**JURNAL INFORMATIKA**, Vol. 9 No. 2 Oktober 2022, Halaman 172~179
ISSN: 2355-6579 | E-ISSN**:** 2528-2247

# Perbandingan Algoritma *Support Vector Machine* dan *KNN* dalam Memprediksi Struktur Sekunder Protein

**Anggi Tasari[1], Dewan Dinata Tarigan[2], Erika Nia Devina Br Purba[3], Kana Saputra S[4]**

[1,2,3,4] Program Studi Ilmu Komputer, FMIPA, Universitas Negeri Medan
Jalan Williem Iskandar Pasar V, Sumatera Utara, Indonesia

e-mail: [1]sarianggita934@gmail.com, [2]dinatadewan@mhs.unimed.ac.id, [3]niapurbaa@gmail.com, [4]kanasaputras@unimed.ac.id

Informasi Artikel Diterima: 24-06-2022 Direvisi: 03-10-2022 Disetujui: 09-10-2022

**Abstrak**

Pendekatan biologi komputasi telah maju secara *eksponensial* dalam prediksi struktur sekunder protein yang sangat penting untuk industri farmasi. Ekstraksi fitur protein di dalam laboratorium memiliki informasi yang cukup untuk prediksi struktur sekunder protein yang digunakan dalam studi bioinformatika. Memprediksi struktur sekunder protein merupakan suatu permasalahan yang terdapat dalam bidang Bioinformatika. Terdapat beberapa metode yang telah diterapkan dengan tingkat akurasi yang dihasilkan berbeda-beda. Penelitian ini bertujuan untuk membandingkan model prediksi *Support Vector Machine* dengan *K-Nearest Neighbor* dalam memprediksi struktur sekunder protein. Dalam penelitian ini, model *Support Vector Machine* dan *K-Nearest Neighbor* disajikan dalam dataset RS126 yang terdiri dari 126 data protein dengan panjang urutan protein rata-rata 185 sekuens. Data RS126 juga terdiri atas 32% *alpha helix (H)*, 21% *beta (E)*, dan 47% *coil (C)*. Masing-masing model prediksi pada penelitian ini diberikan nilai lebar *sliding window* sebesar 15. Nilai K = 5, K=10, dan K= 15 untuk model prediksi KNN serta Nilai C = 1, *Gamma* = 0,1 dan Kernel *Radial Basis Function* untuk model prediksi SVM. Penggunaan model *Support Vector Machine* dan *K-Nearest Neighbor* digunakan untuk memperoleh hasil yang relavan serta akurat dalam prediksi struktur sekunder. Beberapa prinsip yang diusulkan memiliki klarifikasi biologis yang menarik dan relevan. Hasil yang diperoleh menegaskan bahwa keberadaan asam amino tertentu dalam urutan protein meningkatkan stabilitas untuk prakiraan stuktur sekunder protein. Dalam penelitian ini algoritma KNN memiliki performa yang lebih baik dalam memprediksi struktur sekunder protein dibandingkan dengan algoritm SVM.

**Kata Kunci** : Struktur Sekunder Protein; *Support Vector Machine*; Ekstraksi Fitur

***Abstract***

*Computational biology approaches have advanced exponentially in the prediction of the secondary structure of proteins of great importance to the pharmaceutical industry. The extraction of protein features in the laboratory has sufficient information for the prediction of the secondary structure of proteins used in bioinformatics studies. Predicting the secondary structure of proteins is a problem in the field of bioinformatics. There are several methods that have been applied with different levels of accuracy produced. This study aims to compare the Support Vector Machine prediction model with K-Nearest Neighbor in predicting the secondary structure of proteins. In this study, the Support Vector Machine and K-Nearest Neighbor models are presented in the RS126 dataset which consists of 126 protein data with an average protein sequence length of 185 sequences. RS126 data also consists of 32% alpha helix (H) , 21% beta (E), and 47% coil (C). Each prediction model in this study is given a sliding window width value of 15. The value of K = 5, K = 10, and K = 15 for the KNN prediction model and the value of C = 1, Gamma = 0.1 and Kernel Radial Basis Function for SVM prediction model. The use of Support Vector Machine and K-Nearest Neighbor models are used to obtain relevant and accurate results in secondary structure prediction. Some of the proposed principles have interesting and relevant biological clarifications. The obtained results confirm that the presence of certain amino acids in the protein sequence increases the stability for the predicted secondary structure of the protein. In this study, the KNN algorithm has a better performance in predicting the secondary structure of proteins than the SVM algorithm.*

***Keywords :*** *Protein Secondary Structure; Support Vector Machine; Feature Extraction*

## 1. Pendahuluan

Protein merupakan bahan pembentuk dasar suatu struktur sel tubuh dan juga merupakan bagian terbesar kedua pada tubuh setelah air (Agustina et al., 2020). Fungsi protein adalah untuk membangun dan memperbaiki sel yang ada di dalam tubuh (somatik) serta dapat menghasilkan energi. Protein memiliki struktur yang sangat kompleks, terbentuk dari rangkaian asam amino, dan memiliki berbagai sifat dan karakter. Struktur protein dibagi menjadi empat struktur utama, yaitu struktur primer dimana struktur ini terdiri dari rankaian asam amino yang terbentuk dari ikatan peptida, kemudian struktur sekunder yang dimana struktur tersebut terbentuk dari rangkaian asam amino yang membentuk struktur melingkar, dan struktur tersier yang merupakan penggabungan hasil proses pelipatan (*folding*) dari struktur sekunder yang berbeda (Haryanto & Surya, 2015). Setelah rangkaian asam amino terlipat dalam 3D, protein memiliki peran berbeda seperti struktur sebelumnya yaitu struktur primer dan sekunder yang akan menentukan struktur tersier. Oleh karena itu, menentukan struktur sekunder protein merupakan bidang yang sangat penting dalam bioinformatika.

Memprediksi struktur sekunder protein dilakukan berdasarkan urutan asam amino. Dalam bioinformatika dan kimia, memprediksi struktur sekunder protein sangat penting dalam bidang kedokteran dan bioteknologi, salah satunya adalah desain obat dan enzim baru. Prediksi struktur sekunder yang akurat dapat menaikkan tingkat akurasi prediksi pada struktur tersier protein, lantaran struktur protein bisa memilih sifat struktur protein fragmen lokal (Zhou et al., 2018). Untuk memperoleh suatu struktur protein, maka dapat dilakukan eksperimen. Dengan berkembangnya teknologi komputasi, struktur protein dapat diperoleh dengan membangun model prediktif yang dapat digunakan untuk memprediksi struktur sekunder protein menggunakan SVM (*Support Vector Machine*).

SVM (*Support Vector Machine*) adalah algoritma klasfikasi yang memiliki sifat *supervised learning (pembelajaran)* yang bekerja dengan mencari *hyperplane* (batas keputusan) optimal yang memisahkan jarak antar kelas (Wulandari, 2020). SVM termasuk ke dalam 10 algoritma terbaik. Dibandingkan dengan algoritma lain, SVM menunjukkan ketahanan yang lebih tinggu, generalisasi dan akurasi klasifikasi yang stabil. SVM merupakan metode perhitungan terbaik untuk mendapatkan hasil klasifikasi dengan tingkat akurasi yang tinggi (Prakash & Singh, 2015). SVM memiliki akurasi yang tinggi dan bekerja sangat baik dengan dataset yang terbatas. Akurasi merupakan rasio prediksi positif dan negatif yang benar untuk seluruh data (Huang & Chen, 2013)

Penelitian terdahulu terkait prediksi struktur sekunder protein menggunakan *Support Vector Machine* (SVM) dilakukan oleh Yin Fai Chin, dimana ia menggunakan dataset RS126 dengan hasil akurasi sebesar 43.0 (Fai Chin et al., 2012). Penelitian lebih lanjut mengenai ekstraksi ciri fisikokimia untuk memprediksi struktur sekunder ptotein (*Extracting Physicochemical Features to Predict Protein Secondary Structure)* dilakukan oleh Huang dan Cheb, dengan menggunakan model SVM sebagai *classifier* dan menerapkan ekstraksi ciri berupa teknik *sliding windows* dan metode fisikokimia (*physicochemical)*, dengan memperoleh hasil akurasi Q3. Akurasi meningkat sebesar 77.40% hingga 79.52% (Huang & Chen, 2013).

Metode *K-Nearest Neighbors (KNN)* merupakan algoritma yang dapat menghasilkan klasifikasi baru berdasarkan sebagian besar kategori tetangga terdekat. Tujuan dari algoritma ini adalah untuk mengklasifikasikan obyek baru berdasarkan atribut dan data latih atau data *training* (Sutrimo & Wismarini, 2022). Sampai saat ini, belum ada penelitian yang membahas keakuratan metode *K-Nearest Neighbors* dalam memprediksi struktur sekunder protein.

Berdasarkan latar belakang dan sedikit pembahasan dari penelitian yang dilakukan oleh beberapa peneliti sebelumnya, peneliti mengusulkan untuk melakukan penelitian yang akan memprediksi struktur sekunder protein menggunakan penerapan dari model *support vector machine* (SVM) *serta K-Nearest Neighbors* (KNN). Penelitian ini kemudian menentukan nilai *sliding window*, nilai K, nilai C, *gamma*, dan kernel yang optimal untuk mendapatkan akurasi yang sesuai dan tepat.

## 2. Metode Penelitian

### 2.1 Dataset

Dalam penelitian ini menggunakan dataset RS126 yang diambil dari situs https://raw.githubusercontent.com/JiayingLi/Protein-Secondary-Structure-Prediction-using-Convolution-Neural-Network/master/RS126.data. Dataset tersebut berisikan kumpulan data dari pasangan struktur primer protein beserta struktur sekundernya. Dataset RS126 terdiri dari 126 data protein yang memiliki panjang urutan protein rata-rata 185 asam amino. Dengan

32% dari RS126 adalah *alpha helix (H)*, 21% adalah *beta (E)*, dan 47% adalah *coil (C)*.

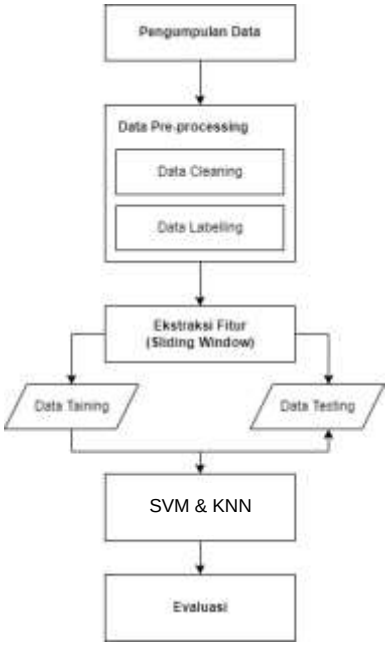
Berikut ilustrasi contoh dari data yang digunakan dalam penelitian ini, data yang diambil sebagai contoh adalah data struktur primer dan sekunder dengan panjang sekuens sama panjang yaitu 108 asam amino. Dataset yang digunakan juga memiliki fitur sebanyak 22.594 data. Contoh dataset ditunjukkan dalam Tabel 1.

Tabel 1. *Contoh Dataset*

| ***Structure*** | ***Sequence*** |
|---|---|
| *Primary* | APAFSVSPASGASDGQSVS VSVAAAGETYYIAQCAPVGG QDACNPATATSFTTDASGA ASFSFTVRKSYAGQTPSGTP VGSVDCATDACNLGAGNSG LNLGHVALTFG |
| *Secondary* | CCEEEEECCCCCCCCCEEE EEEECCCCEEEEEEECEEC CEECCCCCCCCCEEECCCCC CCEEEEECCCEEEEECCCC CEEEEEECCCCCCEEEEEC CCCCCCCCCCCCC |

## 2.2 Tahapan Penelitian

Penelitian ini dimulai dengan mengumpulkan data dan dilanjutkan dengan data *preprocessing* atau praproses data, yang terdiri dari data *cleaning* atau pembersihan data dan data *labelling* atau pelabelan data. Selain itu, ekstraksi fitur dilakukan dengan menggunakan teknik sliding window, sedangkan data pelatihan dan pengujian digunakan untuk melatih dan menguji model menggunakan data *training* dan data *testing*. Kemudian dilakukan evaluasi hasil prediksi dengan membandingkan nilai akurasi algoritma *Support Vector Machine* dan *K-Nearest Neighbor* yang berfungsi untuk memvisualisasikan kinerja model dengan melakukan perhitungan akurasi (Khairul Anam et al., 2022). Gambar 1 menunjukkan tahapan proses penelitian.




Gambar 1. *Tahapan Penelitian*

## 2.3 *Preprocessing* Data

*Preprocessing* data merupakan langkah awal dalam proses penelitian. Dalam tahap *preprocessing* ini dilakukan pembersihan (*cleaning*) dan pelabelan data (*labelling*) untuk selanjutnya dilakukan proses *data cross-validation* dalam menentukan besaran data latih dan data uji, hingga akhirnya dimasukkan ke dalam model algoritma *Support Vector Maching (SVM)* dan *K-nearest Neighbor (KNN)*.

## 2.4 *Support Vector Machine*

Support Vector Machine (SVM) merupakan salah satu metode pembagian atau klasifikasi yang bersifat terstruktur pada data mining. Support Vector Machine (SVM) juga bisa melakukan prediksi baik dalam pembagian atau klasifikasi yang bersifat terstruktur dan juga regresi (Budi, 2007). Cara kerja Support Vector Machine (SVM) adalah mengatasi masalah non-linier dengan cara menggabungkan konsep kernel ke dalam ruang dimensi yang lebih tinggi. Pemisah jarak antar kelas (hyperplane) ini nantinya akan ditemukan dan diperoleh di ruang dimensional tersebut. Fungsi klasifikasi yang memiliki sifat terstruktur pada model Support Vector Machine (SVM) tersebut didefinisikan oleh rumus berikut :

$$f(x) = \text{sign}\ (w.x + b)$$

Jika nilai menurut $w.x + b > 0$ maka diklasifikasikan kedalam kelas +1 & apabila $w.x + b < 0$ data diklasifikasikan kedalam kelas -1. Nilai akurasi menurut contoh yg didapatkan menggunakan memakai Support Vector

Machine (SVM) sangat tergantung dalam fungsi kernel & nilai parameter yg dipakai. Parameter yg dipakai dalam prosedur pemecahan Support Vector Machine (SVM) merupakan parameter Cost (C) & *Gamma* (γ). Semakin besar nilai C akan membuat penalty yang besar juga terhadap proses klasifikasinya. Pada fungsi kernel RBF, parameter gamma (γ) dipakai untuk mentransformasikan data train ke ruang fitur yg lalu dioptimasi memakai metode Lagrange Multipliers, sehingga akibatnya membuat nilai α yg dipakai untuk memilih support vector & memperkirakan koefisien w (bobot) ataupun b (bias) dalam contoh klasifikasi (Agustina et al., 2020).

### 2.5 K-Nearest Neighbors

Algoritma KNN merupakan metode yang menggunakan algoritma Supervised (algoritma yang terawasi). Algoritma Supervised Learning (algoritma yang terawasi) dalam penerapan model KNN tersebut memiliki fungsi untuk menghasilkan atau memperoleh pola atau fitur yang baru pada data, serta ada juga algoritma Unsupervised Learning (algoritma pembelajaran tanpa pengawasan) yang juga memiliki fungsi untuk memperoleh pola atau fitur dari data yang diteliti (Krisandi et al., 2013). Cara kerja untuk penerapan metode K-nearest neighbor (KNN) tersebut adalah mencari jarak terpendek antara data yang dievaluasi dengan metode K-nearest neighbor pada data training. Data training (training data) diproyeksikan ke dalam ruang multidimensi, di mana setiap dimensi mewakili karakteristik data. Ruang panel tersebut akan dibagi menjadi beberapa bagian berdasarkan klasifikasi data pelatihan atau data training. Titik-titik yang ada di dalam ruang panel ini disebut kelas c jika kelas c merupakan klasifikasi yang paling sering ditemukan di dekat k-tetangga terdekat dari titik-titik tersebut (Whidhiasih et al., 2013).

### 2.6 RBF (Radial Basis Function)

Radial Basis Function (RBF) adalah fungsi inti yang sering digunakan dalam analisis ketika data tidak dapat dipisahkan secara linier. RBF memiliki dua parameter: gamma dan Cost (Pakuan Putra. D dan Agus Wardijono. B, 2020). Parameter cost atau biasa dikenal dengan parameter C adalah parameter yang berfungsi menjadi meningkatkan secara optimal SVM untuk menghindari kesalahan penjabaranklasifikasi dalam setiap sampel dataset latih. Parameter gamma bisa memilih seberapa jauh efek sampel dataset latih menggunakan nilai rendah yang berarti jauh & nilai yang tinggi berarti dekat (Heyran Byun dan Seong Whan Lee, 2003). Adapun fungsi atau rumus dari RBF dapat dilihat pada persamaan berikut (Felix et al., 2019) :

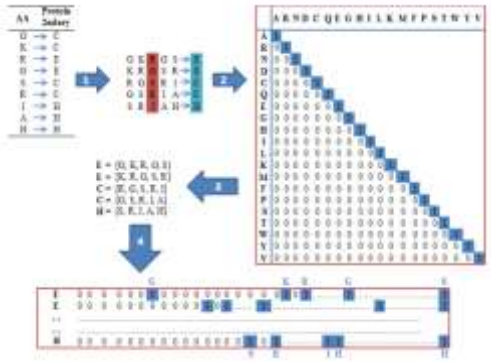
$$K(x_i, x) = exp(-y \mid\mid x_i - x \mid\mid^2, y > 0$$

### 2.7. Sliding Window

Ekstraksi fitur sliding window adalah sebuah ventilasi yang mempunyai ukuran piksel lebar & tinggi tertentu. Ukuran window itu sendiri bervariasi dari skala yang berbeda-beda, tetapi rasionya permanen sama. Window bergeser berdasarkan sudut kiri atas ke arah kanan lalu turun ke bawah & bergeser balik ke kanan & seterusnya sampai ke sudut kanan bawah yg menyerupai alfabet Z berulang. Pergeseran tersebut lalu disetting memakai nilai yang diperoleh berdasarkan output pengurangan ukuran piksel window dengan menggunakan ovelap antar window. Semakin besar nilai pergeserannya maka semakin kecil juga jumlah window yang dihasilkan (Indrabulan & Syarif, 2020).

Tahapan ekstraksi fitur sliding window dimulai dari memilih lebar window yang dipakai, yaitu 15, 17, 19. Tahap ke 2 dilakukan pengambilan data sebesar lebar window yg dipakai, selanjutnya dilakukan ekstraksi untuk setiap asam amino. Tahap terakhir yaitu membangun sebuah inputan. Input yang dipakai untuk proses training adalah panjang fitur output ekstraksi asam amino menggunakan penggunaan lebar sliding window (W) merupakan sebesar W * 20, dimana 20 merupakan ukuran Orthogonal Encoding berdasarkan struktur protein. Sehingga masih ada 300 fitur yg dipakai menjadi input.

*Sliding Window (*Jendela geser) adalah ukuran jendela yang digunakan sebagai pembuat pola untuk melihat urutan asam amino yang berdekatan*. Sliding Window* memiliki sepasang struktur sekunder, menggunakan urutan asam amino di tengah (titik tujuan) sebagai fokus utama (MARDIASIH & Haryanto, 2014).



Gambar 2. *Ilustrasi Sliding Window*

## 3. Hasil dan Pembahasan

### 3.1. Penerapan Algoritma KNN

Penelitian ini menggunakan Bahasa pemrograman *Python* dan *Jupyter Notebook* sebagai IDE *(Integrated Development Environment)* dalam mengolah serta menguji data RS126 sehingga menjadi informasi yang dapat dianalisa untuk membuat suatu kesimpulan.

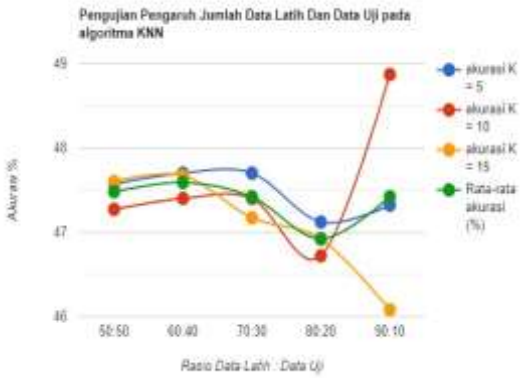
Pada penelitian ini, dilakukan pengujian dengan mengubah rasio masing-masing dari jumlah data latih dan data uji. Terdapat 22.594 data yang memiliki 2 kelas yang berbeda, yaitu kelas struktur primer yang berada diurutan ganjil dan kelas struktur sekunder yang berada pada urutan genap. Dataset tersebut kemudian dibagi menjadi beberapa rasio data latih dan data uji, yaitu diantaranya rasio 50%:50%, 60%:40%, 70%:30%, 80%:20%, dan 90%:10% dari total data yang ada pada dataset. Nilai K yang digunakan di dalam peneltian ini ada 3 yaitu K = 5, K = 10, dan K = 15. Untuk sliding window, lebar window yang digunakan adalah 15. Tabel 2 di bawah ini menunjukkan hasil akurasi pengujian pengaruh jumlah data latih dan data uji terhadap algoritma KNN.

Tabel 2. *Akurasi Pengujian Pengaruh Jumlah Data Latih dan Data Uji KNN*

| SW | % DL | % DU | Akurasi (%) | | | |
|---|---|---|---|---|---|---|
| | | | K = 5 | K = 10 | K = 15 | $\overline{X}$ |
| **15** | 50 | 50 | 47.5 | 47.2 | 47.6 | 47.4 |
| **15** | 60 | 40 | 47.7 | 47.4 | 47.6 | 47.5 |
| **15** | 70 | 30 | 47.7 | 47.4 | 47.1 | 47.4 |
| **15** | 80 | 20 | 47.1 | 46.7 | 46.9 | 46.9 |
| **15** | 90 | 10 | 47.3 | 48.8 | 46.0 | 47.4 |

Berdasarkan Tabel 2, SW merupakan Sliding Window, % DL adalah persentase data latih, dan % DU adalah persentase data uji. Terlihat bahwa hasil akurasi metode KNN pada tahap pengujian cukup berbeda dan berbeda karena pengaruh jumlah data latih dan data uji. Kita juga dapat melihat bahwa ada akurasi rata-rata 49,59% dan akurasi rata-rata 49,59%, yang menunjukkan nilai tertinggi ketika rasio data latih dan data uji adalah 60%:40%. Sedangkan untuk nilai terendah dapat dilihat pada saat rasio data latih dan data uji berada di rasio 80%:20% dengan rata-rata akurasi sebesar 46.92%. Hal ini dikarenakan semakin banyak data latih yang digunakan, maka akan semakin banyak pula data yang dibandingkan dengan semua data uji, sehingga nilai *CosSim* yang diperoleh adalah data yang memiliki *similarity* yang mirip dengan data uji yang terklasifikasi cenderung lebih tinggi untuk masuk ke dalam nilai ketetanggaannya, sehingga sistem akan dapat mengenali data yang lebih beragam dan bervariasi untuk dijadikan sebagai pembelajaran sistem. Begitu juga sebaliknya, apabila menggunakan data latih yang jumlahnya sedikit, maka semakin sedikit juga data yang dibandingkan dengan data uji dikarenakan jumlah data yang beragam sedikit.

Pengaruh jumlah data latih dan data uji menggunakan algoritma KNN ditunjukkan pada grafik yang ada pada Gambar 3 sebagai berikut:




Gambar 3. *Grafik Pengaruh Jumlah Data Latih dan Data Uji KNN*

### 3.2. Implementasi Algoritma SVM

Pembagian data latih dan data uji yang digunakan dalam tahap pengujian ini adalah 50%:50%, 60%:40%, 70%:30%, 80%:20%, dan 90%:10% dengan nilai K yang digunakan adalah K = 5, K = 10, dan K = 15. Kemudian sliding window yang digunakan memiliki lebar window sebesar 15. Hasil akurasi dari pengujian pengaruh jumlah data latih dan data uji pada algoritma SVM ditunjukkan pada Tabel 3.

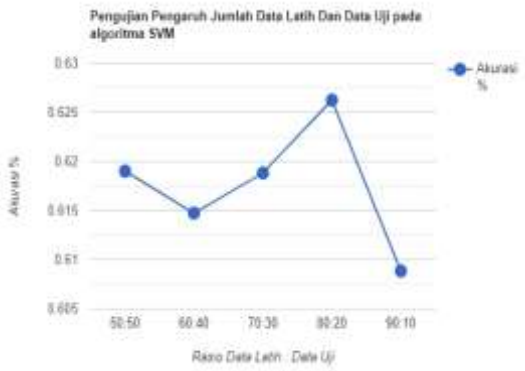
Tabel 3. *Akurasi Pengujian Pengaruh Jumlah Data Latih dan Data Uji SVM*

| % DL | % DU | Kernel | C | $\gamma$ | Akurasi (%) |
|---|---|---|---|---|---|
| **50** | 50 | RBF | 1 | 0,1 | 0.62 |
| **60** | 40 | RBF | 1 | 0,1 | 0.61 |
| **70** | 30 | RBF | 1 | 0,1 | 0.61 |
| **80** | 20 | RBF | 1 | 0,1 | 0.62 |
| **90** | 10 | RBF | 1 | 0,1 | 0.60 |

Berdasarkan Tabel 3 di atas, dapat dilihat bahwa hasil akurasi yang dihasilkan oleh metode SVM untuk tahap pengujian dalam pengaruh jumlah data latih dan data uji cukup beragam. Dalam penelitian ini dilakukan pengujian kernel, nilai C, dan *gamma*. Kernel yang digunakan adalah kernel RBF. Nilai C yang akan diuji yaitu 1 dan nilai *gamma* yang digunakan adalah 0.1. Hasil pengujian menunjukkan bahwa perbedaan pada perbandingan jumlah data latih dan data uji memiliki pengaruh pada nilai akurasi. Pada perbandingan rasio 50%:50% diperoleh rata-rata akurasi sebesar 0.62. Pada perbandingan dengan rasio 60%:40% diperoleh rata-rata akurasi sebesar 0.61. Pada perbandingan dengan rasio 70%:30% diperoleh rata-rata akurasi sebesar 0.61. Pada perbandingan dengan rasio 80%:20% diperoleh rata-rata akurasi sebesar 0.62. Pada perbandingan dengan rasio 90%:10% diperoleh rata-rata akurasi sebesar 0.60.

Dapat dilihat juga bahwa rata-rata akurasi yang menghasilkan nilai tertinggi ketika rasio data latih dan data uji berada di rasio 50%:50%, dengan rata-rata akurasi menunjukkan nilai sebesar 0.62 Sedangkan untuk nilai terendah dapat dilihat pada saat rasio data latih dan data uji berada di rasio 90%:10% dengan rata-rata akurasi sebesar 0.60. Hal ini dikarenakan semakin kecil nilai C semakin tidak toleran terhadap kesalahan sehingga akurasi yang diperoleh semakin baik. Selain itu semakin tinggi nilai gamma yang digunakan untuk tahap pengujian maka semakin tinggi pula nilai akurasi yang diperoleh. Begitu sebaliknya, semakin rendah nilai gamma akan membuat akurasi semakin rendah.

Hasil pengujian pengaruh jumlah data latih dan data uji pada algoritma SVM dapat dilihat pada grafik yang ada pada Gambar 4.




Gambar 4. *Grafik Pengaruh Jumlah Data Latih dan Data Uji SVM*

### 3.3. Perbandingan Algoritma KNN dan SVM

Pengujian perbandingan metode SVM dengan KNN dilakukan untuk mengetahui metode manakah yang memiliki tingkat akurasi terbaik dalam memprediksi struktur sekunder protein. Setelah mendapatkan nilai akurasi dari algoritma *Support Vector Machine* dan *K-Nearest Neighbor*, maka selanjutnya dilakukan perbandingan dengan menormalisasi nilai-nilai akurasi tersebut pada tiap rasio data latih dan data uji algoritma SVM dan KNN. Pada algoritma KNN, nilai akurasi yang diambil adalah nilai rata-rata akurasi pada setiap rasio dengan nilai K yang berbeda. Nilai hasil akurasi yang sudah dinormaliasasi dari kedua algoritma tersebut dapat dilihat pada tabel 4 berikut.

Tabel 4. *Normalisasi Hasil Nilai Akurasi Algoritma SVM & KNN*

| Rasio DL : DU | SVM | KNN |
|---|---|---|
| **50:50** | 0.45302566 | 0.4479651 |
| **60:40** | 0.44571879 | 0.44891018 |
| **70:30** | 0.44571879 | 0.4479651 |
| **80:20** | 0.45302566 | 0.44323973 |
| **90:10** | 0.43841193 | 0.4479651 |

Hasil Normalisasi nilai akurasi dari pengujian metode SVM dengan KNN dapat dilihat melalui grafik yang ditunjukkan pada gambar 5.


Gambar 5. *Grafik Pengujian Perbandingan Metode SVM dengan KNN*

Pada grafik pengujian diatas terlihat jelas bahwa KNN memiliki nilai akurasi yang lebih baik dibandingkan dengan SVM saat menguji data dengan rasio data 90:10, yakni dengan nilai akurasi 0.4479651. Namun, SVM juga terlihat menghasilkan nilai akurasi yang baik Ketika menguji data pada rasio 80:20, yakni dengan nilai akurasi 0.45302566. Berdasarkan grafik pengujian tersebut juga terlihat bahwa SVM dan KNN memiliki nilai akurasi yang tidak terlalu jauh pada saat menguji data dengan rasio data 60:40 dan 70:40.

## 4. Kesimpulan

Berdasarkan hasil pengujian terhadap prediksi struktur sekunder protein menggunakan metode KNN dan SVM, dapat ditarik kesimpulan bahwa prediksi struktur sekunder protein dilakukan dengan cara mengambil urutan ganjil untuk struktur primer dan urutan genap untuk struktur sekunder. Dataset yang digunakan dalam penelitian ini diambil dengan panjang sekuens sama panjang yaitu 108 sekuens dengan fitur sebanyak 22.594 data.

Hasil pengujian diperoleh dengan proses *training* dan *testing* dengan melakukan perhitungan nilai K untuk metode KNN dan perhitungan nilai C*, gamma*, dan *kernel,* dimana kernel yang digunakan adalah RBF untuk metode SVM.

Metode KNN dinilai mampu memprediksi struktur sekunder protein dengan hasil akurasi terbaik pada saat menggunakan nilai K = 5, K = 10, dan K = 15 serta *sliding window* sebesar 15. Akurasi yang dihasilkan yaitu 49.59% menjadi akurasi terbaik, sedangkan metode SVM menggunakan nilai C = 1 dan *gamma* = 0.1 menghasilkan akurasi terbaik sebesar 0.62.

Pada penelitian selanjutnya dapat dilakukan optimasi algoritma (*hyperparameter tuning*) pada algoritma *Support Vector Machine* dengan menggunakan metode *grid search optimization*, sehingga dapat dihasilkan nilai akurasi yang lebih baik dibandingkan nilai akurasi SVM yang ditemukan pada penelitian ini.

## Referensi

Agustina, D., Putri, E., Fauzi, F., Alawiyah, S. N., Wasono, R., Studi, P., Fmipa, S., Semarang, U. M., Program, D., & Fmipa, S. S. (2020). *Prosiding Seminar Edusainstech Penerapan Metode Support Vector Machine (Svm) Untuk Klasifikasi Data Ekspresi Gen Microarray*.

Budi, S. (2007). *Data mining teknik pemanfaatan data untuk keperluan bisnis* (Vol. 978). Garah Ilmu.

Fai Chin, Y., Hassan, R., & Saberi Mohamad, M. (2012). Optimized Local Protein Structure with Support Vector Machine to Predict Protein Secondary Structure. In *CCIS* (Vol. 295).

Felix, F., Faisal, S., Butarbutar, T. F. M., & Sirait, P. (2019). Implementasi CNN dan SVM untuk Identifikasi Penyakit Tomat via Daun. *Jurnal SIFO Mikroskil*, *20*(2), 117–134.

Haryanto, T., & Surya, B. (2015). Penggunaan Fitur Kimiafisik dan Posisi Atom untuk Prediksi Struktur Sekunder Protein. *Jurnal Edukasi Dan Penelitian Informatika*, *1*(2), 133–138. https://doi.org/http://dx.doi.org/10.26418/jp.v1i2.11919

Huang, Y. F., & Chen, S. Y. (2013). Extracting physicochemical features to predict protein secondary structure. *The Scientific World Journal*, *2013*. https://doi.org/10.1155/2013/347106

Indrabulan, T., & Syarif, I. (2020). Algoritma Interest Point dalam segmentasi citra objek kendaraan. *PROtek: Jurnal Ilmiah Teknik Elektro*, *7*(1), 11–15.

Khairul Anam, M., Irawan, Y., & Jamaris, M. (2022). Comparison of support vector machine and XGBSVM in analyzing public opinion on Covid-19 vaccination. *ILKOM Jurnal Ilmiah*, *14*(1), 32–38. https://doi.org/10.33096/ilkom.v14i1.1090.32-38

Krisandi, N., Helmi, B. P., & others. (2013). Algoritma k-Nearest Neighbor dalam Klasifikasi Data Hasil Produksi Kelapa Sawit pada PT. Minamas Kecamatan Parindu. *Bimaster: Buletin Ilmiah Matematika, Statistika Dan Terapannya*, *2*(1).

Mardiasih, I. D. W. I. A. Y. U., & Haryanto, T. (2014). Pemodelan Probabilistic Neural Network (PNN) untuk Prediksi Struktur

Sekunder Protein. *Makalah Kolokium Ekstensi*, *1*(1).

Prakash, N., & Singh, Y. (2015). Support Vector Machines for Face Recognition. *International Research Journal of Engineering and Technology*. www.irjet.net

Sutrimo, & Wismarini, D. (2022). Prediksi Proses Persalinan Menggunakan Algoritma Knn Berbobot Pada Monitoring Elektronik Personal Health Record Ibu Hamil. *MISI : Jurnal Manajemen Informatika & Sistem Informasi*, *5*, 65–76. https://doi.org/10.36595/misi.v5i1http://e-journal.stmiklombok.ac.id/index.php/misi

Whidhiasih, R. N., Wahanani, N. A., & Supriyanto, S. (2013). Klasifikasi Buah Belimbing Berdasarkan Citra Red-Green-Blue Menggunakan Knn Dan Lda. *Penelitian Ilmu Komputer Sistem Embedded Dan Logic*, *1*(1), 155397.

Wulandari, A. (2020). Aplikasi Support Vector Machine(Svm)Untuk Pencarian Binding Siteprotein-LIGAN. *MATHunesa: Jurnal Ilmiah Matematika*, *Vol 8 No 2 (2020)*. https://doi.org/https://doi.org/10.26740/mathunesa.v8n2.p157-161

Zhou, J., Wang, H., Zhao, Z., Xu, R., & Lu, Q. (2018). CNNH_PSS: Protein 8-class secondary structure prediction by convolutional neural network with highway. *BMC Bioinformatics*, *19*. https://doi.org/10.1186/s12859-018-2067-8