Prediksi *Employee Attrition* menggunakan Algoritma *Support Vector Machine* (SVM)

1st Muhammad Abdurrohman Al Fatih
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
abdurrahmanalfatih@students.telkomuniversity.ac.id

2nd Kemas Muslim Lhaksmana
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
kemasmuslim@telkomuniversity.ac.id

Abstrak-*Employee attrition* atau keluarnya karyawan dari perusahaan adalah sebuah tantangan, mengingat karyawan merupakan salah satu aset penting bagi perusahaan. Tingkat *employee attrition* yang tinggi menandakan bahwa seringnya para karyawan keluar dari perusahaan. Hal ini akan merugikan perusahaan dari sisi waktu, biaya, sumber daya manusia dan juga membuat citra perusahaan turun. Perlunya untuk menganalisis dan memprediksi *employee attrition* agar dapat dilakukan tindakan preventif dan persuasif sehingga karyawan tidak keluar dari perusahaan. Oleh karena itu, dibutuhkannya sebuah *tools* atau alat bantu untuk memprediksi apakah seorang karyawan akan keluar dari perusahaan. Pada penelitian ini dilakukan implementasi model *machine learning* untuk memprediksi *employee attrition* dan membandingkan performansi antara algoritma *support vector machine* (SVM) dengan algoritma *k-Nearest Neighbors* (kNN). Data set yang digunakan adalah data set IBM HR *Analytics Employee Attrition & Performance*. Kedua model dievaluasi dengan menggunakan metrik *accuracy*, *F1-score,* dan *geometric-mean*. Hasil dari penelitian ini menunjukkan bahwa model dengan algoritma SVM memiliki nilai metrik yang lebih baik daripada algoritma kNN dengan rata-rata *accuracy* 0.86, *F1-score* 0.59, dan *geometric-mean* 0.75. Ini menandakan bahwa model dengan algoritma SVM lebih baik dalam memprediksikan data ke dalam kelas *attrition* dan kelas *not-attrition* daripada model dengan algoritma kNN.

Kata kunci - prediksi, *employee attrition*, karyawan, *machine learning*.

Abstract-Employee attrition is a challenge, considering that employees are one of the important assets of the company. A high level of employee attrition indicates that employees often resign from the company. That will harm the company in terms of time, cost, human resources, and also make the company's image down. It is necessary to analyze and predict employee attrition to take preventive and persuasive actions. Hence, the employees do not leave the company. Therefore, a tool is needed to predict whether an employee will resign from the company or not. In this study, a machine learning model was implemented to predict employee attrition and compared the performance of the Support Vector Machine (SVM) algorithm with the k-Nearest Neighbor (kNN) algorithm. The data set used are the IBM HR Analytics Employee Attrition & Performance data set. Both models were evaluated using accuracy, F1-score, and geometric-mean metrics. The results of this study indicate that the model with the SVM algorithm has a better metric value than the kNN algorithm. The model with the SVM algorithm has an average accuracy of 0.86, an F1-score of 0.59, and a geometric-mean of 0.75. The model with the SVM algorithm is better at predicting the data into the attrition class and the not-attrition class than the model with the kNN algorithm.

Keywords- prediction, employee attrition, employee, machine learning.

I. PENDAHULUAN

A. Latar Belakang

Kata karyawan menurut Kamus Besar Bahasa Indonesia (KBBI) adalah orang yang bekerja pada suatu lembaga (kantor, perusahaan, dan sebagainya) dengan mendapat gaji (upah). Karyawan direkrut untuk memenuhi sumber daya yang dibutuhkan oleh perusahaan dalam mencapai tujuannya. Karyawan dapat disebut sebagai salah satu aset perusahaan dikarenakan sebuah perusahaan tidak dapat mencapai tujuannya tanpa adanya bantuan dari karyawan.

Employee attrition atau keluarnya karyawan dari perusahaan adalah sebuah tantangan untuk perusahaan mengingat karyawan merupakan salah satu aset penting bagi perusahaan. Perekrutan karyawan baru tentunya merupakan salah satu solusi untuk menangani dampak dari *employee attrition* ini. Namun, perekrutan karyawan tentunya tidak mudah dan memerlukan waktu, biaya, dan sumber daya lain [1]. Berdasarkan penelitian yang dilakukan oleh Marsden [2], untuk setiap karyawan yang keluar dari perusahaan, perusahaan terbebani sebanyak 1 sampai 1.2 gaji tahunan karyawan tersebut. Oleh karena itu, perekrutan karyawan baru mungkin bukanlah solusi yang "murah".

Keluarnya karyawan berdampak pada kualitas perusahaan apabila karyawan tersebut merupakan karyawan yang berkualitas. Keluarnya karyawan juga dapat berdampak pada karyawan yang lain, terutama karyawan-karyawan lain yang sebelumnya tergabung dalam tim yang sama. Hal tersebut dapat mengganggu dinamika tim yang sudah ada. Oleh karena itu, dibutuhkannya solusi lain untuk mencegah *employee attrition* ini.

Perlunya untuk menganalisis dan memprediksi *employee attrition* agar dapat melakukan tindakan

preventif dan persuasif sehingga dapat mencegah kerugian-kerugian yang dapat disebabkan oleh keluarnya karyawan. Oleh karena itu, dibutuhkannya sebuah *tools* atau alat bantu untuk memprediksi apakah seorang karyawan akan keluar dari perusahaan.

Pada era digital 4.0 ini penggunaan *machine learning* sudah sangat luas. *Machine learning* biasa digunakan untuk melakukan klasifikasi dan klasterisasi data. *Machine learning* melakukan klasifikasi dengan mencari rumus atau pola data yang serupa dengan data yang lainnya. Hasil dari klasifikasi *machine learning* dapat digunakan sebagai prediksi. *Machine learning* dapat membuat prediksi data dengan menggunakan algoritma *support vector machine* (SVM), *k-nearest neighbors* (kNN), dan algoritma klasifikasi lainnya [3]. Prediksi menggunakan *machine learning* ini bersifat otomatis. Hal ini dimanfaatkan untuk memprediksi *employee attrition*.

Hasil klasifikasi dari *machine learning* bergantung pada data set. Data set yang distribusi kelasnya seimbang lebih baik daripada data set yang distribusi kelasnya tidak seimbang, dan data set yang jumlah datanya banyak lebih baik daripada data set yang jumlah datanya sedikit [4]. Di *real-world*, permasalahan klasifikasi biasanya memiliki data set yang distribusi kelasnya tidak seimbang dan jumlah datanya sedikit. Data set yang seperti itu biasanya disebut sebagai *imbalanced dataset* dan memiliki dampak buruk pada klasifikasi *machine learning*. Oleh karena itu, pada penelitian ini dilakukan beberapa cara untuk menangani *imbalanced dataset* agar hasil klasifikasi menjadi lebih baik. Pada penelitian ini juga dibandingkan performansi algoritma SVM dan kNN dikarenakan keduanya sama-sama merupakan algoritma yang berbasis jarak, tetapi memiliki pendekatan yang berbeda.

B. Topik dan Batasannya

Berdasarkan latar belakang yang telah diuraikan sebelumnya, didapat rumusan masalah sebagai berikut:

1. Bagaimana cara implementasi *machine learning* untuk memprediksi *employee attrition*?
2. Bagaimana performansi algoritma SVM jika dibandingkan dengan algoritma kNN pada permasalahan prediksi *employee attrition*?
3. Bagaimana pengaruh penanganan *imbalanced dataset* pada algoritma SVM?

Kemudian, penelitian ini memiliki batasan sebagai berikut:

1. Dataset yang digunakan adalah dataset IBM HR *Analytics Employee Attrition & Performance*;
2. Algoritma yang dibahas pada penelitian ini hanya algoritma kNN dan algoritma SVM.

C. Tujuan

Berdasarkan rumusan masalah yang telah diuraikan sebelumnya, didapat tujuan penelitian sebagai berikut:

1. Implementasi model *machine learning* untuk memprediksi *employee attrition*;
2. Membandingkan performansi algoritma SVM dengan algoritma kNN pada permasalahan prediksi *employee attrition*;
3. Mengetahui pengaruh penanganan *imbalanced dataset* pada algoritma SVM.

D. Organisasi Tulisan

Terdapat 5 bagian pada jurnal Tugas Akhir ini. Bagian pertama adalah mendeskripsikan latar belakang, perumusan masalah, batasan, dan tujuan Penelitian ini. Bagian kedua adalah membahas mengenai literatur atau studi yang mengacu dan mendukung tentang penelitian tugas akhir ini yang telah dilakukan sebelumnya. Bagian ketiga adalah pembangunan sistem klasifikasi yang terdiri dari pengumpulan data set, pemrosesan data set, dan membagi data set menjadi 2 bagian (data set latih dan data set uji). Bagian keempat adalah klasifikasi dan analisis hasil evaluasi pengujian dari sistem klasifikasi. Bagian terakhir adalah kesimpulan hasil pengujian beserta saran untuk penelitian selanjutnya.

II. KAJIAN TEORI

A. Employee Attrition

Employee attrition atau keluarnya karyawan dari suatu perusahaan sebenarnya adalah sebuah peristiwa yang biasa saja terjadi. Namun, tingkat *employee attrition* yang tinggi dapat menandakan adanya masalah pada perusahaan yang membuat banyak karyawan keluar dari perusahaan tersebut. Menurut Fred, beberapa penyebab dari *employee attrition* ini adalah kurangnya kesempatan karyawan untuk berkembang, manajer atau *supervisor* yang buruk, gaji yang rendah, dan yang lain-lain [5].

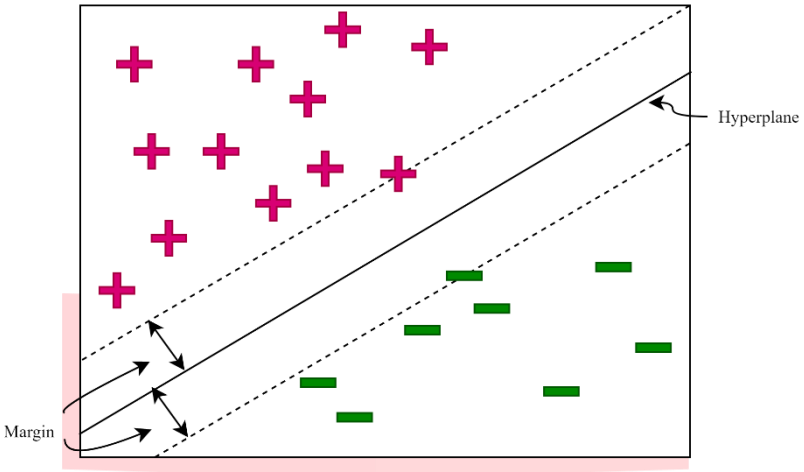
Tingkat *employee attrition* yang terus naik dapat menandakan bahwa ada masalah serius pada perusahaan tersebut. Tidak mudah untuk menentukan tingkat *employee attrition* yang baik, terdapat banyak hal yang dapat mempengaruhi untuk penilaian hal ini seperti seberapa besar perusahaannya, di bidang apa perusahaan tersebut bergerak, dan lain lain. Namun, yang perlu ditekankan adalah tingkat *employee attrition* yang baik adalah ketika pada tingkat tersebut perusahaan dapat berjalan dengan lancar, tingkat kesejahteraan karyawan yang baik, dan performa karyawan tidak menurun.

B. Support Vector Machine

Support vector machine (SVM) adalah salah satu algoritma klasifikasi *machine learning* yang memiliki konsep untuk menemukan *hyperplane* yang memisahkan himpunan data ke dalam dua kelas [6].

Berdasarkan penelitian yang dilakukan oleh Kotsiantis dkk. [7], SVM merupakan algoritma yang memiliki tingkat akurasi paling tinggi secara umum pada 45 kali pengujian dengan data set yang berbeda.

Dalam penelitian ini SVM menemukan *hyperplane* yang memisahkan himpunan data dengan kelas *attrition* dan *not-attrition*. Apabila suatu data uji masuk ke dalam batas kelas *attrition*, maka SVM mengklasifikasi data tersebut ke dalam kelas *attrition* dan begitu juga sebaliknya. Ilustrasi tentang SVM dapat dilihat pada Gambar 1.



GAMBAR 1.
ILUSTRASI SVM.

C. Imbalanced Dataset

Dalam klasifikasi data, suatu data set dapat disebut sebagai *imbalanced dataset* ketika data set tersebut memiliki distribusi kelas klasifikasi yang tidak seimbang. Terdapat suatu kelas klasifikasi yang memiliki frekuensi yang sangat rendah atau sangat tinggi daripada kelas yang lainnya. *Imbalanced dataset* ini mempunyai pengaruh terhadap proses klasifikasi data di mana algoritma klasifikasi cenderung untuk mengklasifikasikan data ke dalam kelas yang memiliki frekuensi paling tinggi sehingga algoritma klasifikasi tersebut tidak mengidentifikasi kelas kelas yang memiliki frekuensi yang sangat rendah.

Dalam kasus di mana kelas yang paling penting untuk diidentifikasi malah memiliki frekuensi yang sangat rendah, *imbalanced dataset* dapat mengakibatkan algoritma mengalami misklasifikasi sehingga tujuan utama klasifikasi tidak dapat dicapai dengan baik. Dampak dari *imbalanced dataset* ini dapat diminimalisir dengan beberapa cara seperti *data resampling*, memilih metrik evaluasi yang sesuai, memvalidasi hasil dengan melakukan beberapa kali klasifikasi dengan komposisi data set latih dan data set uji yang berbeda, dan lain lain [8].

D. Geometric-mean

Geometric-mean adalah salah satu metrik evaluasi yang menghitung hasil perkalian akar sensitivitas dari semua kelas klasifikasi. *Geometric-mean* dapat menunjukkan keseimbangan antara hasil prediksi dari semua kelas klasifikasi, hasil ini penting untuk data set yang *imbalanced* [9]. Jika model tidak dapat mengidentifikasi salah satu kelas klasifikasi dengan baik maka nilai *geometric-mean* menjadi rendah. Rumus dari *geometric-mean* dapat dilihat di bawah ini.

$$True\ Positive\ Rate = \frac{True\ Positive}{True\ Positive + False\ Negative} \quad (1)$$

$$True\ Negative\ Rate = \frac{True\ Negative}{True\ Negative + False\ Positive} \quad (2)$$

$$Geometric - mean = \sqrt{True\ Positive\ Rate} * \sqrt{True\ Negative\ Rate} \quad (3)$$

E. Penelitian Terkait

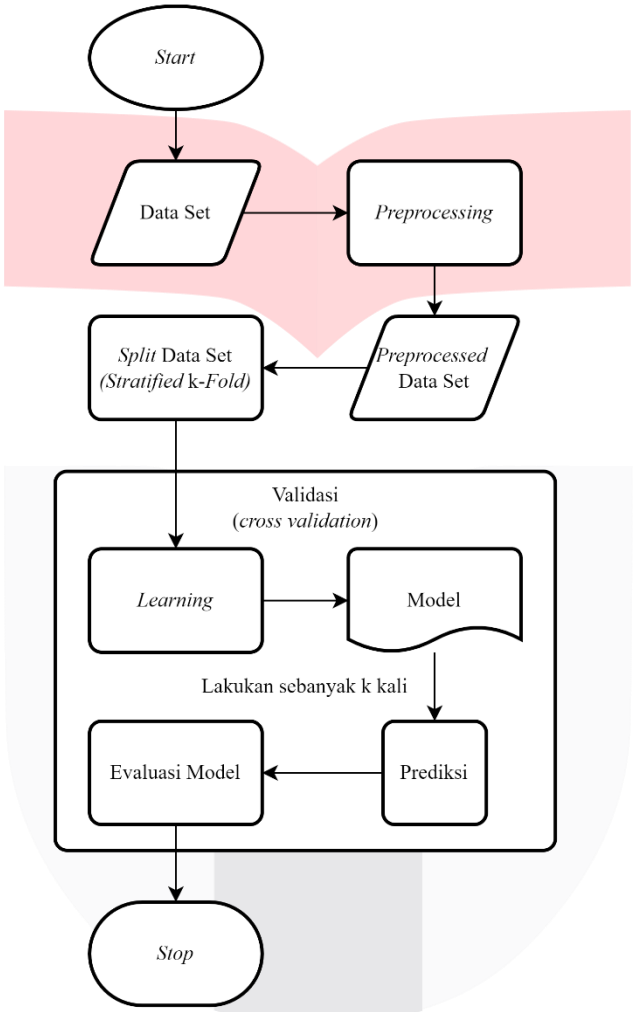
Yedida dkk. [10] memprediksi *employee attrition* dengan menggunakan algoritma *k-nearest neighbors* (kNN). Penelitian tersebut memanfaatkan data performansi karyawan, rata-rata jam kerja, dan berapa lama karyawan tersebut bekerja untuk perusahaan sebagai *feature* untuk memprediksi *attrition*. Hasil yang didapat dari penelitian tersebut adalah algoritma kNN menghasilkan metrik evaluasi yang paling baik diantara algoritma lainnya seperti *Naïve Bayes*, *logistic regression*, dan *multi layer perceptron* dengan nilai akurasi sebesar 0.94 dan *F1-score* sebesar 0.88.

Alao dkk. [11] melakukan analisis untuk mengidentifikasi atribut utama yang berpengaruh terhadap prediksi *employee attrition*. Terdapat 6 atribut yang dianalisis, yaitu jenis kelamin, negara asal karyawan, lama bekerja untuk perusahaan, tingkat jabatan, gaji, dan alasan keluar dari perusahaan. Berdasarkan *rules* yang dihasilkan oleh algoritma *decision tree*, atribut gaji digunakan 100% untuk penentuan klasifikasi, disusul dengan lama bekerja sebesar 49%, dan atribut-atribut lainnya kurang dari sama dengan 16%. Dapat disimpulkan bahwa gaji dan lama bekerja adalah atribut utama dalam memprediksi *employee attrition*. Seorang karyawan yang sudah lama bekerja untuk perusahaan lebih berpotensi keluar apabila tidak ada kenaikan gaji.

III. METODE

A. *Flowchart* Sistem

Sistem dibangun sesuai dengan *flowchart* pada Gambar 2.



GAMBAR 2.
FLOWCHART SISTEM.

A. Data set

Penelitian ini menggunakan data set IBM HR *Analytics Employee Attrition & Performance* [12]. Data set ini terdiri dari 35 kolom dan 1470 baris. Data set ini berisi informasi tentang pendapatan karyawan, tingkat kepuasan karyawan, senioritas, dan beberapa demografi karyawan. Data set ini juga memiliki sebuah kolom yang bernama *attrition* yang menyatakan apakah karyawan yang bersangkutan itu keluar dari perusahaan atau tidak. Kolom *attrition* adalah target yang diprediksi nilainya antara *true* atau *false*. Data set ini termasuk data set yang *imbalanced* dikarenakan 1233 data masuk ke dalam kelas not-*attrition* sedangkan hanya ada 237 data yang masuk ke dalam kelas *attrition*.

B. Preprocessing

Pada tahap *preprocessing*, dilakukan penghapusan fitur-fitur yang hanya memiliki 1 nilai atau tidak relevan. Pada tahap ini juga dilakukan *scaling* dan *encoding* pada data set. *Scaling* data set diperlukan karena fitur-fitur yang bertipe numerik

memiliki rentang nilai yang berbeda. Hal ini dapat menyebabkan suatu fitur dengan rentang nilai yang panjang mendominasi fitur dengan rentang nilai yang pendek. *Encoding* diperlukan karena terdapat beberapa kolom bertipe kategorial dan bukan numerik yang harus diubah menjadi numerik.

C. Split Data set

Data set dibagi dengan menggunakan metode *stratified* k-*fold*. Dengan metode k-*fold,* data set akan dibagi menjadi k bagian sama rata dan acak. Metode *stratified* membagi data set agar setiap bagian memiliki persentase distribusi kelas klasifikasi yang sama dengan data set utama. Metode *stratified* ini digunakan untuk menjaga proporsi dari data yang masuk ke kelas minoritas sehingga meminimalisir *overfitting* pada model *machine learning* [13].

D. Validasi

Tahap validasi ini dilakukan dengan cara *cross validation*. Tahap validasi ini meliputi tahap *learning* dan evaluasi model yang dilakukan sebanyak k kali, sesuai dengan jumlah bagian data set pada tahap *split* data set. Dari data set yang sudah dibagi menjadi k bagian ini, secara bergantian $k - 1$ bagian dijadikan data latih dan 1 bagian lainnya dijadikan data uji. *Cross validation* ini diterapkan untuk mencegah *overfitting* [14].

E. Learning

Pada tahap *learning*, dibangun model *machine learning* menggunakan data set latih dan algoritma yang sudah ditentukan, yaitu SVM dan kNN. Model dibangun dengan bantuan *library* Scikit-Learn [15]. Untuk setiap kali *learning*, dilakukan *resampling* pada data set latih. Metode *resampling* yang dilakukan adalah *oversampling*.
Oversampling akan menambahkan data pada beberapa kelas klasifikasi sehingga persentase distribusi kelas klasifikasi dapat seimbang. *Oversampling* ini digunakan agar kelas klasifikasi lain tidak mendominasi dalam proses klasifikasi sehingga model *machine learning* cenderung untuk mengklasifikasikan data ke dalam kelas mayoritas [16].

F. Evaluasi Model

Pada tahap evaluasi model, model diuji dengan data set uji. Hasil mentah dari pengujian ini adalah sebuah *confusion matrix* yang diolah menjadi metrik-metrik yang sudah ditentukan dan kemudian dilakukan perbandingan performa antara dua model yang sudah dibangun. Masing-masing model diukur menggunakan acuan nilai *accuracy*, *F1-score,* dan *geometric-mean*. Masing-masing metrik ini mengevaluasi aspek prediksi yang berbeda, rumus untuk menghitungnya dari hasil *confusion matrix* dapat dilihat pada Tabel 1.

TABEL 1.
DETAIL METRIK EVALUASI.

Nama Metrik	Rumus	Interval nilai	Target Nilai
Accuracy	$\frac{TP + TN}{TP + TN + FP + FN}$	[0, 1]	~1
F1-score	$\frac{TP}{TP + \frac{1}{2}(FP + FN)}$	[0, 1]	~1
Geometric-mean	$\sqrt{(TP / (TP + FN)} * \sqrt{TN / (TN + FP)}$	[0, 1]	~1

Keterangan :
$TN = True\ Negative$
$FP = False\ Negative$
$FN = False\ Negative$
$TP = True\ Positive$

Accuracy menunjukkan berapa persen prediksi tepat untuk semua kelas klasifikasi. *F1-score* menunjukkan *precision* dan *recall* dari model. *Precision* menjawab pertanyaan "Berapa persen karyawan yang benar keluar dari perusahaan dari keseluruhan karyawan yang diprediksi keluar dari perusahaan". *Recall* adalah seberapa akurat model dalam mengklasifikasi kelas positif. *Recall* memiliki nama lain *true positive rate* dan *sensitivity*. *Recall* menjawab pertanyaan "Berapa persen karyawan yang diprediksi keluar dari perusahaan dibandingkan keseluruhan karyawan yang benar keluar dari perusahaan". *Geometric-mean* menunjukkan keseimbangan antara prediksi di setiap kelas.

IV. HASIL DAN PEMBAHASAN

A. Hasil Pengujian

Dilakukan beberapa kali percobaan untuk menentukan nilai-nilai parameter terbaik untuk model dan pembagian data set. Hasil percobaan tersebut dirangkum dan dapat dilihat pada Tabel 2.

TABEL 2.
PERCOBAAN MENCARI NILAI-NILAI PARAMETER TERBAIK.

Nama Parameter	**Nilai yang Dicoba**	**Hasil**
Parameter *k* pada metode *stratified k-fold* yang membagi *data set*	2, 3, 4, 5, 6, 7, 8, 9	Nilai *k* terbaik adalah 5
Paramater *kernel* pada algoritma SVM yang membentuk *hyperplane* pemisah kelas	*Linear, radial basis function, sigmoid, polynomial*	Nilai *kernel* terbaik adalah *linear*
Parameter *k* pada algoritma kNN yang menentukan jumlah tetangga terdekat yang dipilih	3, 5, 7, 9	Nilai *k* terbaik adalah 5
Parameter *p* pada algoritma kNN yang menentukan derajat *minkowski distance*	1 (*manhattan distance*), 2 (*euclidean distance*)	Nilai *p* terbaik adalah 1 (*manhattan distance*)

Nilai-nilai parameter yang terbaik kemudian digunakan dalam pengujian yang sebenarnya. Berikut adalah hasil pengujian dari model yang dibangun menggunakan algoritma SVM (yang selanjutnya disebut dengan "Model SVM") dan model yang dibangun menggunakan algoritma kNN (yang selanjutnya disebut dengan "Model kNN") dengan penanganan *imbalanced dataset*.

B. Model SVM

Model SVM mendapatkan nilai *accuracy* terendah di 0.84 dan tertinggi di 0.87 dengan rata-rata 0.86. Selanjutnya, Model SVM mendapatkan nilai *F1-score* terendah di 0.56 dan tertinggi di 0.64 dengan rata-rata 0.59. Model SVM juga mendapatkan nilai *geometric-mean* terendah di 0.73 dan tertinggi di 0.78 dengan rata-rata 0.75. Hasil pengujian secara lengkap dapat dilihat pada Tabel 3.

TABEL 3.
HASIL PENGUJIAN MODEL SVM.

Fold	Confusion Matrix				*Accuracy*	*F1-score*	*Geometric-mean*
	True Negative	*False Positive*	*False Negative*	*True Positive*			
1	220	26	19	29	0.84	0.56	0.73
2	226	20	16	32	0.87	0.64	0.78
3	226	21	19	28	0.86	0.58	0.73
4	222	25	17	30	0.85	0.58	0.75
5	226	21	19	28	0.86	0.58	0.73
Mean					0.86	0.59	0.75

C. Model kNN

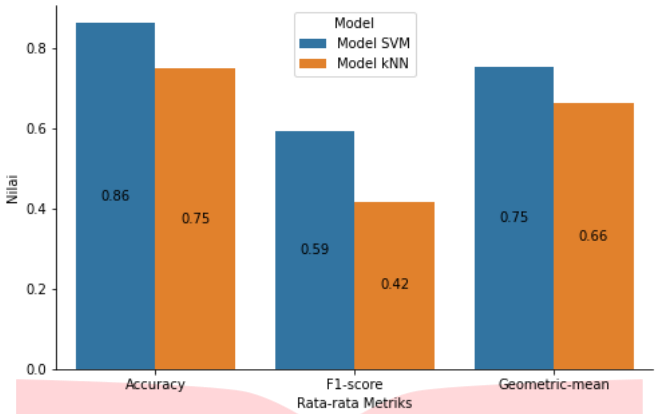
Model kNN mendapatkan nilai *accuracy* terendah di 0.72 dan tertinggi di 0.76 dengan rata-rata 0.74. Selanjutnya, Model kNN mendapatkan nilai *F1-score* terendah di 0.37 dan tertinggi di 0.43 dengan rata-rata 0.41. Model kNN juga mendapatkan nilai *geometric-mean* terendah di 0.61 dan tertinggi di 0.67 dengan rata-rata 0.66. Hasil pengujian secara lengkap dapat dilihat pada Tabel 4.

TABEL 4.
HASIL PENGUJIAN MODEL KNN.

Fold	**Confusion Matrix**				***Accuracy***	***F1-score***	***Geometric-mean***
	True Negative	***False Positive***	***False Negative***	***True Positive***			
1	197	49	21	27	0.76	0.43	0.67
2	193	53	20	28	0.75	0.43	0.67
3	192	55	20	27	0.74	0.41	0.66
4	185	62	18	29	0.72	0.42	0.67
5	198	49	25	22	0.74	0.37	0.61
Mean					0.74	0.41	0.66

D. Analisis Hasil Pengujian

1. Perbandingan Performansi Model SVM dengan Model kNN



GAMBAR 3.
PERBANDINGAN RATA-RATA METRIK MODEL SVM DAN MODEL KNN.

Berdasarkan hasil pengujian dan Gambar 3, dapat dilihat bahwa Model SVM selalu memiliki *accuracy* yang lebih tinggi daripada Model kNN. Semakin tinggi nilai *accuracy* maka semakin akurat sebuah model dalam melakukan prediksi. Berdasarkan hal tersebut, didapatkan bahwa Model SVM memprediksi lebih akurat daripada Model kNN.

Selanjutnya, dapat dilihat bahwa Model SVM selalu memiliki *F1-score* yang lebih tinggi daripada Model kNN. Semakin tinggi nilai *F1-score* maka *precision* dan *recall* model semakin baik. Berdasarkan hal tersebut, didapatkan bahwa Model SVM memiliki *precision* dan *recall* yang lebih baik daripada Model kNN.

Kemudian, dapat dilihat bahwa Model SVM selalu memiliki *geometric-mean* yang lebih tinggi daripada Model kNN. Nilai *geometric-mean* menunjukkan keseimbangan antara prediksi di kelas *attrition* dengan prediksi di kelas *not-attrition*. Artinya, semakin tinggi nilai *geometric-mean* maka semakin seimbang model tersebut dalam mengidentifikasi data yang seharusnya masuk ke kelas *attrition* dan mana yang seharusnya masuk ke kelas *not-attrition*. Berdasarkan hal tersebut, didapatkan bahwa Model SVM lebih seimbang dalam memprediksikan data ke dalam kelas *attrition* dan kelas *not-attrition* daripada Model kNN.

2. Pengaruh Penanganan Imbalanced Dataset pada Model SVM

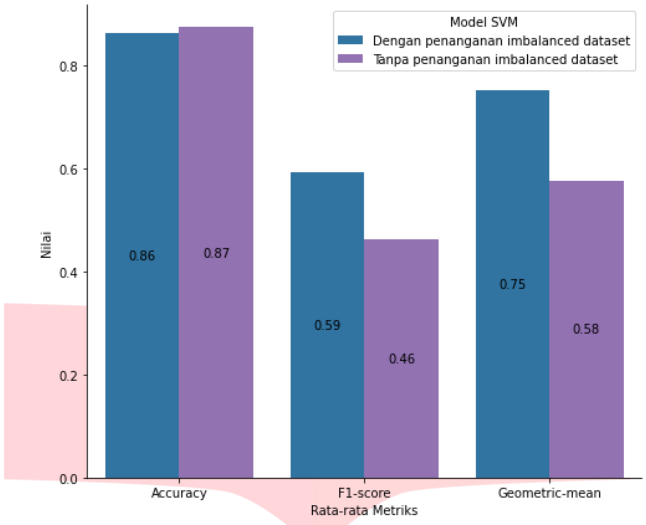
Model SVM tanpa penanganan *imbalanced dataset* ini tidak menggunakan metode *stratified* pada tahap *split dataset* dan tidak melakukan *resampling* pada tahap *learning model*. Dikarenakan pada model ini tidak dilakukan penanganan *imbalanced dataset*, metrik utama untuk mengevaluasi model dengan *imbanced dataset* ini adalah *F1-score* dan *geometric-mean*. Nilai *accuracy* ditampilkan untuk sekedar informasi saja. Model ini mendapatkan nilai *accuracy* terendah di 0.84 dan tertinggi di 0.90 dengan rata-rata 0.87. Selanjutnya, nilai *F1-score* terendah di 0.36 dan tertinggi di 0.57 dengan rata-rata 0.46. Nilai *geometric-mean* terendah di 0.50 dan tertinggi di 0.65 dengan rata-rata 0.57. Hasil pengujian secara lengkap dapat dilihat pada Tabel 5.

TABEL 5.
HASIL PENGUJIAN MODEL SVM TANPA PENANGANAN *IMBALANCED DATASET*.

Fold	**Confusion Matrix**				*Accuracy*	*F1-score*	*Geometric-mean*
	True Negative	*False Positive*	*False Negative*	*True Positive*			
1	245	3	26	20	0.90	0.57	0.65
2	245	2	31	16	0.88	0.49	0.58
3	242	12	24	16	0.87	0.47	0.61
4	231	5	42	16	0.84	0.40	0.51
5	241	7	34	12	0.86	0.36	0.50
Mean					0.87	0.46	0.57

Jika diperhatikan, nilai rata-rata *F1-score* dan *geometric-mean* mengalami penurunan yang cukup signifikan. Nilai rata-rata *F1-score* turun sebanyak 0.13, dari 0.59 ke 0.46. Nilai rata-rata *geometric-mean* turun sebanyak 0.18, dari 0.75 ke 0.57. Sedangkan, nilai rata-rata *accuracy* mengalami kenaikan yang tidak signifikan. Nilai rata-rata *accuracy* naik sebanyak 0.1, dari 0.86

ke 0.87. Hal ini disebabkan karena model lebih condong untuk mengklasifikasi ke kelas mayoritas *not-attrition*. Hal tersebut dapat diamati dari hasil *confusion matrix* yang memiliki nilai *true negative* dan *false negative* lebih banyak daripada model sebelumnya. Dilihat dari nilai *true positive* yang lebih sedikit, kemampuan model dalam mengenali kelas minoritas *attrition* menurun cukup signifikan. Grafik perbandingan dapat dilihat pada Gambar 4.



GAMBAR 4.
PERBANDINGAN RATA-RATA METRIK MODEL SVM DENGAN PENANGANAN DAN TANPA PENANGANAN *IMBALANCED DATASET*.

V. KESIMPULAN

Berdasarkan pengujian dan analisis yang dilakukan, maka dapat disimpulkan pada kasus ini bahwa:

1. Dilihat dari rata-rata nilai *accuracy*, Model SVM lebih akurat dalam memprediksi daripada Model kNN;
2. Dilihat dari rata-rata nilai *F1-score*, Model SVM memiliki *precision* dan *recall* yang lebih baik daripada Model kNN;
3. Dilihat dari rata-rata nilai *geometric mean*, Model SVM lebih seimbang dalam memprediksikan data ke dalam kelas *attrition* dan kelas *not-attrition* daripada Model kNN;
4. Kemampuan Model SVM dalam mengenali kelas minoritas *attrition* akan menurun apabila tidak dilakukan penanganan *imbalanced dataset*.

Dengan dibangunnya model *machine learning* ini diharapkan dapat mempermudah perusahaan dalam menentukan karyawan-karyawan yang memerlukan tindakan preventif dan persuasif dari *employee attrition*. Penulis juga mengharapkan adanya penelitian lanjutan yang menggunakan kombinasi model *machine learning* dengan pengaturan parameter yang lebih baik lagi sehingga hasil metrik yang didapat bisa lebih bagus lagi.

REFERENSI

[1] Tracey, J. B., & Hinkin, T. R. (2006). The costs of employee turnover: When the devil is in the details.
[2] Marsden, T. (2016). What is the true cost of attrition?. Strategic HR Review.
[3] Kotsiantis, S. B., Zaharakis, I. D., & Pintelas, P. E. (2006). Machine learning: a review of classification and combining techniques. Artificial Intelligence Review, 26(3), 159-190.
[4] Althnian, A., AlSaeed, D., Al-Baity, H., Samha, A., Dris, A. B., Alzakari, N., ... & Kurdi, H. (2021). Impact of dataset size on classification performance: an empirical evaluation in the medical domain. Applied Sciences, 11(2), 796.
[5] Masese, O. F. (2016). Employee Attrition Management by Engagement.
[6] Pisner, D. A., & Schnyer, D. M. (2020). Support vector machine. In Machine learning (pp. 101-121). Academic Press.
[7] Kotsiantis, S. B., Zaharakis, I., & Pintelas, P. (2007). Supervised machine learning: A review of classification techniques. Emerging artificial intelligence applications in computer engineering, 160(1), 3-24.
[8] Yadav, S., & Bhole, G. P. (2020, December). Handling imbalanced dataset classification in machine learning. In 2020 IEEE Pune Section International Conference (PuneCon) (pp. 38-43). IEEE.
[9] Akosa, J. (2017). Predictive accuracy: A misleading performance measure for highly imbalanced data. In Proceedings of the SAS Global Forum (Vol. 12).
[10] Yedida, R., Reddy, R., Vahi, R., Jana, R., GV, A., & Kulkarni, D. (2018). Employee attrition prediction. arXiv preprint arXiv:1806.10480.
[11] Alao, D. A. B. A., & Adeyemo, A. B. (2013). Analyzing employee attrition using decision

tree algorithms. Computing, Information Systems, Development Informatics and Allied Research Journal, 4(1), 17-28.

[12] IBM. (2017). *IBM HR Analytics Employee Attrition & Performance* (Version 1) [Data set]. Kaggle. Retrieved December 7, 2021, from https://www.kaggle.com/pavansubhasht/ibm-hr-analytics-attrition-dataset

[13] Gu, Y., Wylie, B. K., Boyte, S. P., Picotte, J., Howard, D. M., Smith, K., & Nelson, K. J. (2016). An optimal sample data usage strategy to minimize overfitting and underfitting effects in regression tree models based on remotely-sensed data. Remote sensing, 8(11), 943.

[14] Berrar, Daniel. (2018). Cross-Validation. Encyclopedia of Bioinformatics and Computational Biology.

[15] Pedregosa, et al. (2011) Scikit-Learn Machine Learning in Python. Journal of Machine Learning Research, 12, 2825-2830.

[16] Barandela, R., Valdovinos, R. M., Sánchez, J. S., & Ferri, F. J. (2004, August). The imbalanced training sample problem: Under or over sampling?. In Joint IAPR international workshops on statistical techniques in pattern recognition (SPR) and structural and syntactic pattern recognition (SSPR) (pp. 806-814). Springer, Berlin, Heidelberg.