Fidelity : Jurnal Teknik Elektro
Vol. 2, No. 2, Mei 2020, page 34-38

p-ISSN: 2686-3650
e-ISSN: 2686-3642

Gambaran Umum Metode Klasifikasi Data Mining

Aryo De Wibowo Muhammad Sidik
Electrical Engineering
Nusa Putra University
Sukabumi, Indonesia
aryo.dewibowo@nusaputra.ac.id

Ilman Himawan Kusumah
Electrical Engineering
Nusa Putra University
Sukabumi, Indonesia
Ilman.himawan@nusaputra.ac.id

Anang Suryana
Electrical Engineering
Nusa Putra University
Sukabumi, Indonesia
anang.suryana@nusaputra.ac.id

Edwinanto
Electrical Engineering
Nusa Putra University
Sukabumi, Indonesia
edwinanto@nusaputra.ac.id

Marina Artiyasa
Electrical Engineering
Nusa Putra University
Sukabumi, Indonesia
marina@nusaputra.ac.id

Anggy Pradiftha Junfithrana
Electrical Engineering
Nusa Putra University
Sukabumi, Indonesia
anggy.pradiftha@nusaputra.ac.id

***Abstrak*—Berbagai metode klasifikasi data mining diperiksa dalam penelitian ini untuk aplikasi database baru. Untuk menemukan suatu model, klasifikasi membagi data ke dalam kelompok-kelompok berdasarkan batasan yang telah ditentukan. Metode klasifikasi penting lainnya adalah algoritma Genetika C4.5, Naive Bayes, dan SVM. Akhirnya, kami membahas penjelasan algoritma.**

Kata Kunci—KNN, C4.5, SVM, CART

I. PENGANTAR

Penambangan data adalah proses menemukan pola dan tautan dalam kumpulan data besar yang sebelumnya tidak diketahui melalui teknik analisis data tingkat lanjut. Model statistik, algoritme matematika, dan metode pembelajaran mesin semuanya dapat disertakan dalam kotak alat ini. Oleh karena itu, data mining lebih dari sekedar mengumpulkan dan mengatur data. Analisis dan peramalan juga disertakan. Klasifikasi menjadi lebih populer karena dapat menangani rentang data yang lebih luas daripada regresi [1]. Banyak kegunaan untuk Machine Learning dapat ditemukan (ML). Namun, penambangan data adalah yang paling penting. Analisis dan, mungkin, upaya untuk menghubungkan beberapa fitur dapat menyebabkan kesalahan dalam interpretasi. Ketidakmampuan mereka untuk fokus pada masalah tertentu membuatnya menjadi tantangan bagi mereka.

Dalam banyak kasus, pembelajaran mesin dapat memecahkan masalah ini, membuat sistem dan desain mesin menjadi lebih efisien. Tugas yang dapat diatur sebagai diawasi ada di banyak aplikasi pembelajaran mesin setiap hari. Dalam artikel ini, kami berfokus pada teknik yang diperlukan. Tantangan klasifikasi di mana keluaran contoh hanya dapat terputus-putus dan tidak teratur menjadi fokus penelitian ini. Pada bagian berikut, kita membahas banyak pendekatan untuk mengkategorikan data. Di Bagian III, kita belajar bagaimana menilai kinerja pengklasifikasi. Bagian terakhir menutup bagian ini.

II. METODE KLASIFIKASI

A. Algoritma Genetika

Strategi ini digunakan dalam GA untuk menemukan jawaban yang tidak diketahui [7]. Hanya beberapa kategori data yang digunakan di GA. Implementasi GA membuat aturan prediksi tingkat tinggi untuk pemilihan atribut yang lebih baik. [10] Metode Michigan memberikan satu arah untuk setiap individu dalam suatu populasi, yang menurunkan biaya. Pendekatan Pittsburgh [5] mewakili seperangkat kriteria prediksi untuk populasi setiap individu. Membandingkan set praktik daripada aturan individu adalah pendekatan umum dalam klasifikasi. Aturan umum atau khusus dapat diimplementasikan menggunakan logika OR dan logika AND operator.

B. Aturan Set

Aturan "jika-maka-" mengatur klasifikasi. Misalnya, pemerintah memiliki reputasi (kondisi) yang buruk. Jika atribut dari instance X memenuhi persyaratan praktik, Aturan r akan mewakili instance sebagai X. Ada hierarki aturan. Mengenai hal ini, tatanan berbasis aturan telah diciptakan. Pengurutan berbasis kelas mengacu pada pengaturan kontrol berdasarkan kesamaannya. Contoh sebanyak mungkin harus dicakup oleh aturan yang baik untuk memastikan keakuratannya.

$$Ketepatan = \frac{contoh\ yang\ diklasifikasikan\ dengan\ benar}{contoh\ yang\ sesuai\ dengan\ aturan} \quad (1)$$

$$Liputan = \frac{contoh\ yang\ sesuai\ dengan\ aturan}{contoh\ di\ set\ kereta} \quad (2)$$

Pisahkan dan taklukkan (Furnkranz 99), dan semua peraturan dipelajari secara berurutan dalam dua set pengetahuan ini. Semua instans yang telah dicakup dihapus dari set pelatihan di Terpisah-Dan-Menaklukkan setelah arah baru terdeteksi. Ketika tidak ada lagi contoh yang harus dilewati, prosesnya berakhir. Keluarga metode induksi aturan AQ (Michal ski 69), CN 2 (Clask a Boswell 91), dan RIPPER K (Clask a Boswell 91) adalah pendekatan terpisah dan menaklukkan (Cohen 95). Alih-alih menggunakan kelas default, metode Different-And-Conquer mengambil peraturan yang lebih baik daripada kelas default dan menggunakan set validasi sampel yang tidak diketahui. Predikat kosong di kumpulan awal aturan CN2 mencakup semua ukuran. Tidak ada algoritma aturan CN2, tetapi ada predikat yang dihasilkan. Predikat tersebut diberikan kepada kelas penguasa dengan cara menentukan kelas mayoritas dari predikat tersebut.

C. C4.5

Nilai yang hilang dan noise dalam data diatasi dengan pengukuran properti numerik C4.5 [3]. Overfitting data dihindari dengan menggunakan pruning di C4.5 [9]. Untuk

penerus komersial, WEKA telah mengimplementasikan C4.8 sebagai J4.8 dan C5.0 (Pencarian aturan) sebagai versi pembaruan C4.5. Rata-rata tertimbang dari perkiraan kesalahan untuk setiap daun subpohon digunakan untuk menghitung perkiraan kesalahan untuk seluruh pohon. Misalnya, jika c adalah 25 persen, z sama dengan 0,69 sebagai estimasi kesalahan untuk sebuah simpul (dari distribusi normal). F adalah kesalahan data pelatihan. Ada berapa kali kemunculan tutupan daun?

D. CART

Menggunakan CART untuk Mengurutkan Data dan Analisis Regresi Ketika data tidak ada atau tidak ada, digunakan akurasi berdasarkan pohon. Metode CHAID dapat menangani nilai yang hilang karena CART memilih sampel secara acak. Persiapan data tidak diperlukan dalam CART. Ia melakukan semua pekerjaan untuk Anda. Nilai yang hilang dianggap sebagai nilai kategorikal yang unik dalam metode CHAID [26, 27]. Juga, C4.5 mengadopsi strategi ini. Pengganti metode lanjutan diperlakukan sebagai bidang utama yang hilang oleh CART. Sebagai pengganti pembagi langsung, pengganti akan digunakan. Menggunakan pemangkasan CART, setiap node dihapus dalam urutan yang tepat saat dibuat. Satu aturan Kesalahan Standar sudah cukup untuk kumpulan data kecil, dan menghasilkan pohon yang optimal. Aturan Kesalahan Standar nol menyediakan pohon nyata untuk kumpulan data substansial. Tidak diragukan lagi bahwa C4.5 dan CART adalah program yang kuat. Fungsi kerugian seperti indeks Gini digunakan ketika klasifikasi Pohon Keputusan salah.

E. Induksi Pohon Keputusan

Para peneliti telah mencari heuristik yang efisien untuk menghasilkan Pohon Keputusan biner yang mendekati optimal karena masalah membangun Pohon Keputusan biner yang ideal adalah NP-COMPLETE. Algoritme Hunt menghasilkan Pohon Keputusan menggunakan teknik top-down atau divide-and-conquer. Ada beberapa kelas data dalam sampel/baris ini. Untuk mengurangi ukuran kumpulan data, jalankan pengujian atribut. Dengan rakus, algoritma Hunt menjaga pemisahan optimal untuk setiap tahap berdasarkan beberapa nilai ambang [2]. Algoritma Hunt menggunakan pendekatan murah untuk pengujian atribut untuk menentukan divisi "terbaik" dan kapan harus berhenti membelah pada kondisi overfit atau underfit. Menggunakan prosedur Hunt, Anda dapat mengevaluasi nilai numerik dan kumpulan titik data. Kesalahan klasifikasi, indeks Gini, dan Entropi harus dipertimbangkan saat membagi. "Information Gain" adalah istilah untuk pengurangan Entropi. Berhenti ketika indeks Gini atau kesalahan klasifikasi Entropi meningkat pada kumpulan data pengujian. Indeks Gini sebuah simpul, diberikan t, adalah

$$GINI(t) = 1 - \sum \left[p\left(\frac{j}{t}\right) \right]^2 \quad (3)$$

Pada simpul t, frekuensi relatif kelas j direpresentasikan dengan [p(j/t)]. CART, SLIQ, dan SPRINT semuanya menggunakan indeks Gini. Kualitas split dihitung sebagai berikut ketika node p dibagi menjadi k divisi (anak):

$$GINI_{split} = \sum_{i=1}^{n} GINI(i) \quad (4)$$

Dimana ni = jumlah record pada anak i, n= jumlah record pada node p.

Membangun pohon keputusan di mana atribut ke cabang bergantung pada pemilihan kualitas untuk membagi data. Tujuannya adalah untuk menghilangkan noise sebanyak mungkin dari data. Semua instance dari kumpulan data harus termasuk dalam kelas yang sama agar subset dianggap murni. Dalam c4.5, heuristiknya adalah memilih properti dengan rasio perolehan atau perolehan informasi yang paling signifikan.

1. Perolehan informasi

Diberikan satu set contoh D, pertama-tama kita menghitung entropi-nya.

$$Entropy(D) = -\sum_{j=1}^{|c|} p\left(c_j\right) log_2 p\left(c_j\right) \quad (5)$$

Di mana $p(c_j)$ adalah probabilitas dari kumpulan data kelas $c_j$in D. kami menggunakan entropi sebagai ukuran ketidakmurnian atau ketidakteraturan kumpulan data D.(atau langkah informasi dalam pohon)[11]. Ketika data menjadi lebih murni dan lebih murni, nilai entropi menjadi lebih kecil dan lebih kecil. Informasi diperoleh dengan memilih atribut Ai[4]. Untuk bercabang ke partisi, datanya adalah

$$Gain(D, A) = Entropy(D) - EntropyA_i(D) \quad (6)$$

Kami mencabangkan/membagi pohon saat ini berdasarkan properti yang memberikan nilai paling banyak kepada pengguna. Menggunakan proses Divide-and-Conquer rekursif, Induksi Pohon Keputusan membangun Pohon Keputusan dari atas ke bawah. Meningkatkan ketidakpastian dapat dicapai dengan salah satu dari dua cara. Pertama, dilakukan pemangkasan terlebih dahulu jika cabang tumbuh terlalu cepat (Aturan penghentian dini). Uji Chi-Squared digunakan sebelum pra-pemangkasan. Setelah pemangkasan, Anda akan memiliki pohon keputusan yang matang dan pohon matahari yang dibuang yang tidak terlalu dapat dipercaya.

2. Model Overfitting

Model klasifikasi melakukan dua jenis kesalahan. Kesalahan disebabkan dalam pelatihan dan generalisasi karena sampel pelatihan mengandung banyak noise. Jumlah kesalahan klasifikasi yang dihasilkan oleh data pelatihan dikenal sebagai kesalahan pelatihan atau kesalahan yang tampak. Kesalahan generalisasi model klasifikasi, di sisi lain, adalah kesalahan yang diprediksi pada bahan yang belum dipelajari sebelumnya. Bahkan ketika tingkat kesalahan pelatihan menurun, tingkat kesalahan pengujian meningkat ketika ukuran pohon terlalu besar. Model overfitting adalah nama yang diberikan untuk praktik ini. Kesalahan pengujian signifikan karena sampel pelatihan berisi data yang bising meskipun kesalahan pelatihan adalah nol untuk pohon berduri.

3. Model yang Kurang Pas

Baik tingkat kesalahan pelatihan dan pengujian model klasifikasi adalah signifikan. Model underfitting adalah istilah untuk masalah ini. Properti biner baru seperti konjungsi, negasi, dan disjungsi digunakan untuk membentuk Pohon Keputusan (Zheng 1998). Ketika kriteria salah, Zheng (2000) mengembangkan setidaknya fitur

M-of-N[4]. Dengan membuat Pohon Keputusan multivariat, Gama dan brazil (99) menggabungkan diskriminasi linier dengan Pohon Keputusan. Untuk menghindari duplikasi data di Pohon Keputusan, terapkan algoritma bangunan FICUS, yang menerima masukan standar atau representasi fitur dan menghasilkan kumpulan fitur yang dibuat oleh Markovitch dan Rosenstein untuk menghindari replikasi data di Pohon Keputusan (2002). Tidak ada kombinasi yang lebih baik dari tingkat kesalahan dan kecepatan daripada C4.5 dalam penelitian ini. EC4.5 adalah versi algoritma yang lebih efisien berdasarkan evolusi analitik ini.

J. Jaringan Bayesian

DAG (Directed Acyclic Graph) dan karakteristik satu-ke-satu [12] membentuk dasar dari jaringan Bayesian. Mempelajari DAG dan struktur jaringan adalah dua pekerjaan yang berbeda untuk jaringan Bayesian [8]. mempelajari parameter dalam tabel probabilitas bersyarat dari struktur jaringan adalah tugas tetap (CPT). “Kebugaran” suatu jaringan dievaluasi mengenai data pelatihan, dan kemudian dilakukan pencarian untuk menemukan jaringan yang optimal berdasarkan skor ini [16] jika strukturnya tidak diketahui. Mempertimbangkan fakta bahwa jaringan Bayesian mempertimbangkan pengetahuan sebelumnya tentang topik yang ada. Oleh karena itu, jaringan Bayesian tidak dapat digunakan untuk menganalisis dataset sebesar itu [6].

G. Pembelajaran Berbasis Instan

Algoritme lambat, pembelajaran berbasis instans menunggu untuk menyimpulkan atau menggeneralisasi hingga klasifikasi selesai. Dibandingkan dengan algoritma pembelajaran yang bersemangat (seperti pohon Keputusan, jaringan saraf & Bayesian), teknik pembelajaran terbaru membutuhkan waktu komputasi yang lebih sedikit selama fase pelatihan tetapi lebih banyak waktu perhitungan selama langkah klasifikasi. Mereka memperoleh metode ICF dan algoritma RT3 di KNN, Brighton & Mellish 2002). (Wilson & Martinez 2003). Klasifikasi membutuhkan waktu lebih lama untuk dihitung dengan Pembelajaran Berbasis Instans karena hal ini. Akurasi klasifikasi dan waktu pemrosesan dapat ditingkatkan dengan memilih fitur input dari yang sudah tersedia (Yu & Liu 2004). Dimungkinkan untuk meningkatkan akurasi pengklasifikasi berbasis instance dengan memilih metrik jarak yang sesuai.

H. Mendukung Mesin Vektor

Untuk data linier dan non-linier, Support Vector Machine adalah pendekatan klasifikasi baru. Data pelatihan asli diubah menjadi dimensi yang lebih tinggi melalui pemetaan non-linier. Sekarang setelah memiliki lebih banyak ruang, ia dapat mencari secara linear untuk batas keputusan terbaik (atau, seperti yang dikenal, "hyperplane keputusan"). Sebuah hyperplane selalu dapat digunakan untuk membagi data dari dua kelas dengan pemetaan non-linier yang sangat baik ke dimensi yang cukup tinggi. Dengan dukungan vektor dan margin, SVM dapat menemukan hyperplane ini [25][26]. Fitur: Mereka dapat mensimulasikan batas keputusan non-linier yang kompleks (margin – memaksimalkan); dengan demikian, pelatihan mungkin dilakukan, tetapi akurasinya bagus. Klasifikasi dan prediksi keduanya dimungkinkan menggunakan SVM.

I. K-Nearest-Neighbour (KNN)

KNN adalah algoritma klasifikasi non-parametrik dasar tetapi efektif [19]. KNN, di sisi lain, memiliki kelemahan yang signifikan. Dalam banyak aplikasi, seperti penambangan web dinamis untuk repositori yang luas, efisiensinya yang rendah mencegahnya digunakan karena merupakan metode pembelajaran yang malas. Misalnya, mengindeks instance pelatihan sebagai pengklasifikasi KNN mengharuskan penyimpanan seluruh set pelatihan saat ini tidak pada redundansi set pelatihan untuk mengurangi kesulitan ini [29, 31, 24, dan 21] dapat sangat meminimalkan komputasi yang diperlukan pada waktu kueri . Versi ringkas dari Nearest Neighbor [Nearest Neighbor] digunakan untuk mengklasifikasikan data input dengan menyimpan hanya sebagian dari data pelatihan. Praktik di set pelatihan mungkin sangat mirip, dan beberapa mungkin dihapus karena tidak memberikan pengetahuan baru. Menggunakan kriteria Reduced Nearest Neighbor (KNN) yang disarankan oleh Gerbang [6], subset yang disimpan dapat diringkas lebih lanjut setelah CNN. Unsur-unsur tersebut dihilangkan dari subkelompok. Oleh karena itu tidak akan ada kesalahan.

III. Mengevaluasi Kinerja Classifier

A. Metode Hold-Out

Data asli dengan contoh berlabel dibagi menjadi dua set, yang dikenal sebagai set pelatihan dan tes [34]. Baik koleksi maupun perangkat pengujian tidak boleh digunakan untuk tujuan pengujian. Akurasi dapat diperkirakan dengan menggunakan test set yang belum pernah dilihat. Teknik ini terutama digunakan ketika berhadapan dengan kumpulan data besar.

B. n-fold Cross-validation

Ada n subkelompok terpisah yang berukuran sama dalam data yang disediakan. Untuk melatih pengklasifikasi, gunakan setiap subset sebagai set pelatihan. Untuk mendapatkan rata-rata n akurasi, metode ini dilakukan sebanyak n kali. Validasi silang dengan kelipatan 10 atau 5 cukup lazim. Ketika data yang diberikan tidak signifikan, strategi ini digunakan.

C. Validasi silang tinggalkan-satu-keluar-

Ketika ada sejumlah kecil data, pendekatan ini diterapkan. Ini adalah contoh validasi silang dalam tindakan. Semua data dari tes digunakan dalam proses pelatihan untuk setiap kali validasi silang [17]. Jika data asli memiliki lebih dari m contoh, validasi silang dilakukan m kali.

D. Set validasi

Ketika ada sejumlah kecil data, pendekatan ini diterapkan. Ini adalah contoh validasi silang dalam tindakan. Semua data dari tes digunakan dalam proses pelatihan untuk setiap kali validasi silang [17]. Jika data asli memiliki lebih dari m contoh, validasi silang dilakukan m kali.

E. Minimum Description Length (MDL)

Nilai yang hilang diperlakukan secara acak oleh MDL. Berkat pendekatan ini, data numerik yang jarang diganti dengan nol, dan data kategorikal diganti dengan vektor nol. Data dalam kolom bersarang yang tidak memiliki nilai dianggap tidak berarti apa-apa. Secara acak, kolom dengan tipe data sederhana diinterpretasikan sebagai tidak ada. [20] MDL memperhitungkan ukuran model dan pengurangan

ketidakpastian yang berasal dari pemanfaatannya. Entropi dan ukuran model keduanya dinyatakan dalam bit.

Model prediktif sederhana dari kelas target adalah semua yang dipertimbangkan MDL untuk setiap karakteristik dalam hal penggunaannya dalam algoritme. Model prediktor tunggal dibandingkan dan diberi peringkat sebagai bagian dari proses pemilihan model. Dimungkinkan untuk menggunakan persiapan data otomatis untuk melakukan MDL [32]. Menggunakan pohon keputusan, binning yang diawasi menciptakan batas bin terbaik. Ini memiliki karakteristik kuantitatif dan kualitatif.

F. Bagging

Pembelajaran ensemble pada awalnya dipraktekkan oleh Breiman (1996), yang menamakannya "bagging" (dari "agregasi bootstrap"), dan itu adalah salah satu metode paling dasar dari arching [1]. Awalnya dikembangkan untuk klasifikasi, meta-algoritma adalah contoh model rata-rata yang biasa digunakan dalam model pohon keputusan, tetapi dapat digunakan dalam klasifikasi atau model regresi apa pun. Beberapa set pelatihan dibuat menggunakan bootstrap, yaitu pengambilan sampel dengan penggantian. Masing-masing bagian data ini digunakan untuk mengembangkan model yang berbeda melalui pembelajaran mesin. Output regresi dan klasifikasi digabungkan dengan rata-rata atau voting untuk memberikan hasil tunggal. Bagging hanya berlaku untuk model nonlinier yang tidak stabil (artinya, bahkan sedikit perubahan pada set pelatihan dapat berdampak signifikan pada model). Gunakan set pelatihan bootstrap untuk membangun pengklasifikasi (digambar dengan penggantian). Distribusi probabilitas yang seragam mengatur pengambilan sampel dengan bantuan. Tidak ada perbedaan ukuran sampel bootstrap D dibandingkan dengan data aslinya. Set pelatihan mungkin berisi berbagai contoh, beberapa di antaranya mungkin muncul beberapa kali. Dengan menurunkan varians dari pengklasifikasi dasar, bagging meningkatkan kinerja generalisasi. Pengklasifikasi dasar mempengaruhi kinerja bagging. Bagging membantu mengurangi kesalahan yang disebabkan oleh osilasi acak dalam data pelatihan jika pengklasifikasi dasar tidak stabil. Bagging mungkin tidak dapat meningkatkan pengklasifikasi dasar yang solid. Alih-alih memperbaikinya, itu bisa memperburuk keadaan.

G. Boosting

Untuk meningkatkan, pengklasifikasi harus diproduksi secara berurutan. Penting untuk diingat bahwa setiap pengklasifikasi bergantung pada yang sebelumnya dan berkonsentrasi pada kesalahan yang sebelumnya. Contoh yang diprediksi dengan buruk di pengklasifikasi sebelumnya dipilih lebih sering dan diberi bobot yang lebih besar. Hanya catatan yang dikategorikan yang akan melihat peningkatan bobot penilaiannya. Pentingnya dokumen yang diklasifikasikan dengan tepat akan berkurang.

H. Ada-Boosting

Ada-meningkatkan tindakan Overfitting adalah masalah umum dengan hipotesis yang kompleks. Konsep sederhana mungkin tidak dijelaskan dengan jelas. Sejumlah asumsi sederhana diintegrasikan ke dalam satu asumsi bermasalah untuk menyederhanakan berbagai hal. Ada beberapa pilihan. Pertama, lebih banyak contoh kasus yang salah diklasifikasikan dipilih dalam pengklasifikasi sebelumnya, sehingga memilih standar berdasarkan kesalahan pemilihan ini lebih umum. Beberapa algoritme tidak bekerja dengan baik dengan kasus yang salah diklasifikasikan dengan bobot yang lebih besar (semua instance bersifat korporat, tetapi konsekuensinya berbeda).

I. Occam’s Razor

Untuk menggunakan Occam's Razor, Anda harus memilih penjelasan paling sederhana yang paling sesuai dengan bukti. Akibatnya, pisau cukur Occam memiliki peluang terbaik untuk mengidentifikasi item yang tidak diketahui dengan benar. Mengingat dua model dengan kesalahan generalisasi yang sama, pisau cukur Occam mengatakan bahwa model yang lebih sederhana lebih disukai daripada yang lebih kompleks [33]. Kesalahan dalam data dapat menyebabkan model canggih yang tidak dipasang dengan benar.

H. Random Forest

WEKA adalah estimator dan pembangun model klasifikasi dan regresi untuk tujuan umum. Mesin vektor peningkat gradien dan dukungan sangat diuntungkan dari akurasi tinggi hutan acak. Random Forest dibangun dari dua jenis data. 1. Pohon klasifikasi dan regresi. 2. Ini adalah sampel dari dataset asli yang telah diambil kembali dari dataset asli dengan menggunakan replacement sampling.

IV. Kesimpulan

Strategi klasifikasi data mining dieksplorasi dalam pekerjaan ini. Seperti yang dijelaskan oleh penelitian ini, masing-masing metode memiliki kelebihan dan kekurangan. Dalam penambangan data, banyak teknik dari berbagai bidang, termasuk pembelajaran mesin, kecerdasan buatan, analisis statistik, dan pengenalan pola, digunakan untuk menganalisis sejumlah besar data. Untuk menjalankan berbagai tugas analisis data, sayangnya beberapa metode data mining telah mengakar di sektor-sektor ini.

Referensi

[1] Jiawei Han and MichelineKamberData Mining: Concepts and Techniques,2ndedition.

[2] Baik, S. Bala, J. (2004), A Decision Tree Algorithm For Distributed Data Mining.

[3] Witten, I. & Frank, E. (2005), "Data Mining: Practical Machine Learning Tools And Techniques", 2nd Edition, Morgan Francisco, 2005.

[4] Zheng, Z. (2000). Constructing X-Of-N Attributes For Decision Tree Learning. Machine Learning 40: 35–75.

[5] Sirgo, J., Lopez, A., Janez, R., Blanco, R., Abajo, N., Tarrio, M., Perez, R., “A Data Mining Engine Based On Internet, Emerging Technologies And Factory Automation”.

[6] Friedman, N., Geiger, D. &Goldszmidt M. (1997). Bayesian Network Classifiers.Machine Learning 29: 131-163.

[7] Fayyad, U., Piatetsky-Shapiro, G., And Smyth P., “From Data Mining To Nowledge Discovery In Databases,” Ai Magazine, American Association For Artificial Intelligence, 1996.

[8] Friedman, N. &Koller, D. (2003). Being Bayesian About Network Structure: A Bayesian Approach To Structure Discovery In Bayesian Networks. Machine Learning 50(1): 95-125.

[9] Quinlan, J.R., C4.5 -- Programs For Machine Learning.Morgan Kaufmann Publishers, San Francisco, Ca, 1993.

[10] Bianca V. D.,PhilippeBoula De Mareüil And Martine Adda-Decker, “Identification Of Foreign-Accented French Using Data Mining Techniques, Computer Sciences Laboratory For Mechanics And Engineering Sciences (Limsi)”.

[11] Breslow, L. A. & Aha, D. W. (1997). Simplifying Decision Trees:A Survey. Knowledge Engineering Review 12: 1–40.

[12] Jensen, F. (1996). An Introduction To Bayesian Networks. Springer.

[13] Introduction To Data Mining By Tan,Steinbach,Kumar.

[14] Collins, M., Schapire, R.E. And Singer, Y. (2000). Logistic Regression, AdaboostAndBregman Distances. Proc. Thirteenth Annual Conference Computational Learning Theory.

[15] T. Mitchell.: Machine Learning. MitpressAndMcgraw-Hill (1997).

[16] Madden, M. (2003), The Performance Of Bayesian Network Classifiers Constructed Using Different Techniques, Proceedings Of European Conference On Machine Learning, Workshopon Probabilistic Graphical Models ForClassification, Pp. 59-70.

[17] Avirm Michael Kearns And Dana Ron,"Algorithmic Stability And Sanity-Check Bounds For Leave-One-Out Cross Validation".

[18] Freund, Y. (1995). Boosting A Weak Learning Algorithm By Majority. Information And Computation 121, 256{285}.

[19] D. Hand, H. Mannila, P. Smyth.: Principles Of Data Mining. The MIT Press. (2001).

[20] Peter D. Grunwald "The Minimum Description Length Principle.

[21] M. Kubat, M. Jr.: Voting Nearest-NeighbourSubclassifiers. Proceedings Of The 17th International Conference Onmachine Learning, ICML-2000, Pp.503-510, Stanford, CA, June 29-July 2, (2000).