BAB II

LANDASAN TEORI

2.1 *Data Mining*

2.1.1 Definisi *Data Mining*

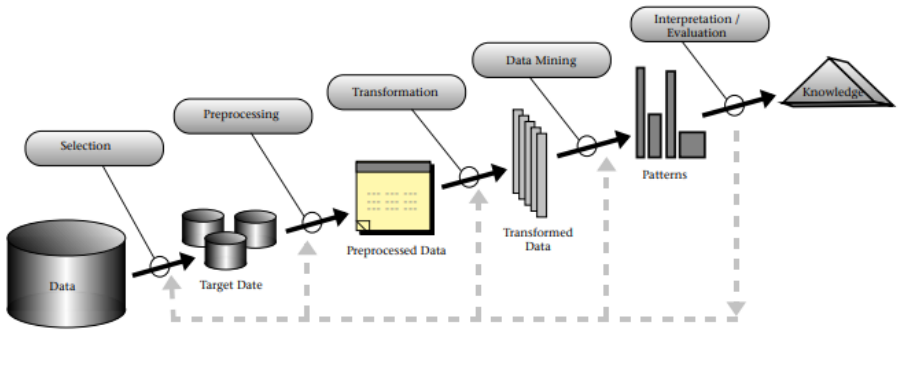
Data mining adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik-teknik, metode-metode, atau algoritma dalam *data mining* sangat bervariasi. Pemilihan metode atau algoritma yang tepat sangat bergantung pada tujuan dan proses *Knowledge Discovery in Database* (KDD) secara keseluruhan. Menurut Gartner Group, *data mining* adalah proses menemukan hubungan baru yang mempunyai arti, pola dan kebiasaan dengan memilah-milah sebagian besar data yang disimpan dalam media penyimpanan dengan menggunakan teknologi pengenalan pola seperti teknik statistik dan matematika. *Data mining* merupakan gabungan dari beberapa disiplin ilmu yang menyatukan teknik dari pembelajaran mesin, pengenalan pola, statistik, database, dan visualisasi untuk penanganan permasalahan pengambilan informasi dari database yang besar. Berdasarkan fungsinya, *data mining* dikelompokkan menjadi 6 yaitu deskripsi, estimasi, prediksi, klasifikasi, *clustering* dan asosiasi (Marcoulides, 2005).

Sementara definisi lain dari *Data Mining* adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai database besar. *Data mining*

merupakan serangkaian proses untuk menggali nilai tambah dari suatu kumpulan data berupa pengetahuan yang selama ini tidak diketahui secara manual (O'Regan, 2011).

2.1.2 Tahapan *Data Mining*

Tahapan yang dilakukan pada proses *data mining* diawali dari seleksi data dari data sumber ke data target, tahap *preprocessing* untuk memperbaiki kualitas data, transformasi, *data mining* serta tahap interpretasi dan evaluasi yang menghasilkan output berupa pengetahuan baru yang diharapkan memberikan konstribusi yang lebih baik (Fayyad, Piatetsky-Shapiro, & Smyth, 2015). Secara detail dijelaskan pada gambar 2.1.



2.1 Gambar 2.1 Tahapan Data Mining (Fayyad et al., 2015)

1. *Data Selection*

Pemilihan (seleksi) data dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalian informasi dalam KDD dimulai. Data hasil seleksi yang digunakan untuk proses *data mining*, disimpan dalam suatu berkas, terpisah dari basis data operasional.

2. *Pre-processing* / Cleaning

Sebelum proses *data mining* dapat dilaksanakan, perlu dilakukan proses *cleaning* pada data yang menjadi fokus KDD. Proses *cleaning* mencakup antara lain membuang duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data.

3. *Transformation*

Coding adalah proses transformasi pada data yang telah dipilih, sehingga data tersebut sesuai untuk proses *data mining*. Proses coding dalam KDD merupakan proses kreatif dan sangat tergantung pada jenis atau pola informasi yang akan dicari dalam basis data.

4. *Data Mining*

Data mining adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik, metode, atau algoritma dalam data mining sangat bervariasi. Pemilihan metode atau algoritma yang tepat sangat bergantung pada tujuan dan proses KDD secara keseluruhan.

5. *Interpretation / Evaluation*

Pola informasi yang dihasilkan dari proses *data mining* perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak yang berkepentingan. Tahap ini merupakan bagian dari proses KDD yang disebut *interpretation*. Tahap ini mencakup pemeriksaan apakah pola atau informasi yang ditemukan bertentangan dengan fakta atau hipotesis yang ada sebelumnya.

2.2 *Machine Learning*

Machine Learning adalah bidang ilmu komputer yang melibatkan studi dan konstruksi teknik yang memungkinkan komputer untuk belajar mandiri berdasarkan data input untuk memecahkan masalah spesifik (Hoang & Nguyen, 2018). Jenis-jenis permasalahan yang umumnya diselesaikan dengan pendekatan *Machine Learning* adalah klasterisasi dan klasifikasi. Klasterisasi adalah aktivitas yang bertujuan mengelompokkan data berdasarkan kedekatan fitur yang dimilikinya, sedangkan klasifikasi bertujuan untuk memisahkan data menjadi kelas-kelas tertentu. Perbedaan yang mendasar antara 2 buah permasalahan ini adalah pada proses klasterisasi. Data-data dikelompokkan tanpa pelabelan, sedangkan klasifikasi mengelompokkan data-data menjadi label tertentu (Utama, 2018).

2.3 Klasifikasi

Klasifikasi (taksonomi) adalah proses menempatkan objek tertentu (konsep) dalam satu set kategori, berdasarkan masing-masing objek (konsep) *property*. Proses klasifikasi didasarkan pada empat komponen mendasar yaitu kelas, prediktor, *training set*, dan pengujian dataset. Di antara model klasifikasi yang paling popular adalah *Decision / Classification Trees, Bayesian Classifiers / Naïve Bayes Classifiers, Neural Networks, Statistical Analysis, Genetic Algorithms, Rough Sets, K-Nearest Neighbor Classifier, Rule based Methods, Memory Based Reasoning, Support Vector*. Klasifikasi adalah proses untuk menemukan model atau fungsi yang menjelaskan atau membedakan konsep atau kelas data, dengan tujuan untuk dapat memperkirakan kelas dari suatu objek yang

labelnya tidak diketahui. Dalam mencapai tujuan tersebut, proses klasifikasi membentuk suatu model yang mampu membedakan data ke dalam kelas-kelas yang berbeda berdasarkan aturan atau fungsi tertentu. Model itu sendiri bisa berupa aturan "jika maka", berupa pohon keputusan, atau formula matedies (Fluorida Fibrianda & Bhawiyuga, 2018). Ada juga pendapat lain tentang klasifikasi yaitu merupakan suatu proses menemukan kumpulan pola atau fungsi yang mendeskripsikan serta memisahkan kelas data yang satu dengan yang lainnya untuk menyatakan objek tersebut masuk pada kategori tertentu yang sudah ditentukan atau dikenal juga sebagai *supervised learning*.

Metode klasifikasi terdiri dari dua proses, yaitu : *learning step* atau tahap *training phase* dimana algoritma klasifikasi membangun *classifier* dengan menganalisis atau "belajar dari" sebuah *training set* yang memiliki label dan telah tersedia sebelumnya, selanjutnya untuk mengetahui akurasi dari *classifier* yang telah dibentuk, pada tahap kedua dilakukan pengujian terhadap *classifier* tersebut dengan menggunakan *test set* yang merupakan kumpulan data baru yang dipilih secara acak dan bersifat independen dari *training set* yang artinya data yang digunakan pada *test set* tidak digunakan untuk membangun *classifier (*Zhao, 2014)

2.4 *Naive Bayes*

Teorema bayes adalah perhitungan statistik dengan menghitung probabilitas kemiripan kasus lama yang ada dibasis kasus dengan kasus baru. *Teorema bayes* memiliki tingkat akurasi yang tinggi dan kecepatan yang baik ketika diterapkan pada *database* yang besar. *Naive bayes* merupakan perhitungan

teorema bayes yang paling sederhana, karena mampu mengurangi kompleksitas komputasi menjadi multiplikasi sederhana dari probabilitas. Selain itu, algoritma *naive bayes* juga mampu menangani set data yang memiliki banyak atribut.

Keuntungan menggunakan *naive bayes* adalah metode ini hanya membutuhkan dua data yaitu data *training* (*training set*) dan data *testing* (*testing* set) untuk menguji suatu data yang ingin diperoleh (Jadhav & Channe, 2016). Persamaan dari *naive bayes* sebagai berikut :

P (C | X), dari P (C), P (X), dan P (X | C)

$$P(C \mid X) = \frac{P(X \mid C).P(C)}{P(X)}$$

Gambar 2.2 Persamaan Klasifikasi *Naive Bayes* (Jadhav & Channe, 2016)

Dimana :

P (C | X) adalah probabilitas posterior dari kelas target.

P (C) disebut probabilitas sebelumnya dari kelas.

P (X | C) adalah kemungkinan yang merupakan probabilitas prediktor kelas yang diberikan.

P (X) adalah probabilitas sebelumnya dari prediktor kelas.

2.5 *Support Vector Machine (SVM)*

Support Vector Machine (SVM) merupakan salah satu metode klasifikasi. Terdapat dua kategori SVM yaitu *Support Vector Machine Classification* dan *Support Vector Machine Regression*. SVM di perkenalkan pertama kali oleh Vapnik pada tahun 1992 sebagai konsep unggulan dalam bidang *pattern recognition*, algoritma ini dapat memilih model otomatis dan tidak memiliki

masalah *overfittin*. Metode SVM sangat baik untuk prediksi karena metode ini dapat meminimalkan kesalahan klasifikasi dan penyimpangan data pada data training. Cara kerja metode SVM yaitu dengan memisahkan beberapa kelompok data dengan garis. Garis ini dikenal dengan *hyperplane*, dengan teknik SVM bertujuan untuk mencari *hyperplane* yang optimal. Kernel merupakan fungsi yang digunakan untuk mendapatkan *hyperplane* yang optimum (Fadilah, Agfiannisa, & Azhar, 2020).

2.6 *Confusion Matrix*

Confusion matrix dapat diartikan sebagai suatu alat yang memiliki fungsi untuk melakukan analisis apakah *classifier* tersebut baik dalam mengenali tuple dari kelas yang berbeda. Nilai dari *True-Positive* dan *True-Negative* memberikan informasi ketika *classifier* dalam melakukan klasifikasi data bernilai benar, sedangkan *False-Positive* dan *False-Negative* memberikan informasi ketika *classifier* salah dalam melakukan klasifikasi data. TP (*True Positive*) merupakan jumlah data dengan nilai sebenarnya positif dan nilai prediksi positif, FP (*False Positive*) merupakan jumlah data dengan nilai sebenarnya negatif dan nilai prediksi positif, FN (*False Negative*) merupakan jumlah data dengan nilai sebenarnya positif dan nilai prediksi negatif, TN (*True Negative*) merupakan jumlah data dengan nilai sebenarnya negatif dan nilai prediksi negatif (Fibrianda & Bhawiyuga, 2018).

2.7 *Parameter Matric*

Accuracy merupakan tingkat keterhubungan antara suatu nilai yang diprediksi dengan nilai aktual yang ada (Devita et al., 2018).

$$Accuracy = \frac{True\ Positive+True\ Negative}{True\ Positive+False\ Negative+False\ Positive+True\ Negative}$$ (Rumus 1)

Precision merupakan pengukuran tingkat ketepatan antara informasi yang diminta oleh pemohon dengan jawaban yang diberikan oleh sistem.

$$Precision = \frac{True\ Positive}{True\ Positive+False\ Positive}$$ (Rumus 2)

Recall merupakan tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi dalam suatu pemrosesan data.

$$Recall = \frac{True\ Positive}{True\ Positive+False\ Negative}$$ (Rumus 3)

2.8 WEKA (*Waikato Environment for Knowledge Analysis*)

WEKA merupakan sebuah *Tools* yang praktis. Kepanjangan dari WEKA adalah *Waikato Environment for Knowledge Analysis*, dibuat di Universitas Waikato, *New Zaeland.* WEKA juga dapat digunakan pada beberapa tingkatan berbeda serta mengandung *Tools* untuk *pre-processing* data yaitu : *classification, regression, clustering, association rules, dan visualization* (Pujiono, Amborowati, Suyanto, & Kunci, 2013). Kelas yang paling penting disini adalah *classifier*, yang mendeklarasikan struktur umum dari skema klasifikasi dan prediksi.

Kelas ini memiliki 2 metoda, yaitu *build classifier* dan *classify instance*, yang harus diimplementasikan oleh kelas-kelas yang menginduk kelas ini. WEKA sangat membantu dalam mengolah data (*Data Mining*), karena dengan software WEKA ini dapat dengan mudah untuk mengolah setumpuk data dan mengambil informasi yang penting saja dari sekumpulan data tersebut. Selain itu juga, WEKA mampu menyelesaikan masalah-masalah *data mining* didunia nyata khususnya klasifikasi yang mendasari pendekatan *machine learning*.

2.9 *State Of The Art (SOTA)*

State Of The Art merupakan pengumpulan data penelitian sebelumnya dengan menentukan sumber atau referensi yang tertera pada tabel 2.1

Tabel 2.1 Referensi Penelitian

No	Peneliti	Tahun Terbit	Judul	Metode/Algoritma	Kesimpulan
1	Devi Nurul Anisa dan Jumanto (2022)	Dinamika Informatika, Vol.14, No.1, Maret 2022 : 33-42, E-ISSN : 2714-8769 | P-ISSN : 2085-3343	Klasifikasi Penyakit Diabetes Menggunakan Algoritma *Naive Bayes*	*Naive Bayes*	Menerapkan metode klasifikasi untuk memprediksi apakah seseorang terkena diabetes atau tidak menggunakan algoritma *Naive Bayes.* Terdapat 390 data yang diklasifikasikan, dalam pengklasifikasiannya menggunakan 9 variabel yaitu *cholesterol, glucose, hdl cholesterol, age, gender, weight, systolic bp, distolic bp, diabetes*. Menghasilkan akurasi 92,3% untuk *Data Train* dan 91,6% untuk *Data Test*.
2	Hilda Apriyani dan Kurniati (2020)	Journal of Information Technology Ampera Vol. 1, No. 3, December 2020 e-ISSN: 2774-2121	Perbandingan Metode *Naïve Bayes* Dan *Support Vector Machine* Dalam Klasifikasi Penyakit Diabetes Melitus	*Naive Bayes* dan *Support Vector Machine (SVM)*	Pengujian yang dilakukan menggunakan data training dan data testing kemudian diukur untuk mengetahui tingkat akurasi dengan evaluasi *confusion matrix*. Untuk algoritma SVM menggunakan dua kernel yaitu kernel polynomial dan RBF. Kernel Polynomial memiliki nilai akurasi yang tertinggi dengan nilai 96.2704% bisa dikatakan lebih akurat bila dibandingkan dengan algoritma *naïve bayes* dengan tingkat akurasi 92.0746%.

Tabel 2.2 Referensi Penelitian (lanjutan)

3	Mercury Fluorida Fibrianda dan Adhitya Bhawiyuga (2018)	Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, Vol. 2, No. 9, September 2018, hlm. 3112-3123, e-ISSN: 2548-964X	Analisis Perbandingan Akurasi Deteksi Serangan Pada Jaringan Komputer Dengan Metode *Naïve Bayes* Dan *Support Vector Machine* (SVM)	*Naive Bayes*, SVM *Linear*, SVM *Polynomial*, dan SVM *Sigmoid.*	Fitur yang digunakan dalam proses klasifikasi yaitu *totalSourceBytes, totalDestinationBytes, totalDestinationPacket, totalSourcePacket, direction, Source TCPFlagsDescription, Destination TCPFlagsDescription, protocolName, sourcePort, Destination, destinationPort, startDateTime,* dan *stopDateTime.* Performa yang dihasilkan dari *confusion matrix* pada masing-masing *classifier Naive Bayes*, SVM *Linear*, SVM *Polynomial*, dan SVM *Sigmoid* menghasilkan persentase akurasi berturut-turut sebesar 85,055%, 99,995%, 99,999% dan 99,995%. Performa kinerja klasifikasi yang dihasilkan dari kurva ROC pada *classifier Naive Bayes* yaitu baik, SVM *Linear* lemah, SVM *Polynomial* sangat lemah, dan SVM *Sigmoid* lemah. Sedangkan jika dilihat dari kurva ROC dengan *cross-validation* menunjukkan bahwa *classifier Naive Bayes* yaitu lemah dengan nilai AUC 0,5, SVM *Linear* baik dengan nilai AUC 0,75, SVM *Polynomial* sangat lemah dengan nilai AUC 0,33 dan SVM *Sigmoid* lemah dengan nilai AUC 0,5.

Tabel 2.3 Referensi Penelitian (lanjutan)

4	Hairani, Khurniawan Eko Saputro dan Sofiansyah Fadli (2020)	Jurnal Teknologi dan Sistem Komputer, 8(2), 2020, 89-93, *DOI:10.14710/jtsiskom.8.2.2020.89-93*	*K-Means-SMOTE* untuk menangani ketidakseimbangan kelas dalam klasifikasi penyakit diabetes dengan C4.5, SVM dan *Naive Bayes*	C4.5, SVM dan *Naive Bayes*	Penelitian ini dilakukan untuk menyelesaikan permasalahan ketidakseimbangan kelas pada dataset penyakit diabetes Pima Indian menggunakan *K-Means-SMOTE*. Dataset tersebut memiliki 268 data dari kelas positif (kelas *minoritas*) dan 500 data dari kelas negatif (kelas *mayoritas*). Kombinasi *K-Means-SMOTE* dengan metode klasifikasi SVM memiliki akurasi dan sensitivitas terbaik, yaitu sebesar 82% dan 77%. Sedangkan dengan metode *Naive Bayes* menghasilkan spesifisitas terbaik sebesar 89%.
5	Abu Wildan Mucholladin, Fitra Abdurrachman Bachtiar dan Muhammad Tanzil Furqon (2021)	Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, Vol. 5, No. 2, Februari 2021, hlm. 622-633, e-ISSN: 2548-964X	Klasifikasi Penyakit Diabetes menggunakan Metode *Support Vector Machine*	*Support Vector Machine (SVM)*	Model SVM dibagi menjadi dua jenis yaitu model *benchmark* yang diimplementasikan menggunakan algoritma *Sequential Minimal Optimization* (SMO) dan model *scratch* yang diimplementasikan menggunakan algoritma *Sequential Learning*. Model yang sudah optimal diuji kembali pada beberapa metrik menggunakan *10-fold cross validation*. Hasil pengujian menunjukkan bahwa model *benchmark* memiliki nilai *mean accuracy* sebesar 0,87, *mean precision* sebesar 0,82, *mean sensitivity* sebesar 0,78, dan *mean specificity* sebesar 0,92. Model *scratch* memiliki nilai *mean accuracy* sebesar 0,78, *mean precision* sebesar 0,69, *mean sensitivity* sebesar 0,59, dan *mean specificity* sebesar 0,87

Tabel 2.4 Referensi Penelitian (lanjutan)

6	Erwin, Laras Azrisa Nurjanah, Dea Sellan Noviyanti dan Yurika (2018)	Prosiding Annual Research Seminar 2018 - Computer Science and ICT, Vol.4 No.1, ISBN : 978-979-587-813-1	Klasifikasi Penyakit *Diabetik Retinopathy* dengan Metode *Naive Bayes* pada Citra Retina	*Naive Bayes*	Dari data uji sebanyak 15 citra retina didapatkan hasil klasifikasi 5 citra Normal, 6 citra NPDR, dan 3 citra PDR. Dengan metode *Naive Bayes* mendapatkan akurasi sebesar 93%. Metode *Naive Bayes* yang digunakan mendapatkan hasil yang baik dalam pengklasifikasian.
7	Sayali D. Jadhav dan H. P. Channe (2013)	International Journal of Science and Research (IJSR) ISSN (Online): 2319-7064 Index Copernicus Value (2013): 6.14 | Impact Factor (2014): 5.611	*Comparative Study of K-NN, Naive Bayes and Decision Tree Classification Techniques*	K-NN, *Naive Bayes* dan *Decision Tree*	Semua algoritma *Decision Tree* lebih akurat dan memiliki tingkat kesalahan yang lebih sedikit dan merupakan algoritma yang lebih mudah dibandingkan *dengan K-NN* dan *Bayesian*. Hasil dari implementasi WEKA pada dataset yang sama menunjukkan bahwa algoritma *Decision Tree* lebih unggul dan algoritma *Naive Bayes* memiliki tingkat akurasi yang hampir sama dengan algoritma *Decision Tree*.
8	Achmad Ridwan (2020)	Jurnal Sistem Komputer dan Kecerdasan Buatan Volume IV – Nomor 1 – September 2020	Penerapan Algoritma *Naïve Bayes* Untuk Klasifikasi Penyakit Diabetes Mellitus	*Naive Bayes*	Data yang digunakan pada analisis ini merupakan data dari dataset *UCI Machine Learning* yaitu *Early Stage Diabetes Risk* tahun 2020 dan terdapat 16 atribut yang mempengaruhi Klasifikasi dataset *Early Stage Diabetes Risk prediction* yaitu *Age, Gender, Polyuria, Polydipsia, Sudden Weight Loss, Weakness, Polyphagia, Genital Thrush, Visual Blurring, Itching, Irritability, Delayed Healing, Partial Paresis, Muscle Stiffness, Alopecia, Obesity, Class*. Analisis yang dilakukan meliputi data *preprocessing*, model, dan evaluasi Hasil klasifikasi menunjukkan akurasi sebesar 90.20% dan nilai AUC nya yaitu 0,95.

Tabel 2.5 Referensi Penelitian (lanjutan)

9	Fatmawati (2016)	Jurnal Techno Nusa Mandiri Vol. XIII, No. 1 Maret 2016 50	Perbandingan Algoritma Klasifikasi *Data Mining* Model C4.5 Dan *Naive Bayes* Untuk Prediksi Penyakit Diabetes	C4.5 dan *Naive Bayes*.	Hasil klasifikasi data di evaluasi dengan menggunakan *Confusion Matrix* dan kurva *ROC* untuk mengetahui tingkat hasil akurasi menggunakan algoritma *Decision Tree* yaitu sebesar 73.30% dan nilai AUC dari kurva *ROC* adalah 0.733 sedangkan algoritma *Naive Bayes* sebesar 75.13% nilai AUC dari kurva *ROC* 0.810 sehingga dapat disimpulkan bahwa dengan menggunakan model *Naive Bayes* lebih tinggi tingkat akurasinya, dengan peningkatan akurasi sebesar 1.83% dan peningkatan nilai AUC sebesar 0.077 sedangkan hasil pengujian dari prediksi diabetes hasilnya termasuk *Good Clasification.*
10	Dian Pramadhana (2021)	Jurnal Pendidikan Informatika Vol. 5 No. 1, Juni, 2021, Hal. 89-98, e-ISSN 2549-7472 DOI: 10.29408/edumatic.v5i1.3336	Klasifikasi Penyakit Diabetes Menggunakan Metode *CFS* dan *ROS* dengan Algoritma J48 Berbasis *Adaboost*	J48	Penggunaan seleksi fitur dengan metode *CFS (Correlation Feature Selection)* dan *Resample (Random Over Sampling)* yang didapatkan pada algoritma J48 menghasilkan nilai akurasi yang lebih baik yaitu sebesar 92,3%. Dengan penambahan metode CFS dan *Resample (Random Over Sampling)* serta adaboost dalam pengklasifikasian penyakit diabetes dapat meningkatkan kinerja Algoritma J48 sehingga membuat semakin lebih optimal dibandingkan tanpa menggunakan metode CFS dan *Resample (Random Over Sampling)* serta adaboost.

Tabel 2.6 Referensi Penelitian (lanjutan)

11	Hiya Nalatissifa, Windu Gata, Sri Diantika, Khoirun Nisa (2020)	Jurnal Informatika Universitas Pamulang, Vol. 5, No. 4, Desember 2020 (578-584), ISSN: 2541-1004, e-ISSN: 2622-4615	Perbandingan Kinerja Algoritma Klasifikasi *Naive Bayes, Support Vector Machine (SVM),* dan *Random Forest* untuk Prediksi Ketidakhadiran di Tempat Kerja	*Naive Bayes, Support Vector Machine (SVM),* dan *Random Forest*	Memprediksi ketidakhadiran di tempat kerja berdasarkan dataset *Absenteeism at work* menggunakan aplikasi Weka 3.8 dan algoritma Naïve *Bayes, Support Vector Machine (SVM),* dan *Random Forest.* Pada hasil penelitian, algoritma *Random Forest* memperoleh nilai akurasi, presisi, dan *recall* yang paling tinggi dibandingkan dengan algoritma *Naïve Bayes* dan *SVM,* yaitu menghasilkan nilai akurasi sebesar 99.38%, presisi 99.42% dan recall 99.39%.
12	Yusra, Dhita Olivita, Yelfi Vitriani (2016)	Jurnal Sains, Teknologi dan Industri, Vol. 14, No. 1, Desember 2016, pp. 79 - 85 ISSN 1693-2390 print/ISSN 2407-0939 online	Perbandingan Klasifikasi Tugas Akhir Mahasiswa Jurusan Teknik Informatika Menggunakan Metode Naïve Bayes Classifier dan K-Nearest Neighbor	*Naïve Bayes Classifier* dan *K-Nearest Neighbor*	Pengujian akurasi metode pada penelitian ini dilakukan dengan *test option* 10-*fold cross validation* dan evaluasi data uji menggunakan *confusion matrix*. Dari penelitian yang telah dilakukan, didapatkan hasil pada seratus data tugas akhir dengan jumlah kelas acak, metode *Naïve Bayes* menghasilkan nilai akurasi lebih baik, yaitu sebesar 87%. Pengujian pada metode *K-Nearest Neighbor* menghasilkan nilai akurasi 84% dengan nilai k=3, 85% dengan nilai k=5, 86% dengan nilai k=7 dan 84% dengan nilai k=9
13	Sri Diantika, Windu Gata, Hiya Nalatissifa, Mareanus Lase (2021)	Jurnal Ilmiah Elektronika dan Komputer, Vol.14, No.1, Juli 2021, pp. 10 – 15, p-ISSN : 1907-0012, e-ISSN : 2714-5417	Komparasi Algoritma SVM Dan Naive Bayes Untuk Klasifikasi Kestabilan Jaringan Listrik	*Support Vector Machine (SVM)* dan *Naive Bayes*	Dilakukan perbandingan penerapan algoritma klasifikasi SVM dan *Naïve Bayes* terhadap dataset *Electrical Grid Stability Simulated* yang diambil dari UCI Machine Learning. Dari hasil pengujian klasifikasi kestabilan jaringan listrik yang telah dilakukan menggunakan aplikasi WEKA 3.8.2. Metode *Support Vector Machine (SVM)* menunjukan tingkat *accuracy* yang lebih baik yaitu sebesar 98.9% jika

					dibandingkan dengan metode *Naive Bayes* yang menghasilkan nilai akurasi sebesar 97.64% Hasil akurasi ini akan menunjukan hasil yang berbeda tergantung dengan jenis data, jumlah instance, label class dan Percentage split data yang digunakan.

2.10 Matriks Penelitian

Matriks penelitian menjelaskan tentang perbedaan antara penelitian yang dilakukan dengan penelitian terkait. Terdapat beberapa indikator yang menunjukkan perbedaan dan persamaan antara penelitian terdahulu dengan penelitian yang dilakukan.

Berikut merupakan tabel matriks penelitian :

Tabel 2.2 Matriks Penelitian

No	Peneliti	Judul	Algoritma							Parameter Metric						
			Naive Bayes	SVM	C4.5	KNN	Decision Tree	J48	Random Forest	Accuracy	Precision	Recall	Error Rate	F1 Score	Spesifisitas	ROC Curve
1	Devi Nurul, dkk (2022)	Klasifikasi Penyakit Diabetes Menggunakan Algoritma *Naive Bayes*	✓							✓						
2	Hilda Apriyani, dkk (2020)	Perbandingan Metode *Naïve Bayes* Dan *Support Vector Machine* Dalam Klasifikasi Penyakit Diabetes Melitus	✓	✓						✓			✓			

Tabel 2.2 Matriks Penelitian (lanjutan)

3	Mercury Fluorida Fibrianda, dkk (2018)	Analisis Perbandingan Akurasi Deteksi Serangan Pada Jaringan Komputer Dengan Metode *Naïve Bayes* Dan *Support Vector Machine* (SVM)	✓	✓						✓	✓	✓		✓		
4	Hairani, dkk (2020)	*K-Means-SMOTE* untuk menangani ketidakseimbangan kelas dalam klasifikasi penyakit diabetes dengan C4.5, SVM dan *Naive Bayes*	✓	✓	✓					✓		✓			✓	
5	Abu Wildan Mucholladin, dkk (2021)	Klasifikasi Penyakit Diabetes menggunakan Metode *Support Vector Machine*		✓						✓	✓	✓			✓	
6	Erwin, dkk (2018)	Klasifikasi Penyakit *Diabetik Retinopathy* dengan Metode *Naive Bayes* pada Citra Retina	✓							✓						

Tabel 2.2 Matriks Penelitian (lanjutan)

7	Sayali D. Jadhav, dkk (2013)	*Comparative Study of K-NN, Naive Bayes and Decision Tree Classification Techniques*	✓			✓	✓			✓						✓
8	Achmad Ridwan (2020)	Penerapan Algoritma *Naïve Bayes* Untuk Klasifikasi Penyakit Diabetes Mellitus	✓							✓	✓	✓				
9	Fatmawati (2016)	Perbandingan Algoritma Klasifikasi *Data Mining* Model C4.5 Dan *Naive Bayes* Untuk Prediksi Penyakit Diabetes	✓		✓		✓			✓						
10	Dian Pramadhana (2021)	Klasifikasi Penyakit Diabetes Menggunakan Metode *CFS* dan *ROS* dengan Algoritma J48 Berbasis *Adaboost*						✓		✓	✓	✓				

Tabel 2.2 Matriks Penelitian (lanjutan)

11	Hiya Nalatissifa, dkk (2020)	Perbandingan Kinerja Algoritma Klasifikasi *Naive Bayes*, *Support Vector Machine (SVM)*, dan *Random Forest* untuk Prediksi Ketidakhadiran di Tempat Kerja	✓	✓					✓	✓	✓	✓				
12	Yusra, Dhita Olivita, dkk (2016)	Perbandingan Klasifikasi Tugas Akhir Mahasiswa Jurusan Teknik Informatika Menggunakan Metode Naïve Bayes Classifier dan K-Nearest Neighbor	✓			✓				✓						
13	Sri Diantika, dkk (2021)	Komparasi Algoritma SVM Dan Naive Bayes Untuk Klasifikasi Kestabilan Jaringan Listrik	✓	✓						✓	✓	✓				✓
14	Syifa Nazila Fauziah (2023)	Komparasi Algoritma *Naive Bayes* Dan *Support Vector Machine (SVM)* Untuk Klasifikasi Data Pada *Diabetes Prediction Dataset*	✓	✓						✓	✓	✓				