KOMPARASI METODE PEMBELAJARAN MESIN UNTUK IMPLEMENTASI PENGAMBILAN KEPUTUSAN DALAM MENENTUKAN PROMOSI JABATAN KARYAWAN

Pristian Luthfy Romadloni, Bagus Adhi Kusuma, Wiga Maulana Baihaqi
Program Studi Teknik Informatika S1, Fakultas Ilmu Komputer
Universitas Amikom Purwokerto, Jl. Let. Jend. Pol. Soemarto, Purwokerto, Indonesia
luthfypristian@gmail.com

ABSTRAK

Di era industri 4.0, perusahaan multinasional dituntut untuk beradaptasi dengan kemajuan teknologi dan perlu untuk bergerak cepat. Jenjang karier karyawan yang seimbang dengan beban kerja dan kebutuhan karyawan di lapangan, merupakan salah satu kunci pertahanan perusahaan. Selama ini tidak sedikit divisi pengelola sumber daya manusia di perusahaan yang menetapkan jenjang karier karyawan, dilakukan penilaian secara konvesional, yaitu dengan melihat nilai-nilai karyawan dari *database* perusahaan dan menimbang beberapa kriteria yang ada. Hal tersebut kemudian memakan waktu yang cukup lama. Sedangkan permintaan untuk menentukan promosi jabatan akan ada di setiap bulan. Kemudian dari masalah tersebut, dengan menggunakan data penelitian yang di peroleh dari perusahaan PT. Telkom Akses yang mana kriteria untuk promosi jabatan berbeda dengan perusahaan lain. Pada penelitian ini kami ingin memberikan solusi yaitu membuat perbandingan atau komparasi metode pembelajaran mesin untuk implementasi pengambilan keputusan dalam menentukan promosi jabatan karyawan, dengan menggunakan media aplikasi Rapidminer versi 9.10 dan dua metode, *Decision Tree* dan *Naïve Bayes*. Besar harapan kami dapat menjadi referensi khasanah ilmu pengetahuan baru di bidang Pengelolaan Sumber Daya Manusia. Pada penelitian ini, didapatkan hasil akurasi yang tertinggi yaitu, pada metode *Naïve Bayes* dengan nilai akurasi 92.29%, nilai presisi 97.05% dan nilai *recall* 89.86%.

Kata kunci : *Decision Tree, Komparasi, Naïve Bayes, Rapidminer, SDM.*

1. PENDAHULUAN

Pada perushaan, salah satu elemen yang sangat penting yaitu karyawan, dimana pengelolaan Sumber Daya Manusia/SDM akan sangat mempengaruhi banyak aspek penentu keberhasilan di perusahaan. Dalam perencanaan serta usaha untuk memenuhi kebutuhan SDM dilakukan kenaikan jabatan karyawan yang dikelola secara profesional sehingga dapat menentukan mutu serta kesuksesan perusahaan. Seleksi yang baik dan akurat dari kenaikan jabatan karyawan akan menghasilkan karyawan yang berkualitas untuk perusahaan [1].

Pada perusahaan PT. Telkom Akses, proses penentuan kenaikan jabatan karyawan masih menggunakan cara konvensional. Dimana HR Regional mengirim data usulan promosi jabatan karyawan yang kemudian dikolektif oleh HR Pusat untuk diolah dan ditentukan yang berhak dan tidak berhak promosi. Dalam prakteknya, dengan jumlah usulan promosi jabatan di setiap bulan lebih dari 300 usulan, diperlukan waktu 10 – 20 menit per karyawan untuk mempertimbangkan karyawan yang berhak dan tidak berhak promosi, dimana waktu tersebut relatif cukup lama hanya menentukan satu orang, karena harus dilakukan validasi disetiap parameter pada setiap variabel.

Maka dari masalah yang ada, pada penelitian ini dilakukan komparasi algoritma pengambilan keputusan menggunakan, terdapat dua algoritma pembanding yaitu Decision Tree dan Naïve Bayes, dan nantinya dalam pengolahan data akan dilakukan menggunakan software pendukung yaitu RapidMiner versi 9.10.

Pada penelitian ini, terdapat rumusan masalah yang dapat di garis bawahi, yaitu. Menggunakan komparasi dua algoritma, *Decision Tree* dan *Naïve Bayes* pada kasus penentuan promosi/kenaikan jabatan untuk mengetahui nilai akurasi, presisi, recall, dan F1 *Score* yang terbaik. Karena pada penelitian ini, data penelitian yang digunakan berbeda variabel/parameter dengan data penelitian lain, hal tersebut akan mempengaruhi penentuan hasil algoritma yang digunakan.

2. TINJAUAN PUSTAKA

2.1. Pengelolaan Sumber Daya Manusia (SDM)

Pengelolaan SDM merupakan suatu proses dimana pencapaian tujuan organisasi melalui penggunaan manusia di dalamnya. Tujuan pengelolaan karyawan, agar karyawan memiliki kompetisi serta keahlian yang dibutuhkan dalam mendukung pekerjaannya. Selain itu pengelolaan SDM juga dapat didefinisikan sebagai pendekatan strategis untuk pengelolaan aset yang paling berharga pada suatu organisasi, karena karyawan yang bekerja untuk organisasi tersebut, yang secara individu atau tim berkotribusi terhadap pencapain yang telah ditetapkan [2].

2.2. Pembelajaran Mesin

Pembelajaran mesin merupakan teknik untuk melakukan inferensi terhadap data dengan pendekatan matematis. Pada intinya, pembelajaran mesin adalah untuk membuat model (matematis) yang merekflesikan pola data. Teknik pembelajaran mesin juga menjadi semakin populer akibat kemampuan komputasi yang terus meningkat [3].

2.3. Data Penelitian

Data didefinisikan sebagai deskripsi maupun keterangan sebuah objek yang belum memiliki makna secara utuh. Data dapat berupa angka, huruf/teks, gambar, suara, maupun lambang/simbol. Data akan sangat menentukan tingkat validitas keluaran (output) yang dihasilkan. Salah satu yang mententukan tingkat validitas yaitu kesesuaian antara jenis data dan tools analisys yang digunakan [4].

2.4. Klasifikasi Data

Klasifikasi data merupakan pengelompokan data berdasarkan beberapa aspek, diantaranya: berdasarkan sumber data, cara memperolehnya, waktu pengumpulan, jenis data (data primer dan sekunder misalnya), dan sifat data. Teknik Klasifikasi menggunakan algoritma pembelajaran mesin untuk mengidentifikasi model yang memberikan hubungan yang paling sesuai antara himpunan atribut dan label kelas dari data input. Model yang dibangun dengan sebuah algoritme pembelajaran haruslah sesuai dengan data input dan memprediksi dengan benar label kelas dari record yang belum pernah terlihat sebelumnya [5].

2.5. Algoritma *Decision Tree*

Algoritma *Decision Tree* digunakan untuk memodelkan persoalan yang terdiri dari berbagai serangkaian keputusan dan mengarah ke solusi. Tiap simpul adalah solusi sedangkan daun adalah keputusan. Konsep *Decision Tree* yaitu mengubah tumpukan data menjadi *Decision Tree* yang mempresentasikan aturan - aturan dari sebuah keputusan [6].

2.6. Algoritma *Naïve Bayes*

Algoritma *Naïve Bayes* merupakan sebuah metode klasifikasi yang menggunakan metode probabilitas statistik. *Naïve Bayes* memprediksi peluang di masa depan berdasarkan pengalaman di masa lampaunya karena hal tersebut *Naïve Bayes* dapat digunakan untuk pengambilan keputusan. *Naïve Bayes* ini mempunyai ciri utama yaitu asumsi yang sangat kuat akan kemandirian/independensi dari masing-masing kondisi/kejadian [7].

2.7. Perangkat Lunak RapidMiner 9.10

RapidMiner bekerja untuk memberikan solusi dengan melakukan analisis terhadap data mining, text mining dan analisis prediksi. RapidMiner menggunakan berbagai teknik deskriptif dan prediksi dalam memberikan wawasan kepada user sehingga dapat membuat keputusan yang paling baik.

Pada aplikasi RapidMiner terdapat operator *Cross Validation* yang merupakan metode statistik dan digunakan untuk mengevaluasi kinerja model atau algoritma, dimana data dipisahkan menjadi dua subset yaitu *data trainning* dan *data testing* [8].

Biasanya *Cross Validation* atau K-fold digunakan karena dapat mengurangi waktu komputasi dengan tetap menjaga keakuratan estimasi, K-fold dengan K = 10, merupakan salah satu yang direkomendasikan untuk pemilihan model terbaik karena cenderung memberikan estimasi akurasi yang lebih baik dan terukur.

Pada K-*fold* K= 10, data dibagi menjadi 10*fold* berukuran sama, sehingga kita memiliki 10 *subset* data untuk mengevaluasi kinerja model atau algoritma. Untuk masing-masing dari 10 *subset* data tersebut, *Cross Validation* akan menggunakan 9*fold* untuk pelatihan dan 1fold untuk pengujian seperti diilustrasikan pada Tabel 1. Skema *Cross Validation*.

Tabel 1. Skema *Cross Validation*.

1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10

Data Testing
DataTrainning

2.8. *Confusion Matrix*

Confusion Matrix digunakan sebagai pengukur kinerja setelah mengolah *data mining* dengan model klasifikasi. Pada dasarnya, pengukuran dengan *Confusion Matrix* digunakan untuk memberikan informasi perbandingan dari hasil klasifikasi yang dilakukan oleh algoritma yang digunakan dengan hasil klasifikasi sebenarnya. Disajikan pada Tabel 2. *Confusion Matrix*.

Tabel 2. *Confusion Matrix*.

	ACTUAL	
PREDICTED	*TRUE*	*FALSE*
TRUE	*TP (True Positive)*	*FP (False Positive)*
FALSE	*TN (True Negative)*	*FN (False Negative)*

Pada tabel di atas, terdapat empat istilah yang merepresentasikan hasil proses klasifikasi pada confusion matrix, yaitu:

- *True Positive* (TP), data positif yang diprediksi benar.
- *True Negative* (TN), data negatif yang diprediksi benar.
- *False Positive* (FP), error tipe 1, data negatif namun diprediksi sebagai data positif.
- *False Negative* (FN), error tipe 2, data positif namun diprediksi sebagai data negatif.

Pada prakteknya confusion matrix digunakan untuk menghitung nilai akurasi, presisi, dan recall.

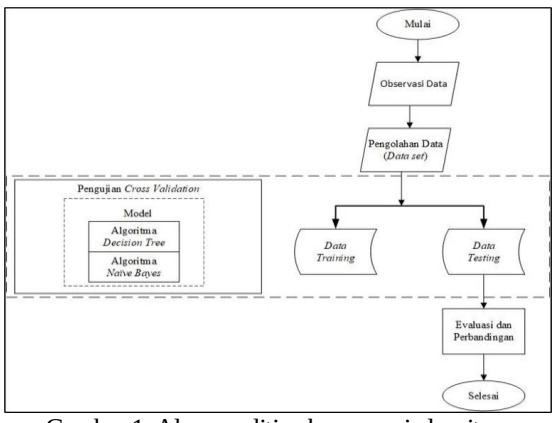
- Nilai akurasi, menggambarkan seberapa akurat model dalam mengklasifikasikan dengan benar.
  Akurasi = (TP+TN) / (TP+FP+FN+TN)
- Presisi, menggambarkan akurasi antara data yang diminta dengan hasil prediksi yang diberikan oleh model.
  Presisi = (TP) / (TP + FP)
- *Recall* atau sensitivity, menggambarkan keberhasilan model dalam menemukan kembali sebuah informasi.
  Recall = TP / (TP + FN)
- F1 *Score*, merupakan perbandingan rata-rata presisi dan recall yang dibobotkan
  F1 Score = 2 * (Recall*Presisi) / (Recall +Presisi) [9].

Dikemukakan oleh Salma Ghoneim seorang karyawan di Microsoft, untuk memilih acuan perbandingan performansi algoritma machine learning yang akan dipilih dan tepat guna, yaitu:

- Pilih algoritma yang memiliki Akurasi yang tinggi jika, *dataset* yang dimiliki jumlah data *False* Negatif dan *False* Positif yang sangat mendekati (Symmetric). Namun jika jumlahnya tidak mendekati, maka sebaiknya gunakan F1 Score sebagai acuan.
- Pilih algoritma yang memiliki Presisi yang tinggi jika, menginginkan terjadinya *True* Positif dan sangat tidak menginginkan terjadinya *False* Positif.
- Pilih algoritma yang memiliki *Recall* yang tinggi jika, lebih memilih *False* Positif lebih baik terjadi daripada *False* Negatif. *Recal*l digunakan karena lebih baik algoritma memprediksi positif tetapi sebenarnya false daripada algoritma salah memprediksi *false* padahal sebenarnya positif [10].

3. METODE PENELITIAN

Metode penelitian ini menggunakan alur penelitian seperti yang disajikan pada Gambar alur penelitian komparasi algoritma. Alur penelitian ini dimulai dari observasi data, pengolahan data menjadi data set, pengujian dengan menggunakan *cross validation* pada setiap model agoritma, diakhiri dengan evaluasi hasil dan perbandingan kinerja menggunakan *confusion matrix*.



Gambar 1. Alur penelitian komparasi algoritma

Dapat dijelaskan diagram alur penelitian ini seperti yang telah disajikan di atas, sebagai berikut:

1. Observasi, pada tahap ini dilakukan untuk mengumpulkan data sebagai bahan penelitian, observasi ini peneliti melakukan wawancara dengan para pihak yang berperan di penelitian ini, studi kepustakaan, hingga dokumentasi.
2. Pengolahan data, setelah tahap observasi selesai dilakukan selanjutnya adalah pengolahan data, data mentah yang didapat dari hasil observasi diolah, diklasifikasikan berdasarkan atribut/parameter menjadi sebuah *data set*.
3. Pengujian, pada tahap ini dilakukan pengujian algoritma dengan menggunakan teknik *Cross Validation*. Dimana teknik tersebut membagi *data set* penelitian menjadi data training dan data testing.
4. Evaluasi dan Perbandingan, pada tahap ini dilakukan pengamatan hasil *cross validation* dan perhitungan dengan metode *confusion matrix* untuk mengetahui metode algoritma yang nilai akurasi, prediksi, dan recall yang paling tinggi.

4. HASIL DAN PEMBAHASAN

4.1. Pengolahan Data

Pengolahan data pada penelitian ini, dilakukan secara klasifikasi dimana parameter atribut yang ditandai sebagai atribut penting dalam penentu pengambilan keputusan. Pada Tabel 3. Atribut parameter penentu keputusan dirincikan sebagai berikut.

Tabel 3. Atribut parameter penentu keputusan

Variabel	Atribut	Parameter
X1	Masa Kerja Dalam Tahun	2 Tahun/Lebih
		2 Tahun Kurang
X2	SK Terakhir	None
		1 Tahun Kurang
		1 Tahun/Lebih
X3	BPJS	Sudah Terdaftar/Proses
		Belum Terdaftar
X4	Jabatan	Ada
		Formasi Penuh
X5	Sumber Pengajuan	Ada
		Tidak Ada
X6	Apraisal	C1
		C2
		C3
		C4
		C5
X7	Performansi	P1
		P2
		P3
Y	Status	OK
		NOK

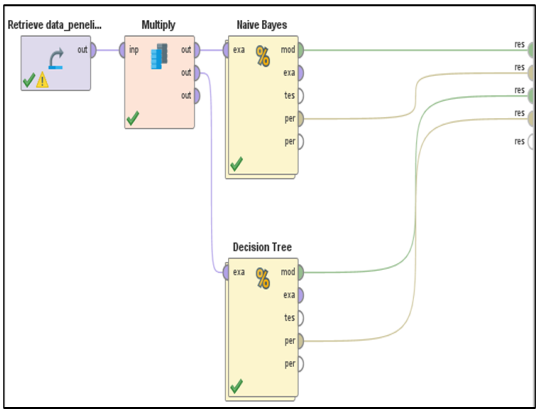
Keterangan pada Tabel 3. Atribut parameter penentu keputusan, terdapat tujuh variabel X dan satu varibel Y yang dapat diuraikan sebagai berikut.

- X1, yaitu variabel masa kerja. Pada variabel tersebut mempunyai dua kategori, yaitu 2 tahun kurang dan 2 tahun/Lebih. Dimana untuk masa kerja dibawah dua tahun, tidak dapat promosi jabatan.

- X2, yaitu variabel SK terakhir. Pada variabel tersebut mempunyai dua kategori, yaitu 1 tahun kurang dan 1 tahun/lebih. Dimana untuk rentang waktu karyawan menerima SK (Surat Keputusan) adalah satu tahun.
- X3, yaitu variabel BPJS mempunyai keterangan Sudah terdaftar/Diproses dan Belum terdaftar. Dimana variabel ini digunakan untuk memeriksa kedisplinan karyawan dalam memperhatikan jaminan kesehatan dengan mengurus ke divisi *Human Capital Management.*
- X4, yaitu varibel Jabatan. Pada variabel tersebut mempunyai dua kategori, yaitu ada dan formasi Penuh. Dimana untuk promosi jabatan, harus tersedia kuota jabatan. Untuk mengetahui ketersediaan kuota jabatan, informasi tersebut dapat diperoleh di divisi *Human Capital Management.*
- X5, yaitu variabel Sumber (Nota dinas). Pada variabel tersebut mempunyai dua kategori, yaitu ada dan tidak. Nota dinas digunakan untuk memeriksa atas rekomendasi siapa yang mengusulkan karyawan tersebut promosi jabatan. Nota dinas hanya dapat diterbitkan oleh level manajer ke atas.
- X6, yaitu variabel Nilai Kompetensi/Apraisal. Pada variabel tersebut mempunyai lima kategori yaitu C1, C2, C3, C4, dan C5. Dimana C adalah *Competency*/Kompetensi kinerja karyawan. Nilai kompetensi ini diambil setiap enam bulan sekali, atas penilaian dari sesama karyawan, atasan, dan divisi *Human Capital Management*. C1 merupakan kategori Sangat Kompeten hingga C5 yang merupakan Sangat Tidak Kompeten.
- X7, yaitu variabel Nilai Performansi. Pada variabel tersebut mempunyai tiga kategori yaitu P1, P2, dan P3. Dimana P adalah *Performance*/Performansi kinerja karyawan. Nilai performansi ini diambil setiap enam bulan sekali, atas penilaian dari sesama karyawan, atasan, dan divisi *Human Capital Management*. P1 merupakan kategori performansi Sangat Baik hingga P3 yang merupakan performansi Buruk.
- Y, yaitu variabel Status. Pada variabel ini merupakan penentu dari setiap kategori variabel di atas. Variabel Y mempunyai dua kategori, yaitu OK dan *Not* OK (NOK).

4.2. Pengujian

Pengujian dilakukan dengan aplikasi RapidMiner versi 9.10 dengan metode *Cross Validation* dan menggunakan dua algoritma yang berbeda, yaitu *Naïve Bayes* serta *Decision Tree*. Disajikan pada Gambar 2. Desain model perbandingan algoritma yang digunakan pada penelitian ini.



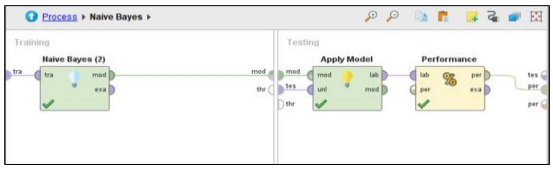
Gambar 2. Desain model perbandingan algoritma

Dari gambar desain diatas, dapat dijabarkan menjadi beberapa tahap untuk pengujian pada penelitian ini, yaitu:

a. Data Penelitian yang sudah direduksi dan siap digunakan pada peneletian ini.
b. *Multiply*, digunakan untuk membuat salinan objek pada RapidMiner. Operator ini mengambil objek dari *port input* dan mengirimkan salinannya ke port output. Setiap *port* yang terhubung membuat salinan yang tidak terikat.
c. *Cross Validation*, digunakan untuk melakukan validasi berulang di mana dataset dibagi menjadi banyak himpunan atau latih dan validasi. Setiap iterasi memvalidasi satu himpunan data dengan himpunan yang tersisa sebagai data latih.
d. *Result*, pada penelitian ini result yang ditampilkan adalah model dan *performance* dari masing-masing algoritma yang digunakan untuk pengujian.

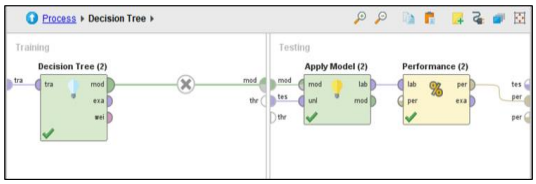
Dari beberapa tahap di atas, pada tahap cross validation terdapat dua katergori *cross validation* yaitu *Naïve Bayes* dan *Decision Tree*. Kemudian disetiap kategori *cross validation* dapat dijabarkan lagi menjadi beberapa tahap.

Pada gambar di bawah ini, disajikan Gambar 3 dan Gambar 4. Proses *cross validation* pada *Naïve Bayes.*



Gambar 3. Proses *cross validation* pada *Naïve Bayes*

Proses *cross validation* pada *Decision Tree.*



Gambar 4. Proses *cross validation* pada *Decision Tree.*

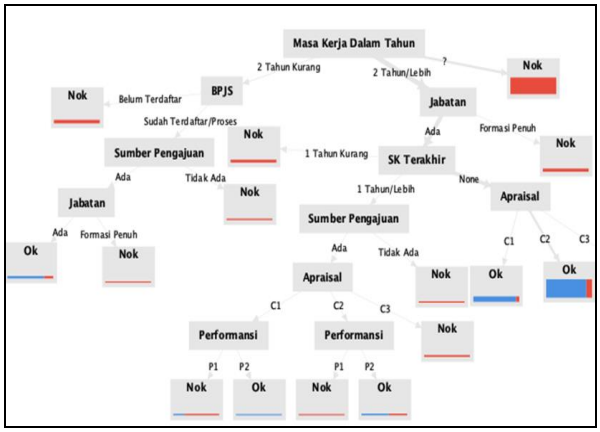
Pada dua gambar di atas, rangkaian desain hanya dibedakan di kolom *training*, kolom *training* diisi dengan metode *Naïve Bayes* dan metode *Decision Tree*. Selanjutnya pada kolom *testing* diberi operator *Apply Model* dan *Performance*. Dimana *Apply Model* digunakan untuk menerapkan model yang telah dilatih sebelumnya, menggunakan *data training* dan memiliki tujuan mendapatkan prediksi pada *data testing* yang belum memiliki label. Sedangkan *Performance* digunakan untuk mengukur hasil kinerja model dan memberikan daftar nilai kriteria kinerja secara otomatis sesuai tugas yang diberikan.

4.3. Evaluasi Hasil Komparasi

Setelah data set dan desain penelitian dibuat, kemudian dilakukan penelitian komparasi perbandingan metode *Naïve Bayes* dan *Decision Tree*. Penelitian ini dilakukan dengan *Cross Validation* mulai dari K = 2 hingga K = 10 dengan jenis pengambilan tipe sampel *Automatic*. Hasil dari penelitian didapatkan sebagai berikut:

1. Metode *Naïve Bayes*
   Diketahui pada penelitian ini, terdapat tujuh atribut dan dua class dari data training dengan hasil keterangan "OK" sebanyak 142 data dari 350 data sedangkan hasil dengan keterangan "NOK" sebanyak 208 data dari 350 data. Maka, dapat diperhitungkan bahwa:
   Class OK = 142 ÷ 350 = 0.406
   Class NOT = 208 ÷ 350 = 0.594
   Untuk memprediksi dengan menggunakan metode *Naïve Bayes*, sistem akan menggunakan perhitungan. Jika prediksi nilai OK lebih besar dari NOK maka *class data* tersebut adalah OK. Begitu pula sebaliknya, jika nilai NOK lebih besar dari OK maka *class data* tersebut adalah NOK.

2. Metode *Decision Tree*
   Metode Decision Tree pada penelitian ini digunakan sebagai pembanding dari metode Naïve Bayes. Disajikan pada Gambar 5. Model *Decision Tree* yang didapat dari RapidMiner.



Gambar 5. Model *Decision Tree* yang didapat dari RapidMiner

3. Hasil Komparasi
   Hasil Komparasi yang didapat dari penelitian ini, dilakukan perbandingan dari *Cross Validation* dengan nilai K = 2 hingga K = 10. Disajikan pada Tabel 4. Komparasi *Cross Validation*. Tabel tersebut menampilkan hasil komparasi akurasi, prediksi, dan *recall* dari metode *Naïve Bayes* dan *Decision Tree* sebanyak sembilan kali perulangan.

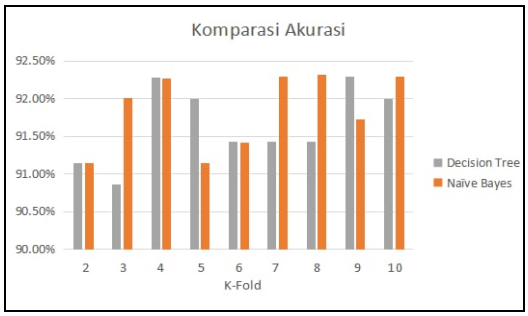
Tabel 4. Komparasi *Cross Validation*

K-Fold	Naïve Bayes			Decision Tree		
	Akurasi	Presisi	Recall	Akurasi	Presisi	Recall
2	91.14%	94.95%	89.90%	91.14%	95.86%	88.45%
3	92.01%	96.54%	89.91%	90.86%	97.98%	86.51%
4	92.27%	96.03%	90.87%	92.28%	98.44%	88.46%
5	91.14%	96.01%	88.95%	92.00%	98.45%	88.95%
6	91.42%	95.81%	89.90%	91.43%	98.51%	86.96%
7	92.29%	96.56%	90.38%	91.43%	98.03%	87.57%
8	92.31%	96.90%	89.90%	91.43%	97.30%	87.98%
9	91.72%	96.50%	89.43%	92.29%	98.43%	88.41%
10	92.29%	97.05%	89.86%	92.00%	97.95%	88.45%

Dari Tabel 4. Komparasi *Cross Validation* di atas dapat dijelaskan bahwa pada pengujian *Cross Validation*/K-*fold* dimulai dari nilai K = 2. Setelah dilakukan perhitungan hingga K=10, metode *Naïve Bayes* dan metode *Decision Tree* memiliki hasil perhitungan yang bersifat dinamis, dimana pada setiap uji coba nilai K-*fold* memiliki hasil yang berbeda.

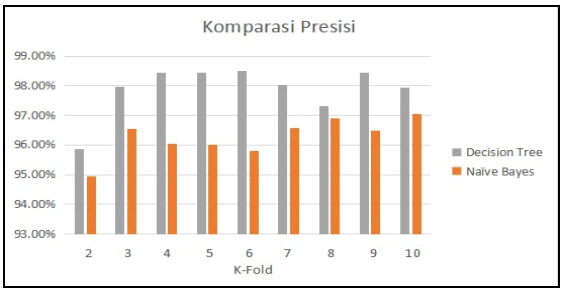
Untuk memudahkan visualisai, hasil pengujian K-fold ditampilkan pada gambar di bawah ini.

a. Ditampilkan pada Gambar 6. Grafik Komparasi Akurasi.



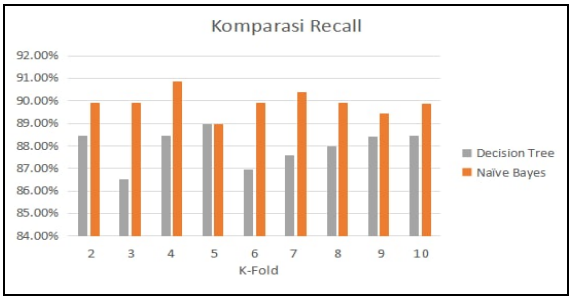
Gambar 6. Grafik Komparasi Akurasi

b. Ditampilkan pada Gambar 7. Grafik Komparasi Presisi.



Gambar 7. Grafik Komparasi Presisi.

c. Ditampilkan pada Gambar 8. Grafik Komparasi *Recall*.



Gambar 8. Grafik Komparasi *Recall*

Dari tabel dan diagram di atas diketahui nilai akurasi, prediksi, dan recall yang tidak terlampau jauh, diambil pada data K-*fold* ke sepuluh untuk mencari nilai F1 *Score*. F1 *Score* merupakan perbandingan rata-rata presisi dan *recall* yang dibobotkan. Dengan rumus F1 *Score* = 2 * (Presisi * *Recall*) / (Presisi + *Recall*). Maka dapat diketahui, sebagai berikut.

A. Nilai F1 *Score* pada Naïve Bayes

F1 *Score* = 2 * (Presisi * *Recall)* / (Presisi + *Recall*)
= 2 * (97.05% * 89.86%) / (97.05% + 89.86%)
= 0.9331

B. Nilai F1 Score pada Decision Tree

F1 Score = 2 * (Presisi * *Recall*) / (Presisi + *Recall*)
= 2 * (97.95% * 88.45%) / (97.95% + 88.45%)
= 0.9295

Pada RapidMiner, ditujukan pada tabel dibawah ini hasil dari K-*fold*, dengan nilai K = 10 dari *Naïve Bayes*.

Tabel 5. Hasil Perhitungan K-fold, K=10 *Naïve Bayes*

accuracy: 92.29% +/- 3.82% (micro average: 92.29%)			
	true Ok	*true Nok*	*class precision*
pred. Ok	136	21	86.62%
pred. Nok	6	187	96.89%
class recall	95.77%	89.90%	
precision: 97.05% +/- 4.13% (micro average: 96.89%)			
	true Ok	*true Nok*	*class precision*
pred. Ok	136	21	86.62%
pred. Nok	6	187	96.89%
class recall	95.77%	89.90%	
recall: 89.86% +/- 4.34% (micro average: 89.90%)			
	true Ok	*true Nok*	*class precision*
pred. Ok	136	21	86.62%
pred. Nok	6	187	96.89%
class recall	95.77%	89.90%	

Pada Tabel 5. Hasil Perhitungan K-fold, K = 10 *Naïve Bayes*, didapatkan nilai akurasi 92.29% dengan kenaikan disetiap perulangan sebesar 3.82%, Nilai presisi 97.05% dengan kenaikan disetiap perulangan sebesar 4.13%, dan nilai recall 89.86% dengan kenaikan disetiap perulangan sebesar 4.34%. Serta nilai recall masing – masing kelas sebesar 95.77 % untuk kelas OK dan 89.90% untuk kelas NOK. Kemudian hasil tabel dibawah ini hasil dari K = 10 dari *Decision Tree*.

Tabel 6. Hasil Perhitungan K-fold, K=10 *Decision Tree*

accuracy: 92.00% +/- 3.24% (micro average: 92.00%)			
	true Ok	*true Nok*	*class precision*
pred. Ok	138	24	85.19%
pred. Nok	4	184	97.87%
class recall	97.18%	88.46%	
precision: 97.95% +/- 3.63% (micro average: 97.87%)			
	true Ok	*true Nok*	*class precision*
pred. Ok	138	24	85.19%
pred. Nok	4	184	97.87%
class recall	97.18%	88.46%	
recall: 88.45% +/- 4.06% (micro average: 88.46%)			
	true Ok	*true Nok*	*class precision*
pred. Ok	138	24	85.19%
pred. Nok	4	184	97.87%
class recall	97.18%	88.46%	

Pada Tabel 6. Hasil Perhitungan K-*fold*, K = 10 Decision Tree, didapatkan nilai akurasi 92.00% dengan kenaikan disetiap perulangan sebesar 3.24%, nilai presisi 97.95% dengan kenaikan disetiap perulangan sebesar 3.63%, dan nilai *recall* 88.45% dengan kenaikan disetiap perulangan sebesar 4.06%. Serta dan nilai *recall* masing – masing kelas sebesar 97.18 % untuk kelas OK dan 88.46% untuk kelas NOK.

Dari pengujian yang telah dilakukan sebanyak K = 10 pada setiap algoritma *Naïve Bayes* dan metode Decision Tree, maka metode *Naïve Bayes* dapat diimplementasikan sebagai algoritma pengambil keputusan untuk promosi jabatan level *Team Leader* di perusahaan, khususnya PT. Telkom Akses.

5. KESIMPULAN DAN SARAN

Pada penelitian ini, setelah ditentukan parameter – parameter untuk menentukan promosi jabatan Team Leader dan dilakukan pengujian *Cross Validation* dengan nilai K = 2 hingga K=10, antara metode *Naïve Bayes* dan metode *Decision Tree*. Memiliki hasil perhitungan yang bersifat dinamis, dimana pada setiap uji coba nilai K-*fold* memiliki hasil yang berbeda, serta *gap* hasil komparasi akurasi, presisi, *recall* tidak terlampau jauh selanjutnya dipilih nilai F1 *Score* yang terbaik. Sehingga dapat disimpulkan bahwa metode *Naïve Bayes* dapat digunakan sebagai algoritma pengambil keputusan menentukan promosi jabatan level *Team Leader* di Perusahaan PT. Telkom Akses.

Dengan metode pembelajaran mesin di aplikasi RapidMiner versi 9.10, metode *Naïve Bayes* mampu memangkas waktu dan menghasilkan keluaran

dengan nilai akurasi yang baik untuk menentukan promosi jabatan *Team Leader*.

Pada penelitian ini, masih banyak kekurangan yang seharusnya dapat menjadi koreksi untuk penelitian lebih lanjut. Saran yang dapat kami sampaikan untuk penelitian lebih lanjut adalah pembuatan sistem di berbasis web untuk implementasi hasil metode *Naïve Bayes* sebagai metode pengambilan keputusan promosi jabatan *Team Leader* PT. Telkom Akses.

DAFTAR PUSTAKA

[1] Widodo, Anang, A., & Misdram. (2019). Sistem Pendukung Keputusan Kenaikan Jabatan Menggunakan Metode Profile Matching (Studi Kasus: Pt. Metsuma Anugrah Graha). Jurnal Universitas Merdeka Pasuruan, 2.

[2] Suryani, N. K., & Foeh, J. (2019). Manajemen Sumber Daya Manusia: Tinjauan Praktis Aplikatif. Bali: Nilacakra.

[3] Yusuf, M., & Dari, L. (2019). Analisis data penelitian: teori & aplikasi dalam bidang perikanan. Bogor: Penerbit IPB Press.

[4] Prasetyowati, E. (2017). Data mining pengelompokan data untuk informasi dan evaluasi. Pamekasan: Duta Media Publishing.

[5] Putra, J. W. (2020). Pengenalan Konsep Pembelajaran Mesin dan Deep Learning. Tokyo: ResearchGate.

[6] Yuswardi. (2022). Sistem Pendukung Keputusan Pada Teknologi Informasi. Padang: Get Press.

[7] Isa, I. G. (2022). Buku Ajar Sistem Pendukung Keputusan. Pekalongan: Penerbit NEM.

[8] RapidMiner. (2022, Februari 11). RapidMiner Documentation. Retrieved from RapidMiner 9: Operator Reference Manual: https://docs.rapidminer.com/latest/studio/operators/rapidminer-studio-operator-reference.pdf

[9] Mustika. (2021). Data Mining Dan Aplikasinya. Widina.

[10] Ghoneim, S. (2019, April 2). Accuracy, Recall, Precision, F-Score & Specificity, which to optimize on? Retrieved Juli 14, 2022, from towardsdatascience: https://towardsdatascience.com/accuracy-recall-precision-f-score-specificity-which-to-optimize-on-867d3f11124.