Jurnal Teknik Informatika (JUTIF)
Vol. 4, No. 4, August 2023, pp. 779-786

DOI: https://doi.org/10.52436/1.jutif.2023.4.4.704
p-ISSN: 2723-3863
e-ISSN: 2723-3871

IMPLEMENTATION OF CLUSTERING ON TWEET UPLOADING SIDE EFFECTS OF COVID-19 POST VACCINATION USING K-MEANS ALGORITHM

Santi[*1], Herny Februariyanti[2]

[1,2]Information Systems Study Program, Faculty of Information Technology and Industry,
Universitas Stikubank Semarang, Indonesia
Email: [1] santisanti@mhs.unisbank.ac.id , [2] hernyfeb@edu.unisbank.ac.id

(Article Received: November 25, 2022; Revision: Desember 8, 2022; Published: August 18, 2023)

Abstract

The Covid-19 Vaccination Program has become pros and cons among Indonesian people including Twitter social media users. When the program was running, Twitter users started uploading tweets regarding the side effects that occurred, ranging from mild to severe, both scientifically proven. or not. Of all uploaded tweets, by applying text mining, only tweets containing the queries "vaccine effect" and "post vaccine" in the period from January to June 2022 and Indonesian language tweets will be used and based on these parameters a total of 4800 tweets have been collected, of the total These tweets will be further processed using the clustering method, the k-means algorithm and the silhouette coefficient. The results of implementing the silhouette coefficient show that the best cluster is in cluster 2 with a score of 0.6228720387313319 and the results of the clustering algorithm k-means for 4800 tweets obtained 3917 members in cluster 0, and 883 members in cluster 1 placement. The feature is that cluster 0 contains tweets that state program effects explicitly or explicitly, while cluster 1 states program effects that arise implicitly.

Keywords: *Clustering, Covid-19 Vaccination Effects, K-Means, Text Mining, Twitter.*

IMPLEMENTASI CLUSTERING TERHADAP UNGGAHAN TWEET EFEK SAMPING PASCA VAKSINASI COVID-19 MENGGUNAKAN ALGORITMA K-MEANS

Abstrak

Program Vaksinasi *Covid-19* telah menjadi pro dan kontra di kalangan masyarakat Indonesia termasuk para pengguna Media sosial *Twitter*, pada saat program tersebut berjalan para pengguna *Twitter* mulai mengunggah *tweet* terkait efek samping yang timbul, mulai dari yang ringan sampai parah, baik yang terbukti secara ilmiah maupun tidak. Dari seluruh *tweet* yang diunggah, dengan menerapkan *text mining* hanya *tweet* yang memuat *query* "efek vaksin" dan "pasca vaksin" pada kurun waktu Januari sampai Juni 2022 dan *tweet* berbahasa Indonesia yang akan digunakan dan berdasarkan paramater tersebut berhasil dihimpun total 4800 *tweet*, dari total *tweet* tersebut akan diolah lebih lanjut menggunakan metode *clustering*, algoritma *k-means* serta *silhouette coefficient.* Hasil penerapan *silhouette coefficient* menunjukkan bahwasanya *cluster* terbaik pada *cluster* 2 dengan *score* 0.6228720387313319 dan hasil *clustering* algoritma *k-means* atas 4800 *tweet* diperoleh 3917 anggota pada *cluster* 0, dan 883 anggota menempati *cluster* 1. Berdasarkan penerapan metode dan data tweet yang telah dikelompokkan, diperoleh suatu ciri bahwasanya *cluster* 0 memuat *tweet* yang menyatakan efek vaksinasi secara jelas atau eksplisit, sedangkan *cluster* 1 menyatakan efek vaksinasi yang timbul secara implisit.

Kata kunci: *Clustering, Efek Vaksinasi Covid-19, K-Means, Text Mining, Twitter.*

1. PENDAHULUAN

Pada tanggal 3 Maret 2022 Indonesia mengkonfirmasi kasus positif pertama yang disebabkan oleh *Covid*-19, beberapa hari kemudian tepatnya pada tanggal 12 Maret 2020 *World Health Organization* (WHO) menetapkan *Covid*-19 sebagai pandemi. *Covid*-19 atau *Corona Virus Disease*-2019 merupakan jenis penyakit menular yang disebabkan *SARS-CoV*2, dan merupakan penyakit infeksi saluran pernapasan akut yang ditularkan melalui saluran pernapasan [1].

Seiring dengan meningkatnya kasus kematian serta kasus positif yang disebabkan oleh *Covid*-19, Pemerintah Indonesia melalui Badan Nasional Penanggulangan Bencana berupaya menekan laju penyebaran *Covid*-19 dengan melakukan Vaksinasi

Covid-19. Vaksinasi *Covid*-19 merupakan suatu program yang bertujuan untuk menciptakan *herd immunity* bagi penerima vaksin dan berguna melawan infeksi *Covid*-19 [2], program tersebut terdiri dari beberapa dosis yaitu dosis 1, dosis 2, serta dosis 3 (*booster*) yang telah diberikan kepada masyarakat umum.

Disebutkan dalam [3] bahwasanya program tersebut dapat membentuk sebuah persepsi dikarenakan kurangnya pemahaman masyarakat terkait vaksinasi *Covid-19.* Sehingga, setelah proses vaksinasi tersebut berjalan para pengguna media sosial, khususnya *Twitter* membuat unggahan *tweet* terkait efek samping yang dirasakan setelah mengikuti program tersebut, mulai dari rasa pegal pada bekas suntikan sampai efek vaksin yang belum tentu kebenarannya.

Guna melihat opini yang timbul dari pengguna *Twitter* terkait efek samping yang dirasakan pasca vaksinasi *Covid*-19 secara mendalam perlu dilakukan sebuah penelitian lebih lanjut sehingga nantinya akan diketahui tren pembahasan di kalangan masyarakat saat program vaksinasi tersebut berjalan.

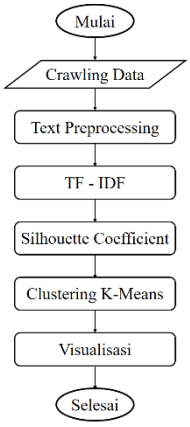
Terdapat penelitian terkait yang menggunakan metode *clustering* terhadap suatu data tertentu, dalam [4] penelitian tersebut mengakuisisi data tekstual dan dilakukan secara *real time* dari media sosial Twitter dan diperoleh hasil perhitungan menggunakan *silhouette coefficient* diketahui 21 *cluster* memiliki nilai *positif*, 3 *cluster* memiliki nilai 0, dan 4 *cluster* memiliki nilai *negative* dan pada penelitian berikut [5] didapatkan hasil penerapan algoritma *k-means* diperoleh 5 *cluster* yaitu pangan, produksi, lahan, ekspor dan teknologi. Terdapat pula penelitian [6] yang menyimpulkan bahwasanya dengan penerapan *k-means clustering* terbentuk suatu *cluster tweet* mengenai konten aktivitas perbelanjaan, penawaran *mall*, *event*, serta penawaran produk tertentu. Dipaparkan pula dalam [7] bahwasanya dengan dengan data yang diperoleh dari media sosial dapat dikelompokkan menjadi 3 *cluster* berdasarkan frekuensi kemunculan produk penjualan toko online yaitu yaitu sering, sedang dan jarang dengan tingkat akurasi 92,86%.

Berdasarkan penelitian [4]-[7] dapat disimpulkan bahwasanya penerapan *text mining* bermanfaat untuk memperoleh data maupun informasi guna menganalisis sebuah tren pembahasan yang tengah diperbincangkan dikalangan pengguna media social tertentu, dan fokus penelitian yang telah dipaparkan adalah melakukan analisis atas sentimen/ opini yang timbul terhadap topic tertentu dan berdasarkan label yang dimiliki data tersebut.

Berbeda dengan penelitian sebelumnya, penelitian ini bertujuan untuk menggelompokkan unggahan *tweet* pengguna *Twitter* berdasarkan efek samping yang timbul pasca vaksinasi *Covid*-19 dengan menggunakan metode *clustering* algoritma *k-means* dan *silhouette coefficient* dan penerapan metode tersebut akan mengelompokkan *tweet* kedalam masing-masing *cluster* menurut relevan atau jelas tidaknya *tweet* yang diunggah terkait efek samping yang dirasakan.

2. METODE PENELITIAN

Alur penelitian guna melakukan *clustering* atas unggahan *tweet* terkait efek samping pasca vaksinasi *Covid-19* menggunakan algoritma *k-means* pada penelitian ini ditampilkan pada Gambar 1.



Gambar 1. Alur Penelitian

Gambar 1 menunjukkan alur penelitian meliputi *Crawling* data, *text preprocessing*, TF-IDF, *Silhouette Coefficient*, *Clustering K-Means*, dan Visualisasi.

2.1 *Crawling* Data

Crawling Data bertujuan memperoleh data dari suatu database tertentu [8]. Pada tahap ini dihimpun data yang berasal dari media sosial *Twitter* yaitu unggahan tweet berdasarkan *query* "efek vaksin" dan "pasca vaksin" kurun waktu 1 Januari 2022 sampai 30 Juni 2022 dalam tweet bahasa Indonesia dan memiliki 3 atribut antara lain *datetime*, *username*, dan *content*. Hasil *Crawling* data tweet ditunjukkan pada Tabel 1.

Tabel 1. Hasil *Crawling* Data

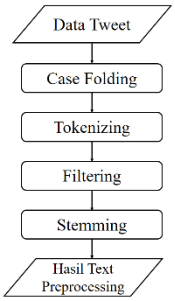
Datetime	***Username***	***Content***
2022-01-26 00:52:02+00:00	inewsdotid	Komnas KIPI: Efek Serius Vaksin Covid-19 pada Anak Lebih Rendah Dibandingkan Dewasa https://t.co/Gyk1889Sva #vaksin
2022-01-26 00:59:39+00:00	deenoo_o	@catboybie Nenek sama budhe ku juga gitu kak, abis vaksin booster sama sama demam. Emang efek sampingnya gitu kali yaa. Anw semoga cepet pulih ya kakk

2022-01-26 01:00:37+00:00	hantiww	ini efek semalem abis nangis atau emg pilek ya...udah mah gampang pilek abis vaksin malah nambah gampang pilek

Pada Tabel 1 ditampilkan hasil *crawling* data *tweet*, karena data tersebut tidak memiliki duplicate *tweet* dan atribut *content* merupakan data *unstructured* maka diperlukan tahap *Text Preprocessing.*

2.2 *Text Preprocessing*

Tahap *Text Preprocessing* bertujuan guna membersihkan data tweet dari data yang tidak diperlukan[9] hal ini dikarenakan *tweet* yang dihimpun dapat berupa teks singkat, gambar maupun video dan merupakan sebuah data *unstructured*. Selanjutnya pada Gambar 2 ditampilkan alur tahapan dalam *text preprocessing.*



Gambar 2. Tahapan *Text Preprocessing*

Dalam penelitian ini tahapan *text preprocessing* yang digunakan antara lain:

1. *Case Folding*, dilakukan guna menyeragamkan seluruh huruf dalam *tweet* menjadi huruf kecil (*lower case*) serta menghilangkan karakter kecuali a-z.
2. *Tokenizing* merupakan tahap memisahkan *tweet* berdasarkan kata yang membangunnya menjadi kata atau entitas lainnya yang disebut sebagai token.
3. *Filtering* merupakan tahap menghapus kata tidak penting yang termuat dalam *stopwords*, sehingga kata yang termuat dalam *tweet* hanya kata yang bermakna.
4. *Stemming* merupakan tahapan menelusuri akar (*root*) pada tiap kata dari setiap token dengan cara mengembalikan kata berimbuhan ke bentuk dasarnya (*stem*).

2.3 *Term Frequency – Index Document Frequency (TF-IDF)*

Kata atau *term* yang telah terbentuk akan diketahui pula bobot kemunculannya dengan menggunakan *Term Frequency – Inverse Document Frequency* dan nantinya akan diketahui bobot atau seberapa penting kata tersebut dalam tweet. *Term Frequency* akan memperlihatkan kata yang memiliki frekuensi kemunculan paling banyak dalam suatu tweet, berikut adalah persamaan guna menghitung *term frequency*[10]:

$$tf_{ij} = \frac{f_d(i)}{max_{j \in d}\, f_d(j)} \quad (1)$$

Keterangan :
tf = *term frequency* pada sebuah dokumen
i = term pada *document*
j = term pada database
fd = *frequency document*
d = *document*

Berbanding terbalik dengan *term frequency*, *index document frequency* (IDF), semakin tinggi frekuensi kemunculan term, semakin rendah bobot dari term tersebut [11], guna mencari nilai IDF digunakan persamaan [12] :

$$idf\ (t, D) = log\left(\frac{N}{df(t)+1}\right) \quad (2)$$

Keterangan :
t = term ke -n
D = dokumen ke -n
N = jumlah total dokumen
$df(t)$ = jumlah dokumen dalam total dokumen yang mengandung term t

2.4 *Silhouette Coefficient*

Dalam [13] *Silhouette Coefficient* merupakan metode yang bertujuan guna mencari kekuatan dari sebuah *cluster*, dan pada [14] bahwa metode *silhouette coefficient* merupakan gabungan dari metode *cohesion* dan *separation* yang berfungsi mengukur serta melihat kedekatan suatu *cluster* dengan *cluster* lainnya. Guna menghitung *score silhouette coefficient* perlu dilakukan sebuah perhitungan dengan menggunakan persamaan [14]:

$$silhouette\ score = \frac{p-q}{\max(p-q)} \quad (3)$$

Keterangan :
p = jarak rata-rata, ke *centroid* terdekat
q = jarak rata-rata *cluster* ke semua *centroid* pada *cluster*nya sendiri

2.5 *Clustering K-Means*

Clustering merupakan tahap pengelompokkan sekumpulan data ke dalam suatu *cluster* tertentu sehingga data dalam *cluster* memiliki kemiripan yang cukup signifikan, tetapi memiliki perbedaan dengan *cluster* lainnya [15] Serta guna mendukung *clustering* tersebut diperlukan sebuah algoritma, salah satunya adalah algoritma *k-means*, pada [16] algoritma *k-means* merupakan metode yang berguna memisahkan data kedalam suatu *cluster* yang

berbeda (*Partition-Based Clustering*) dan secara iterative mampu meminimalkan rata-rata jarak setiap data ke suatu *cluster.*

Dalam [17] dipaparkan bahwasanya kombinasi *clustering* dan algoritma *k-means* mampu mengolah data yang banyak dengan efektif dan efisien.

3. HASIL DAN PEMBAHASAN

Berdasarkan data tweet dan metode yang digunakan hasil pada masing-masing tahapan dipaparkan sebagai berikut :

3.1 *Crawling Data*

Total 4800 *tweet* yang berhasil dihimpun menggunakan *package Snscrape tweet* yang diperoleh pada Januari sampai Maret 2022 adalah *tweet* dengan *query* "efek vaksin covid" sedangkan "pasca vaksin" diterapkan pada *tweet* bulan April sampai Juni 2022. Perbedaan penggunaan *query* tersebut guna memperoleh *tweet* yang lebih beragam namun tetap relevan dengan penelitian.

3.2 *Text Preprocessing*

Setelah proses *crawling* data selesai dan dipastikan tidak ada duplikasi tweet dalam data tersebut, didalam tahapan ini terdapat pula tahap lainnya antara lain:

1. *Case Folding*

Proses *case folding* pada data *tweet* dilakukan sebanyak 1 kali pada 4800 *content* (*tweet)* dengan menggunakan *function* :

df['content']=df['content'].str.lower()

Gambar 3. Menunjukkan hasil dari *function* pada tahap *case folding.*

```
Hasil Case Folding :

0       efek serius vaksin covid-19 pada anak lebih re...
1       komnas kipi: efek serius vaksin covid-19 pada ...
2       @catboybie nenek sama budhe ku juga gitu kak, ...
3       ini efek semalem abis nangis atau emg pilek ya...
4       inilah salah satu efek buruk pemaksaan vaksin ...
                              ...
4795    tiba-tiba badanku gak enak, sesek nafas juga.....
4796    tadi pagi habis vaksin booster, trus dengan pd...
4797    ini efek vaksin booster baru berasa sekarang y...
4798    @paejjj @hannafarhana_ aku rasa dia immortal ð...
4799    ya bener, lemah letih lesu ku kira karena efek...
```

Gambar 3. Hasil *Case Folding*

2. *Tokenizing*

Tokenizing diterapkan pada data *tweet* guna membersihkan dari nomor, *whitespace*, tanda baca maupun karakter *special* lainnya, pada proses ini dilakukan sebanyak 6 kali dan dibantu dengan *package string*, *regex* serta *library* NLTK dan diterapkan *function* :

def remove_tweet_special(text):
def remove_number(text):
def remove_punctuation(text):
def remove_whitespace_LT(text):
def remove_whitespace_multiple(text):
def remove_singl_char(text):

Gambar 4 menunjukkan hasil *tokenizing* berdasarkan *function* diatas

```
0       {'efek': 1, 'serius': 1, 'vaksin': 1, 'covid':...
1       {'komnas': 1, 'kipi': 1, 'efek': 1, 'serius': ...
2       {'nenek': 1, 'sama': 3, 'budhe': 1, 'ku': 1, '...
3       {'ini': 1, 'efek': 1, 'semalem': 1, 'abis': 2,...
4       {'inilah': 1, 'salah': 1, 'satu': 1, 'efek': 1...
                              ...
4795    {'tibatiba': 1, 'badanku': 1, 'gak': 1, 'enak'...
4796    {'tadi': 1, 'pagi': 1, 'habis': 1, 'vaksin': 2...
4797    {'ini': 1, 'efek': 1, 'vaksin': 1, 'booster': ...
4798    {'aku': 1, 'rasa': 1, 'dia': 2, 'immortal': 1,...
4799    {'ya': 1, 'bener': 1, 'lemah': 1, 'letih': 1, ...
```

Gambar 4. Hasil *Tokenizing*

3. *Filtering*

Proses selanjutnya adalah menghapus kata yang tidak memiliki makna dan tidak relevan dengan topic penelitian ini dalam data tweet, dengan proses *filtering* ini diperlukan *library* NLTK dan tambahan package stopwords diterapkan *function* :

def stopwords_removal(words):
return [word for word in words if word not in list_stopwords]

Gambar 5 menampilkan hasil penerapan *function* tahap *filtering.*

```
0    [efek, serius, vaksin, covid, anak, rendah, di...
1    [komnas, kipi, efek, serius, vaksin, covid, an...
2    [nenek, budhe, ku, gitu, kak, abis, vaksin, bo...
3    [efek, semalem, abis, nangis, emg, pilek, yaud...
4              [salah, efek, buruk, pemaksaan, vaksin]
```

Gambar 5. Hasil *Filtering*

4. *Stemming*

Tahap terakhir dalam text preprocessing adalah tahap *stemming*, dalam term yang termuat pada data tweet akan diubah menjadi kata dasar pada masing-masing *term.* Guna mendukung proses *stemming* tersebut diperlukan suatu package tambahan yaitu Sastrawi dan *Swifter*, dan menjalakan *function* :

def stemmed_wrapper(term):
return stemmer.stem(term)

Pada penerapan function tersebut diperoleh 9840 term dalam 4800 data tweet yang telah diubah menjadi kata dasar, seperti "dibandingkan" menjadi "banding" dan Gambar 6 menampilkan hasil *stemming* pada masing-masing term.

```
0       [efek, serius, vaksin, covid, anak, rendah, ba...
1       [komnas, kipi, efek, serius, vaksin, covid, an...
2       [nenek, budhe, ku, gitu, kak, abis, vaksin, bo...
3       [efek, semalem, abis, nang, emg, pilek, yaudah...
4                  [salah, efek, buruk, paksa, vaksin]
                              ...
4795    [tibatiba, badan, enak, sek, nafas, efek, vaks...
4796    [pagi, habis, vaksin, booster, trus, pd, ngomo...
4797    [efek, vaksin, booster, asa, yak, lemes, bange...
4798    [immortal, pastu, manusia, luarbiasa, vaksin, ...
4799    [bener, lemah, letih, lesu, ku, efek, habis, v...
```

Gambar 6. Hasil *Stemming*

Guna melihat perbandingan sebelum dan sesudah melalui tahapan *text preprocessing* secara lebih sistematis ditampikan pada Tabel 2.

Tabel 2. Perbandingan Hasil Tahap *Text Preprocessing*

Tahap	Sebelum	Sesudah
Case Folding	Efek Serius Vaksin Covid-19 pada Anak Lebih Rendah Dibandingkan Dewasa #LengkapCepatBeritanya #BeritaTerkini #Berita #News #BeritaNasional . https://t.co/CzMrg3KMfS	efek serius vaksin covid-19 pada anak lebih rendah dibandingkan dewasa #lengkapcepatberitanya #beritaterkini #berita #news #beritanasional . https://t.co/czmrg3kmfs
Tokenizing	efek serius vaksin covid-19 pada anak lebih rendah dibandingkan dewasa #lengkapcepatberitanya #beritaterkini #berita #news #beritanasional . https://t.co/czmrg3kmfs	['efek', 'serius', 'vaksin', 'covid', 'pada', 'anak', 'lebih', 'rendah', 'dibandingkan', 'dewasa']
Filtering	['efek', 'serius', 'vaksin', 'covid', 'pada', 'anak', 'lebih', 'rendah', 'dibandingkan', 'dewasa']	['efek', 'serius', 'vaksin', 'covid', 'anak', 'rendah', 'dibandingkan', 'dewasa']
Stemming	[['efek', 'serius', 'vaksin', 'covid', 'anak', 'rendah', 'dibandingkan', 'dewasa']	[['efek', 'serius', 'vaksin', 'covid', 'anak', 'rendah', 'banding', 'dewasa']

3.3 *Term Frequency – Index Document Frequency (TF-IDF)*

Setelah data tweet melalui tahap *text preprocessing*, diperoleh hasil 9840 kata dari 4800 data tweet, dari 9840 kata tersebut akan dihitung bobot masing-masing kata dalam data tweet dengan menggunakan metode *term frequency – index document frequency.* Pada penerapannya digunakan *package ast* dan *function* :

```
def calc_DF(tfDict):
def calc_IDF(__n_document, __DF):
def calc_TF_IDF(TF):
ranking.sort_values('rank', ascending=False)
```

Gambar 7 menampilkan hasil ranking term menggunakan *function* tersebut dan diperoleh perhitungan bobot pada masing-masing *term* dalam *tweet.*

	term	rank
2	booster	185.109870
3	samping	162.894009
12	rasa	100.842885
4	sakit	97.916385
5	covid	89.445882
7	badan	86.382329

Gambar 7. *Ranking Term* Data *Tweet*

Gambar 7 menunjukkan bahwasanya *term* atau kata *"booster"* menempati posisi pertama dengan bobot 185.109870 diikuti kata "samping" dengan perolehan 162.894009.

Selanjutnya guna mempermudah proses penentuan *cluster* bobot pada masing-masing term akan dikonversi menjadi bentuk matriks, dengan menggunakan *function*:

```
tfidf_mat = normalized_counts.multiply(IDF_vector).toarray()
```

Gambar 8 menampilkan matriks hasil perhitungan TF-IDF menggunakan *function* diatas.

	aamiin	abai	abis	abis booster	abis makan	abis suntik	abis suntik vaksin	abis vaksin	...
0	0.0	0.0	0.00000	0.0	0.0	0.0	0.00000	0.00000	...
1	0.0	0.0	0.00000	0.0	0.0	0.0	0.00000	0.00000	...
2	0.0	0.0	0.24014	0.0	0.0	0.0	0.25281	0.27447	...
3	0.0	0.0	0.96057	0.0	0.0	0.0	0.50563	0.00000	...
4	0.0	0.0	0.00000	0.0	0.0	0.0	0.00000	0.00000	...
...	...	...	...	...	...	...	...	...	...
4798	0.0	0.0	0.00000	0.0	0.0	0.0	0.00000	0.00000	...
4799	0.0	0.0	0.00000	0.0	0.0	0.0	0.00000	0.00000	...

Gambar 8. Matriks Perhitungan TF-IDF

3.4 *Silhouette Cofficient*

Metode *silhouette coefficient* digunakan karena mampu menunjukkan ciri *cluster* terbaik berdasarkan hasil perhitungan pada masing-masing, pada penerapannya dibutuhkan *package* Pandas dan *library scikit-learn* serta *matplotlib* selanjutnya digunakan parameter *cluster* 2 sampai 10, dengan *function* sebagai berikut:

```
for k in range (2,10):
    labels =KMeans(n_clusters=k,init="random",random_state=42).fit(vt).labels_
    print("silhouette score for clusters= "+str(k)+" is "+str
(metrics.silhouette_score(vt,labels,metric="euclidean",sample_size=1000,random_state=42)))
```

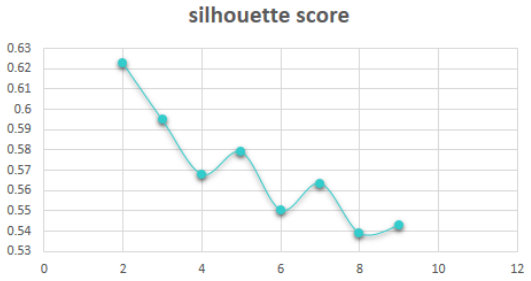
Gambar 9 menampilkan *silhouette score* masing-masing *cluster*..

```
silhouette score for clusters= 2 is 0.6228720387313319
silhouette score for clusters= 3 is 0.5947986690155999
silhouette score for clusters= 4 is 0.5680017440705517
silhouette score for clusters= 5 is 0.5789488603300184
silhouette score for clusters= 6 is 0.5503396932862007
silhouette score for clusters= 7 is 0.5635190653558664
silhouette score for clusters= 8 is 0.5387970263740304
silhouette score for clusters= 9 is 0.5429536727221593
```

Gambar 9. *Silhouette Score*

Diperoleh hasil *cluster* terbaik pada *cluster* 2 dengan perolehan *score* 0.6228720387313319, hal ini dikarenakan dari seluruh perhitungan, *score cluster* 2 yang paling mendekati nilai 1.

Gambar 10 menampilkan grafik perolehan *silhouette score* pada masing-masing *cluster.*



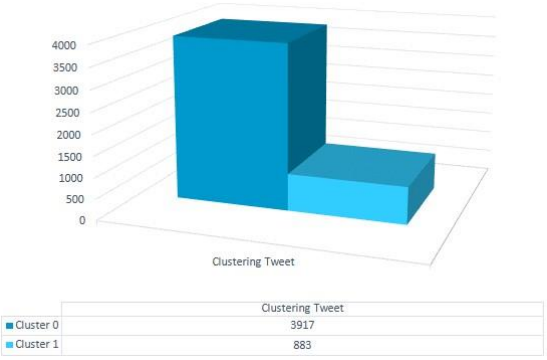
Gambar 10. Grafik *Silhouette Score*

3.5 *Clustering K-Means*

Tahapan utama dari penelitian ini adalah mengelompokkan (*clustering*) data *tweet* menggunakan algoritma *k-means*, dan pada penggunaan bahasa *Pyhton* pada *tools Jupyter Notebook* diperlukan bantuan dari *package scikit-learn* dengan menggunakan parameter hasil *silhouette coefficient* yaitu *cluster* (k) = 2, selanjutnya diterapkan *function* sebagai berikut :

```
kmeans_model = KMeans(n_clusters=k,init="random",n_init=10,max_iter=300,random_state=42)
kmeans=kmeans_model.fit(tfidf_mat)
labels=kmeans.labels_
```

Pada penerapan *function* tersebut diperoleh dua *cluster*, dengan *cluster* 0 berisi 3917 *tweet* dan *cluster* 1 berisi 883 *tweet*, yang selanjutnya ditampilkan pada Gambar 11 diagram hasil *clustering* masing-masing
cluster



Gambar 11. Diagram Hasil *Clustering*

Berdasarkan hasil cluster tersebut, ditampilkan Gambar 11 yang merupakan diagram batang berdasarkan jumlah anggota yang dimiliki pada masing-masing *cluster. Cluster* 0 diwakili warna biru tua memiliki jumlah anggota terbanyak yaitu 3917 atau 82% dan *Cluster* 1 dengan warna biru muda memiliki jumlah anggota sebanyak 883 atau 18% dari total keseluruhan data *tweet* yaitu 4800.

3.6 *Visualisasi*

Pada penelitian ini, visualisasi bertujuan melihat representasi dari data tweet yang telah dikelompokkan, dan diketahui pula kata yang paling sering muncul dalam masing-masing cluster. Oleh karena itu, guna memvisualisasikan hasil *clustering* berdasarkan frekuensi kemunculan kata pada suatu *cluster* dipilhlah visualisasi menggunakan *word cloud.*

Pada *Cluster* 0 ditampilkan Gambar 12 berupa hasil visualisasi menggunakan *word cloud.*

Gambar 12. *Word Cloud Cluster* 0

Visualisasi *word cloud cluster* 0 didominasi kata “efek”, hal tersebut menunjukkan bahwasanya kata tersebut memiliki frekuensi kemunculan terbanyak dalam *cluster 0* dan hampir muncul pada setiap tweet didalam *cluster* tersebut, yang selanjutnya diikuti kata “samping”,”vaksin”,”booster”,”sakit”, serta “demam”.

Gambar 13 menanmpilkan hasil visualisasi *word cloud cluster* 1.

Gambar 13. *Word Cloud Cluster* 1

Gambar 13 hasil visualisasi c*luster* 1 juga menunjukkan *word cloud* yang serupa, bahwasanya pada cluster tersebut frekuensi kemunculan kata terbanyak dimiliki oleh kata "efek" namun terdapat beberapa perbedaan urutan kata, yaitu pada *cluster* 1 diikuti kata "vaksin","booster","ngantuk","sakit", serta "demam".

4. DISKUSI

Penelitian ini bertujuan mengelompokkan data *tweet* efek samping pasca vaksinasi *covid-19* tanpa melakukan label sentiment terlebih dahulu, hal ini dikarenakan menghindari perbedaan persepsi atas unggahan *tweet* dan dengan menerapkan metode *clustering k-means* pada 4800 data *tweet* yang telah dihimpun mampu mengelompokkan menjadi 2 *cluster* berdasarkan relevan atau tidaknya tweet tersebut atas topic penelitian.

Hasil penelitian tersebut yang menjadikan pembeda dengan penelitian terdahulu yang berfokus mengelompokkan *tweet* berdasarkan label *sentiment positif*, netral dan *negative*.

5. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan dapat ditarik kesimpulan bahwasanya dari 4800 *tweet* setelah melalui tahapan *text preprocessing* diperoleh 9840 kata dan hasil pembobotan kata menunjukkan kata *"booster"* menempati peringkat pertama dengan bobot 185.109870 dan hasil penerapan metode *silhouette coefficient* atas matriks tf-idf *tweet* efek samping yang timbul pasca vaksinasi *Covid*-19 dapat dikelompokkan menjadi 2 *cluster*, serta *clustering k-means* menunjukkan hasil yang selaras yaitu *cluster* 0 yang ditempati 4917 *tweet* memaparkan efek samping vaksinasi secara tegas seperti pusing, demam, kantuk dan lain sebagainya, dan *cluster* 1 justru memiliki anggota *tweet* yang sebaliknya, hal ini dikarenakan penggunaan kata yang tidak sesuai makna seperti nanonano, dahsyat, dan hadeh.

DAFTAR PUSTAKA

[1] S. Baloch, M. A. Baloch, T. Zheng, and X. Pei, "The coronavirus disease 2019 (COVID-19) pandemic," *Tohoku J. Exp. Med.*, vol. 250, no. 4, pp. 271–278, 2020.

[2] F. F. Nur and V. N. Rahman, "Penyuluhan Program Vaksinasi Covid-19 Pada Mayarakat Desa Pakistaji," *Budimas J. Pengabdi. Masy.*, vol. 3, no. 2, 2021.

[3] N. P. Astuti, E. G. Z. Nugroho, J. C. Lattu, I. R. Potempu, and D. A. Swandana, "Persepsi masyarakat terhadap penerimaan vaksinasi COVID-19: Literature review," *J. Keperawatan*, vol. 13, no. 3, pp. 569–580, 2021.

[4] D. S. Indraloka and B. Santosa, "Penerapan Text Mining untuk Melakukan Clustering Data Tweet Shopee Indonesia," *J. Sains dan Seni ITS*, vol. 6, no. 2, pp. A51–A56, 2017.

[5] H. Irsyad and M. R. Pribadi, "IMPLEMENTASI TEXT MINING DALAM PENGELOMPOKAN DATA TWEET PERTANIAN INDONESIA DENGAN K-MEANS," *Kurawal-Jurnal Teknol. Inf. dan Ind.*, vol. 3, no. 2, pp. 164–172, 2020.

[6] S. M. Fani, R. Santoso, and S. Suparti, "Penerapan Text Mining untuk Melakukan Clustering Data Tweet Akun Blibli Pada Media Sosial Twitter Menggunakan K-Means Clustering," *J. Gaussian*, vol. 10, no. 4, pp. 583–593, 2021.

[7] A. Faesal, A. Muslim, A. H. Ruger, and K. Kusrini, "Sentimen analisis pada data tweet pengguna twitter terhadap produk penjualan toko online menggunakan metode k-means," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 19, no. 2, pp. 207–213, 2020.

[8] J. E. Sembodo, E. B. Setiawan, and Z. A. Baizal, "Data Crawling Otomatis pada Twitter," in *Indonesian Symposium on Computing (Indo-SC)*, 2016, pp. 11–16.

[9] M. Rivki and A. M. Bachtiar, "Implementasi algoritma K-Nearest Neighbor dalam pengklasifikasian follower twitter yang menggunakan Bahasa Indonesia," *J. Sist. Inf.*, vol. 13, no. 1, pp. 31–37, 2017.

[10] D. Prasetyantoro, "Penerapan Algoritma K-Means Clustering untuk Mengelompokkan Data Opini Pengguna Twitter mengenai Pandemi Covid-19," Stikubank (Unisbank) Semarang, 2021.

[11] F. Khoirunnisa and H. Februariyanti, "Analisis Sentimen Kualitas Layanan Google Meet Menggunakan Naïve Bayes Classifiers Dan Association," *semanTIK*, vol. 8, no. 1, p. 35, 2022, doi: 10.55679/semantik.v8i1.25120.

[12] S. Karyadi, H. Yasin, and M. A. Mukid, "Analisis Kecenderungan Informasi Dengan Menggunakan Metode Text Mining (Studi Kasus: Akun Twitter@ Detikcom)," *J. Gaussian*, vol. 5, no. 4, pp. 763–770, 2016.

[13] D. F. Pramesti, M. T. Furqon, and C. Dewi, "Implementasi Metode K-Medoids Clustering Untuk Pengelompokan Data Potensi Kebakaran Hutan/Lahan Berdasarkan Persebaran Titik Panas (Hotspot)," *J. Pengemb. Teknol. Inf. dan Ilmu Komput. e-ISSN*, vol. 2548, p. 964X, 2017.

[14] R. Hidayati, A. Zubair, A. H. Pratama, and L. Indana, "Analisis Silhouette Coefficient pada 6 Perhitungan Jarak K-Means Clustering," *Techno. Com*, vol. 20, no. 2, pp. 186–197, 2021.

[15] D. Suyanto, "Data Mining untuk klasifikasi dan klasterisasi data," *Bandung Inform. Bandung*, 2017.

[16] Y. W. Syaifudin and R. A. Irawan, "Implementasi Analisis Clustering Dan Sentimen Data Twitter Pada Opini Wisata Pantai Menggunakan Metode K-Means," *J. Inform. Polinema*, vol. 4, no. 3, p. 189, 2018.

[17] D. Adillah *et al.*, "Implementation of K-Means Clustering Analysis To Determine Barriers To Online Learning Case Study : Swasta Yapendak Implementasi K-Means Clustering Analysis Untuk Menentukan Hambatan Pembelajaran Daring Pada Siswa Studi Kasus : Smp," vol. 3, no. 3, pp. 1–7, 2022.