

Optimasi Parameter *Artificial Neural Network* (ANN) Menggunakan *Particle Swarm Optimization* (PSO) Untuk Pengkategorian Nasabah Bank

Lennox Larwuy

Program Studi Matematika, Fakultas Sains dan Matematika, Universitas Kristen Satya

Email: Lennox.larwuy@uksw.edu

ABSTRAK

Neural Network (ANN) merupakan salah satu metode yang sering digunakan untuk pengenalan pola, klasifikasi, peramalan, dan regresi tergantung dari permasalahan atau data yang dipakai dimana hasil yang diperoleh cukup bagus tetapi sering bermasalah pada penentuan parameter sebagai bobot awal dimana hasil yang diperoleh bisa tidak konvergen untuk itulah dibutuhkan sebuah metode untuk mengoptimalkan parameter ANN guna memperoleh hasil yang lebih baik. PSO dipilih sebagai metode untuk mengoptimalkan parameter ANN (PSO-ANN) dengan terlebih dahulu ditentukan nilai parameter terbaik untuk PSO dimana $w = 0.8$ dan $c_1, c_2 = 1.5$ yang kemudian PSO-ANN dilatih menggunakan dataset nasabah bank guna mengetahui kategori nasabah dengan kredit bermasalah atau tidak. Hasil yang diperoleh ialah dibandingkan dengan ANN tanpa optimasi parameter, dimana tingkat *accuracy* sebesar 82.6%, *precision* sebesar 91.1% dan *recall* sebesar 37.1% hal ini lebih baik jika dibandingkan dengan hasil dari ANN tanpa pengoptimalan parameter dengan nilai *accuracy* sebesar 80.1%, *precision* sebesar 89.5% dan *recall* sebesar 32.4%.

Kata kunci: optimasi, artificial neural network, particle swarm optimization

ABSTRACT

Artificial Neural Network (ANN), also known as *Jaringan Saraf Tiruan*, is one of the methods commonly used for pattern recognition, classification, forecasting, and regression, depending on the problem or data used. While the results obtained are generally good, there are often issues with determining the initial parameters as the initial weights, which can lead to non-convergence of results. This is why a method is needed to optimize the ANN parameters to achieve better outcomes. Particle Swarm Optimization (PSO) was chosen as the method to optimize the ANN parameters (PSO-ANN). The best parameter values for PSO were predefined, with w (inertia weight) set to 0.8 and c_1 and c_2 (acceleration coefficients) set to 1.5. Subsequently, PSO-ANN was trained using a bank customer dataset to determine the categories of customers with credit problems or not. The results were compared with using ANN without parameter optimization. The obtained results showed an accuracy rate of 82.6%, Precision of 91.1%, and Recall of 37.1%. This represents an improvement compared to the results of ANN without parameter optimization, which had an Accuracy rate of 80.1%, Precision of 89.5%, and Recall of 32.4%.

Keywords: optimization, artificial neural network, particle swarm Optimization

1. Pendahuluan

Jaringan Saraf Tiruan atau *Artificial Neural Network* (ANN) adalah sistem yang terdiri dari lapisan-lapisan neuron, termasuk *input*, *output*, dan *hidden layer*. Neuron-neuron ini saling terhubung dan memiliki bobot. ANN adalah alat yang sangat efektif yang telah digunakan dalam berbagai konteks seperti pengenalan pola, klasifikasi, peramalan, dan regresi. Selama proses pembelajaran, hasil dan kesimpulan dari ANN didasarkan pada pengalaman yang diperoleh selama pelatihan. Pelatihan ANN dimulai dengan menginisialisasi bobot awal, dan data yang

sudah diketahui hasilnya kemudian dimasukkan ke dalam unit pada lapisan input.

Sejumlah algoritma tradisional telah diajukan dan diperbaiki untuk melatih ANN. Beberapa penelitian juga mencoba menggunakan algoritma evolusi dan terinspirasi oleh prinsip alam untuk meningkatkan efisiensi pembelajaran [1]. Metode metaheuristik yang digunakan dalam pelatihan jaringan saraf mencakup pencarian lokal, metode berbasis populasi.

Salah satu karya yang sangat bagus di mana para penulis menunjukkan tinjauan literatur yang luas tentang algoritma evolusi yang digunakan untuk

mengembangkan ANN adalah [2]. Pada penelitian lainnya, para peneliti mengusulkan pendekatan yang mengkombinasikan *Ant Colony Optimization* (ACO) untuk menemukan koneksi dalam ANN [3]. Penelitian lain seperti yang diuraikan dalam [4], mengimplementasikan modifikasi PSO yang dipadukan dengan *Simulated Annealing* (SA) untuk menghasilkan kumpulan bobot sinaptik dan ambang ANN. Selain itu, dalam penelitian [5], penulis menggunakan *Evolutionary Programming* untuk mendapatkan struktur dan bobot sinaptik dengan tujuan menyelesaikan masalah klasifikasi dan prediksi. Dalam penelitian [6], para peneliti membandingkan metode *backpropagation* dengan PSO dasar untuk mengatur hanya bobot sinaptik ANN guna menyelesaikan masalah klasifikasi. Terakhir, dalam penelitian [7], pengaturan kumpulan bobot dilakukan dengan menggunakan *Differential Evolution* dan PSO dasar.

Untuk mengatasi masalah optimalisasi bobot jaringan pada ANN maka digunakan *Particle Swarm Optimization* (PSO), karena merupakan salah satu teknik AI terbaik untuk optimasi dan perkiraan parameter. Penggunaan optimasi pada bobot jaringan syaraf tiruan akan mempercepat waktu kalkulasi pada proses selanjutnya untuk mencapai keadaan konvergen. Model akan dievaluasi dengan mengaplikasikannya pada dataset nasabah bank dimana akan dicari nasabah mana yang tergolong yang layak diberi kredit dan mana yang tidak. Penelitian ini akan fokus pada penerapan *particle swarm optimization* sebagai metode pelatihan bobot ANN. Hasil yang diharapkan ialah ANN yang telah dioptimasi oleh PSO mampu untuk mengkategorikan data berdasarkan tingkat akurasi terbaik.

Tulisan ini disajikan dengan susunan sebagai berikut. Pada bagian dua dibahas metode pemecahan masalah. Selanjutnya, pada bagian tiga disajikan bagian pembahasan. Makalah ini ditutup dengan bagian kesimpulan dan saran yang diberikan pada bagian empat.

2. Metode

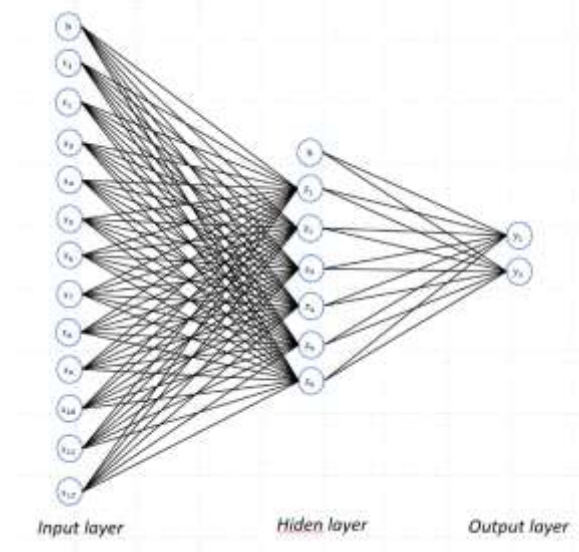
2.1 Data

Data yang digunakan adalah data sekunder berisi informasi status peminjaman bank yang diperoleh dari [8] dan telah digunakan dalam penentuan prediksi penerimaan tawaran pinjaman dari bank menggunakan *Support Vector Machines* [9], dataset ini berisi informasi tentang status peminjaman atau pinjaman yang diberikan oleh sebuah bank kepada nasabahnya dengan jumlah 5000 nasabah dengan 12

variabel sebagai pelengkap data diri atau atribut, yang selanjutnya akan dibagi guna keperluan pengolahan dengan 70% untuk *training* dan 30% untuk pengujian.

2.2 Arsitektur Artificial Neural Network (ANN)

Pada penelitian ini jaringan syaraf tiruan yang digunakan terdiri atas 1 buah *input layer*, 1 buah *hidden layer*, dan 1 buah *output layer*. Arsitektur ANN yang digunakan ditunjukkan pada Gambar 1. Dimana *input layer* terdiri atas 12 buah neuron *input* (sesuai dengan atribut yang digunakan pada data), enam buah neuron pada *hidden layer*, dan dua buah neuron *output layer*. Serta terdapat bias pada *input layer* dan *hidden layer*. Arsitektur ANN adalah tetap, karena pada penelitian ini tidak berfokus pada pengaruh arsitektur jaringan syaraf tiruan terhadap hasil, melainkan pada parameter dari metode optimasi yang digunakan yaitu PSO.



Gambar 1. Arsitektur ANN

2.3 Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) adalah salah satu dari algoritma metaheuristik yang digunakan dalam berbagai bidang penelitian. PSO terinspirasi dari perilaku sosial burung-burung. Dalam algoritma ini, partikel-partikel diberi penyesuaian awal secara acak sebagai solusi awal. Kemudian, posisi dan kecepatan partikel-partikel diperbarui berdasarkan pengalaman pribadi dan global seluruh kelompok. Algoritma ini berbasis populasi dan diulang selama beberapa episode untuk mencapai solusi optimal [10]. Setiap partikel memiliki dua properti, yaitu kecepatan (V) dan posisi (X). Kecepatan dan posisi masing-masing partikel ditentukan oleh Persamaan 1 dan 2, secara berturut-turut.

$$v_i^{t+1} = wv_i^t + r_1c_1(Pbest_i^t - x_i^t) + r_2c_2(Gbest^t - x_i^t) \quad (1)$$

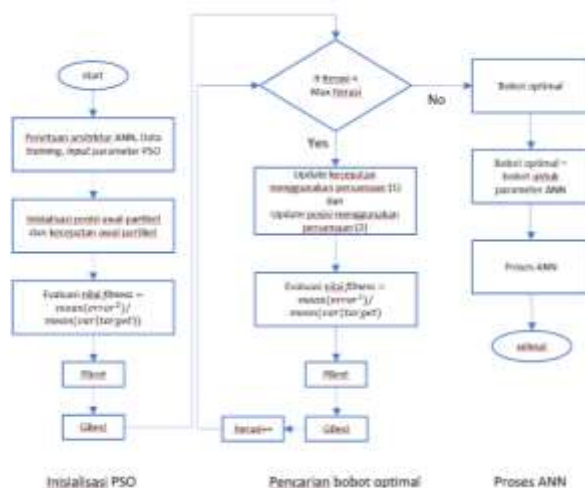
$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (2)$$

Dimana v_i^t adalah kecepatan partikel ke i pada iterasi ke t , dan x_i^t adalah posisi partikel ke i pada saat iterasi ke t , sedangkan untuk w adalah bobot inersia yang mengontrol perubahan kecepatan perpindahan partikel, c_1, c_2 adalah konstanta kecepatan bernilai positif, r_1, r_2 adalah variabel acak berdistribusi *uniform* pada selang antara 0 sampai 1, sedangkan Pbest dan Gbest secara berturut-turut adalah pengalaman pribadi terbaik dari partikel dan pengalaman terbaik secara keseluruhan (kawanan)

2.4 Optimasi ANN menggunakan PSO (PSO-ANN)

Penelitian ini bertujuan untuk memperoleh hasil paling akurat dengan menggabungkan algoritma PSO dan ANN sederhana, dimana PSO digunakan untuk memperoleh bobot optimal untuk parameter ANN (PSO-ANN) dengan melakukan proses analisis terhadap parameter PSO yang kemudian hasil analisis parameter terbaik yang didapatkan akan digunakan sebagai pembobotan pada prosedur ANN yang kemudian akan digunakan untuk mengkategorikan nasabah yang baik dan tidak baik berdasarkan data yang digunakan untuk pelatihan metode. Hasil akhir yang didapatkan kemudian dibandingkan dengan ANN tanpa optimasi parameter

Diagram alir berisi rincian tahapan pengerjaan pengoptimalan ANN menggunakan PSO dapat dilihat pada Gambar 2



Gambar 2. Diagram alir proses pembobotan dengan PSO pada ANN

Gambar 2 merupakan diagram alir dari proses pembobotan dengan PSO pada ANN. Pada Gambar 2 tersebut terdapat inputan berupa nilai dari parameter PSO serta jumlah neuron yang dibutuhkan pada *hidden layer*. Dalam proses PSO akan dibangkitkan posisi awal dan kecepatan awal PSO sesuai dengan ukuran partikel yang ingin dibuat serta jumlah bobot jaringan yang dibutuhkan. Pada hasil akhir (kondisi selesai PSO) bobot terbaik yang dihasilkan akan digunakan sebagai bobot pada jaringan syaraf tiruan atau ANN.

2.5 Evaluasi

Pada beberapa penelitian sebelumnya evaluasi hasil percobaan dilakukan dengan menghitung nilai *error* yang diperoleh dari evaluasi nilai bobot. Pada penelitian ini untuk mengevaluasi hasil yang didapatkan, digunakan perhitungan *Accuracy*, *Precision*, *Recall* setelah diperoleh *Confusion matrix* [9]. Matriks ini membantu dalam mengukur sejauh mana model dapat memprediksi dengan benar kelas-kelas target dari dataset yang diberikan. Dimana, *True Positives* (TP) adalah jumlah kasus di mana model benar-benar memprediksi positif (kelas yang diminati) dengan benar, *True Negatives* (TN) adalah jumlah kasus di mana model benar-benar memprediksi negatif (kelas yang bukan target) dengan benar, *False Positives* (FP) adalah jumlah kasus di mana model salah memprediksi positif saat seharusnya negatif, *False Negatives* (FN) adalah jumlah kasus di mana model salah memprediksi negatif saat seharusnya positif.

Predicted Value	True Positive (TP)	False Positive (FP)
	False Negative (FN)	True Negative (TN)
		Actual Value

Gambar 3. Confusion matrix

Untuk menghitung *Accuracy*, *Precision*, *Recall* digunakan rumus berikut:

$$Accuracy = (TP + TN)/(TP + TN + FP + FN) \quad (3)$$

$$Precision = (TP)/(TP + FP) \quad (4)$$

$$Recall = (TP)/(FP + FN) \quad (5)$$

Accuracy digunakan untuk menghitung presentase tebakan yang benar dari jumlah total prediksi, *Precision* digunakan untuk menghitung seberapa

besar presentasi model bekerja dengan baik dalam melakukan prediksi, *Recall* digunakan untuk menghitung presentase sejauh mana model mampu mendeteksi semua kasus positif yang sebenarnya.

3. Hasil dan Pembahasan

Proses pengujian dilakukan terhadap 70% data yang dimiliki. Akan dilakukan proses analisis terhadap hasil akurasi yang dibutuhkan untuk proses *train* pada ANN dengan optimasi menggunakan *Particle Swarm Optimization* (PSO) yang nantinya akan dibandingkan dengan hasil ANN tanpa optimasi parameter. Proses pencarian parameter terbaik dilakukan dengan menganalisis hasil dari masing – masing parameter yang dibangkitkan dan dipilih satu buah nilai yang paling tepat pada masing – masing partikel. Parameter yang akan dianalisis pada penelitian ini adalah nilai bobot inersia (w) dengan rentang 0.2 sampai 1, nilai konstanta kecepatan (c_1, c_2) antara 1 sampai 2.5 Perubahan pada nilai w, c_1 , dan c_2 bertujuan untuk mendapatkan nilai parameter terbaik dari PSO yang akan digunakan untuk proses optimasi pembobotan pada ANN.

3.1 Pengujian Parameter PSO Untuk Optimasi ANN

Prosedur dimulai dengan inisialisasi awal untuk parameter yang akan digunakan pada PSO, dimana jumlah partikel yang digunakan ialah 100 partikel dengan kecepatan awal setiap partikel diperoleh secara acak antara 0 sampai 1 serta posisi awal partikel juga diperoleh secara acak, posisi dari masing-masing partikel ini lah yang digunakan sebagai posisi terbaik partikel (P_{best}) setelah nilai bobot partikel di evaluasi, sedangkan untuk posisi terbaik keseluruhan (G_{best}) diperoleh dengan cara membandingkan setiap P_{best} yang ada, bobot terbaik dari hasil perbandingan inilah yang dijadikan sebagai G_{best} dimana nilai G_{best} inilah yang akan dijadikan bobot optimal untuk parameter ANN setelah dievaluasi nilai bobotnya di setiap iterasi yang kemudian prosedur ini akan terus diulang hingga maksimal iterasi yang telah ditentukan dengan kecepatan dan posisi partikel terus diperbaharu menggunakan persamaan (1) dan (2), maksimal iterasi yang kami gunakan ialah 200 dengan pertimbangan jika terlalu sedikit maka bobot optimal yang di inginkan tidak akan diperoleh jika maksimal iterasi terlalu besar maka akan memperlabat waktu komputasi sedangkan hasil bobot optimal telah didapatkan, untuk penentuan maksimal iterasi ini baiknya dilakukan percobaan lebih lanjut untuk memperoleh maksimum itarsi yang diinginkan tetapi

pada penelitian ini tidak kami lakukan karena banyak melibatkan variable acak dan bisa saja berubah tergantung data atau perlakuan yang dipakai.

Berikut ini adalah hasil yang diperoleh untuk pengujian parameter w dengan nilai konstanta kecepatan (c_1, c_2) yang dipakai ialah 1

Tabel 1. Hasil pengujian parameter w

w	c_1, c_2	Accuracy	Time(seconds)
0.2	1	80.34%	135.489
0.4	1	81.81%	127.761
0.6	1	81.65%	130.543
0.8	1	82.68%	125.571
1	1	81.18%	120.832

Berdasarkan hasil pengujian table 1 diperoleh parameter terbaik untuk w ialah 0.8 dengan akurasi 96.47% dari hasil table 1 dapat dilihat juga bahwa semakin besar nilai w maka semakin besar pula *accuracy* yang diperoleh, walaupun untuk $w = 1$ lebih rendah dibandingkan $w = 0.8$ hal ini bisa saja terjadi selisih nilai yang tidak terpaut jauh, hal ini berbanding terbalik dengan waktu yang dibutuhkan dimana semakin besar nilai w maka semakin sedikit waktu komputasi yang dibutuhkan untuk menjalankan 200 iterasi pencarian

Berikutnya dilakukan pengujian parameter konstanta kecepatan (c_1, c_2) dengan menggunakan $w = 0.8$

Tabel 2. Hasil pengujian parameter c_1, c_2

w	c_1, c_2	Accuracy	Time(seconds)
0.8	1	81.47%	125.571
0.8	1.5	82.68%	122.153
0.8	2	82.25%	119.427
0.8	2.5	81.45%	118.036

Dari hasil pengujian pada table 2 diperoleh nilai terbaik untuk dijadikan konstanta kecepatan ialah 1.5 dengan akurasi 82.68 % dan waktu 122.153

3.2 Evaluasi Optimasi Parameter ANN Menggunakan PSO

Berdasarkan hasil pengujian pengaruh parameter w dan c_1, c_2 yang telah dilakukan dimana hasil yang

diperoleh akan dijadikan sebagai nilai parameter untuk prosedur PSO yang kemudian akan dilibatkan dalam proses optimasi parameter ANN untuk menentukan nasabah dengan pengajuan kredit lancar atau tidak lancar. nilai parameter PSO yang dipakai ialah $w = 0.8$ dan $c_1, c_2 = 1.5$, jumlah iterasi = 200, jumlah partikel = 100, jumlah neuron pada *input layer* = 12, *hidden layer* = 6 dan *output layer* = 2 dengan jumlah data yang dipakai untuk *training* sebesar 70% atau 3500 nasabah. Hasil akurasi PSO-ANN sebesar 82.68 % dengan jumlah *True Positive* (TP) sebesar 2353, *True Negative* (FN) sebesar 543, *False Positive* (FP) sebesar 220, *False Negative* (FN) sebesar 386

Predicted Value	TP = 2253	FP = 220
	FN = 386	TN = 641
	Actual Value	

Gambar 4. Hasil *Confusion matrix*

Berdasarkan hasil diatas diperoleh nilai untuk *Accuracy, Precision, Recall* untuk hasil *training* PSO-ANN beserta hasil perbandingan dengan ANN tanpa optimasi dapat dilihat pada tabel berikut

Tabel 3. Hasil Perbandingan *Accuracy, Specificity, Recall* untuk ANN dan PSO-ANN

Value	ANN	PSO-ANN
<i>Accuracy</i>	80.1%	82.6%
<i>Precision</i>	89.5%	91.1%
<i>Recall</i>	32.4%	37.1%

4 Kesimpulan dan Saran

4.1 Kesimpulan

Berdasarkan penelitian yang dilakukan dapat disimpulkan bahwa untuk pengujian parameter terbaik yang dapat digunakan pada *Particle Swarm Optimization* (PSO) diperoleh hasil $w = 0.8$ dan $c_1, c_2 = 1.5$ dan dengan menggunakan parameter ini PSO dapat digabungkan dengan *Artificial Neural Network* (ANN) untuk meningkatkan akurasi ANN dengan menjadikan hasil PSO sebagai bobot awal pada ANN. Gabungan antara PSO dan ANN ini terbukti dapat diterapkan dalam kasus penentuan nasabah bank dengan kredit bermasalah dan tidak bermasalah

dimana hasil yang diperoleh menunjukkan angka yang cukup baik yakni tingkat *Accuracy* sebesar 82.6%, *Precision* sebesar 91.1% dan *Recall* sebesar 37.1% hal ini lebih baik jika dibandingkan dengan hasil dari ANN tanpa pengoptimalan parameter dengan nilai *Accuracy* sebesar 80.1%, *Precision* sebesar 89.5% dan *Recall* sebesar 32.4% .

4.2 Saran

Pada penelitian ini hanya berfokus pada pemilihan parameter c_1, c_2 yang dianggap sama sehingga dibutuhkan percobaan lain dengan melibatkan nilai yang berbeda, kemudian untuk pemilihan *Hidden layer* diharapkan agar menggunakan metode yang memungkinkan untuk jumlahnya menyesuaikan dengan kebutuhan serta data yang digunakan.

Daftar Pustaka

- [1] X. Yao, "Evolving artificial neural networks," *Proc. IEEE*, vol. 87, no. 9, pp. 1423–1447, 1999, doi: 10.1109/5.784219.
- [2] J. Yu, L. Xi, and S. Wang, "An improved particle swarm optimization for evolving feedforward artificial neural networks," *Neural Process. Lett.*, vol. 26, no. 3, pp. 217–231, 2007, doi: 10.1007/s11063-007-9053-x.
- [3] R. Bhavya, K. Sivaraj, and L. Elango, "Ant Colony Based Artificial Neural Network for Predicting Spatial and Temporal Variation in Groundwater Quality," *Water*, vol. 15, no. 12, p. 2222, 2023, doi: 10.3390/w15122222.
- [4] Y. Da and G. Xiurun, "An improved PSO-based ANN with simulated annealing technique," *Neurocomputing*, vol. 63, no. SPEC. ISS., pp. 527–533, Jan. 2005, doi: 10.1016/J.NEUCOM.2004.07.002.
- [5] X. Yao and Y. Liu, "A new evolutionary system for evolving artificial neural networks," *IEEE Trans. Neural Networks*, vol. 8, no. 3, pp. 694–713, 1997, doi: 10.1109/72.572107.
- [6] B. A. Garro and R. A. Vázquez, "Designing Artificial Neural Networks Using Particle Swarm Optimization Algorithms," *Comput. Intell. Neurosci.*, vol. 2015, no. June, 2015, doi: 10.1155/2015/369298.
- [7] B. A. Garro, H. Sossa, and R. A. Vázquez, "Design of artificial neural networks using

- differential evolution algorithm,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6444 LNCS, no. PART 2, pp. 201–208, 2010.
- [8] “Bank_Personal_Loan_Modelling | Kaggle.” <https://www.kaggle.com/datasets/krantiskalk/e/bank-personal-loan-modelling> (accessed Sep. 25, 2023).
- [9] M. F. AKÇA and O. SEVLİ, “Predicting acceptance of the bank loan offers by using support vector machines,” *Int. Adv. Res. Eng. J.*, vol. 6, no. 2, pp. 142–147, 2022, doi: 10.35860/iarej.1058724.
- [10] J. Kennedy and R. Eberhart, “Particle swarm optimization,” *Proc. ICNN’95 - Int. Conf. Neural Networks*, vol. 4, pp. 1942–1948, doi: 10.1109/ICNN.1995.488968.

Received: September 05, 2023

Revised: Desember 10, 2023

Published: Januari 31, 2024