TUGAS AKHIR – KS 184822

TEXT CLUSTERING PADA AKUN *TWITTER* LAYANAN EKSPEDISI JNE, J&T, DAN POS INDONESIA MENGGUNAKAN METODE *DENSITY-BASED SPATIAL CLUSTERING OF APPLICATIONS WITH NOISE* (DBSCAN) DAN *K-MEANS*

DEVI PUTRI ISNARWATY
NRP 062117 4500 0032

Dosen Pembimbing
Irhamah, S.Si, M.Si., Ph.D

PROGRAM STUDI SARJANA
DEPARTEMEN STATISTIKA
FAKULTAS MATEMATIKA, KOMPUTASI, DAN SAINS DATA
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2019

TUGAS AKHIR – KS 184822

TEXT CLUSTERING PADA AKUN *TWITTER* LAYANAN EKSPEDISI JNE, J&T, DAN POS INDONESIA MENGGUNAKAN METODE *DENSITY-BASED SPATIAL CLUSTERING OF APPLICATIONS WITH NOISE* (DBSCAN) DAN *K-MEANS*

DEVI PUTRI ISNARWATY
NRP 062117 4500 0034

Dosen Pembimbing
Irhamah, S.Si, M.Si., Ph.D

PROGRAM STUDI SARJANA
DEPARTEMEN STATISTIKA
FAKULTAS MATEMATIKA, KOMPUTASI, DAN SAINS DATA
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2019

FINAL PROJECT – **KS 184822**

TEXT CLUSTERING ON TWITTER OF JNE, J&T, AND POS INDONESIA EXPEDITION SERVICES USING DENSITY-BASED SPATIAL CLUSTERING OF APPLICATIONS WITH NOISE (DBSCAN) AND K-MEANS METHOD

DEVI PUTRI ISNARWATY
SN 062117 4500 0034

Supervisor
Irhamah, S.Si, M.Si., Ph.D

UNDERGRADUATE PROGRAMME
DEPARTMENT OF STATISTICS
FACULTY OF MATHEMATICS, COMPUTING, AND DATA SCIENCE
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2019

LEMBAR PENGESAHAN

TEKS CLUSTERING PADA AKUN _TWITTER_ LAYANAN EKSPEDISI JNE, J&T, DAN POS INDONESIA MENGGUNAKAN METODE _DENSITY-BASED SPATIAL CLUSTERING_ (DBSCAN) DAN _K-MEANS_

TUGAS AKHIR

Diajukan untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Statistika
pada
Program Studi Sarjana Departemen Statistika
Fakultas Matematika, Komputasi, dan Sains Data
Institut Teknologi Sepuluh Nopember

Oleh :
DEVI PUTRI ISNARWATY
NRP. 062117 4500 0032

Disetujui oleh Pembimbing:
Irhamah, S.Si, M.Si, Ph.D
NIP. 19780406 200112 2 002

( )

Mengetahui,
Kepala Departemen Statistika

Dr. Suhartono
NIP. 19710929 199512 1 001

SURABAYA, JULI 2019

Halaman ini sengaja dikosongkan

TEXT CLUSTERING PADA AKUN *TWITTER* LAYANAN EKSPEDISI JNE, J&T, DAN POS INDONESIA MENGGUNAKAN METODE *DENSITY-BASED SPATIAL CLUSTERING OF APPLICATIONS WITH NOISE* (DBSCAN) DAN *K-MEANS*

Nama	**: Devi Putri Isnarwaty**
NRP	**: 062117 4500 0032**
Departemen	**: Statistika-FMKSD-ITS**
Dosen Pembimbing	**: Irhamah, S.Si, M.Si, Ph.D**

Abstrak

Tingginya minat masyarakat untuk berbelanja online membuat meningkatnya layanan ekspedisi yang digunakan untuk mengirimkan produk dari transaksi secara online maupun offline. Ada banyak perusahaan ekspedisi yang populer di Indonesia misalnya JNE, J&T, dan Pos Indonesia. Perusahaan ekspedisi gencar melakukan promosi lewat media sosial, misalnya saja Twitter. Akun Twitter ini dapat digunakan sebagai media bagi pelanggan untuk memberikan pendapat, kritik, maupun saran, dan bagi pihak perusahaan untuk memberikan tanggapan maupun informasi. Analisis terhadap twitter yang dikirim, berguna bagi perusahaan untuk meningkatkan performa layanan. Dokumen twitter berupa teks sehingga diperlukan text mining untuk menganalisisnya. Dalam penelitian ini, text clustering digunakan untuk mengelompokkan pendapat menjadi beberapa kategori. Metode yang digunakan pada text clustering adalah metode K-Means dan Density-Based Spatial Clustering of Applications with Noise (DBSCAN). DBSCAN adalah sebuah metode yang membentuk cluster dari data-data yang saling berdekatan, sedangkan data yang saling berjauhan tidak akan menjadi anggota cluster. K-Means adalah teknik clustering yang sederhana dan cepat dalam proses clustering obyek serta mampu mengelompokkan data dalam jumlah cukup besar. Berdasarkan nilai silhouette coefficient, metode DBSCAN merupakan metode terbaik untuk mengelompokkan tweet yang ditujukan kepada layanan ekspedisi JNE, J&T, dan Pos Indonesia.

Kata kunci : *Clustering, Ekspedisi, DBSCAN, K-Means, Text Mining.*

Halaman ini sengaja dikosongkan

TEXT CLUSTERING ON TWITTER OF JNE, J&T, AND POS INDONESIA EXPEDITION SERVICE USING DENSITY-BASED SPATIAL CLUSTERING OF APPLICATIONS WITH NOISE (DBSCAN) AND K-MEANS METHODS

Nama	**: Devi Putri Isnarwaty**
Student Number	**: 06211745000032**
Departement	**: Statistics**
Supervisor	**: Irhamah, S.Si, M.Si, Ph.D**

Abstract

The high of public interest for online shop made the expedition services grow up to service product delivery whether online or offline service. There are many popular expedition companies in Indonesia, such as JNE, J&T, and Pos Indonesia. Expedition companies are often promote their services by social medias, one of them is Twitter. This Twitter is used as medias for customers to give opinions, criticisms, or suggestions, and on the expedition company to provide feedback and information. Analysis of the twitter sent, useful for companies to improve service performance. Twitter documents require text form, hence the text mining analysis is needed. In this study, text clustering is needed for clustering customer's opinion. The text clustering's method which could be used are K-Means and Density-Based Spatial Clustering of Applications with Noise (DBSCAN). DBSCAN is a method which is clustering high density data, while low density data is neglected. K-Means is a simple and fast clustering technique in object clustering and could solve a big data. Based on the silhouette coefficient value, DBSCAN method is the best method to clustering JNE, J&T, and Pos Indonesia customer's opinion.

Keywords : *Clustering, Expedition,, DBSCAN, K-Means, Text Mining.*

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji Syukur kehadirat Allah SWT yang telah memberikan rahmat, taufik dan hidayah-Nya. Sholawat dan salam semoga senantiasa tercurahkan kepada Nabi Muhammad SAW atas suri tauladan dalam kehidupan ini sehingga penulis dapat menyelesaikan tugas akhir yang berjudul **"*TEXT CLUSTERING* PADA AKUN *TWITTER* LAYANAN EKSPEDISI JNE, J&T, DAN POS INDONESIA MENGGUNAKAN METODE *DENSITY-BASED SPATIAL CLUSTERING OF APPLICATIONS WITH NOISE* (DBSCAN) DAN *K-MEANS*".**

Terselesaikannya Tugas Akhir ini tak lepas dari peran serta berbagai pihak. Oleh karena itu, penulis ingin mengucapkan banya terima kasih kepada :

1. Ibu Irhamah, S.Si, M.Si, Ph.D selaku dosen pembimbing Tugas Akhir yang telah dengan sabar memberikan bimbingan, masukan dan dukungan bagi penulis untuk dapat menyelesai-kan Tugas Akhir ini.
2. Bapak Dr. Suhartono selaku Kepala Departemen Statistika FMKSD ITS yang telah memberikan nasehat, motivasi, serta bimbingan kepada penulis untuk dapat menyelesaikan Tugas Akhir ini.
3. Ibu Santi Wulan Purnami, S.Si, M.Si, selaku ketua Program Studi Sarjana yang telah membimbing dan memberikan motivasi penulis selama menjadi mahasiswa.
4. Bapak Prof. Drs. Nur Iriawan, MIKom, Ph,D dan Ibu Pratnya Paramitha Oktaviana, S.Si, M.Si selaku dosen penguji yang telah banyak memberikan saran dan masukan dalam penyelesaian Tugas Akhir ini.
5. Ibu Dra. Wiwiek Setya Winahju, M.S selaku Dosen Wali yang telah membimbing dan memberikan motivasi kepada penulis sejak awal perkuliahan.
6. Seluruh dosen dan staf karyawan Departemen Statistika ITS yang telah memberikan ilmu – ilmu dan pengalaman yang bermanfaat bagi penulis.

7. Mama, Papa, Mbak Rina, Mas Riyan, Rafif dan semua keluarga yang telah memberikan do'a, motivasi, dukungan, nasehat, kasih sayang, perhatian dan kesabaran yang tidak akan pernah bisa digantikan dengan apapun.
8. Dinda, Riris, Irfan, Bibeh, dan teman-teman lainnya yang memberikan semangat, motivasi dan do'a dalam penyelesaian Tugas Akhir ini.
9. Erna, Vriesia, Vida, Razty, Lely, Kunthi, Tila, yang selalu memberikan semangat, saran, dan motivasi dalam penyelesaian Tugas Akhir ini.
10. Teman-teman S1 LJ Jurusan Statistika ITS 2017 dan *Pioneer* yang telah mendukung dan memotivasi penulis dalam penyelesaian Tugas Akhir ini.
11. Semua pihak yang sudah banyak membantu penulis dalam proses pengerjaan Tugas Akhir ini yang tidak dapat penulis sebutkan satu per satu.

Penulis menyadari bahwa laporan ini masih jauh dari kesempurnaan, untuk itu penulis menerima saran dan kritik yang diberikan untuk penyempurnaan laporan Tugas Akhir ini. Penulis berharap semoga laporan ini dapat memberikan banyak manfaat untuk pembaca.

Surabaya, Juli 2019

Penulis

DAFTAR ISI

Halaman

Halaman

DAFTAR TABEL

Halaman

Halaman

DAFTAR GAMBAR

Halaman

Halaman

DAFTAR LAMPIRAN

Halaman

Halaman ini sengaja dikosongkan

BAB I
PENDAHULUAN

1.1. Latar Belakang

Perkembangan belanja *online* di masyarakat Indonesia berkembang sangat pesat dalam waktu yang relatif singkat. Belanja *online* sangat membantu dan mempermudah bagi masyarakat yang ingin berbelanja tanpa harus pergi ke toko yang menjual barang yang dibutuhkan. Kegiatan berbelanja secara *online* didorong oleh pertumbuhan industri *e-commerce* di Indonesia yang cukup tinggi, misalnya saja *e-commerce* yang menjadi langganan bagi masyarakat Indonesia adalah Shopee, Tokopedia, Bukalapak, dan lain-lain. Tingginya minat masyarakat untuk berbelanja *online* dan pengiriman barang, tentunya tidak lepas dari dibutuhkannya terhadap layanan pengiriman atau ekspedisi yang digunakan untuk mengirimkan barang/produk dari transaksi jual/beli secara *online* hingga berada di tangan *customer*. Hal ini menjadikan peluang bisnis bagi perusahaan ekspedisi belakangan ini.

Semakin banyak pelaku bisnis di bidang ekspedisi atau pengiriman barang yang bisa diakses dengan mudah oleh masyarakat luas. Tingkat pelayanan yang berbeda-beda dan juga tarif tentu akan menjadi pertimbangan khusus bagi para pengguna layanan pengiriman. Ada banyak perusahaan pengiriman barang yang populer di Indonesia misalnya saja JNE, J&T, dan Pos Indonesia. Alasan Pos Indonesia banyak diminati karena memiliki tarif yang lebih terjangkau dan jangkauan pengiriman Pos Indonesia sangat luas bahkan bisa sampai pelosok atau pedalaman yang jarak diakses oleh perusahaan pengiriman barang lainnya. Lain halnya dengan ekspedisi JNE banyak diminati dikarenakan memiliki banyak layanan yang dapat dipilih oleh konsumen berdasarkan lama pengiriman yang diinginkan serta memiliki sistem *tracking* yang baik, sehingga dengan mudah dilacak keberadaan dan proses pengiriman barang serta memiliki layanan *customer service* yang cepat dan responsif. Sedangkan untuk ekspedisi J&T meskipun baru beroperasi pada September 2015, namun memiliki sistem pe-

lacakan tepat waktu dan tetep beroperasi pada hari libur (Hadijah, 2016).

Bukan hanya *online shop*, perusahaan ekspedisi juga gencar melakukan promosi lewat media sosial, misalnya saja Instagram, Facebook dan *Twitter*. Media sosial ini bukan hanya digunakan sebagai media promosi tetapi juga sebagai interaksi antara perusahaan ekspedisi dengan pelanggannya. Perusahaan ekspedisi juga dapat menerima *feedback* dan pendapat secara terbuka yang diberikan oleh pelanggan melalui media sosial tersebut. Layanan media sosial, sebagian besar didasari oleh *web based* yang menyediakan fasilitas untuk berinteraksi antar pengguna. Media sosial Facebook dan Instagram tidak bisa mengakses percakapan antar pengguna, sedangkan media sosial *Twitter* menyediakan fasilitas "*search*" yang dapat melihat percakapan antara pengguna satu dengan pengguna lainnya.

Pada setiap perusahaan, memiliki *customer service* yang bertugas untuk memberikan pelayanan kepada pelanggan termasuk menerima keluhan/masalah yang sedang dihadapi oleh pelanggan tersebut. *Customer service* bukan hanya dapat diakses melalui telepon atau email tetapi juga dapat diakses melalui media sosial *Twitter*. Perusahaan ekspedisi JNE, J&T, dan Pos Indonesia memiliki akun *Twitter* customer care yaitu @JNECare untuk JNE, @jntexpressid untuk J&T dan @PosIndonesia untuk Pos Indonesia. Akun tersebut digunakan sebagai layanan pelanggan secara *online* yang disediakan untuk memberikan tanggapan, pendapat, kritik, saran ataupun keluhan dari pelanggan. Komentar dalam *Twitter* berbentuk teks, sehingga perlu dilakukan analisis *text mining*. *Text mining* dapat memberikan solusi dari permasalahan seperti pemrosesan, pengelompokkan, dan menganalisa *unstructured text* dalam jumlah besar. Salah satu teknik analisis dalam *text mining* adalah *text clustering*. Menurut Siregar & Puspabhuana (2002), *text clustering* merupakan pengelompokkan data tanpa berdasarkan kelas data tertentu yang biasanya dipakai untuk memberikan label pada kelas data yang belum diketahui. *Clustering* sering digolongkan sebagai metode *unsupervised*

learning karena prinsip dari *clustering* adalah memaksimalkan kesamaan antar anggota satu kelas dan meminimumkan kesamaan antar *cluster*.

Pendapat masyarakat yang ditujukan pada akun *customer care* layanan ekspedisi di media sosial dapat dikelompokkan menjadi beberapa kategori/*cluster*. Untuk mempermudah pihak layanan ekspedisi dalam menanggapi pendapat terkait keluhan, memberikan jawaban atas pertanyaan dan memberikan informasi kepada masyarakat terkait promosi atau info penting di media sosial, sehingga perlu dilakukan *text clustering*. *Text clustering* dapat mengelompokkan kata pada sebuah pendapat yang memiliki kemiripan untuk menjadi beberapa kategori atau *cluster*. Penentuan kategori atau kelompok dari *tweet* yang ditujukan terhadap layanan ekspedisi JNE, J&T, dan Pos Indonesia dalam mengantisipasi banyaknya *tweet*, seperti membuat *template* tanggapan untuk setiap kategori, dan mengelompokkan *tweet* yang masuk berdasarkan kategori dari hasil peng*cluster*an. Pada penelitian *text clustering* ini dilakukan dengan menggunakan metode *K-Means* dan *Density-Based Spatial Clustering of Applications with Noise* (DBSCAN). DBSCAN adalah algoritma pengelompokkan yang didasarkan pada kepadatan (*density*) data (Tan, 2007). Metode ini membentuk *cluster* dari data-data yang saling berdekatan/rapat, sedangkan data yang saling berjauhan tidak akan menjadi anggota cluster (Adinugroho & Sari, 2018). Sedangkan *K-Means* adalah metode *clustering* yang memiliki kemampuan untuk mengelompokkan data dalam jumlah yang cukup besar dengan waktu komputasi. *K-Means* adalah teknik yang cukup sederhana dan cepat dalam proses *clustering* obyek (*clustering*) (Irwansyah & Faisal, 2015).

Penelitian yang pernah dilakukan mengenai *text clustering* oleh Devi (2015) mengenai Implementasi Metode *Clustering* DBSCAN pada Proses Pengambilan Keputusan menghasilkan kesimpulan bahwa Proses *clustering* menggunakan Algoritma DBSCAN didapatkan sejumlah *cluster* dengan rata-rata nilai peindeks validitas menggunakan Algoritma *Silhouette* lebih besar

dari 0 dan mendekati 1, yang berarti bahwa proses *clustering* menggunakan algoritma DBSCAN telah dapat dikategorikan baik. Penelitian lain dilakukan oleh Budiman (2016) mengenai perbandingan metode *K-Means* dan DBSCAN pada Pengelompokkan Rumah Kost Mahasiswa di Kelurahan Tembalang Semarang yang menghasilkan kesimpulan bahwa metode *K-Means* bekerja lebih baik daripada DBSCAN untuk mengelompokkan asrama mahasiswa dibuktikan dari nilai koefisien siluet metode *K-Means* lebih tinggi dibandingkan DBSCAN. Penelitian lainnya dilakukan oleh Arsih (2016) mengenai Metode Peng*cluster*an Berbasis Densitas Menggunakan Algoritma DBSCAN. Hasil dari penelitian adalah Metode DBSCAN lebih baik dari *K-Means* berdasarkan lima kriteria pembanding yaitu kompleksitas, Bentuk *cluster*, parameter *input*, kemampuan menghadapi *noise*, dan *run times*. Penelitian lain yaitu dilakukan oleh Tamaela (2017) mengenai *Cluster Analysis* Menggunakan Algoritma *Fuzzy C-Means* dan *K-Means* untuk Klasterisasi dan Pemetaan Lahan Pertanian di Minahasa Tenggara menghasilkan kesimpulan bahwa algoritma *K-Means* dapat bekerja lebih baik dibandingkan dengan *Fuzzy C-Means* dengan menghasilkan jumlah kluster yang lebih banyak atau bervariasi. Penelitian lain dilakukan oleh Thomas dan Harode (2015) yan membahas tentang *A Comparative Study on K-Means and Hierarchical Clustering*. Hasil yang diperoleh adalah kinerja algoritma *K-Means* lebih baik dari algoritma *hierarchical clustering* serta Kinerja algoritma *K-Means* meningkat seiring menurunnya RMSE dan RMSE berkurang karena jumlah *cluster* meningkat.

1.2. Rumusan Masalah

Pendapat masyarakat yang berkaitan tentang layanan ekspedisi merupakan hal yang penting dan diperlukan bagi pihak layanan ekspedisi. Pendapat perlu diidentifikasi dan dikelompokkan menjadi beberapa kategori. Kategori pendapat dapat berupa keluhan, saran, opini, pemberian informasi, dan lain-lain. Pengidentifikasian informasi dari sebuah *text* diperlukan analisis menggunakan *text*

mining. Penentuan kategori pengelompokkan pada *text mining* dapat dilakukan dengan menggunakan metode *text clustering*. Metode *text clustering* yang digunakan adalah metode *Density-Based Spatial Clustering of Applications with Noise* (DBSCAN) dan *K-Means*. Dari kedua metode tersebut, akan dipilih metode terbaik yang akan digunakan dalam membantu pihak layanan ekspedisi untuk mengetahui kategori pendapat yang paling sering diberikan oleh masyarakat.

1.3. Tujuan Penelitian

Berdasarkan rumusan masalah pada penelitian ini, tujuan yang akan dicapai dalam penelitian ini adalah sebagai berikut.

1. Mengetahui karakteristik data berdasarkan *tweet* tentang layanan ekspedisi JNE, J&T dan Pos Indonesia.
2. Membandingkan hasil *clustering* pendapat terhadap layanan ekspedisi JNE, J&T dan Pos Indonesia menggunakan metode DBSCAN dan *K-Means.*

1.3. Manfaat Penelitian

Hasil penelitian ini diharapkan dapat bermanfaat membantu pihak layanan ekspedisi JNE, J&T dan Pos Indonesia dalam memahami pendapat masyarakat mengenai pelayanan yang diberikan serta mendapatkan informasi pengelompokkan kategori pendapat, misalnya pendapat mengenai keluhan, saran, opini, informasi atau lain-lain yang sering muncul pada *tweet* masyarakat dengan menggunakan *clustering*. Serta mempermudah pihak layanan ekspedisi dalam menanggapi pendapat dari masyarakat.

1.4. Batasan Masalah

Batasan masalah yang digunakan dalam penelitian ini adalah sebagai berikut.

1. Penelitian hanya menggunakan akun *Twitter customer care* dari layanan ekspedisi JNE (@JNECare), *customer care* dari layanan ekspedisi J&T (@jntexpressid), dan *customer care* dari layanan ekspedisi Pos Indonesia (@PosIndonesia).

2. Penelitian hanya melakukan analisis terhadap *tweet* berbahasa Indonesia.
3. Penelitian ini tidak mengatasi atau kalimat yang cara penulisannya disingkat.

BAB II
TINJAUAN PUSTAKA

2.1. *Text Mining*

Text mining dapat didefinisikan sebagai proses menggali informasi dimana pengguna bisa berinteraksi dengan beberapa sumber dari waktu ke waktu menggunakan *tools analysis*. *Text mining* bertujuan mengekstraksi informasi yang berguna dari beberapa sumber data melalui identifikasi dan eksplorasi pola yang menarik. Sumber data yang digunakan adalah kumpulan dokumen dan pola yang ditemukan berada dalam data tekstual yang tidak terstruktur (Feldman & Sanger, 2007). *Text mining* dapat digunakan untuk proses penemuan *rule* baru dengan algoritma pengelompokan, asosiasi, dan *ranking*. Pengelompokan adalah fungsi yang banyak dilakukan yaitu dengan metode *text clustering* dan *text classification*. *Text clustering* berhubungan dengan proses menemukan sebuah struktur kelompok yang belum terlihat (*unsupervised*) dari sekumpulan dokumen. Sedangkan *text classification* dapat dianggap proses untuk membentuk golongan dari dokumen berdasarkan pada kelas kelompok yang sudah diketahui sebelumnya (*supervised*) (Durajati & Gumelar, 2012). Proses *clustering* (*unsupervised*) adalah proses yang untuk mengkategorikan pendapat berdasarkan beberapa kategori, misalnya keluhan, saran, opini atau yang lainnya. Metode yang termasuk dalam *clustering* diantaranya adalah *DBSCAN* dan *K-Means*.

2.2. *Text Preprocessing*

Text Preprocessing merupakan sebuah langkah penting dalam Data Mining untuk membuat data mentah menjadi data yang berkualitas. Data perlu dilakukan *preprocessing* karena dalam data mentah terdapat data yang tidak lengkap, *noise*, dan tidak konsisten (Alfarisi, 2017). *Text preprocessing* terdiri dari pemisahan kata dengan tepat, normalisasi kata, pemfilteran kata terhadap istilah dan tanda baca tertentu. Salah satu tantangan dari *text mining* adalah mengubah teks terstruktur dan semi terstruktur menjadi model ruang vektor terstruktur. Hal ini yang harus dilakukan

sebelum melakukan *text mining* atau analisis lanjutan (Miner, dkk., 2012). Langkah – langkah dalam praproses teks adalah sebagai berikut.

1. Data *Cleaning*, tahap *cleaning* data merupakan tahapan untuk menghilangkan kata yang tidak diperlukan misalnya karakter HTML, link URL, *username* (@username), emoticons, dan hashtag (#). Tahap cleansing ini diperlukan karena kata-kata tersebut dianggap noise yang tidak diperlukan pada proses data (Buntoro, Adji, & Purnamasari, 2014).
2. *Case Folding*, tahap *case folding* merupakan tahapan untuk menghilangkan angka dan tanda baca, serta mengubah karakter teks menjadi huruf kecil semua. Sistem kerja *case folding* yaitu memproses huruf alphabet "a" sampai dengan "z", sehingga karakter diluar alphabet akan dihilangkan seperti halnya tanda baca dan angka (Weiss, Indurkhya, Zhang, & Damerau, 2005).
3. *Stemming*, tahap *stemming* ini merupakan tahap mendapatkan kata dasar. Sistem kerja tahap *stemming* ini adalah menghilangkan awalan, akhiran, sisipan, dan *confixes* (kombinasi dari awalan dan akhiran) (Ariadi & Fithriasari, 2015).
4. *Stopwords*, tahap *stopwords* merupakan tahap penghilangan kosakata yang bukan termasuk kata unik atau tidak menyampaikan pesan apapun secara signifikan pada teks. Kosakata yang dimaksud seperti kata penghubung dan kata keterangan misalnya "oleh", "di", "yang", "jadi", dan sebagainya (Dragut, Fang, Sistla, & Yu, 2009).
5. Tokenizing, tahap *tokenizing* merupakan tahapan memutuskan kata per kata pada kalimat. Tahapan ini bertujuan untuk memecah yang semula berupa kalimat menjadi potongan-potongan kata, sehingga urutan *string* akan terputus menjadi potongan-potongan kata penyusunnya (Bing, 2010).

2.3. TF-IDF

Term Frequency-Inverse Document Frequency (TF-IDF) adalah skema populer yang digunakan sebagai perhitungan dan pembobotan pada setiap kata. TF-IDF digunakan untuk mengukur seberapa penting suatu kata dalam dokumen. Frekuensi kemuncul-

an setiap kata dalam dokumen diberikan sebagai skema pembobotan kata yang diasumsikan bahwa *stopwords* yang paling sering muncul dalam teks sepenuhnya dihapus terlebih dahulu. Kata-kata yang diberikan bobot (*weight*) dihitung frekuensi kemunculannya dalam sebuah dokumen (Jo, 2018). *Term Frequency* (*TF*) meringkas seberapa sering suatu kata tertentu muncul dalam dokumen, sedangkan *Inverse Document Frequency* (*IDF*) menurunkan ukuran kata-kata yang sering muncul dalam dokumen (Manning, dkk., 2008). Berikut adalah persamaan yang membentuk TF-IDF yang dapat dilihat pada persamaan (2.1) dan (2.2).

$$W_{i,j} = TF_{i,j} \times IDF_j \,, \tag{2.1}$$

$$IDF_j = \log\left(\frac{N}{DF_j}\right), \tag{2.2}$$

keterangan :

$W_{i,j}$ = bobot dari kata ke *j* pada *tweet* ke *i*
DF_j = banyaknya *tweet* yang mengandung kata *j*
$TF_{i,j}$ = jumlah kemunculan kata ke *j* pada *tweet* ke *i*
IDF_j = *inverse document frequency* pada kata ke *j*
N = jumlah keseluruhan *tweet*

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

Density Based Spatial Clustering Algorithm with Noise (DBSCAN) adalah algoritma pengelompokkan yang didasarkan pada kepadatan (*density*) data. Konsep kepadatan yang dimaksud dalam DBSCAN adalah jumlah daya yang berada dalam radius *MinPts* (jumlah minimal data dalam radius *Eps*), data tersebut masuk dalam kategori kepadatan yang diinginkan, jumlah data dalam radius tersebut termasuk data inti itu sendiri. Konsep kepadatan seperti ini melahirkan tiga macam status dari setiap data, yaitu inti (*core*), batas (*border*), dan *noise*. Sebuah data akan dimasukkan sebagai inti jika jumlah data tetangga dan dirinya sendiri pada radius $Eps \geq MinPts$. Nilai radius *Eps* dan *MinPts* ini ditetapkan secara mandiri. Data yang jumlah tetangga dan dirinya

sendiri dalam radius *Eps* < *MinPts* tetapi tetangganya menjadi inti karena kehadirannya, data tersebut dikategorikan sebagai batas. Jika jumlah tetangga dan dirinya sendiri dalam radius *Eps* < *MinPts* dan tidak ada tetangga yang menjadi inti kaena kehadirannya, data tersebut dikategorikan sebagai *noise* (Tan, 2007).

Algoritma DBSCAN membutuhkan dua parameter penting, yaitu parameter radius (*Eps*) dan jumlah minimum poin untuk membentuk kelompok (*MinPts*). Algoritma dari DBSCAN adalah sebagai berikut (Han, dkk., 2012).

1. Menentukan parameter *MinPts* dan *Eps*. Proses penentuan parameter *MinPts* dan *Eps* berdasarkan hasil kombinasi *Eps* dengan *MinPts* yang memiliki nilai *silhouette coefficient* tertinggi.
2. Pilih *tweet p* secara acak
3. Menghitung jumlah *tweet* yang ditentukan oleh parameter radius (*Eps*). Jika jumlahnya mencukupi (lebih dari atau sama dengan *Eps*), data akan ditandai sebagai inti (*core point*).
4. Menghitung jarak *tweet* yang ditandai sebagai *core point* dengan *point* yang lain menggunakan jarak *Euclidean*. Berikut adalah rumus jarak *Euclidean* yang ditujukan pada persamaan (2.3).

$$d_{ip} = \sqrt{\sum_{j=1}^{m} (x_{ji} - y_{jp})^2} \ , \qquad (2.3)$$

keterangan :

d_{ip} = jarak *Euclidean* dari *tweet* ke-*i* ke pusat *cluster* ke-*k*

x_{ji} = frekuensi kemunculan kata ke-*j* pada *tweet* ke-*i*

y_{jp} = frekuensi kemunculan kata ke-*j* pada titik pusat ke-*p*

m = banyak kata

5. Buat *cluster* baru dengan menambahkan *tweet p* ke dalam *cluster*.
6. Melakukan identifikasi pada data yang ditandai sebagai *core point*
7. Lanjutkan proses sampai semua point telah diproses

8. Jika ada *tweet* yang tidak masuk ke dalam *cluster* manapun akan ditandai sebagai *noise*.

DBSCAN mencari *cluster* dengan memeriksa parameter radius (*Eps*) dari setiap titik dalam dataset. Jika *Eps* pada *tweet p* berisi lebih dari MinPts, sebuah cluster baru dengan *p* sebagai *core point* terbentuk. DBSCAN kemudian secara iteratif mengumpulkan langsung objek yang dapat dijangkau kerapatan dari *core point* ini, yang mungkin melibatkan penggabungan beberapa *cluster* yang dapat dijangkau kerapatan. Proses berakhir ketika tidak ada titik baru yang dapat ditambahkan ke *cluster* apa pun (Ye, dkk., 2003).

2.5. *K-Means*

K-Means adalah metode *clustering* yang memiliki kemampuan untuk mengelompokkan data dalam jumlah yang cukup besar dengan waktu komputasi. *K-Means* adalah teknik yang cukup sederhana dan cepat dalam proses *clustering* obyek (*clustering*). Algoritma *K-Means* mendefinisikan *centroid* atau pusat *cluster* dari *cluster* menjadi rata-rata poin dari *cluster* tersebut. Dalam penerapan algoritma *K-Means*, jika diberikan sekumpulan data $X = [x_1, x_2, \ldots, x_m]$ dimana $\boldsymbol{x}_j = [x_{j1}, x_{j2}, \ldots, x_{jm}]$ adalah sistem dalam ruang real *Rm*, maka algoritma *K-Means* akan menyusun partisi *X* dalam sejumlah *k cluster*. Setiap *cluster* memiliki titik tengah (*centroid*) yang merupakan nilai rata-rata dari data-data dalam *cluster* tersebut (Irwansyah & Faisal , 2015).

Algoritma *K-Means* memiliki kelebihan berupa algoritma yang ringkas dan efisien. Algoritma ini sangat bergantung pada titik awaal sampel (Li & Wu, 2012). Berikut adalah algoritma dari metode K-Means (Thomas & Harode, 2015).

1. Memilih secara acak *k centroid* awal dalam data
2. Menentukan jarak setiap kata terhadap pusat *cluster* (*centroid*)
3. Mengelompokkan setiap kata berdasarkan kedekatannya dengan *centroid* (jarak terkecil). Perhitungan *centroid* terdekat dengan menggunakan *Euclidean distance*. Berikut adalah rumus jarak *Euclidean* yang ditujukan pada persamaan (2.4).

$$d_{ik} = \sqrt{\sum_{j=1}^{m} (x_{ji} - y_{jk})^2} \ , \qquad (2.4)$$

keterangan :

d_{ik} = jarak *Euclidean* dari *tweet* ke-*i* ke pusat *cluster* ke-*k*

x_{ji} = frekuensi kemunculan kata ke-*j* pada *tweet* ke-*i*

y_{jk} = frekuensi kemunculan kata ke-*j* pada pusat *cluster* ke-*k*

m = banyak kata

4. Hitung ulang pusat *cluster* (*centroid*) baru menggunakan

$$v_{ik} = {}^{1}\!/_{n_k} \sum_{j=1}^{m} x_{ji} \ , \qquad (2.5)$$

keterangan :

v_{ik} = *centroid* (rata-rata *cluster* ke-*k* untuk *tweet* ke-*i*)

n_k = banyaknya *tweet* yang menjadi anggota *cluster* ke-*k*

x_{ji} = frekuensi kemunculan kata ke-*j* yang berada dalam *cluster* tersebut untuk *tweet* ke-*i*

5. Ulangi langkah 2 hingga 4, sampai anggota yang ada pada tiap *cluster* tidak berubah.

Jika jumlah *cluster k* belum diketahui, dapat menggunakan *Variance Ratio Criterion* (*VRC*). *VRC* dapat digunakan untuk menentukan jumlah *cluster k* yang optimum. *VRC* adalah rasio antara jumlah kuadrat dari jarak antara *tweet-tweet* dari *cluster* satu ke *cluster* lain atau *Between Group Sum of Square* (*BGSS*) dengan jumlah kuadrat jarak antara *tweet-tweet* yang termasuk dalam *cluster* yang sama atau *Within Group Sum of Square* (*WGSS*) (Calinski & Harabasz, 1974). Berikut adalah rumus dari *Variance Ratio Criterion* yang dapat dilihat pada persamaan (2.6), (2.7), dan (2.8).

$$VRC = \frac{BGSS / (K-1)}{WGSS / (N-K)} \ , \qquad (2.6)$$

$$BGSS = \sum_{k=1}^{K} \sum_{l=1, l \neq k}^{K} \sum_{j=1}^{m} (y_{jk} - y_{jl})^2 , \qquad (2.7)$$

$$WGSS = \sum_{k=1}^{K} \sum_{i=1}^{I} \sum_{j=1}^{m} (x_{ji} - y_{jk})^2 , \qquad (2.8)$$

keterangan :

K = banyak *cluster*

N = banyak *tweet*

m = banyak kata

I = banyak *tweet* yang menjadi anggota *cluster* ke-*k*

x_{ji} = frekuensi kemunculan kata ke-*j* yang berada dalam *cluster* tersebut untuk *tweet* ke-*i*

y_{jk} = frekuensi kemunculan kata ke-*j* pada pusat *cluster* ke-*k*

y_{jl} = frekuensi kemunculan kata ke-*j* pada pusat *cluster* ke-*l*

2.6. *Silhouette Coefficient*

Setelah terbentuk hasil *cluster*, perlu dihitung *silhouette coefficient*. *Silhouette coefficient* merupakan metode yang digunakan untuk mengevaluasi hasil *clustering* dengan memeriksa seberapa baik kelompok-kelompok (*cluster*) yang dihasilkan. Untuk kumpulan data *D*, dari *n* objek, misalkan *D* dipartisi menjadi *k* *cluster*, C_1, ..., C_k. Untuk setiap objek $i \in D$, kami menghitung $a(i)$ sebagai jarak rata-rata antara *i* dan semua objek lain di *cluster* yang dimiliki *i*. Demikian pula, $b(i)$ adalah jarak rata-rata minimum dari *i* ke semua *cluster* yang bukan milik *i*. Secara formal, anggaplah $i \in C_i$ $(1 \leq i \leq k)$ (Han, dkk., 2012). Berikut adalah langkah-langkah dalam perhitungan *silhouette coefficient*.

1. Menghitung jarak rata-rata dari tweet *i* dengan semua *tweet* yang ada di dalam *cluster* yang sama

$$a(i) = \frac{\sum_{j \in C_i, i \neq j} dist(i, j)}{|C_i| - 1} , \qquad (2.9)$$

keterangan :

C_i = banyak *tweet* dalam *cluster* C_i

j = *tweet* lain dalam *cluster* C_i
dist(i, j) = jarak *Euclidean* antara *tweet i* dengan *tweet j*

2. Menghitung jarak rata-rata dari *tweet i* dengan semua *tweet* yang berada di *cluster* berbeda dan didapatkan nilai terkecil.

$$\mathrm{b}(i) = \min_{C_l : 1 \le l \le k, l \ne 1} \left\{ \frac{1}{|C_l|} \sum_{j \in C_l} dist(i,l) \right\}, \qquad (2.10)$$

keterangan :
C_l = banyak *tweet* dalam *cluster* C_l
l = *tweet* lain dalam *cluster* yang berbeda
dist(i, l) = jarak *Euclidean* antara *tweet i* dengan *tweet l*

3. Hitung *silhouette coefficient*

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}, \qquad (2.11)$$

Nilai *silhouette coefficient* $s(i)$ adalah antara -1 dan 1. Nilai negatif $s(i)$ menunjukkan bahwa $b(i) < a(i)$, dan mungkin pengelompokan yang lebih mendekati pada *cluster* lainnya dibandingkan dalam *cluster* yang sama. Ketika nilai $s(i)$ ada di sekitar 1 yang berarti memiliki pengelompokkan jauh dari *cluster* lain (Kogan, 2007).

2.7. *Word Cloud*

Word cloud adalah visualisasi umum lainnya berdasarkan frekuensi. Di *word cloud*, kata-kata diwakili dengan ukuran *font* yang bervariasi. Dalam *word cloud* sederhana, hanya satu dimensi informasi yang ditampilkan. Khususnya, ukuran *font* yang sesuai dengan frekuensi *n* gram yang berarti bahwa semakin besar kata dalam *world cloud* dapat diubah digunakan sebagai informasi baru, misalnya dalam warna dan pengelompokkan (Kwartler, 2017). *Word cloud* menawarkan cara cepat dan mudah untuk memvisualisasikan frekuensi kata dan terutama untuk menemukan konsep yang berpotensi penting dalam satu set teks. *Word cloud* memiliki daftar berhenti untuk menghapus kata-kata umum dari analisis dan memilih berapa banyak kata yang akan divisualisasi-

kan. Misalnya, *word cloud* berisi kata-kata yang memiliki frekuensi kemunculan yang besar yaitu 50, 500, atau 5.000 dari sebuah teks. Walaupun *word cloud* merupakan alat yang sederhana, tetapi memiliki keuntungan untuk memvisualisasikan dengan lebih banyak kata dalam ruang kecil dibandingkan dengan daftar jumlah kata pada diagram batang (Bernard, dkk., 2017). Berikut adalah contoh visualisasi *word cloud* yang ditampilkan pada Gambar 2.1.

Gambar 2.1 Visualisasi *Word Cloud* (Sumber : Silge & Robinson, 2017).

2.8. Layanan Ekspedisi

Perusahaan Jasa Pengiriman atau ekspedisi merupakan sebuah perusahaan yang bergerak pada bidang layanan pengiriman, yang dalam hal ini adalah pengiriman barang. Pengiriman barang adalah proses memindahkan barang dari satu tempat ke tempat yang lainnya. Pengiriman barang dilakukan karena beberapa hal yaitu terdapat transaksi jual beli barang, adanya kebutuhan barang di suatu tempat dan untuk mengisi kebutuhan stok barang di lokasi yang lainnya (Rapi, 2017).

2.8.1 JNE

PT. Tiki Jalur Nugraha Ekakurir atau JNE adalah perusahaan layanan jasa yang berdiri pada tanggal 26 November 1990 yang memulai kegiatan usahanya terpusat pada penanganan kegiatan kepabeanan atau impor kiriman barang/dokumen serta pengantar-

annya dari luar negeri ke Indonesia. Divisi Ekspress JNE telah melayani kirman paket dan dokumen dalam negeri melalui lebih dari 1.500 titik layanan eksklusif dari penjemputan hingga pengantaran yang tersebar di seluruh Indonesia. JNE menyediakan layanan yang dapat dipilih dan disesuaikan oleh kebutuhan pelanggan (JNE, 2015).

2.8.2 J&T *Express*

J&T *Express* adalah perusahaan pengiriman *express* yang menerapkan pengembangan teknologi sebagai sistem dasar. Jaringan luas yang dimiliki oleh J&T *Express* memfasilitasi layanan *express* untuk pelanggan di seluruh Indonesia. Pelayanan pengiriman dilakukan bukan hanya dalam Kota, tetapi juga antar Kota, antar Provinsi, dan juga pelang-gan *e-commerce*. Meskipun J&T *Express* masih tergolong perusahaan yang baru, namun telah menyediakan layanan yang dapat mendukung pertumbuhan bisnis *e-commerce* salah satunya adalah layanan penjemputan (J&T, 2018).

2.8.3 Pos Indonesia

Pos Indonesia adalah perusahaan jasa yang berdiri pada tahun 1746 di Jakarta dengan tujuan awal untuk lebih menjamin keamanan surat-surat. Sejak itulah pelayanan pos lahir untuk mengemban peran dan fungsi pelayanan kepada publik. Selama 17 tahun Pos Indonesia berstatus perusahaan umum, pada Juni 1995 berubah menjadi Perseroan Terbatas dengan nama PT Pos Indonesia (Persero). Kini, Pos Indonesia telah mampu menunjukkan kreatifitasnya dalam pengembangn bidang perposan Indonesia dengan memanfaatkan infrastruktur jejaring yang mencapai sekitar 24 ribu titik layanan yang menjangkau 100% Kota/Kabupaten, hampir 100% Kecamatan dan 42% Kelurahan/Desa, dan 940 lokasi transmigrasi terpencil, serta Pos Indonesia memiliki lebih dari 3.800 Kantor pos *online*. Pos Indonesia juga telah memiliki berbagai macam layanan, salah satunya adalah Pos Express yang merupakan layanan premium milik Pos Indonesia untuk pengiriman dokumen, surat, paket serta barang dagangan *online* dengan

cepat dan aman hingga jangkauan luas ke seluruh wilayah Indonesia (Pos-Indonesia, 2017).

Halaman ini sengaja dikosongkan

BAB III
METODOLOGI PENELITIAN

3.1. Sumber Data dan Variabel Penelitian

Data yang digunakan dalam penelitian ini adalah data yang diambil dari kumpulan *tweet* dari pengguna *Twitter* di Indonesia. Akun *Twitter* yang digunakan dalam analisis adalah *customer care* dari Layanan Ekspedisi JNE (@JNECare), *customer care* dari Layanan Ekspedisi J&T (@jntexpressid), dan *customer care* dari Layanan Ekspedisi Pos Indonesia (@PosIndonesia). Data *tweet* diambil dari tanggal 4 Februari 2019 hingga 7 Mei 2019 dengan menggunakan Twitter API. Variabel penelitian yang digunakan adalah frekuensi kemunculan kata dasar dari setiap *tweet* yang ditujukan kepada @JNECare, @jntexpressid, dan @PosIndonesia yang telah dilakukan *preprocessing*, dinotasikan sebagai variabel *A* dengan skala rasio.

3.2. Struktur Data

Struktur data yang digunakan ditampilkan pada Tabel 3.1.

Tabel 3.1 Struktur Data

Layanan Ekspedisi	***Tweet* ke-**	A_1	A_2	...	A_m
JNE (1)	1	A_{111}	A_{112}	...	A_{11m}
	2	A_{121}	A_{122}	...	A_{12m}
	:				
	n	A_{1n1}	A_{1n2}	...	A_{1nm}
J&T (2)	1	A_{211}	A_{212}	...	A_{21m}
	2	A_{221}	A_{222}	...	A_{22m}
	:				
	n	A_{2n1}	A_{2n2}	...	A_{2nm}
Pos Indonesia (3)	1	A_{311}	A_{312}	...	A_{31m}
	2	A_{321}	A_{322}	...	A_{32m}
	:				
	n	A_{3n1}	A_{3n2}	...	A_{3nm}

3.3 Langkah Analisis

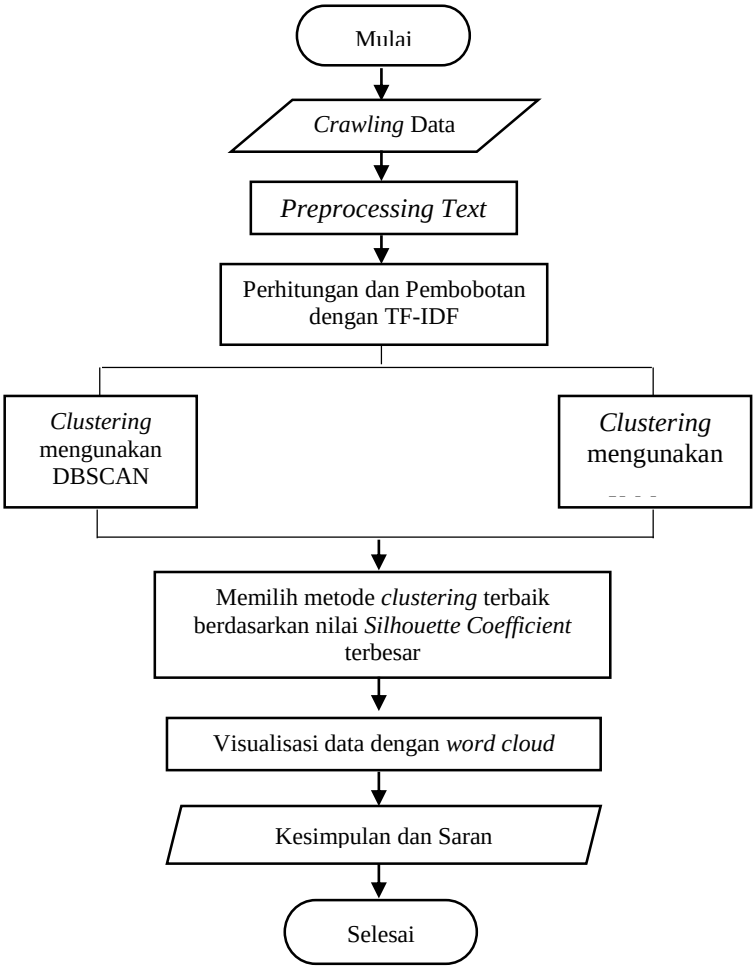
Langkah analisis yang digunakan pada penelitian ini adalah sebagai berikut.

1. Mengambil data *tweet* dengan menggunakan *Twitter* API
   a. Memasukkan *keyword* yang berhubungan dengan akun *Twitter* @JNECare, @jntexpressid dan @PosIndonesia
   b. Menyimpan hasil *crawling* ke database
2. Melakukan *text preprocessing* pada data *tweet* layanan ekspedisi JNE, J&T, dan Pos Indonesia
   a. Melakukan *cleaning* data, yaitu menghapus karakter HTML, *link* URL, *retweet* (RT), *username* (*@username*), baris baru, angka, *hashtag* (#), *emoticons*, tanda baca, dan *double space*.
   b. Melakukan *case folding*, yaitu mengubah seluruh teks dengan huruf kecil (non kapital).
   c. Melakukan *tokenizing*, yaitu memecah *tweet* menjadi kata per kata.
   d. Melakukan *stemming*, yaitu menghilangkan kata imbuhan sehingga didapatkan kata dasar.
   e. Menghapus kata pada *tweet* yang terdapat pada *stopwords*.
3. Mengubah data *tweet* ke dalam bentuk frekuensi kemunculan kata menggunakan metode TF-IDF.
4. Melakukan *clustering* data
   a. Melakukan *clustering* dengan metode DBSCAN
      i. Melakukan *clustering* sesuai langkah-langkah pada subbab 2.5
      ii. Menghitung nilai *silhouette coefficient* setiap kombinasi *Eps* dan memilih parameter *Eps* yang memiliki nilai *silhouette coefficient* tertinggi
   b. Melakukan *clustering* dengan metode *K-Means*
      i. Melakukan *clustering* sesuai langkah-langkah pada subbab 2.6
   c. Memilih hasil *clustering* terbaik dengan melihat nilai *silhoutte coefficient* terbesar.
5. Melakukan visualisasi *tweet* dengan *Word Cloud*

6. Melakukan interpretasi dan menarik kesimpulan.

3.4. Diagram Alir

Langkah analisis sebagaimana telah dijelaskan pada sebelumnya dapat digambarkan dengan diagram alir yang dapat dilihat pada Gambar 3.1.



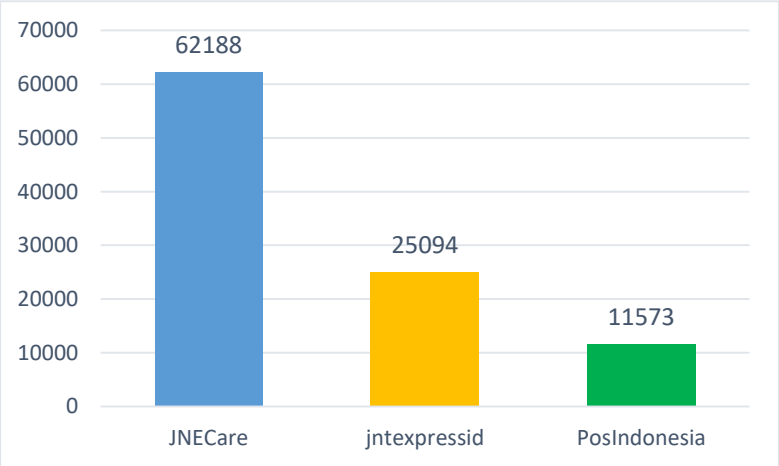
Gambar 3.1 Diagram Alir

Halaman Sengaja Dikosongkan

BAB IV
ANALISIS DAN PEMBAHASAN

4.1. Karakteristik Data

Dari hasil *crawling* menggunakan Twitter API, didapatkan 62.188 *tweet* yang ditujukan kepada akun layanan ekspedisi JNE yang diambil dari tanggal 17 Februari 2019 – 7 Mei 2019, 25.094 *tweet* yang ditujukan kepada akun layanan ekspedisi J&T yang diambil dari tanggal 17 Februari 2019 – 7 Mei 2019, dan 11.573 *tweet* yang ditujukan kepada akun layanan ekspedisi Pos Indonesia yang diambil dari tanggal 4 Februari 2019 – 7 Mei 2019. Data yang telah didapatkan perlu dilakukan pendeskripsian data. Pendeskripsian karakteristik data berupa jumlah *tweet* yang didapatkan dari hasil *crawling* yang digambarkan dalam bentuk diagram batang. Berikut adalah perbandingan jumlah *tweet* pada akun layanan ekspedisi JNE, JNT, dan Pos Indonesia ditampilkan pada Gambar 4.1.



Gambar 4.1 Perbandingan Jumlah *Tweet* pada Akun Layanan Ekspedisi.

Berdasarkan Gambar 4.1 menunjukkan bahwa *tweet* yang paling banyak diberikan oleh masyarakat adalah *tweet* pada akun layanan ekspedisi JNE. Hal ini dapat dilihat bahwa layanan

ekspedisi JNE lebih sering digunakan oleh masyarakat untuk mengirim barang. Sedangkan jumlah *tweet* yang paling sedikit diberikan oleh masyarakat adalah *tweet* yang ditujukan pada layanan ekpedisi Pos Indonesia.

Sebelum melakukan analisis *clustering* perlu dilakukan *text preprocessing* pada kumpulan *tweet* yang dihasilkan. *Text pre-processing* dilakukan agar data mentah yang didapatkan dapat menjadi tersruktur untuk mempermudah analisis. *Text pre-processing* yang dilakukan yaitu menghapus link, menghapus *hastag*, menghapus simbol retweet (RT), menghapus *username*, menghapus angka, menghapus *emoticon*, menghapus baris kosong, menghapus *punctuation*, menghapus spasi berlebih, *case folding*, *stemming*, *stopword*, dan *tokenizing*. Berikut adalah contoh *preprocessing* data *tweet* pada salah akun yang ditujukan kepada akun layanan ekspedisi JNE (@JNECare).

Tabel 4.1 Contoh Praproses

Praproses	Hasil
Tweet	@JNECare siang JNE, nama saya wahyu, no resi saya 400180011290819. Saya cek status pengiriman paket saya rumah tidak di huni. Padahal saya lagi beribadah tadi waktu ada panggilan mas kurirnya. Apakah barang di antar lagi atau bagaimana ?
Menghapus *username*	siang JNE, nama saya wahyu, no resi saya 400180011290819. Saya cek status pengiriman paket saya rumah tidak di huni. Padahal saya lagi beribadah tadi waktu ada panggilan mas kurirnya. Apakah barang di antar lagi atau bagaimana ?
Menghapus angka	siang JNE, nama saya wahyu, no resi saya . Saya cek status pengiriman paket saya rumah tidak di huni. Padahal saya lagi beribadah tadi waktu ada panggilan mas kurirnya. Apakah barang di antar lagi atau bagaimana ?
Menghapus *punctuation*	siang JNE nama saya wahyu no resi saya Saya cek status pengiriman paket saya rumah tidak di huni Padahal saya lagi beribadah tadi waktu ada panggilan mas kurirnya Apakah barang di antar lagi atau bagaimana

Tabel 4.1 Contoh Praproses (Lanjutan)

Praproses	Hasil
Menghapus spasi berlebih	siang JNE nama saya wahyu no resi saya Saya cek status pengiriman paket saya rumah tidak di huni Padahal saya lagi beribadah tadi waktu ada panggilan mas kurirnya Apakah barang di antar lagi atau bagaimana
Case folding	siang jne nama saya wahyu no resi saya saya cek status pengiriman paket saya rumah tidak di huni padahal saya lagi beribadah tadi waktu ada panggilan mas kurirnya apakah barang di antar lagi atau bagaimana
Stemming	siang jne nama saya wahyu no resi saya saya cek status kirim paket saya rumah tidak di huni padahal saya lagi ibadah tadi waktu ada panggil mas kurir apakah barang di antar lagi atau bagaimana
Stopwords	siang jne nama wahyu no resi cek status kirim paket rumah tidak huni ibadah panggil kurir barang

4.2. Layanan Ekspedisi JNE

Langkah selanjutnya adalah perhitungan frekuensi kemunculan masing-masing kata pada setiap *tweet* yang selanjutnya akan dihitung bobot atau nilai TF-IDF. Berikut adalah struktur data berdasarkan frekuensi kemunculan kata yang diperoleh setelah dilakukan *text prepocessing* pada akun layanan ekspedisi JNE yang ditampilkan pada Tabel 4.2.

Tabel 4.2 Struktur Data Layanan Ekspedisi JNE Setelah Dilakukan *Preprocessing*

Tweet ke-	Kata						
	alamat	…	cek	…	kirim	…	nomor
1	0	…	0	…	0	…	0
2	2	…	1	…	1	…	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮
15.003	1	…	0	…	0	…	2
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮
35.123	0	…	0	…	0	…	0

Tabel 4.2 menunjukkan bahwa frekuensi kemunculan setiap kata pada setiap *tweet* yang didapatkan dari proses *text pre-processing*. Diketahui bahwa jumlah kata telah berkurang dikarenakan ada beberapa kata yang tidak memiliki makna dalam kalimat dihapus dan tidak digunakan. Jumlah dari kata yang dihasilkan setelah dilakukan *text preprocessing* adalah 339 kata dan kata-kata tersebut merupakan variabel penelitian yang akan digunakan. Sebelum dilakukan *text clustering*, perlu dilakukan pembobotan pada kata dasar dari setiap *tweet* dengan menggunakan *Term Frequency – Inverse Document Frequency* (TF-IDF). Perhitungan TF-IDF mengggunakan rumus pada persamaan (2.1) dan (2.2). Berikut adalah hasil perhitungan TF-IDF yang ditampilkan pada Tabel 4.3.

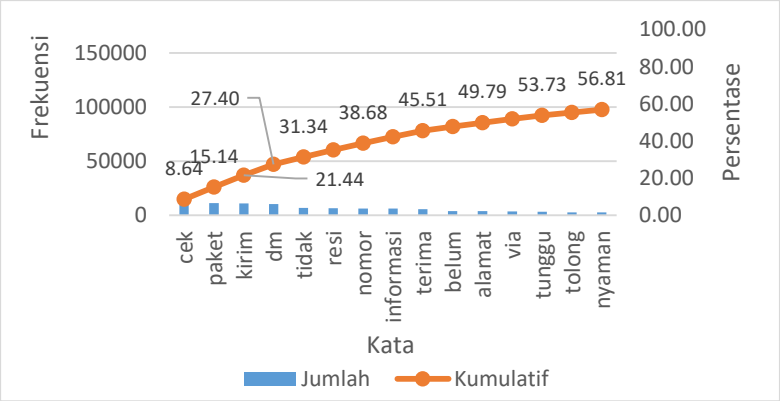
Tabel 4.3 Hasil Perhitungan TF-IDF Layanan Ekspedisi JNE

Tweet ke-	Kata						
	alamat	…	cek	…	kirim	…	nomor
1	0	…	0	…	0	…	0
2	0,4032	…	0,1380	…	0,1447	…	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮
15.003	0,2016	…	0	…	0	…	0,3384
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮
35.123	0	…	0	…	0	…	0

Hasil perhitungan TF-IDF pada Tabel 4.3 diketahui bahwa kata “alamat” pada *tweet* ke-2 memiliki bobot sebesar 0,4032 dan pada *tweet* ke-15.003 memiliki bobot sebesar 0,2016. Hal ini menunjukkan bahwa kata “alamat” pada *tweet* ke-2 memiliki frekuensi kemunculan yaitu 2 × 0,2016 = 0,4032, sedangkan pada *tweet* ke-15.003 memiliki frekuensi kemunculan kata yaitu 1 × 0,2016 = 0,2016. Begitu pula perhitungan TF-IDF dari kata-kata lainnya. Selanjutnya dilakukan visualisasi data menggunakan *word cloud* dengan teknik *n*-gram, yaitu dengan *n* sebesar 1 atau *unigram*. Berikut adalah *word cloud* yang dibentuk berdasarkan bobot TF-IDF dari setiap kata pada akun layanan JNE yang ditampilkan pada Gambar 4.2.

Gambar 4.2 Visualisasi *Word Cloud Unigram* pada Layanan Ekspedisi JNE.

Berdasarkan diagram pareto pada Gambar 4.3, diketahui bahwa jika diambil 30% dari total frekuensi kemunculan kata, maka kata yang sering muncul adalah pada *tweet* yang ditujukan pada layanan ekspedisi JNE adalah "cek", "paket", "kirim", dan "dm". Hal ini sesuai dengan hasil visualisasi *word cloud unigram* pada Gambar 4.2. Kata yang lebih diutamakan untuk ditindak lanjuti adalah 30% dari total frekuensi kemunculan kata.



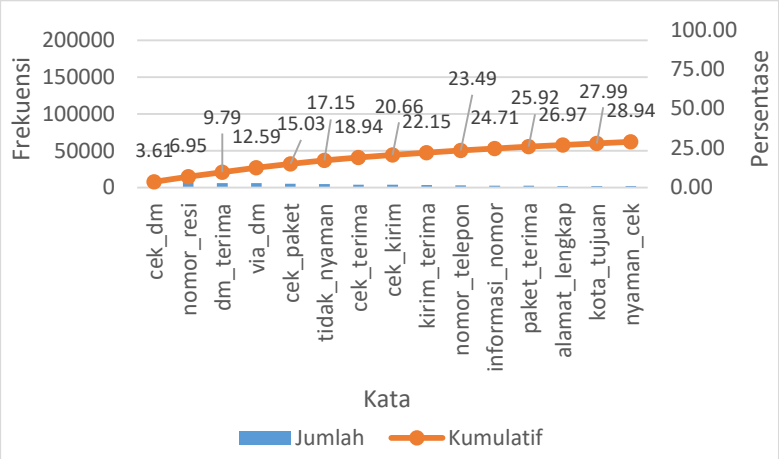
Gambar 4.3 Diagram Pareto *Unigram* Layanan Ekspedisi JNE

Berdasarkan dari visualisasi *word cloud* dengan menggunakan teknik *unigram* dirasa belum memberikan informasi yang lengkap. Sehingga digunakan *word cloud* dengan *n* sebesar 2 atau *bigram*. Berikut adalah visualisasi *word cloud* menggunakan bigram yang ditampilkan pada Gambar 4.4.

Gambar 4.4 Visualisasi *Word Cloud Bigram* pada Akun Layanan Ekspedisi JNE.

Gambar 4.5 adalah diagram pareto yang menunjukkan bahwa jika pasangan kata diambil 15% dari total frekuensi kemunculan kata, maka pasangan kata yang sering muncul adalah pada *tweet* yang ditujukan pada layanan ekspedisi JNE adalah “cek_dm”, “nomor_resi”, “dm_terima”, dan “via_dm”. Pasangan kata tersebut sesuai dengan hasil visualisasi *word cloud bigram* pada Gambar 4.2. Kata yang lebih diutamakan untuk ditindak lanjut adalah 15% dari total frekuensi kemunculan. Kata “cek dm” merupakan bentuk permintaan pelanggan kepada pihak ekspedisi JNE untuk membaca dan merespon dm atau pesan yang dikirimkan pelanggan. Pasangan kata “dm terima” adalah bentuk informasi yang disampaikan oleh

pihak ekspedisi JNE bahwa dm yang dikirim oleh pelanggan telah diterima, sedangkan pasangan kata “nomor resi” menunjukkan alat bukti pelacakan barang atau paket yang dikirimkan.



Gambar 4.5 Diagram Pareto *Bigram* Layanan Ekspedisi JNE.

4.2.1 *Clustering* menggunakan Metode DBSCAN

Dari hasil *preprocessing*, data selanjutkan akan dianalisis menggunakan *clustering*. Metode *clustering* yang digunakan adalah metode *Density-Based Spatial Clustering of Applications with Noise* (DBSCAN) dan *K-Means*. Berikut adalah hasil analisis *clustering* menggunakan metode DBSCAN dan *K-Means*. DBSCAN adalah metode yang menggunakan algoritma pengelompokkan yang didasarkan pada kepadatan (*density*) data. Hasil *clustering* yang terbentuk dari metode DBSCAN akan di evaluasi menggunakan *silhouette coefficient*. Berikut adalah ilustrasi perhitungan manual *clustering text* menggunakan metode DBSCAN.

1. Data Ilustrasi yang digunakan ditampilkan pada Tabel 4.4.

Tabel 4.4 Data Ilustrasi

Tweet	Kata	
	Cek	DM
1	0,6557	0,7550
2	0	0
3	0	0
4	0	0
5	0	0
6	0	0,4935
7	0	0
8	0	0
9	0,0997	0,1148
10	0	0,4023

2. Menentukan nilai parameter *MinPts* dan *Eps*

Proses penentuan parameter *MinPts* dan *Eps* berdasarkan hasil kombinasi *Eps* dengan *MinPts* yang memiliki nilai *silhouette coefficient* tertinggi. Nilai *MinPts* yang dipilih sebesar 5 dengan nilai *Eps* yang dipilih sebesar 1.

3. Memilih titik pusat awal secara acak.

Pemilihan titik pusat awal dapat dipilih secara manual atau acak. Titik pusat awal yang dipilih adalah sebagai berikut.

Pusat titik A = [0,076, 0,177]

4. Menghitung jarak titik *core point* dengan *point* yang lain

Perhitungan jarak titik *core point* dengan *point* yang lain menggunakan jarak *Euclidean*. Perhitungan jarak *Euclidean* menggunakan rumus pada persamaan (2.3). Berikut adalah hasil perhitungan jarak *Euclidean* yang ditampilkan pada Tabel 4.5.

Tabel 4.5 Hasil Perhitungan Jarak *Euclidean* DBSCAN

Jarak	Hasil	Jarak	Hasil
1A	0,8193	3A	0,1920
2A	0,1920	4A	0,1920

Tabel 4.5 Hasil Perhitungan Jarak *Euclidean* DBSCAN (Lanjutan)

Jarak	Hasil	Jarak	Hasil
5A	0,1920	8A	0,1920
6A	0,3259	9A	0,0663
7A	0,1920	10A	0,2380

Berdasarkan hasil perhitungan jarak *Euclidean* pada Tabel 4.4, dilanjutkan pada langkah selanjutnya.

5. Melakukan pengelompokkan berdasarkan hasil *cluster*

Pengelompokkan dilakukan berdasarkan jarak *Euclidean* yaitu jarak titik *core point* dengan *point lainnya* yang memiliki jarak tidak lebih atau sama dengan nilai *Eps* yang ditentukan. Berikut data yang memenuhi syarat dilihat dari kedekatan jarak antara titik *core point* dengan *point* yang lain ditampilkan pada Tabel 4.6.

Tabel 4.6 Pemilihan Titik *Core Point*

Tweet	Kata	
	Cek	DM
1	0,6557	0,7550
2	0	0
3	0	0
4	0	0
5	0	0
6	0	0,4935
7	0	0
8	0	0
9	0,0997	0,1148
10	0	0,4023

Berdasarkan jumlah titik yang terpilih adalah 10. Jumlah ini sudah memenuhi terbentuknya *neighborhood core object* karena jumlah objek *e-neighborhood* sudah memenuhi jumlah *MinPts* = 5, dimana syarat terbentuknya *neighborhood core object* adalah sebuah objek berisi setidaknya jumlah minimal *MinPts*.

6. Menentukan titik pusat (*core point*) baru

Penentuan titik pusat (*core point*) yang baru dilakukan dengan cara memilih titik yang memiliki jarak terjauh yang masih termasuk dalam *core object* pada iterasi sebelumnya. Berikut adalah titik-titik yang terpilih berdasarkan jarak terjauh yang ditampilkan pada Tabel 4.7.

Tabel 4.7 Pemilihan Titik *Core Point* Baru

Tweet	Kata		Jarak
	Cek	DM	
1	0,6557	0,7550	**0,8193**
2	0	0	0,1920
3	0	0	0,1920
4	0	0	0,1920
5	0	0	0,1920
6	0	0,4935	0,3259
7	0	0	0,1920
8	0	0	0,1920
9	0,0997	0,1148	0,0663
10	0	0,4023	0,2380

Berdasarkan Tabel 4.7, diketahui bahwa titik pusat yang bisa dipilih untuk iterasi selanjutnya adalah titik pusat pada *tweet* 1. Selanjutnya ulangi langkah-langkah diatas hingga semua point telah diproses. Jika tidak ada *tweet* yang tidak masuk kedalam *cluster* manapun akan ditandai sebagai *noise*.

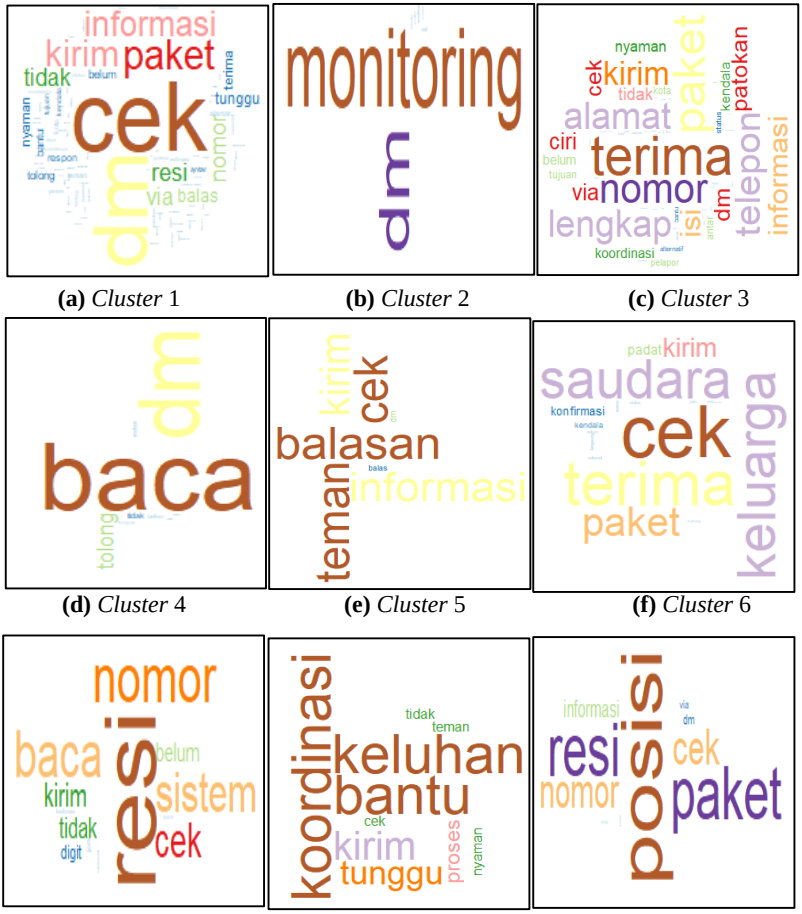
Clustering dengan metode DBSCAN menggunakan *MinPts* sebesar 50 dengan berbagai kombinasi *Eps*. Kombinasi *Eps* yang digunakan adalah 0,6 – 0,68. Parameter *Eps* digunakan untuk menentukan radius (jarak maksimum) *tweet* anggota *cluster* dari pusat *cluster*. Parameter *Eps* yang digunakan, ditentukan oleh peneliti dari hasil percobaan secara manual. Penggunaan parameter *Eps* pada layanan ekspedisi JNE berbeda dengan layanan ekspedisi J&T dan Pos Indonesia dikarenakan ketika menggunakan *Eps*

sebesar 0,1, memiliki jarak yang sangat dekat yang menyebabkan *tweet* tidak masuk *cluster* yang tepat dan jumlah *cluster* yang dihasilkan cukup besar. Sehingga perlu diperlebar jarak maksimum yang digunakan. Ketika mengguna-kan parameter *Eps* sebesar 1, jumlah *cluster* yang terbentuk adalah 1 (tidak terbentuk *cluster*).

Tabel 4.8 Nilai *Silhouette Coefficient* dari Metode DBSCAN pada Akun Layanan Ekspedisi JNE

Eps	***Silhouette Coefficient***
0,6	0,26518
0,61	0,21345
0,62	0,19498
0,63	0,09340
0,64	0,12935
0,65	0,01673
0,66	0,03505
0,67	0,03288
0,68	0,02545

Tabel 4.8 menunjukkan bahwa nilai *silhouette coefficient* tertinggi yang diperoleh dari metode DBSCAN pada akun layanan ekspedisi JNE dilakukan menggunakan parameter *Eps* sebesar 0,6. Nilai *silhouette* coefficient yang dihasilkan dengan menggunakan parameter *Eps* sebesar 0,6 dan *MinPts* sebesar 50 adalah 0,26518. Jumlah *cluster* yang diperoleh dari *clustering* menggunakan metode DBSCAN dengan *MinPts* sebesar 50 dan *Eps* sebesar 0,6 adalah sebanyak 18 *cluster* dengan 15.209 *tweet* sebagai *noise*. Oleh karena itu, dengan menggunakan *MinPts* sebesar 50 diperoleh *Eps* optimum sebesar 0,6. Berikut adalah hasil *clustering* dari metode DBSCAN yang ditampilkan secara visual menggunakan *word cloud* pada akun layanan ekspedisi JNE ditunjukkan pada Gambar 4.6.



(a) *Cluster* 1 **(b)** *Cluster* 2 **(c)** *Cluster* 3

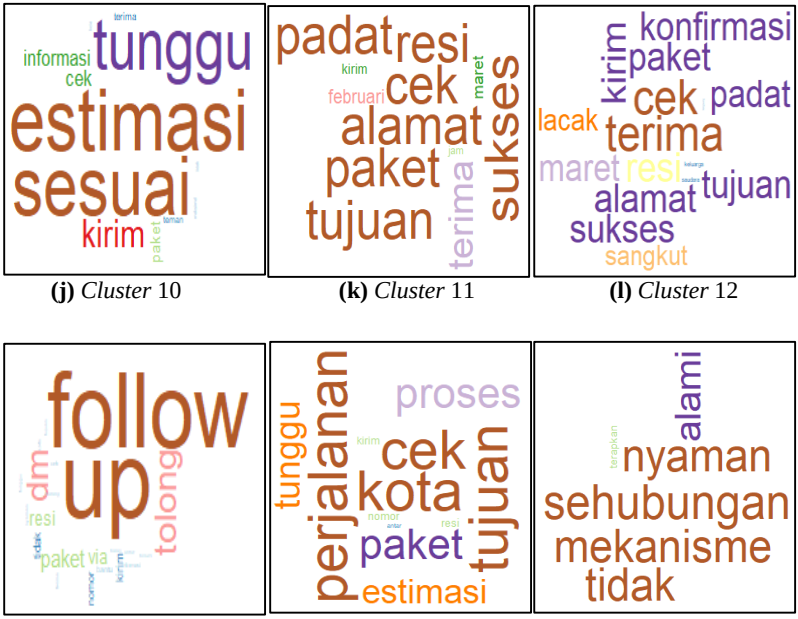
(d) *Cluster* 4 **(e)** *Cluster* 5 **(f)** *Cluster* 6

(g) *Cluster* 7 **(h)** *Cluster* 8 **(i)** *Cluster* 9

Gambar 4.6 *Word Cloud* Hasil *Cluster* DBSCAN Layanan Ekspedisi JNE 9 *Cluster* Pertama; **(a)** *Cluster* 1, **(b)** *Cluster* 2, **(c)** *Cluster* 3, **(d)** *Cluster* 4, **(e)** *Cluster* 5, **(f)** *Cluster* 6, **(g)** *Cluster* 7, **(h)** *Cluster* 8, dan **(i)** *Cluster* 9.

Berdasarkan Gambar 4.6, *cluster* 1 berisi pasangan kata yang berkaitan tentang apresiasi terhadap permintaan cek dm. *Cluster* 2 berisi pasangan kata yang berkaitan tentang berkaitan sistem pengawasan. *Cluster* 3 berisi pasangan kata yang berkaitan tentang

permohonan melengkapi informasi tambahan. *Cluster* 4 berisi pasangan kata yang berkaitan tentang permintaan baca dm. *Cluster* 5 berisi pasangan kata yang berkaitan tentang permintaan pengecekan barang. *Cluster* 6 berisi pasangan kata yang berkaitan konfirmasi penerima paket. *Cluster* 7 berisi pasangan kata yang berkaitan tentang informasi pengecekan paket. *Cluster* 8 berisi pasangan kata yang berkaitan tentang permintaan koordinasi untuk meng-atasi keluhan. *Cluster* 9 berisi pasangan kata yang berkaitan posisi paket.



(j) *Cluster* 10 **(k)** *Cluster* 11 **(l)** *Cluster* 12

(m) *Cluster* 13 **(n)** *Cluster* 14 **(o)** *Cluster* 15

Gambar 4.7 *Word Cloud* Hasil *Cluster* DBSCAN Layanan Ekspedisi JNE 6 *Cluster*; **(j)** *Cluster* 10, **(k)** *Cluster* 11, **(l)** *Cluster* 12, **(m)** *Cluster* 13, **(n)** *Cluster* 14, dan **(o)** *Cluster* 15.

Gambar 4.7 menunjukkan visualisasi hasil *cluster* diketahui bahwa *cluster* 10 berisi pasangan kata yang berkaitan tentang estimasi pengiriman. *Cluster* 11 berisi pasangan kata yang

berkaitan tentang informasi estimasi pengiriman. *Cluster* 12 berisi pasangan kata yang berkaitan tentang informasi paket diterima. *Cluster* 13 berisi pasangan kata yang berkaitan tentang konfirmasi pengiriman diterima. *Cluster* 14 berisi pasangan kata yang berkaitan tentang informasi informasi estimasi pengiriman paket. *Cluster* 15 berisi pasangan kata yang berkaitan tentang laporan mekanisme.

(p) *Cluster* 16 **(q)** *Cluster* 17 **(r)** *Cluster* 18

Gambar 4.8 *Word Cloud* Hasil *Cluster* DBSCAN Layanan Ekspedisi JNE 3 *Cluster* Terakhir; **(p)** *Cluster* 16, **(q)** *Cluster* 17, dan **(r)** *Cluster* 18.

Berdasarkan hasil *cluster* menggunakan metode DBSCAN pada Gambar 4.8 diketahui bahwa *cluster* 16 berisi pasangan kata yang berkaitan tentang informasi balasan. *Cluster* 17 berisi pasangan kata yang berkaitan tentang informasi pengiriman padat, sedangkan *cluster* 18 berisi pasangan kata yang berkaitan tentang informasi jasa layanan.

4.2.2 *Clustering* menggunakan Metode *K-Means*

K-Means adalah metode *clustering* yang memiliki kemampuan untuk mengelompokkan data dalam jumlah yang cukup besar dengan waktu komputasi. Berdasarkan data ilustrasi pada Tabel 4.3 dilakukan perhitungan manual *clustering text* menggunakan metode *K-Means*.

1. Memilik secara acak *k centroid* awal dalam data

   Inisialisasi *k centroid* dapat ditentuka secara manual ataupun random. Inisialisai yang dipilih adalah

C1 = [0,6557, 0,7550] dan C2 = [0, 0]

2. Menghitung jarak setiap kata terhadap pusat *cluster* (*centroid*)

Perhitungan jarak setiap kata terhadap pusat *cluster* (*centroid*) terdekat menggunakan *Euclidean distance*. Rumus jarak *Eucludian* yang digunakan sesuai pada persamaan (2.4). Berikut adalah hasil perhitungan jarak setiap kata terhadap pusat *cluster* (*centroid*) C1 dan C2 yang ditampilkan pada Tabel 4.9.

Tabel 4.9 Perhitungan Jarak Setiap Kata terhadap *Centroid*

Variabel	**Centroid C1**	**Variabel**	**Centroid C1**
d(X1,C1)	0	d(X1,C2)	1
d(X2,C1)	1	d(X2,C2)	0
d(X3,C1)	1	d(X3,C2)	0
d(X4,C1)	1	d(X4,C2)	0
d(X5,C1)	1	d(X5,C2)	0
d(X6,C1)	0,70590	d(X6,C2)	0,49355
d(X7,C1)	1	d(X7,C2)	0
d(X8,C1)	1	d(X8,C2)	0
d(X9,C1)	0,84794	d(X9,C2)	0,15206
d(X10,C1)	0,74454	d(X10,C2)	0,40231

3. Mengelompokkan data sesuai dengan hasil *cluster*

Data dikelompokkan berdasarkan hasil *cluster* yaitu data yang memiliki perhitungan *centroid* yang terdekat dengan jarak *Euclidean*. Jika nilai $d(X_i, C_1) < d(X_2, C_2)$, maka data akan masuk ke dalam *cluster* 1. Begitupun sebaliknya. Berikut adalah pengelompokkan hasil *cluster* yang ditampilkan pada Tabel 4.10.

Tabel 4.10 Hasil *Cluster* Ilustrasi

Tweet	Kata		d(Xi,C1)	d(Xi,C1)	*Cluster*
	Cek	DM			
1	0,6557	0,7550	0,000	1,000	1
2	0	0	1	0	2

Tabel 4.10 Hasil *Cluster* Ilustrasi (Lanjutan)

Tweet	Kata		d(Xi,C1)	d(Xi,C1)	*Cluster*
	Cek	DM			
3	0	0	1	0	2
4	0	0	1	0	2
5	0	0	1	0	2
6	0	0,4935	0,7059	0,4935	2
7	0	0	1	0	2
8	0	0	1	0	2
9	0,0997	0,1148	0,8479	0,1521	2
10	0	0,4023	0,7445	0,4023	2

4. Hitung ulang pusat *cluster* (*centroid*) yang baru

Centroid baru dihitung berdasarkan nilai rata-rata data pada setiap *cluster*. Jika *centroid* baru berbeda dengan *centroid* sebelumnya, maka proses *clustering* selesai. Perhitungan *centroid* baru menggunakan rumus pada persamaan (2.5). Berikut adalah hasil perhitungan *centroid* baru yang ditampilkan pada Tabel 4.11.

Tabel 4.11 *Centroid* Baru

		Kata	
		Cek	DM
Centroid	C1	0	1
	C2	0,92204	0,1164

5. Menghitung jarak setiap kata terhadap *centroid* baru

Berdasarkan *centroid* baru, dilakukan perhitungan jarak setiap kata terhadap *centroid*. Berikut adalah perhitungan jarak setiap kata terhadap *centroid* baru yang ditampilkan pada Tabel 4.12.

Tabel 4.12 Perhitungan Jarak Setiap Kata terhadap *Centroid* Baru

Variabel	Centroid C1	Variabel	Centroid C1
d(X1,C1)	0,7000	d(X1,C2)	0,6919

Tabel 4.12 Perhitungan Jarak Setiap Kata terhadap *Centroid* Baru (Lanjutan)

Variabel	Centroid C1	Variabel	Centroid C1
d(X2,C1)	1	d(X2,C2)	0,9294
d(X3,C1)	1	d(X3,C2)	0,9294
d(X4,C1)	1	d(X4,C2)	0,9294
d(X5,C1)	1	d(X5,C2)	0,9294
d(X6,C1)	0,5065	d(X6,C2)	0,9962
d(X7,C1)	1	d(X7,C2)	0,9294
d(X8,C1)	1	d(X8,C2)	0,9294
d(X9,C1)	0,8908	d(X9,C2)	0,8223
d(X10,C1)	0,5977	d(X10,C2)	0,9653

6. Mengelompokkan data sesuai dengan hasil *cluster*

Berikut adalah pengelompokkan data berdasarkan hasil *cluster* yang ditampilkan pada Tabel 4.13.

Tabel 4.13 Hasil *Cluster* Ilustrasi Menggunakan *Centroid* Baru

Tweet	Cek	DM	d(Xi,C1)	d(Xi,C1)	*Cluster*
1	0,656	0,755	0,423	0,692	1
2	0	0	1,414	0,929	2
3	0	0	1,414	0,929	2
4	0	0	1,414	0,929	2
5	0	0	1,414	0,929	2
6	0	0,4935	1,121	0,996	2
7	0	0	1,414	0,929	2
8	0	0	1,414	0,929	2
9	0,0997	0,1148	1,263	0,822	2
10	0	0,4023	1,165	0,965	2

Berdasarkan perhitungan *cluster* berdasarkan jarak terdekat dengan *centroid* baru, diketahui bahwa menghasilkan *cluster* yang sama dengan sebelumnya. Langkah ini diulangi terus-menerus

hingga menghasilkan *centroid* yang sama dengan *centroid* sebelumnya.

Dalam metode *K-Means*, perlu menentukan jumlah *cluster* (*K*) terlebih dahulu dengan menggunakan nilai *Variance Ratio Criterion* (VRC). VRC adalah metode terbaik dalam menentukan jumlah *cluster*. Jumlah *cluster* optimum adalah *cluster* memiliki nilai *Variance Ratio Criterion* paling tinggi. *Index* yang digunakan untuk penentuan jumlah *cluster* optimum dilakukan dengan *K* mulai dari 2 hingga 20. Berikut adalah nilai VRC pada data akun layanan ekspedisi JNE yang ditampilkan pada Tabel 4.14.

Tabel 4.14 Nilai VRC pada Layanan Ekspedisi JNE

Cluster	Nilai VRC	*Cluster*	Nilai VRC
2	**3.239,8522**	12	1.094,1687
3	2.304,4487	13	1.082,5548
4	1.949,2326	14	1.048,0313
5	1.751,9372	15	986,1774
6	1.613,7094	16	970,2019
7	1.459,8178	17	922,2360
8	1.391,1281	18	915,0433
9	1.278,7757	19	877,9982
10	1.182,0085	20	854,2907
11	1.122,9674		

Berdasarkan hasil nilai VRC pada Tabel 4.14 menunjukkan jumlah *cluster* (*K*) terbentuk sebanyak 20 dan yang memiliki nilai VRC paling tinggi yaitu sebesar 3.239,8552 pada 2 *cluster*. Hasil *clustering* yang terbentuk dari metode *K-Means* akan di evaluasi menggunakan *silhouette coefficient*. Berikut adalah hasil nilai *silhouette coefficient* yang ditampilkan pada Tabel 4.15.

Tabel 4.15 Nilai *Silhouette Coefficient* dari Metode *K-Means* pada Layanan Ekspedisi JNE

Cluster	*Silhouette Coefficient*	*Cluster*	*Silhouette Coefficient*
2	**0,0819**	3	0,0744

Tabel 4.15 Nilai *Silhouette Coefficient* dari Metode *K-Means* pada Layanan Ekspedisi JNE (lanjutan)

Cluster	*Silhouette Coefficient*	*Cluster*	*Silhouette Coefficient*
4	0,0915	13	0,1406
5	0,1103	14	0,1441
6	0,1159	15	0,1422
7	0,1168	16	0,1406
8	0,1268	17	0,1451
9	0,1227	18	0,1437
10	0,1289	19	0,1562
11	0,1325	20	0,1541
12	0,1375		

Gambar 4.15 menunjukkan bahwa dari jumlah *cluster* sebanyak 2 (K = 2), diperoleh nilai *silhouette coefficient* sebesar 0.0819. Hal ini menunjukkan bahwa nilai *silhouette coefficient* yang dihasilkan adalah rendah yang artinya sebagian besar *tweet* berada diantara dua *cluster* sehingga kurang jelas harus dimasukkan ke dalam *cluster* yang mana. Hasil *clustering* pada layanan ekspedisi JNE menggunakan metode *K-Means* ditampilkan dengan visualisasi *word cloud* yang ditunjuk-kan pada Gambar 4.9.

Hasil *clustering* menggunakan metode *K-Means* pada layanan ekspedisi JNE yang ditunjukkan pada Gambar 4.9 diketahui bahwa hasil *cluster* optimum yang dihasilkan adalah 2 *cluster*, dimana *cluster* 1 berisi kata yang memiliki bobot yang paling besar, yaitu “cek” dan “dm” dan didukung dengan kata lain yaitu “kirim”, “tidak”, “balas”. Kata-kata tersebut merupakan bentuk keluhan *customer* kepada pihak layanan ekspedisi JNE dikarenakan tidak membalas atau merespon dm (*direct message*), sedangkan *cluster* 2, frekuensi kemunculan kata yang paling banyak adalah kata “cek” serta didukung oleh kata–kata lain yaitu “kirim”, “paket”, “informasi”, dan “tidak”. Kata tersebut berkaitan tentang

permohonan dan pengecekan informasi mengenai paket yang dikirim.

(a) *Cluster* 1

(b) *Cluster* 2

Gambar 4.9 Visualisasi *Word Cloud* pada Hasil *Cluster K-Means* Layanan Ekspedisi JNE; **(a)** *Cluster* 1 dan **(b)** *Cluster* 2

Berikut adalah perbandingan antara metode DBSCAN dengan *K-Means* pada akun layanan ekspedisi JNE yang dilihat berdasarkan nilai *silhouette coefficient* tertinggi ditampikan pada Tabel 4.16.

Tabel 4.16 Perbandingan Metode DBSCAN dan *K-Means* pada Layanan Ekspedisi JNE

Metode *Clustering*	***Silhouette Coefficient***	**Jumlah *Cluster***
K-Means	0,0819	2
DBSCAN	0,26518	18

Berdasarkan nilai *silhouette coefficient* yang terdapat pada Tabel 4.16, diketahui bahwa metode DBSCAN lebih baik apabila dibandingkan dengan metode *K-Means.* Hal ini dikarenakan, DBSCAN memiliki nilai *silhouette coefficient* tertinggi yaitu sebesar 0,1542.

4.3. Layanan Ekspedisi J&T

Berikut adalah struktur data yang diperoleh setelah dilakukan *text prepocessing* pada akun layanan ekspedisi J&T yang ditampilkan pada Tabel 4.17.

Tabel 4.17 Struktur Data Layanan Ekspedisi J&T Setelah Dilakukan *Preprocessing*

Tweet ke-	Kata						
	barang	…	nama	…	status	…	terima
1	0	…	0	…	0	…	0
2	0	…	0	…	1	…	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮
10.006	0	…	0	…	0	…	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮
23.775	1	…	0	…	0	…	2

Jumlah dari frekuensi kemunculan setiap kata pada setiap *tweet* setelah dilakukan *preprocessing* yang ditampilkan pada Tabel 4.17 diketahui sebanyak 464 kata. Kata-kata tersebut yang akan dipergunakan sebagai variabel penelitian. Berikut adalah hasil perhitungan TF-IDF yang ditampilkan pada Tabel 4.18.

Tabel 4.18 Hasil Perhitungan TF-IDF Layanan Ekspedisi J&T

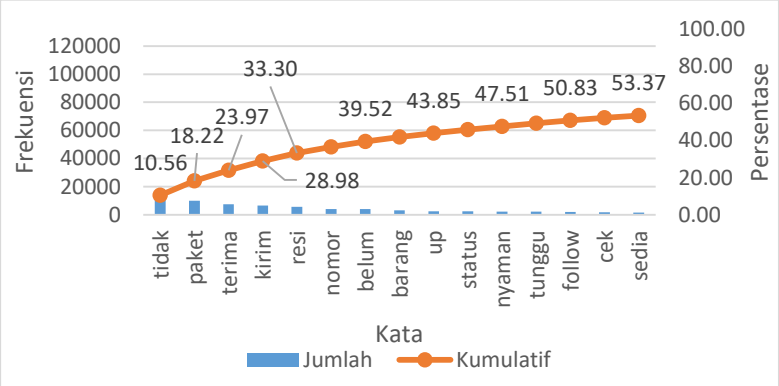
Tweet ke-	Kata						
	barang	…	nama	…	status	…	terima
1	0	…	0	…	0	…	0
2	0	…	0	…	0,4690	…	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮
10.006	0	…	0,3527	…	0	…	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮
23.775	0,6839	…	0	…	0	…	0,2719

Tabel 4.18 merupakan hasil perhitungan TF-IDF pada layanan ekspedisi J&T yang menunjukkan bahwa kata “barang” pada *tweet* ke-23.775 memiliki bobot sebesar 0,6839. Hal ini menunjukkan bahwa kata “barang” pada *tweet* ke-23.775 memiliki frekuensi kemunculan yaitu 1 × 0,6839 = 0,6839, Begitu pula dengan perhitungan TF-IDF untuk kata-kata yang lain. Berikut visualisasi data menggunakan *word cloud* dibentuk berdasarkan bobot TF-IDF dari setiap kata pada akun layanan J&T yang ditampilkan pada Gambar 4.10.

Gambar 4.10 Visualisasi *Word Cloud Unigram* pada Layanan Ekspedisi J&T.

Gambar 4.11 adalah diagram pareto yang menunjukkan 30% kata yang sering muncul atau memiliki frekuensi kemunculan yang paling besar dari total frekuensi kemunculan semua kata adalah kata “tidak”, “paket”, “terima”, dan “kirim”. Hal ini sesuai dengan hasil visualisasi *word cloud unigram* pada Gambar 4.10. Kata yang perlu diutamakan untuk ditindak lanjuti yaitu 30% kata yang muncul dari total frekuensi kemunculan kata. Kata “paket” menunjukkan barang atau produk dalam bungkusan yang dikirim oleh perusahaan ekspedisi. Kata “tidak” merupakan bentuk kata penolakan atau penyangkalan. Karena kata “paket” dan “tidak” tidak cukup memberikan informasi yang lebih dan lengkap,

sehingga perlu dilakukan visualisasi *word cloud* menggunakan teknik *bigram*.

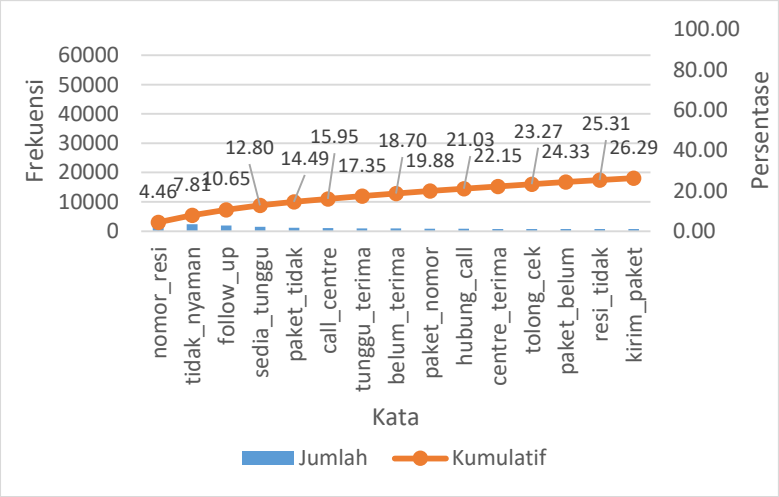


Gambar 4.11 Diagram Pareto *Unigram* Layanan Ekspedisi J&T

Berikut adalah visualisasi *word cloud* menggunakan teknik *bigram* pada akun layanan ekspedisi J&T yang ditampilkan pada Gambar 4.12.

Gambar 4.12 Visualisasi *Word Cloud Bigram* pada Akun Layanan Ekspedisi J&T.

Berdasarkan diagram pareto pada Gambar 4.13 diketahui bahwa 15% pasangan kata dari total frekuensi kemunculan kata adalah pasangan kata "nomor_resi", "tidak_nyaman", "*follow*_up", "sedia_tunggu", dan "paket_tidak". Pasangan kata tersebut sesuai dengan hasil visualisasi *word cloud bigram* pada Gambar 4.12. Kata yang perlu diutamakan untuk ditindak lanjuti adalah 15% pasangan kata dari total frekuensi kemunculan kata. Pasangan kata "nomor resi" merupakan nomor unik yang biasa digunakan pelanggan untuk melacak keberadaan lokasi paket yang sedang dikirim dan sebagai bukti/tanda terima pengiriman barang dari layanan ekspedisi untuk pelanggan.



Gambar 4.13 Diagram Pareto *Bigram* Layanan Ekspedisi J&T.

4.3.2 *Clustering* menggunakan Metode DBSCAN

Berikut adalah hasil *clustering* dengan menggunakan metode DBSCAN pada layanan ekspedisi J&T. Nilai parameter *MinPts* yang dipilih adalah sebesar 30. Penetuan parameter *Eps* dilakukan dengan melakukan berbagai kombinasi antara parameter *MinPts* dengan Eps, sehingga diperoleh nilai *silhouette coefficient*. Berikut adalah nilai *silhouette coefficient* hasil kombinasi dengan *MinPts*

sebesar 30 untuk layanan ekspedisi J&T yang ditampilkan pada Tabel 4.19.

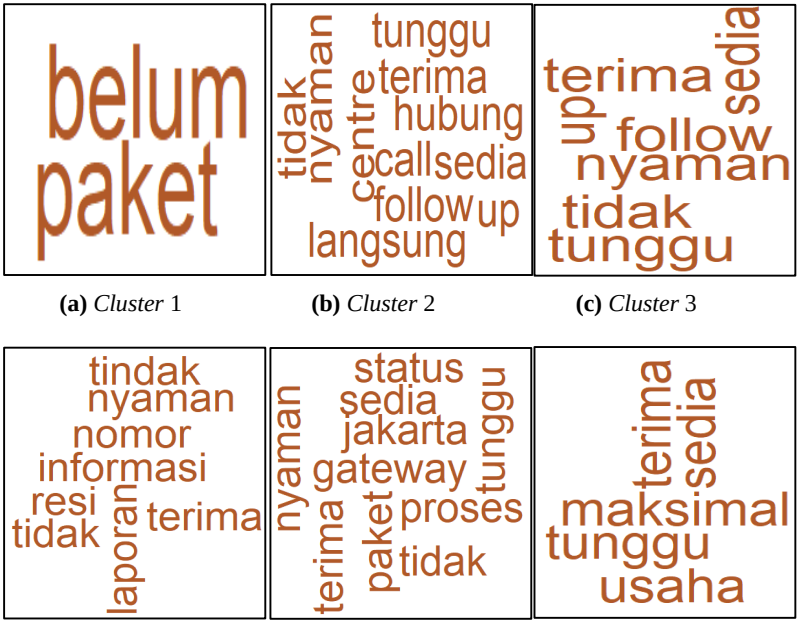
Tabel 4.19 Nilai *Silhouette Coefficient* dari Metode DBSCAN pada Akun Layanan Ekspedisi J&T

Eps	***Silhouette Coefficient***
0.1	0,9999464
0.2	0,9997138
0.3	0,9473227
0.4	0,7583361
0.5	0,3388869
0.6	0,1443148
0.7	0,0171568
0.8	0,0008924

Nilai *silhouette coefficient* tertinggi yang diperoleh dari metode DBSCAN pada akun layanan ekspedisi J&T dilakukan menggunakan parameter *Eps* sebesar 0,1 dilihat berdasarkan Tabel 4.19 adalah 0,9999464. Nilai *silhouette coefficient* tersebut dapat dikatakan bahwa hasil *clustering* telah sangat baik karena *tweet* di dalam *cluster* yang ada telah kompak dan *tweet* dalam suatu *cluster* telah jauh dari *cluster* yang lain. Oleh karena itu, jumlah *cluster* optimum yang diperoleh dari *clustering* meng-gunakan metode DBSCAN dengan *MinPts* sebesar 30 dan *Eps* sebesar 0,1 adalah sebanyak 22 *cluster* dengan 20.384 *tweet* sebagai *noise*. Berikut adalah hasil *clustering* dari metode DBSCAN yang ditampilkan secara visual menggunakan *word cloud* pada akun layanan ekspedisi J&T ditunjukkan pada Gambar 4.14.

Berdasarkan Gambar 4.14 yang merupakan visualisasi *word cloud* dari hasil *clustering* menggunakan metode DBSCAN pada layanan ekspedisi J&T, diketahui bahwa c*luster* 1 berisi pasangan kata yang berkaitan tentang informasi paket. *Cluster* 2 berisi pasangan kata yang berkaitan tentang layanan *call centre*. *Cluster* 3 berisi pasangan kata yang berkaitan tentang permohonan pengecekan keberadaan paket. *Cluster* 4 berisi pasangan kata yang berkaitan tentang permohonan informasi keberadaan paket. *Cluster* 5 berisi pasangan kata yang berkaitan tentang keluhan. *Cluster* 6

berisi pasangan kata yang berkaitan tentang peningkatan usaha maksimal.



(a) *Cluster* 1 **(b)** *Cluster* 2 **(c)** *Cluster* 3

(d) *Cluster* 4 **(e)** *Cluster* 5 **(f)** *Cluster* 6

Gambar 4.14 Visualisasi *Word Cloud* Hasil *Cluster* DBSCAN Layanan Ekspedisi J&T 6 *Cluster* Pertama*;* **(a)** *Cluster* 1, **(b)** *Cluster* 2, **(c)** *Cluster* 3, **(d)** *Cluster* 4, **(e)** *Cluster* 5, dan **(f)** *Cluster* 6.

Gambar 4.15 adalah visualisasi *word cloud* dari hasil *clustering* menggunakan metode DBSCAN, diketahui bahwa *cluster* 7 berisi pasangan kata yang berkaitan tentang kendala pengiriman. *Cluster* 8 berisi pasangan kata yang berkaitan tentang waktu penerimaan paket. *Cluster* 9 berisi pasangan kata yang berkaitan tentang informasi paket direima. *Cluster* 10 berisi pasangan kata yang berkaitan tentang informasi paket dikirim. *Cluster* 11 berisi pasangan kata yang berkaitan tentang pengiriman. *Cluster* 12 berisi pasangan kata yang berkaitan tentang permohonan menunggu. *Cluster* 13 berisi pasangan kata yang

berkaitan tentang informasi nomor resi dari paket yang dikirimkan. *Cluster* 14 berisi pasangan kata yang berkaitan tentang permohonan *follow up* untuk keberadaan paket. *Cluster* 15 berisi pasangan kata yang berkaitan tentang permohonan bantuan.

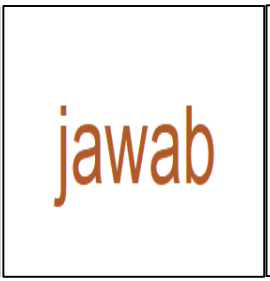
(g) *Cluster* 7 **(h)** *Cluster* 8 **(i)** *Cluster* 9

(j) *Cluster* 10 **(k)** *Cluster* 11 **(l)** *Cluster* 12

(m) *Cluster* 13 **(n)** *Cluster* 14 (**o)** *Cluster* 15

Gambar 4.15 Visualisasi *Word Cloud* Hasil *Cluster* DBSCAN Layanan Ekspedisi J&T 9 *Cluster* Berikutnya; **(g)** *Cluster* 7, **(h)** *Cluster* 8, **(i)** *Cluster* 9, **(j)** *Cluster* 10, **(k)** *Cluster* 11, **(l)** *Cluster* 12, **(m)** *Cluster* 13, **(n)** *Cluster* 14, dan **(o)** *Cluster* 15.

(p) *Cluster* 16



(q) *Cluster* 17

(**r**) *Cluster* 18

(s) *Cluster* 19

(t) *Cluster* 20

(**u**) *Cluster* 21

(v) *Cluster* 22

Gambar 4.16 Visualisasi *Word Cloud* Hasil *Cluster* DBSCAN Layanan Ekspedisi J&T 5 *Cluster* Terakhir; **(p)** *Cluster* 16, **(q)** *Cluster* 17, **(r)** *Cluster* 18, **(s)** *Cluster* 19, **(t)** *Cluster* 20, **(u)** *Cluster* 21, dan **(v)** *Cluster* 22.

Berdasarkan hasil *clustering* menggunakan metode DBSCAN pada Gambar 4.16 diketahui bahwa *cluster* 16 berisi pasangan kata yang berkaitan tentang permohonan atas tidak nyaman. *Cluster* 17 berisi pasangan kata yang berkaitan tentang

permohonan membalas. *Cluster* 18 berisi pasangan kata yang berkaitan tentang permohonan tindak lanjut pengiriman. *Cluster* 19 berisi pasangan kata yang berkaitan tentang informasi event. *Cluster* 20 berisi pasangan kata yang berkaitan tentang kendala pelacakan alamat. *Cluster* 21 berisi pasangan kata yang berkaitan tentang sistem pengiriman, sedangkan *cluster* 22 berisi pasangan kata yang berkaitan tentang laporan keterlambatan. Dari 22 *cluster* yang terbentuk terdapat beberapa *cluster* yang hanya memiliki 1 anggota yaitu *cluster* 11, *cluster* 12, *cluster* 15, dan *cluster* 17. Hal ini disebabkan oleh hasil *preprocessing* yang menghilangkan beberapa kata dalam suatu *tweet* dan hanya menyisakan satu kata.

4.3.2 *Clustering* menggunakan Metode *K-Means*

Berikut adalah hasil *clustering* dengan menggunakan metode *K-Means* pada layanan ekspedisi J&T. Penentuan jumlah kelompok optimum berdasarkan dari nilai *Variance Ratio Criterion* (VRC) yang dilihat dari tingginya nilai VRC. Berikut adalah tabel dari nilai VRC dari *K* mulai dari 2 hingga 20 pada data akun layanan ekspedisi J&T yang ditampilkan pada Tabel 4.20.

Tabel 4.20 Nilai VRC pada Layanan Ekspedisi J&T

Cluster	Nilai VRC	*Cluster*	Nilai VRC
2	**891,3695**	12	424,9503
3	812,7370	13	423,6279
4	700,6291	14	410,3131
5	648,5735	15	382,1339
6	571,9525	16	381,3517
7	543,5358	17	364,3592
8	477,6850	18	347,6021
9	490,4468	19	360,1340
10	453,4703	20	345,5969
11	441,6000		

Tabel 4.20 menunjukkan bahwa nilai VRC tertinggi adalah sebesar 891,3695 dengan *cluster* yang terbentuk adalah 2 *cluster*. Jumlah *cluster* optimum dipilih berdasarkan nilai VRC yang paling tinggi. Sehingga jumlah *cluster* optimum pada layanan ekspedisi J&T adalah 2 *cluster*. Berdasarkan penentuan jumlah kelompok yang optimum didapatkan nilai *silhouette clustering*. Berikut adalah hasil nilai *silhouette coefficient* yang ditunjukkan pada Tabel 4.21.

Tabel 4.21 Nilai *Silhouette Coefficient* dari Metode *K-Means* pada Akun Layanan Ekspedisi J&T

Cluster	***Silhouette Coefficient***	***Cluster***	***Silhouette Coefficient***
2	**0,04830**	12	0,06838
3	0,04416	13	0,07325
4	0,04717	14	0,07651
5	0,04907	15	0,07514
6	0,05161	16	0,07644
7	0,05539	17	0,07992
8	0,05715	18	0,08138
9	0,06145	19	0,08446
10	0,06614	20	0,08436
11	0,06343		

Berdasarkan dari hasil *clustering* yang terbentuk dari metode *K-Means* akan di evaluasi menggunakan *silhouette coefficient*. Hasil nilai *silhouette coefficient* ditampilkan pada Tabel 4.21, diketahui bahwa jumlah *cluster* sebanyak 2 ($K = 2$), diperoleh nilai *silhouette coefficient* sebesar 0.04830. Berikut adalah hasil *clustering* menggunakan metode *K-Means* pada layanan ekspedisi J&T divisualisasikan menggunakan *word cloud* yang ditunjukkan pada Gambar 4.17.

Gambar 4.17 menunjukkan bahwa hasil *clustering* dengan metode *K-Means* pada layanan ekspedisi J&T. Berdasarkan Gambar diketahui bahwa hasil *cluster* optimum yang dihasilkan adalah 2 *cluster*, dimana *cluster* 1 berisi kata yang memiliki frekuensi kemunculan paling banyak, yaitu “kirim” dan “paket”. Kata lain yang mendukung adalah kata “tidak”, “belum”, “terima”,

“tolong” “barang”, ”tidak”, dan lainnya. Kata-kata tersebut merupakan bentuk keluhan bahwa barang yang dikirimkan belum datang, sedangkan *cluster* 2, *cluster* 2, frekuensi kemunculan kata yang paling banyak adalah kata “paket” dan “tidak”, didukung oleh kata ”kirim”, “*follow*”, “*up*”, “nyaman”, “nomor”, “resi” dan lainnya. Kata tersebut merupakan bentuk permintaan untuk *follow up* nomor resi dari barang yang dikirimkan.

(a) *Cluster* 1

(b) *Cluster* 2

Gambar 4.17 Visualisasi *Word Cloud* pada Hasil *Cluster K-Means* Layanan Ekspedisi J&T; **(a)** *Cluster* 1 dan **(b)** *Cluster* 2

Berikut adalah perbandingan metode DBSCAN dengan *K-Means* pada akun layanan ekspedisi J&T yang ditampilkan pada Tabel 4.22.

Tabel 4.22 Perbandingan Metode DBSCAN dan *K-Means* pada Akun Layanan Ekspedisi J&T

Metode *Clustering*	***Silhouette Coefficient***	**Jumlah *Cluster***
K-Means	0.04830	2
DBSCAN	0,99995	22

4.4. Layanan Ekspedisi Pos Indonesia

Berikut adalah struktur data yang diperoleh setelah dilakukan *text prepocessing* pada akun layanan ekspedisi Pos Indonesia yang ditampilkan pada Tabel 4.23.

Tabel 4.23 Struktur Data Layanan Ekspedisi Pos Indonesia Setelah Dilakukan *Preprocessing*

Tweet ke-	Kata						
	cek	…	kirim	…	paket	…	twitter
1	1	…	0	…	0	…	0
2	0	…	1	…	0	…	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮
3.000	0	…	3	…	2	…	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮
6.796	0	…	0	…	0	…	1

Jumlah dari frekuensi kemunculan setiap kata pada setiap *tweet* setelah dilakukan *preprocessing* yang ditampilkan pada Tabel 4.23 diketahui sebanyak 485 kata. Kata-kata tersebut yang akan dipergunakan sebagai variabel penelitian. Berikut adalah hasil perhitungan TF-IDF yang ditampilkan pada Tabel 4.24.

Tabel 4.24 Hasil Perhitungan TF-IDF Layanan Ekspedisi Pos Indonesia

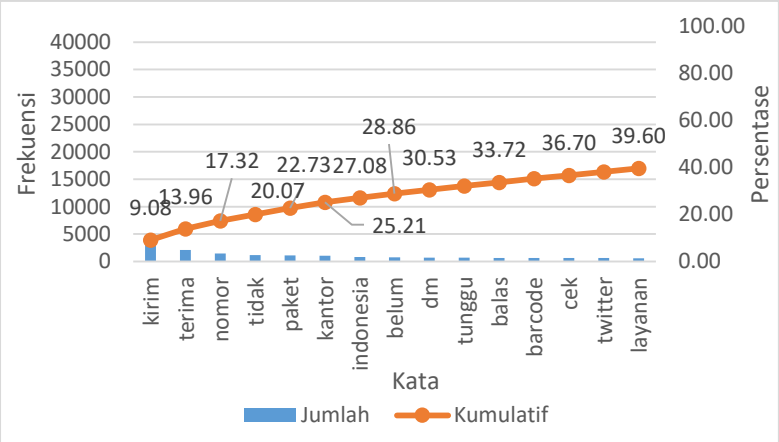
Tweet ke-	Kata						
	cek	…	kirim	…	paket	…	twitter
1	0,5526	…	0	…	0	…	0
2	0	…	0,2314	…	0	…	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮
3.000	0	…	0,6942	…	0,3573	…	0
⋮	⋮	⋱	⋮	⋱	⋮	⋱	⋮
6.796	0	…	0	…	0	…	1

Hasil perhitungan TF-IDF pada Tabel 4.24 menunjukkan bahwa kata “cek” pada *tweet* ke-1 memiliki bobot sebesar 0,5526. Hal ini menunjukkan bahwa kata “cek” pada *tweet* ke-1 memiliki frekuensi kemunculan yaitu 1 × 0,5526 = 0,5526. Begitu pula dengan perhitungan TF-IDF untuk kata-kata yang lain. Berikut visualisasi data menggunakan *word cloud* dibentuk berdasarkan bobot TF-IDF dari setiap kata pada akun layanan J&T yang ditampilkan pada Gambar 4.18.

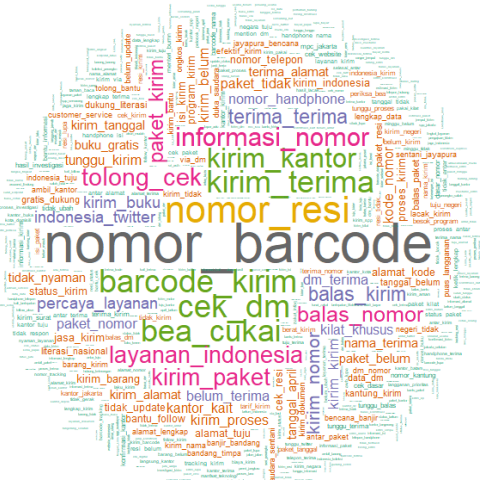
Gambar 4.18 Visualisasi *Word Cloud Unigram* pada Layanan Ekspedisi Pos Indonesia.

Berdasarkan diagram pareto pada Gambar 4.19 diketahui bahwa 30% kata yang sering muncul atau memiliki frekuensi kemunculan yang paling besar dari total frekuensi kemunculan semua kata adalah kata “kirim”, “terima”, “nomor”, “tidak”, dan “paket”. Hal ini sesuai dengan hasil visualisasi *word cloud unigram* layanan Pos Indonesia pada Gambar 4.18. Kata perlu diutamakan untuk ditindak lanjuti yaitu 30% kata yang muncul dari total frekuensi kemunculan kata. Kata “kirim” merupakan kata yang memiliki frekuensi bobot yang paling besar. Kata “kirim” biasa digunakan untuk menanyakan keadaan atau lokasi paket.

Kata tersebut tidak memiliki informasi yang lengkap, sehingga perlu informasi tambahan untuk dapat menginterpretasikan kata yang dihasilkan dari *word cloud*. Hasil visualisasi *word cloud* menggunakan teknik *bigram* yang ditampilkan pada Gambar 4.19.

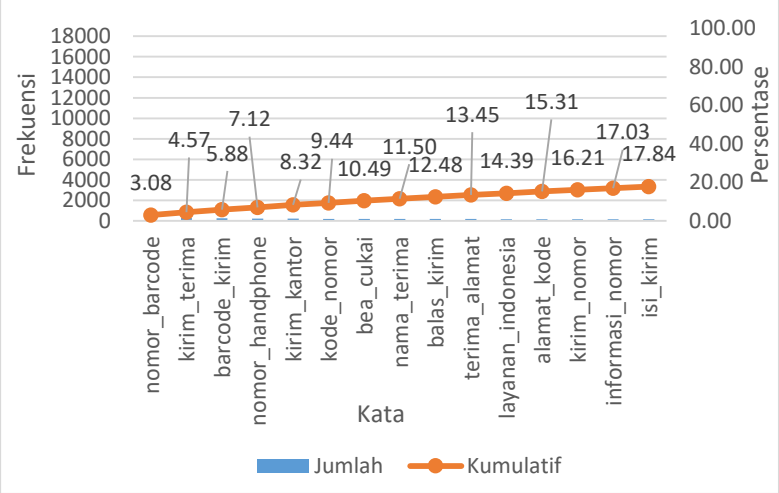


Gambar 4.19 Diagram Pareto *Unigram* Layanan Ekspedisi Pos Indonesia



Gambar 4.20 Visualisasi *Word Cloud Bigram* pada Layanan Ekspedisi Pos Indonesia.

Dari hasil diagram pareto pada Gambar 4.21 menunjukkan bahwa 10% pasangan kata dari total frekuensi kemunculan kata adalah pasangan kata “nomor_barcode”, “kirim_terima”, “barcode_kirim”, “nomor_handphone”, “kirim_kantor”, dan “kode_nomor”. Pasangan kata tersebut sesuai dengan hasil visualisasi *word cloud bigram* pada Gambar 4.20. Kata yang lebih diutamakan agar ditindak lanjuti yaitu 10% pasangan kata dari total frekuensi kemunculan kata.



Gambar 4.21 Diagram Pareto *Bigram* Layanan Ekspedisi Pos Indonesia.

4.4.1 *Clustering* Menggunakan Metode DBSCAN

Berikut adalah hasil *clustering* dengan metode DBSCAN dengan *MinPts* sebesar 30 dengan berbagai kombinasi *Eps* diperoleh nilai *silhouette coefficient* untuk layanan ekspedisi Pos Indonesia yang ditampilkan pada Tabel 4.25.

Tabel 4.25 Nilai *Silhouette Coefficient* dari Metode DBSCAN pada Akun Layanan Ekspedisi Pos Indonesia

Eps	***Silhouette Coefficient***
0,1	0,9974903

Tabel 4.25 Nilai *Silhouette Coefficient* dari Metode DBSCAN pada Akun Layanan Ekspedisi Pos Indonesia (Lanjutan)

Eps	***Silhouette Coefficient***
0,2	0,9715307
0,3	0,9406119
0,4	0,8782036
0,5	0,7608612
0,6	0,3346245
0,7	0,2394187
0,8	0,0807623

Berdasarkan Tabel 4.25, nilai *silhouette coefficient* tertinggi yang diperoleh dari metode DBSCAN pada akun layanan ekspedisi Pos Indonesia dilakukan menggunakan parameter *Eps* sebesar 0,1. Nilai *silhouette coefficient* yang dihasilkan dengan menggunakan parameter *Eps* adalah 0,9974903. Hasil *clustering* dikatakan cukup baik dikarenakan nilai *silhouette coefficient* mendekati 1. Jumlah *cluster* yang diperoleh dari *clustering* meng-gunakan metode DBSCAN dengan *MinPts* sebesar 30 dan *Eps* sebesar 0,1 adalah sebanyak 11 *cluster* dengan 5.815 *tweet* sebagai *noise*. Oleh karena itu, dengan menggunakan *MinPts* sebesar 30 diperoleh *Eps* optimum sebesar 0,1. Berikut adalah hasil *clustering* dari metode DBSCAN yang ditampilkan secara visual menggunakan *word cloud* pada akun layanan ekspedisi Pos Indonesia ditunjukkan pada Gambar 4.22.

Berdasarkan hasil clustering pada layanan ekspedisi Pos Indonesia menggunakan metode DBSCAN, didapatkan hasil yang ditampilkan pada Gambar 4.22. *Cluster* 1 berisi pasangan kata yang berkaitan tentang permintaan informasi tambahan. *Cluster* 2 berisi pasangan kata yang berkaitan tentang pengecekan informasi paket dengan *barcode*. *Cluster* 3 berisi pasangan kata yang ber-kaitan tentang permohonan menunggu. *Cluster* 4 berisi pasangan kata yang berkaitan tentang bentuk rasa syukur. *Cluster* 5 berisi pasangan kata yang berkaitan tentang filateli. *Cluster* 6 berisi pasangan kata yang berkaitan tentang balasan dari pengecekan

informasi paket berdasarkan barcode. *Cluster* 7 berisi pasangan kata yang berkaitan tentang layanan pos indoensia. *Cluster* 8 berisi pasangan kata yang berkaitan tentang informasi pengecekan pengiriman, dan *cluster* 9 berisi pasangan kata yang berkaitan tentang program literasi masyarakat.

(a) *Cluster* 1 **(b)** *Cluster* 2 **(c)** *Cluster* 3

(d) *Cluster* 4 **(e)** *Cluster* 5 **(f)** *Cluster* 6

(g) *Cluster* 7 **(h)** *Cluster* 8 **(i)** *Cluster* 9

Gambar 4.22 *Word Cloud* pada Hasil *Cluster* DBSCAN Layanan Ekspedisi Pos Indonesia 9 *Cluster*; **(a)** *Cluster* 1, **(b)** *Cluster* 2, **(c)** *Cluster* 3, **(d)** *Cluster* 4, **(e)** *Cluster* 5, **(f)** *Cluster* 6, **(g)** *Cluster* 7, **(h)** *Cluster* 8, dan **(i)** *Cluster* 9.

(j) *Cluster* 10 **(k)** *Cluster* 11

Gambar 4.23 *Word Cloud* pada Hasil *Cluster* DBSCAN Layanan Ekspedisi Pos Indonesia 2 *Cluster* Terakhir; **(j)** *Cluster* 10, dan **(k)** *Cluster* 11

Gambar 4.23 menunjukkan visualisasi *word cloud* dari hasil *clustering* menggunakan metode DBSCAN didapatkan hasil yaitu *cluster* 10 berisi pasangan kata yang berkaitan tentang bencana banjir bandang Jayapura, sedangkan *cluster* 11 berisi pasangan kata yang berkaitan tentang promosi layanan. Dari 11 *cluster* yang terbentuk, terdapat beberapa *cluster* yang hanya memiliki 1 anggota yaitu *cluster* 3, *cluster* 4, dan *cluster* 5. Hal ini disebab-kan oleh hasil *preprocessing* yang menghilangkan beberapa kata dalam suatu *tweet* dan hanya menyisakan satu kata.

4.4.2 *Clustering* Menggunakan Metode *K-Means*

Berikut adalah hasil *clustering* dengan menggunakan metode *K-Means* pada layanan ekspedisi J&T. Penentuan jumlah kelompok optimum berdasarkan dari nilai *Variance Ratio Criterion* (VRC) yang dilihat dari tingginya nilai VRC. Berikut adalah grafik dari nilai VRC dari *K* mulai dari 2 hingga 20 pada data akun layanan ekspedisi Pos Indonesia yang ditampilkan pada Tabel 4.26.

Tabel 4.26. Nilai VRC pada Layanan Ekspedisi Pos Indonesia

Cluster	VRC	*Cluster*	VRC
2	**195,03808**	5	142,98908
3	162,77934	6	146,72316
4	151,56805	7	128,89546

Tabel 4.26. Nilai VRC pada Layanan Ekspedisi Pos Indonesia (Lanjutan)

Cluster	VRC	*Cluster*	VRC
8	140,48242	15	108,86752
9	133,59514	16	110,75153
10	134,69985	17	104,81078
11	119,89227	18	103,57125
12	117,23284	19	102,15842
13	116,14811	20	94,412212
14	111,89251		

Berdasarkan hasil dari perhitungan nilai VRC pada Tabel 4.26 diketahui bahwa nilai VRC tertinggi adalah 195,0381. Sehingga dapat disimpulkan bahwa jumlah *cluster* optimum untuk metode *K-Means* adalah sebanyak 2 *cluster*. Berdasarkan penentuan jumlah kelompok yang optimum didapatkan nilai *silhouette clustering*. Berikut adalah hasil nilai *silhouette coefficient* yang ditunjukkan pada Tabel 4.27.

Tabel 4.27 Nilai *Silhouette Coefficient* dari Metode *K-Means* pada Akun Layanan Ekspedisi Pos Indonesia

Cluster	*Silhouette Coefficient*	*Cluster*	*Silhouette Coefficient*
2	**0,0107**	12	0,0598
3	0,0174	13	0,0626
4	0,0206	14	0,0642
5	0,0161	15	0,0663
6	0,0300	16	0,0749
7	0,0398	17	0,0787
8	0,0456	18	0,0761
9	0,0549	19	0,0833
10	0,0500	20	0,0724
11	0,0462		

Berdasarkan dari nilai VRC tertinggi pada *cluster* 2, didapatkan nilai *silhouette coefficient* yang dapat dilihat pada Tabel 4.27 yaitu sebesar 0,0107. Nilai *silhouette coefficient* terbilang rendah, sehingga dapat dikatakan hasil *cluster*nya tidak bagus. Berikut adalah hasil *clustering* dari metode *K-Means* yang ditampilkan secara visual menggunakan *word cloud* pada akun layanan ekspedisi Pos Indonesia ditunjukkan pada Gambar 4.13.

(a) *Cluster* 1

(b) *Cluster* 2

Gambar 4.24 Visualisasi *Word Cloud* pada Hasil *Cluster K-Means* Layanan Ekspedisi Pos Indonesia; **(a)** *Cluster* 1 dan **(b)** *Cluster* 2.

Berdasarkan hasil clustering menggunakan metode *K-Means* pada layanan ekspedisi Pos Indonesia, didapatkan hasil yang ditampilkan pada Gambar 4.24. Jumlah *cluster* yang dipilih adalah 2 *cluster*. Diketahui bahwa pada *cluster* 1, frekuensi kemunculan kata yang paling banyak muncul adalah “kirim” dan “paket”,

didukung oleh kata lain yaitu “nomor”, “layanan”, “kirim”, dan lainnya Hal ini menunjukkan keterangan mengenai permintaan pelacakan paket dari *customer* kepada pihak layanan ekspedisi Pos Indonesia, sedangkan pada *cluster* 2, frekuensi kemunculan kata yang paling banyak muncul adalah “kirim” dan “kantor”, didukung kata lainnya yaitu “terima”, “nomor”, “tunggu”, “barcode” dan lain-lain. Hal ini menunjukkan keterangan mengenai informasi penerimaan paket di kantor.

Berikut adalah perbandingan metode DBSCAN dengan *K-Means* pada akun layanan ekspedisi Pos Indonesia yang ditampilkan pada Tabel 4.28.

Tabel 4.28 Perbandingan Metode DBSCAN dan *K-Means* pada Akun Layanan Ekspedisi Pos Indonesia

Metode *Clustering*	***Silhouette Coefficient***	**Jumlah *Cluster***
K-Means	0,0107	2
DBSCAN	0,9975	11

4.5 Perbandingan Metode *Clustering* DBSCAN dan *K-Means*

Perbandingan metode *clustering* diperlukan untuk mengetahui metode yang terbaik yang dapat digunakan untuk mengelompokkan *tweet* pada akun layanan ekspedisi JNE, J&T, dan Pos Indonesia. Perbandingan metode *clustering* ini didasar-kan pada hasil nilai *silhouette coefficient* yang diperoleh dari metode *K-Means* dan DBSCAN. Berikut adalah perbandingan dari kedua metode yang digunakan yang dapat dilihat pada Tabel 4.29.

Tabel 4.29 Perbandingan Antar Metode DBSCAN dan *K-Means*

Layanan Ekspedisi	**Metode *Clustering***	***Silhouette Coefficient***	**Jumlah *Cluster***
JNE	*K-Means*	0,0819	2
	DBSCAN	0,2652	18
J&T	*K-Means*	0.0483	2
	DBSCAN	0,9999	22

Tabel 4.29 Perbandingan Antar Metode DBSCAN dan *K-Means*

Layanan Ekspedisi	**Metode *Clustering***	***Silhouette Coefficient***	**Jumlah *Cluster***
Pos Indonesia	*K-Means*	0,0107	2
	DBSCAN	0,9975	11

Berdasarkan Tabel 4.29 dapat dilihat dari nilai *silhouette coefficient* pada metode DSBCAN di semua layanan ekspedisi JNE, J&T, dan Pos Indonesia jauh lebih baik dalam mengelompokkan *tweet* dibandingkan dengan metode *K-Means*. Hal ini dapat dilihat dari nilai *silhouette coefficient* pada metode DBSCAN yang mendekati 1 yang berarti bahwa *cluster* yang terbentuk telah kompak dan terpisahkan antara satu dengan yang lain serta *tweet* yang ada telah berada dalam *cluster* yang tepat. Jumlah kelompok yang terbentuk banyak pada hasil *clustering* menggunakan metode DBSCAN dapat menguntungkan pihak layanan ekspedisi JNE, J&T, dan Pos Indonesia untuk memberikan informasi yang lebih variatif agar dapat digunakan sebagai bahan evaluasi atau monitoring kedepannya.

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan analisis dan pembahasan yang telah diuraikan pada bab sebelumnya, diperoleh kesimpulan sebagai berikut.

1. Berdasarkan hasil *crawling* data, diketahui bahwa jumlah *tweet* yang dihasilkan oleh layanan ekspedisi JNE lebih banyak dibandingkan dengan lainnya. Kata yang sering muncul pada *tweet* yang ditujukan pada layanan ekspedisi JNE adalah "cek" dan "dm". Kata yang sering muncul pada *tweet* yang ditujukan pada layanan ekspedisi J&T adalah "paket", sedangkan kata yang sering muncul pada *tweet* yang ditujukan pada layanan ekspedisi Pos Indonesia adalah "kirim".
2. Berdasarkan nilai *silhouette coefficient* tertinggi yang diperoleh, didapatkan hasil bahwa metode DBSCAN merupakan metode terbaik untuk mengelompokkan *tweet* yang ditujukan kepada akun media sosial *Twitter* layanan ekspedisi JNE, J&T, dan Pos Indonesia. *Clustering* dengan metode terbaik menghasilkan 18 *cluster* untuk layanan ekspedisi JNE, 22 *cluster* untuk layanan ekspedisi J&T, dan 11 *cluster* untuk layanan ekspedisi Pos Indonesia.

5.2 Saran

Saran yang dapat diberikan kepada layanan ekspedisi JNE, J&T, dan Pos Indonesia yaitu dapat mempertimbangkan hasil *clustering* dalam memberikan respon yang cepat kepada *tweet* pelanggan layanan ekspedisi JNE, J&T, dan Pos Indonesia serta dapat dijadikan sebagai bahan evaluasi dan monitoring. Saran untuk penelitian selanjutnya, diharapkan agar lebih teliti pada saat *preprocessing* dan dapat menggunakan *feature selection genetic algorithm*. Hal ini dikarenakan hasil *preprocessing* akan dapat mempengaruhi hasil *cluster* serta dengan adanya *feature selection* dapat memperkecil jumlah hasil *cluster* menjadi lebih spesifik.

Halaman ini sengaja dikosongkan

DAFTAR PUSTAKA

Adinugroho, S. & Sari, Y. A. (2018). *Implementasi Data Mining Menggunakan Weka.* Malang: UB Press.

Alfarisi. (2017). *Data Preprocessing - Konsep Pembelajaran Data Mining.* Diakses 9 Desember 2018 dari https://steemit.com/education/@alfarisi/data-preprocessing-konsep-pembelajaran-data-mining.

Ariadi, D. & Fithriasari, K. (2015). Klasifikasi Berita Indonesia Menggunakan Bayesian Classification dan Support Vector Machine dengan Confix Stripping Stemmer. *Jurnal Sains dan Seni ITS*, Vol. 4, No. 2.

Arsih, N. (2016). Metode Pengclusteran Berbasis Densitas Menggunakan Algoritma DBSCAN. *Prosiding Statistika*, Vol. 2, Hal. 153-163.

Bernard, H. R., Wutich, A. & Ryan, G. W. (2017). *Analyzing Qualitative Data : Systematic Approaches.* Singapura: Sage Publications, Inc.

Bing, L. (2010). *Handbook of Natural Language Processing.* Boca Raton: CRC Press.

Budiman, S. A. D. (2016). Perbandingan Metode K-MEANS dan Metode DBSCAN pada Pengelompokkan Rumah Kost Mahasiswa di Kelurahan Tembalang Semarang. *Jurnal Gaussian*, Vol. 5, Hal. 757-762.

Buntoro, G. A., Adji, T. B., & Purnamasari, A. E. (2014). Sentiment Analysis Twitter dengan Kombinasi Lexicon Based dan Double Propagation. *Conference on Information Technology and Electrical Engineering* (CITEE), Hal. 39-43.

Calinski, T. & Harabasz, J. (1974). *A Dendritc Method for Cluster Analysis*. *Communications in Statistics*, Vol. 3, Hal. 1-27.

Devi, N. M. A. S. (2015). Implementasi Metode Clustering DBSCAN pada Proses Pengambilan Keputusan. *Jurnal Ilmiah Lontar Komputer*, Vol. 6, Hal. 665-661.

Dragut, E., Fang, F., Sistla, P., & Yu, C. (2009). *Stop Word and Related Problems in Web Interface Integration.* Chicago: University of Illinois.

Durajati, C., & Gumelar, A. B. (2012). Pemanfaatan Teknik Supervised untuk Klasifikasi Teks Bahasa Indonesia. *Journal Link* Vol 16/No. 1(ISSN 1858 - 4667), Hal 1-8.

Feldman, R., & Sanger, J. (2007). *Text Mining Handbook : Advanced Approaches in Analyzing Unstructured Data.* New York: Cambridge University Press.

Hadijah, S. (2016). *Perbedaan Jasa Pengiriman.* Diakses 21 Januari 2019 dari https://www.cermati.com/artikel/perbeda-an-jasa-pengiriman-pos-indonesia-jne-tiki-fedex.

Han, J., Kamber, M., & Pei, J. (2012). *Data Mining : Concepts and Technique*. USA: Elsevier.

Irwansyah, E., & Faisal, M. (2015). *Advanced Clustering : Teori dan Aplikasi.* Yogyakarta: DeePublish.

J&T. (2018). *About Us.* Diakses 26 Februari 2019 dari http://www.jet.co.id/about.

JNE. (2015). *Profil Perusahaan JNE Express.* Diakses 27 Februari 2019 dari https://www.jne.co.id/id/perusahaan/profil-peru-sahaan/visi-dan-misi.

Jo, T. (2018). *Text Mining : Concepts, Implementation, and Big Data Chalenge.* Seoul: Springer Internasional Publishing.

Kogan, J. (2007). *Introduction to Clustering Large and High-Dimensional Data.* New York: Cambrige University Press.

Kwartler, T. (2017). *Text Mining in Practice with R.* United States: John Wiley & Sons.

Li, Y., & Wu, H. (2012). *A Clustering Method Based on K-Means Algorithm. International Conference on Solid State Devices and Materials Science,* Hal. 1104-1109.

Manning, C. D., Raghavan, P., & Schutze, H. (2008). *Introduction to Information Retrieval.* Cambridge: Cambridge University Press.

Miner, G., Elder, J., Fast, A., Hill, T., Nisbet, R., & Delen, D, (2012). *Practical Text Mining and Statistical Analysis for*

Non-structured Text Data Applications. Waltham: Academic Press.

Pos-Indonesia. (2017). Diakses 26 Februari 2019 dari http://www.posindonesia.co.id/index.php/pos-express.

Rapi. (2017). *Mengenal Sedikit Mengenai Perusahaan Jasa Pengiriman Barang.* Diakses 21 Januari 2019 dari https://rapi.co.id/mengenal-sedikit-mengenai-perusahaan-jasa-pengiriman-barang.

Silge, J., & Robinson, D., (2017). *Text Mining With R.* United States: O"Reilly Media, Inc.

Siregar, A. M., & Puspabhuana, A., (2002). *Data Mining : Pengolahan Data Menjadi Informasi dengan RapidMiner.* Sukoharjo: CV Kekata Group.

Tamaela, J. (2017). *Cluster Analysis* Menggunakan Algoritma *Fuzzy C-Means* dan *K-Means* untuk Klasterisasi dan Pemetaan Lahan Pertanian di Minahasa Tenggara. *Jurnal Buana Informatika,* Vol. 8, Hal. 151-160.

Tan, H. P., Plowman, D., & Hancock, P. (2007). *Intellectual Capital and Financial Returns of Companies. Journal of Intellectual Capital,* Vol. 8, Hal 76-91.

Thomas, S. T., & Harode, U. (2015). A *Comparative Study on K-Means and Hierarchical Clustering. International Journal of Electronics, Electrical and Computational System (IJEECS),* Vol. 4, Hal. 5-10.

Weiss, S. M., Indurkhya, N., Zhang, T., & Damerau, F. J. (2005). *Text Mining Predictive Methods for Analyzing Unstructures Information.* New York: Spinger Science Business Media.Inc.

Ye, Q., Gao, W., & Zeng, W. (2003). *Color Image Segmentation Using Density-Based Clustering. International Conference on Multimedia and Expo* (*ICME*) Vol. 3, Hal. III-346.

Halaman ini sengaja dikosongkan

LAMPIRAN

Lampiran 1. *Syntax Crawling* Data Menggunakan R

```
library(twitteR)
library(rtweet)
library(tidyverse)
library(tm)

create_token (
app                 = "my_twitter_research_app",
consumer_key        = 'xNkNrYjOVqpenmDZO2uB9gE5K',
consumer_secret     = 'TGfPUpFq2PBQVRApbgSqfXK0twkCjoGlaWT
                      ssM8JlKoNXGFvoI',
access_token        = '1061919402277433345-IQ6HyGoSk3kFg7WU1
                      mCn4daeswlaws',
access_secret       ='k4VFAgAiN6fFHrXEUmwzREZvpZGubW4ySv
                      W1nqXJlpGfE')

JNE        <- search_tweets ("JNECare", lang="id", n=15000,
                tweet_mode="extended", include_rts=FALSE)
JNECRAW <-  data.frame(JNE$created_at, JNE$screen_name,
                JNE$text, JNE$display_text_width,
                as.character(JNE$mentions_screen_name))

JNT        <-  search_tweets("jntexpressid", lang="id", n=15000,
                tweet_mode="extended", include_rts=FALSE)
JNTCRAW <-  data.frame(JNT$created_at, JNT$screen_name,
                JNT$text, JNT$display_text_width,
                as.character(JNT$mentions_screen_name))

POS        <- search_tweets("PosIndonesia", lang="id", n=15000,
                tweet_mode="extended", include_rts=FALSE)
POSCRAW <- data.frame(POS$created_at, POS$screen_name,
                POS$text, POS$display_text_width,
                as.character(POS$mentions_screen_name))
```

Lampiran 2. *Syntax* Karakteristik Data Menggunakan Python

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

frequencies = [62188, 25094, 11573]
freq_series = pd.Series(frequencies)
x_labels = ['JNECare', 'Jntexpressid', 'PosIndonesia']

plt.figure(figsize=(12, 8))
ax = freq_series.plot(kind='bar')
ax.set_title('Amount Frequency')
ax.set_xlabel('Amount ($)')
ax.set_ylabel('Frequency')
ax.set_xticklabels(x_labels)

def add_value_labels(ax, spacing=5):
    for rect in ax.patches:
        y_value = rect.get_height()
        x_value = rect.get_x() + rect.get_width() / 2
        space = spacing
        va = 'bottom'
        va = 'top'
        label = "{:.1f}".format(y_value)

        ax.annotate(
            label,
            (x_value, y_value),
            xytext=(0, space),
            textcoords="offset points",
            ha='center',
            va=va)
add_value_labels(ax)
plt.savefig("jumlah tweet1.png")
```

Lampiran 3. *Syntax* Visualisasi *Word Cloud* Menggunakan R

```
library(RColorBrewer)
library(wordcloud)
library(tm)

dataTFIDF = read.csv("TFIDFP.csv", header = TRUE)
jumlahdata = rowSums(dataTFIDF)
gabTFIDF = data.frame(dataTFIDF, jumlahdata)
TFIDF = gabTFIDF[which(gabTFIDF$jumlah!=0),-1548]

a = sort(colSums(TFIDF), decreasing = TRUE)
b = data.frame(word = names(a), freq=a)
win.graph()
wordcloud(words = b$word, freq=b$freq, scale = c(4,0,4),
random_color = FALSE, random.order = FALSE, rot.per = 0.25,
min_freq = 3, max.words = Inf, colors = brewer.pal(8, "Dark2"))
```

Lampiran 4. *Syntax Text Preprocessing* Menggunakan Python

```
import pandas as pd
import re
import nltk
import sys
import string
from nltk.tokenize import word_tokenize
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer

data = pd.read_excel("D://DATA FIX.xlsx")
text = data['TWEET FULL']

#Menghapus link
dataclearlink = []
for line in text:
  result = re.sub(r"http\S+"," ", line)
  dataclearlink.append(result)
  print(result)

#Menghapus Hastag
dataclearhastag = []
for line in dataclearlink :
  result = re.sub(r"#\S+", " ", line)
  dataclearhastag.append(result)
  print(result)

#Menghapus Simbol Retweet (RT)
dataclearrt = []
for line in dataclearhastag:
  result = re.sub(r"RT", " ", line)
  dataclearrt.append(result)
  print(result)
```

Lampiran 4. *Syntax Text Preprocessing* Menggunakan Python (Lanjutan)

```
#Menghapus Username
dataclearusername = []
for line in dataclearrt:
  result = re.sub(r"@\S+", " ", line)
  dataclearusername.append(result)
  print(result)

#Menghapus angka
dataclearangka = []
for line in dataclearusername :
  result = re.sub("\d", " ", line)
  dataclearangka.append(result)
  print(result)

#Menghapus emoticon
dataclearemoticon = []
for line in dataclearangka :
  result = re.sub(r'<.*?>', " ", line)
  dataclearemoticon.append(result)
  print(result)

#Menghapus baris baru
dataclearline = []
for line in dataclearemoticon :
  result = re.sub("\n", " ", line)
  dataclearline.append(result)
  print(result)
```

Lampiran 4. *Syntax Text Preprocessing* Menggunakan Python (Lanjutan)

```
#Menghapus punctuation
dataclearpunctuation = []
for line in dataclearline :
  result = re.sub(r"[^\w\s]", " ", line)
  dataclearpunctuation.append(result)
  print(result)

#Menghapus spasi berlebih
datacleardoublespace = []
for line in dataclearpunctuation :
  result=re.sub(r'\s+', ' ', line)
  datacleardoublespace.append(result)
  print(result)

#Menghapus Underscore
dataclearunderscore = []
for line in datacleardoublespace :
  result = re.sub(r"\S+_", " ", line)
  dataclearunderscore.append(result)
  print(result)

#Case Folding
datalower = []
for line in dataclearunderscore:
  result = line.lower()
  datalower.append(result)
  print(result)
```

Lampiran 4. *Syntax Text Preprocessing* Menggunakan Python (Lanjutan)

```
#Stemming
factory = StemmerFactory()
stemmer = factory.create_stemmer()
data_stemmed = map(lambda x: stemmer.stem(x), datalower)
datastemmed = list(data_stemmed)

#Replace Kata
kata ={ "lampir":"lampiran", "ttp":"tetap", "milu":"pemilu",
"langgan":"langganan", "layan":"layanan", "lapor":"laporan",
"duit":"uang", "perhati":"perhatian", "pket":"paket", "trims":
"terima kasih", "tq":"terima kasih", "simpen":"status", "thx":
"terima kasih", "stts":"status", "tlpon":"telepon", "sttus":"status"}

from collections import OrderedDict
def replace_all(text, dic):
    for i,j in dic.items():
        text = text.replace(i,j)
    return text
dic = OrderedDict(kata)

datachange = []
for line in datastemmed:
    result = replace_all(line, dic)
    datachange.append(result)
    print(result)

#Stopword dan tokenizing
stopwords=open('D://stopword.txt', 'r').read()
datafinal=[]
for line in datachange:
   word_token=nltk.word_tokenize(line)
   word_token=[word for word in word_token if not word in
stopwords and not word[0].isdigit()]
   datafinal.append(" ".join(word_token))
```

Lampiran 4. *Syntax Text Preprocessing* Menggunakan Python (Lanjutan)

```
#Count Vectorizer
vectorizer = CountVectorizer(min_df=10)
JNT = vectorizer.fit_transform(datafinal)
JNT_ = pd.DataFrame(JNT.toarray(),
                columns=vectorizer.get_feature_names())
JNT_.to_csv("JNTCVCLEAR1.csv")

#TF-IDF
vectorizer = TfidfVectorizer(min_df=10)
TFIDF = vectorizer.fit_transform(datafinal)
TFIDFJNT=pd.DataFrame(TFIDF.toarray(),columns=vectorizer.
get_feature_names())
TFIDFJNT.to_csv('TFIDFJNTFIXCLEAR1.csv')

#Count Vectorizer BIGRAM
vectorizer = CountVectorizer(min_df=10, ngram_range=(2,2))
JNT_BG = vectorizer.fit_transform(datafinal)
term = vectorizer.get_feature_names()
term1 = [word.replace(',','_') for word in term]
JNT_BG_=pd.DataFrame(JNT_BG.toarray(), columns=term1)
JNT_BG_.to_csv("JNT_BGCLEAR1.csv")

#TF-IDF BIGRAM
vectorizer = TfidfVectorizer(min_df=10, ngram_range=(2,2))
TFIDF_BG = vectorizer.fit_transform(datafinal)
term = vectorizer.get_feature_names()
term1 = [word.replace(',','_') for word in term]
TFIDF_BG_ = pd.DataFrame(TFIDF_BG.toarray(),
                 columns = term1)
TFIDF_BG_.to_csv("TFIDFJNT_BGCLEAR.csv")
```

Lampiran 5. *Syntax Text Clustering* dengan Metode DBSCAN Menggunakan R

```
library(cluster)
library(dbscan)
library(fpc)

data_CountVect = read.csv("JNTCV.csv")
data_TFIDF=read.csv("TFIDFJNT.csv")
jumlah = rowSums(data_CountVect)
gabunganCV = data.frame(data_CountVect, jumlah)
CountVect = gabunganCV[which(gabunganCV$jumlah!=0),-668]
gabunganTFIDF = data.frame(data_TFIDF, jumlah)
TFIDF = gabunganTFIDF[which(gabunganTFIDF$jumlah!=0),-
        668]


#DBSCAN
DBSCAN = dbscan::dbscan(TFIDF, eps = 0.2, minPts = 20)
DBSCAN
x <- data.frame(TFIDF, DBSCAN$cluster)
y <- x[which(x$DBSCAN.cluster!=0),]
n = ncol(y);n
silcoef <- silhouette(y$DBSCAN.cluster, dist(y[,-668]))
silcoef
sil_coef_0.1 <- summary(silcoef)$avg.width
sil_coef_0.1

#Word Cloud Every Cluster
library(wordcloud)
library(RColorBrewer)

dtm = data.frame(CountVect, DBSCAN$cluster)

for (i in 2:46)
{
```

```
  a = dtm[which(dtm$DBSCAN.cluster==i),-668]
  b = sort(colSums(a), decreasing = TRUE)
  c = data.frame(word=names(b), freq=b)
  win.graph()
  wordcloud(words = c$word, freq=c$freq, scale = c(4,0,4),
random_color = FALSE,
          random.order = FALSE, rot.per = 0.35, min_freq = 1,
max.words = Inf,
          colors = brewer.pal(12, "Paired"))
}
```

Lampiran 6. *Syntax Text Clustering* dengan Metode *K-Means* Menggunakan Python

```
from sklearn import metrics
from sklearn.metrics import pairwise_distances
from sklearn import datasets
import pandas as pd

X = pd.read_csv("C://TFIDFPOSFIXNEW.csv")

#Nilai CHI
import numpy as np
from sklearn.cluster import KMeans
kmeans_model = KMeans(n_clusters=3, random_state=1).fit(X)
labels = kmeans_model.labels_
metrics.calinski_harabaz_score(X, labels)
for k in range(2, 21):
    kmeans_model = KMeans(n_clusters=k,
random_state=1).fit(X)
    labels = kmeans_model.labels_
    print(k, metrics.calinski_harabaz_score(X, labels))

#Nilai Silhouette
from sklearn.cluster import KMeans
from sklearn import metrics
from sklearn.metrics import silhouette_score
from sklearn.metrics import pairwise_distances

kmeans_model = KMeans(n_clusters=3, random_state=1).fit(X)
labels = kmeans_model.labels_

for k in range(2,21):
    kmeans = KMeans(n_clusters = k, random_state=1).fit(X)
    label = kmeans.labels_
    sil_coeff = silhouette_score(X, label, metric='euclidean')
    print(k, sil_coeff)
```

```
# Word Cloud Every Cluster
import csv
input_df=pd.DataFrame(data=X)
matrix = pd.read_csv("C:/USERS/ASUS/DOWNLOADS/DATA
BARU_1/DATAJNEFIX1.csv")

from sklearn.externals import joblib
num_clusters = 2
km = KMeans(n_clusters = num_clusters, random_state=1)
%time km.fit(matrix)
clusters = km.labels_.tolist()
y_kmeans = km.predict(matrix)
input_df['klaster']=pd.Series(y_kmeans, index=input_df.index)

from sklearn.externals import joblib
joblib.dump(km, 'doc_cluster.pkl')
km = joblib.load('doc_cluster.pkl')
clusters = km.labels_.tolist()
clusters_df = pd.DataFrame(clusters)
clusters_df['klaster'] = pd.DataFrame(clusters)
clusters_df['klaster'].value_counts()

import csv
train_final=pd.read_csv("C:/USERS/ASUS/Downloads/DATAJN
TFIX.csv")
train_final_df.columns = ['text']
train_final_df['klaster']=pd.Series(y_kmeans,
index=input_df.index)
devi = pd.DataFrame(train_final)
putri = pd.DataFrame(train_final_df['klaster'], columns=['class'])
gabungan = pd.concat([devi, putri], axis=1)
print(train_final)

cluster1 = gabungan[gabungan['class'] == 1 ]
```

```
cluster1 = cluster1['text']
cluster2 = gabungan[gabungan['class'] == 0 ]
cluster2 = cluster2['text']

a = []
b = []
for l in b:
    a+= l
final={v: a.count(v) for v in set(a)}

c = []
d = []
for l in d:
    c+= l
final={v: c.count(v) for v in set(c)}

datastr1 = str(cluster1)
datagab1 = re.sub(r"","", datastr1)
datastr2 = str(cluster2)
datagab2 = re.sub(r"","", datastr2)

import numpy as np
import matplotlib as mpl
import matplotlib.pyplot as plt
get_ipython().run_line_magic('matplotlib', 'inline')
from subprocess import check_output
from wordcloud import WordCloud, STOPWORDS
mpl.rcParams['font.size']=12 #10
mpl.rcParams['savefig.dpi']=100 #72
mpl.rcParams['figure.subplot.bottom']=.1
stopwords=open('D://stopword.txt', 'r').read()

#Word Cloud Cluster 1
wordcloud = WordCloud(collocations = False,
                      background_color='white',
```

```
                stopwords=stopwords,
                max_words=50,
                max_font_size=500,
                random_state=42
                ).generate(datagab1)
print(wordcloud)
fig = plt.figure(1)
plt.imshow(wordcloud)
plt.axis('off')
plt.show()
fig.savefig("D:/Cluster 1 JNE.png", dpi=900)

# Word Cloud Cluster 2
wordcloud = WordCloud(collocations = False,
                background_color='white',
                stopwords=stopwords,
                max_words=50,
                max_font_size=200,
                random_state=42
                ).generate(datagab2)
print(wordcloud)
fig = plt.figure(1)
plt.imshow(wordcloud)
plt.axis('off')
plt.show()
fig.savefig("D:/Cluster 2 JNE.png", dpi=900)
```

Lampiran 7. Surat Keterangan Data

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini, mahasiswa Departemen Statistika FMKSD ITS:

Nama : Devi Putri Isnarwaty

NRP : 06211745000032

menyatakan bahwa data yang digunakan dalam Tugas Akhir / ~~Thesis~~ ini merupakan data sekunder yang diambil dari ~~penelitian~~ / ~~Buku~~ / ~~Tugas Akhir~~ / ~~Thesis~~ / publikasi lainnya yaitu:

Sumber : Twitter API (*Application Program Interface*)

Keterangan : Data *tweet* dengan *keywords* "JNECare", "jntexpressid", dan "Pos Indonesia"

Surat Pernyataan ini dibuat dengan sebenarnya. Apabila terdapat pemalsuan data maka saya siap menerima sanksi sesuai aturan yang berlaku.

Mengetahui
Pembimbing Tugas Akhir

(Dr. Irhamah, S.Si, M.Si)
NIP. 19780406 200112 2 002

Surabaya, 14 Juni 2019

(Devi Putri Isnarwaty)
NRP. 06211745000032

*(coret yang tidak perlu)

Halaman ini sengaja dikosongkan

BIODATA PENULIS

Penulis bernama Devi Putri Isnarwaty, dilahirkan di Surabaya, 30 Desember 1995. Penulisa adalah anak kedua dari dua bersaudara oleh pasangan Narto dan Suliswaty. Motto hidup penulis adalah jangan pernah takut bermimpi. Pendidikan yang telah diselesaikan penulis adalah SDN Kebraon IV Surabaya, SMP Negeri 16 Surabaya, dan SMA Negeri 22 Surabaya. Setelah lulus dari SMA, penulis diterima di Program Studi Diploma III Departemen Statistika Bisnis Fakultas Vokasi Institut Teknologi Sepuluh Nopember Surabaya pada tahun 2014 dan lulus pada tahun 2017. Selama perkuliahan, penulis pernah aktif dalam beberapa organisasi antara lain sebagai Staff Departemen Riset dan Teknologi (RISTEK) HIMADATA-ITS periode 2015/2016 dan sebagai Ketua Departemen Keilmiahan dan Keprofesian (IMPROF) HIMADATA-ITS 2016/2017. Selain itu, penulis juga aktif mengikuti kepanitian seperti GERIGI ITS, INTERVAL ITS, dan Pekan Raya Statistika ITS 2016. Penulis melanjutkan kuliah di Program Studi Lintas Jalur (LJ) - S1 di Departemen Statistika Fakultas Matematika, Komputasi, dan Sains Data, Institut Teknologi Sepuluh Nopember Surabaya. Penulis mendapatkan kesempatan Kerja Praktek di PT. Askrindo (Persero) Distribusi Jawa Timur. Segala kritik dan saran akan diterima penulis untuk perbaikan kedepannya. Jika ada keperluan berdiskusi dengan penulis dapat melalui email devisnarwaty@gmail.com.

Halaman ini sengaja dikosongkan