Klasifikasi Batubara Berdasarkan Jenis Kalori Dengan Menggunakan *Algoritma Modified K-Nearest Neighbor* (Studi Kasus : PT.Pancaran Surya Abadi)

Classification Of Coal Based On Calorie Type Using Modified K-Nearest Neighbor Algorithm (*Case Study:* PT.Pancaran Surya Abadi)

Imalita Agustin[1], Yuki Novia Nasution[2], dan Wasono[3]
[1]Laboratorium Statistika Komputasi FMIPA Universitas Mulawarman
[2,3]Laboratorium Matematika Komputasi FMIPA Universitas Mulawarman
E-mail: imalitastatistika13@gmail.com

Abstract

Coal is a sedimentary rock containing the main elements Carbon (C), Hydrogen (H), and Oxygen (O). Examination of coal samples in the laboratory according to company operational standard based on Air Dried Basis (ADB) are the amount of water, ash content, flying substance, solid carbon, sulfur, and Gross Calorific Value. At PT. Pancaran Surya Abadi Anggana Subdistrict Kutai Kartanegara, coal is classified based on its calorie type namely Antrachite, Bituminous, and Sub-Bituminous. In this research Modified K-Nearest Neighbor (MKNN) Algorithm is used to predict the classification. The k-Fold Cross Validation technique is used to obtain the optimal K value on MKNN Algorithm for accuracy. A measurement based on this research, the K-Optimal value used in MKNN Algorithm for coal classification in PT.Pancaran Surya Abadi is 3-NN. The value of K = 3 produces the prediction accuracy of Coal Classification based on the type of calories in PT.Pancaran Surya Abadi on 100% testing data.

Keywords : Coal, Classification ,k-Fold Cross Validation, MKNN

Pendahuluan

Data mining bukan merupakan suatu bidang yang baru, *data mining* mewarisi banyak aspek dan teknik dari bidang-bidang ilmu yang sudah mapan terlebih dahulu. *Data mining* memiliki akar yang panjang dari bidang ilmu seperti kecerdasan buatan (*artificial intelligent), statistic database, information retrieval* dan juga, *machine learning.* Pada bidang *data mining* terdapat banyak metode atau fungsi yang bisa digunakan untuk menemukan, menggali dan menambang pengetahuan, salah satunya adalah klasifikasi.

Klasifikasi adalah bagian dari *machine learning* yang merupakan proses penemuan model atau fungsi yang menjelaskan atau membedakan konsep atau kelas data, dengan tujuan untuk dapat memperkirakan kelas dari suatu objek yang labelnya tidak diketahui. Metode-metode dalam algoritma klasifikasi berdasarkan pembelajaran terbagi menjadi dua, yakni metode *eager learner* dan *lazy learner*. Algoritma yang termasuk dalam kategori *eager learner* diantaranya adalah *Artificial Neural Network (ANN), Support Vector Machine (SVM), Decision Tree* dan *Bayesian*. Sementara Algoritma yang masuk ke dalam kategori *lazy learner* diantaranya adalah *Rote Clasifier, K-Nearest Neighbor* (K-NN), *Fuzzy K-Nearest Neighbor* (FK-NN) dan Regresi Linier (Prasetyo, 2014).

Algoritma *K-Nearest Neighbor* (K-NN) adalah salah satu dari metode klasifikasi yang berbasis *Nearest Neighbor* yang paling tua dan populer. Nilai K yang digunakan menyatakan jumlah tetangga terdekat yang dilibatkan dalam penentuan prediksi label kelas pada data *testing*. Dari K tetangga terdekat yang terpilih kemudian dilakukan *voting* kelas. Kelas dengan jumlah suara tetangga terbanyaklah yang diberikan sebagai label kelas hasil prediksi pada data *testing* (Presetyo,2014).

Akurasi kinerja K-NN banyak dipengaruhi oleh beberapa hal contohnya, pemilihan nilai K. Apabila nilai K terlalu kecil maka berakibat pada hasil prediksi yang dapat terganggu oleh keberadaan *noise* (gangguan). Di sisi lain, jika nilai K terlalu besar maka tetangga terdekat yang dipilih, mungkin terlalu banyak dari kelas yang lain yang sebenarnya tidak relevan disebabkan jarak yang terlalu jauh. Untuk memperkirakan nilai K yang terbaik, bisa dilakukan menggunakan teknik *cross validation.*

Batubara merupakan komoditas energi yang semakin banyak dieksplorasi dan dieksploitasi, untuk pemenuhan kebutuhan energi masyarakat dunia. Di Kalimantan Timur banyak sekali ditemukan tambang-tambang batubara yang masih beroperasi maupun yang telah tutup, hal ini dapat dibuktikan dengan banyak nya kuasa pertambangan di Kalimantan Timur sebanyak 1.180 pada tahun 2009.

PT. Pancaran Surya Abadi adalah perusahaan penambangan batubara yang berlokasi di Kota Samarinda dan memiliki cabang di Kecamatan Anggana, Kutai Kartanegara Kalimantan Timur. Batubara yang di tambang oleh PT. Pancaran Surya Abadi di klasifikasikan menjadi 3 yaitu *Anthracite, Bituminous, Sub-Bituminous*.

Berdasarkan uraian tentang metode klasifikasi MKNN dan Klasifikasi Batubara di PT. Pancaran Surya Abadi, maka penulis tertarik untuk melakukan analisis tersebut, dengan mengambil studi kasus klasifikasi batubara di PT. Pancaran Surya Abadi, yang mana sebelumnya penelitian serupa telah dilakukan oleh Pratiwi,dkk (2017), menggunakan metode K-NN

dengan akurasi tertinggi prediksi pada 3-NN sebesar 87,8%. Dengan demikian, penulis mengusulkan penelitian dengan judul "Klasifikasi Batubara Berdasarkan Jenis Kalori Dengan Menggunakan Algoritma *Modified K-Nearest Neighbor (*MKNN*)* (Studi Kasus: PT Pancaran Surya Abadi Kecamatan Anggana Kabupaten Kutai Kartanegara Provinsi Kalimantan Timur)".

Data Mining

Menurut Tampubolon,dkk (2013), *Data mining* didefinisikan sebagai sebuah proses untuk menemukan hubungan, pola, dan tren baru yang bermakna dengan menyaring data yang sangat besar, yang tersimpan dalam penyimpanan yang menggunakan teknik pengenalan pola seperti teknik statistik dan matematika..

Menurut Presetyo (2014), ada 4 teknik-teknik dalam *data mining* yaitu:

1. *Classification*
   Classification (klasifikasi), merupakan proses penemuan model atau fungsi yang menjelaskan atau membedakan konsep atau kelas data, dengan tujuan untuk dapat memperkirakan kelas dari suatu objek yang labelnya tidak diketahui.
2. *Clustering*
   Clustering digunakan untuk melakukan pengelompokan dengan mengidentifikasi data yang memiliki karakteristik tertentu.
3. Analisis Asosiasi
   Analisis asosiasi (*association analysis*) digunakan untuk menemukan pola yang menggambarkan kekuatan hubungan fitur dalam data. Tujuannya adalah untuk menemukan pola yang menarik dengan cara efisien. Contoh yang paling dekat dengan kehidupan sehari-hari adalah analisis data keranjang belanja.
4. Teknik *forecasting* akan mengambil sederetan angka yang menunjukkan nilai yang berjalan seiring waktu dan kemudian teknik ini akan menghubungkan nilai masa depan dengan menggunakan metode *forecasting* yang berhubungan dengan musim, *trend*, dan *noise* pada data. Contoh metode atau algoritma dalam fungsi ini adalah *Back Propagation* pada *Neural Network*

Konsep Klasifikasi

Berdasarkan cara pelatihan, algoritma algoritma klasifikasi dapat dibagi menjadi dua macam, yaitu *eager learner* dan *lazy learner*.

Algoritma-algoritma klasifikasi yang masuk kategori *eager learner* diantaranya adalah *Artificial Neural Network (ANN), Support Vector Machine (SVM), dan Bayesian*. Algoritma- algoritma klasifikasi yang masuk kategori ini diantaranya adalah *Rome classifier, K Nearest Neighbor* (K-NN), *Fuzzy K-Nearest Neighbor (*FK.NN), dan regresi linier (Prasetyo, 2014).

Metode K-NN

Metode K-NN adalah metode yang melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Metode ini bertujuan untuk mengklasifikasikan objek baru berdasarkan atribut dan *training sample*. Misalkan diberikan suatu titik *query*, selanjutnya akan ditemukan sejumlah K objek atau titik *training* yang paling dekat dengan titik *query*. Nilai prediksi dari *query* akan ditentukan berdasarkan klasifikasi tetanggaan (Dzikrulloh, 2016).

Langkah pertama sebelum mencari jarak data ke tetangga adalah menentukan nilai K tetangga (*Neighbor)*. Kemudian, untuk mendefinisikan jarak antara dua titik, yaitu (x_1, y_1) pada data *training* dan titik (x_2, y_2) pada data uji, maka digunakan jarak Euclidean. Jarak Euclid dapat dihitung dengan rumus:

$$d(x,y) = \sqrt{\sum_{i=1}^{r}(x_{ik} - y_{ik})^2} \qquad (1)$$

dimana :

d : Jarak Euclidean
x_{ik} : nilai ke-*i* variabel ke-*k* dari *x*
y_{ik} : nilai ke-*i* variabel ke-*k* dari *y*
r : jumlah variabel

Metode *k-Fold Cross Validation*

Di dalam algoritma klasifikasi, sebuah data baru diklasifikasikan berdasarkan jarak data baru tersebut dengan tingkat kemiripan data baru yang terdekat dengan pola data. Jumlah data tetangga terdekat ditentukan dan dinyatakan dengan k. Untuk memperkirakan nilai k yang terbaik, bisa dilakukan dengan menggunakan teknik *Cross Validation.* (Banjarsari, 2105).

k-Fold Cross Validation dapat digunakan untuk memperkirakan tingkat kesalahan yang terjadi, karena data *training* pada setiap *fold* cukup berbeda dengan data *training* yang asli. Setiap perulangan disisakan satu subset untuk *testing* dan subset lainnya untuk data *training*. Jumlah data di dalam satu subset dapat dihitung menggunakan rumus:

$$b = \frac{n}{k} \qquad (2)$$

Dimana :

$b =$ banyak data di dalam satu subset
$n =$ banyak data yang digunakan
$k =$ nilai *k-Fold Cross Validation*

Metode *Modified K-Nearest Neighbor* (MKNN)

Algoritma *Modified K-Nearest Neighbor* (MKNN*)* ialah pengembangan dari metode KNN yang diusulkan oleh Parvin dkk, yang sebagian bertujuan untuk mengatasi masalah tingkat akurasi yang rendah pada algoritma K-NN. Pengembangan dilakukan dengan melakukan modifikasi pada algoritma K-NN yang bertujuan untuk meningkatkan kinerja algortima K-NN. Ide utama dari pengembangan algoritma K-NN yang dilakukan adalah untuk menggunakan tetangga yang kuat dalam dataset.

Metode MKNN menambahkan proses validasi pada setiap dataset. Selanjutnya proses klasifikasi dijalankan dengan melakukan pembobotan pada dataset dengan menggunakan nilai validasi sebagai faktor perkalian.

(Parvin, dkk, 2010)

Proses MKNN

Menurut Basuki (2015), langkah-langkah pada algoritma *Modified* K-NN yaitu :

1. Menentukan nilai K, jumlah data, nilai α dan variabel klasifikasi
2. Membagi antara data *training* dan data *testing* menggunakan uji proporsi

$$jumlah\,data\,training = \frac{proporsi\,data\,training}{100} \times N \quad (3)$$

3. Menghitung jarak antar data *training* dan data *testing* menggunakan jarak Euclidean
4. Validitas data t*raining* merupakan proses perhitungan jumlah titik dengan label yang sama pada semua data *training*. Setiap data memiliki validitas yang bergantung pada tetangga terdekatnya. Rumus yang digunakan untuk menghitung validitas pada data *training* yaitu:

$$Validitas(x) = \frac{1}{H}\sum_{i=1}^{H} S(lbl(x), lbl(N_i), (x))) \quad (4)$$

Dimana:

H = Jumlah ketetanggaan

$lbl(x)$ = Kelas χ

$lbl(N_i)$ =Kelas titik ke-i yang terdekat dari χ

Dimana S menghitung kesamaan antara titik a dan data ke-b pada tetanggga terdekat dengan menggunakan persamaan:

$$S(a,b) = \begin{cases} 1, jika\ a=b \\ 0\ jika\ a \neq b \end{cases} \quad (5)$$

dimana *a* merupakan kelas *a* pada data *training* dan *b* merupakan kelas selain *a* pada data *training*.

5. *Weight voting*
Perhitungan ini menggunakan K tetangga terdekat yang merupakan variasi metode *K-Nearest Neighbor*. Selanjutnya dilakukan validitas dari setiap data *training* yang akan dikalikan dengan *weight voting* berdasarkan jarak pada setiap tetangganya. Perhitungan *weight voting* dilakukan dengan persamaan:

$$W_{(i)} = Validitas(i) \times \frac{1}{d + 0{,}5} \quad (6)$$

Dimana:

$W_{(i)}$ = *Weigh Voting*

$Validitas(i)$ = Validasi Data

d = Jarak Euclidean

0,5 = Nilai Regulator Smoothing

6. Menentukan kelas dari data *testing* dengan memilih bobot terbesar sesuai dengan nilai *k.*

(Parvin, dkk, 2010).

Standarisasi Data

Variabel dengan nilai yang besar memiliki pengaruh yang lebih besar dalam melakukan prediksi klasifikasi daripada variabel dengan nilai yang kecil. Untuk mengatasi masalah tersebut, digunakan teknik standarisasi sehingga semua variabel berada pada jangkauan yang sama dan tidak ada variabel yang memiliki pengaruh dominasi terhadap variabel lainnya. Untuk menghitung standarisasi data dapat menggunakan rumus :

$$\bar{X}_1 = \frac{1}{N}\sum_{i=1}^{N} x_{i1} \quad (7)$$

$$\sigma_i^2 = \frac{1}{N-1}\sum_{i=1}^{n} (X_{i1} - \bar{X}_1)^2 \quad (8)$$

$$\hat{X}_{ik} = \frac{x_{ik} - \bar{x}_k}{\sigma_k} \quad (9)$$

Dimana:

x_{ik} = data ke-i pada variabel ke-k dimana k =1,2,..,r

$\bar{x}_k$ = rata-rata pada variabel-k

σ_k = standar deviasi

$\hat{X}_{ik}$ = normalisasi data ke-i variabel ke-k

Akurasi Prediksi

Sebuah sistem yang melakukan klasifikasi diharapkan dapat melakukan klasifikasi semua set data dengan benar, tetapi tidak dipungkiri bahwa kinerja suatu sistem tidak bisa 100% benar sehingga sebuah sistem klasifikasi juga harus diukur kinerjanya (Rodiansyah, 2013).

Untuk menghitung akurasi prediksi digunakan rumus:

$$a_{ij} = \frac{Jumlah\,data\,yang\,diprediksi\,benar}{b} \times 100\% \quad (10)$$

Dimana:

a_{ij} = akurasi untuk subset ke-i dan K-NN ke-j

b = banyak data di dalam satu subset data *testing*

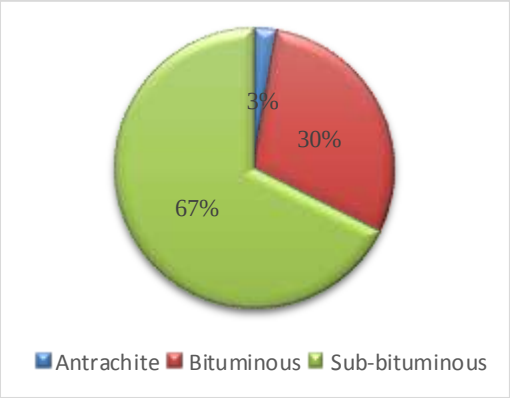
Klasifikasi Batubara

Klasifikasi batubara digunakan untuk menggolongkan batubara berdasarkan pemanfaatannya. Klasifikasi penting untuk menjadi sarana komunikasi masing-masing sektor. Secara luas, klasifikasi batubara terdiri dari aspek komersial dan aspek ilmiah. Klasifikasi batubara untuk kepentingan ilmiah antara lain mencakup genesa batubara dan *ranknya*, sedangkan untuk kebutuhan komersil antara lain nilai perdagangan

dan pemanfaatannya. Klasifikasi batubara menurut kalori di PT. Pancaran Surya Abadi adalah *Anthracite, Bituminous, Sub-Bituminous.*

Hasil dan Pembahasan

1. Statistika Deskriptif



Gambar. 1 Diagram Klasifikasi Batubara

Berdasarkan gambar 1 PT. Pancaran Surya Abadi melakukan *sampling* batubara sebanyak 37 kali. Hasil *sampling* tersebut ternyata didominasi oleh batubara dengan jenis *sub-bituminous* sebanyak 25 data atau sebanyak 67% dari jumlah keseluruhan data yang ada.

2. Standarisasi data

Langkah standarisasi data diawali dengan merandomisasi ke 37 data, kemudian digunakan persamaan (7), (8),(9)

3. Membagi data *training* dan *testing*

Langkah selanjutnya adalah membagi data ke dalam 2 bagian yakni data *training* dan *testing.*

Tabel 1. Data *Training*

Sampel	TM	FM	Ash	VM	FC	TS	GCV	Klasifikasi
A-31	1,42	-0,18	-0,52	1,12	0,014	2,88	0,8	3
A-32	1,03	0,42	-0,4	1,048	-0,33	2,96	0,7	3
A-33	0,48	0,11	-0,68	0,437	0,487	2,93	0,4	3
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
A-61	0,204	-0,76	0,006	0,253	0,348	-0,51	-0,1	3
A-62	1,209	-0,44	0,27	0,934	-0,54	-0,37	-0,2	2
A-63	0,278	-1,21	-0,57	0,551	0,929	-0,73	0,9	3

Tabel 2. Data *Testing*

Sampel	TM	FM	Ash	VM	FC	TS	GCV	Klasifikasi
A-64	1,167	-0,17	0,302	0,487	-0,38	0,56	-0,1	2
A-65	1,01	-0,17	0,338	0,228	-0,24	0,56	-0,1	2
A-66	1,04	-0,98	0,647	0,773	-0,57	0,26	0,02	2
A-67	1,94	-0,97	-0,14	0,519	0,404	-0,07	0,3	3

4. Penentuan Nilai k Optimal Dengan *k-Fold Cross Validation*

Setelah diketahui jumlah data *training* dan data *testing*, maka langkah selanjutnya adalah menentukan nilai k optimal dengan menggunakan *10-Fold Cross Validation* agar dapat diperoleh nilai k yang optimal pada klasifikasi MKNN.

Langkah -langkah dalam menentukan nilai K optimal dengan 10-*Fold Cross Validation* ada 4 yakni :

a. Menentukan *subset* data

Untuk 10-*Fold Cross Validation*, jumlah data dalam satu *subset* dapat dihitung dengan menggunakan Persamaan (2)

$$b = \frac{n}{k} = \frac{33}{10} = 3,3$$

Berdasarkan perhitungan tersebut, jumlah data dalam satu *subset* adalah tiga atau empat data. Setiap *subset* masing-masing memiliki giliran untuk dijadikan sebagai data *testing*. Apabila satu *subset* berlaku sebagai data *testing* maka 9 *subset* lainnya akan berlaku sebagai data *training*. Contoh data *training* dan data *testing* dapat dilihat pada Tabel. 3 dan Tabel. 4

Tabel. 3 Data *Training Subset* 2 sampai 10

Subset 2

Sampel	TM	FM	ASH	VM	FC	TS	GCV	Klasifikasi
A-40	-0,56	0,79	-0,64	0,56	0,07	0,39	0,11	3
A-62	1,20	-0,43	0,26	0,93	-0,54	-0,37	-0,16	2
A-36	-1,17	-1,15	-0,22	-0,05	0,98	0,98	0,42	3

⋮

Subset 10

Sampel	TM	FM	ASH	VM	FC	TS	GCV	Klasifikasi
A-57	0,10	-0,69	-0,5	0,78	0,66	-0,70	0,65	3
A-58	0,01	0,18	-0,42	0,14	0,42	-0,70	-0,08	3
A-55	0,12	1,67	0,17	-0,30	-0,54	-0,51	-3,10	2
A-32	1,03	0,42	-0,4	1,04	-0,33	2,96	0,65	3

b. Menghitung Jarak Euclidean *Subset* 1

Proses perhitungan jarak Euclidean pada *Subset* 1 dengan menggunakan Persamaan (2)

Tabel.4 Data *Testing* Pertama *Subset* 1

Subset 1

Sampel	TM	FM	ASH	VM	FC	TS	GCV	Klasifikasi
A-31	1,41	-0,17	-0,52	0,56	0,01	2,8	0,84	3
A-53	0,38	0,03	0,23	0,93	-0,31	-0,2	-0,16	2
A-38	-0,48	-0,02	-0,30	-2,72	-0,99	-0,4	0,05	3

- Perhitungan jarak Euclidean data *training* sampel A-40 dengan data *Testing* sampel
- A-31. Data *training* sampel A-40 dinyatakan sebagai x dan data *Testing* sampel A-31 dinyatakan sebagai y

$$d(x,y) = \sqrt{((-0,569)-(-0,481))^2 + \cdots + (0,1106-(-0,052))^2}$$

$$= 3,7$$

Jarak Euclidean dihitung hingga data *training* sampel A-38 *Subset* 10.

Kemudian dilakukan perangkingan pada setiap perhitungan jarak Euclidean.

c. Prediksi Klasifikasi Sesuai Perhitungan Jarak Euclidean Pada *Subset* 1

Dalam menentukan prediksi klasifikasi perlu dilakukan *voting* kelas pada data yang

telah diurutkan mulai dari jarak Euclidean terkecil hingga yang terbesar. Tujuan dilakukan *voting* kelas adalah untuk memilih klasifikasi yang paling banyak muncul (modus).

Tabel. 5 Perbandingan Prediksi Klasifikasi dengan Data Asli untuk 10- *Fold Cross Validation Subset* 1

Data *Testing*	Hasil Prediksi Pada Nilai K					Klasifikasi pada data asli
	1	3	5	7	9	
A-31	3	3	3	3	3	3
A-53	2	2	3	2	2	3
A-38	3	2	2	2	3	2
Prediksi Benar	1	2	3	2	1	

Berdasarkan Tabel 5 dapat diketahui bahwa angka yang bercetak tebal merupakan angka yang memiliki perbedaan dengan klasifikasi data asli. Semakin banyak prediksi yang sama dengan data asli maka nilai K tersebut menjadi lebih baik atau optimal untuk digunakan dalam prediksi klasifikasi.

d. Akurasi Hasil Prediksi Klasifikasi *Subset* 1

Perhitungan akurasi prediksi klasifikasi dihitung dengan menggunakan persamaan (10)

$$a_{11} = \frac{jumlah\ data\ yang\ diprediksi\ benar}{banyak\ data\ dalam\ satu\ subset\ data\ testing}$$

- Perhitungan akurasi dengan 1-NN

$$a_{11} = \frac{1}{3} = 0,333$$

Tabel.6 Akurasi Prediksi *Subset* 1 untuk 1,3,5,7,9-NN

Data *Testing*	Hasil Prediksi Pada Nilai K					Klasifikasi pada data asli
	1	3	5	7	9	
A-31	3	3	3	3	3	3
A-53	2	2	3	2	2	3
A-38	3	2	2	2	3	2
Prediksi Benar	1	2	3	2	1	

Perhitungan jarak Euclidean *subset* 1, dan akurasi hasil prediksi klasifikasi dilanjutkan untuk perhitungan *subset* 2 hingga *subset* 10. Kemudian dicari nilai rata-rata untuk setiap *subset*. Hasil Perhitungan rata-rata akurasi 10 *subset* dapat dilihat sebagai berikut:

- Presentase akurasi hasil prediksi dengan K = 1

$$a_1 = \frac{0,333+1+1+0,67+1+1+1+0,5+0,75+1}{10} = 0,825 \times 100\% = 82,5\%$$

Tabel. 7 Akurasi Tiap Subset

Nilai k	*Subset*										Rata-rata	Akurasi (%)
	1	2	3	4	5	6	7	8	9	10		
1	0,33	1	1	0,67	1	1	1	0,5	0,75	1	0,825	82,5
3	0,67	0,67	1	1	1	1	1	0,5	0,5	1	0,834	**83,4**
5	1	1	1	1	0,67	0,67	1	0,5	0,75	1	0,792	79,2
7	0,67	0,33	1	1	0,67	1	1	0,5	0,75	1	0,792	79,2
9	0,33	0,33	1	1	1	0,67	1	0,5	0,75	1	0,758	75,8

Berdasarkan Tabel 7, dapat dilihat bahwa nilai k optimal terdapat pada nilai K=3 karena memiliki akurasi tertinggi, dengan presentase sebesar = 83,4%. Hal ini menyatakan bahwa nilai K=3 akan digunakan dalam Algoritma *Modified K-Nearest Neighbor* (MKNN).

Algoritma *Modified K-Nearest Neighbor* (MKNN)

1. **Penentuan nilai K tetangga terdekat**
   Pada penelitian ini diperoleh nilai K=3
2. **Perhitungan jarak Euclidean antar data *training***
   Dipilih 9 data training baru secara acak sesuai Tabel 8.

Tabel. 8 Data *Training*

Sampel	TM	FM	Ash	VM	FC	TS	GCV	Klasifikasi
A-31	1,4179	-0,1784	-0,524	1,1204	0,0136	2,879	0,85	3
A-32	1,0303	0,42055	-0,403	1,0483	-0,353	2,962	0,659	3
A-33	0,4802	0,11156	-0,68	0,4372	0,4872	2,934	0,409	3
A-34	-1,381	-0,392	-0,281	0,2101	0,4976	-0,26	-0,1	3
A-46	1,2624	1,33991	0,0021	0,9117	0,0728	-0,4	-0,48	2
A-47	-0,917	1,01947	-0,178	0,7427	0,0275	-0,21	-0,22	2
A-48	1,5177	1,87017	-0,036	0,716	-0,335	-0,62	-0,62	2
A-49	-0,499	3,19389	5,5844	-0,181	-4,969	-0,68	-1,1	2
A-51	-1,312	0,18022	0,0793	-2,815	0,2365	-0,35	2,798	1

Tabel. 9 Data *Testing*

Sampel	TM	FM	Ash	VM	FC	TS	GCV	Klasifikasi
A-64	1,167	-0,17	0,302	0,487	-0,38	0,56	-0,1	2
A-65	1,01	-0,17	0,338	0,228	-0,24	0,56	-0,1	2
A-66	1,04	-0,98	0,647	0,773	-0,57	0,26	0,02	2
A-67	1,94	-0,97	-0,14	0,519	0,404	-0,07	0,3	3

Berikut ini perhitungan jarak Euclidean pada data *training* pertama yaitu d(1,2) perhitungan ini dilakukan sampai menghitung jarak ke d(8,9) dengan menggunakan persamaan (1)

$$d(x,y) = \sqrt{\sum_{i=1}^{7}(x_{1i} - y_{2i})^2} \qquad (1)$$

$$(10,1) = \sqrt{((1,167-1,4179)^2 + \cdots + (-0,11-0,85)^2)} = 2,754$$

Tabel.10 Jarak Euclidean Antar Data *Training*

d	Jarak Euclidean
(1,2)	0,832
(1,3)	1,369
(1,5)	3,895
(1,4)	4,223
(1,6)	4,443
(1,7)	4,726
(1,8)	6,140
(1,9)	9,764

Tabel. 11 Jarak Euclidean Terdekat Sesuai Nilai *K* Pada Data *Training* Pertama

d	Jarak Euclidean
(1,2)	0,832
(1,3)	1,369
(1,5)	3,895

3. Perhitungan validasi data *training*

Berikut adalah perhitungan untuk mencari nilai validasi data *training* pertama

$$Validitas(1) = \frac{1}{3}\sum_{i=1}^{H} S(lbl(1,2),(lbl(1,3),(lbl(1,5))))$$

$$Validitas(1) = \frac{1}{3} \times (1+1+0) = 0,667$$

Dari perhitungan validitas data *training* pertama diperoleh nilai 0,667 dimana nilai tersebut didapatkan berdasarkan kedekatan dari jarak data *training* (1,2), (1,3) dan (1,5).

4. Perhitungan jarak Euclidean data *training* dengan data *testing*

$$d(x,y) = \sqrt{\sum_{i=1}^{7}(x_{1i} - y_{2i})^2} \qquad (1)$$

$$(10,1) = \sqrt{((1{,}167-1{,}4179)^2 + \cdots + (-0,11-0,85)^2)} = 2,754$$

5. Perhitungan *weight voting*

$$W(1) = Validitas(1) \times \frac{1}{d(10,1)+0,5} \qquad (6)$$

$$= 0,667 \times \frac{1}{2,754+0,5} = 0,204$$

Tabel. 12 Perhitungan *Weight Voting* untuk data *testing* A-64

		A-64		
No.	Nilai Validasi	d(x,y)		W
1	0,667	(10,1)	2,754	0,204
2	0,667	(10,2)	2,744	0,205
3	0,667	(10,3)	2,858	0,198
4	0,667	(10,4)	2,899	0,097
5	1	(10,5)	3,960	0,271
6	1	(10,6)	2,616	0,214
7	1	(10,7)	2,743	0,205
8	1	(10,8)	8,125	0,115
9	0	(10,9)	5,186	0

6. Penentuan kelas dari data *testing*

Penetuan klasifikasi dari data *testing* dilakukan dengan mengurutkan nilai *Weight Voting* dari yang terbesar kemudian diambil sebanyak nilia *k*=3.

Tabel. 13 Perhitungan *Weight Voting* Yang Telah Diurutkan sesuai nilai *K*

No.	d(x,y)	*W*	Klasifikasi Awal	Klasifikasi A-64
1	d(10,5)	0,271	2	2
2	d(10,6)	0,214	2	
3	d(10,7)	0,205	2	
Jumlah		0,690		

Berdasarkan Tabel 11 diketahui bahwa nilai d(10,5) adalah nilai *weight* terbesar 0,271 pada data *testing* sampel A-64. Kemudian nilai keseluruhan *weight* dijumlahkan untuk mengetahui total dari nilai *weight* sampel A-64, karena dari ketiga nilai data *training* memiliki klasifikasi 2 (*bituminous*) maka dapat disimpulkan kelas data *testing* sampel A-64 adalah 2 (*bituminous*). Perhitungan yang sama dilakukan pada ketiga data *testing* yang lain yakni sampel A-65, A-66, A-67.

7. Menentukan akurasi prediksi MKNN

Tabel.14 Perbandingan Klasifikasi Awal dan MKNN

Sampel	W	Klasifikasi MKNN	Klasifikasi Awal
A-64	0,690	2	2
A-65	0,473	2	2
A-66	0,402	2	2
A-67	0,392	3	3

Menentukan akurasi prediksi MKNN dilakukan dengan melihat kesamaan antara klasifikasi awal dan prediksi yang berdasarkan dari nilai *weight voting*. Presentase akurasi dihitung dengan menggunakan persamaan (10)

$$presentase\ akurasi = \frac{4}{4} \times 100\% = 100\%$$

Berdasarkan perhitungan akurasi prediksi tersebut dapat diketahui bahwa presentase akurasi prediksi klasifikasi batubara di PT. Pancaran Surya Abadi dengan menggunakan Algoritma *Modified K-Nearest Neighbor* dengan nilai *K=3* pada data *testing* adalah 100%.

Kesimpulan

Berdasarkan hasil analisis maka hasil penelitian ini dapat disimpulkan sebagai berikut:

1. Nilai K optimal yang digunakan untuk prediksi klasifikasi batubara di PT. Pancaran Surya Abadi dengan menggunakan metode Algoritma *Modified* K*-Nearest Neighbor* berdasarkan perhitungan 10 *Fold Cross Validation* adalah 3-NN.
2. Presentase akurasi prediksi batubara di PT. Pancaran Surya Abadi menggunakan

3. Algortima *Modified* K*-Nearest Neighbor* dengan nilai K = 3 pada data *testing* adalah sebesar 100%.

Daftar Pustaka

Banjarsari, Mutiara Ayu. (2015). Penerapan K-optimal pada Algoritma K-NN untuk Prediksi Kelulusan Tepat Waktu Mahasiswa Program Studi Ilmu Komputer Fmipa Unlam Berdasarkan IP sampai dengan 4 Semester. *Jurnal Ilmu Komputer* (*KLIK*) (2).

Dzikrulloh, Nihru Nafi. (2017). Penerapan Metode K-Nearest Neighbor (K-NN) dan Metode Weighted Product (WP) Dalam Penerimaan Calon Guru dan Karyawan Tata Usaha Baru Berwawasan Teknologi. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer* ISSN: 2548-964X.

Irwandy, Arif. (2014). *Batubara Indonesia.* Jakarta: PT Gramedia Pustaka Utama

Mitchell, Tom M. (2008). *Machine Learning*. New York : McGraw-Hill Science.

Pandie, Emerensye. (2012). Implementasi Algoritma Data Mining K-Nearest Neighbor (KNN) dalam pengambilan Keputusan Pengajuan Kredit. *Jurnal Ilmu Komputer Universitas Nusa Cendana.*

Parvin, H., Alizadeh, H., & Minati, B. (2008). MKNN : Modified K-Nearest Neighbor. *Proceedings of the World Congress on Engineering and Computer Science*, WCECS, 22–24.

Parvin, H., Alizadeh, H., & Minati, B. (2010). A Modification on K-Nearest Neighbor Classifier. *Global Journal of Computer Science and Technology*. Vol. 10. hal.37-41.

Prasetyo, Eko (2014). *Data Mining Mengolah Data Menjadi Informasi Menggunakan Matlab Edisi.I*. Yogyakarta: Penerbit Andi.

Pratiwi, Retno., Sri Wahyuningsih., Fidia Deny.T.A. (2017). Klasifikasi Batubara Berdasarkan Jenis Kalori Dengan Menggunakan Algoritma K-Nearest Neighbor (K-NN). *Prosiding Seminar Nasional Matematika, Statistika, dan Aplikasinya 2017*. ISBN: 978-602-50321-0-3.

Rodiyansyah, Sandi F. (2013). Klasifikasi Posting Twitter Kemacetan Lalu Lintas Kota Bandung Menggunakan *Naive Bayesian Classification*. *Jurnal Universitas Pendidikan Indonesia.* Vol.07, No.01, hal.13-22.

Sukandarrumidi. (2017). *Batubara dan Pemanfaatannya: Pengantar Teknologi Batubara Menuju Lingkungan Bersih Edisi Ketiga.* Yogyakarta: UGM Press.

Tampubolon, Kennedi., Saragih, Hoga, dan Reza, Bobby. (2013). Implementasi Data Mining Algoritma Apriori pada Sistem Persediaan Alat-alat Kesehatan . *Informasi dan Teknologi Ilmiah* (*INTI*) ISSN: 2339-210X.