yang dapat digunakan untuk membangun suatu decision tree, salah satunya ialah algoritma C45.

Algoritma C4.5 dapat menangani data numerik dan diskret. Algoritma C.45 menggunakan rasio perolehan (gain ratio). Sebelum menghitung rasio perolehan, perlu dilakukan perhitungan nilai informasi dalam satuan bits dari suatu kumpulan objek, yaitu dengan menggunakan konsep entropi.

Konsep Entropy

Entropy(S) merupakan jumlah bit yang diperkirakan dibutuhkan untuk dapat mengekstrak suatu kelas (+ atau -) dari sejumlah data acak pada ruang sampel S. Entropy dapat dikatakan sebagai kebutuhan bit untuk menyatakan suatu kelas. semakin kecil nilai Entropy maka akan semakin Entropy digunakan dalam mengekstrak suatu kelas. Entropi digunakan untuk mengukur ketidakaslian S.



Gambar 4.2 Grafik Entropi

Besarnya Entropy pada ruang sampel S didefinisikan dengan:

$$Entropy(S) \equiv -p_{\oplus} \log_2 p_{\oplus} - p_{\ominus} \log_2 p_{\ominus}$$

Dimana:

- S : ruang (data) sampel yang digunakan untuk pelatihan
- $p_{\oplus}$: jumlah yang bersolusi positif atau mendukung pada data sampel untuk kriteria tertentu
- $p_{\ominus}$: jumlah yang bersolusi negatif atau tidak mendukung pada data sampel untuk kriteria tertentu.

- Entropi(S) = 0, jika semua contoh pada S berada dalam kelas yang sama.
- Entropi(S) = 1, jika jumlah contoh positif dan negative dalam S adalah sama.
- 0 > Entropi(S) > 1, jika jumlah contoh positif dan negative dalam S tidak sama.

Konsep Gain

Gain (S,A) merupakan Perolehan informasi dari atribut A relative terhadap output data S. Perolehan informasi didapat dari output data atau variabel dependent S yang dikelompokkan berdasarkan atribut A, dinotasikan dengan gain (S,A).

$$Gain(S,A) \equiv Entropy(S) - \sum_{i=1}^{n} \frac{|Si|}{|S|} * Entropy(Si)$$

Dimana:

- A : Atribut
- S : Sampel
- n : Jumlah partisis himpunan atribut A
- $|S_i|$: Jumlah sampel pada pertisi ke –i
- $|S|$: Jumlah sampel dalam S

Untuk memudahkan penjelasan mengenai algoritma C4.5berikut ini disertakan contoh kasus yang dituangkan dalam Tabel 4.1:

Tabel 4.1 Keputusan Bermain Tenis

No	OUTLOOK	TEMPERATURE	HUMIDITY	WINDY	PLAY
1	Sunny	Hot	High	FALSE	No
2	Sunny	Hot	High	TRUE	No
3	Cloudy	Hot	High	FALSE	Yes
4	Rainy	Mild	High	FALSE	Yes
5	Rainy	Cool	Normal	FALSE	Yes
6	Rainy	Cool	Normal	TRUE	Yes
7	Cloudy	Cool	Normal	TRUE	Yes
8	Sunny	Mild	High	FALSE	No
9	Sunny	Cool	Normal	FALSE	Yes
10	Rainy	Mild	Normal	FALSE	Yes
11	Sunny	Mild	Normal	TRUE	Yes
12	Cloudy	Mild	High	TRUE	Yes
13	Cloudy	Hot	Normal	FALSE	Yes
14	Rainy	Mild	High	TRUE	No

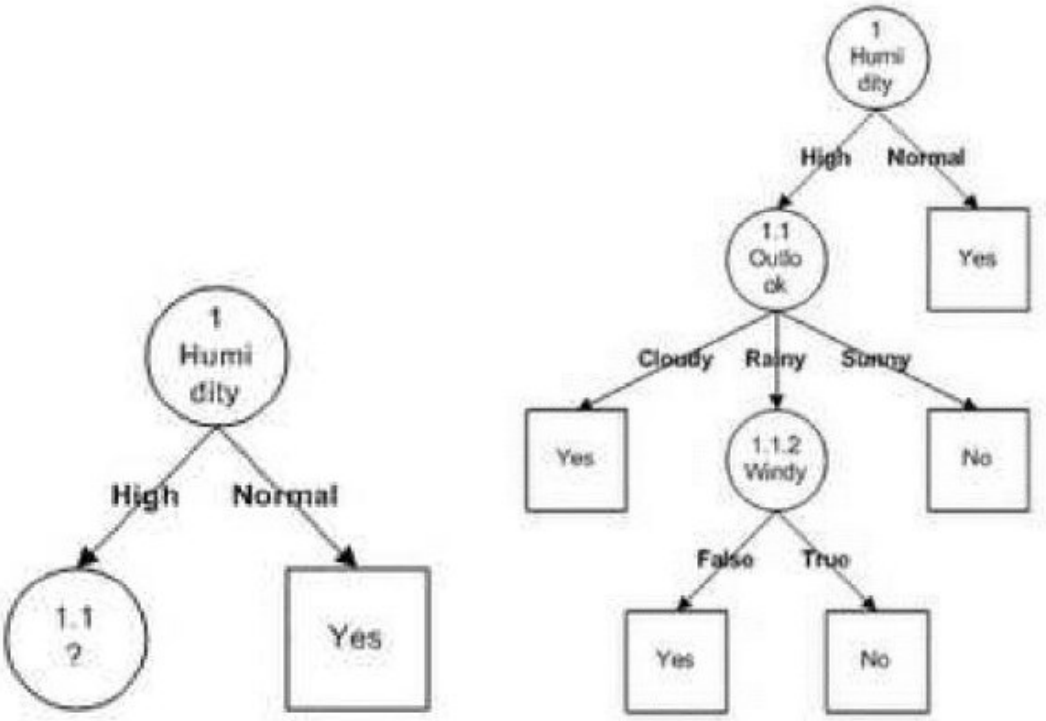
Tabel 1 merupakan kasus yang akan dibuat pohon keputusan untuk menentukan main tenis atau tida. Data ini memiliki atribut-atribut yaitu, keadaan cuaca (outlook), temperatur, kelembaban (humidity) dan keadaan angin (windy).

Berikut merupakan cara membangun pohon keputusan dengan menggunakan algoritma:

1. Pilih atribut sebagai akar. Sebuah akar didapat dari nilai gain tertinggi dari atribut-atribut yang ada.
2. Buat cabang untuk masing-masing nilai
3. Bagi kasus dalam cabang
4. Ulangi proses untuk masing-masing cabang sampai semua kasus pada cabang memiliki kelas yang sama.

Tabel 4.2 Perhitungan Simpul 1

NODE			JUMLAH KASUS	NO (S_1)	YES (S_2)	ENTROPY	GAIN
1	TOTAL		14	4	10	0.863120569	
	OUTLOOK						0.258521037
		CLOUDY	4	0	4	0	
		RAINY	5	1	4	0.721928095	
		SUNNY	5	3	2	0.970950594	
	TEMPERATURE						0.183850925
		COOL	4	0	4	0	
		HOT	4	2	2	1	
		MILD	6	2	4	0.918295834	
	HUMIDITY						0.370506501
		HIGH	7	4	3	0.985228136	
		NORMAL	7	0	7	0	
	WINDY						0.005977711
		FALSE	8	2	6	0.811278124	
		TRUE	6	4	2	0.918295834	



Dari hasil pada Tabel 4.2 dapat diketahui bahwa atribut dengan Gain tertinggi adalah HUMIDITY yaitu sebesar 0.37. Dengan demikian HUMIDITY dapat menjadi node akar. Ada 2 nilai atribut dari HUMIDITY yaitu HIGH dan NORMAL. Dari kedua nilai atribut tersebut, nilai atribut NORMAL sudah mengklasifikasikan kasus menjadi 1 yaitu keputusan-nya Yes, sehingga tidak perlu dilakukan perhitungan lebih lanjut, tetapi untuk nilai atribut HIGH masih perlu dilakukan perhitungan lagi hingga semua kasus masuk dalam kelas seperti yang terlihat pada Gambar di sebelah kanan.

Kelebihan Pohon Keputusan

Dalam membuat keputusan dengan menggunakan pohon keputusan, metode ini memiliki kelebihan sebagai berikut:

a. Daerah pengambilan keputusan lebih simpel dan spesifik.
b. Eliminasi perhitungan-perhitungan tidak diperlukan, karena ketika menggunakan metode pohon keputusan maka sample diuji hanya berdasarkan kriteria atau kelas tertentu.
c. Fleksibel untuk memilih fitur dari internal node yang berbeda. Sehingga dapat meningkatkan kualitas keputusan yang dihasilkan jika dibandingkan ketika menggunakan metode penghitungan satu tahap yang lebih konvensional.
d. Dengan menggunakan pohon keputusan, penguji tidak perlu melakukan estimasi pada distribusi dimensi tinggi ataupun parameter tertentu dari

distribusi kelas tersebut. Karena metode ini menggunakan kriteria yang jumlahnya lebih sedikit pada setiap node internal tanpa banyak mengurangi kualitas keputusan yang dihasilkan.

Kekurangan Pohon Keputusan

Pohon keputusan sangat membantu dalam pengambilan keputusan, namun pohon keputusan juga memiliki beberapa kekurangan, diantaranya:

a. Kesulitan dalam mendesain pohon keputusan yang optimal.
b. Hasil kualitas keputusan yang didapat sangat tergantung pada bagaimana pohon tersebut didesain. Sehingga jika pohon keputusan yang dibuat kurang optimal, maka akan berpengaruh pada kualitas dari keputusan yang didapat.
c. Terjadi overlap terutama ketika kelas-kelas dan criteria yang digunakan jumlahnya sangat banyak sehingga dapat menyebabkan meningkatnya waktu pengambilan keputusan dan jumlah memori yang diperlukan.
d. Pengakumulasian jumlah error dari setiap tingkat dalam sebuah pohon keputusan yang besar.

Isu Terkait Decision Tree

Sekali decision tree dibangun berdasarkan objek yang dimiliki dalam data latih, maka pohon tersebut sebenarnya alami dan sedikit atau banyak sudah merefleksikan objek-objek tersebut. Biasanya banyak cabang yang secara kuat dipengaruhi anomali data (data yang menyimpang) yang mungkin ada di set data. Data seperti ini disebut noise atau outlier. Data-data yang menyimpang seperti ini sebaiknya sudah dilakukan pemangkasan di awal pemrosesan sehingga tidak mempengaruhi kinerja algoritme utama yang digunakan dalam data mining. Secara prinsip, jika pohon dibangun dari data mentah yang belum mengalami pemrosesan awal sama sekali, maka dipastikan bahwa decision tree secara penuh merefleksikan semua isi set data latih. Karena decision tree dibangun untuk menyelesaikan kasus klasifikasi hingga sisa terkecil, maka decision tree bisa mengalami keadaan yang di atas normal. Di sisi lain, jika decision tree yang dibangun terlalu simpel terhadap data latih yang digunakan untuk membangunnya maka akan mengalami keadaan yang di bawah normal terhadap data latih. Untuk menangani masalah tersebut, maka diperlukan adanya pemrosesan pemangkasan (pruning) cabang yang memberikan informasi redundan berulang), atau yang tidak mengikuti pola data umumnya. Dengan cara ini, maka akan didapatkan pohon yang tidak terlalu ‘rindang; tetapi lebih besar skalabilitas dan kecepatan prediksinya.

Ada dua jenis pemangkasan dalam decision tree:

a. Pre-pruning
   Pendekatan ini berarti bahwa secara praktik akan menghentikan ‘pertumbuhan’ selama proses induksi pohon dengan memilih berhenti pada sebuah node, yang kemudian node tersebut akan menjadi daun dan diberikan label kelas sesuai dengan elemen data terbanyak. Syarat utama penggunaan pendekatan ini adalah bahwa semua objek data dimiliki oleh kelas yang sama atau semua nilai fiturnya sama

b. Post-pruning
   Pendekatan ini digunakan setelah pohon tumbuh lengkap. Pendekatan 'bottom-up' didasarkan pada nilai error prediksi. Node akan dipangkas dengan membuang cabang. Akibatnya, node menjadi daun dan diberi label kelas sesuai dengan elemen data terbanyak. Error prediksi dapat dikurangi dengan cara ini.

B. Naïve Bayes

Naïve Bayes adalah teknik prediksi berbasis probabilistik sederhana yang berdasar pada penerapan teorema Bayes (aturan Bayes) dengan sebuah asumsi independensi (ketidaktergantungan) yang kuat (naif). Dapat dikatakan, pada *Naïve Bayes* model yang digunakan adalah "model fitur independen". Dalam Bayes (terutama *Naïve Bayes*), makna independensi yang kuat pada fitur adalah bahwa sebuah fitur dalam suatu data tidak berkaitan dengan ada atau tidaknya fitur lain dalam data yang sama.

Teori keputusan *bayes* adalah pendekatan statistik yang fundamental dalam pengenalan pola (*pattern recognition)* , pendekatan ini didasarakan pada kuantifikasi *trade-off* antara berbagai keputusan klasifikasi dengan menggunakan probabilitas dan ongkos yang di timbulkan dalam keputusan tersebut. Selain itu *Bayesian clasification* juga dapat memprediksi probabilitas keanggotaan suatu *class*. pada teorema *bayes* yang memiliki kemampuan klasifikasi serupa dengan *decision tree* dan *neural network* . *Bayesian classification* terbukti memiliki akurasi dan kecepatan yang tinggi saat diaplikasikan ke dalam *database* dengan data yang besar.

Algoritma *naïve* Bayes merupakan salah satu algoritma yang terdapat pada teknik klasifikasi. *Naive* Bayes merupakan pengklasifikasian dengan metode probabilitas dan statistik yang dikemukan oleh ilmuwan Inggris *Thomas* Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai teorema Bayes. Teorema tersebut dikombinasikan dengan *naive* di mana diasumsikan kondisi antar atribut saling bebas. Klasifikasi *naive* Bayes diasumsikan bahwa ada atau tidak ciri tertentu dari sebuah kelas tidak ada hubungannya dengan ciri dari kelas lainnya.

Naive Bayes digunakan untuk memprediksi peluang dimasa yang akan datang berdasarkan pengalaman pada waktu lampau, sehingga disebut dengan Teorema Bayes. Teorema ini digabungkan dengan Naive yang mengasumsikan kondisi antar elemen saling bebas. *Classification* Naive Bayes diupayakan bahwa ada atau tidak ada ciri tertentu dari sebuah *class* tidak ada hubungannya dengan ciri dari kelas lainnya.

Kecerdasan Buatan (*Artificial Intelligence*) merupakan salah satu bagian dari ilmu komputer yang mempelajari bagaimana membuat mesin (komputer) dapat melakukan pekerjaan yaitu seperti dan sebaik yang dilakukan oleh manusia bahkan bisa lebih baik dari yang dilakukan manusia. Salah satu penerapan yaitu pada sistem pakar (*Expert System).* Sistem pakar adalah suatu sistem yang berbasis komputer yang menggunakan

pengetahuan, fakta serta teknik penalaran dalam memecahkan masalah yang biasanya hanya dapat dipecahkan oleh seorang pakar dalam bidang tersebut.

Salah satu penerapan dalam sistem pakar ada pada bidang kesehatan yaitu sitem pakar penyakit pada ibu hamil menggunakan pendekatan metode naïve bayes yaitu menetukan penyakit berdasarkan gejala umum yang diderita oleh seorang ibu hamil serta dengan menghitung peluang seorang ibu hamil mengidap penyakit kehamilan dan memberikan memberikan solusi pencegahan berdasarkan pada jenis penyakit yang diderita. Penerapan system pakar yang berguna untuk mendapatkan informasi tentang awal penyakit yang dialami ibu hamil dan apakah system pakar dengan metode *Naïve Bayes* ini dapat digunakan untuk mendaptkan informasi tentang awal penyakit yang terjadi pada masa kehamilan yang diakibatkan oleh gangguan yang muncul akibat kehamilan tersebut.

Metode *Naïve Bayes Classifier* merupakan pengklasifikasi probabilitas yang sederhana yang diberdasarkan pada teorema Bayes. Teorema Bayes dikombinasi dengan "*Naïve*" yang artinya setiap atribut atau variabel bersifat bebas (*independent*). *Naïve Bayes Classifier* bisa dilatih dengan efisien dalam suatu pembelajaran terawasi *(supervised learning)*. Keuntungan klasifikasi ini adalah hanya membutuhkan jumlah kecil data pelatihan yang berguna untuk memperkirakan parameter (sarana dan varians dari variabel) yang diperlukan dalam klasifikasi. Karena variabel independen diasumsikan, hanya variasi dari variabel untuk masing-masing kelas harus ditentukan, bukan seluruh matriks kovarians. *Naive Baye Classifier* tersebut mengestimasi peluang kelas bersyarat dengan mengasumsikan atributnya yaitu independen secara bersyarat yang diberikan label dengan kelas y. Teorema perhitungan *Naive Bayes Classifire* berdasarkan probabilitas sebagai berikut

Persamaan dari teorema Bayes adalah

$$P(C \mid F) = \frac{P(C).P(F \mid C)}{P(F)} \quad (1)$$

Keterangan :

$P(C|F)$:Probabilitas akhir bersyarat (*posterior*) suatu Kelas C terjadi jika diberikan Petunjuk (atribut) F terjadi

$P(C)$:Probabilitas awal (*prior*) Kelas C terjadi tanpa memandang petunjuk (atribut) apapun

$P(F|C)$:Probabilitas sebuah petunjuk (atribut) F terjadi akan mempengaruhi Kelas C

$P(F)$:Probalilitas awal (*prior*) petunjuk (atribut) F terjadi tanpa memandang Kelas apapun

Adapun alur dari metode Naive Bayes, sebagai berikut:

1. Membaca data training
2. Hitung Jumlah dan probabilitas, namun apabila data numerik maka.
   a. Mencari *value* rata-rata dan standar deviasi dari tiap-tiap parameter yang merupakan data berangka. Adapun kesamaan yang menggunakan dalam nilai rata-rata hitung (*mean*) dapat dijabarkan sebagai berikut.

$$\mu = \frac{\sum_{i=1}^{n} x_i}{n} \quad (2.2)$$

dimana :

μ : rata-rata hitung (*mean*)

X_i: nilai sample ke-i

n : jumlah sample

Dan persamaan dalam hitung *value* simpangan baku (standar deviasi), sebagai berikut.

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \mu)^2}{n-1}} \qquad (2.4)$$

dimana :

σ : standar Deviasi

X_i : nilai x ke-i

μ : rata-rata hitung (*mean*)

n : jumla sample

   b. Mencari *value* probabilitas dengan langkah hitung jumlah data yang sesuai dari bilangan yang sama dibagi dengan jumlah data pada kategori tersebut.

3. Memperoleh *value* dalam tabel rata-rata, standar deviasi dan peluang.

Kelebihan *Naive Bayes*:

1. Mudah diimplementasikan
2. Hasilnya *robust* untuk data yang memuat *noisy* dan untuk data yang tidak berkaitan
3. Dapat menangani *missing value*
4. Hasilnya cukup baik untuk sebagian besar kasus

Kekurangan *Naive Bayes*:

1. Adanya asumsi saling bebas antar atributnya terkadang akan menurunkan tingkat akurasinya

2. Biasanya dalam kehidupan nyata selalu ada hubungan antar atribut sehingga asumsi saling bebas menjadi tidak dipenuhi

DAFTAR PUSTAKA

Bustami. (2013). Penerapan Algoritma Naïve Bayes Untuk Mengklasifikasi Data Nasabah Asuransi. *TECHSI : Jurnal Penelitian Teknik Informatika*, 3(2), 129-132

Prasetyo, Eko. 2012. *DATA MINING – Konsep dan Aplikasi menggunakan MATLAB.* Yogyakarta: Andi.

Han, J., & Kamber, M. (2006). Data Mining Concept and Tehniques. San Fransisco: Morgan Kauffman.

https://www.datascience.or.id/detail_artikel/52/supervised-and-unsupervised-learning