

Clustering Application for UKT Determination Using Pillar K-Means Clustering Algorithm and Flask Web Framework

¹Ahmad Luky Ramdani, ²Hafiz Budi Firmansyah

^{1,2}Department of Informatic, Institut Teknologi Sumatera, Indonesia

Email: ¹ahmadluky@if.itera.ac.id, ²hafiz.budi@if.itera.ac.id

Article Info

Article history:

Received May 13th 2018

Revised Jun 16th 2018

Accepted Jul 05th 2018

Keywords:

K-means

Pillar

Silhouette Coefficient

Uang Kuliah Tunggal

ABSTRACT

Clustering is one of technique in data mining which has purpose to group data into a cluster. At the end, a cluster will have different data compared with others. This paper discussed about the implementation of clustering technique in determining UKT (Uang Kuliah Tunggal) / Tuition Fee in Indonesia. UKT is a tuition fee where its amount is determined by considering students purchasing power. Most of University in Indonesia often use manual technique in order to classify UKT's group for each student. Using web-based application, this paper proposed a new approach to automatise UKT's grouping which leads to give a reasonable recommendation in determining the UKT's group. Pillar K-Means algorithm had been implemented to conduct data clustering. This algorithm used pillar algorithm to initiate centroid value in K-means algorithm. By deploying students data at Institut Teknologi Sumatera Lampung as case study, the result illustrated that Pillar K-Means and silhouette coefficient value might be adopted in determining UKT's group.

Copyright © 2018 Puzzle Research of Data Technology

Corresponding Author:

Ahmad Luky Ramdani

Department of Informatic

Institut Teknologi Sumatera

Bandar Lampung, Indonesia

Email: ahmadluky@if.itera.ac.id

1. PENDAHULUAN

Uang kuliah tunggal (UKT) merupakan bagian biaya kuliah tunggal (BKT) yang ditanggung kepada setiap mahasiswa berdasarkan kemampuan ekonominya dalam pembiayaan pendidikan di suatu perguruan tinggi. Nominal UKT ditetapkan berdasarkan BKT dikurangi dengan biaya yang ditanggung oleh pemerintah yang akan dilaksanakan oleh perguruan tinggi negeri. Kebijakan ini bertujuan untuk mengurangi biaya perkuliahan yang dibayarkan oleh mahasiswa dari awal masuk perkuliahan hingga lulus. Tata cara penggunaan dan penetapan nominal UKT telah diatur oleh undang-undang sesuai dengan kebutuhan operasional perguruan tinggi negeri dalam menyelenggarakan kegiatan perkuliahan. Undang-undang yang dimaksud salah satunya adalah undang-undang No.12 Tahun 2012 [1] tentang pendidikan tinggi negeri terkait bantuan operasional perguruan tinggi negeri (BOPTN), BKT dan UKT.

Mekanisme penentuan UKT umumnya masih dilakukan secara manual dan membutuhkan waktu yang cukup lama dalam menganalisis data mahasiswa baru. Berangkat dari permasalahan tersebut, dibutuhkan sebuah mekanisme penentuan UKT yang dapat dilakukan dengan mudah. Penelitian ini bertujuan untuk mengembangkan sebuah aplikasi berbasis web yang membantu dalam pengelompokan data mahasiswa dalam menentukan UKT. Teknik *clustering* tertentu dirasa dapat diimplementasikan dalam menentukan kelompok UKT bagi mahasiswa baru. *Clustering* merupakan salah satu teknik dalam data mining di mana pengguna dapat mengelompokkan sejumlah data atau objek ke dalam suatu *cluster* (group). Di akhir proses, *cluster* tersebut akan berisi data yang mirip dan berbeda dengan objek pada *cluster* yang lainnya.

Algoritma data mining yang banyak digunakan untuk teknik *clustering* adalah K-Means. Algoritma K-Means mempunyai kemampuan mengelompokkan data dalam jumlah yang cukup besar dengan waktu komputasi yang relatif cepat dan efisien [2]. Namun, algoritma K-Means memiliki kelemahan yang disebabkan oleh penentuan pusat awal *cluster* (*centroid*). Hasil *cluster* yang terbentuk dari algoritma K-Means ini sangat

tergantung pada inisiasi nilai *centroid* yang diberikan [2]. Sehingga penggunaan algoritma Pillar menjadi salah satu alternatif untuk mengoptimalkan penentuan nilai *centroid* pada algoritma K-Means[3]. Metode ini disebut sebagai algoritma Pillar K-Means. Penggunaan algoritma Pillar pada K-Means menghasikan kinerja lebih baik dalam proses *clustering* [3]. Penelitian sebelumnya yang menggunakan metode Pillar K-Means diantaranya pada kasus segmentasi citra [4] dan pengelompokan data proposal pengajuan kredit [5]. Selain algoritma Pillar K-Means proses optimasi nilai *centroid* juga dilakukan pada K-Means++ [6]. Namun, algoritma K-Means++ tidak mempunyai mekanisme untuk mendeteksi *centroid* yang berupa *outlier* dari data.

Dalam penelitian ini ukuran data yang digunakan berukuran relatif besar. Sehingga, implementasi algoritma Pillar K-Means, analisis dan presentasi data pada proses *clustering* membutuhkan bahasa pemrograman dengan kinerja tinggi. Oleh karena itu, dalam penelitian ini digunakan Flask Web Framework dengan bahasa pemrograman python. Python merupakan bahasa pemrogram yang populer untuk analisis data [6]. Flask menggunakan python sebagai bahasa pemrograman yang menyediakan konten web yang interaktif.

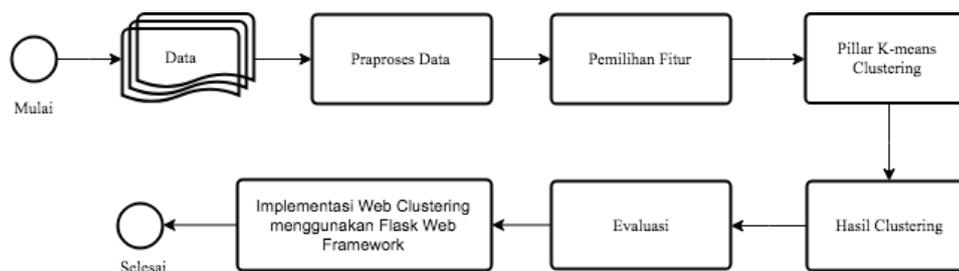
2. DATA DAN METODE

2.1 Data

Data yang digunakan dalam penelitian ini adalah data mahasiswa Institut Teknologi Sumatera pada tahun akademik 2015, 2016 dan 2017. Data mahasiswa yang digunakan meliputi identitas diri, kebutuhan khusus, orang tua mahasiswa, keluarga, wali, penghasilan orang tua, kondisi rumah, kepemilikan asset, penggunaan/tagihan listrik, barang elektronik, tagihan air dan kredit orang tua yang bersumber dari 153 pertanyaan yang dijawab secara online.

2.2 Metode

Pada bagian ini diuraikan tahapan-tahapan sistematis yang akan dijadikan acuan sebagai kerangka penelitian. Tahapan penelitian terdiri dari praproses data, pemilihan fitur, *clustering* dengan algoritma Pillar K-Means, evaluasi dan implementasi. Diagram alir metode penelitian dapat dilihat pada Gambar 1.



Gambar 1. Diagram alir penelitian

2.2.1. Praproses Data

Praproses data merupakan proses menormalisasi data di mana tahapan ini terdiri dari filtering atau standarisasi data, konversi tipe data misalkan nominal ke kategori dan sebaliknya, mengganti data yang bernilai null dan penghapusan data.

2.2.2. Pemilihan Fitur

Berdasarkan data yang telah diisi oleh mahasiswa, kemudian diambil beberapa fitur utama untuk menggambarkan kemampuan ekonomi dari keluarga mahasiswa. Hal ini didasari kepada ketentuan di Institut Teknologi Sumatera (ITERA). Adapun fitur tersebut antara lain:

1. Perkerjaan ayah dan ibu
2. Penghasilan ayah dan ibu.
3. Biaya tanggungan anak
4. Tagihan PBB
5. Tagihan listrik
6. Tagihan telepon
7. Tagihan air

Nilai ketujuh fitur tersebut kemudian divalidasi dengan dokumen yang diugah oleh mahasiswa baru yang kemudian ditentukan index dari setiap nilai yang mungkin. Fitur biaya tanggungan anak dijadikan faktor penentu nilai index pada penghasilan ayah dan ibu. Sementara itu, nilai index pekerjaan ayah dan ibu ditentukan dengan nilai index maksimal antara pekerjaan ayah dan ibu.

Tabel 1. Nilai index data UKT

Pekerjaan ayah dan ibu (Poin1)			
1.	Pejabat Tinggi Negara, Rektor, Anggota Legislatif, Guru Besar, Perwira Tinggi, Profesional, Pejabat Eselon I, Kepala Daerah, Pengusaha Besar dan Menengah, Direksi, Manager, Dokter Spesialis, Pengacara, Komisaris. (nilai 8)		
2.	Pejabat Eselon II, Dekan, Kepala Biro, Kepala Badan, Asisten Manager, Kepala Dinas, Perwira Menengah, Pengusaha Kecil, Pilot, Nahkoda Kapal, Karyawan BUMN. (nilai 7)		
3.	Pejabat Eselon III, Supervisor, Kabag, PNS Gol IV, Camat, Perwira Pertama, Pengusaha Mikro, Karyawan Bank. (nilai 6)		
4.	Pejabat Eselon IV, Bintara Tinggi, Kasubag, PNS Gol III, Lurah/kepala Desa, Bidan. (nilai 5)		
5.	Pedagang besar, Perternak besa. (nilai 4)		
6.	Bintara, PNS Gol II, Kaur, Kasi, Pensiunan PNS Gol III, Pensiunan PNS Gol IV, Perawat, Masinis, Anak Buah Kapal. (nilai 3)		
7.	Tamtama, PNS Gol I, Pensiunan PNS Gol II, Pengemudi, Petani. (nilai 2)		
8.	Anak Yatim Piatu, Ibu Rumah Tangga, Kuli Bangunan, Tukang Parkir, Buruh Tani, Tukang Becak, Pensiunan PNS Gol I, PKL, Pembantu Rumah Tangga, buruh/pelayan, satpam, cleaning service, pemulung, sopir kendaraan umum, Tukang, Nelayan. (nilai 1)		
Penghasilan kotor ayah dan ibu (Poin2)	Tagihan PBB (Poin3)		
1.	>= 20.000.000 (nilai 8)	1.	> 600.000 (nilai 8)
2.	15.000.001 - 20.000.000 (nilai 7)	2.	500.001 - 600.000 (nilai 7)
3.	10.000.001 - 15.000.000 (nilai 6)	3.	300.001 - 500.000 (nilai 6)
4.	6.500.001 - 10.000.000 (nilai 5)	4.	200.001 - 300.000 (nilai 5)
5.	4.000.001 - 6.500.000 (nilai 4)	5.	150.001 - 200.000 (nilai 4)
6.	2.500.001 - 4.000.000 (nilai 3)	6.	100.001 - 150.000 (nilai 3)
7.	1.740.001 - 2.500.000 (nilai 2)	7.	60.001 - 100.000 (nilai 2)
8.	<= 1.740.000 (nilai 1)	8.	<= 60.000 (nilai 1)
Tagihan listrik (Poin4)	Tagihan telepon dan air (Poin5 dan Poin6)		
1.	> 600.000 (nilai 8)	1.	>300.000 (nilai 8)
2.	500.001 - 600.000 (nilai 7)	2.	250.001 - 300.000 (nilai 7)
3.	400.001 - 500.000 (nilai 6)	3.	150.001 - 250.000 (nilai 6)
4.	300.001 - 400.000 (nilai 5)	4.	100.001 - 150.000 (nilai 5)
5.	200.001 - 300.000 (nilai 4)	5.	80.001 - 100.000 (nilai 4)
6.	100.001 - 200.000 (nilai 3)	6.	60.001 - 80.000 (nilai 3)
7.	50.001 - 100.000 (nilai 2)	7.	40.001 - 60.000 (nilai 2)
8.	<= 50.000 (nilai 1)	8.	<= 40.000 (nilai 1)

Luaran yang dihasilkan pada tahapan ini yaitu, data dengan dimensi $n \times 6$, dengan n adalah jumlah mahasiswa dan nilai 6 adalah jumlah fitur (fitur biaya tanggungan anak dijadikan faktor penentu index penghasilan ayah dan ibu). Data pada tahapan ini kemudian disimpan pada basis data MySQL yang kemudian dilakukan analisis *clustering*. Pembagian kelompok UKT didasari kepada sebaran ketentuan di ITERA, yaitu UKT golongan 1 sampai 8. Tabel 2 ditampilkan contoh data hasil dari proses pemilihan fitur.

Tabel 2. Data hasil tahapan pemilihan fitur dengan $n = 4$

No. Pendaftaran	Fitur1	Fitur 2	Fitur 3	Fitur 4	Fitur 5	Fitur 6
1153020351	1	1	1	1	2	1
1151111592	6	6	4	5	2	1
1151114392	1	4	5	2	6	4
1151400217	1	1	2	2	2	2

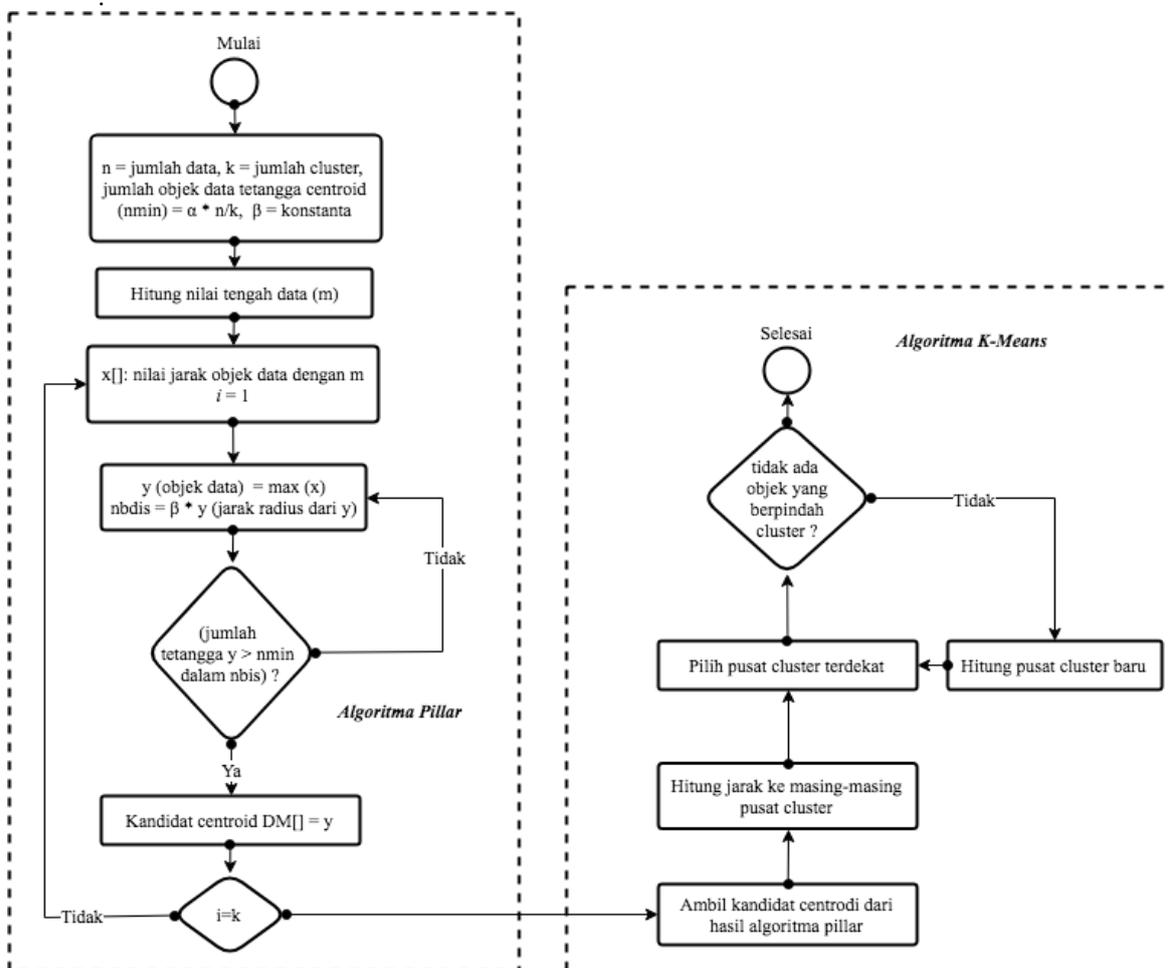
2.2.3. Pillar k-means clustering

Pada tahapan ini dilakukan pengelompokan data menggunakan algoritma Pillar dan K-Means. Algoritma pillar digunakan untuk memperbaiki kelemahan pada algoritma K-Means, yaitu dalam menentukan inisialisasi *centroid*. Selanjutnya, *centroid* yang diperoleh digunakan untuk proses pengelompokan data dengan algoritma K-Means. Gambar 2 adalah diagram alir yang menjelaskan tahapan *clustering* dengan menggunakan algoritma Pillar K-Means.

Algoritma Pillar terinspirasi oleh fungsi pilar pada bangunan atau konstruksi. Ide dasar algoritma Pillar, bahwa sebuah pilar pada bangunan ditempatkan di setiap tepi atau sudut pada sebuah bangunan, sehingga massa bangunan terkonsentrasi di setiap pilar [3]. Ide yang sama diadopsi untuk proses *clustering* suatu data. Bahwa dalam proses *clustering*, *centroid* awal terbaik diperkirakan ada di tepi data atau objek-objek terluar dalam suatu data namun bukan merupakan *outlier* dari data.

Pada algoritma Pillar, sebelum dilakukan proses pencarian kandidat *centroid*, terlebih dahulu ditentukan beberapa variabel utama yaitu k (jumlah *cluster*), $nmin$ (jumlah tetangga *centroid*), konstanta α dan β . Nilai α dan β digunakan untuk menentukan $nmin$ (jumlah data yang berdekatan dengan *centroid*) dan $nbdis$ (jarak radius dari *centroid*). Nilai $nmin$ dan $nbdis$ digunakan untuk memastikan bahwa *centroid* bukan merupakan *outlier* data. Kemudian dihitung nilai tengah (m) dari data, dengan menggunakan persamaan jarak, hitung jarak antara nilai tengah dengan setiap objek data (X). Persamaan jarak yang digunakan adalah jarak Euclidian [7]. Berdasarkan hasil perhitungan jarak, urutkan secara menurun X . Nilai maksimal dari X adalah calon kandidat *centroid*, yang merupakan objek dari data. Hitung nilai $nbdis$, berdasarkan nilai $nbdis$ dan $nmin$ periksa apakah jumlah tetangga dari X pada jarak $nbdis$ lebih besar dari $nmin$. Jika tidak, ambil nilai maksimal

dari X berikutnya, lakukan pengecekan terhadap nilai n_{bdis} dan n_{min} dan jika ya, maka X adalah kandidat centroid (DM[]). Nilai tengah berikutnya adalah X. Hal ini akan terus dilakukan sampai k iterasi. Pada akhir tahap algoritma Pillar akan diperoleh objek data yang merupakan kandidat *centroid*.



Gambar 2. Algoritma pillar k-means

Setelah sampai pada iterasi ke- k , maka dilanjutkan dengan algoritma K-Means. Pada tahapan ini, proses diawali dengan mendapatkan kandidat *centroid* berdasarkan hasil algoritma Pillar (DM[]). Selanjutnya, dihitung jarak objek data ke setiap *centroid* dengan menggunakan persamaan jarak Euclidian [7]. Setelah didapatkan hasilnya, objek data dimasukkan ke dalam *cluster* yang memiliki jarak yang paling dekat dengan *centroid*. Iterasi pada K-Means akan berhenti ketika semua data yang berada pada sebuah *cluster* tertentu tidak berpindah ke *cluster* yang lainnya atau nilai *centroid* tidak berubah. Hasil dari algoritma K-Means adalah daftar objek data beserta *cluster*-nya.

2.2.4. Evaluasi

Pada tahapan ini dilakukan pengujian terhadap hasil dari tahapan sebelumnya. Pengujian ini dilakukan untuk melihat kualitas *cluster* yang terbentuk. Pengujian dilakukan dengan menggunakan Silhouette Index. Nilai Silhouette Index suatu objek i dihitung dengan menggunakan Persamaan 1 [8].

$$s(i) = \frac{b(i) - a(i)}{\max \{a(i), b(i)\}} \quad (1)$$

dengan $a(i)$ merupakan jarak rata-rata antara objek i dengan seluruh objek yang berada dalam satu *cluster* yang sama (*intra-cluster*). $b(i)$ adalah jarak rata-rata antara objek i dengan seluruh objek yang berada pada *cluster* terdekat (*nearest-cluster*) [9]. Nilai koefisien Silhouette(SC) suatu *cluster* yang terdiri dari N jumlah objek dapat dihitung dengan rata-rata dari seluruh nilai Silhouette Index setiap objek $s(i)$ dengan menggunakan Persamaan 2 [8].

$$SC = \frac{1}{N} \sum_{i=1}^N s(i) \quad (2)$$

Nilai SC berada pada rentang nilai 1 dan -1. SC dengan nilai mendekati 1 hal ini berarti meminimumkan jarak antar objek data dalam suatu *cluster* dan sekaligus memaksimumkan jarak antar *cluster*.

2.2.4. Implementasi *Web Clustering*

Prototipe aplikasi *clustering* dibangun dalam sebuah sistem berbasis Web. Bahasa pemrograman yang digunakan adalah python 3.6 dengan menggunakan Flask sebagai web *framework* versi 1.0.2 dan library python numpy [10], sklearn [11] dan pickle [12]. Numpy dan sklearn merupakan *library* python untuk proses *pre-processing* data dan pickle untuk menyimpan model hasil proses pelatihan. Adapun lingkungan pengembangan aplikasi adalah macOS Sierra sebagai sistem operasi dan database MySQL 5.5 (Oracle 2010).

3. HASIL DAN PEMBAHASAN

3.1 Praproses Data

Data UKT diperoleh dari unit pelaksana teknis (UPT) teknologi dan informasi komunikasi ITERA yang berupa basis data relasional dari sistem informasi mahasiswa baru. Pada proses *clustering*, data dibagi menjadi dua bagian yaitu, data untuk pelatihan dan data prediksi golongan UKT. Jumlah data mahasiswa tahun akademik 2015, 2016 dan 2017 yang digunakan sebanyak 3.917. Proses prediksi menggunakan data mahasiswa baru tahun 2018 sebanyak 962.

Dalam tahapan praproses data, ditemukan bahwa data mahasiswa baru terdapat banyak data dengan nilai *null*. Hal tersebut disebabkan karena mahasiswa baru tidak mengisi data secara lengkap. Sehingga diperlukan mekanisme penggantian data bernilai *null* dengan rata-rata pada data tersebut. Sebagai contoh *null* pada data tagihan PBB, digantikan dengan rata-rata nilai tagihan PBB. Data hasil praproses kemudian dilakukan pemilihan fitur dan disimpan pada basis data MySQL.

3.2 Hasil *Clustering* dan Evaluasi

Proses *clustering* dengan menggunakan Pillar K-Means dilakukan dengan variasi nilai α , β dan k dengan sebaran nilai α dan β antara 0 dan 1. Pada penelitian, nilai k belum ditentukan karena akan diamati setiap kemungkinan nilai k dari $k=1$ hingga $k=8$ dimana nilai k didasarkan pada sebaran data. Pada penelitian ini juga diamati nilai optimal parameter α dan β pada algoritma Pillar untuk mendapatkan hasil *clustering* yang optimal. Gambar 3 adalah cuplikan kode bahasa pemrograman python untuk implementasi algoritma Pillar K-Means *clustering* beserta fungsi silhouette (Persamaan 1 dan 2) untuk evaluasi. Data disajikan dengan format *list* pada python.

```
alpha = 0.4
beta = 0.6
while (k>=1 and k<=8):
    while (alpha<1.0):
        while (beta<1.0):
            Centroidi = Pillar_Algoritma (P, k, alpha, beta)
            Solution = K_Means (k, Centroids)
            Silhouette_Score = Silhouette_Function(Solution)
            beta+=0.05
        if int(beta)>=1:
            beta=0.5
            alpha+=0.05
        if int(alpha)>=1:
            alpha=0.4
            k+=1
```

Gambar 3. Kode program pillar k-means clustering dan fungsi silhouette.

Berdasarkan hasil percobaan yang dilakukan, nilai α dan β menunjukkan peran penting dalam penentuan nilai koefisien Silhouette. Terlihat jelas bahwa penentuan nilai terendah α dan β dengan nilai 0.4 dan 0.6 memungkinkan untuk inialisasi dalam jumlah *cluster* antara 1 dan 8. Hasil percobaan pada data 3.917 data mahasiswa menghasilkan 1.578 solusi cluster yang mungkin dengan waktu yang dibutuhkan 152 menit. Akan tetapi, penelitian ini hanya mengambil *cluster* dengan nilai koefisien silhouette tertinggi untuk setiap nilai α , β , dan k seperti terdapat pada Tabel 3. Berdasarkan Tabel 3, terlihat bahwa nilai $k=3$ adalah solusi *cluster* terbaik pada proses *clustering* data.

Tabel 3. Jumlah *cluster* terbaik berdasarkan nilai koefisien silhouette

k	α	β	nilai koefisien silhouette
1	0.7	0.65	0.35789
2	0.65	0.7	0.31213
3	0.8	0.7	0.61211
4	0.6	0.9	0.52337
5	0.6	0.85	0.42294
6	0.9	0.65	0.4320
7	0.65	0.7	0.52235
8	0.8	0.6	0.34679

Sedangkan untuk penentuan golongan UKT pada jumlah *cluster* yang terbentuk, ditentukan berdasarkan observasi objek data dan koefisien silhouette pada setiap cluster. Koefisien silhouette terbesar menunjukkan UKT terkecil dan sebaliknya. Tabel 4 menunjukkan koefisien silhouette index dan golongan UKT hasil penelitian. Selain didapatkan jumlah cluster dan UKT, penelitian ini juga mendapatkan model yang berisi nilai *centroid* dari setiap *cluster* yang terbentuk. Berdasarkan nilai *centroid* tersebut, dilakukan proses prediksi UKT kemudian model disimpan dalam format data.pkl.

Tabel 4. Koefisien silhouette dan golongan UKT dengan $k=3$

cluster ke-	Koefisien silhouette index	Golongan UKT
1	0.48562	UKT 3
2	0.49110	UKT 2
3	0.53111	UKT 1

3.3 Implementasi Web Clustering pada Flask Web

Hasil dari proses *clustering* digunakan untuk mengembangkan aplikasi web penentuan UKT yang memungkinkan dilakukan secara otomatis. Pengguna dapat memilih dan menentukan data apa saja yang akan diprediksi berdasarkan data *training* yang telah ditentukan sebelumnya. Aplikasi melakukan pelatihan pada data yang telah pilih. Nilai *centroid* dari setiap cluster yang dihasilkan dari proses pelatihan kemudian digunakan untuk memprediksi golongan UKT pada mahasiswa yang terdapat pada data prediksi. Gambar 4 menunjukkan antarmuka aplikasi web untuk proses pemilihan data pelatihan dan prediksi serta hasil proses prediksi golongan UKT.

No. Pendaftaran	Nama	Program Studi	Golongan UKT
4180123175	Firman Andrianto	Teknik Mesin	3
4180401538	Nuril Huda	Teknik Industri	2
4180154204	Hasanudin	Teknik Elektro	1
4180491867	Nanang Sugandri	Perencanaan Wilayah dan Kota	2
4180187993	Ade Ayu Oktaviana	Sains Atmosfer dan Keplanetan	3
4180425875	Maulidya Paramitha	Perencanaan Wilayah dan Kota	2

Gambar 4. Antarmuka aplikasi *clustering* penentuan UKT

4. KESIMPULAN

Penelitian ini mengembangkan aplikasi *clustering* berbasis web untuk penentuan golongan UKT dengan menggunakan algoritma Pillar K-Means *clustering* dan Flask Web Framework. Hasil penelitian

menunjukkan bahwa algoritma Pillar K-Means dan nilai koefisien Silhouette dapat digunakan dalam proses penentuan golongan UKT. Hal ini didasari pada nilai koefisien Silhouette yang bernilai lebih dari 0.5. Namun, algoritma Pillar K-Means sangat ditentukan oleh nilai α , β dan k . Penelitian ini juga menunjukkan bahwa algoritma Pillar K-Means membutuhkan waktu yang relatif lama dalam pengolahan data. Sehingga diharapkan pada penelitian selanjutnya, implementasi algoritma Pillar K-Means dapat dikembangkan menggunakan konsep pemrosesan terdistribusi pada *cluster* komputer.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada UPT Teknologi dan Informasi Komunikasi, Institut Teknologi Sumatera (ITERA) yang telah menyediakan data dan memfasilitasi penelitian ini.

REFERENSI

- [1] Republik Indonesia, Undang-Undang No.12. 2012
- [2] Arai K, Barakbah AR. Hierarchical K-means: an algorithm for centroids initialization for K-means. *Reports of the Faculty of Science and Engineering*, 2007, 36.1: 25-31.
- [3] Barakbah AR, Kiyoki Y. A Pillar algorithm for k-means optimization by distance maximization for initial centroid designation. *Computational Intelligence and Data Mining, CIDM'09. IEEE Symposium on. IEEE*. 2009; 61-6
- [4] Barakbah AR, Kiyoki Y. A new approach for image segmentation using Pillar-Kmeans algorithm. *World Academy of Science, Engineering and Technology*. 2009; 59: 23-28.
- [5] Wahyudin I, Taufik D, and Wisnu AK. Cluster analysis for SME risk analysis documents based on Pillar K-Means. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*. 2016; 14(2): 674-683.
- [6] Arthur D, Sergei V. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics, 2007.
- [7] Pedregosa F. Scikit-learn: Machine learning in Python. *Journal of machine learning research*. 2011; 12: 2825-2830.
- [8] Kaufman L, Rousseeuw PJ. 1990. *Finding Groups in Data: An Introduction to Cluster Analysis*. New Jersey: John Wiley and Sons
- [9] Rousseeuw PJ. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*. 1987; 20: 53-65.
- [10] <http://www.numpy.org/>
- [11] Scikit-learn. `sklearn.metrics.silhouette_score` [internet]. [diacu 2018 Mei 10]. Tersedia dari: http://scikit-learn.org/stable/modules/generated/sklearn.metrics.silhouette_score.html
- [12] <https://docs.python.org/3/library/pickle.html>
- [13] Flask Web Framework. Flask (A Python Microframework)[internet]. [diacu 2018 Mei 10]. <http://flask.pocoo.org>

BIBLIOGRAFI PENULIS



Ahmad Luky Ramdani terdaftar sebagai staf pengajar di program studi teknik informatika, Institut Teknologi Sumatera (ITERA), Lampung. Penulis juga tergabung di Laboratorium Intelijensi Buatan (AI) ITERA, komunitas Data Science Indonesia dan IdBigData.



Hafiz Budi Firmansyah merupakan staf pengajar di program studi teknik informatika, Institut Teknologi Sumatera (ITERA), Lampung. Selain itu, penulis juga tergabung di Laboratorium Rekayasa Perangkat Lunak (RPL) ITERA dan komunitas Tech in Asia City Chapter Bandar Lampung