BAB II

TINJAUAN PUSTAKA

2.1 Covid-19

Covid-19 adalah penyakit akibat suatu virus baru yang sebelumnya tidak teridentifikasi pada manusia. Virus Corona adalah suatu kelompok virus yang ditemukan pada hewan dan manusia. Penyakit ini disebabkan oleh infeksi *Severe Acute Respiratory Syndrome Coronavirus 2* (SARS-CoV-2). Pada umumnya, Covid-19 menyebabkan gejala ringan seperti pilek, sakit tenggorokan, batuk, dan demam. Bagi beberapa orang, gejalanya bisa lebih parah, dan menimbulkan radang paru-paru atau sulit bernapas. Sejumlah kecil kasus penyakit ini menyebabkan kematian. Dan Saat ini sudah ada pengobatan atau vaksin untuk Covid-19. Namun, masih sementara dalam tahap pengujian kepada masyarakat.

Covid-19 pertama kali ditemukan di Wuhan pada bulan Desember 2019. Virus ini menyebar dengan cepat ke berbagai Negara hampir ke seluruh dunia sehingga WHO menyatakan bahwa Covid-19 sebagai pandemic global. Peningkatan status dari epidemi ke pandemi yang secara resmi diumumkan *World Health Organization* (WHO) pada tanggal 11 Maret 2020 (WHO, 2020b) tersebut menjadi salah satu kejadian luar biasa yang tidak pernah diperkirakan sebelumnya. Penetapan Pandemi sendiri mempertimbangkan suatu penyakit yang bersifat menular dan menyebar ke banyak Wilayah atau Negara. Pandemi global Covid-19 sendiri sampai dengan tanggal 16 Agustus 2020 telah menyebar ke 213 Negara/teritorial. Secara global,

sampai dengan pukul 01:00 PM, 16 Agustus 2020, ada 21.480.111 kasus Covid-19 yang dikonfirmasi, termasuk 771.518 kematian (3,59%), dilaporkan kepada WHO (WHO, 2020a).

2.2 Data Mining

Data mining merupakan suatu proses untuk menemukan informasi dari jumlah data yang besar (Zaki & Meira, 2014). *Data mining* mempunyai lima peran utama yaitu estimasi, prediksi, klasifikasi, klaster dan asosiasi (P.-N. Tan, Steinbach, & Kumar, 2006). Peran *data mining* yang sering digunakan adalah klasifikasi dan klaster karena dapat digunakan untuk atribut yang banyak (Fan, Wallace, Rich, & Zhang, 2006).

Pengelompokan *Data Mining* dibagi menjadi beberapa kelompok yaitu:

a. Estimasi

   Estimasi hampir sama dengan klasifikasi, kecuali apabila variable target estimasi lebih ke arah numerik daripada ke arah kategori. Model yang dibangun menggunakan record lengkap yang menyediakan nilai variable target sebagai nilai prediksi.

b. Prediksi

   Prediksi digunakan untuk memperkirakan nilai di masa mendatang. Untuk melakukan prediksi, biasanya diperlukan data yang telah terjadi sebelumnya. Kemudian dengan metode yang diinginkan, diterapkan pada data tersebut

untuk mengetahui kebiasaan yang terjadi sehingga dapat memperkirakan nilai yang terjadi di masa mendatang.

c. Klasifikasi

Dalam klasifikasi terdapat target variable kategori, misal penggolongan pendapatan dapat dipisahkan dalam tiga kategori, yaitu tinggi, sedang, dan rendah. Klasifikasi juga dapat digunakan untuk memprediksi suatu data tergolong ke dalam kategori tertentu.

d. Pengklasteran

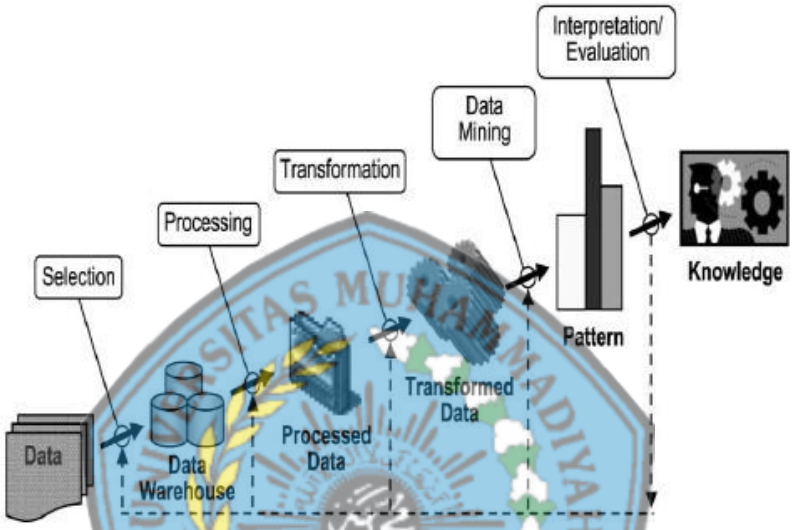
Pengklasteran merupakan pengelompokan *record*, pengamatan, atau memperhatikan dan membentuk kelas objek-objek yang memiliki kemiripan. Pada umumnya, pengklasteran melibatkan perhitungan jarak. Hal ini dilakukan untuk mengetahui tingkat kemiripan yang berada dalam data.

e. Asosiasi

Menemukan atribut yang muncul dalam satu waktu. Dalam dunia bisnis lebih sering disebut analsis keranjang belanja. Mendeteksi kumpulan atribut-atribut yang muncul bersamaan *(co-occur*) dalam frekuensi yang sering, dan membentuk sejumlah kaidah darikumpulan-kumpulan tersebut.

Data mining adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik-teknik, metode-metode, atau algoritma dalam *data mining* sangat bervariasi. Pemilihan metode atau algoritma yang tepat sangat bergantung pada tujuan dan proses *Knowledge Discovery in Database* (KDD) secara keseluruhan. *Knowledge*

Discovery in Database (KDD) mengacu pada keseluruhan proses menemukan pengetahuan yang bermanfaat dari data. Ini melibatkan evaluasi dan kemungkinan interpretasi pola untuk membuat keputusan tentang apa yang memenuhi syarat sebagai pengetahuan.



Gambar 2.1 Tahapan dalam KDD

Secara garis besar, *Knowledge Discovery dan Data Mining* (KDD) adalah proses yang dibantu oleh komputer untuk menggali dan menganalisis sejumlah besar himpunan data dan mengekstrak informasidan pengetahuan yang berguna. Istilah *Knowledge Discovery in Database* (KDD) dan data mining seringkali digunakan secara bergantian untuk menjelaskan proses penggalian informasi tersembunyi dalam suatu basis data yang besar. Sebenarnya kedua istilah tersebut memiliki konsep yang berbeda, tetapi berkaitan satu sama lain, dan salah satu tahapan dalam keseluruhan proses KDD adalah data mining.

2.2.1 Data Selection

Pemilihan (seleksi) data dari sekumpulan data operasional perlu dilakukan sebelum tahap penggalian informasi dalam *Knowledge Discovery in Database* (KDD) dimulai. Data hasil seleksi yang akan digunakan untuk proses *data mining*, disimpan dalam suatu berkas terpisah dari basis data operasional.

Data selection merupakan proses meminimalkan jumlah data yang digunakan untuk proses mining dengan tetap merepresentasikan data aslinya. Data selection dapat berupa sampling, denoising, dan feature extraction (Sulastri & Gufroni, 2017).

2.2.2 Pre-processing / Cleaning

Sebelum proses data mining dapat dilaksanakan, perlu dilakukan proses cleaning pada data yang menjadi fokus *Knowledge Discovery in Database (KDD)*. Proses cleaning mencakup antara lain membuang duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data, seperti kesalahan cetak. Juga dilakukan proses enrichment, yaitu proses "memperkaya" data yang sudah ada dengan data atau informasi lain yang relevan dan diperlukan untuk *Knowledge Discovery in Database* (KDD), seperti data atau informasi eksternal lainnya yang diperlukan.

Menurut Eska (2016), sebelum proses data mining dapat dilakukan, perlu adanya proses cleaning pada data yang menjadi fokus *Knowledge Discovery in Data* (KDD). Proses cleaning mencakup pembuangan duplikasi data, memeriksa data yang inkonsisten, dan memperbaiki kesalahan pada data. Selain itu, terdapat juga proses *enrichment*, yaitu proses "memperkaya" data yang sudah ada dengan data

atau informasi lain yang relevan dan diperlukan untuk KDD, seperti data atau informasi eksternal.

2.2.3 Transformation

Coding adalah proses transformasi pada data yang telah dipilih, sehingga data tersebut sesuai untuk proses data mining. Proses coding dalam *Knowledge Discovery in Database* (KDD) merupakan proses kreatif dan sangat tergantung pada jenis atau pola informasi yang akan dicari dalam basis data.

Transformasi data merupakan proses perubahan atau penggabungan data ke dalam format yang sesuai untuk diproses dalam data mining. Beberapa metode data mining membutuhkan format data yang khusus sebelum bisa diaplikasikan. Sebagai contoh beberapa metode standar seperti analisis asosiasi dan clustering hanya bisa menerima input data kategorikal. Karenanya data berupa angka numerik yang berlanjut perlu dibagi-bagi menjadi beberapa interval. (Eska, 2016).

2.2.4 Data Mining

Data mining adalah proses mencari pola atau informasi menarik dalam data terpilih dengan menggunakan teknik atau metode tertentu. Teknik-teknik, metode-metode, atau algoritma dalam data mining sangat bervariasi. Pemilihan metode atau algoritma yang tepat sangat bergantung pada tujuan dan proses *Knowledge Discovery in Database* (KDD) secara keseluruhan.

Data yang telah dikumpulkan harus dianalisis, diproses, dan diubah kedalam informasi yang dapat menginformasikan, menginstruksi, menjawab ataupun memberikan pemahaman dan pembuatan keputusan. Kemampuan untuk mengekstraksi seperti ini sangatlah berguna, biasanya kebutuhan pengetahuan

tersembunyi dari data menjadi meningkat seiring berkembangnya dunia kompetitif saat ini. Misalnya ketika data telah digunakan untuk memprediksi, maka dalam pengambilan keputusan akan memberikan hasil sesuai dengan yang diinginkan. (Stubbe dan Coleman, 2014)

2.2.5 Interpretation / Evaluation

Pola informasi yang dihasilkan dari proses data mining perlu ditampilkan dalam bentuk yang mudah dimengerti oleh pihak yang berkepentingan. Tahap ini merupakan bagian dari proses *Knowledge Diccovery in Database* (KDD) yang disebut interpretation. Tahap ini mencakup pemeriksaan apakah pola atau informasi yang ditemukan bertentangan dengan fakta atau hipotesis yang ada sebelumnya.

Evaluasi pola (*pattern evaluation*) digunakan untuk mengidentifikasi pola- pola menarik kedalam knowledge based yang ditemukan. Dalam tahap ini hasil dari teknik data mining berupa pola-pola yang khas maupun model prediksi dievaluasi untuk menilai apakah hipotesa yang ada memang tercapai (Eska, 2016).

2.3 Clustering

Clustering merupakan salah satu metode yang ada dalam data mining yang bertujuan untuk mengelompokkan data set ke dalam kelompok tertentu (*cluster*) dengan menggunakan parameter tertentu sehingga objek yang terdapat dalam sebuah cluster memiliki tingkat similaritas yang tinggi satu sama lain. Saat ini, para peneliti terus melakukan perbandingan model clustering guna mendapatkan cluster yang cocok digunakan pada sebuah penelitian (A. Ramadhan, Efendi, dan Mustakim, 2017).

Clustering merupakan suatu proses pengelompokkan record suatu , observasi, atau mengelompokkan kelas yang memiliki kesamaan objek. Perbedaan clustering dengan klasifikasi yaitu tidak adanya variabel target dalam melakukan suatu pengelompokan pada proses *clustering*. Clustering sering dilakukan sebagai untuk langkah awal dalam proses data mining saat melakukkan suatu metode analisis. Terdapat banyak algoritma Clustering yang telah digunakan oleh peneliti sebelumnya seperti K-Means, Improved K-Means, Fuzzy C-Means, DBSCAN, K-Medoids (PAM), CLARANS dan Fuzzy Substractive. Setiap algortima memiliki kelebihan dan kekurangan masing-masing, namun prinsip algoritma sama, yaitu mengelompokkan data sesuai dengan karakteristik dan mengukur jarak kemiripan antar data dalam satu kelompok. Kualitas cluster diukur menggunakan *Silhouette Coefficient*

2.4 *Density-Based Spatial Clustering of Application with Noise* (DBSCAN)

Density Based Clustering Density-Based Spatial Clustering of Application with Noise (DBSCAN) adalah metode pengelompokan berdasarkan tingkat kepadatan data (*density-based*). DBSCAN algoritma menumbuhkan area-area dengan kepadatan yang cukup tinggi ke dalam *cluster-cluster* dan menemukan *cluster-cluster* dalam bentuk yang sembarang dalam suatu *database spatial* yang memuat *noise* (Sander *et al.*, 1998). DBSCAN mendefinisikan *cluster* sebagai himpunan maksimum dari titik-titik kepadatan yang terkoneksi (*density-connected*). Semua objek yang tidak masuk ke dalam *cluster* manapun dianggap sebagai *noise*.

Tahapan DBSCAN yaitu menghitung jarak titik pusat ke titik yang lain menggunakan jarak Euclidean lalu dinyatakan seperti Persamaan:

$$\text{Jarak} = \sqrt{x - xp)^2 + (y - yp)^2} \quad (2.1)$$

Keterangan:

x: Koordinat sumbu x titik tujuan

y: Koordinat sumbu y titik tujuan

xp: Koordinat pusat sumbu x

yp: Koordinat pusat sumbu y

Setelah terbentuk kelompok dilanjutkan dengan menghitung silhouette yang hasilnya bervariasi antara -1 hingga 1. Pengertian nilai dalam silhouette jika 1 maka berada dalam kelompok yang tepat. Jika 0 maka berada diantara dua kelompok sehingga tidak jelas masuk kelompok A atau B. Jika -1 maka struktur kelompok overlapping dan lebih tepat dimasukkan kekelompok lain. Jika lebih besar dari 0 dan mendekati 1 maka kelompok yang dihasilkan sudah optimal.

2.5 Konsep Kepadatan (*Density Concept*)

DBSCAN menentukan sendiri jumlah *cluster* yang akan dihasilkan sehingga kita tidak perlu lagi untuk menentukan jumlah *cluster* yang diinginkan, tapi memerlukan 2 input lain, yaitu:

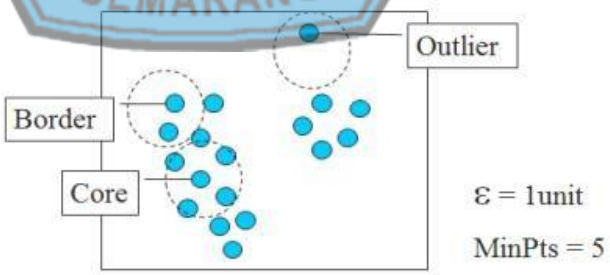
a. *MinPts*: minimum banyak items dalam cluster
b. *Eps:* nilai untuk jarak antar-items yang menjadi dasar pembentukkan *neighborhood* dari suatu titik item.

Neighborhood yang terletak di dalam radius (ε) disebut ε – neighborhood dari objek data.

Jika ε-*neighborhood* dari suatu objek berisi paling sedikit suatu angka yang minimum, *MinPts* dari suatu objek, objek tersebut disebut *core* object.

Kepadatan data merupakan jumlah data yang berada dalam radius *minpts* berupa jumlah data minimum dalam radius *epsilon* (ε). Konsep kepadatan memiliki tiga status (Prasetyo, 2012), yaitu:

1. Poin inti (*core*) : jumlah tetangga dan dirinya sendiri berada dalam radius *epsilon* (ε) ≥ *minpts*.
2. Poin batas (border) : jumlah tetangga dan dirinya sendiri dalam radius *epsilon* (ε) ≤ *minpts*, tetapi tetangga menjadi inti karena kehadirannya.
3. Poin pencilan (outlier) : jumlah tetangga dan dirinya sendiri dalam radius *epsilon* (ε) kurang dari *minpts* dan tidak ada tetangga yang menjadi inti karna kehadirannya.



Gambar 2.2. Gambar 1 *Core* dan *Border*

Menurut definisi, ada 2 jenis titik (*points*) dalam suatu *cluster*: di dalam *cluster* (*core points*) dan di tepian *cluster* (*border points*) di mana *neighborhood* dari border points berisi jauh lebih sedikit items daripada *neighborhood* dari *core*

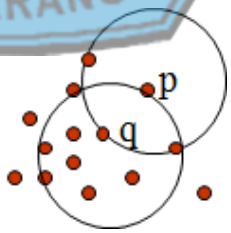
points (Ester *et al.*, 1996). Suatu *border point* bisa jadi termasuk ke dalam lebih dari 1 *cluster*.

DBSCAN menelusuri *cluster-cluster* dengan memeriksa *Epsilon* (ε) – *neighborhood (Eps-neighborhood)* dari tiap-tiap point dalam *database*. Jika *Epsilon* (ε) - *neighborhood* dari point p mengandung lebih dari *MinPts, cluster* baru dengan p sebagai *core object* diciptakan.

Kemudian DBSCAN secara iteratif mengumpulkan secara langsung objek-objek *density-reachable* dari *core object* tersebut, dimana mungkin melibatkan penggabungan dari beberapa cluster-cluster yang *density-reachable*.

2.5.1 Directly density-reachable

Sebuah titik item dikatakan directly density-reachable dari titik lainnya jika jarak di antara mereka tidak lebih dari nilai Eps. *Directly density-reachable* = titik q dikatakan *directly density-reachable* dari titik p jika titik q adalah Neps(p) dan p adalah core point.

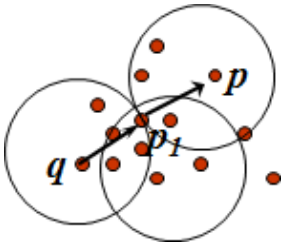


Gambar 2.3 *Directly Density-Reachable*

2.5.2 Density-reachable

Sebuah titik *item* dikatakan *density-reachable* dari titik *item* yang lain jika ada suatu rantai yang menghubungkan keduanya yang berisi hanya titik-titik yang *directly density-reachable* dari titik-titik sebelumnya.

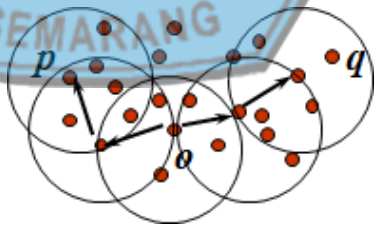
Suatu objek p adalah density reachable dari objek q dengan respek ke *Epsilon* (ε) dan MinPts dalam suatu set objek D jika terdapat suatu rantai objek $p_1,p_2,\ldots,p_n$, dimana p_1 = q dan p_n = p, di mana p_i +1 *density reachable* secara langsung dari p_i dengan respek ke *epsilon* (ε) dan *minpts*.



Gambar 2.4 *Density-Reachable*

2.5.3 Density-connected

Sebuah obyek *p* adalah *density-connected* terhadap obyek *q* dengan memperhatikan *epsilon* (ε) dan *minpts* dalam set obyek *D*, jika ada sebuah obyek *o* elemen *D* sehingga *p* dan *q* keduanya *density-reachable* dari *o* dengan memperhatikan *epsilon* (ε) dan *minpts*.



Gambar 2.5 Density-Connected

2.5 Algoritma *K-Nearest Neighbor* (KNN)

Algoritma k-nearest neighbor (Pencarian tetanga terdekat) merupakan teknik klasifikasi yang sangat popular yang diperkenalkan oleh Fix dan Hodges (1951), yang telah terbukti menjadi algoritma sederhana yang baik. KNN

merupakan salah satu metode yang digunakan dalam pengklasifikasian dengan menggunakan algoritma supervised (Chan et al. 2010).

Pada fase pembelajaran, algoritma ini hanya melakukan penyimpanan vektor-vektor fitur dan klasifikasi dari data pembelajaran. Pada fase klasifikasi, fitur-fitur yang sama dihitung untuk data test (yang klasifikasinya tidak diketahui). Jarak dari vektor yang baru ini terhadap seluruh vektor data pembelajaran dihitung, dan sejumlah k buah yang paling dekat diambil. Titik yang baru klasifikasinya diprediksikan termasuk pada klasifikasi terbanyak dari titik-titik tersebut.

Nilai k yang terbaik untuk algoritma ini tergantung pada data. Umumnya, nilai k yang tinggi akan mengurangi efek noise pada klasifikasi, tetapi membuat batasan antara setiap klasifikasi menjadi lebih kabur. Nilai k yang bagus dapat dipilih dengan optimasi parameter, misalnya dengan menggunakan cross-validation. Kasus khusus di mana klasifikasi diprediksikan berdasarkan data pembelajaran yang paling dekat (dengan kata lain, k = 1) disebut algoritma nearest neighbor.

2.6 Silhouette Coefficient

Silhouette coefficient merupakan metode evaluasi untuk menguji optimal atau ketepatan sebuah cluster yang telah terbentuk dari proses *clustering* (Tanzil Furqon and Muflikhah 2016). *Silhouette coefficient* memberikan hasil kualitas visual objek dalam tiap *cluster*, memberikan informasi sesuai dengan jumlah cluster pada data set. Untuk setiap objek dinotasikan oleh cluster dimana dia

berasal (Swindiarto 2018). Metode ini merupakan gabungan dari metode separation dan cohesion. Tahapan perhitungan *Silhouette coefficient* adalah sebagai berikut (Handoyo, Rumani, and Nasution, 2014):

1. Hitung rata-rata jarak dari suatu data, menggunakan Persamaan 2.2 maka didapatkan rata-rata dengan cara memisalkan *i* terhadap semua data lain yang berada dalam satu *cluster* sebagai berikut.

$$\alpha(i) = \frac{1}{|A|-1} \sum j \epsilon A, j \neq i \; d(i,j) \qquad (2.2)$$

Dimana :

α(i) = Perbedaan rata-rata objek (i) ke semua objek lain pada A

d(i,j) = jarak antara data i dengan j

A = Cluster

2. Hitung rata-rata jarak data *i* tersebut dengan semua data di *cluster* lain, dan diambil nilai terkecilnya menggunakan Persamaan 2.3.

$$d(i,C) \frac{1}{|C|} \sum j \epsilon c \; d(i,j) \qquad (2.3)$$

Dimana :

d(i,C) = Perbedaan rata-rata objek (i) ke semua objek lain pada C

C = cluster lain selain cluster A atau cluster C tidak sama dengan cluster A.

3. Setelah menghitung *d(i,C)* untuk semua C, maka diambil nilai terkecil dengan menggunakan Persamaan 2.4

$$b(i) = min \; c \neq A \; d(i,C) \qquad (2.4)$$

Cluster B yang mencapai minimum (yaitu, d(i,B) = b(i)) disebut tetangga dari objek (i). Ini adalah cluster terbaik kedua untuk objek (i)

4. Nilai *Silhouette Coefficient* didefinisikan seperti pada Persamaan 2.5.

$$S(i) = \frac{b(i) - \alpha(i)}{max\,\alpha(i), b(i)} \quad (2.5)$$

Dimana :

S(i) = Nilai Silhouette Coofficient

b(i) = Nilai minimum objek i dengan objek pada cluster lain C

b(i) = Rata-rata jarak objek ke i dengan semua objek yang berada di dalam suatu cluster

Tabel 2.1 Tabel Nilai *Silhouette Kaufman* dan *Rousseeuw*

Nilai Silhouette Coofficient	Struktur
0.7 < SC <= 1	Struktur Kuat
0.5 < SC <= 0.7	Struktur Sedang
0.25 < SC <= 0.5	Struktur Lemah
SC <= 0.25	Tidak terstruktur

Setelah terbentuk kelompok dilanjutkan dengan menghitung silhouette yang hasilnya bervariasi antara -1 hingga 1. Pengertian nilai dalam silhouette jika 1 maka berada dalam kelompok yang tepat. Jika 0 maka berada diantara dua kelompok sehingga tidak jelas masuk kelompok A atau B. Jika -1 maka struktur kelompok overlapping dan lebih tepat dimasukkan kekelompok lain. Jika lebih besar dari 0 dan mendekati 1 maka kelompok yang dihasilkan sudah optimal.