KLASIFIKASI SITUS HIDROKSILASI PROTEIN PADA PROLIN DAN LISIN MENGGUNAKAN *EXTREME GRADIENT BOOSTING* (XGBOOST)

(Skripsi)

Oleh

INTANIA RAHMADHILLA
1817051025

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2022

KLASIFIKASI SITUS HIDROKSILASI PROTEIN PADA PROLIN DAN LISIN MENGGUNAKAN *EXTREME GRADIENT BOOSTING* (XGBOOST)

Oleh

Intania Rahmadhilla

Skripsi

Sebagai Salah Satu Syarat untuk Memperoleh Gelar

SARJANA ILMU KOMPUTER

Pada

Jurusan Ilmu Komputer

Fakultas Matematika dan Ilmu Pengetahuan Alam

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2022

ABSTRAK

KLASIFIKASI SITUS HIDROKSILASI PROTEIN PADA PROLIN DAN LISIN MENGGUNAKAN EXTREME GRADIENT BOOSTING (XGBOOST)

Oleh

Intania Rahmadhilla

Modifikasi pasca-translasi (PTM) memiliki peran penting dalam berbagai sel dan proses biologis. Sebagian besar protein menjalankan fungsi biologisnya setelah melalui modifikasi pasca-translasi. Hidroksilasi protein merupakan salah satu jenis PTM yang terjadi pada prolin (P) dan lisin (K). Diketahui bahwa sekuens protein mengandung banyak residu P dan K yang tidak dikarakterisasi, pertanyaannya mana yang bisa dihidroksilasi dan mana yang tidak bisa. Untuk mengatasi hal tersebut, metode komputasi sangat disarankan untuk mengidentifikasi situs hidroksilasi karena tidak memakan waktu dan biaya yang mahal. Distribusi kelas yang tidak seimbang menjadi tantangan berat bagi algoritma pembelajaran tradisional untuk meningkatkan hasil akurasi keseluruhan, sehingga penelitian ini menggunakan pendekatan machine learning dengan metode Extreme Gradient Boosting (XGBoost) dan implementasi algoritma SMOTE untuk mengatasi kelas data tidak seimbang. Fitur ekstraksi pada penelitian ini terdiri atas tiga macam fitur, yaitu Pseudo Amino Acid Composition (PseAAC), CTD, dan AAindex. Berdasarkan dataset uji, hasil akurasi mencapai 97.9% dan 99.1% untuk prediksi situs hidroksilasi prolin dan hidroksilasi lisin. Sementara itu, diperoleh sensitivitas sebesar 93.6% dan 95.4%, f-1 score sebesar 92.9% dan 97.6%, serta MCC 95.3% dan 97.1% untuk residu P dan K.

Kata Kunci : PTM, hidroksilasi, data tidak seimbang, fitur ekstraksi, xgboost, SMOTE

ABSTRACT

CLASSIFICATION OF PROTEIN HYDROXYLATION SITES ON PROLINE AND LYSINE USING EXTREME GRADIENT BOOSTING (XGBOOST)

By

Intania Rahmadhilla

Post-translational modification (PTMs) has an important role in various cells and biological processes. Most proteins perform their biological functions after going through post-translational modifications. Protein hydroxylation is a type of PTM that occurs in proline (P) and lysine (K). It is known that protein sequences contain many uncharacterized residues of P and K, the question is which ones can be hydroxylated and which ones can't. To overcome this, computational methods are highly recommended to identify hydroxylation sites because they are not time-consuming and costly. Imbalanced class distribution present a major challenge for traditional learning algorithms to improve overall accuracy results. Therefore in this study, we use an Extreme Gradient Boosting (XGBoost) machine learning approach and the implementation of SMOTE algorithm to overcome imbalanced data classes. The extraction features in this study consist of three kinds of features, namely Pseudo Amino Acid Composition (PseAAC), CTD, and AAindex. Based on the test data set, the accuracy results reached 97.9% and 99.1% for the prediction of proline hydroxylation sites and lysine hydroxylation. Meanwhile, sensitivity was obtained by 93.6% and 95.4%, f-1 scores of 92.9% and 97.6%, and MCC of 95.3% and 97.1% for residues of P and K.

Keywords : PTMs, hydroxylation, imbalanced data, feature extraction, xgboost, SMOTE

Judul Skripsi	: **KLASIFIKASI SITUS HIDROKSILASI PROTEIN PADA PROLIN DAN LISIN MENGGUNAKAN *EXTREME GRADIENT BOOSTING* (XGBOOST)**
Nama Mahasiswa	: Intania Rahmadhilla
Nomor Pokok Mahasiswa	: 1817051025
Jurusan	: Ilmu Komputer
Fakultas	: Matematika dan Ilmu Pengetahuan Alam

MENYETUJUI

1. Komisi Pembimbing

Favorisen R. Lumbanraja, Ph.D.
NIP 19830110 200812 1 002

2. Ketua Jurusan Ilmu Komputer

Didik Kurniawan, S.Si., M.T.
NIP 19800419 200501 1 004

MENGESAHKAN

1. Tim Penguji

Ketua : **Favorisen R. Lumbanraja, Ph.D.** .....

Penguji I
Penguji Pembahas : **M. Reza Faisal, S.T., M.T., Ph.D.** .....

Penguji II
Penguji Pembahas : **Dr. Ir. Kurnia Muludi, M.S.Sc.** .....

2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam

Dr. Eng. Suripto Dwi Yuwono, S.Si., M.T.
NIP 19740705 200003 1 001

Tanggal Lulus Ujian Skripsi : **5 Oktober 2022**

PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama: Intania Rahmadhilla

NPM: 1817051025

Dengan ini menyatakan bahwa skripsi saya yang berjudul "KLASIFIKASI SITUS HIDROKSILASI PROTEIN PADA PROLIN DAN LISIN MENGGUNAKAN EXTREME GRADIENT BOOSTING (XGBOOST)" adalah benar hasil karya sendiri dan bukan orang lain. Seluruh tulisan yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Jika di kemudian hari terbukti skripsi saya adalah hasil penjiplakan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan peraturan yang berlaku.

Bandar Lampung, 9 November 2022

Penulis

Intania Rahmadhilla

1817051025

RIWAYAT HIDUP

Penulis dilahirkan pada tanggal 21 Maret 2000. Penulis merupakan anak kedua dari Bapak Lukman dan Ibu Hermalita. Penulis menempuh Pendidikan Sekolah Pertama (SMP) di SMPN 4 Metro pada 2012-2015. Kemudian melanjutkan Pendidikan Sekolah Menengah Atas (SMA) di SMAN 1 Metro pada tahun 2015-2018.

Penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung pada tahun 2018 melalui jalur Seleksi Nasional Masuk Perguruan Tinggi Negeri (SNMPTN). Selama menjadi mahasiswa, penulis melakukan beberapa kegiatan antara lain.

1. Menjadi anggota Adapter Himpunan Mahasiswa Jurusan Ilmu Komputer pada periode 2018/2019.
2. Menjadi bendahara bidang Media Informasi Himpunan Mahasiswa Jurusan Ilmu Komputer pada periode 2019/2020.
3. Melaksanakan Kerja Praktik di PT PLN (Persero) UP3 Tanjung Karang pada tahun 2021.
4. Melaksanakan Kuliah Kerja Nyata (KKN) pada tahun ajaran 2021/2022 di Desa Sri Basuki, Kecamatan Kalirejo, Kabupaten Lampung Tengah.

MOTTO

"Be patient, indeed the (best) outcome is for the righteous."

(Surah Hud 49)

"Don't grieve, indeed Allah is with us."

(Surah At-Taubah 40)

Your efforts will never betray you. All your efforts will pay off."

(Taeyong Lee)

"Fall down seven times, stand up eight. Someday, we'll all succeed."

(Jeon Wonwoo)

PERSEMBAHAN

Alhamdulillahirobbilalamin

Puji syukur kepada Allah SWT yang telah memberikan rahmat dan hidayah-Nya sehingga dapat menyelesaikan penulisan skripsi ini. Sholawat dan salam saya sanjungkan kepada Nabi Muhammad SAW.

Ku persembahkan karya ini kepada:

Orangtuaku Tercinta

Sebagai tanda terimakasihku kepada orangtuaku tercinta dan tersayang. Terima kasih telah mendidik dan membesarkanku dengan kasih sayang kalian. Terimakasih atas semua pengorbanan, dukungan, dan doa yang tiada henti.

Keluargaku Tercinta

Terima kasih telah memberikan semangat, dukungan, dan doa.

Sahabat dan Teman-teman yang selalu memberikan semangat dan dukungan

Almamater Tercinta, Universitas Lampung

SANWACANA

Puji syukur kehadirat Allah SWT, karena telah memberikan rahmat dan hidayahNya kepada saya sehingga saya dapat menyelesaikan skripsi dengan judul "Klasifikasi Situs Hidroksilasi Protein pada Prolin dan Lisin menggunakan Extreme Gradient Boosting (XGBoost)". Saya berharap skripsi ini dapat menambah pengetahuan bagi pembaca tentang PTM, hidroksilasi protein, dan metode XGBoost.

Proses penulisan skripsi ini tidak terlepas dari dukungan banyak pihak yang telah membimbing, membantu, dan mendukung, sehingga pada kesempatan ini saya ingin menyampaikan ungkapan terima kasih kepada:

1. Orangtua, kakak, dan keluarga yang selalu mendoakan, memberi dukungan, kasih sayang, dan semangat baik secara moral maupun material dalam menyelesaikan skripsi ini.
2. Bapak Favorisen R. Lumbanraja, Ph.D. sebagai pembimbing yang telah membimbing, memberikan kritik dan saran dalam menyelesaikan skripsi ini yang dapat diselesaikan dengan baik.
3. Bapak M. Reza Faisal, S.T., M.T., Ph.D. sebagai pembahas pertama yang telah membimbing dalam memberikan ide, kritik, saran sehingga penulisan skripsi ini dapat diselesaikan dengan baik.
4. Bapak Dr. Ir. Kurnia Muludi, M.S.Sc. sebagai pembahas kedua yang telah membimbing dalam memberikan ide, kritik, saran sehingga penulisan skripsi ini dapat selesai dengan baik
5. Bapak Dr. Eng. Suripto Dwi Yuwono, S.Si., M.T. selaku dekan FMIPA Universitas Lampung.

6. Bapak Dwi Sakethi, M.Kom. sebagai pembimbing akademik yang telah membantu dan selalu mendukung peningkatan akademik.
7. Bapak Didik Kurniawan, S.Si., M.T., selaku Ketua Jurusan Ilmu Komputer Universitas Lampung.
8. Bapak Dr. rer. nat. Akmal Junaidi, M. Sc. selaku sekretaris Jurusan Ilmu Komputer FMIPA Universitas Lampung.
9. Bapak dan Ibu Dosen Jurusan Ilmu Komputer FMIPA Universitas Lampung yang telah memberikan ilmu dan pengalaman selama di perkuliahan.
10. Teman-teman saya, Aulia Ahmad Nabil, Dewi Lestari, Fikri Mulya Permana, Rahmadila Nurjannah, Ratih Indah Wardani, dan Suci Hikmawati yang telah menemani, membantu, dan memotivasi selama masa perkuliahan.
11. Teman seperbimbingan saya, Annisa Nurwalikadani, M. Fajru Ramadhan, M. Sepryan Astrayesa, Ridho Alrafi, dan Syela Septania yang telah membantu, menemani, dan memberi semangat satu sama lain.
12. NCT dan BTS yang karyanya telah memotivasi dan memberi semangat selama proses pengerjaan skripsi.
13. Teman-teman Ilmu Komputer 2018 yang telah memberikan pengalaman tak ternilai semasa duduk di bangku kuliah..
14. Semua pihak yang telah berpartisipasi baik secara langsung maupun tidak langsung dalam membantu penyusunan skripsi ini.

Saya menyadari bahwa dalam penulisan skripsi ini masih terdapat banyak kekurangan. Oleh karena itu, saran dan kritik yang membangun sangat diharapkan sebagai bahan evaluasi untuk kedepannya. Semoga skripsi ini dapat bermanfaat bagi semua pihak.

Bandar Lampung, 9 November 2022

Intania Rahmadhilla

DAFTAR ISI

Halaman

DAFTAR TABEL

Tabel Halaman

DAFTAR GAMBAR

Gambar Halaman

DAFTAR KODE PROGRAM

Potongan Kode Program Halaman

I. PENDAHULUAN

1.1. Latar Belakang

Modifikasi pasca-translasi (PTM) adalah modifikasi kimia protein setelah translasi. Modifikasi pasca-translasi berarti perubahan rantai samping asam amino protein setelah biosintesisnya. Sebagian besar protein harus menjalani modifikasi pasca-translasi (PTM) sebelum menjalankan fungsi biologisnya. Selama sintesis protein, protein dibangun menggunakan dua puluh asam amino yang berbeda, namun setelah translasi, modifikasi asam amino pasca-translasi dapat diamati dengan menempelkannya pada gugus fungsional biokimia (Basu and Plewczynski, 2010). Hidroksilasi protein merupakan salah satu jenis PTM yang memperkenalkan gugus hidroksil (-OH) ke dalam residu. Prolin (P) dan lisin (K) adalah dua residu terhidroksilasi yang umum dalam protein manusia yang dapat dihidroksilasi, masing-masing membentuk hidroksiprolin atau HyP dan hidroksilisin atau HyL (Xu et al., 2014).

Protein adalah kelompok biomolekul berukuran besar yang terbentuk dari satu rantai panjang asam amino atau lebih. Hidroksilasi residu prolin dan lisin dalam protein merupakan salah satu proses modifikasi protein pasca-translasi yang paling melimpah yang dikatalisis oleh tiga enzim, prolyl 4-hydrolase, prolyl 3-hydrolase dan lysyl hydrolase. Prediksi komputasi yang akurat dari situs hidroksilasi protein merupakan pendekatan yang berharga dan efisien untuk mengidentifikasi situs hidroksilasi potensial baru.

Saat ini, spektrometri massa adalah eksperimen yang paling umum digunakan dalam mengidentifikasi residu hidroksilasi (Cockman et al., 2009). Dengan bantuan spektrometri massa skala besar, informasi hidroksiprolin dan hidroksilisin dapat ditentukan. Namun, hal ini memakan waktu dan biaya mahal. Oleh karena itu, metode komputasi untuk mengatasi masalah tersebut sangat dibutuhkan. Banyak peneliti menggunakan metode pembelajaran mesin yang berbeda untuk memprediksi residu hidroksilasi. Sebagai contoh, pada penelitian Xu, et al. (2014) menyajikan prediktor yang disebut iHyd-PseAAC untuk memprediksi situs hidroksilasi protein. Selanjutnya penelitian Qiu, et al. (2016) menyajikan klasifikasi metode prediksi baru menggunakan Random Forest. *Machine learning* banyak digunakan oleh ahli biologi untuk merancang dan menafsirkan eksperimen. Machine learning sebagai metode komputasi yang memanfaatkan *experience* untuk meningkatkan akurasi prediksi (Mohri et al., 2012). Data yang digunakan berupa data elektronik yang dikumpulkan dan disediakan untuk analisis (training dataset).

Penelitian ini menggunakan metode Extreme Gradient Boosting (XGBoost) sebagai klasifikasi situs hidroksilasi. XGBoost merupakan algoritma berbasis pohon yang digunakan untuk klasifikasi. Metode XGBoost mampu 10 kali lebih cepat dibandingkan metode *gradient boosting* lainnya (Chen & Guestrin, 2016).

1.2. Rumusan Masalah

Adapun rumusan masalah dalam penelitian ini sebagai berikut.

1. Apakah metode *Extreme Gradient Boosting* (XGBoost) lebih baik daripada metode pada penelitian sebelumnya?
2. Berapa tingkat akurasi metode *Extreme Gradient Boosting* (XGBoost) untuk klasifikasi hidroksilasi protein?

1.3. Batasan Masalah

Adapun batasan masalah penelitian ini adalah sebagai berikut.

1. Metode penelitian yang digunakan berfokus pada metode *Extreme Gradient Boosting* (XGBoost).
2. Data yang digunakan diperoleh dari dataset benchmark berdasarkan penelitian yang digunakan oleh (Xu et al., 2014) dan (Qiu et al., 2016).

1.4. Tujuan Penelitian

Adapun tujuan penelitian ini adalah sebagai berikut.

1. Mengetahui hasil kinerja model klasifikasi *Extreme Gradient Boosting* (XGBoost) pada data hidroksilasi residu prolin dan lisin.
2. Membandingkan hasil kinerja klasifikasi dengan penelitian sebelumnya oleh (Qiu et al., 2016) yang melakukan identifikasi hidroksiprolin dan hidroksilisin dalam protein dengan menggunakan metode Random Forest.

1.5. Manfaat

Adapun manfaat penelitian ini adalah sebagai berikut.

1. Menambah pengetahuan tentang klasifikasi data menggunakan metode *Extreme Gradient Boosting* (XGBoost).
2. Hasil penelitian ini diharapkan dapat menjadi informasi untuk penelitian selanjutnya dalam prediksi hidroksilasi protein.

II. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Penelitian terdahulu dalam penelitian ini digunakan sebagai acuan dan perbandingan untuk hasil klasifikasi. Beberapa penelitian yang digunakan ini dapat dilihat pada Tabel 1.

Tabel 1. Penelitian Terdahulu Tentang Identifikasi Hidroksilasi

No	Penelitian	Data	Metode	Hasil
1	Identify hydroxyproline and hydroxylysine in proteins by incorporating sequence-coupled effects into general PseAAC (Qiu et al., 2016)	**Dataset 1** **Prolin (P)** Jumlah : 4356 Positif : 851 Negatif : 3505 **Dataset 2** **Lisin (K)** Jumlah : 1122 Positif : 142 Negatif : 980	Random Forest	Dataset 1 : Akurasi : 96.58% Sn : 86.35% Sp : 99.12% MCC : 0.89% Dataset 2 : Akurasi : 97.08% Sn : 78.77% Sp : 99.80% MCC : 0.86% *Sn = sensitifitas Sp = spesifisitas MCC = *Matthew Correlation Coefficient*

Tabel 1. Penelitian Terdahulu Tentang Identifikasi Hidroksilasi (Lanjutan)

No	Penelitian	Data	Metode	Hasil
2	A Hybrid Deep Learning Model for Predicting Protein Hydroxylation Sites (Long et al., 2018)	**Dataset 1** **Prolin (P)** Jumlah : 3335 Positif : 636 Negatif : 2699	CNN+LSTM	*Dataset* 1 = Akurasi : 90.08% Sn : 94.52% Sp : 97.43% MCC : 0.91%
		***Dataset* 2** **Lisin (K)** Jumlah : 943 Positif : 107 Negatif : 836		*Dataset* 2 = Akurasi : 93.27% Sn : 97.30% Sp : 99.84% MCC : 0.94
		***Dataset* 3** **Prolin (P)** Jumlah : 306 Positif : 306 Negatif : 1035		*Dataset* 3 = Akurasi : 89.15% Sn : 92.24% Sp : 95.72% MCC : 0.04%
		Dataset 4 **Lisin (K)** Jumlah : 572 Positif : 44 Negatif : 528		*Dataset* 4 = Akurasi : 96.91% Sn : 98.84% Sp : 97.66% MCC : 0.97%
		***Dataset 5* =** **Prolin (P)** Jumlah : 4356 Positif : 851 Negatif : 3505		*Dataset* 5 = Akurasi : 96.55% Sn : 92.06% Sp : 98.39% MCC : 0.91%
		***Dataset 6* =** **Lisin (K)** Jumlah : 1122 Positif :142 Negatif : 980		*Dataset* 6 = Akurasi : 97.19% Sn : 94.75% Sp : 95.58% MCC : 0.89%

Tabel 1. Penelitian Terdahulu Tentang Identifikasi Hidroksilasi (Lanjutan)

No	Penelitian	Data	Metode	Hasil
3	A tool for hydroxyproline and hydroxylysine sites prediction in the human proteome (Huang et al., 2020)	***Dataset 1*** **Prolin (P)** Jumlah : 1431 Positif : 190 Negatif : 1241	Random Forest	Akurasi : 90.74% Sn : 96.29% Sp : 85.18% MCC: 0.82 %
		Dataset 2 **Lisin (K)** Jumlah : 934 Positif : 58 Negatif : 438		Akurasi : 81.25% Sn :75% Sp : 87.50% MCC : 0.63%

Berdasarkan Tabel 1 terdapat tiga penelitian terdahulu yang digunakan. Penelitian pertama dilakukan oleh Qiu, et al. (2016) yang menggunakan metode *random forest*. Data yang digunakan didapat dari *benchmark dataset*. Dalam penelitian ini prediktor baru dikembangkan untuk memprediksi situs hidroksilasi pada lisin dan prolin yang disebut dengan IHyd-PseCp. Pada hasil kinerja *dataset* 1, prediktor ini mendapatkan nilai akurasi sebesar 96.58%, Sn *(Sensitivity*) 86.35%, Sp *(Specificity)* 99.12%, dan MCC 0.89%. Sedangkan pada *dataset* 2, hasil yang didapatkan adalah ACC *(accuracy)* 97.08%, Sn *(Sensitivity*) 78.77%, Sp *(Specificity)* 99.80%, dan MCC 0.86%.

Pada penelitian kedua dilakukan oleh Long, et al. (2018), penelitian ini menggunakan dataset menggunakan PseAAC dan PSSM sebagai *feature extraction*. Penelitian ini mengembangkan prediktor baru untuk mengidentifikasi hidroksiprolin dan hidroksilisin dalam protein dengan model *hybrid deep learning model convolutional neural network* (CNN) dan *long short-term memory network* (LSTM). Selanjutnya, penelitian ini dibandingkan dengan beberapa metode, yaitu CNN, iHyd-PseCp, dan iHyd-PseAAC. Hasilnya, CNN+LSTM mengungguli metode lain di hampir semua kriteria. Di antara semua prediktor, kinerja iHyd-PseAAC adalah yang terburuk, sementara

CNN dan iHyd-PseCp memiliki hasil komparatif. Data yang digunakan adalah enam dataset. Dataset pertama diunduh dari iHyd-PseAAC dan dua dataset terakhir diunduh dari iHyd-PseCp. Pada hasil kinerja dataset 1, metode CNN+LSTM mendapatkan Sn *(sensitivity)* sebesar 94.52%, Spe *(Specificity)* 97,43&, ACC *(accuracy)* 90,68 %, dan MCC sebesar 0,91%. Pada dataset 2, metode CNN+LSTM mendapatkan Sn *(sensitivity)* sebesar 97.30%, Spe *(Specificity)* 99,84% &, ACC *(accuracy)* 93,27 %, dan MCC sebesar 0,94%. Pada dataset 3, metode CNN+LSTM mendapatkan Sn *(sensitivity)* sebesar 92.24%, Spe *(Specificity)* 95,72% &, ACC *(accuracy)* 89,15 %, dan MCC sebesar 0,90%. Pada dataset 4, metode CNN+LSTM mendapatkan Sn *(sensitivity)* sebesar 98.84%, Spe *(Specificity)* 97,66 %, ACC *(accuracy)* 96,91 %, dan MCC sebesar 0,97%. Pada dataset 5, metode CNN+LSTM mendapatkan Sn *(sensitivity)* sebesar 92.06%, Spe *(Specificity)* 98,39%, ACC *(accuracy)* 96,55 %, dan MCC sebesar 0,91%. Pada dataset terakhir, metode CNN+LSTM mendapatkan Sn *(sensitivity)* sebesar 94,75%, Spe *(Specificity)* 97,43%, ACC *(accuracy)* 97,19 %, dan MCC sebesar 0,89%.

Penelitian yang ketiga dilakukan oleh Huang, et al. (2020) yang mengusulkan HydLoc, yaitu prediktor berbasis *random forest* untuk identifikasi situs hidroksilasi manusia. Data yang digunakan adalah *database* UniProt dengan total 567 situs hidroksilasi prolin dari 38 protein dan 78 situs hidroksilasi lisin dari 18 protein. Setelah itu hasilnya dibandingkan dengan iHydPseAAC dan HydPred berdasarkan dataset uji independen. Hasil dari perbandingan ketiga prekdiktor tersebut menunjukan bahwa tingkat akurasi HydLock sebesar 90.74%, iHydPseAAC 72.22%, dan HydPred 68.51%. Untuk lebih menunjukkan kinerja metode yang digunakan, dilakukan lagi pengujian untuk melakukan perbandingan dengan metode *Support Vector Machine* (SVM), *Gaussian Naïve Bayes* (GNB) dan *Gradient Tree Boosting* (GTB) berdasarkan set uji independen. Hasilnya, untuk residu prolin (P) *Random Forest* memperoleh ACC *(accuracy)* 90,74%, spe *(specificity)* 85,18%, dan MCC

0,8198 tertinggi dibandingkan ketiga metode lainnya. Sementara itu, *random forest* (RF) mencapai sen *(sensitivity)* sebesar 96,29%, hanya 0,01% lebih rendah dari GTB. Untuk Lys (K), performa RF dan GNB sama dan keduanya memperoleh ACC *(accuracy)* tertinggi 81,25%, sen *(sensitivity)* 75,00%, spe *(Specificity)* 87,50%, dan MCC 0,6299. Hal tersebut menunjukkan bahwa algoritma *random forest* lebih unggul. Setelah melakukan pengujian pada tingkat protein, peneliti membuktikan bahwa HydLoc dapat diandalkan untuk mencari situs hidroksilasi potensial dari protein manusia.

2.2. *Post Translational Modification*

Hampir setiap protein dalam sel mengalami modifikasi selama masa hidupnya. Modifikasi pasca-translasi terlibat dalam regulasi dan stabilisasi struktural protein eukariotik (Minguez et al., 2013). Modifikasi pasca-translasi mengacu pada penambahan kovalen dan enzimatik modifikasi protein selama atau setelah biosintesis protein, yang memainkan peran penting dalam memodifikasi fungsi protein dan mengatur ekspresi gen. Sebagian besar protein menjalankan fungsinya setelah modifikasi pasca-translasi (PTM). Jenis-jenis PTM dapat dilihat pada Tabel 2 dan Gambar 1.

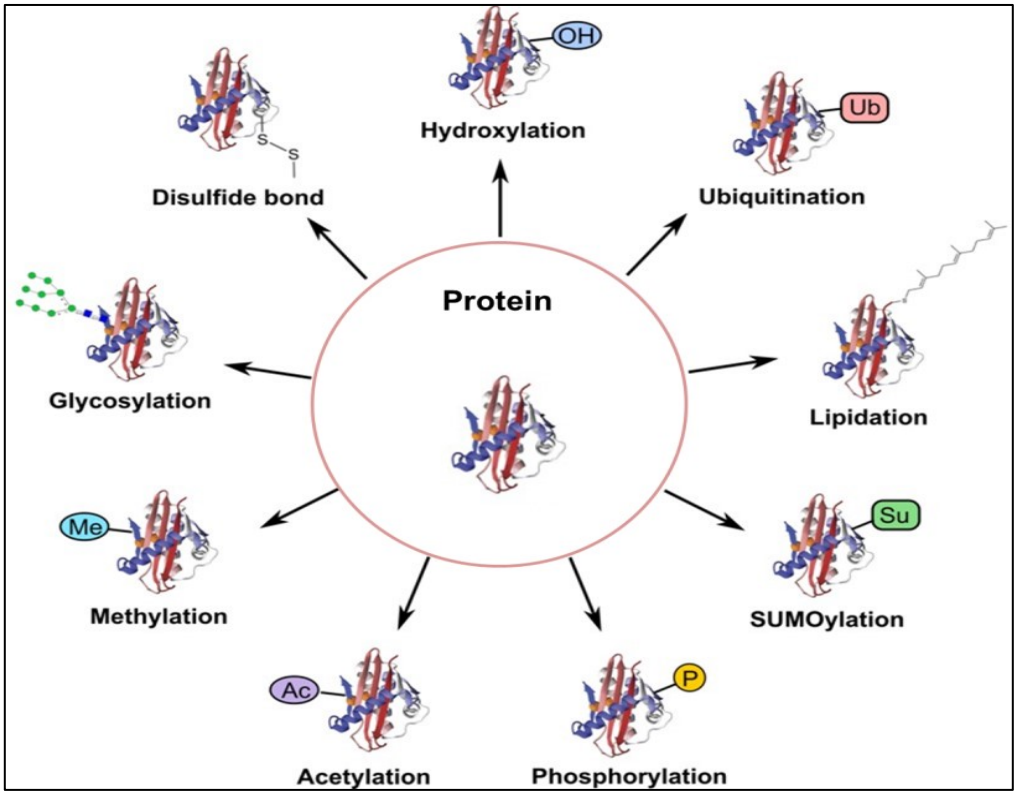
Tabel 2. Jenis-jenis *Post Translational Modification* (PTM) (Long et al., 2018)

No	PTM	Penjelasan
1	Hidroksilasi	Proses kimia yang memasukkan gugus hidroksil (-OH) ke dalam senyawa organik.
2	Glikolisasi	Proses penambahan gugus gula (glikosil) pada untaian polipeptida.
3	Ubiquitinasi	Proses penambahan protein ubikitin ke substrat protein.
4	Sumoilasi	proses penambahan gugus SUMO pada substrat protein.

Tabel 2. Jenis-jenis *Post Translational Modification* (PTM) (Lanjutan)

No	PTM	Penjelasan
5	Metilasi	Penambahan gugus metil pada substrat.
6	Asetilasi	Reaksi kimia yang melibatkan proses introduksi gugul asetil ke senyama kimia lain.

Detail gambar jenis-jenis PTM dapat dilihat pada Gambar 1.



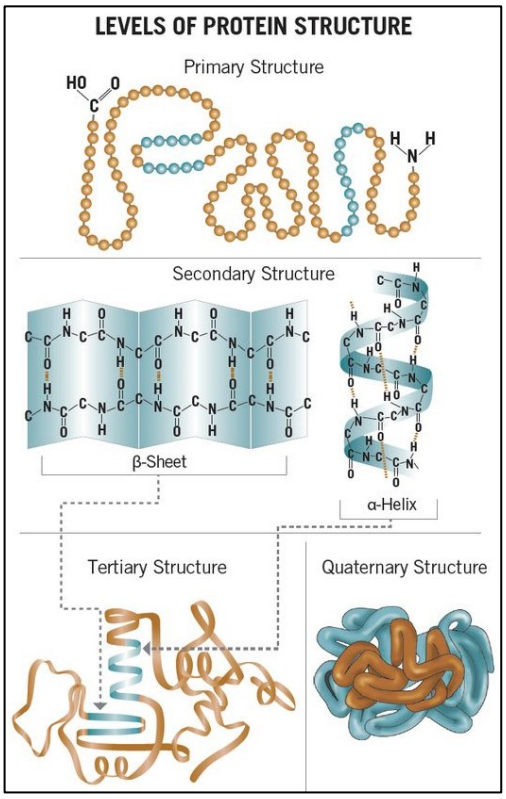
Gambar 1. Jenis-jenis Post Translational Modification. (https://www.creative-proteomics.com/blog/index.php/strategies-for-post-translational-modifications-ptms/).

2.3. Protein

Protein adalah dasar dari fungsi seluler dan fisiologis dalam organisme hidup. Protein adalah polimer dari monomer asam amino yang dihubungkan satu sama

lain dengan ikatan peptida. Molekul protein mengandung karbon, hidrogen, oksigen, nitrogen dan kadang kala sulfur serta fosfor. Protein berperan penting dalam struktur dan fungsi semua sel makhluk hidup. Kebanyakan protein merupakan enzim atau subunit enzim (Simamora, 2013).

Ada 20 jenis asam amino yang menyusun protein, termasuk 9 asam amino esensial dan 1 asam amino non-esensial (Diana, 2009). Setiap protein terdiri dari satu atau lebih rantai peptida, sehingga terdapat empat struktur protein.



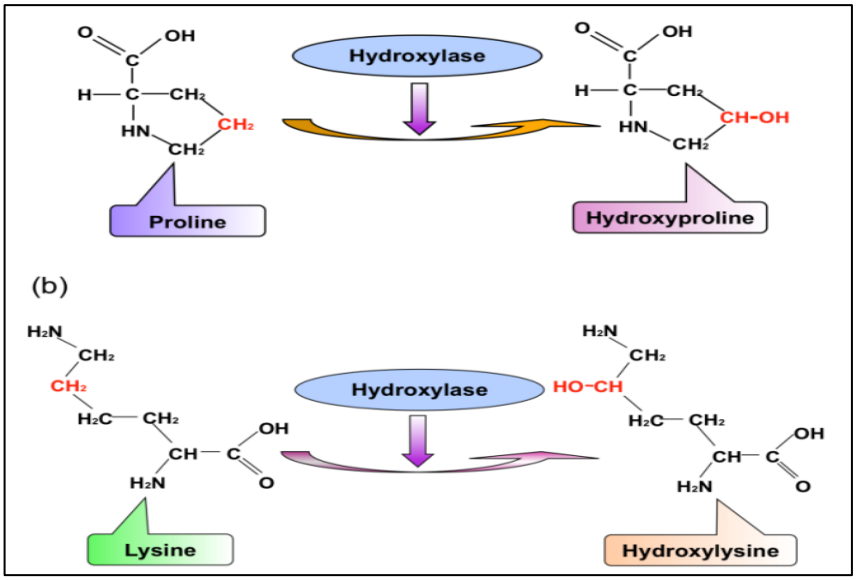
Gambar 2. Empat Tingkatan Struktur protein (*https://lubrizolcdmo.com/technicalbriefs/protein-structure/*).

Gambar 2 merupakan tingkatan struktur protein. Struktur primer protein adalah ikatan peptida yang membentuk protein, struktur sekunder protein terkait dengan bentuk rantai polipeptida, struktur tersier protein menunjukkan pembentukan lipatan *(folding polypeptide),* yang disebabkan oleh adanya

interaksi antar gugus pada R, dan struktur kuartener adalah penggabungan subunit atau 1 rantai polipeptida penyusun protein.

2.4. Hidroksilasi

Hidroksilasi protein adalah modifikasi pascatranslasi (PTM), dimana gugus CH pada residu Prolin (P) atau Lisin (K) diubah menjadi gugus COH atau gugus hidroksil (-OH). Hidroksiprolin memainkan peran penting dalam stabilisasi kolagen dan perkembangan beberapa jenis kanker, seperti kanker perut dan kanker paru-paru, sedangkan hidroksilisin berkontribusi pada fibrillogenesis, ikatan silang, dan mineralisasi matriks. Sehingga, memprediksi situs hidroksiprolin dan hidroksilisin dalam protein dapat memberikan informasi yang berguna untuk penelitian biomedis (Long et al., 2018). Berikut adalah skema hidroksilasi protein yang terjadi pada prolin dan lisin.



Gambar 3. Skema hidroksilasi protein pada prolin dan lisin (Xu et al., 2014).

Prolin (P) dan lisin (K) adalah dua residu terhidroksilasi yang umum dalam protein manusia yang dapat dihidroksilasi, masing-masing membentuk hidroksiprolin atau HyP dan hidroksilisin atau HyL (Xu et al., 2014). Selama sintesis protein, sebuah protein terdiri dari dua puluh asam amino yang berbeda,

namun setelah translasi, modifikasi asam amino pasca-translasi dapat diamati dengan menempelkannya pada gugus fungsi biokimia lainnya seperti asetat, fosfat, berbagai lipid dan karbohidrat, dengan mengubah sifat kimia asam amino, atau dengan membuat perubahan struktural, seperti pembentukan jembatan disulfida (Basu and Plewczynski, 2010).

2.5. *Feature Extraction*

Ekstraksi fitur digunakan untuk mengenali objek dengan melihat karakteristik khusus yang dimiliki objek tersebut. Tujuan dari ekstraksi fitur untuk menjalankan perhitungan yang digunakan untuk mengklasifikasikan ciri-ciri yang dimiliki oleh suatu objek (Bahri & Maliki, 2012). Ekstraksi fitur pada *sequence* protein disebut dengan protein *descriptor*. Protein *descriptor* yang digunakan menggunakan *package BioSeqClass*. Protein *descriptor* digunakan untuk melakukan proses ekstraksi fitur. Penelitian ini menggunakan *descriptor* PseAAC, CTD, dan AAindex sebagai fitur ekstraksi. Dari ketiga fitur ekstraksi tersebut panjang fitur yang dihasilkan disebut *dynamic length* dan *fixed length*. *Dynamic length* merupakan *descriptor* yang jumlah fiturnya berubah-ubah sesuai panjang *sequence*, sedangkan *fixed length* merupakan kebalikan dari *dynamic length* yaitu *descriptor* yang memiliki jumlah fitur yang sama, dengan kata lain fitur yang dihasilkan sudah ditetapkan dan tidak berubah. Pada penelitian ini fitur ekstraksi yang termasuk *dynamic length* adalah PseAAC dan AAindex, sedangkan fitur ekstraksi yang termasuk *fixed length* adalah CTD.

2.5.1. PseAAC (*Pseudo Amino Acid Composition*)

PseAAC diusulkan oleh Chou (2001). Ini dirancang untuk meningkatkan kualitas prediksi atribut protein, termasuk lokalisasi subseluler dan jenis protein membran. Dibandingkan dengan komposisi asam amino konvensional, komposisi asam amino semu tidak hanya mengubah urutan protein dengan berbagai panjang menjadi vektor

digital dengan panjang tetap, tetapi juga menyimpan informasi urutan urutan yang cukup besar (Du et al., 2012). PseAAC dapat digunakan untuk mewakili urutan protein dengan model diskrit tanpa kehilangan informasi urutan urutan sepenuhnya. Menurut definisinya, PseAAC dari sampel protein yang diberikan diwakili oleh satu set lebih dari 20 faktor diskrit, di mana 20 faktor pertama mewakili komponen komposisi asam amino (AA) konvensionalnya sedangkan faktor tambahan menggabungkan beberapa komponennya. Jumlah fitur yang dihasilkan PseACC pada penelitian ini berjumlah 40 fitur. PseAAC termasuk *dynamic length* karena jumlah fitur yang dihasilkan tergantung dengan panjang *sequence* yang diberikan. Contoh program ekstraksi fitur PseAAC dapat dilihat pada Potongan Kode Program 1.

```
Input :
LIGSA
IGSAS


featurePseudoAAComp(seq,d,w=0.5)
```

Potongan Kode Program 1. Kode Program PseAAC menggunakan *package bioseqclass.*

Seq adalah vektor string untuk urutan protein, DNA, atau RNA, *d* adalah bilangan bulat yang digunakan sebagai parameter *featurePseudoAAComp* (d>=1), sedangkan *w* adalah nilai numerik untuk faktor bobot efek urutan urutan di *featurePseudoAAComp*. *Output* dari Potongan Kode Program 1 dapat dilihat pada Tabel 3.

Tabel 3. Contoh *output* fitur ekstraksi PseAAC

	PAC:R	PAC:K	PAC:E	PAC:G	PAC:S	PAC: …
LIGSA	0	0	0	0.1722578	0.1722578	…..
IGSAS	0	0	0	0.1672177	0.3344353	…..

2.5.2. *Composition, Transition, Distribution* (CTD)

CTD pertama kali diperkenalkan oleh Dubchak, et al. (1995) untuk memprediksi kelas pelipatan protein. CTD terdiri dari sifat-sifat seperti hidrofobisitas, polaritas, volume van der Waals yang dinormalisasi, polarisasi, struktur sekunder yang diprediksi, dan aksesibilitas pelarut. *Composition* atau komposisi (C) mewakili persentase komposisi setiap kelompok dalam urutan peptida (Manavalan et al., 2018). *Composition* (C) dapat dihitung dengan Persamaan 1.

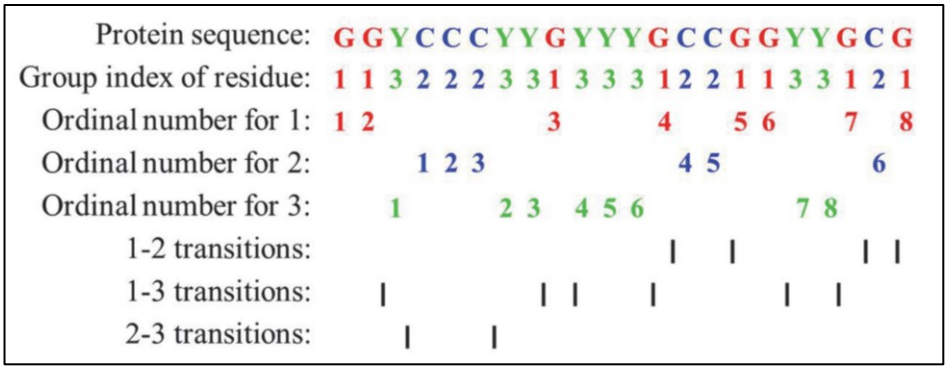
$$Composition\ (C) = \frac{Ne}{N} \quad \textbf{(1)}$$

Transition atau transisi (T) merupakan probabilitas transisi antara dua asam amino tetangga milik dua kelompok yang berbeda (Manavalan et al., 2018). *Transition* (T) dapat dihitung dengan Persamaan 2.

$$Transition\ (T) = \frac{N_{nm} + N_{mn}}{N-1} \quad \textbf{(2)}$$

Di mana $N_{nm} + N_{mn}$ adalah jumlah peptida dan N adalah panjang *sequence.*

Distribution atau Distribusi (D) mewakili posisi asam amino (25, 50, 75, atau 100% pertama) di setiap kelompok dalam urutan protein (Manavalan et al., 2018). Untuk contoh perhitungan fitur ekstraksi CTD dapat dilihat pada Gambar 4.



Gambar **4**. Contoh fitur ekstraksi CTD (You et al., 2015).

Gambar 4 memiliki input GGYCCCYYGYYYGCCGGYYGCG yang dimana setiap huruf diwakili oleh angka 113222331333122113312 1. Terdapat huruf G berjumlah 8, C berjumlah 6, dan Y berjumlah 8 pada urutan *sequence* protein tersebut. Selanjutnya, *composition* dihitung dengan Persamaan 1. Sehingga didapatkan hasil sebagai berikut.

$$C(G) = \frac{8}{8+6+8}\ x\ 100\% = 36.36\%$$

$$C(C) = \frac{6}{8+6+8}\ x\ 100\% = 27.27\%$$

$$C(Y) = \frac{8}{8+6+8}\ x\ 100\% = 36.36\%$$

Selanjutnya transitions dapat dihitung dengan Persamaan 2. Ada 3 *transitions*, yaitu dari ‘1’ ke ‘2’ atau dari ‘2’ ke ‘1’ dengan persentase $T(1,2) = \frac{2+2}{22-1} = 19\%$. Kedua dari ‘1’ ke ‘3’ atau dari ‘3’ ke ‘1’ dengan persentase $T(1,3) = \frac{3+3}{22-1} = 28.75\%$, dan ketiga dari ‘2’ ke ‘3’ dan dari ‘3’ ke ‘2’ dengan persentase $T(2,3) = \frac{1+1}{22-1} = 9.52\%$. Untuk *distribution D*, ada 8 residu yang dikodekan sebagai ‘1’ pada Gambar 4. Pada residu pertama ‘1’, residu kedua ‘1’ (25% x 8 = 2), residu keempat ‘1’ (50% x 8 = 4), residu ke enam ‘1’ (75% x 8 = 6) dan residu ke delapan ‘1’ (100% x 8 = 8) dalam urutan masing-masing *sequence* protein yaitu 1, 2, 13, 17, 22. Jadi deskriptor D untuk '1' adalah (1/22) ×100% = 4.55%, (2/22) ×100% = 9.09%, (13/22) ×100% = 59.09%, (17/22) ×100% = 77.27%, (22/22)×100% = 100%, berturut-turut. Demikian pula, deskriptor D untuk '2' dan '3' adalah (18.18%, 18.18%, 27.27%, 63.64%, 95.45%) dan (13.64%, 31.82%, 45.45%, 54.55%, 86.36%) masing-masing.

CTD menghasilkan 21 fitur, jumlah fitur yang dihasilkan tidak dipengaruhi oleh panjang *sequence*. Jumlah fitur yang dihasilkan CTD

tidak akan berubah sepanjang apapun *sequence* nya sehingga disebut *fixed length*. Contoh program ekstraksi fitur CTD dapat dilihat pada Potongan Kode Program 2.

```
Input :
LIGSA
IGSAS

featureCTD(seq,class=aaClass("aaV"))
```

Potongan Kode Program 2. Kode program CTD menggunakan *package bioseqclass.*

Output CTD adalah kombinasi dari tiga matriks yang berbeda, yaitu komposisi, transisi, dan distribusi. *Ouput* hasil fitur CTD pada Pseudocode 2 dapat dilihat pada Tabel 4.

Tabel 4. Contoh *output* fitur ekstraksi CTD

	CTD:C: small	CTD:C: medium	CTD:C: Large	CTD:D:_ small_25%	CTD:D:_ small_50%	CTD:D: ….
LIGSA	0.6	0.4	0	0.6	0.8	….
IGSAS	0.8	0.2	0	0.4	0.6	….

2.5.3. AAindex

Amino Acid Indices (AAindex) adalah kumpulan data yang berisi ratusan berbagai sifat fisikokimia dan biologis asam amino. Setiap indeks disajikan dalam bentuk matriks numerik yang mewakili satu jenis sifat asam amino (Cai & Lu, 2008). Dalam *database* AAindex terdapat 554 indeks asam amino. AAindex terdiri dari tiga bagian, yaitu AAindex1, AAindex2, dan AAindex 3. AAindex1 untuk asam amino dari 20 nilai numerik, AAindex2 untuk matriks substitusi asam amino, dan AAindex3 untuk potensi kontak protein statistik.

Pada penelitian ini AAindex yang digunakan adalah AAindex1 yang bertujuan untuk memberikan informasi urutan *sequence* protein yang diberikan. *Output* AAindex tergantung pada parameter outFormat yang dapat berupa 'matrix' atau 'txt'. Jika outFormat adalah 'matrix', fungsi mengembalikan matriks fitur untuk urutan dengan panjang yang sama sehingga jumlah kolom adalah (panjang *sequence*) x (jumlah indeks asam amino yang dipilih) dan jumlah baris sama dengan jumlah urutan. Panjang AAindex termasuk *dynamic length* karena fitur yang dihasilkan berubah-ubah sesuai dengan panjang *sequence*. Rumus AAindex dapat dilihat pada Persamaan 3.

$$aaindex_{(i)} = \frac{\sum_{n=1}^{N} Aaindex_i(aa_n)}{N} \quad \textbf{(3)}$$

Di mana i pada Persamaan 3 menunjukkan x indeks $AAindexi$, N menunjukkan jumlah residu pada urutan sedangkan aan menunjukkan asam amino pada posisi n. Penggunaan Persamaan 3 itu sesuai dari jenis aaindex yang digunakan, dimana setiap *amino acid* sudah ditentukan nilainya. Jumlah fitur yang dihasilkan AAindex sama dengan panjang *sequence* yaitu 21 fitur. Contoh program ekstraksi fitur AAindex dapat dilihat pada Potongan Kode Program 3.

```
Input :
LIGSA
IGSAS

featureAAIndex(seq,”ANDN920101”))
```

Potongan Kode Program 3. Kode Program AAindex menggunakan *package bioseqclass.*

Percobaan parameter *(aaindex.name)* pada penelitian ini adalah ”ANDN920101”. *Output* Potongan Kode Program dapat dilihat pada Tabel 5.

Tabel 5. Contoh *output* fitur ekstraksi AAindex

	ANDN920101_1	ANDN920101_2	ANDN920101_n
LIGSA	4.17	3.95	…..
IGSAS	3.95	3.97	….

2.6. *Imbalanced* Data

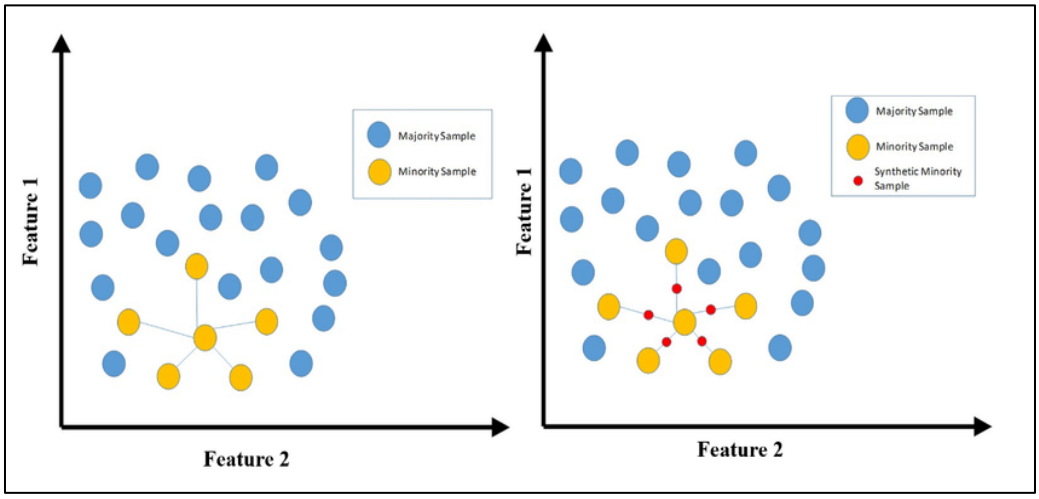
Dalam data set dunia nyata, distribusi anggota kelas tidak selalu seragam. Ini mengarah pada apa yang dikenal sebagai masalah ketidakseimbangan. Kumpulan data dengan distribusi kelas yang tidak seimbang menimbulkan tantangan berat bagi algoritma yang dirancang untuk meningkatkan akurasi klasifikasi secara keseluruhan (Mathew et al., 2015).

Masalah ketidakseimbangan data set terjadi pada klasifikasi, dimana jumlah *instance* dari satu kelas jauh lebih sedikit dibandingkan dengan *instanc*e dari kelas lainnya (Ramyachitra & Manikandan, 2014). Kumpulan data yang tidak seimbang biasanya mempengaruhi proses pelatihan model, karena sebagian besar algoritma pelatihan mengasumsikan distribusi yang merata di antara kelas, atau biaya kesalahan klasifikasi yang sama. Algoritma *machine learning* mengasumsikan bahwa kumpulan data seimbang dengan bobot kelas yang sama, dan karenanya cenderung mengklasifikasikan setiap sampel kasus uji ke dalam kelas mayoritas untuk meningkatkan metrik akurasi (Mishra, 2017).

2.7. *Synthetic Minority Oversampling Technique* (SMOTE)

Kumpulan data yang tidak seimbang dapat menjadi salah satu kendala bagi banyak algoritma *machine learning*. Ada tiga kategori pendekatan tingkat data. Algoritma SMOTE diusulkan oleh Chawla, et al. (2002) merupakan salah satu metode *oversampling* yang paling banyak digunakan. SMOTE menyeimbangkan *dataset* dengan menghasilkan titik data minoritas secara sintetis (Chawla et al., 2002). Dalam penelitian ini, teknik SMOTE diterapkan untuk menyelesaikan masalah ketidakseimbangan kelas pada dataset yang

digunakan. Teknik ini mensintesis sampel baru dari kelas minoritas untuk menyeimbangkan kumpulan data dengan melakukan resampling kelas minoritas (Siringoringo, 2018). Untuk ilustrasi SMOTE bisa dilihat pada Gambar 5.



Gambar 5. Ilustrasi oversampling menggunakan SMOTE (Vijayvargiya et al., 2021).

Metode SMOTE digunakan untuk menyeimbangkan dataset dengan cara menduplikasi kelas minoritas agar seimbang dengan kelas mayoritas dengan cara membuat *instanc*e baru berdasarkan k-*nearest neighbor* dengan pendekatan *Euclidean distance*. Secara umum, cara kerja algoritma *k-nn* adalah sebagai berikut.

1. Tentukan jumlah *k* yang akan digunakan
2. Hitung jarak dari data baru ke masing-masing data *point* di dataset.
3. Ambil sejumlah *k* data dengan jarak terdekat, kemudian tentukan kelas dari data baru tersebut.

Langkah-langkah melakukan SMOTE dimulai dengan menghitung jarak antar data minoritas, kemudian menentukan nilai persentase SMOTE dan menentukan jumlah *k* terdekat, dan terakhir menghasilkan data sintetik baru dengan Persamaan 4.

$$x_{syn} = x_i + (x_{knn} - x_i) \times \delta \quad \text{.....(4)}$$

Kode Program SMOTE dapat dilihat pada Potongan Kode Program 4.

```
data_example = sample_generator(10000,ratio = 0.80)
genData = SMOTE(data_example[,-3],data_example[,3])
genData_2 = SMOTE(data_example[,-
3],data_example[,3],K=5)
stimulateData = ceiling(abs(nrow(genData$syn_data)-
(balanceStep*(chunk_number-x))))
```

Potongan Kode Program 4. Kode Program SMOTE.

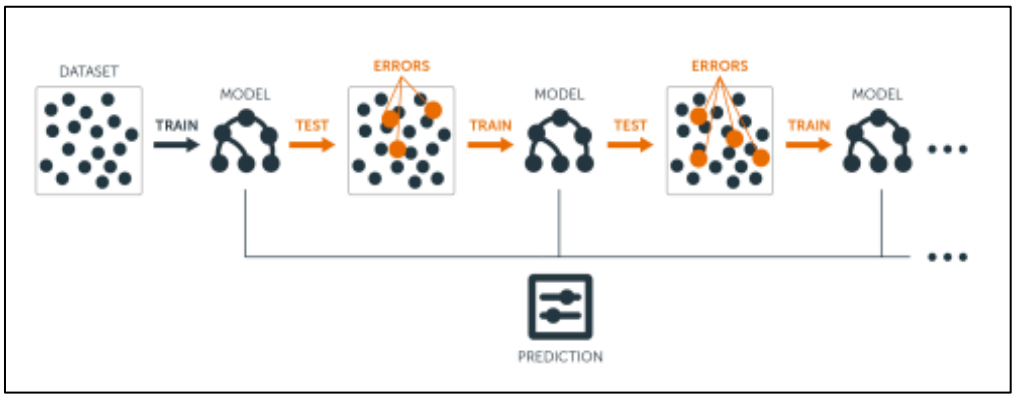
Konsep SMOTE adalah membuat data sintetis baru untuk menyeimbangkan kelas mayoritas dan minoritas. Proses pertama adalah proses *tuning* untuk menyeimbangkan kelas mayoritas dan kelas minoritas. Selanjutnya *package* yang digunakan adalah *smotefamily* untuk menyeimbangkan data dimana parameter yang dikirim adalah *k* dan *dup_size. k* berarti jumlah *neighbor* terdekat selama proses sampling dan *dup_size* adalah jumlah atau vektor yang mewakili waktu yang diinginkan dari *instance* minoritas sintetis di atas jumlah asli *instance* mayoritas. Setelah penyeimbangan data berhasil diproses lalu dilanjutkan ke proses pembulatan data dengan *ceiling* dan menampilkan proses *balancing* data serta menampilkan nilai dari sampel positif dan sampel negatifnya.

2.8. *Extreme Gradient Boosting* (XGBoost)

Algoritma *Gradient Boosting* dikembangkan untuk kemampuan prediksi yang sangat tinggi. *Gradient Boosting Decision Tree* (GBDT) merupakan sub-bab dari *decision tree* yang mencakup model XGBoost. Dalam GBDT, beberapa pohon keputusan *(decision tree)* dibangun di mana pelatihan setiap pohon bergantung pada pohon yang dilatih sebelumnya (Sagi & Rokach, 2021). Model *decision tree* banyak digunakan untuk memecahkan masalah pembelajaran mesin karena kesederhanaannya dan mudah dipahami (Alajali et al., 2018). Penggunaan GBDT telah menjadi lebih luas dalam beberapa tahun terakhir

karena beberapa perkembangan baru. XGBoost atau *Extreme Gradient Boosting* merupakan salah satu teknik pembelajaran mesin untuk mengatasi permasalahan regresi dan klasifikasi berdasarkan *Gradient Boosting Decision Tree* (GBDT).

Algoritma ini merupakan implementasi lanjutan dari *gradient boosting machines* (GBM). Membangun model baru untuk memprediksi error dari model sebelumnya digunakan dalam metode *boosting*. Pohon keputusan adalah representasi yang disederhanakan dari teknik klasifikasi yang merupakan proses mempelajari fungsi tujuan yang memetakan setiap set atribut ke salah satu kelas yang ditentukan sebelumnya**.** XGBoost pertama kali diperkenalkan oleh Friedman (2001). XGBoost merupakan sistem penambah pohon gradien yang dioptimalkan, dengan beberapa inovasi algoritmik. Algoritma ini membutuhkan lebih sedikit waktu pelatihan dan prediksi, dan mendukung berbagai fungsi objektif, termasuk klasifikasi dan regresi.



Gambar 6. Proses *Gradient Boosting Machines.*

Proses penambahan pohon ini terjadi satu per satu. *Output* yang dihasilkan di pohon baru kemudian ditambahkan ke output dari urutan pohon yang sudah ada sebelumnya untuk meningkatkan keluaran akhir model. Proses ini berhenti setelah nilai optimal yang tepat untuk *loss function* tercapai. Parameter pada metode XGBoost bisa dilihat pada Tabel 6.

Tabel 6. Parameter XGBoost

Parameter	Penjelasan
N_estimator [default=100]	Jumlah *cycle* hingga optimal
eta [default=0.3, alias: learning_rate]	Penyusutan ukuran yang digunakan untuk mencegah *overfitting*
gamma [default=0, alias: min_split_loss	Nilai minimum *loss reduction*
max_depth [default=6]	Kedalaman maksimum pada *tree*
min_child_weight [default=1]	Jumlah bobot minimum pada *child node*
subsample [default=1]	Jumlah sampel yang digunakan saat proses pelatihan sebelum membangun *tree*

Model XGBoost merupakan *tree ensemble*. Model tree ensemble adalah kumpulan pohon klasifikasi dan regresi (CART). Tabel 7 merupakan contoh sederhana dari cara kerja XGBoost.

Tabel 7. Contoh data sederhana untuk membangun XGBoost

Height (m)	Favorite Color	Gender	Weight (kg)
1.6	Biru	Laki-laki	88
1.6	Hijau	Perempuan	76
1.5	Biru	Perempuan	56
1.8	Merah	Laki-laki	73
1.5	Hijau	Laki-laki	77
1.4	Biru	Perempuan	57

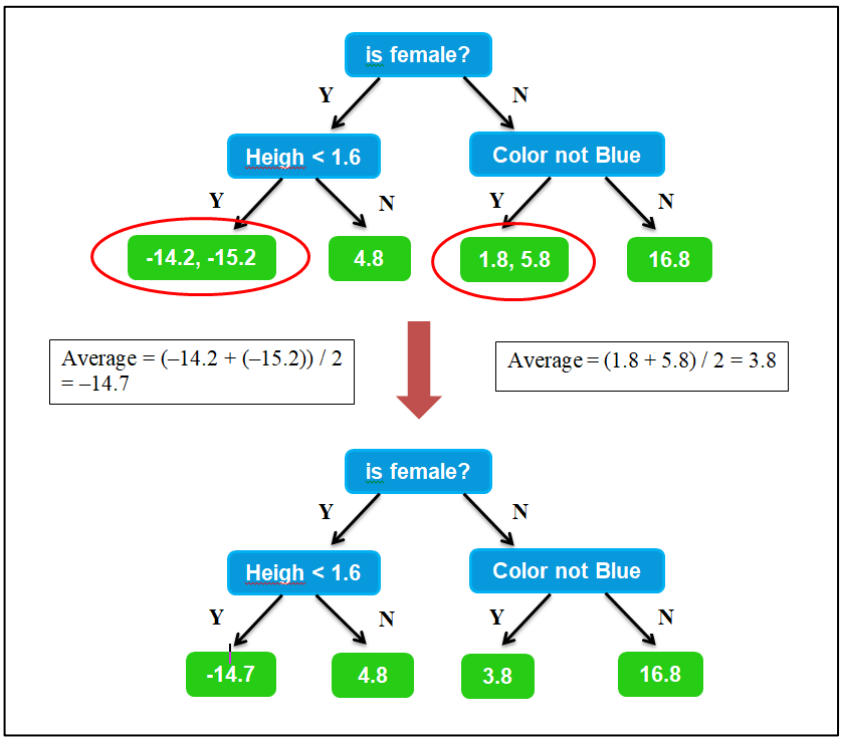
Tabel 7 memperlihatkan tinggi badan dari enam orang, warna favorit, gender, dan bobot. Bobot rata-rata badan ke enam orang tersebut adalah 71.2 kg. Selanjutnya adalah membangun pohon berdasarkan kesalahan (*error)* dari

pohon pertama. Kesalahan yang dibuat pohon sebelumnya adalah perbedaan antara bobot yang diamati dan bobot yang diprediksi. Selanjutnya hitung dengan memasukkan 71.2 kg untuk bobot yang diprediksi dan kemudian masukkan bobot yang diamati terlebih dahulu. Perhitungan nilai *error* atau *residuals* dapat dilihat pada Tabel 8.

Tabel 8. Perhitungan Nilai *Residuals*

Residuals (Error) = True – Predicted
88 – 71.2 = 16.8
76 – 71.2 = 4.8
56 – 71.2 = –15.2
73 – 71.2 = 1.8
77 – 71.2 = 5.8
57 – 71.2 = –14.2

Selanjutnya pohon *(tree)* akan dibangun menggunakan *Height, Favorite color,* dan *Gender* untuk memprediksi *Residuals.* Pembentukan pohon XGBoost dapat dilihat pada Gambar 7.



Gambar 7. Pembentukan Pohon XGBoost.

Berdasarkan hasil *tree* pada Gambar 7, terdapat dua data menuju *leaf* yang sama, untuk itu ganti *residuals* dengan rata-rata nya. *Gradient Boosting* mengatasi masalah ini dengan menggunakan *learning rate* untuk menskalakan kontribusi dari pohon baru. *Learning rate* adalah nilai antara 0 dan 1. Pada contoh ini, *learning rate* yang akan digunakan adalah 0.1.

Secara matematis, model dapat didefinisikan dalam bentuk Persamaan 5.

$\hat{y}_i = \sum_{k=1}^{K} f_k\,(x_i),\quad f_k \in \mathrm{F}$ .....**(5)**

Di mana K mewakili jumlah pohon dalam model, f_k mewakili pohon ke-*k, dan* $\hat{y}_i$ merupakan nilai prediksi. Nilai prediksi adalah jumlah skor yang diprediksi oleh pohon K. Sedangkan F adalah *space* dari pohon regresi atau yang biasa dikenal dengan CART. Untuk menyelesaikan masalah tersebut, perlunya himpunan fungsi yang digunakan dalam model dengan meminimalkan *loss* dan regularisasi.

$obj\ (\theta) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega\ (f_k)$ **(6)**

Di mana $l(y_i, \hat{y}_i)$ adalah *loss function* dan $\Omega\ (f_k)$ adalah *regularization term* atau istilah regularisasi, Ω membantu menghindari model yang terlalu *over-fitting*, dan dihitung menggunakan Persamaan 7.

$\Omega\ (f_k) = \gamma T + \frac{1}{2}\lambda\ \|w\|^2$ **(7)**

T pada persamaan di atas merupakan jumlah daun dari pohon dan w adalah berat setiap daun.

2.9. *Confusion Matrix*

Confusion matrix merupakan metode yang digunakan untuk mengukur kinerja suatu model klasifikasi. Untuk memberikan metode yang lebih intuitif dan mudah dipahami untuk mengukur kualitas prediksi, rangkaian *confusion matrix* diadopsi berdasarkan formulasi yang digunakan oleh (Chou, 2000) dalam memprediksi sinyal peptide. Contoh *confusion matrix* untuk klasifikasi dua kelas dapat dilihat pada Tabel 9.

Tabel 9. Confusion Matrix (Bekkar et al., 2013)

	Actual Positive	**Actual Negative**
Predicted Positive	TP	FP
Predicted Negative	FN	TN

Akronim yang digunakan pada Tabel 9 adalah :

a. TP *(True Positive)* : Jumlah kasus positif yang diidentifikasi dengan benar sebagai positif.
b. TN *(True Negative)* : Jumlah kasus negatif yang benar diidentifikasi sebagai kasus negatif.
c. FP *(False Positive)* : Jumlah kasus negatif yang salah diidentifikasi sebagai kasus positif.

d. FN *(False Negative)* : Jumlah kasus positif yang salah diklasifikasikan sebagai kasus negatif.

Beberapa matriks pengukuran yang digunakan untuk mengevaluasi kinerja prediksi model adalah *accuracy* (Acc), *specificity* (Spe), *sensitivity* (Sn), dan *Matthew Correlation Coefficient* (MCC).

2.9.1. *Accuracy*

Akurasi adalah proporsi dari jumlah total prediksi yang benar (Powers, 2020). Untuk mendapatkan nilai akurasi matrik dapat dihitung dengan Persamaan 8.

$$\boldsymbol{ACC = \frac{TP+TN}{TP+FP+TN+FN}} \quad \textbf{(8)}$$

2.9.2. *Specificity*

Specificity adalah proporsi kasus negatif yang diidentifikasi dengan benar (Powers, 2020). Untuk mendapatkan nilai *specificity* dapat dihitung dengan Persamaan 9.

$$\boldsymbol{Sp = \frac{TN}{(TN+FP)}} \quad \textbf{(9)}$$

2.9.3. *Sensitivity*

Sensitivity merupakan proporsi kasus positif yang diidentifikasi dengan benar (Powers, 2020). Untuk mendapatkan nilai *sensitivity* dapat dihitung dengan Persamaan 10.

$$\boldsymbol{Sn = \frac{TP}{(TP+FN)}} \quad \textbf{(10)}$$

2.9.4. F1-*Score*

F1 score merupakan *harmonic mean* dari *precision* dan *recall*. Untuk mendapatkan nilai *f1-score* dapat dihitung dengan Persamaan 11.

$$F_1 = 2\,x\,\frac{Precision\ x\ Recall}{Precision+Recall} \quad \text{..... (11)}$$

2.9.5. *Matthew Correlation Coefficient* (MCC)

Matthew Correlation Coefficient (MCC) digunakan untuk menilai kinerja prediksi struktr sekunder protein. MCC diperkenalkan oleh B.W. Matthews. Untuk mendapatkan nilai *MCC* dapat dihitung dengan Persamaan 12.

$$MCC = \frac{TP\ x\ TN - FP\ x\ FN}{\sqrt{(TP+FN\ x\ (TN+FP)\ x\ (TP+FP)\ x\ (TN+FN)}} \quad \text{..... (12)}$$

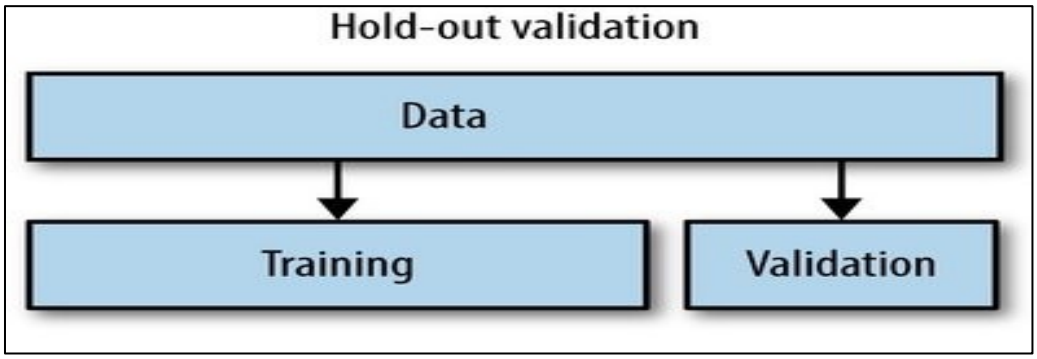
2.10. *Jackknife testing*

Dengan seperangkat metrik yang terdefinisi dengan baik untuk mengukur kualitas prediktor, hal berikutnya yang diperhatikan adalah jenis metode validasi yang digunakan untuk menilai metrik ini. Untuk mendapatkan nilai parameter terbaik untuk pengklasifikasi atau prediktor, tiga metode yang paling sering digunakan adalah *k-fold cross-validation* (subsampling), *independent dataset test*, dan *jackknife testing* (Chen Chou & Ting Zhang, 1995). *Cross validation* merupakan metode resampling yang banyak digunakan untuk mengevaluasi kinerja dan mencegah *overfitting*. Tes *jackknife* telah dikenal secara luas dan semakin banyak digunakan oleh para peneliti untuk memeriksa kualitas berbagai prediktor. Uji *jackknife* dianggap sebagai metode yang paling efektif untuk validasi silang dalam statistik (Chou, 2000).

Pengujian *Jackknife* adalah salah satu teknik *re-sampling* yang paling umum digunakan dan matang. Dalam teknik *sub-sampling* seperti itu, seleksi yang sangat kecil digunakan untuk pengujian dan seleksi yang berbeda dapat menghasilkan hasil yang sama sekali berbeda (Akmal et al., 2017). Tes *jackknife* selalu dapat menghasilkan hasil yang baik untuk kumpulan data

benchmark yang diberikan dan telah banyak digunakan dalam bioinformatika (Lin et al., 2014).

Cara kerja metode *jackknife* sama dengan *hold-out. Hold-out* adalah pembagian dataset menjadi set *'train'* dan *'test'*. Set pelatihan adalah tempat model dilatih, dan set pengujian digunakan untuk melihat seberapa baik model tersebut bekerja pada data yang tidak terlihat.

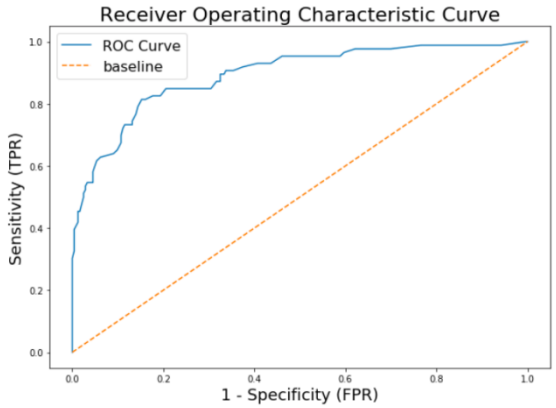


Gambar 8. Proses *Hold-out validation.*

Gambar 8 merupakan ilustrasi dari *hold-out validation.* Dataset dibagi menjadi dua, yaitu *data training* dan *testing*. *Hold-out validation* hanya melakukan sekali pembagian data. Komposisi umum yang digunakan adalah 80% *training* 20% *testing* dan 90% *training* 10% *testing*.

2.11. *Receiver Operating Characteristics (ROC)*

ROC adalah cara untuk memvisualisasikan, mengatur, dan memilih klasifikasi berdasarkan kinerjanya. ROC mengekspresikan *confusion matrix* dan merupakan grafik 2 dimensi dimana *True Positive Rate (sensitivity)* adalah sumbu Y dan *False Positive Rate (specificity)* adalah sumbu X. Area yang berada di bawah kurva merupakan wilayah yang menunjukkan tingkat keakuratan dari model predikasi dan dihitung dengan metode yang disebut *Area Under Curve* (AUC). Nilai AUC berada di antara 0 dan 1.



Gambar 9. Contoh Grafik ROC.

Gambar 9 memperlihatkan contoh grafik dari ROC. Kurva ROC dari model klasifikasi yang dibentuk akan semakin baik ketika semakin ke atas dari garis diagonal, sebaliknya jika semakin ke bawah dari garis diagonal maka artinya model klasifikasi tersebut tidak baik. Hubungan nilai TPR dan FPR saling terikat satu sama lain, apabila terjadi terjadi peningkatan pada TPR maka FPR akan mengalami penurunan dan sebaliknya.

III. DATA DAN METODOLOGI

3.1. Tempat dan Waktu

3.1.1. Tempat Penelitian

Penelitian dilakukan di Lab Rekayasa Perangkat Lunak (RPL), Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung.

3.1.2. Waktu dan Jadwal Penelitian

Penelitian ini dilakukan mulai dari bulan November 2021 hingga penyelesaian pada pertengahan tahun 2022. Alur pengerjaan dapat dilihat pada Tabel 10.

Tabel 10. Alur Waktu Pengerjaan

No	Kegiatan	2021												2022																			
		November				Desember				Januari				Februari				Maret				April				Mei				Juni				Juli			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Penentuan tema dan pengumpulan studi literatur																																				
2	Perancangan metode																																				
3	Pengumpulan *dataset*																																				
4	Penyusunan *draft* proposal (BAB I - III)																																				
5	*Feature extraction*																																				
6	*Jackknife testing*																																				
7	SMOTE																																				
8	Pemodelan dan prediksi XGBoost																																				
9	Evaluasi *confusion matrix*																																				
10	Penyusunan draft proposal (BAB IV - V)																																				

Tabel 10 menunjukkan rencana kegiatan penelitian yang dimulai pada bulan November 2021 sampai bulan Juli 2022. Secara umum tahapan penelitian ini dibagi menjadi tiga, yaitu

1. Perancangan Penelitian

   Penelitian ini dimulai dengan perancangan metode dan pengumpulan data. Data pada penelitian ini bersumber dari penelitian terdahulu oleh Qiu et al. (2016). Dataset penelitian dapat diakses di https://www.mdpi.com/1422-0067/15/5/7594#supplementary. Terdapat dua dataset yang digunakan yaitu prolin dan lisin. Dataset prolin terdiri dari 3505 data negatif dan 851 data positif dengan jumlah total 4356 data. Dataset lisin terdiri dari 980 data negatif dan 142 data positif dengan jumlah total 1122 data. Kemudian dilanjutkan dengan *preprocessing* melalui fitur ekstraksi.

2. Pelaksanaan Penelitian

   Pada tahap ini dilakukan *feature exctraction*. *Feature exctraction* yang digunakan adalah PseAAC, CTD, dan AAindex. Berdasarkan data yang digunakan, jumlah sampel postif dan negatif tidak seimbang. Untuk mengatasi hal tersebut digunakan teknik sampling untuk menyeimbangkan dataset yaitu dengan metode SMOTE. Selanjutnya dilakukan pemodelan dan prediksi menggunakan metode XGBoost.

3. Evaluasi Penelitian

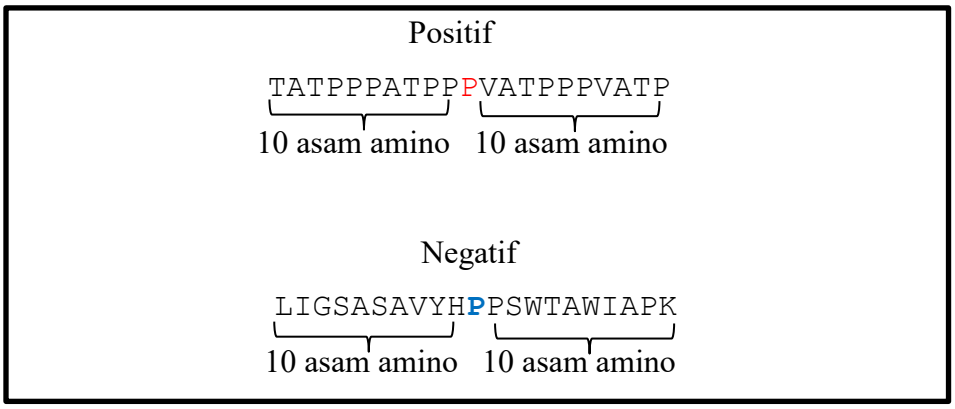
   Pada tahapan ini dilakukan evaluasi menggunakan confusion matrix dengan parameter Sn *(sensitivity)*, Spe *(Specificity)*, ACC *(accuracy), F1-score,* dan MCC (*Matthew Correlation Coefficient)*.

3.2. Data dan Alat

3.2.1. Data

Data yang digunakan berasal dari dataset *benchmark* dari penelitian Qiu, et al. (2016) dan dapat diakses melalui link https://www.mdpi.com/1422-0067/15/5/7594#supplementary. Dataset ini berisi

sampel positif dan sampel negatif yang di esktraksi dari hidroksiprolin dan hidroksilisin yang berjumlah 21 asam amino. Data prolin dan lisin yang digunakan masing-masing berjumlah 4356 dan 1122 data. Ilustrasi data *sequence* dapat dilihat pada Gambar 10.



Gambar 10. Ilustrasi data *sequence* protein.

Rincian data hidroksilasi yang digunakan pada penelitian ini dapat dilihat pada Tabel 11.

Tabel 11. Data Hidroksilasi (Qiu et al., 2016)

Jenis Data	Positif	Negatif
Prolin (P)	851	3505
Lisin (K)	142	980

3.2.2. Alat

Alat yang digunakan dalam penelitian ini terdiri dari *hardware* dan *software.*

3.2.2.1. *Hardware*

a) *Processor* : Intel® Core(TM) i5-8265U CPU @ 1.60GHz to 1.80 GHz (6M Cache, up to 3.9 GHz, 4 cores)

b) RAM : 8.00 GB, LPDDR3, 2133 MHz
c) *Storage* : SSD M.2 NVMe 256 GB
d) *Network Interface* : Intel® Wireless-AC 9560 160MHz
e) *Video Graphics Array* (VGA): Intel® UHD Graphics 620

3.2.2.2. *Software*

a) *Operating System* : Windows 10 Home Single Language 64-bit
b) *Tools* : Rstudio 1.3.1093 & R *programming* 3.6.3
c) *Library*

Beberapa *library* yang digunakan sebagai berikut.

1. *Library caret 6.0-86*

   Library caret merupakan *package* yang digunakan untuk klasifikasi dan regresi. *Package caret* berfungsi untuk melakukan seluruh proses pemodelan mulai dari *pre-pocessing* hingga evaluasi model.

2. *Library BioSeqClass 1.45.0*

   Library BioSeqClass digunakan untuk klasifikasi data berdasarkan urutan biologis.

3. *Package xgboost 1.5.0.2*

   Package ini digunakan untuk pemodelan xgboost dan dapat secara otomatis melakukan komputasi parallel pada satu mesin yang bisa 10 kali lebih cepat dari *package gradient boosting* lain.

4. *Library mccr 0.4.4*

   Library mccr digunakan untuk menghitung nilai *Matthew Correlation Coefficient* (MCC).

5. *Library smotefamily 1.3.1*

   Library smotefamily digunakan untuk mengatasi imbalance data dengan teknik SMOTE.

6. *Library dplyr 1.0.8*

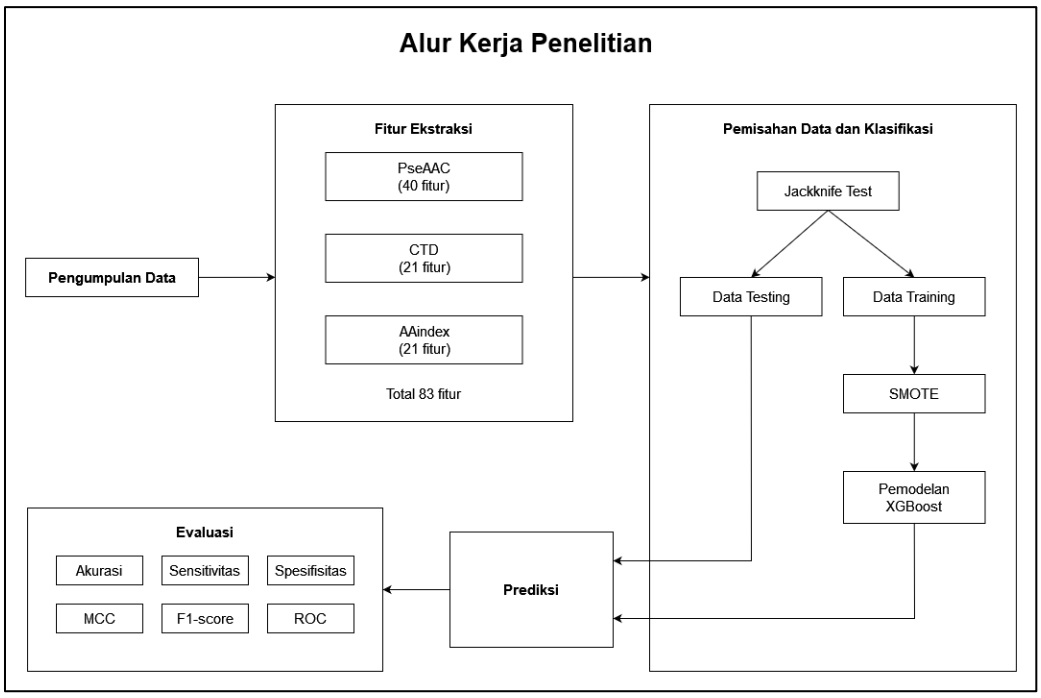
   Library ini bekerja pada data *frame. Library dplyr* digunakan untuk memanipulasi kumpulan data secara efisien di R.

7. *Library ROCR 1.0-11*

   Library ini digunakan untuk membuat kurva ROC.

3.3. Metodologi

Alur kerja penelitian ini didasari oleh penelitian Qiu, et al. (2016). *Flowchart* rencana penelitian dapat dilihat pada Gambar 11.



Gambar 11. *Flowchart* penelitian prediksi situs hidroksilasi.

Secara detail, proses rencana kerja penelitian pada Gambar 11 dapat dijelaskan sebagai berikut.

3.3.1. Pengumpulan Data

Data yang digunakan pada penelitian ini didapat dari penelitian (Qiu et al., 2016) yang terdiri dari *dataset* prolin dan lisin yang berjumlah 851 sampel positif dan 3505 negatif pada situs prolin dan 142 sampel positif dan 980 sampel negatif pada situs lisin. Data tersebut akan disimpan dalam format .csv.

3.3.2. Ekstraksi Fitur

Fitur esktraksi yang digunakan adalah *Pseudo Amino Acid Composition* (PseAAC), *Composition, Transition, and Distribution* (CTD), AAIndex. Fitur ekstraksi digunakan untuk mengubah data string menjadi data numerik.

3.3.3. Pemisahan Data dan Klasifikasi

Setelah tahap *feature extraction* dilakukan, selanjutnya dilakukan pemisahan data dengan membagi 20% data *testing* 80% data *training* 10% data *testing* 90% data *training*. Selanjutnya dilakukan klasifikasi menggunakan metode XGBoost. Pemodelan klasifikasi pada penelitian ini menggunakan *Extreme Gradiant Boosting* (XGBoost). Selanjutnya dilakukan validasi model menggunakan *jackknife testing. Jackknife testing* pada penelitian ini digunakan untuk menilai hasil metrik.

3.3.4. Prediksi

Pada tahap prediksi digunakan model klasifikasi yang sudah dilatih yang dimana inputnya adalah data uji. Setelah mendapat hasil prediksi label, selanjutnya dibandingkan dengan label yang sebenarnya.

3.3.5. Evaluasi Model

Tahap evaluasi ini menghasilkan *confusion matrix* dengan hasil nilai akurasi, sensitifitas, spesifisitas, *F-1 score*, MCC, dan ROC.

V. PENUTUP

5.1. Simpulan

Dari penelitian yang telah dilakukan tentang klasifikasi situs hidroksilasi protein pada prolin dan lisin menggunakan metode *Extreme Gradient Boosting* dapat diambil kesimpulan sebagai berikut.

1. Setelah dilakukan perbandingan dengan penelitian sebelumnya, hasil penelitian yang dilakukan oleh (Qiu et al., 2016) dengan menggunakan metode Random Forest memiliki akurasi 1.32% lebih rendah daripada XGBoost dengan penerapan SMOTE pada dataset prolin dan akurasi 2.02% lebih rendah pada dataset lisin.
2. a) Pada dataset 1 tingkat akurasi dengan metode XGBoost dengan mendapatkan hasil sebesar 93.9% pada skenario pertama, setelah diterapkan SMOTE hasil akurasi naik menjadi 95.8%. Sedangkan pada skenario kedua, hasil akurasi tertinggi adalah 97.9%.
   b) Pada dataset 2 hasil akurasi pada skenario pertama sebesar 94.6%, setelah diimplementasikan SMOTE hasil akurasi naik menjadi 95.5%. Sedangkan pada skenario kedua setelah diterapkan SMOTE hasil akurasi sebesar 99.1%.

5.2. Saran

Dalam penelitian ini saya dibatasi dengan waktu. Sehingga adapun saran yang saya berikan pada penelitian ini adalah sebagai berikut.

1. Penelitian ini dapat menggunakan metode *machine learning* lain, seperti SVM, Random Forest, atau K-*Nearest neighbors* (KNN) untuk mendapatkan perbandingan hasil yang digunakan pada penelitian ini.
2. Penelitian ini dapat menggunakan teknik lain dalam menangani *imbalance* data, seperti metode *hybrid*, ADASYN, atau *undersampling*.

DAFTAR PUSTAKA

Akmal, M. A., Rasool, N., & Khan, Y. D. 2017. Prediction of N-linked glycosylation sites using position relative features and statistical moments. *PLoS ONE*, *12*(8), 1–21.

Alajali, W., Zhou, W., Wen, S., & Wang, Y. 2018. Intersection traffic prediction using decision tree models. *Symmetry*, *10*(9), 1–16.

Bahri, R. S., & Maliki, I. 2012. PERBANDINGAN ALGORITMA TEMPLATE MATCHING DAN FEATURE EXTRACTION PADA OPTICAL CHARACTER RECOGNITION. *Jurnal Komputer Dan Informatika (KOMPUTA) 2012*, *1*(1), 187–198.

Basu, S., & Plewczynski, D. 2010. AMS 3.0: prediction of post-translational modifications. *BMC Bioinformatics*.

Cai, Y. D., & Lu, L. 2008. Predicting N-terminal acetylation based on feature selection method. *Biochemical and Biophysical Research Communications*, *372*(4), 862–865.

Chawla, N. V, Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. 2002. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*.

Chen Chou, K., & Ting Zhang, C. 1995. Prediction of protein structural classes. *Biochimie*, *82*(8), 783–785.

Chen, T., & Guestrin, C. 2016. XGBoost: A scalable tree boosting system. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, *13-17-Augu*, 785–794.

Chou, K. C. 2000. Prediction of protein signal sequences and their cleavage sites. *Proteins: Structure, Function and Genetics*, *42*(1), 136–139.

Chou, K. C. 2001. Prediction of protein cellular attributes using pseudo-amino acid composition. *Proteins: Structure, Function and Genetics*, *43*(3), 246–255.

Diana, F. M. 2009. Fungsi dan Metabolisme Protein dalam Tubuh Manusia. *Jurnal Kesehatan Masyarakat*, *4*(1), 49.

Du, P., Wang, X., Xu, C., & Gao, Y. 2012. PseAAC-Builder: A cross-platform stand-alone program for generating various special Chou's pseudo-amino acid compositions. *Analytical Biochemistry*, *425*(2), 117–119.

Dubchak, I., Muchnik, I., Holbrook, S. R., & Kim, S. H. 1995. Prediction of protein folding class using global description of amino acid sequence. *Proceedings of the National Academy of Sciences of the United States of America*, *92*(19), 8700–8704.

Friedman, J. H. 2001. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, *29*(5), 1189–1232.

Huang, Q., Chen, X., Wang, Y., Li, J., Liu, H., Xie, Y., Dai, Z., Zou, X., & Li, Z. 2020. HydLoc: A tool for hydroxyproline and hydroxylysine sites prediction in the human proteome. *Chemometrics and Intelligent Laboratory Systems*, *202*(May), 104035.

Lin, H., Deng, E. Z., Ding, H., Chen, W., & Chou, K. C. 2014. IPro54-PseKNC: A sequence-based predictor for identifying sigma-54 promoters in prokaryote with pseudo k-tuple nucleotide composition. *Nucleic Acids Research*, *42*(21), 12961–12972.

Long, H., Liao, B., Xu, X., & Yang, J. 2018. A hybrid deep learning model for predicting protein hydroxylation sites. *International Journal of Molecular Sciences*, *19*(9).

Manavalan, B., Shin, T. H., & Lee, G. 2018. PVP-SVM: Sequence-based prediction of phage virion proteins using a support vector machine. *Frontiers in Microbiology*, *9*(MAR), 1–10.

Mathew, J., Luo, M., Pang, C. K., & Chan, H. L. 2015. *Kernel-Based SMOTE for SVM Classification of Imbalanced Datasets*. 1127–1132.

Minguez, P., Letunic, I., Parca, L., & Bork, P. 2013. PTMcode: A database of known and predicted functional associations between post-translational modifications in proteins. *Nucleic Acids Research*, *41*(D1), 306–311.

Mishra, S. 2017. Handling Imbalanced Data : SMOTE vs . Random Undersampling. *International Research Journal of Engineering and Technology (IRJET)*, *04*(08).

Powers, D. M. W. 2020. *Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation*. *May*.

Qiu, W. R., Sun, B. Q., Xiao, X., Xu, Z. C., & Chou, K. C. 2016. IHyd-PseCp: Identify hydroxyproline and hydroxylysine in proteins by incorporating

sequence-coupled effects into general PseAAC. *Oncotarget*, *7*(28), 44310–44321.

Ramyachitra, D., & Manikandan, P. 2014. *IMBALANCED DATASET CLASSIFICATION AND SOLUTIONS : A REVIEW*. *5*(4).

Rosmawati, R. 2013. LAMA PEREBUSAN TERHADAP KANDUNGAN PROTEIN PADA KERANG DARAH (Anadara granosa). *Biosel: Biology Science and Education*, *2*(2), 103.

Sagi, O., & Rokach, L. 2021. Approximating XGBoost with an interpretable decision tree. *Information Sciences*, *572*, 522–542.

Siringoringo, R. 2018. Klasifikasi Data Tidak Seimbang Menggunakan Algoritma SMOTE dan k-Nearest Neighbor. *Jurnal ISD*, *3*(1), 44–49.

Xu, Y., Wen, X., Shao, X. J., Deng, N. Y., & Chou, K. C. 2014. iHyd-PseAAC: Predicting hydroxyproline and hydroxylysine in proteins by incorporating dipeptide position-specific propensity into pseudo amino acid composition. *International Journal of Molecular Sciences*, *15*(5), 7594–7610.

You, Z., Chan, K. C. C., & Hu, P. 2015. *Predicting Protein-Protein Interactions from Primary Protein Sequences Using a Novel Multi-Scale Local Feature Representation Scheme and the Random Forest*. 1–19.