

Klasifikasi Sinyal EEG Menggunakan Model K-Nearest Neighbor Untuk Pengenalan Kata Yang Dibayangkan

Abdul Rauf¹; Efy Yosrita²; Rosida Nur Aziza³

^{1,2,3} Institut Teknologi – PLN

¹ abdul1731375@itpln.ac.id

ABSTRACT

Locked in syndrome (LIS) is a condition of complete paralysis in which people with LIS are conscious but unable to move or communicate verbally except to move their eyes or blink. One way that can help LIS sufferers to communicate and interact is through recording brain signals called Electroencephalogram (EEG). In this study, the data from the recording of the EEG signal has gone through the extraction stage. The extracted data is preprocessed and classified using the K-Nearest Neighbor (K-NN) algorithm to be visualized using a web-based application. The results of the classification using the K-Nearest Neighbor algorithm with a value of $K = 1$ resulted in 82% accuracy, 82% precision and 82% recall.

Keywords: LIS, EEG, K-Nearest Neighbor

ABSTRAK

Locked in syndrome (LIS) adalah suatu kondisi kelumpuhan total dimana penderita LIS dalam keadaan sadar tetapi tidak dapat bergerak atau berkomunikasi secara verbal kecuali untuk menggerakkan mata atau berkedip. Salah satu cara yang dapat membantu penderita LIS untuk dapat berkomunikasi dan berinteraksi melalui perekaman sinyal otak yang dinamakan Elektroencephalogram (EEG). Pada penelitian ini menggunakan data hasil perekaman sinyal EEG yang sudah melalui tahapan ekstraksi. Data hasil ekstraksi dilakukan tahapan preprocessing dan proses klasifikasi menggunakan algoritma K-Nearest Neighbor (K-NN) untuk divisualisasikan menggunakan aplikasi berbasis web. Hasil dari klasifikasi menggunakan algoritma K-Nearest Neighbor dengan nilai $K=1$ menghasilkan accuracy sebesar 82%, precision 82% dan recall 82%

Kata Kunci: LIS, EEG, K-Nearest Neighbor

1. PENDAHULUAN

Elektroencephalogram (EEG) merupakan sebuah gelombang yang didapatkan melalui perangkat yang dinamakan dengan *Emotiv Epoc+* yang digunakan untuk merekam dan mengambil data atau informasi dari aktifitas sinyal yang ada pada otak manusia [1]–[5]. Sinyal EEG merupakan sinyal elektrik yang didapatkan dari sepanjang kulit kepala manusia yang direkam untuk menginformasikan aktifitas listrik dalam otak untuk mendapatkan informasi yang akan digunakan sebagai media komunikasi [6]–[9].

Algoritma *K-Nearest Neighbor* adalah salah satu metode untuk melakukan klasifikasi yang sudah digunakan diberbagai bidang, diantaranya klasifikasi detak jantung pada bidang kesehatan [10]–[12], klasifikasi pada bidang ekonomi [13]–[15], klasifikasi pada bidang pendidikan [16]–[18] dan penelitian pada bidang lainnya. Dengan demikian dapat disimpulkan bahwa algoritma *K-Nearest Neighbor* sudah banyak digunakan untuk proses klasifikasi. Algoritma *K-NN* merupakan metode untuk melakukan klasifikasi terhadap objek yang berdasarkan dari data pembelajaran yang jaraknya paling dekat dengan objek tersebut [19][20]. Algoritma *K-NN* memiliki beberapa kelebihan yang diantaranya sangat mudah dipelajari, pelatihan data yang sangat cepat, tahan terhadap data training yang memiliki *noise* dan cukup efektif untuk pelatihan data yang banyak [21][22].

2. METODE PENELITIAN

2.1. Identifikasi Masalah

Permasalahan dalam penelitian ini yaitu bagaimana menerapkan algoritma *K-Nearest Neighbor* pada klasifikasi sinyal EEG dan menampilkan hasil klasifikasi kata seperti kata makan, kata minum, kata lapar, kata haus, kata senang, kata sedih, kata sakit dan kata toilet menggunakan aplikasi berbasis web.

2.2. Studi Literatur

Pada tahapan ini mencari informasi dan membaca referensi yang dibutuhkan seperti referensi dari buku, referensi dari jurnal yang berhubungan dengan klasifikasi sinyal EEG menggunakan algoritma *K-Nearest Neighbor* untuk dapat menjadi acuan dalam melakukan penelitian.

2.3. Pengumpulan Data

Pengumpulan data dalam penelitian ini dilakukan dengan cara melakukan perekaman sinyal EEG. Perekaman dilakukan di Institut Teknologi PLN pada 11 orang mahasiswa yang terdiri dari 4 orang mahasiswa perempuan dan 7 orang mahasiswa laki-laki dengan menggunakan perangkat *Emotiv Epoc+* untuk melakukan proses perekaman sinyal otak pada manusia. Setiap mahasiswa yang melakukan perekaman harus dalam keadaan sehat dan memikirkan kata yang akan direkam. Kata yang digunakan untuk proses perekaman berjumlah 8 kata, yaitu kata makan, kata minum, kata lapar, kata haus, kata senang, kata sedih, kata sakit dan kata toilet. Setiap kali perekaman dalam satu kata akan menghasilkan 80 data, sehingga dalam satu kali perekaman 8 kata akan menghasilkan 640 data. Setiap mahasiswa yang melakukan perekaman akan melakukan perekaman sebanyak 5 kali, sehingga dalam satu orang mahasiswa yang melakukan perekaman sinyal EEG akan mendapatkan 3200 data. Jumlah data yang akan didapatkan dari keseluruhan mahasiswa yang melakukan perekaman sinyal EEG adalah sebanyak 35.200 data.

2.4. Metode Analisis Data

Metode analisis data pada penelitian ini terdiri dari proses *preprocessing* data, pembuatan mesin pembelajaran menggunakan algoritma *K-Nearest Neighbor* dan evaluasi.

2.4.1. Preprocessing Data

Pada tahapan *preprocessing*, data yang digunakan merupakan data hasil perekaman sinyal EEG menggunakan alat Emotiv Epoc+ dan sudah melewati proses ekstraksi. Data yang telah melewati proses ekstraksi masih terdapat data duplikat, data hilang dan jumlah data perekaman yang tidak sesuai dengan jumlah data yang seharusnya. Data yang terdapat duplikat dilakukan penghapusan terhadap salah satu data. Data yang hilang dilakukan imputasi dengan menggunakan metode mean. Imputasi dengan menggunakan metode mean mengisi *missing value* dengan mencari nilai rata-rata dari semua nilai yang diketahui. Dan untuk jumlah data yang tidak sesuai akan dilakukan *resize* terhadap data tersebut agar jumlah semua data menjadi sama banyak.

Tabel 1. Sampel Data

LABEL	AF3_THETA	AF3_ALPHA	AF3_LOW_BETA	AF3_HIGH_BETA	AF3_GAMMA
0	1.702458	1.203181	0.460678	3.085534	0.838527
0	1.648683	1.529774	0.511154	3.368891	1.029981
0	4.464708	1.795277	0.778195	6.278142	0.948076
0	3.03347	2.028336	0.93333	6.659462	0.916482
1	15.72491	1.967517	1.765821	1.672216	0.649823
1	26.21277	2.704909	1.200279	1.455554	0.580426
1	23.68836	2.477452	0.933676	1.220289	0.662042
1	13.14493	1.668847	1.003609	1.45057	0.701185
2	1.24741	5.522269	0.469974	1.321909	0.81894
2	1.129376	4.443482	0.395184	1.395564	0.795054
2	61.49565	6.425679	0.584725	1.082075	0.539777
2	55.15988	5.94287	0.507969	0.784493	0.495501
3	4.262504	8.72702	0.66385	0.84655	0.366165
3	5.29493	8.05006	0.474413	0.695293	0.395047
3	32.08557	2.689841	0.544226	1.033932	0.414889
3	36.26672	2.938309	0.536058	0.711465	0.412336
4	5.605134	7.234293	2.773499	1.803725	0.699277
4	4.69874	7.588797	2.774168	1.535441	0.604417
4	3.171229	0.965054	0.794836	4.672685	1.780786
4	5.13151	1.16903	0.605117	4.363001	1.720501
5	2.597531	3.48814	2.350743	1.871006	0.912681
5	2.368087	3.625955	2.451953	1.945879	0.908033
5	3.325207	0.94129	0.877731	1.674244	0.655777
5	7.192244	1.157571	1.036718	1.452815	0.567138
6	4.355445	2.494485	0.932097	3.049842	0.52168
6	3.967897	2.436238	0.615354	2.825515	0.473766
6	6.765077	6.589856	2.282284	5.23023	1.044323
6	7.911431	6.948172	2.323259	5.734205	1.212722

LABEL	AF3_THETA	AF3_ALPHA	AF3_LOW_BETA	AF3_HIGH_BETA	AF3_GAMMA
7	3.060828	1.230584	1.24461	1.469193	0.759646
7	3.03226	1.442779	1.166397	1.555305	0.765534
7	5.994707	4.503517	1.467686	1.747511	0.668197
7	4.250394	3.291226	1.391709	1.339633	0.559932

Tabel di atas merupakan data sampel hasil proses ekstraksi pada dataset sinyal EEG menggunakan *fast fourier transform* yang sudah dilakukan pada penelitian sebelumnya. Proses ekstraksi ini menghasilkan data ekstraksi sebanyak 640 baris setiap sesi subjek dan memiliki 70 kolom yang terdiri dari gabungan antara 14 elektroda dan 5 gelombang yaitu *theta*, *alpha*, *low beta*, *high beta* dan *gamma*.

2.4.2. Pembuatan Mesin Pembelajaran K-NN

Pada tahapan pembuatan mesin pembelajaran menggunakan bahasa pemrograman *Python* dan algoritma *K-Neareast Neighbor* sebagai model untuk melakukan proses klasifikasi data. Adapun tahapan perhitungan dari metode K-NN yaitu sebagai berikut:

1. Menyiapkan data latih

Tabel 2. Sampel Data Latih

No	Label	AF3_THETA	AF3_ALPHA	AF3_LOW_BETA	AF3_HIGH_BETA	AF3_GAMMA
1	0	1.648683	1.529774	0.511154	3.368891	1.029981
2	0	1.855305	1.987211	0.560252	3.761023	1.148339
3	1	2.352318	4.524428	1.451414	1.736114	0.839391
4	1	2.573653	5.476707	1.895669	1.72398	0.895624
5	2	2.874563	2.656353	1.257373	1.587613	0.604079
6	2	2.713335	3.144904	1.363784	1.622506	0.619114
7	3	11.66675	1.7308	0.654298	2.147005	0.981446
8	3	7.292709	1.619289	0.743364	2.384083	0.972761
9	4	21.63486	4.070159	1.401522	1.462895	0.480931
10	4	26.97585	4.905097	1.660201	1.673036	0.513786
11	5	8.307137	6.143272	1.109404	4.112378	0.585811
12	5	12.68122	6.677336	1.051835	3.566047	0.59651
13	6	6.438546	1.880114	1.562376	3.374364	0.859908
14	6	6.604903	1.852004	1.572287	2.833242	0.982443
15	7	2.16924	0.614311	0.528808	1.147088	0.496631
16	7	2.980366	0.831523	0.522589	1.159548	0.504039

2. Menyiapkan data yang akan diuji

Tabel 3. Sampel Data Uji

No	Label	AF3_THETA	AF3_ALPHA	AF3_LOW_BETA	AF3_HIGH_BETA	AF3_GAMMA
1	0	1.702458	1.203181	0.460678	3.085534	0.838527
2	1	2.883851	6.130239	2.331827	1.646764	0.953047
3	2	2.733988	3.639893	1.424514	1.67527	0.701165
4	3	1.902099	1.714063	0.983486	2.479318	0.836102
5	4	21.99435	4.547952	1.971865	1.71733	0.727782
6	5	17.74884	6.926249	0.926378	2.723347	0.639606

3. Tentukan nilai K

Nilai K yang digunakan yaitu K = 1

4. Hitung nilai *Euclidean Distance* dari data uji terhadap keseluruhan data latih dengan rumus:

$$d_i = \sqrt{\sum_{j=1}^p (x_{2i} - x_{1i})^2} \quad (1)$$

Keterangan:

d : hasil berupa jarak antara data sampel dengan data uji

x1 : sampel data uji

x2 : sampel data latih

p : dimensi data yang akan dibandingkan dengan sampel data uji

5. Langkah ke 5 mengurutkan objek-objek berdasarkan jarak *Euclidean Distance* dari yang terkecil ke terbesar

6. Langkah ke 6 mencari objek terdekat dengan nilai K = 1

7.

Tabel 4. Hasil Klasifikasi Data Uji 1-6

Actual		Prediksi		Keterangan
Label	Kata	Label	Kata	
0	Makan	0	Makan	Sesuai
1	Minum	1	Minum	Sesuai
2	Lapar	2	Lapar	Sesuai
3	Haus	0	Makan	Tidak Sesuai
4	Senang	4	Senang	Sesuai
5	Sedih	5	Sedih	Sesuai

Tabel di atas merupakan data hasil klasifikasi menggunakan metode *K-Nearest Neighbor* dengan K=1 pada data uji 1-6.

2.4.3. Evaluasi

Evaluasi dilakukan untuk menghitung performa algoritma *K-Nearest Neighbor* dengan K=1 pada kelas makan, kelas minum, kelas lapar, kelas haus, kelas senang, kelas sedih, kelas sakit dan kelas toilet digunakan metode *Confusion Matrix*. Evaluasi algoritma *K-Nearest Neighbor* dengan K=1 pada data uji 1-6 menggunakan metode *Confusion Matrix* sebagai berikut:

Tabel 5. Hasil *Confusion Matrix*

		PREDIKSI							
		Makan	Minum	Lapar	Haus	Senang	Sedih	Sakit	Toilet
AKTUAL	Makan	1	0	0	0	0	0	0	0
	Minum	0	1	0	0	0	0	0	0
	Lapar	0	0	1	0	0	0	0	0
	Haus	1	0	0	0	0	0	0	0
	Senang	0	0	0	0	1	0	0	0
	Sedih	0	0	0	0	0	1	0	0
	Sakit	0	0	0	0	0	0	0	0
	Toilet	0	0	0	0	0	0	0	0

Berdasarkan hasil *confusion matrix* diatas maka dihitung nilai *accuracy* menggunakan rumus:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (2)$$

$$accuracy = \frac{5+0}{5+0+1+0} = 0.833$$

$$\text{persentase accuracy} = 0.833 \times 100\% = 83.3\%$$

2.5. Pembuatan Aplikasi

Pada tahapan pembuatan aplikasi yaitu menggunakan basis data dan bahasa pemrograman PHP. Basis data digunakan untuk menyimpan data pengujian dan data hasil dari pengujian yang dilakukan. Bahasa pemrograman PHP digunakan untuk membuat aplikasi yang berbasis web sebagai tempat untuk user melakukan proses klasifikasi data.

2.6. Pembuatan Laporan

Tahapan ini merupakan tahapan terakhir dengan pembuatan laporan hasil dari proses penelitian yang telah dilakukan.

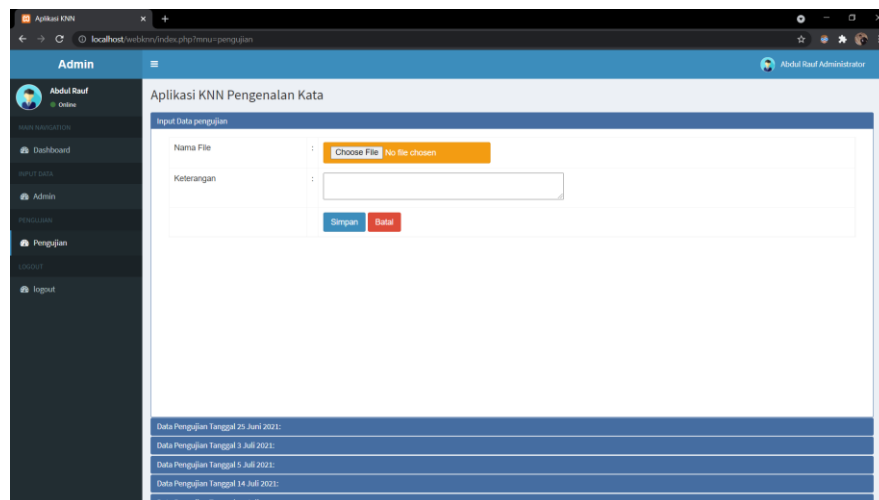
3. HASIL DAN PEMBAHASAN

3.1. Hasil

Pada bab ini menampilkan hasil klasifikasi sinyal EEG menggunakan algoritma *K-Nearest Neighbor* dengan melakukan pengujian menggunakan bahasa pemrograman *Python* dan *PHP*. Implementasi dari bahasa pemrograman ini akan mendapatkan hasil dari penerapan model *K-Nearest Neighbor* untuk klasifikasi sinyal EEG pengenalan kata yang dibayangkan.

1. Halaman Pengujian

Halaman pengujian merupakan halaman untuk melakukan pengujian data. Pada halaman ini user dapat melakukan pengujian dengan mengupload file data yang akan di uji. Setelah melakukan *upload* data, user dapat mengisi keterangan dan menekan tombol simpan. Maka secara otomatis data yang di *upload* tersebut akan tersimpan dalam database pengujian.



Gambar 1. Halaman Pengujian

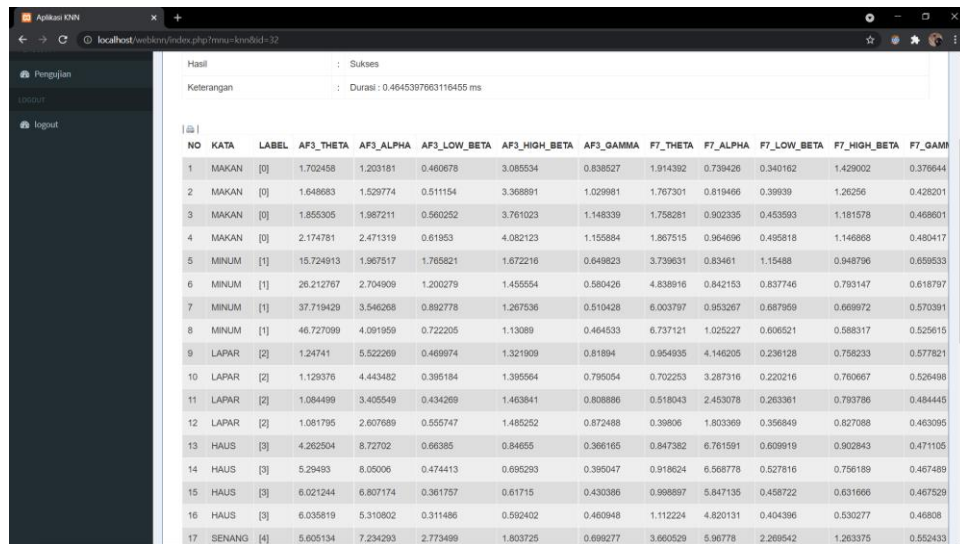
Gambar 2. Gambar Data Uji

Setelah file berhasil tersimpan dalam database pengujian, maka selanjutnya file tersebut akan dibaca oleh mesin pembelajaran *Python* dengan menggunakan algoritma *K-Nearest Neighbor*. Data tersebut akan terbagi menjadi data training 75% dan data testing 25% dan secara otomatis akan melakukan pengujian.

```
UPDATE 'tb_detail' set 'label'=[0], 'kata'='MAKAN', 'status'='1' where 'id'='1477'
UPDATE 'tb_detail' set 'label'=[0], 'kata'='MAKAN', 'status'='1' where 'id'='1478'
UPDATE 'tb_detail' set 'label'=[0], 'kata'='MAKAN', 'status'='1' where 'id'='1479'
UPDATE 'tb_detail' set 'label'=[0], 'kata'='MAKAN', 'status'='1' where 'id'='1480'
UPDATE 'tb_detail' set 'label'=[1], 'kata'='MINUM', 'status'='1' where 'id'='1481'
UPDATE 'tb_detail' set 'label'=[1], 'kata'='MINUM', 'status'='1' where 'id'='1482'
UPDATE 'tb_detail' set 'label'=[1], 'kata'='MINUM', 'status'='1' where 'id'='1483'
UPDATE 'tb_detail' set 'label'=[1], 'kata'='MINUM', 'status'='1' where 'id'='1484'
UPDATE 'tb_detail' set 'label'=[2], 'kata'='LAPAR', 'status'='1' where 'id'='1485'
UPDATE 'tb_detail' set 'label'=[2], 'kata'='LAPAR', 'status'='1' where 'id'='1486'
UPDATE 'tb_detail' set 'label'=[2], 'kata'='LAPAR', 'status'='1' where 'id'='1487'
UPDATE 'tb_detail' set 'label'=[2], 'kata'='LAPAR', 'status'='1' where 'id'='1488'
UPDATE 'tb_detail' set 'label'=[3], 'kata'='HAUS', 'status'='1' where 'id'='1489'
UPDATE 'tb_detail' set 'label'=[3], 'kata'='HAUS', 'status'='1' where 'id'='1490'
UPDATE 'tb_detail' set 'label'=[3], 'kata'='HAUS', 'status'='1' where 'id'='1491'
UPDATE 'tb_detail' set 'label'=[3], 'kata'='HAUS', 'status'='1' where 'id'='1492'
UPDATE 'tb_detail' set 'label'=[4], 'kata'='SEHANG', 'status'='1' where 'id'='1493'
UPDATE 'tb_detail' set 'label'=[4], 'kata'='SEHANG', 'status'='1' where 'id'='1494'
UPDATE 'tb_detail' set 'label'=[4], 'kata'='SEHANG', 'status'='1' where 'id'='1495'
UPDATE 'tb_detail' set 'label'=[4], 'kata'='SEHANG', 'status'='1' where 'id'='1496'
UPDATE 'tb_detail' set 'label'=[5], 'kata'='SEDIH', 'status'='1' where 'id'='1497'
UPDATE 'tb_detail' set 'label'=[5], 'kata'='SEDIH', 'status'='1' where 'id'='1498'
UPDATE 'tb_detail' set 'label'=[5], 'kata'='SEDIH', 'status'='1' where 'id'='1499'
UPDATE 'tb_detail' set 'label'=[5], 'kata'='SEDIH', 'status'='1' where 'id'='1500'
UPDATE 'tb_detail' set 'label'=[6], 'kata'='SAKIT', 'status'='1' where 'id'='1501'
UPDATE 'tb_detail' set 'label'=[6], 'kata'='SAKIT', 'status'='1' where 'id'='1502'
UPDATE 'tb_detail' set 'label'=[6], 'kata'='SAKIT', 'status'='1' where 'id'='1503'
UPDATE 'tb_detail' set 'label'=[6], 'kata'='SAKIT', 'status'='1' where 'id'='1504'
UPDATE 'tb_detail' set 'label'=[7], 'kata'='TOILET', 'status'='1' where 'id'='1505'
UPDATE 'tb_detail' set 'label'=[7], 'kata'='TOILET', 'status'='1' where 'id'='1506'
UPDATE 'tb_detail' set 'label'=[7], 'kata'='TOILET', 'status'='1' where 'id'='1507'
UPDATE 'tb_detail' set 'label'=[7], 'kata'='TOILET', 'status'='1' where 'id'='1508'
UPDATE 'tb_pengujian' set 'rekapitulasi'='Proses perbandingan sesuai metode KNN', 'hasil'='sukses', 'status'='1', 'keterangan'='Durasi : 0.4645397663116455 ms' where 'id_pengujian'='32'
```

Gambar 3. Proses pengujian KNN

Setelah melakukan pengujian, hasil pengujian tersebut akan masuk ke dalam *database*. Hasil tersebut dapat dilihat pada data pengujian dalam halaman pengujian sesuai dengan tanggal dilakukannya pengujian tersebut.

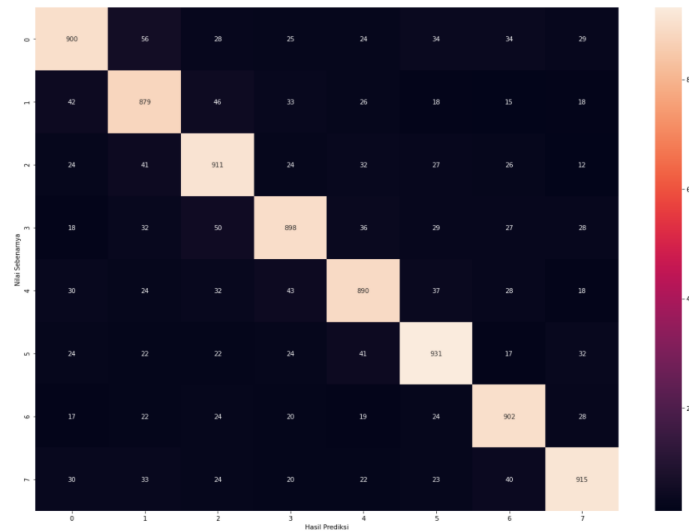


NO	KATA	LABEL	AF3_THETA	AF3_ALPHA	AF3_LOW_BETA	AF3_HIGH_BETA	AF3_GAMMA	F7_THETA	F7_ALPHA	F7_LOW_BETA	F7_HIGH_BETA	F7_GAMMA
1	MAKAN	[0]	1.702458	1.203181	0.460678	3.085534	0.838527	1.914392	0.739426	0.340162	1.429002	0.376644
2	MAKAN	[0]	1.648683	1.529774	0.511154	3.368891	1.029981	1.767301	0.819466	0.39939	1.26256	0.428201
3	MAKAN	[0]	1.855305	1.987211	0.560252	3.761023	1.148339	1.756281	0.902335	0.453593	1.181578	0.468801
4	MAKAN	[0]	2.174781	2.471319	0.61953	4.082123	1.155884	1.867515	0.964696	0.495818	1.146868	0.480417
5	MINUM	[1]	15.724913	1.967517	1.765821	1.672216	0.649823	3.739631	0.83461	1.15488	0.948796	0.658933
6	MINUM	[1]	26.212767	2.704909	1.200279	1.455554	0.580426	4.838916	0.842153	0.837746	0.793147	0.618797
7	MINUM	[1]	37.719429	3.546268	0.892778	1.267536	0.510428	6.003797	0.953267	0.687959	0.669972	0.570391
8	MINUM	[1]	46.727099	4.091959	0.722205	1.13089	0.464533	6.737121	1.025227	0.606521	0.588317	0.525615
9	LAPAR	[2]	1.24741	5.522269	0.469974	1.321909	0.81894	0.964935	4.146205	0.236128	0.758233	0.577821
10	LAPAR	[2]	1.129376	4.443482	0.395184	1.395564	0.795054	0.702253	3.287316	0.220216	0.760667	0.526488
11	LAPAR	[2]	1.084499	3.405549	0.434269	1.463841	0.808886	0.518043	2.453078	0.263361	0.793786	0.484445
12	LAPAR	[2]	1.081795	2.607689	0.555747	1.485252	0.872488	0.39606	1.803369	0.356849	0.827088	0.463095
13	HAUS	[3]	4.262504	8.72702	0.66385	0.84655	0.366165	0.847382	6.761591	0.609919	0.902843	0.471105
14	HAUS	[3]	5.29493	6.05006	0.474413	0.695293	0.395047	0.918624	6.568778	0.527816	0.756189	0.467489
15	HAUS	[3]	6.021244	6.807174	0.361757	0.61715	0.430386	0.988897	5.847135	0.458722	0.631666	0.467529
16	HAUS	[3]	6.035819	5.310802	0.311486	0.592402	0.460948	1.112224	4.820131	0.404396	0.530277	0.46808
17	SENANG	[4]	5.605134	7.234293	2.773499	1.803725	0.696277	3.600529	5.96778	2.289542	1.263375	0.552433

Gambar 4. Data Hasil Pengujian

3.2. Pembahasan

Aplikasi yang dihasilkan yaitu aplikasi berbasis web untuk melakukan klasifikasi data sinyal EEG. Aplikasi ini menggunakan basis data sebagai tempat untuk menyimpan data pengujian dan menyimpan data hasil dari pengujian yang dilakukan. Proses pengujian dilakukan pada aplikasi web dengan mesin pembelajaran *Python* yang sudah dirunning terlebih dahulu. Untuk melakukan pengujian user mengupload data yang akan diuji pada aplikasi web, dan secara otomatis data uji tersebut akan masuk ke dalam *database* dan mesin pembelajaran akan mengambil data uji dari *database* untuk dilakukan pengujian. Setelah pengujian selesai, hasil dari pengujian secara otomatis masuk ke dalam *database* dan dapat dilihat pada aplikasi web. Pada pengujian ini menggunakan 35.200 data yang terdiri dari kelas makan, kelas minum, kelas lapar, kelas haus, kelas senang, kelas sedih, kelas sakit dan kelas toilet dengan setiap melakukan pengujian data akan terbagi menjadi 75% data training dan 25% data testing sehingga data training berjumlah 26.400 dan data testing berjumlah 8.800. Proses pengujian yang dilakukan menggunakan $K=1$. Hasil dari pengujian dilakukan evaluasi dengan metode *Confusion Matrix* menggunakan mesin pembelajaran *Python* dengan hasil seperti ditunjukkan gb.5.



Gambar 5. Hasil Confusion Matrix

Gambar diatas merupakan hasil *Confusion Matrix* dengan menggunakan data pengujian sebanyak 8.800 data.

Tabel 6. Laporan Hasil Klasifikasi

Label	Precision	Recall	F1-Score	Support
0	80%	83%	81%	1085
1	82%	79%	80%	1109
2	83%	80%	82%	1137
3	80%	83%	81%	1087
4	81%	82%	81%	1090
5	84%	83%	83%	1123
6	85%	83%	84%	1089
7	83%	85%	84%	1080

Tabel diatas merupakan nilai *precision*, *recall* dan *f1-score* yang didapatkan dari *Confusion Matrix*. Dapat diketahui nilai *accuracy* sebesar 82% dengan menggunakan 8800 data uji dan nilai rata-rata *precision* sebesar 82%, *recall* 82%, *f1-score* 82%.

4. PENUTUP

4.1. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan penulis dapat diambil kesimpulan yaitu sebagai berikut:

1. Hasil dari klasifikasi sinyal EEG untuk pengenalan kata yang dibayangkan menggunakan model K-Nearest Neighbor dengan nilai K=1 menghasilkan *accuracy* sebesar 82%, *precision* 82% dan *recall* 82%.
2. Hasil dari klasifikasi sinyal EEG untuk pengenalan kata yang dibayangkan menggunakan model K-Nearest Neighbor divisualisasikan dengan menggunakan aplikasi berbasis web

4.2. Saran

Berdasarkan hasil penelitian yang telah dilakukan, penulis memberikan saran untuk pengembangan selanjutnya yaitu sebagai berikut:

1. Dalam melakukan proses klasifikasi disarankan untuk menggunakan data dalam jumlah yang lebih banyak untuk mendapatkan hasil yang lebih maksimal.
2. Dari hasil penelitian ini disarankan untuk menggunakan metode klasifikasi lainnya seperti Random Forest, Support Vector Machine (SVM), Learning Vector Quantization (LVQ) untuk klasifikasi sinyal EEG pengenalan kata yang dibayangkan.

DAFTAR PUSTAKA

- [1] G. . Putra, A. Desrianty, and Yuniar, “Analisis Tingkat Kelelahan dan Kantuk Pada Supir Bus Berdasarkan Heart Rate Variability (HRV) Dan Electroencephalogram,” *J. Online Inst. Teknol. Nas.*, vol. 03, no. 04, pp. 207–218, 2015.
- [2] R. Ramirez and Z. Vamvakousis, “Detecting Emotion from EEG Signals Using the Emotive Epoc Device,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7670 LNAI, no. December, 2012, doi: 10.1007/978-3-642-35139-6.
- [3] M. N. Fakhruzzaman, E. Riksakomara, and H. Suryotrisongko, “EEG Wave Identification in Human Brain with Emotiv EPOC for Motor Imagery,” *Procedia Comput. Sci.*, vol. 72, pp. 269–276, 2015, doi: 10.1016/j.procs.2015.12.140.
- [4] S. . Dewangga, H. Tjandrasa, and D. Herumurti, “Robot motion control using the emotiv EPOC EEG system,” *Bull. Electr. Eng. Informatics*, vol. 7, no. 2, pp. 279–285, 2018, doi: 10.11591/eei.v7i2.678.
- [5] J. Kosiński, K. Szklanny, A. Wiczorkowska, and M. Wichrowski, “An analysis of game-related emotions using EMOTIV EPOC,” *Proc. 2018 Fed. Conf. Comput. Sci. Inf. Syst. FedCSIS 2018*, vol. 15, pp. 913–917, 2018, doi: 10.15439/2018F296.
- [6] G. . Siregar and Y. Rahayu, “Sistem Pendeteksi Gelombang Otak berbasis Electroencephalogram (EEG) pada Studi Kasus Anak Penderita Autisme,” *Jom FTEKNIK*, vol. 6, no. 0, pp. 1–8, 2020, [Online]. Available: <https://jom.unri.ac.id/index.php/JOMFTEKNIK/article/view/26340>.
- [7] M. . Firdaus, H. Fauzi, and Y. . Fuadah, “ANALISA KOMPOSISI KANAL AKTIF PADA SINYAL EEG STROKE MENGGUNAKAN METODE SPATIAL SELECTION THE ANALYSIS OF ACTIVE CHANNEL COMPOSITION ON EEG STROKE SIGNAL,” vol. 7, no. 2, pp. 3147–3154, 2020.
- [8] G. Virgenius and Jondri, “Klasifikasi Keadaan Mata Berdasarkan Sinyal Electroencephalography (EEG) dengan Menggunakan Hierarchical Temporal Memory Eye State Classification Based on Electroencephalography (EEG) Signal Using Hierarchical Temporal Memory,” vol. 4, no. 3, pp. 4667–4674, 2017.
- [9] I. Wijayanto, S. Hadiyoso, and Y. . Hariyani, “IDENTIFIKASI KONDISI SINYAL ELECTROENCEPHALOGRAM (EEG) YANG TERPAPAR RADIASI SINYAL GAWAI 4G LTE 1800 MHz MENGGUNAKAN LEARNING VECTOR QUANTIZATION (LVQ),” *TEKTRIKA - J. Penelit. dan Pengemb. Telekomun. Kendali, Komputer, Elektr. dan Elektron.*, vol. 1, no. 2, pp. 138–143, 2019, doi: 10.25124/tektrika.v1i2.1746.
- [10] U. Hasanah, L. . Resita, A. Pratama, and I. Cholissodin, “Perbandingan Metode SVM, Fuzzy-

- KNN, dan BDT-SVM Untuk Klasifikasi Detak Jantung Hasil Elektrokardiografi,” 2019.
- [11] A. Eviyanti, H. Hindarto, and M. Abror, “Klasifikasi Normal , Tanpa Kejang , Kejang Terhadap Sinyal Epilepsi Menggunakan Metode Teknik Sampling dan K-Nearest Neighbor,” 2020.
- [12] M. Angga, “Classification of Encephalo Graph (EEG) Signals For Epilepsy Using Discrete Wavelet Transform and K-Nearest Neighbor Methods,” vol. 1, no. 1, 2021.
- [13] H. Leidiyana, “Penerapan Algoritma K-Nearest Neighbor Untuk Penentuan Resiko Kredit Kepemilikan Kendaraan Bermotor,” pp. 217–224, 2014.
- [14] R. . Rustiyan and Mustakim.M, “Klasifikasi Jenis Perairan pada Kapal Perikanan di Indonesia Menggunakan K-Nearest Neighbor,” *Semin. Nas. Teknol. Informasi, Komun. dan Ind.* 9, pp. 178–185, 2017.
- [15] W. . Shidiq, “Penerapan K-Nearest Neighbor Berbasis Algoritma Genetika Untuk Klasifikasi Mutu Padi Organik,” pp. 121–126, 2017.
- [16] G. . Pradnyana and A. A. . Permana, “Sistem Pembagian Kelas Kuliah Mahasiswa Dengan Metode K-Means Dan K-Nearest Neighbors Untuk Meningkatkan Kualitas Pembelajaran,” *JUTI J. Ilm. Teknol. Inf.*, vol. 16, no. 1, p. 59, 2018, doi: 10.12962/j24068535.v16i1.a696.
- [17] M. Mustakim and G. Oktaviani, “Algoritma K-Nearest Neighbor Classification Sebagai Sistem Prediksi Predikat Prestasi Mahasiswa,” vol. 13, no. 2, pp. 195–202, 2016.
- [18] T. Setiyorini and T. . Rizky, “KOMPARASI METODE DECISION TREE , NAIVE BAYES DAN K-NEAREST NEIGHBOR PADA KLASIFIKASI KINERJA SISWA,” vol. 15, no. 2, pp. 85–92, 2018.
- [19] D. Yunita, “Perbandingan Algoritma K-Nearest Neighbor dan Decision Tree untuk Penentuan Risiko Kredit Kepemilikan Mobil,” *J. Inform. Univ. Pamulang*, vol. 2, no. 2, p. 103, 2017, doi: 10.32493/informatika.v2i2.1512.
- [20] A. . Johar, D. Yanosma, and K. Anggriani, “IMPLEMENTASI METODE K-NEAREST NEIGHBOR (KNN) DAN SIMPLE ADDITIVE WEIGHTING (SAW) DALAM PENGAMBILAN KEPUTUSAN SELEKSI PENERIMAAN ANGGOTA PASKIBRAKA,” vol. III, no. 0065, pp. 98–112, 2016.
- [21] S. Mutrofin, A. Kurniawardhani, A. Izzah, and M. Masrur, “Optimasi Teknik Klasifikasi Modified k Nearest Neighbor Menggunakan Algoritma Genetika,” *J. Gamma*, vol. 10, no. 1, pp. 1–5, 2015.
- [22] Y. Kustiyahningsih and N. Syafaah, “Sistem Pendukung Keputusan Untuk Menentukan Jurusan Pada Siswa SMA Menggunakan Metode KNN,” *J. Istek*, vol. VI, no. 1, pp. 40–42, 2013.