



TUGAS AKHIR - KS141501

**RANCANG BANGUN APLIKASI UNTUK
KATEGORISASI KOMENTAR NETIZEN PADA MEDIA
SOSIAL PEMERINTAH DAERAH TERHADAP SKPD
BERDASARKAN FREKUENSI KATA KUNCI SKPD**

*DEVELOPING AN APPLICATION FOR NETIZEN
COMMENTS CATEGORIZATION ON SOCIAL MEDIA OF
INDONESIA LOCAL GOVERNMENTS TO LOCAL
GOVERNMENTS AGENCY BASED ON LOCAL
GOVERNMENT AGENCY KEYWORD FREQUENCY*

NODY RISKY PRATOMO
NRP 0521 14 4000 0166

Dosen Pembimbing
Nur Aini Rakhmawati, S.Kom, M.Sc.Eng, Ph.D

Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018

Halaman ini sengaja dikosongkan



TUGAS AKHIR - KS141501

RANCANG BANGUN APLIKASI UNTUK KATEGORISASI KOMENTAR NETIZEN PADA MEDIA SOSIAL PEMERINTAH DAERAH TERHADAP SKPD BERDASARKAN FREKUENSI KATA KUNCI SKPD

NODY RISKY PRATOMO

NRP 05211440000166

Dosen Pembimbing
Nur Aini Rakhmawati, S.Kom, M.Sc.Eng, Ph.D

Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi

Institut Teknologi Sepuluh Nopember
Surabaya 2018



Halaman ini sengaja dikosongkan



TUGAS AKHIR - KS141501

DEVELOPING AN APPLICATION FOR NETIZEN COMMENTS CATEGORIZATION ON SOCIAL MEDIA OF INDONESIA LOCAL GOVERNMENTS TO LOCAL GOVERNMENT AGENCY BASED ON LOCAL GOVERNMENT AGENCY KEYWORD FREQUENCY

NODY RISKY PRATOMO
NRP 0521144000166

Supervisor

Nur Aini Rakhmawati, S.Kom, M.Sc.Eng, Ph.D



Departement of Information System
Faculty of Information Technology and Communication
Institut Teknologi Sepuluh Nopember
Surabaya 2018



Halaman ini sengaja dikosongkan

LEMBAR PENGESAHAN

RANCANG BANGUN APLIKASI UNTUK KATEGORISASI KOMENTAR NETIZEN PADA MEDIA SOSIAL PEMERINTAH DAERAH TERHADAP SKPD BERDASARKAN FREKUENSI KATA KUNCI SKPD

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Departemen Sistem Informasi
Fakultas Teknologi Informasi dan Komunikasi
Institut Teknologi Sepuluh Nopember

Oleh:

NODY RISKY PRATOMO

NRP. 0521 14 4000 0166

Surabaya, Juni 2018

**KEPALA
DEPARTEMEN SISTEM INFORMASI**

Dr. Ir. Aris Tjahyanto, M.Kom,
NIP 19650310 199102 1 001

Halaman ini sengaja dikosongkan

LEMBAR PERSETUJUAN

RANCANG BANGUN APLIKASI UNTUK KATEGORISASI KOMENTAR NETIZEN PADA MEDIA SOSIAL PEMERINTAH DAERAH TERHADAP SKPD BERDASARKAN FREKUENSI KATA KUNCI SKPD

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Departemen Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

NODY RISKY PRATOMO

NRP. 0521 14 4000 0166

Disetujui Tim Penguji: Tanggal Ujian: 4 Juni 2018

Periode Wisuda: September 2018

Nur Aini Rakhmawati, S.Kom, M.Sc.Eng, Ph.D

(Pembimbing I)

Faizal Johan Atletiko, S.Kom., M.T

(Penguji I)

Renny Pradina, S.T, M.T

(Penguji II)

X

Halaman ini sengaja dikosongkan

RANCANG BANGUN APLIKASI UNTUK KATEGORISASI KOMENTAR NETIZEN PADA MEDIA SOSIAL PEMERINTAH DAERAH TERHADAP SKPD BERDASARKAN FREKUENSI KATA KUNCI SKPD

Nama Mahasiswa : Nody Risky Pratomo
NRP : 05211440000166
Departemen : Sistem Informasi FTIK-ITS
Dosen Pembimbing : Nur Aini Rakhmawati, S.Kom,
M.Sc.Eng, Ph.D

ABSTRAK

Menurut survei internetworkstats yang dirilis pada Januari 2018, jumlah pengguna internet di Indonesia telah mencapai 132,7 juta jiwa. Dari jumlah tersebut, 40% diantaranya merupakan pengguna aktif media sosial. Hal ini mengakibatkan peningkatan penerapan konsep e-government pada pemerintah. Melalui sosial media, masyarakat dapat melakukan interaksi kepada akun sosial media pemerintah kota dalam memberikan informasi atau kritik dan saran terkait kotanya. Dengan mengetahui informasi atau permasalahan yang terjadi di suatu kota, pemerintah kota dapat melakukan aksi sesuai dengan SKPD terkait sesuai dengan tugas dan fungsinya masing-masing.

Melalui fenomena tersebut, diperlukanlah suatu aplikasi yang bertujuan untuk melakukan kategorisasi komentar masyarakat di sosial media berdasarkan SKPD di masing-masing kota. Proses kategorisasi dilakukan dengan melihat kesesuaian kata kunci SKPD dengan komentar pada sosial media Facebook, Twitter, dan Youtube akun resmi pemerintah kota dan influencer. Apabila komentar memiliki kata kunci SKPD maka komentar akan dikategorikan sesuai SKPD, apabila komentar memiliki lebih dari satu kata kunci, maka komentar tersebut dikategorikan sebagai “duplicate”, dan apabila komentar tidak

memiliki kata kunci maka akan dikategorikan sebagai “uncategorized”. Selain itu, aplikasi ini dapat memberikan rekomendasi kata kunci berdasarkan komentar yang masuk. Proses rekomendasi dilakukan dengan cara menghitung nilai IDF pada kata dalam komentar yang telah dikategorikan sesuai SKPD. Rekomendasi kata kunci yang diambil adalah kata kunci yang memiliki nilai IDF paling rendah.

Hasil pengujian untuk kota Depok menunjukkan bahwa dari 4.325 komentar yang telah dilakukan preprocessing, 10,35% komentar berhasil dikategorikan sesuai SKPD. Jumlah komentar yang terkategori SKPD dipengaruhi oleh pemilihan kata kunci SKPD. Pengujian rekomendasi kata turut dilakukan dengan mengambil dinas dalam pemerintahan kota yang sama. Dari 19 dinas, terdapat 3 dinas yang mendapatkan rekomendasi kata. Hal ini disebabkan karena 3 dinas tersebut memiliki jumlah komentar yang lebih besar dari rata-rata komentar yang telah terkategorisasi. Dari 3 dinas, 1 dinas memiliki kata kunci yang relevan dan 2 sisanya tidak relevan dengan SKPD. Hal ini disebabkan tidak tepatnya kata kunci SKPD untuk melakukan kategorisasi serta jumlah komentar terkategorisasi yang masih sedikit.

Kata Kunci: egovernment, sosial media, kategorisasi, regular expression

DEVELOPING AN APPLICATION FOR NETIZEN COMMENTS CATEGORIZATION ON SOCIAL MEDIA OF INDONESIA LOCAL GOVERNMENTS TO LOCAL GOVERNMENT AGENCY BASED ON LOCAL GOVERNMENT AGENCY KEYWORD FREQUENCY

Nama Mahasiswa : Nody Risky Pratomo

NRP : 05211440000166

Departemen : Sistem Informasi FTIK-ITS

Dosen Pembimbing :

1. Nur Aini Rakhmawati, S.Kom, M.Sc.Eng, Ph.D

ABSTRACT

According to a survey conducted by internetworldstats in 2018, the number of internet users in Indonesia has reached 132,7 million people. From that number, 40% of them are active users of social media. This resulted in improvement of e-government concept on government. Through social media, people could interact with social media account in government in order to give information or criticism and suggestion to their city. With knowing information or issues in a city, the government could react through their local government agency related to their tasks and functions.

Through the phenomenon, there is a needed an application that aims to categorize public comment on social media based on local government agency in each city in Indonesia. Categorization process is based on a keyword that related to local government agency in each city. Categorization process is performed by looking at the suitability of words in local government agency to comments on the official account and influencer account of Facebook, Twitter, and Youtube. If the

comment has a keyword from local government agency then it will be categorized as local government agency itself, if the comment has more than one keyword then it will be categorized as “duplicate”, and if the comment does not have keyword then it will be categorized as “uncategorized”. Furthermore, the application could give the recommendation of keyword based on the comments which have been categorized based on local government agency. Recommendation process is performed by calculating IDF value in each word on comments that have been categorized. Keyword recommendations taken are keywords that have the lowest IDF value

To test the categorization process, Depok city is being used. The test result shows that from 4.325 comments which have been preprocessing, 10,35% of them successfully categorized. The number of categorized comments based on local government agency is influenced by keyword selection. Testing keyword recommendation is also done by taking local government agency in the same city government. From 19 local government agency, there are 3 of them that get keyword recommendation. It is caused by the number of comments among 3 local government agency is more than the average number of comments that have categorized. From 3 local government agency, one of them has a keyword that is relevant for local government agency itself, meanwhile, the rest of them are not relevant. This happens because of the suitability of keyword in local government agency for categorization process and the number of categorized comments.

Keywords: egovernment, social media, categorization, regular expression

KATA PENGANTAR

Segala puji dan syukur pada Allah SWT yang telah melimpahkan rahmat dan anugerah-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul “Rancang Bangun Aplikasi untuk Kategorisasi Komentar Netizen pada Media Sosial Pemerintah Daerah terhadap SKPD berdasarkan Frekuensi Kata Kunci SKPD” sebagai salah satu syarat kelulusan pada Departemen Sistem Informasi, Fakultas Teknologi Informatika dan Komunikasi, Institut Teknologi Sepuluh Nopember Surabaya. Dalam penyusunan tugas akhir, penulis mendapatkan dukungan dari berbagai pihak dalam bentuk doa, motivasi, semangat, kritik, saran, dan bantuan lainnya. Untuk itu secara khusus penulis menyampaikan ucapan terima kasih kepada:

1. Allah SWT yang telah memberikan rejeki dan kemudahan bagi penulis dalam mengerjakan Tugas Akhir ini.
2. Keluarga penulis, terutama kedua orang tua, Heru Dwi Atmoko (Alm) dan Eka Tjandrawinastuti, serta adik penulis, Arya Dwi Pamungkas, yang selalu senantiasa membantu, mendoakan, dan memotivasi penulis.
3. Bapak Dr. Ir. Aris Tjahyanto, M.Kom., selaku Ketua Departemen Sistem Informasi ITS, Bapak Nisfu Asrul Sani, S.Kom, M.Sc selaku ketua prodi S1 Sistem Informasi ITS serta seluruh dosen pengajar beserta staff dan karyawan di Departemen Sistem Informasi FTIK ITS Surabaya
4. Ibu Nur Aini Rakhmawati, S.Kom., M.Sc., Eng. Ph.D selaku dosen pembimbing yang telah meluangkan waktu luan untuk membimbing, mengarahkan, dan mendukung dalam pemberian ilmu, petunjuk, dan motivasi dalam menyelesaikan Tugas Akhir ini.

5. Bapak Faizal Johan Atletika, S.Kom., M.T, dan Ibu Renny Pradina, S.T., M.T selaku dosen pengaji yang telah memberikan kritik, saran, dan masukan dalam penyelesaian Tugas Akhir ini.
6. Anak-anak bimbingan Bu Iin yaitu Dito, Fikry, Guntur, Hendro, Bang In, dan Berli serta keluarga UKM Cinta Lab sebagai teman yang selalu menemani penulis dan meramaikan lab dalam mengerjakan tugas akhir ini. Terima kasih atas saran, masukkan, dan motivasi yang diberikan.
7. Keluarga YSC yaitu Depe, Cindy, Elroy, Ninda, Nita, Fia, Patty, Rachel, Rara, Risha, Opor, Septy, Dhira, Tania, dan Yunis yang telah menjadi teman yang baik selama masa perkuliahan serta dukungan yang telah diberikan bagi penulis dalam masa pengerjaan tugas akhir.
8. Keluarga Warung Squad yaitu Adhen, Adit, Alden, Bram, Dito, Fachrur, Fadel, Fata, Leon, Obik, Rafi, Rysma, Satria dan Boy yang telah menjadi teman bermain dan teman kontrak selama akhir perkuliahan. Terima kasih atas semua dukungan yang diberikan.
9. Sahabat penulis, Muhamad Danang Priambodo, yang juga sedang mengerjakan tugas akhir, terima kasih telah menjadi sahabat yang baik dan memotivasi penulis selama pengerjaan tugas akhir.
10. Serta seluruh teman-teman penulis yang tidak bisa disebutkan satu persatu khususnya teman-teman OSIRIS, BEM FTIF Presisi Bermanfaat, dan Departemen Entreprenurship.

Terima kasih atas segala bantuan yang diberikan kepada penulis. Semoga Allah SWT membala kebaikan kalian semua. Akhir kata, penulis menyadari bahwa masih banyak kekurangan dalam pengerjaan Tugas Akhir ini, oleh karena itu penulis mengharapkan saran dan kritik bagi Tugas Akhir ini. Semoga Tugas Akhir ini dapat memberikan kebermanfaatan bagi kita semua.

DAFTAR ISI

LEMBAR PENGESAHAN	vii
LEMBAR PERSETUJUAN	ix
ABSTRAK	xi
ABSTRACT	xiii
KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xxi
DAFTAR TABEL	xxiii
DAFTAR KODE	xxv
1 BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	3
1.3 Batasan Maslaah	3
1.4 Tujuan	4
1.5 Manfaat	4
1.6 Relevansi	4
2 BAB II TINJAUAN PUSTAKA	5
2.1 Penelitian Sebelumnya	5
2.2 Dasar Teori	7
2.2.1 E-government	7
2.2.2 Sosial Media	8
2.2.3 Facebook	9
2.2.4 Twitter	9
2.2.5 Youtube	9

2.2.6	Regular Expression	10
2.2.7	N-grams	10
2.2.8	Web Crawling.....	11
2.2.9	Visualisasi.....	11
2.2.10	TF-IDF.....	12
2.2.11	Text Preprocessing.....	12
2.2.12	Stop word removal.....	12
2.2.13	Stemming.....	13
2.2.14	Peraturan Pemerintah tentang Sosial Media Pemerintah Daerah	13
2.2.15	Apache Kafka	14
3	BAB III METODOLOGI	15
3.1.	Tahapan Pelaksanaan Tugas Akhir	15
3.2.	Uraian Metodologi.....	17
3.2.1.	Studi Literatur.....	17
3.2.2.	Akuisisi Data	17
3.2.3.	<i>Text Preprocessing</i>	17
3.2.4.	Pengkategorian Komentar.....	20
3.2.5.	Rekomendasi Kata	20
3.2.6.	Perancangan dan Pembuatan Visualisasi dengan Website	21
3.2.7	Penyusunan Tugas Akhir	25
4	BAB IV PERANCANGAN.....	27
4.1	Pengambilan Data.....	27
4.2	Desain Database	37
4.2.1	Database Pemda.....	37
4.2.2	Database Facebook Comments	38

4.2.3	Database Twitter Comments.....	39
4.2.4	Database Youtube Comments.....	39
4.3	Metodologi Implementasi.....	40
4.3.1	Streaming Data	40
4.3.2	Pra-proses Data.....	40
4.3.3	Pengkategorian Data.....	41
4.3.4	Rekomendasi Keyword oleh Aplikasi.....	41
4.3.5	Desain antarmuka aplikasi visualisasi.....	45
5	BAB V IMPLEMENTASI	49
5.1	Lingkungan Implementasi	49
5.2	Pembuatan aplikasi.....	50
5.2.1	Streaming Data dari Social Media	50
5.2.2	Pengkategorian Komentar	76
5.2.3	Rekomendasi Keyword.....	77
5.2.4	Visualisasi Data	78
5.2.5	Fitur pada Aplikasi	87
6	BAB VI HASIL DAN PEMBAHASAN	95
6.1	Analisa Pengujian Use Case	95
6.2	Hasil Kategorisasi.....	97
6.3	Rekomendasi Kata Kunci	108
7	BAB VII KESIMPULAN DAN SARAN.....	111
7.1	Kesimpulan	111
7.2	Saran.....	112
8	DAFTAR PUSTAKA.....	113
9	LAMPIRAN A TAMPILAN ANTAR MUKA APLIKASI 117	
10	BIODATA PENULIS.....	141

Halaman ini sengaja dikosongkan

DAFTAR GAMBAR

Gambar 3.1 Metodologi Tugas Akhir.....	15
Gambar 3.2 Arsitektur Sistem	16
Gambar 3.3 Memasukkan Dinas Pemerintah Kota.....	23
Gambar 3.4 Dashboard Jumlah Komentar Setiap Bulannya..	24
Gambar 3.5 Dashboard Jumlah Kategori Komentar	24
Gambar 4.1 Jumlah Akun Facebook Resmi dan Influencer Pemerintah Kota.....	28
Gambar 4.2 Jumlah Akun Twitter Resmi dan Influencer Pemerintah Kota.....	29
Gambar 4.3 Jumlah Akun Youtube Resmi dan Influencer Pemerintah Kota.....	29
Gambar 4.4 Tree Diagram Facebook	34
Gambar 4.5 Tree Diagram Twitter	35
Gambar 4.6 Tree Diagram Youtube	36
Gambar 4.7 Use Case untuk Akun Pemerintah Kota.....	43
Gambar 4.8 Use Case untuk Admin	44
Gambar 4.9 Jumlah Komentar yang Masuk Pada Hari Ini	46
Gambar 4.10 Jumlah Komentar yang Masuk Pada Periode Tertentu.....	47
Gambar 4.11 Tampilan User Ketika Memasukkan Dinas dan Keyword.....	47
Gambar 4.12 Tampilan Aplikasi Menampilkan Rekomendasi Kepada User.....	48
Gambar 6.1 Komentar yang Masuk ke Database dengan Label Dinas	106
Gambar 6.2 Komentar yang Masuk ke Database dengan Label Duplicate	106
Gambar 6.3 Komentar yang Masuk ke Database dengan Label Uncategorized	107
Gambar 6.4 Dashboard Kategorisasi Komentar	108
Gambar 9.1 Tampilan Regier Akun Pemda oleh Admin	117
Gambar 9.2 Tampilan Edit Akun Pemerintah Daerah Oleh Admin	118
Gambar 9.3 Tampilan Menu Lihat Dinas, Edit Pemda, dan Hapus Akun Pemda oleh Admin	119

Gambar 9.4 Tampilan Menu Restore Akun dan Hapus Permanen Pemerintah Daerah oleh Admin.....	120
Gambar 9.5 Tampilan Tambah Dinas oleh Admin	121
Gambar 9.6 Tampilan Edit Dinas oleh Admin	122
Gambar 9.7 Tampilan Menu Hapus Dinas oleh Admin.....	123
Gambar 9.8 Tampilan Email Akun Pemda Telah Diverifikasi	124
Gambar 9.9 Tampilan Email Akun ditolak.....	125
Gambar 9.10 Tampilan Register Pemerintah Daerah	126
Gambar 9.11 Tampilan Edit Akun Pemerintah Daerah Oleh User	127
Gambar 9.12 Tampilan Tambah Dinas Oleh User.....	128
Gambar 9.13 Tampilan Edit Dinas Oleh User	129
Gambar 9.14 Tampilan Menu Hapus dan Edit Dinas	130
Gambar 9.15 Tampilan Email Permintaan Reset Password.	131
Gambar 9.16 Tampilan Reset Password	132
Gambar 9.17 Grafik Jumlah Komentar 10 Hari Terakhir	133
Gambar 9.18 Jumlah Komentar Seluruh Pemda	134
Gambar 9.19 Top 5 Kategorisasi Komentar Dinas Terbanyak	135
Gambar 9.20 Tabel Jumlah Komentar Setiap Dinas	136
Gambar 37 Tampilan Komentar Kategori Dinas	137
Gambar 38 Edit Komentar Kategori Dinas.....	138
Gambar 39 Komentar Kategori Duplicate	139
Gambar 40 Komentar Kategori Uncategorized	140

DAFTAR TABEL

Tabel 3.1 Contoh Hasil Stemming	18
Tabel 3.2 Contoh Hasil Stopword Removal	18
Tabel 3.3 Contoh Hasil N-Grams	19
Tabel 4.1 Tabel Jumlah Akun Resmi dan Influencer.....	27
Tabel 4.2 Tabel Field Facebook yang Digunakan	30
Tabel 4.3 Tabel Field Twitter yang Digunakan	31
Tabel 4.4 Tabel Field Youtube yang Digunakan	32
Tabel 5.1 Tabel Spesifikasi Perangkat Keras yang Digunakan	49
Tabel 5.2 Tabel Perangkat Lunak yang Digunakan	49
Tabel 6.1 Use Case Pengujian Untuk User.....	95
Tabel 6.2 Use Case Pengujian Untuk Admin	96
Tabel 6.3 Lama Waktu Loading Halaman Dashboard.....	97
Tabel 6.4 Jumlah Komentar Kota Depok	97
Tabel 6.5 Dinas dan Keyword Dinas Kota Depok.....	98
Tabel 6.6 Jumlah Komentar Setelah Preprocessing	100
Tabel 6.7 Jumlah Komentar Terkategorisasi	101
Tabel 6.8 Jumlah Komentar Untuk Dinas Kota Depok	103
Tabel 6.9 Rekomendasi Kata Kunci	108

Halaman ini sengaja dikosongkan

DAFTAR KODE

Kode 5.1 Method ScrapeFacebookPageFeedStatus	53
Kode 5.2 Method getCommentsForStatuses	55
Kode 5.3 Method Main	57
Kode 5.4 Method getCommentDocument	57
Kode 5.5 Method send_message	58
Kode 5.6 Method pushCommentDocument	58
Kode 5.7 Method StdOutListener.....	59
Kode 5.8 Method ambil_list.....	60
Kode 5.9 Method Main	61
Kode 5.10 method channelusernameToid.....	62
Kode 5.11 Method channelNameForfilename.....	63
Kode 5.12 Method crawlVideo	65
Kode 5.13 Method crawlComments	67
Kode 5.14 Method Main	69
Kode 5.15 Parsing_facebookComment	70
Kode 5.16 parsing_tweetReply	71
Kode 5.17 parsing_youtubeComment	71
Kode 5.18 Method processTwitter	72
Kode 5.19 Method processFacebook.....	73
Kode 5.20 Method processYoutube	74
Kode 5.21 stopword removal	74
Kode 5.22 stemming	75
Kode 5.23 N-grams	75
Kode 5.24 Method insert_mongo	76
Kode 5.25 Pseudocode pengkategorian komentar	77
Kode 5.26 Pseudocode Rekomendasi Keyword	78
Kode 5.27 Query Lima Kategori Dinas Terbanyak	81
Kode 5.28 Query jumlah komentar 10 hari terakhir	82
Kode 5.29 Query jumlah komentar setiap pemda.....	84
Kode 5.30 Method standar deviasi dan rata-rata	85
Kode 5.31 Query tabel jumlah komentar setiap dinas	87
Kode 5.32 Mengubah Kategori Dinas	88
Kode 5.33 Mengubah Kategori Duplicate	91
Kode 5.34 Mengubah Kategori Uncategorized	93
Kode 5.35 Validasi Kata Kunci.....	94

Halaman ini sengaja dikosongkan

BAB I

PENDAHULUAN

Bab ini akan menguraikan gambaran umum penelitian meliputi latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan relevansi terhadap pengerjaan tugas akhir. Tujuan dan manfaat dilakukannya penelitian diharapkan dapat memberikan gambaran awal mengenai penelitian.

1.1 Latar Belakang

Dalam pemerintahan tingkat kota, terdapat Satuan Kerja Perangkat Daerah (SKPD) yang bertujuan untuk menjalankan kebijakan, program, dan kegiatan pembangunan berdasarkan tugas dan fungsinya[1]. Dasar hukum yang mengatur dalam pembentukan SKPD terdapat dalam pasal 120 UU no. 32 tahun 2004 tentang Pemerintahan Daerah. SKPD adalah pelaksana fungsi eksekutif yang harus berkoordinasi satu sama lain agar penyelenggaraan pemerintahan dapat berjalan dengan baik. SKPD sendiri terbagi menjadi sekretariat daerah, staf-staf ahli, dinas-dinas, badan-badan, inspektorat daerah, dan lembaga-lembaga lain yang bertanggung jawab langsung kepada kepala daerah. Dengan adanya SKPD, pemerintah kota dapat membagi program pembangunan berdasarkan tugas dan fungsinya masing-masing. Di lain sisi, pemerintah sedang mengencarkan konsep *e-government* sebagai upaya pemanfaatan dan penggunaan teknologi informasi dan komunikasi untuk meningkatkan efisiensi dan efektifitas pemerintahan. Melalui *e-government*, pemerintah dapat memberikan berbagai jasa pelayanan kepada masyarakat, menyediakan akses informasi dan mengadakan proses pemerintahan yang bertanggung jawab dan transparan kepada masyarakat.

Penerapan konsep *e-government* tidak terlepas dari kemajuan pengguna internet di Indonesia yang mengalami pengembangan

yang pesat. Berdasarkan survey yang dilakukan oleh internetworldstats yang dirilis pada Januari 2018, jumlah pengguna internet di Indonesia telah mencapai 132,7 juta dari jumlah penduduk Indonesia sebesar 256,2 juta jiwa[2]. Hal ini menandakan kenaikan sebesar 51,8% dibandingkan dengan jumlah pengguna internet di Indonesia pada tahun 2014 dengan jumlah 88 juta pengguna internet. Dari jumlah tersebut, 40% diantaranya merupakan pengguna aktif media sosial. Dengan jumlah sebesar 106 juta jiwa yang menggunakan sosial media, 85% diantaranya mengakses sosial media melalui perangkat seluler[3].

Melalui sosial media, masyarakat dapat melakukan interaksi kepada akun sosial media pemerintah kota dalam memberikan informasi atau kritik dan saran terkait kotanya. Berdasarkan penelitian yang dilakukan oleh Sultana Lubna Alam dan Aodah Diamah pada halaman akun Facebook Pariwisata Pemerintah Australia, masyarakat secara aktif melakukan interaksi dengan cara memberikan informasi terbaru di halaman Facebook Pariwisata Pemerintahan Australia[4]. Dengan melihat kesempatan ini, pemerintah kota dapat mengetahui informasi dari komentar masyarakat melalui sosial media terkait permasalahan yang sedang terjadi atau topik yang sedang ramai dibicarakan masyarakat. Dengan mengetahui informasi atau permasalahan yang terjadi di suatu kota, pemerintah kota dapat melakukan aksi sesuai dengan SKPD terkait sesuai dengan tugas dan fungsinya masing-masing.

Melalui fenomena tersebut, diperlukanlah suatu aplikasi yang bertujuan untuk melihat komentar masyarakat di sosial media yang telah terkategorisasi berdasarkan SKPD di masing-masing kota. Dengan aplikasi ini, pemerintah kota dapat mengetahui topik apa yang sedang ramai dibicarakan masyarakat atau masalah yang sedang terjadi. Proses kategorisasi disesuaikan dengan kata kunci yang terkait dengan SKPD di masing-masing kota agar pemerintah kota langsung memberikan arahan kepada SKPD terkait

1.2 Perumusan Masalah

Berdasarkan latar belakang diatas, maka rumusan masalah yang akan diteliti pada tugas akhir ini adalah:

1. Bagaimana cara mengakuisisi data komentar masyarakat melalui sosial media *Facebook*, *Twitter*, dan *Youtube*?
2. Bagaimana cara melakukan kategorisasi komentar masyarakat berdasarkan SKPD pemerintah kota?
3. Bagaimana cara aplikasi memberikan rekomendasi *keyword* yang terkait dengan SKPD pemerintah kota?
4. Bagaimana bentuk visualisasi data yang tepat untuk topik komentar yang diberikan masyarakat ke dalam sebuah gambar atau *dashboard*?

1.3 Batasan Masalah

Batasan masalah dalam penggerjaan tugas akhir ini yaitu:

1. Data komentar yang diambil merupakan komentar masyarakat pada sosial media *Facebook*, *Twitter*, dan *Youtube* akun resmi pemerintah kota dan akun *influencer* di setiap kota di Indonesia.
2. Akun *influencer* merupakan akun sosial media yang memiliki social authority tertinggi untuk kategori akun berita dan informasi pada situs moz.com/followerwonk dan hasil pencarian melalui Google Search.
3. Komentar yang digunakan dalam proses kategorisasi merupakan komentar yang menggunakan Bahasa Indonesia.
4. Data komentar yang diambil pada akun sosial media merupakan komentar pada 10 *post* terakhir untuk sosial media Facebook dan Youtube karena tidak menyediakan API streaming.
5. Pengambilan data menggunakan API di masing-masing sosial media berdasarkan regulasi yang diberikan.

1.4 Tujuan

Adapun tujuan yang ingin dicapai dari penggerjaan tugas akhir ini yaitu membangun sebuah aplikasi yang dapat digunakan pemerintah kota untuk mengetahui topik apa saja yang sering dibicarakan masyarakat dan rekomendasi kata kunci berdasarkan SKPD pemerintah kota di masing-masing kota. Dengan adanya aplikasi ini diharapkan pemerintah kota dapat mengetahui kinerja dari masing-masing SKPD yang dimiliki berdasarkan komentar masyarakat.

1.5 Manfaat

Manfaat yang diharapkan dari tugas akhir ini adalah:

1. Memfasilitasi pemerintah kota di setiap kota dalam melihat kinerja SKPD yang dimiliki
2. Memfasilitasi mahasiswa, khususnya Departemen Sistem Informasi dalam mempelajari *social media analysis*
3. Menyediakan data yang dapat digunakan sebagai acuan untuk menentukan tindakan lebih lanjut dalam hal *egovbench* untuk seluruh pemerintah kota di Indonesia.

1.6 Relevansi

Tugas akhir berkaitan dengan mata kuliah Pemrograman Berbasis Web, Analisa dan Desain Perangkat Lunak dan Konstruksi Pengembangan Perangkat Lunak, Sistem Cerdas, dan Penggalian Data Analitika Bisnis.

BAB II

TINJAUAN PUSTAKA

Bab ini menjelaskan mengenai penelitian sebelumnya dan dasar teori yang dijadikan acuan atau landasan dalam pengerjaan tugas akhir. Landasan teori akan memberikan gambaran secara umum dari landasan penjabaran tugas akhir ini.

2.1 Penelitian Sebelumnya

Pada subbab ini dijelaskan tentang referensi penelitian sebelumnya yang terkait dengan tugas akhir. Bagian ini memaparkan acuan penelitian sebelumnya yang digunakan oleh penulis dalam melakukan penelitiannya.

1. Penelitian pertama mengenai *Twitter and Health in the Australian Context: What Types of Information are Health-related Organizations Tweeting?* oleh Dan Dumbrell dan Robert Steele pada tahun 2013. Penelitian ini bertujuan untuk mengetahui informasi apa yang dibagikan dan area kesehatan apa yang diberikan oleh 114 akun Twitter kesehatan di Australia. 114 akun Twitter kesehatan tersebut dibagi menjadi 3 sektor yaitu *government*, *for-profit*, dan *non-profit*. Dalam melakukan kategorisasi tweet, penulis melakukan kategorisasi manual karena alat analisis dari Twitter sendiri belum dapat melakukan kategori khusus pada bidang kesehatan. Adapun hasil dari penelitian ini informasi yang diberikan oleh 114 akun Twitter kesehatan di Australia berupa fundraising, saran mengenai kesehatan masyarakat, dan berita organisasi. Sementara area kesehatan yang sering dibahas pada akun Twitter kesehatan yaitu mengenai kanker dan kesehatan mental.

2. Penelitian kedua mengenai *Twitter for City Police Department Information Sharing* oleh Thomas Heverin dan Lisl Zach pada tahun 2016. Penelitian ini bertujuan untuk menemukan informasi apa yang dibagikan oleh 30 akun Twitter departemen kepolisian di Amerika Serikat dan melihat bagaimana interaksi masyarakat terhadap informasi yang diberikan melalui akun Twitter masyarakat di Amerika Serikat. Metode yang digunakan untuk melakukan kategorisasi berupa pendekatan *Open Coding*. Melalui pendekatan *Open Coding*, kategori yang dihasilkan untuk informasi yang diberikan oleh akun Twitter kepolisian yaitu *Crime/Incident Information*, *Department Information*, *Event Information*, *Traffic Information*, *Prevention Information*, *Person Identification*, *Reply/Mention*, *Retweet*, *Data*, dan *Other*. Berdasarkan hasil diketahui bahwa Akun Twitter departemen kepolisian di seluruh Amerika Serikat membagikan informasi mengenai kejahatan dan insiden. Sementara itu, kategori yang dihasilkan dari akun Twitter masyarakat yaitu *Direct Communication*, *Mention*, *Retweet*, dan *Retweet with Comment*. Berdasarkan hasil diketahui bahwa interaksi masyarakat terhadap informasi yang diberikan mayoritas berupa retweet informasi dan melakukan mention kepada akun Twitter kepolisian.
3. Penelitian ketiga mengenai *Short Text Classification in Twitter to Improve Information Filtering* oleh Bharath Sriram pada tahun 2010. Penelitian ini berawal dari permasalahan metode kategorisasi dengan menggunakan “*Bag-of-Words*” yang tidak cocok digunakan pada teks pendek seperti tweet. Penulis mengusulkan metode klasifikasi dengan menggunakan filtering fitur pada untuk setiap kategori tweet. Penulis melakukan kategorisasi tweet menjadi lima kategori yaitu *News*,

Events, Opinions, Deal, dan Private Messages dimana setiap masing-masing kategori memiliki fiturnya. Selain itu, penulis memberikan usulan agar pengguna dapat memberikan kategori baru berdasarkan keinginan pengguna. Untuk menghindari turunnya akurasi karena tingginya jumlah kategori maka pengguna harus melakukan pembuatan fitur baru sesuai dengan kategori baru. Berdasarkan hasil akurasi, metode ini memiliki nilai akurasi yang lebih tinggi dibandingkan model *Bag-of-Words*.

2.2 Dasar Teori

Berikut penjelasan teori-teori yang mendukung serta berkaitan dalam pengerjaan tugas akhir yang dikerjakan

2.2.1 E-government

E-government merupakan upaya pemanfaatan dan penggunaan teknologi informasi dan komunikasi untuk meningkatkan efisiensi dan efektifitas pemerintahan. Selain itu, e-government bertujuan untuk memberikan berbagai jasa pelayanan kepada masyarakat secara lebih baik, menyediakan secara luas akses informasi kepada publik, dan mengadakan proses pemerintahan yang lebih bertanggung jawab dan transparan kepada masyarakat[5].

Model e-government yang diterapkan pada negara luar menggunakan empat fase perkembangan yang meliputi[6]:

- a. *Fase pertama*, berupa tampilan website yang berisikan informasi dasar yang dibutuhkan masyarakat.
- b. *Fase kedua*, berupa fase interaksi pada website yang membuat isi informasi lebih bervariasi seperti fasilitas download dan komunikasi e-mail dalam website pemerintah.
- c. *Fase ketiga*, berupa fase transaksi seperti penerapan aplikasi/formulir untuk secara online mulai diterapkan.

- d. *Fase keempat*, berupa fase transformasi dalam pelayanan yang terintegrasi. Terintegrasi yang dimaksud tidak hanya pemerintah dengan masyarakat, tetapi juga dengan organisasi lain.

Di Indonesia sendiri, tahapan pengembangan e-government diatur dalam Buku Panduan Kominfo yang terbagi menjadi 4 tingkatan yaitu[7]:

- a. Tingkat 1, merupakan tingkat persiapan berupa pembuatan situs web sebagai media informasi dan komunikasi pada setiap lembaga serta sosialisasi situs web untuk internal dan publik.
- b. Tingkat 2, merupakan tingkat pematangan berupa pembuatan situs web informasi public yang bersifat interaktif dan pembuatan antar muka keterhubungan dengan lembaga lain.
- c. Tingkat 3, merupakan tingkat pemantapan yang berisi pembuatan situs web yang bersifat transaksi pelayanan publik dan pembuatan interoperabilitas aplikasi dan data dengan lembaga lain.
- d. Tingkat 4, merupakan tingkat pemanfaatan yang berisi pembuatan aplikasi untuk pelayanan yang bersifat *Government to Government* (G2G), *Government to Business* (G2B), dan *Government to Consumers* (G2C).

Pengembangan E-government di Indonesia dijadikan fasilitas agar sistem pemerintahan di Indonesia dapat berjalan secara efektif, efisien, transparatif, dan akuntabilitas.

2.2.2 Sosial Media

Menurut O'Reilly, social media adalah platform yang mampu memfasilitasi berbagai kegiatan seperti mengintegrasikan situs web, interaksi sosial, dan pembuatan konten berbasis komunitas[8]. Jumlah pengguna aktif sosial media di Indonesia terhitung pada bulan Januari tahun 2018 berjumlah 130 juta atau sebesar 49% dari total penduduk Indonesia seluruhnya[2]. Dengan melihat tingginya jumlah pengguna sosial media pada

masyarakat hal ini dapat membantu tujuan dari penerapan e-government agar pemerintah dapat menampung semua aspirasi dari seluruh golongan masyarakat.

2.2.3 Facebook

Menurut Boyd dan Ellison, Facebook merupakan sebuah platform online dimana pengguna dapat membuat profil, menghasilkan dan membagikan konten informasi, serta melakukan interaksi dengan kontak baik yang diketahui maupun tidak diketahui[9]. Sosial media yang didirikan pada tahun 2004 ini telah mencapai lebih dari satu miliar pengguna aktif pada tahun 2012[10]. Di Indonesia sendiri, jumlah pengguna aktif Facebook mencapai urutan ke tiga sebagai pengguna terbanyak dari seluruh dunia[11]. Dengan melihat jumlah tersebut, Facebook dapat dimanfaatkan oleh pemerintah untuk dapat membagikan informasi dan menampung aspirasi dari seluruh golongan masyarakat.

2.2.4 Twitter

Twitter merupakan layanan microblogging yang berdiri sejak tahun 2006. Dengan jumlah user mencapai 41 juta tiga tahun setelah perilisannya, Twitter memberikan keterbatasan sebesar 140 karakter dalam postingannya yang disebut *tweet*. Pengguna Twitter dapat melakukan *follow* kepada pengguna Twitter lainnya untuk menerima semua *tweet* pengguna. Pada Twitter terdapat istilah RT yang berarti *retweet*, '@' untuk melakukan mention kepada pengguna Twitter, dan '#' untuk memberikan kata sebagai sebuah *hashtag*[12].

2.2.5 Youtube

Youtube merupakan konten komunitas yang didirikan pada tahun 2005. Melalui Youtube, pengguna dapat melakukan posting, melihat, memberikan komentar, dan *link* video di situs Youtube. Pengguna dapat melakukan pengaturan profil mereka agar dapat menampilkan siapa saja profil yang mereka *subscribe*, aktivitas terbaru, jumlah teman, komentar, dan video favorit[13]. Selain pengaturan profil, pengguna dapat memberikan *tag* kepada video yang mereka unggah dengan *keywords* atau frasa yang sesuai dengan video tersebut. Melalui

tag ini Youtube dapat memberikan daftar video yang saling terkait[14]. Berdasarkan riset, Youtube merupakan situs berbagi video terbesar dengan 100 juta video yang diakses setiap harinya dan 65.000 video yang diunggah setiap harinya[15]. Berdasarkan penelitian yang dilakukan oleh Kruitbosch dan Nack di tahun 2008 serta Burgess dan Green di tahun 2009, video yang sering dilihat pada situs Youtube merupakan video yang diproduksi secara profesional[16], sementara video yang sering diberikan komentar adalah video yang diproduksi oleh user secara amatir[17].

2.2.6 Regular Expression

Regular Expression atau *Regex* merupakan pattern yang mengspesifikasikan kumpulan karakter yang ada dalam sebuah kalimat. Regex merupakan kalimat yang merupakan kombinasi dari *normal characters* dan *metacharacters* atau *metasequences*. *Metacharacters* atau *metasequences* merupakan sebuah karakter yang mewakili makna seperti kuantitas, lokasi, atau tipe karakter untuk melakukan *regex*[18]. Sekumpulan kata yang didapatkan dari *regular expression* dinamakan dengan *Pattern matching*.

2.2.7 N-grams

N-grams merupakan sebuah teknik untuk memecah kalimat menjadi potongan N kata. Dengan melakukan pemecahan kalimat menjadi kata, N grams dapat digunakan untuk memprediksi kata selanjutnya dalam sebuah kalimat. Hal ini membuat prediksi kata tidak hanya berupa satu kata namun lebih dari satu kata. N-grams terbagi menjadi empat jenis yaitu Unigrams dengan N=1, Bigrams untuk N=2, Trigrams dengan N=3, dan N grams untuk N>3. Dalam memberikan prediksi kata, N-grams bekerja menggunakan Markov Assumption[19]. Hal ini bertujuan agar prediksi kata yang digunakan tidak melihat kata sebelumnya yang terlalu jauh. Perhitungan prediksi kata selanjutnya dihitung dengan menggunakan formula:

$$P(Wn | Wn - 1) = \frac{Count(Wn - 1 \text{ } Wn)}{Count (Wn - 1)}$$

W_{n-1} = kata sebelumnya

W_n = prediksi kata

Equation 1 Rumus N-gram

Dengan melihat kemungkinan tertinggi dari semua kata, prediksi pemilihan kata dapat dilakukan. Hal ini dinamakan dengan *Maximum Likelihood Estimation*.

Salah satu permasalahan yang terdapat pada N-grams yaitu bergantungnya struktur kalimat dalam pemilihan kata. Sebagai contoh, apabila kalimat yang diberikan memiliki keterangan tempat setelah subyek baru kemudian prediksi kata maka prediksi kata akan melihat kata terakhir dalam keterangan. Hal ini akan mengakibatkan bias karena prediksi kata akan lebih tepat jika melihat pada subjek yang diberikan.

2.2.8 Web Crawling

Web Crawling merupakan aktivitas unduh dari suatu halaman website yang dilakukan secara berulang untuk membuat *repository* dari sebuah halaman website. Repository dari sebuah halaman website nantinya akan dilakukan indeks dan biasa digunakan untuk web search atau dilakukan analisa dari karakteristik website tersebut[20]. Sebuah web crawler dapat melakukan perhitungan tambahan saat melakukan crawling untuk mengidentifikasi situs halaman yang berhubungan atau menolak situs halaman yang teridentifikasi duplikasi[21].

2.2.9 Visualisasi

Visualisasi adalah konversi data ke dalam format visual atau table sehingga karakteristik dari data dan relasi di antara item data atau atribut dapat di analisis dan di laporan[22]. Dengan melakukan visualisasi, pengguna dapat menangkap konsep dari data yang ditampilkan karena manusia lebih mudah mengartikan gambar daripada teks yang menjelaskan arti dari gambar tersebut[23].

2.2.10 TF-IDF

Metode TF-IDF merupakan suatu cara untuk memberikan bobot hubungan suatu kata (term) terhadap dokumen. Metode ini menggabungkan dua konsep perhitungan bobot yaitu frekuensi kemunculan kata dalam dokumen dan *inverse* frekuensi dokumen yang mengandung kata tersebut[24]. TF adalah algoritma pembobotan heuristic yang menentukan bobot dokumen berdasarkan kemunculan term (istilah). Semakin sering sebuah term (istilah) muncul maka semakin tinggi bobot dokumen untuk istilah tersebut[25]. IDF merupakan banyaknya istilah tertentu dalam keseluruhan dokumen. Semakin sedikit jumlah dokumen yang mengandung term yang dimaksud maka nilai pada idf akan semakin besar.

$$idf\ j = \log \frac{n}{n_j}$$

n = jumlah seluruh dokumen

n = jumlah dokumen yang mempunyai istilah j

Equation 2 Rumus IDF

2.2.11 Text Preprocessing

Text preprocessing merupakan aktivitas untuk mewakilkan setiap dokumen menjadi sebuah vector dimana aktivitas ini bertujuan untuk memisahkan sebuah teks menjadi individual kata[26]. Memilih kata kunci dalam proses pemilihan kata merupakan langkah penting dalam melakukan index pada dokumen. Dimana kata kunci merupakan kata yang memberikan makna pada sebuah dokumen dan sebagai pembeda antar dokumen. Dalam melakukan *text preprocessing*, terdapat tiga tahapan yang umum dilakukan yaitu *stop word removal*, *stemming*, dan *document indexing*.

2.2.12 Stop word removal

Stop word merupakan kata-kata yang sering muncul dalam sebuah dokumen namun tidak memberikan informasi penting dari dokumen itu sendiri. Beberapa jenis kata yang termasuk *stop word* yaitu *pronouns* (kata ganti orang), *prepositions* (kata

yang menerangkan tempat), *conjunctions* (kata penghubung). *Stop word removal* merupakan langkah pertama dalam melakukan *text preprocessing* dan merupakan langkah yang penting untuk dilakukan[27].

2.2.13 Stemming

Stemming digunakan untuk mengetahui kata dasar dari sebuah kata dalam dokumen. *Stemming* dilakukan karena setiap kata dapat memiliki arti yang sama dengan kata dasarnya. Hal ini membuat makna kata dalam dokumen digabungkan dalam satu kata dasar.

2.2.14 Peraturan Pemerintah tentang Sosial Media Pemerintah Daerah

Peraturan pemerintah mengenai sosial media pada pemerintah daerah diatur dalam Peraturan Menteri Pendayagunaan Aparatur Negara Dan Reformasi Birokrasi Republik Indonesia Nomor 83 Tahun 2002 Tentang Pedoman Pemanfaatan Media Sosial Instansi Pemerintah[28]. Pada peraturan tersebut disebutkan jenis media sosial yaitu:

- Media publikasi / blog, seperti Blogspot, Blogger, dan Wordpress
- Microblog, seperti Twitter dan Plurk
- Media berbagi (media-sharing), seperti Flickr, YouTube, dan Slideshare
- Media jejaring social, seperti Facebook, MySpace, Hi5, Google+, LinkedIn, dan WAYN
- Media kolaborasi/wiki, seperti Wikipedia, Wikimapia, dan Wikileaks
- Forum diskusi, seperti Kaskus dan www.webcosmoforums.com
- Media percakapan, seperti Google Talk, Yahoo Messenger, dan Skype
- Situs ulasan (review) seperti Goodread dan Yelp.

Sementara itu, tujuan instansi dalam menggunakan media social yaitu:

- Menyimak (listening), yaitu instansi menggunakan media social untuk memahami dan menyerap aspirasi kebutuhan khayalak
- Berbicara (talking), yaitu instansi menggunakan media social untuk menyebarluaskan pesan dan informasi
- Menyemangati (energizing), yaitu instansi menggunakan media sosial untuk membangun semangat dan keterlibatan serta mendorong khayalak menyebarluaskan pesan melalui percakapan dari mulut ke mulut (*word-of-mouth*) dan komunikasi viral (melalui internet)
- Mendukung (supporting), yaitu instansi menggunakan media sosial untuk membantu khayalak agar saling mendukung sehingga tercipta dukungan yang lebih besar
- Merangkul (embracing), yaitu instansi menggunakan media sosial untuk melibatkan khalayak ke dalam kegiatan instansi, termasuk dalam memberikan masukan, saran, gagasan, dan/atau tindakan nyata.

2.2.15 Apache Kafka

Apache Kafka merupakan aplikasi yang dikembangkan dengan tujuan untuk memecahkan permasalahan terhadap penyampaian data dengan volume besar ke pelanggan yang beragam[29]. Sejak tahun 2012, Kafka telah menjadi aplikasi *open-source* Apache tingkat atas. Aplikasi ini dikembangkan mengingat pentingnya penyampaian data dengan aktivitas volume yang tinggi secara *real-time* kepada pengguna dan sistem. Dengan menggunakan Kafka, maka arsitektur terbentuk secara terdistribusi yang dapat memberikan daya tahan dan ketersedian yang efisien. Beberapa perusahaan telah menggunakan Apache Kafka dalam menjalankan operasional perusahaan seperti LinkedIn, Twitter, Foursquare, DataSift, dan Square[30].

BAB III

METODOLOGI

Pada bab metodologi penelitian menjelaskan tahapan-tahapan yang dilakukan dalam pengerjaan tugas akhir beserta deskripsi dan penjelasan dari setiap tahapan.

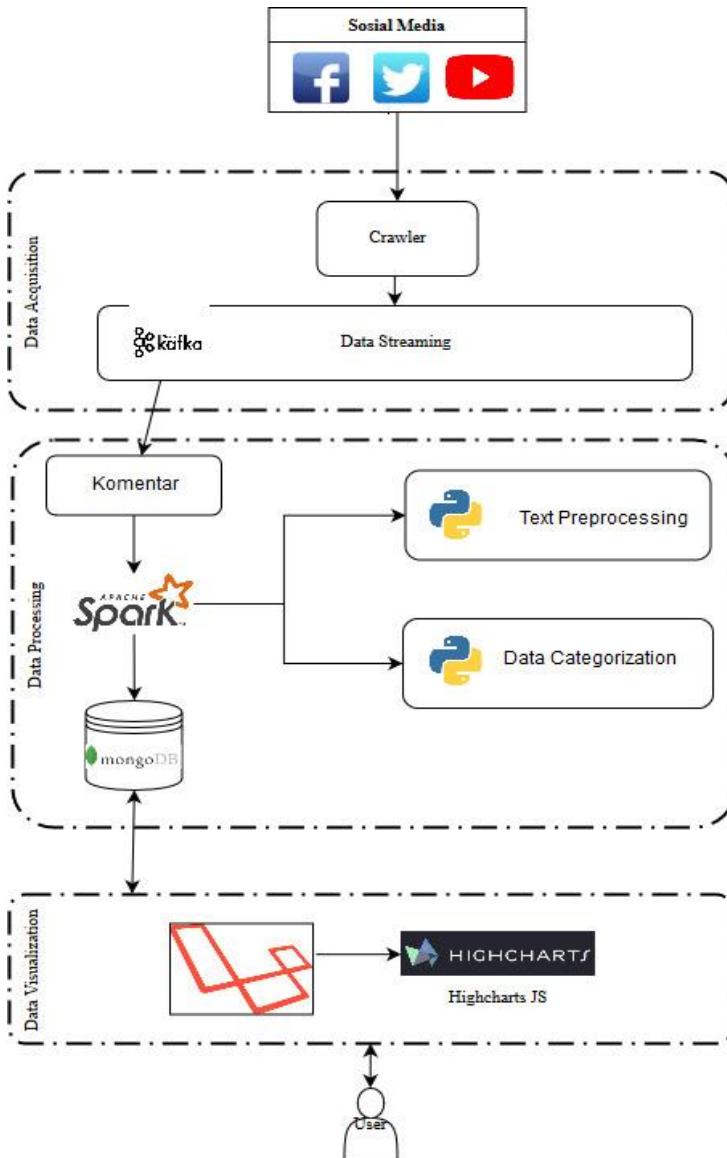
3.1. Tahapan Pelaksanaan Tugas Akhir

Pada sub bab ini menjelaskan mengenai tahapan-tahapan dalam pelaksanaan tugas akhir. Metodologi ini dapat dilihat pada gambar 3.1.



Gambar 3.1 Metodologi Tugas Akhir

Berikut merupakan arsitektur sebagai usulan tugas akhir ini



Gambar 3.2 Arsitektur Sistem

3.2. Uraian Metodologi

Pada sub bab ini menjelaskan secara lebih rinci masing-masing tahapan yang dilakukan untuk penyelesaian tugas akhir

3.2.1. Studi Literatur

Tahapan studi literatur dilakukan untuk memahami konsep, metode, dan teknologi sesuai bahasan dan permasalahan sehingga penulis dapat memberi solusi atas permasalahan yang digunakan dalam penyusunan tugas akhir. Tahapan studi literatur dilakukan dengan mempelajari literatur-literatur terkait konsep yang digunakan pada penelitian-penelitian sebelumnya.

3.2.2. Akuisisi Data

Tahap ini dilakukan dengan pengambilan data pada akun resmi sosial media pemerintah kota di Indonesia dan akun *influencer* di masing-masing kota. Adapun sosial media yang digunakan yaitu *Facebook*, *Twitter*, dan *Youtube*. Proses pengambilan data dilakukan dengan teknik *crawling*. Proses *crawling* memanfaatkan *Application Programming Interface* (API) dari masing-masing sosial media. Adapun entitas yang diambil dalam proses *crawling* yaitu komentar yang masuk pada akun resmi sosial media pemerintah kota. Dalam melakukan akuisisi data digunakan aplikasi agregasi Apache Kafka untuk menyatukan data yang bersifat *streaming* dan digunakan database MongoDB sebagai tempat penyimpanan data.

3.2.3. *Text Preprocessing*

Tahapan *preprocessing data* dilakukan untuk mengubah setiap kata pada komentar di sosial media yang masuk menjadi sebuah vector. Dalam melakukan *text preprocessing*, terdapat tiga tahapan yang dilakukan yaitu *stemming*, *stop word removal*, *n-grams*, dan *document indexing*

3.2.3.1 Stemming

Tahapan selanjutnya yaitu melakukan proses *stemming* untuk mengubah kata dalam komentar yang masuk menjadi kata dasar. Proses perubahan dilihat berdasarkan kata apakah suatu

kata memiliki awalan, sisipan, akhiran, dan kombinasi dari awalan dan akhiran. Proses ini bertujuan untuk menyeragamkan kata-kata yang bermakna sama tetapi menjadi berbeda arti karena adanya imbuhan. *Stemming* dilakukan dengan menggunakan *library* Sastrawi dan Apache Spark. Berikut merupakan contoh untuk *stemming*:

Tabel 3.1 Contoh Hasil Stemming

Sebelum <i>Stemming</i>	Setelah <i>Stemming</i>
@SapawargaSby Apakah untuk pengurusan pindah datang masuk ke KK dan ber KTP Surabaya harus bekerja di Surabaya jg? Krn suami saya bekerja di Jakarta & berKK Bogor.	sapawargasby apakah untuk urus pindah datang masuk ke kk dan ber ktp surabaya harus kerja di surabaya jg krn suami saya kerja di jakarta berkk bogor

3.2.3.2 *Stopword Removal*

Tahapan penghilangan kata yang sering muncul dan tidak memiliki makna bertujuan untuk memudahkan proses identifikasi topik dari komentar yang masuk. Melalui tahapan ini, kata-kata yang memiliki frekuensi kemunculan yang tinggi dan tidak memiliki makna akan dihilangkan. Adapun daftar *stopword* yang digunakan adalah *stopwords* Bahasa Indonesia sesuai dengan *library* Sastrawi dengan kamus kata dasar yang berasal dari kateglo.com dan Apache Spark. Berikut merupakan contoh untuk *stopword removal*:

Tabel 3.2 Contoh Hasil Stopword Removal

Sebelum <i>Stopword Removal</i>	Setelah <i>Stopword Removal</i>
sapawargasby apakah untuk urus pindah datang masuk ke kk dan ber ktp surabaya harus kerja di surabaya jg krn	Sapawargasby untuk urus pindah datang masuk kk ber ktp surabaya kerja surabaya

suami saya kerja di jakarta berkk bogor	jg krn suami kerja Jakarta berkk bogor
--	---

3.2.3.3 N-Grams

Langkah selanjutnya yaitu melakukan N-grams dari setiap kata kunci yang telah dikategorikan. N-grams bertujuan untuk memberikan rekomendasi untuk kata kunci dengan melihat pola dari kata kunci yang telah ditentukan sebelumnya. Dalam melakukan N-Grams digunakan library n-grams pada python dan Apache Spark. Contoh pemanfaatan N-grams untuk dinas PDAM, kata kunci “air” memiliki hubungan dengan kata kunci “mati”.

Tabel 3.3 Contoh Hasil N-Grams

Sebelum n-grams	Setelah n-grams
Sapawargasby untuk urus pindah datang masuk kk ber ktp surabaya kerja surabaya jg krn suami kerja Jakarta berkk bogor	Sapawargasby
	surabaya kerja
	Jakarta berkk
	kerja Jakarta
	suami kerja
	krn suami
	jg krn
	surabaya jg
	kerja surabaya
	ktp surabaya
	Sapawargasby untuk
	ber ktp
	kk ber

	masuk kk
	datang masuk
	pindah datang
	urus pindah
	untuk urus
	berkk bogor

3.2.4. Pengkategorian Komentar

Tahap ini bertujuan untuk melakukan kategorisasi dari komentar yang masuk. Dengan melakukan pattern matching, komentar yang memiliki kata kunci yang ditemukan dari *text preprocessing* akan disesuaikan dengan SKPD di pemerintah kota. Dalam melakukan pattern matching, terdapat langkah yang dilakukan yaitu *Regular Expressions*.

3.2.4.1 Regular Expressions

Regular Expressions (Regex) bertujuan untuk menemukan karakter yang ada dalam sebuah kalimat. Melalui *Regex*, kata kunci yang ditemukan pada *text preprocessing* dapat ditemukan sehingga dapat disesuaikan dengan SKPD di pemerintah kota. Adapun tools yang digunakan dalam proses regular expressions yaitu python. Contoh penggunaan *Regex* yaitu kata “jalan”, “rusak”, dan “macet” masuk ke dalam kategori unit dinas perhubungan.

3.2.5. Rekomendasi Kata

Pada tahap ini dilakukan perhitungan IDF untuk komentar yang telah memiliki label SKPD. Kata yang memiliki nilai IDF paling rendah akan diambil dan digunakan sebagai rekomendasi kata kunci kepada SKPD terkait.

3.2.6. Perancangan dan Pembuatan Visualisasi dengan Website

Pada tahap ini dilakukan tahapan perancangan dan pembuatan visualisasi dengan website. Tahap ini bertujuan untuk menghasilkan rancangan dan hasil akhir aplikasi berupa website sebagai bentuk visual dan memudahkan pemerintah kota untuk mengetahui komentar yang masuk sesuai dengan SKPD masing-masing. Tahapan ini terbagi menjadi empat bagian yaitu:

3.2.6.1 Perencanaan

Tahap awal dilakukan dengan menggali kebutuhan aplikasi untuk mengetahui fungsionalitas dari aplikasi ini. Tujuan dari tahapan perencanaan yaitu untuk mengetahui fitur apa saja yang akan diimplementasikan, waktu penggeraan yang dibutuhkan, hingga aplikasi dapat diuji dan digunakan.

3.2.6.2 Desain

Tahapan selanjutnya yaitu melakukan desain aplikasi dimana tahapan ini bertujuan untuk memberikan gambaran sebelum aplikasi dibuat. Desain yang dibuat meliputi :

1. Desain database
2. Desain *web-crawler*
3. Desain *user interface web*
4. Prototype aplikasi

3.2.6.3 Rancang Bangun (*Coding*)

Tahapan selanjutnya yaitu proses pembangunan aplikasi melalui kode program. Tahapan ini mencakup:

1. Rancang bangun *web crawler*

Pembuatan kode program untuk *web crawler* bertujuan untuk mengakuisisi data yang digunakan untuk proses kategorisasi. Data yang diambil berupa komentar yang masuk pada akun resmi sosial media pemerintah kota dan akun *influencer* di masing-masing kota.

2. Rancang bangun *text preprocessing*

Pembuatan kode program untuk tahapan *text preprocessing* yang terbagi menjadi stopword removal, stemming, dan document indexing. Proses pembuatan kode dibuat dengan menggunakan library Sastrawi dengan Bahasa pemrograman *Python*.

3. Rancang bangun pengkategorian komentar

Pembuatan kode program untuk tahapan pengkategorian komentar terdiri atas regular expression dengan menggunakan Bahasa pemrograman Phyton.

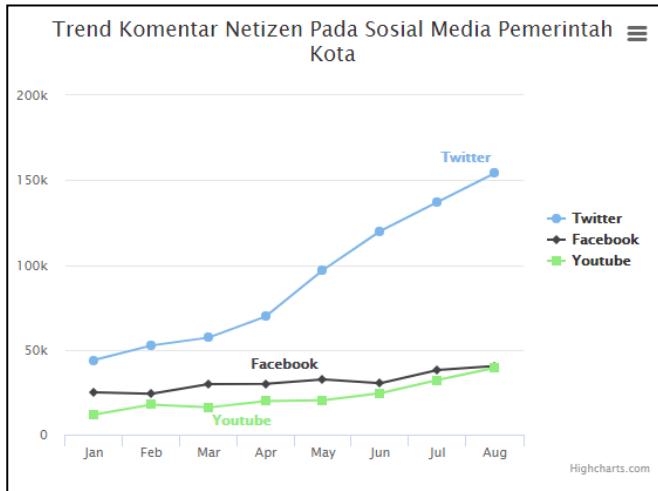
4. Rancang bangun aplikasi website

Setelah data melewati tahapan *preprocessing text* dan pengkategorian komentar maka dibuat tampilan aplikasi pada website egovbench.addi.is.its.ac.id. Bentuk aplikasi ditunjukkan untuk dua jenis akun yaitu admin dan pemerintah kota. Bagi akun admin, admin dapat melakukan pengaturan pada akun pemerintah kota seperti menambahkan akun pemerintah kota baru, menedit akun pemerintah kota, menghapus pemerintah kota, dan melihat keseluruhan akun pemerintah kota yang telah dibuat. Pada akun pemerintah kota, pemerintah kota dapat membuat akun baru, melakukan edit akun, menghapus akun, menambahkan dinas baru dan kata kunci sesuai dengan dinas, melakukan edit pada dinas, menghapus dinas, dan visualisasi kategori komentar pada akun pemerintah kota. Pada tahapan pembuatan dinas, sistem memunculkan rekomendasi kata kunci berdasarkan data komentar yang masuk sebelumnya. Dalam melakukan visualisasi data, penelitian merujuk terhadap penelitian milik Degananda Ferdian Priyambada dengan judul Rancang Bangun Aplikasi Backend Dan Web

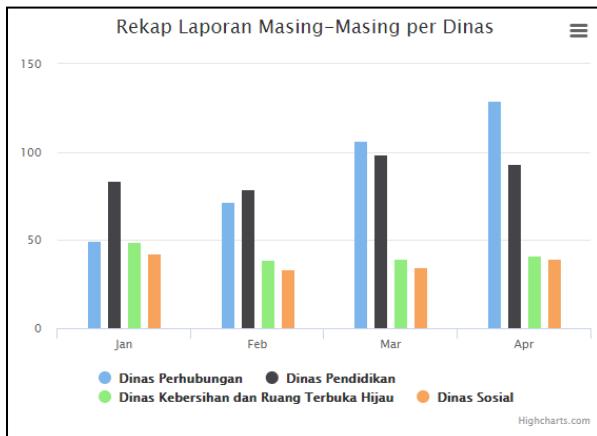
Services Penyampaian Laporan Masyarakat Berbasis Crowdsourcing Melalui Jejaring Sosial Twitter[31]. Penulis menggunakan rujukan tersebut karena penelitian tersebut memiliki kesamaan entitas seperti akun pemerintah kota, dinas, dan komentar yang ada dalam sosial media. Bentuk visualisasi terbagi atas jumlah komentar setiap bulannya untuk menunjukkan trend, jumlah komentar yang masuk berdasarkan kategori dinas dalam bentuk grafik batang, dan jumlah komentar yang masuk untuk hari ini yang telah dikategorikan. Pembuatan website dibangun menggunakan bahasa pemrograman PHP dan menggunakan *framework* Laravel.

The screenshot shows a form interface with a light gray background and a white rectangular input area. At the top left, the text "Nama Pemerintah Kota" is displayed above a dropdown menu labeled "Pilih Pemerintah Kota". Below this is a section labeled "Nama Dinas" with an input field containing the placeholder "Masukkan Nama Dinas". Underneath is a larger text area labeled "Deskripsi Dinas" with a placeholder "Masukan Deskripsi Dinas". At the bottom left is a section labeled "Kata Kunci Dinas" with an input field containing the placeholder "Masukkan Kata Kunci Dinas". At the bottom right of the input area are two buttons: a light gray "Reset" button and a dark gray "Buat" button.

Gambar 3.3 Memasukkan Dinas Pemerintah Kota



Gambar 3.4 Dashboard Jumlah Komentar Setiap Bulannya



Gambar 3.5 Dashboard Jumlah Kategori Komentar

3.2.6.4 Pengujian

Pengujian aplikasi bertujuan untuk memeriksa apakah aplikasi berjalan sesuai dengan fungsionalitas yang sudah direncanakan. Pengujian dilakukan dengan mencocokan use case yang telah

dibuat sebelum dilakukan perancangan aplikasi dengan aplikasi yang telah diimplementasi.

3.2.7 Penyusunan Tugas Akhir

Tahapan penyusunan tugas akhir dilakukan sebagai dokumentasi penggerjaan tugas akhir ini. Laporan tugas akhir dibuat berdasarkan format yang telah ditentukan. Tahapan penyusunan tugas akhir dilakukan dari awal hingga berakhirnya proses penggerjaan tugas akhir.

Halaman ini sengaja dikosongkan

BAB IV

PERANCANGAN

Pada bab ini dijelaskan terkait alur perencangan terkait beberapa hal yang diperlukan dalam pembuatan aplikasi sesuai dengan alur yang dijelaskan pada bab 3.

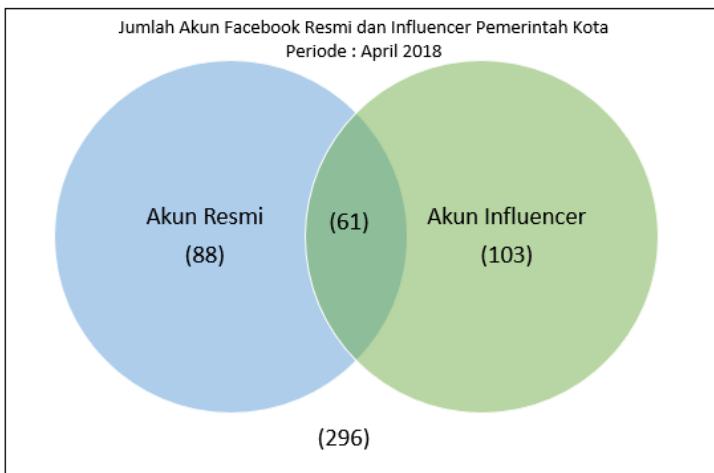
4.1 Pengambilan Data

Dalam melakukan pengambilan data, objek yang diambil merupakan komentar pada sosial media Facebook, Twitter, dan Youtube pada akun resmi pemerintah kota dan akun influencer di masing-masing daerah. Proses pengambilan data dilakukan dengan dua cara yaitu crawling dan streaming. Pengambilan data terlebih dahulu mengambil data postingan dan komentar pada ke tiga sosial media dengan menggunakan API yang tersedia yang kemudian dilakukan preprocessing data, dilanjutkan dengan pengkategorian komentar berdasarkan dinas, dan terakhir disimpan ke dalam database menggunakan MongoDB. Berikut merupakan jumlah akun sosial media resmi pemerintah kota dan influencer untuk periode April 2018:

Tabel 4.1 Tabel Jumlah Akun Resmi dan Influencer

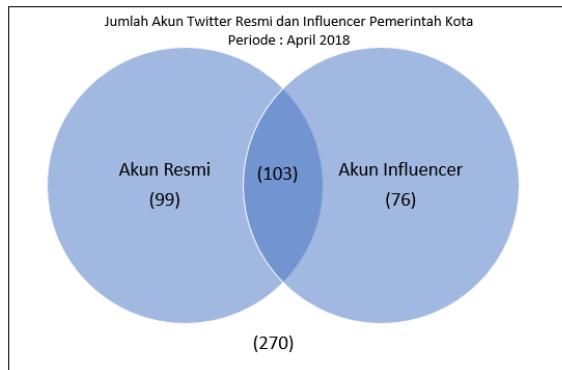
Sosial Media	Jumlah Akun Resmi	Jumlah Akun Influencer
Facebook	260	164
Twitter	297	179
Youtube	223	40

Pada sosial media Facebook, terdapat 88 provinsi/kota yang hanya memiliki akun resmi, 103 provinsi/kota yang hanya memiliki akun influencer, 61 provinsi/kota yang keduanya memiliki akun resmi dan influencer, dan 296 provinsi/kota yang tidak memiliki akun resmi dan influencer.



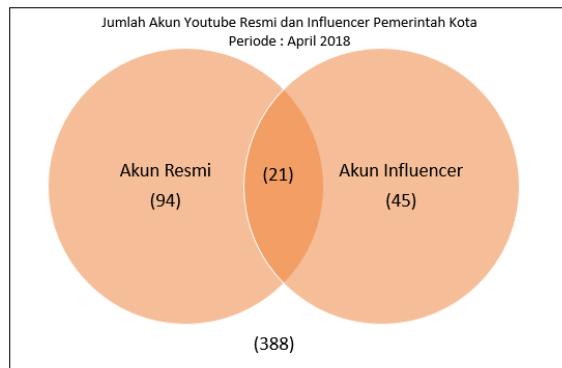
Gambar 4.1 Jumlah Akun Facebook Resmi dan Influencer Pemerintah Kota

Pada sosial media Twitter, terdapat 99 provinsi/kota yang hanya memiliki akun resmi, 76 provinsi/kota yang hanya memiliki akun influencer, 103 provinsi/kota yang keduanya memiliki akun resmi dan influencer, dan 270 provinsi/kota yang tidak memiliki akun resmi dan influencer.



Gambar 4.2 Jumlah Akun Twitter Resmi dan Influencer Pemerintah Kota

Pada sosial media Youtube, terdapat 94 provinsi/kota yang hanya memiliki akun resmi, 45 provinsi/kota yang hanya memiliki akun influencer, 21 provinsi/kota yang keduanya memiliki akun resmi dan influencer, dan 388 provinsi/kota yang tidak memiliki akun resmi dan influencer.



Gambar 4.3 Jumlah Akun Youtube Resmi dan Influencer Pemerintah Kota

Adapun atribut data yang diakuisisi pada media sosial Facebook yaitu:

Tabel 4.2 Tabel Field Facebook yang Digunakan

Field	Keterangan	Contoh
page_id	Nomor ID facebook pemda	111335038951442
page_name	Nama facebook pemda	Sapawarga Kota Surabaya
posts.post_id	Nomor ID postingan	111335038951442_164769 7335315197
posts.post_message	Isi postingan	Pemkot Surabaya dan Batam Aero Technic (BAT) Siap Jalin Kerjasama.
posts.post_type	Tipe post yang dikirim	link
posts.post_sharecount	Jumlah tautan dibagikan	0

posts.post_comment.id	Nomor ID komentar	1647697335315197_1647865441965053
posts.post_comment.message	Isi komentar	Semoga surabaya menjadi kota yang lebih baik lagi
posts.post_comment.created_time	Waktu pengiriman komentar	2018-04-05T10:51:17+0000
posts.post_commentcount	Jumlah komentar	3
posts.created_time	Waktu pengiriman post	2018-04-04 19:12:06

Atribut yang diambil pada sosial media Twitter yaitu:

Tabel 4.3 Tabel Field Twitter yang Digunakan

Field	Keterangan	Contoh
id	Nomor ID tweet	982281671130017793
text	Isi tweet	@pln_123 @SapawargaSby Lampu jalan kota di JL. TAMBANG BOYO mati... Kenapa yaa??
screenname	Nama akun	aljoki1990

favorite_count	Jumlah favorit pada tweet	0
retweet_count	Jumlah retweet pada tweet	0
created_at	Waktu pengiriman post	Fri Apr 06 15:39:34 +0000 2018
type	Tipe media tweet	

Dan atribut yang diakuisisi untuk sosial media Youtube yaitu:

Tabel 4.4 Tabel Field Youtube yang Digunakan

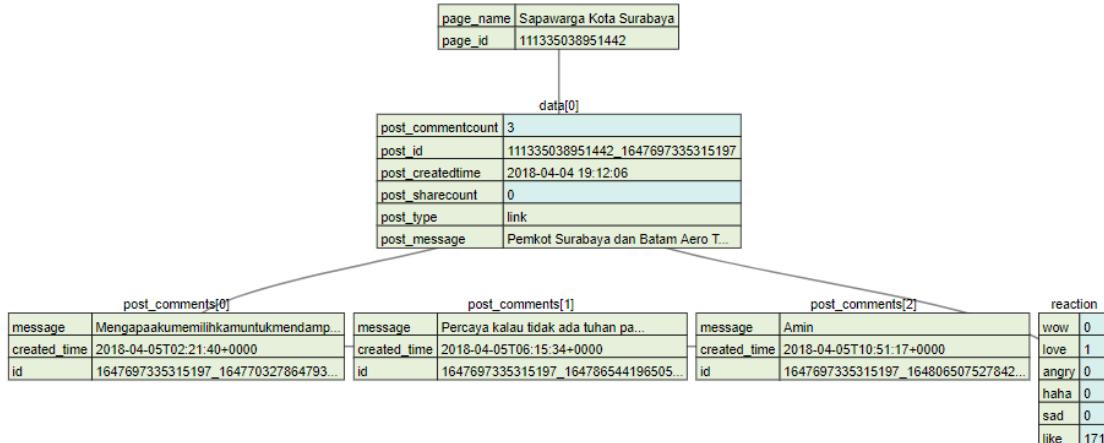
Field	Keterangan	Contoh
channel_id	Nomor ID channel	UCj8YSC3qXItFmHuiYU2Afqg
channel_name	Nama youtube pemda	Pemkot Bekasi
video.video_id	Nomor ID video	0FGWCyJcddQ
video.video_info.video_title	Judul video	Walikota Bekasi Dr Rahmat Effendi : Explore Potensi Curug Parigi untuk

		Wujudkan #BekasiHijau
video.video_info.video_statis tic.likecount	Jumlah like video	15
video.video_info.video_statis tic.commentcount	Jumlah komentar video	7
video.video_info.video_statis tic.dislikecount	Jumlah dislike video	2
video.video_comments.comment_date	Waktu pengiriman komentar	2017-05-15T17:01:37.000Z
video.video_comments.comment_message	Isi komentar	Bisanya ngomong doang

Keterangan:

 : Merupakan atribut yang digunakan pada penelitian ini

Berikut merupakan visualisasi untuk struktur data yang diakuisisi melalui sosial media Facebook:



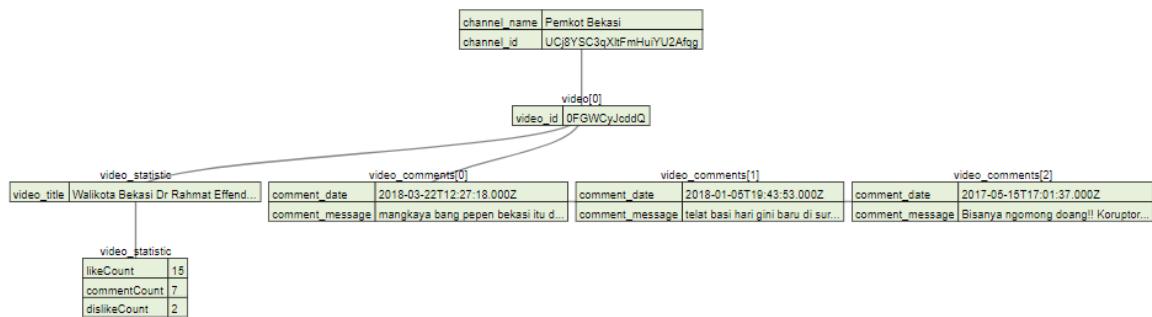
Gambar 4.4 Tree Diagram Facebook

Berikut merupakan visualisasi untuk struktur data yang diakuisisi melalui sosial media Twitter:

id	982281671130017800
text	@pln_123 @SapawargaSby Lampu jal...
screenname	aljoki1990
favorite_count	0
retweet_count	0
created_at	Fri Apr 06 15:39:34 +0000 2018
type	

Gambar 4.5 Tree Diagram Twitter

Berikut merupakan visualisasi untuk struktur data yang diakuisisi melalui sosial media Youtube:



Gambar 4.6 Tree Diagram Youtube

4.2 Desain Database

Database digunakan sebagai tempat penyimpanan akuisisi data dan hasil kategorisasi. Database dibedakan menjadi empat database yaitu listpemda, facebook_comments, twitter_comments, dan youtube_comments.

4.2.1 Database Pemda

Database ini berisikan data-data yang berkaitan dengan identitas pemerintah daerah seperti id dan nama pemerintah daerah. Selain identitas pemerintah daerah, database ini berisikan username facebook resmi dan influencer, username twitter resmi dan influencer, dan username youtube resmi dan influencer.

Table 4.5 Tabel Collection Pemerintah Daerah

Field	Tipe Data	Keterangan
_id	Integer	Nomor ID komentar
name	String	Nama Pemerintah Daerah
facebook_resmi	String	Nama akun facebook resmi
facebook_influencer	String	Nama akun facebook influencer
twitter_resmi	String	Nama akun twitter resmi
twitter_influencer	String	Nama akun twitter influencer
youtube_resmi	String	Nama akun youtube resmi

youtube_influencer	String	Nama akun youtube influencer
--------------------	--------	------------------------------

4.2.2 Database Facebook Comments

Database Facebook Comments berisikan jumlah komentar facebook akun resmi dan influencer yang telah terkategorisasi. Penyimpanan data komentar facebook pada database Facebook Comments bertujuan untuk proses visualisasi. Berikut merupakan field yang disimpan dalam facebook comments

Table 4.6 Tabel Collection Facebook Comments

Field	Tipe Data	Keterangan
_id	String	ID komentar facebook
page_id	String	Nama halaman akun
comment_createdDate	String	Waktu pengiriman komentar
comment_message	String	Isi pesan komentar
class	String	Hasil tipe klasifikasi
category	String	Hasil tipe kategori dinas

Pada penelitian ini, data yang akan divisualisasikan adalah jumlah setiap *category* di masing-masing pemerintah daerah untuk komentar pada Facebook yang berisikan label berdasarkan dinas pemerintah daerah.

4.2.3 Database Twitter Comments

Database twitter comments berisikan tweet yang masuk ke dalam akun resmi dan influencer twitter pemerintah daerah. Tabel 4.7 menunjukkan desain database twitter comments

Table 4.7 Tabel Collection Twitter Comments

Field	Tipe Data	Keterangan
_id	String	Nomor ID komentar
account_id	String	Nama akun
tweet_created Date	String	Waktu pengiriman komentar
text	String	Isi pesan komentar
class	String	Hasil tipe klasifikasi
category	String	Hasil tipe kategori dinas

Pada penelitian ini, data yang akan divisualisasikan adalah jumlah setiap *category* di masing-masing pemerintah daerah untuk komentar pada Twitter yang berisikan label berdasarkan dinas pemerintah daerah.

4.2.4 Database Youtube Comments

Database youtube comments berisikan komentar yang masuk pada setiap video di akun resmi dan influencer youtube pemerintah daerah. Tabel 4.8 menunjukkan desain database youtube comments

Table 4.8 Tabel Collection Youtube Comments

Field	Tipe Data	Keterangan
_id	String	ID komentar youtube

channel_id	String	ID channel
comment_createdDate	String	Waktu pengiriman komentar
comment_message	String	Isi pesan komentar
class	String	Hasil tipe klasifikasi
category	String	Hasil tipe kategori dinas

Pada penelitian ini, data yang akan divisualisasikan adalah jumlah setiap *category* di masing-masing pemerintah daerah untuk komentar pada Youtube yang berisikan label berdasarkan dinas pemerintah daerah.

4.3 Metodologi Implementasi

Metodologi implementasi dalam penelitian ini merupakan tahapan yang bertujuan untuk menyelesaikan penelitian dengan komputasi secara otomatis. Tahapan ini terbagi menjadi 5 tahapan yaitu

4.3.1 Streaming Data

Streaming data merupakan tahapan pengambilan data postingan dan komentar pada akun sosial media resmi dan influencer pemerintah daerah secara langsung. Tahapan streaming ini menggunakan python dan API di masing-masing sosial media untuk mendapatkan JSON yang berisikan data komentar. Proses pengambilan data secara streaming akan dimulai ketika sistem aplikasi pertama kali berjalan.

4.3.2 Pra-proses Data

Tahapan pra-proses data mencakup beberapa langkah penggerjaan yaitu pembersihan data dari tag atau karakter-karakter tertentu, pengubahan data menjadi huruf kecil casefolding, stemming, stopword removal, serta tokenization.

1. Cleansing merupakan tahapan untuk menghapus karakter-karakter yang tidak terpakai pada komentar seperti angka, symbol, dan beberapa row yang kosong.
2. Casefolding merupakan tahapan untuk menyeragamkan kata-kata menjadi huruf kecil.
3. Stemming merupakan tahapan untuk menghilangkan kata imbuhan dan akhiran menjadi kata dasar dengan memanfaatkan modul Sastrawi.
4. Tokenization merupakan tahapan untuk memecah atau memotong kalimat menjadi kata. Proses tokenization menggunakan modul scikit.
5. Stopwords removal merupakan tahapan untuk menghilangkan kata-kata yang sering muncul namun tidak penting. Proses tokenization menggunakan modul Sastrawi.

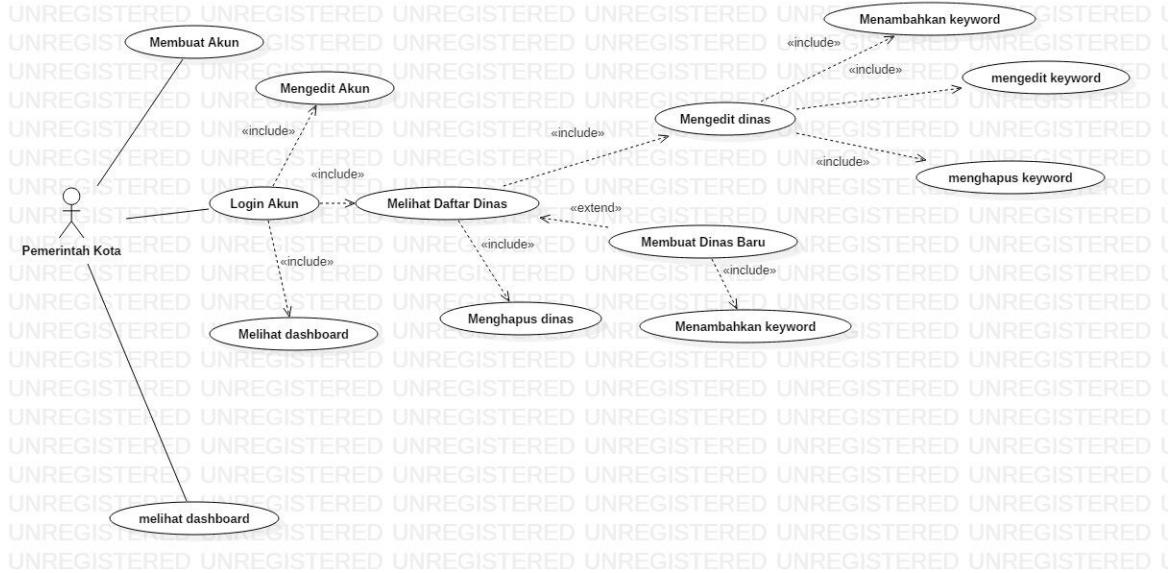
4.3.3 Pengkategorian Data

Pada tahapan ini dilakukan pengakategorian data berdasarkan model yang telah dibuat. Data streaming yang masuk melalui Apache Kafka akan disalurkan kepada Apache Spark untuk dilakukan proses kategorisasi. Hasil dari kategorisasi data selanjutnya akan disimpan ke dalam MongoDB.

4.3.4 Rekomendasi Keyword oleh Aplikasi

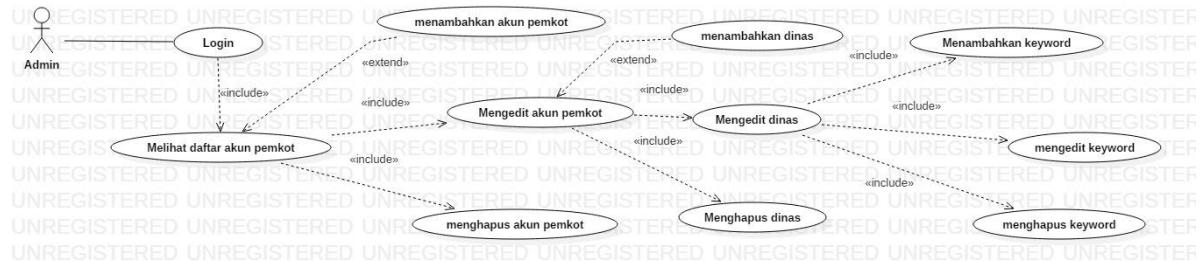
Aplikasi akan memberikan rekomendasi keyword berdasarkan komentar yang telah diberi label berdasarkan SKPD. Rekomendasi keyword diberikan berdasarkan kata yang memiliki nilai IDF paling rendah. Kata yang memiliki nilai IDF paling rendah nantinya akan dimasukkan ke dalam database agar dapat ditampilkan kepada user.

Sebelum dilakukan pembuatan aplikasi, terlebih dahulu dilakukan penggambaran kelakukan (*behavior*) aplikasi yang akan dibuat. Dalam melakukan penggambaran kelakukan aplikasi digunakan diagram use case yang mendeskripsikan interaksi antara satu atau lebih aktor dengan aplikasi yang akan dibuat. Dalam penelitian ini terdapat dua aktor yaitu pemerintah kota dan admin. Berikut merupakan use case diagram untuk aktor pemerintah kota:



Gambar 4.7 Use Case untuk Akun Pemerintah Kota

Sedangkan berikut merupakan use case diagram untuk aktor admin:



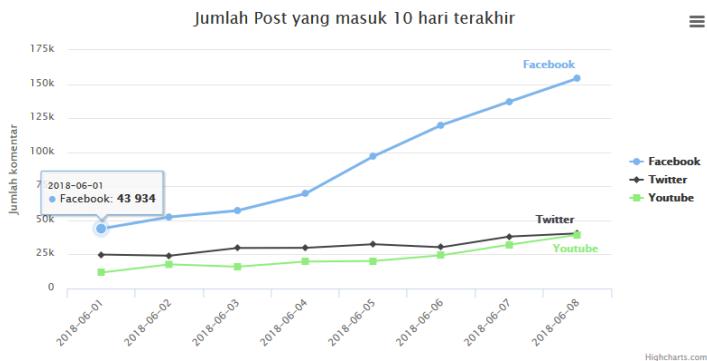
Gambar 4.8 Use Case untuk Admin

4.3.5 Desain antarmuka aplikasi visualisasi

Melalui model yang didapatkan serta data streaming yang sudah dikategorikan bedasarkan SKPD, maka langkah selanjutnya yaitu menampilkan data dalam bentuk visualisasi. Visualisasi data yang digunakan dalam penelitian ini menggunakan bahasa pemrograman PHP. Rancangan visualisasi yang akan ditampilkan dibedakan menjadi beberapa konten yaitu jumlah komentar yang masuk untuk hari ini, jumlah komentar yang masuk dalam 10 hari terakhir, dan jumlah komentar yang telah terkategorisasi dalam bentuk grafik batang. Gambar 4.3 menjelaskan tentang jumlah komentar yang masuk pada tanggal har ini, gambar 4.4 menjelaskan jumlah komentar pada 10 hari terakhir, gambar 4.5 menjelaskan jumlah komentar berdasarkan kategori SKPD, gambar 4.6 menjelaskan tampilan pemerintah kota membuat dinas, dan gambar 4.7 menjelaskan tampilan sistem memberikan rekomendasi keyword kepada pemerintah kota.

Daily Stats	
Tanggal Hari Ini	2018-06-13
Total Pemerintah Daerah	548
Total Facebook Resmi / Influencer	148 / 155
Total Twitter Resmi / Influencer	201 / 175
Total Youtube Resmi / Influencer	113 / 59
Total Post Masuk	0
Total Komentar Masuk	0
Total Post Terklasifikasi	0
Total Komentar Terklasifikasi	0
Total Komentar Terkategorisasi	6
Total Komentar Tidak Terkategorisasi	1
Total Komentar Terduplikasi	1
Rata-Rata Komentar per Pemda	0.01%

Gambar 4.9 Jumlah Komentar yang Masuk Pada Hari Ini



Gambar 4.10 Jumlah Komentar yang Masuk Pada Periode Tertentu

Nama Pemerintah Kota

Pemerintah Kota

Nama Dinas

Masukkan nama dinas tanpa singkatan

Deskripsi Dinas

Masukan deksripsi untuk dinas

Keyword

Masukan 1 kata kunci untuk dinas



Cancel

Tambah

Gambar 4.11 Tampilan User Ketika Memasukkan Dinas dan Keyword

Nama Dinas	Keyword	Rekomendasi Keyword	Action	
Dinas Perhubungan	jalan	macet	Tambah Rekomendasi	Hapus Rekomendasi
Dinas Kesehatan	puskesmas	sehat	Tambah Rekomendasi	Hapus Rekomendasi
Dinas Pemadam Kebakaran	api	padam	Tambah Rekomendasi	Hapus Rekomendasi

Gambar 4.12 Tampilan Aplikasi Menampilkan Rekomendasi Kepada User

BAB V

IMPLEMENTASI

Bab ini menjelaskan terkait proses implementasi pada perancangan perangkat lunak yang dibangun. Pada bagian ini akan dijelaskan bagaimana lingkungan diimplementasi, pembuatan fitur-fitur aplikasi, dan proses pengujian aplikasi.

5.1 Lingkungan Implementasi

Pada bagian ini dibahas mengenai lingkungan implementasi yang digunakan dalam implementasi tugas akhir baik perangkat keras maupun perangkat lunak. Tabel 5.1. menjelaskan spesifikasi perangkat keras untuk implementasi tugas akhir ini. Sementara tabel 5.2. menjelaskan spesifikasi perangkat lunak untuk implementasi tugas akhir.

Tabel 5.1 Tabel Spesifikasi Perangkat Keras yang Digunakan

Perangkat	Spesifikasi
Processor	Intel(R) Core(TM) i5-5200U CPU @ 2.20GHz
RAM	8 GB
Hard Disk Drive	500GB

Tabel 5.2 Tabel Perangkat Lunak yang Digunakan

Nama Perangkat Lunak	Kegunaan dalam implementasi
PHP 7.2.5	Webserver
Atom 1.2.7	Editor
MongoDB	Database
Apache Kafka	Data aggregator
Apache Spark	Data processing
Google Chrome	Web browser

5.2 Pembuatan aplikasi

Pada bagian ini akan dijelaskan mengenai penerapan desain aplikasi dalam bentuk *source code* yang menggunakan bahasa pemrograman PHP dan python. Adapun bagian ini akan membahas *crawling data* dari social media pemerintah daerah, fungsi-fungsi pada admin dan user, pengkategorian komentar, dan visualisasi data yang ditampilkan dalam bentuk grafik dan tabel.

5.2.1 Streaming Data dari Social Media

Streaming pada social media pemerintah daerah digunakan untuk mengambil postingan dan komentar kepada pemerintah daerah itu sendiri. *Library* yang terlibat dalam proses crawling ini adalah PHP Graph SDK dalam proses *crawling* Facebook pemerintah daerah, Twitter API untuk proses *crawling* Twitter pemerintah daerah, dan Google API untuk proses *crawling* Youtube pemerintah daerah.

Dalam melakukan proses streaming digunakan Apache Kafka sebagai data pipeline khususnya untuk data streaming. Proses konfigurasi dapat dilihat pada penelitian Guntur Kondang Prakoso[32]

Proses yang dilakukan dalam proses streaming ini adalah *streaming* social media, pemisahan hasil streaming postingan dan komentar, pengolahan data, dan penyimpanan ke dalam database.

5.2.1.1 Streaming Social Media Pemerintah Daerah

Proses pertama yang dilakukan yaitu melakukan *streaming* terhadap social media pemerintah daerah di Indonesia. Sosial media yang dilakukan *streaming* yaitu Facebook, Twitter, dan Youtube.

a. Streaming Parameter Facebook

Pada proses *streaming* parameter Facebook yang dilakukan adalah melakukan pengambilan data menggunakan *library* dari Facebook API library untuk mengambil *fields* yang dibutuhkan. Proses streaming merupakan proses *crawling* yang dilakukan

secara berulang karena Facebook tidak menyediakan API untuk melakukan streaming data. Data komentar dapat dilakukan pengambilan apabila data post telah didapatkan terlebih dahulu. Setelah proses crawling selesai maka crawling akan mengulang pada akun pertama. Proses streaming melibatkan apache kafka sehingga sebelum diteruskan menuju kafka, perlu dilakukan parsing pada data JSON hasil crawling agar mendapatkan data komentar.

```
1. def scrapeFacebookPageFeedStatus(page_id, access
   _token, since_date, until_date, number, pemdanam
   e, type):
2.     has_next_page = True
3.     num_processed = 0
4.     scrape_starttime = datetime.datetime.now
()
5.     after = ''
6.     base = "https://graph.facebook.com/v2.12
"
7.     node = "/{}".format(page_id)
8.     parameters = "?access_token={}&fields=id
,name,fan_count,posts.limit({})".format(access_t
oken, 100)
9.     since = ".since({})".format(since_date)
if since_date \
10.      is not '' else ''
11.     until = ".until({})".format(until_date)
if until_date \
12.      is not '' else ''
13.
14.     print("Scraping No.{0} :{1}\n Pemda : {2}\n T
ype : {3}\n Page ID : {4}\n at : {5}\n".format(num
ber, pemdaname, type, page_id, scrape_starttime))
15.
16.     complete_dict = {}
17.     complete_dict['page_id_name'] = page_id
18.     complete_dict['post'] = []
19.
20.     base_url = base + node + parameters + si
nce + until
21.
```

```
22.         url = getFacebookPageFeedUrl(base_url)
23.
24.         j_input = json.loads(request_until_succe
ed(url).decode())
25.
26.         complete_dict['page_id_number'] = j_inpu
t['id']
27.         complete_dict['page_name'] = j_input['na
me']
28.         complete_dict['page_fanCount'] = j_input
['fan_count']
29.
30.         print(' Page ID (number) : {} \n Page nam
e : {} \n Total fans : {} \n'.format(complete_dict
['page_id_number'], complete_dict['page_name'],
complete_dict['page_fanCount']))
31.
32.         while has_next_page:
33.
34.             after = '' if after is '' else ".aft
er({})".format(after)
35.             base_url = base + node + parameters
+ since + until + after
36.
37.             url = getFacebookPageFeedUrl(base_ur
l)
38.
39.             j_input = json.loads(request_until_s
ucceed(url).decode())
40.
41.             if 'posts' in j_input:
42.                 after = j_input['posts']['paging
']['cursors']['after']
43.                 comments = getCommentsForStatuse
s(base_url)
44.
45.                 for status in j_input['posts']['
data']:
46.
47.                     status_data = processFaceboo
kPageFeedStatus(status)
48.                     status_data['post_comments']
= comments[status_data['post_id']]
49.                     complete_dict['post'].append
(status_data)
```

```

50.
51.                     num_processed += 1
52.                     if num_processed % 100 == 0:
53.                         print("{} Statuses Proce
      ssed: {}".format
54.                               (num_processed, da
      tetime.datetime.now()))
55.
56.                     else:
57.                         has_next_page = False
58.
59.                     print("\nDone!\n{} Statuses Processed in
      {}.".format(
60.                           num_processed, datetime.datetime.n
      ow() - scrape_starttime))
61.
62.                     return complete_dict

```

Kode 5.1 Method ScrapeFacebookPageFeedStatus

Kode program diatas merupakan *method* `scrapeFacebookPageFeedStatus` yang bertujuan untuk mendapatkan post dari halaman facebook suatu pemerintah daerah. Dalam melakukan pengambilan post dari halaman facebook dibutuhkan request URL serta field yang dibutuhkan antara lain id, name, dan posts. Parameter id digunakan untuk mendapatkan `page_id` halaman facebook, name bertujuan untuk mendapatkan nama akun dari halaman facebook dan posts bertujuan untuk mendapatkan data post yang nantinya digunakan untuk mendapatkan data komentar. Data post yang telah didapatkan dibuat sebagai dictionary agar dapat dilakukan pengolahan data di tahap selanjutnya. Pada *method* ini juga dilakukan pemanggilan *method* `getCommentForStatuses` yang bertujuan untuk mengambil komentar pada post tersebut.

```

1. def getCommentsForStatuses(base_url):
2.
3.     comments_dict = {}
4.
5.     fields = "{id,comments}"

```

```
6.     url = base_url + fields
7.
8.     data = json.loads(request_until_succeed(url)
9.                 .decode())['posts']['data']
10.    for status in data:
11.        id = status['id']
12.        komentar = []
13.
14.        if 'comments' in status:
15.            komentar = status['comments']['data']
16.        paging = status['comments']['paging']
17.
18.        if 'next' in paging:
19.            nextcommentpage = status['comments']['paging']['next']
20.
21.            comments_has_next_page = True
22.
23.            while comments_has_next_page:
24.                nextcommentpage = '' if nextcommentpage is '' else "{}".format(nextcommentpage)
25.
26.                crawlnextpage = json.loads(request_until_succeed(nextcommentpage).decode())
27.
28.                komentar = komentar + crawlnextpage['data']
29.
30.                if crawlnextpage['data']:
31.                    if 'next' in crawlnextpage['paging']:
32.                        nextcommentpage = crawlnextpage['paging']['next']
33.                    else:
34.                        comments_has_next_page = False
35.                else:
36.                    comments_has_next_page = False
37.            comments_dict[id] = komentar
```

```
38.     return comments_dict
```

Kode 5.2 Method getCommentsForStatuses

Kode diatas merupakan *method* getCommentsForStatuses yang bertujuan untuk mendapatkan komentar sesuai post. Dalam melakukan pengambilan komentar dari halaman facebook dibutuhkan request URL serta field yang dibutuhkan antara lain id dan comments. Parameter id digunakan untuk mendapatkan id dari suatu posts dan comments bertujuan untuk mendapatkan data komentar dalam suatu post. Data komentar akan dimasukkan sebagai dictionary untuk memudahkan proses pengolahan data dan kepemilikan data komentar dalam JSON.

```
1. if __name__ == '__main__':
2.
3.     start_row = 1
4.
5.     namapemda = spreadsheet.scrapingcolumn(1, st
art_row, None)
6.     idpemda = spreadsheet.scrapingcolumn(3, star
t_row, None)
7.     idpagepemda = spreadsheet.scrapingcolumn(7,
start_row, -1)
8.     idpageinfluencer = spreadsheet.scrapingcolum
n(10, start_row, -1)
9.
10.    since_date = "2018-05-01"
11.    until_date = ""
12.
13.    access_token = <Access Token Facebook>
14.
15.    for satuanidpemda in idpemda[298:]:
16.
17.        pemda_dict = {}
18.        pemda_dict['pemda_id'] = satuanidpemda
19.        pemda_dict['pemda_name'] = namapemda[idp
emda.index(satuanidpemda)]
20.        pemda_dict['resmi'] = {}
21.        pemda_dict['influencer'] = {}
22.
```

```
23.         satuanidpagepemda = idpagepemda[idpemda.
   index(satuanidpemda)]
24.         satuanidinfluencer = idpageinfluencer[id
   pemda.index(satuanidpemda)]
25.
26.         if satuanidpagepemda is not '':
27.             pemda_dict['resmi'] = scrapeFacebook
   PageFeedStatus(satuanidpagepemda, access_token,
   since_date, until_date, satuanidpemda, pemda_dict
   ['pemda_name'], 'resmi')
28.
29.         elif satuanidpagepemda is '':
30.             pemda_dict['resmi']['post'] = []
31.             pemda_dict['resmi']['post'].append({
   'post_comments': []})
32.
33.         if satuanidinfluencer is not '':
34.             pemda_dict['influencer'] = scrapeFac
   ebookPageFeedStatus(satuanidinfluencer, access_t
   oken, since_date, until_date, satuanidpemda, pem
   da_dict['pemda_name'], 'influencer')
35.
36.         elif satuanidinfluencer is '':
37.             pemda_dict['influencer']['post'] = [
   ]
38.             pemda_dict['influencer']['post'].app
   end({'post_comments': []})
39.
40.         if satuanidpemda is idpemda[0]:
41.             exported = '[' + str(json.dumps(pemda_
   dict, indent=4)) + ','
42.
43.             with open('facebook_lengkap_mei2222.
   json', 'a') as file:
44.                 file.write(exported)
45.
46.         elif satuanidpemda is not idpemda[-1]:
47.             exported = str(json.dumps(pemda_dict
   , indent=4)) + ','
48.
49.             with open('facebook_lengkap_mei2222.
   json', 'a') as file:
50.                 file.write(exported)
51.
52.         else:
```

```

53.             exported = str(json.dumps(pemda_dict
54.                               , indent=4)) + ']'
55.             with open('facebook_lengkap_mei2222.
56.                           json', 'a') as file:
57.                             file.write(exported)

```

Kode 5.3 Method Main

Kode program diatas merupakan potongan kode untuk *method main*. Akun pemerintah daerah dan influencer dimuat sebagai list yang sebelumnya sudah terdaftar melalui *google spreadsheets*. Crawler akan berjalan secara berulang sejumlah akun facebook resmi dan influencer yang terdaftar. Pada *method* ini juga dilakukan pengecekan akun pemerintah daerah yang memiliki akun resmi namun tidak memiliki akun influencer dan begitu juga sebaliknya. Data akhir dalam proses crawling ini yaitu JSON lengkap keseluruhan pemerintah daerah.

```

1. def getCommentDocument(self, complete_dict):
2.
3.     commentdocument = complete_dict['comment
4.     []']
5.     for comment in commentdocument:
6.         comment['_id'] = comment[self.commen
7.         t_id_key]
8.         comment.pop(self.comment_id_key, Non
9. e)
10.        comment[self.account_id_key] = compl
11. ete_dict['account'][self.account_id_key].lower()
12.
13.    return commentdocument

```

Kode 5.4 Method getCommentDocument

Kode diatas merupakan potongan kode untuk *method getCommentDocument* untuk melakukan parsing JSON masing-masing komentar sebelum dikirimkan melalui kafka. Parameter yang digunakan yaitu *_id* dan *self.account_id_key*. Parameter *_id* digunakan untuk mendapatkan id suatu komentar

sementara self.account_id_key digunakan untuk menyetarakan account id menjadi huruf kecil.

```

1. def send_message(self, dict_or_json):
2.
3.     producer = self.topic.get_producer(
4.         serializer=self.serializer_method,
5.         sync=True,
6.         max_request_size=1024 * 1024 * 10,
7.         use_rdkafka=True,
8.         delivery_reports=False,
9.     )
10.
11.    with producer as producer:
12.        try:
13.            producer.produce(dict_or_json)
14.        except Exception as e:
15.            logging.warning(e)
16.
17.    latest_available_offsets = json.dumps(self.t
opic.latest_available_offsets())
18.    logging.debug('Latest available offsets: '
+ latest_available_offsets)

```

Kode 5.5 Method send_message

Kode diatas merupakan potongan kode untuk *method* send_message yang bertujuan mengaktifkan producer pada kafka yang akan mengirimkan data komentar.

```

1. def pushCommentDocument(self, complete_dict):
2.
3.     comment_document = self.yp.getCommentDoc
ument(complete_dict)
4.
5.     for comment in comment_document:
6.         self.ymc.updateComment(comment_docum
ent)
7.         self.ykc.send_message(comment)

```

Kode 5.6 Method pushCommentDocument

Kode diatas merupakan potongan kode untuk *method* pushCommentDocument yang bertujuan untuk memanggil *method* getCommentDocument dan send_message yang seacra berulang berdasarkan jumlah komentar. Hasil komentar JSON Facebook dapat dilihat pada Tabel 4.2 Tabel Field Facebook yang Digunakan

b. Streaming Parameter Twitter

Proses streaming Twitter berbeda dengan Facebook karena Twitter API menyediakan proses streaming data. Proses streaming data turut menggunakan Kafka sebagai pipeline data. Proses pengambilan data menggunakan library Twitter API untuk mengambil fields yang diperlukan. Tweet komentar yang diambil merupakan tweet balasan yang dikirimkan oleh netizen kepada akun twitter resmi dan influencer pemerintah kota.

```

1. class StdOutListener(StreamListener):
2.     def on_data(self, data):
3.         producer.send_messages("twitter-stream-
   reply", data.encode('utf-8'))
4.         print(data)
5.         return True
6.
7.     def on_error(self, status):
8.         print(status)

```

Kode 5.7 Method StdOutListener

Kode diatas merupakan potongan kode untuk *method* StdOutListener yang bertujuan untuk mengaktifkan producer untuk mengirimkan data komentar.

```

1. def ambil_list():
2.     lists = collection_pemda.find({})
3.     for pemda in list(lists):
4.         if pemda['twitter_resmi'] != '':
5.             list_accountID.append(pemda['twitter-
   resmi'])
6.         if pemda['twitter_influencer'] != '':

```

```

7.         list_accountID.append(pemda['twitter
     _influencer']))
8.
9.     return list_accountID

```

Kode 5.8 Method ambil_list

Kode diatas merupakan potongan kode untuk *method ambil_list* yang bertujuan untuk membuat list yang berisikan nama username twitter resmi dan twitter influencer.

```

1. if __name__ == '__main__':
2.     #Twitter token
3.     access_token = <Twitter Access Token>
4.     access_token_secret = <Twitter Access Token
     Secret>
5.     consumer_key = <Twitter Consumer Key>
6.     consumer_secret = <Twitter Consumer
     Secret>
7.
8.     #Mongo Client
9.     client = pymongo.MongoClient()
10.    database = client['egovbench']
11.    collection = database['list_pemda']
12.
13.    #Setup kafka client
14.    kafka = KafkaClient("localhost:9092")
15.    #Kafka simple producer
16.    producer = SimpleProducer(kafka)
17.    l = StdOutListener()
18.    #Twitter auth
19.    auth = OAuthHandler(consumer_key, consumer_s
     ecret)
20.    auth.set_access_token(access_token, access_t
     oken_secret)
21.    #Create stream api
22.    stream = Stream(auth, l, wait_on_rate_limit=
     True, wait_on_rate_limit_notify=True)
23.
24.    while True:
25.        try:
26.            print("Stream started")
27.            stream.filter(track=ambil_list())

```

```

28.         except Exception as e:
29.             print("Stream stopped")
30.             print("type error: " + str(e))

```

Kode 5.9 Method Main

Kode diatas merupakan potongan kode untuk *main method* yang bertujuan untuk memanggil *method-method* sebelumnya dan melakukan streaming tweet komentar. Method ambil_list berfungsi untuk mendapatkan keseluruhan akun twitter dalam bentuk list. Selain itu fungsi stream menggunakan twitter API yang ada dalam method OAuthHandler yang dibantu dengan library tweepy dan dikirimkan melalui kafka. Proses streaming dijalankan secara terus menerus dengan dilakukan pengecekan error maka streamer akan menjalankan lagi pengambilan data. Data yang dihasilkan dari proses streaming berupa JSON keseluruhan tweet komentar pemerintah daerah. Hasil komentar JSON Twitter dapat dilihat pada Tabel 4.3 Tabel Field Twitter yang Digunakan

c. Streaming Parameter Youtube

Proses streaming Youtube sama seperti proses streaming pada Facebook. Hal ini dikarenakan Youtube tidak menyediakan API untuk melakukan proses streaming. Proses streaming dilakukan dengan melakukan pengulangan pada crawler untuk mendapatkan komentar pada Youtube.

Proses crawling parameter Youtube menggunakan library Youtube API untuk mengambil fields yang diperlukan. Sama seperti Facebook, Youtube membutuhkan induk video untuk mendapatkan komentar pada suatu video. Pada Youtube, terdapat dua jenis username yaitu id channel dan username. Sehingga perlu dilakukan pengkonversian nama username menjadi id channel.

```

1. def channelusernameid(idchannel):
2.     parameters = {"part": "snippet",
3.                   "forUsername": idchannel,
4.                   "key": api_key,

```

```

5.          "fields": "items(id,snippet/
6.          title)"}
7.          url = "https://www.googleapis.com/youtube/
8.          v3/channels"
9.          usernameconverter = {}
10.         page = requests.request(method='get', url=
11.         url, params=parameters)
12.         j_results = json.loads(page.text)
13.         usernameconverter['title'] = j_results['it
14.         ems'][0]['snippet']['title']
15.         usernameconverter['id'] = j_results['items
16.         '][0]['id']
17.     return usernameconverter

```

Kode 5.10 method channel usernametoid

Kode diatas merupakan potongan kode untuk *method* channel usernametoid yang bertujuan untuk mengubah username menjadi channel id. Parameter yang digunakan dalam method ini yaitu id dan title. Parameter id digunakan untuk mendapatkan id suatu channel dan title digunakan untuk mendapatkan nama dari channel tersebut. Hal ini dilakukan karena untuk mendapatkan data komentar dibutuhkan channel id pada Youtube API.

```

1. def channelnameforfilename(idchannel):
2.     parameters = {"part": "snippet",
3.                    "id": idchannel,
4.                    "key": api_key,
5.                    "fields": "items/snippet/tit
6.                    le"}
7.     url = "https://www.googleapis.com/youtube/
8.     v3/channels"
9.     page = requests.request(method='get', url=
10.     url, params=parameters)
11.    j_results = json.loads(page.text)
12.    if j_results['items']:

```

```

13.         idconverter['title'] = j_results['it
   ems'][0]['snippet']['title']
14.         idconverter['id'] = idchannel
15.     else:
16.         idconverter = channelusername2oid(id
   channel)
17.     return idconverter

```

Kode 5.11 Method channelnameforfilename

Kode diatas merupakan potongan kode untuk *method* channelnameforfilename yang bertujuan untuk mendapatkan identitas suatu channel berupa id channel dan nama channel. Selain itu pada *method* ini dilakukan pengecekan kembali apakah username sudah dilakukan perubahan menjadi channel id atau tidak.

```

1. def crawlvideo(api_key, channelId, pemdaID, pemd
   aName, channelType):
2.     temp = channelnameforfilename(channelId)
3.
4.     channelName = temp['title']
5.     idchannel = temp['id']
6.
7.     parameters = {"part": "snippet",
8.                     "channelId": idchannel,
9.                     "maxResults": 50,
10.                    "publishedAfter": "2018-05-
   01T00:00:00Z",
11.                   "key": api_key,
12.                   "type": "video",
13.                   "fields": "items(id/videoId,snip
   pet/channelTitle),nextPageToken"}
14.     url = "https://www.googleapis.com/youtube/v3/s
   earch"
15.     video_search_nextpage = True
16.     num_processed = 0
17.     scrape_starttime = datetime.datetime.now()
18.
19.     print("\nScraping start for pemda no. : {} ({})
   ) !\nYoutube Channel: {}\\nType: {}\\nat: {}\\n".fo
   rmat(pemdaID, pemdaName, channelName, channelTyp
   e, scrape_starttime))

```

```
20.
21.     complete_dict = {}
22.     complete_dict['channel_id'] = parameters['chan
    nelId']
23.     complete_dict['video'] = []
24.
25.     while video_search_nextpage:
26.         page = requests.request(method='get', ur
l=url, params=parameters)
27.         j_results = json.loads(page.text)
28.
29.         if j_results['items']:
30.
31.             for video in j_results['items']:
32.
33.                 complete_dict['channel_name'
] = video['snippet']['channelTitle']
34.
35.                 complete_video = {}
36.
37.                 complete_video['video_id'] =
    video['id']['videoId']
38.
39.                 returned_statistics = {}
40.                 returned_statistics = crawls
    tatics(video['id']['videoId'])
41.
42.                 complete_video['video_title'
] = returned_statistics[complete_video['video_id
']][['video_title']]
43.                 complete_video['video_date']
    = returned_statistics[complete_video['video_id
']][['video_date']]
44.
45.                 statistics = returned_statis
    tics[complete_video['video_id']]['video_statisti
    c']
46.                 complete_video['video_dislik
    eCount'] = int(statistics['dislikeCount']) if 'd
    islikeCount' in statistics else 0
47.                 complete_video['video_likeCo
    unt'] = int(statistics['likeCount']) if 'likeCo
    nt' in statistics else 0
```

```

48.           complete_video['video_viewCo
unt'] = int(statistics['viewCount']) if 'viewCou
nt' in statistics else 0
49.           complete_video['video_commen
tCount'] = int(statistics['commentCount']) if 'c
ommentCount' in statistics else 0
50.
51.           returned_comments = {}
52.           returned_comments = crawlcom
ments(video['id']['videoId'])
53.           complete_video['video_commen
ts'] = returned_comments
54.
55.           complete_dict['video'].appen
d(complete_video)
56.
57.           num_processed += 1
58.           if num_processed % 10 == 0:
59.               print("{} Video Processe
d: {}".format
60.                     (num_processed, da
tetime.datetime.now()))
61.
62.           if 'nextPageToken' in j_results:
63.               parameters['pageToken'] = j_result
s['nextPageToken']
64.           else:
65.               video_search_nextpage = False
66.
67.           print("\nDone!\n{} Videos Processed in {}".for
mat(
68.             num_processed, datetime.datetime.now() -
scrape_starttime))
69.
70.           return complete_dict

```

Kode 5.12 Method crawlvideo

Kode diatas merupakan potongan kode untuk *method* crawlvideo yang bertujuan untuk mengambil data video di channel Youtube. Parameter yang digunakan dalam penelitian ini yaitu video comment. Parameter video comment digunakan

untuk mendapatkan komentar dari suatu video melalui method crawlcomments. Data akhir dalam *method* ini akan disimpan sebagai dictionary dalam bentuk JSON.

```
1. def crawlcomments(idvideo):
2.     comm_dict_returned = []
3.     parameters = {"part": "snippet",
4.                     "maxResults": 100,
5.                     "videoId": idvideo,
6.                     "key": api_key,
7.                     "fields": "items(snippet(top
8.                         LevelComment(id,snippet(publishedAt,textOriginal
9. ))),nextPageToken"
10.                        }
11.                        url = "https://www.googleapis.com/youtube/
12. v3/commentThreads"
13. nextPageToken = ''
14. has_next_page = True
15.
16.         while has_next_page:
17.             parameters['pageToken'] = '' if '' e
18.             lse nextPageToken
19.
20.             try:
21.                 page = requests.request(method="ge
22. t", url=url, params=parameters)
23.                 page.raise_for_status()
24.             comment_results = json.loads(page.
25. text)
26.
27.             if comment_results['items']:
28.                 for comment in comment_results['
29. items']:
30.                     comm_dict = {}
31.                     comm_dict['comment_id'] =
32.                     comment['snippet']['topLevelComment']['id']
33.                     comm_dict['comment_message'
34. '] = comment['snippet']['topLevelComment'][ 'snip
35. pet']['textOriginal']
36.                     comm_dict['comment_date']
37. = comment['snippet']['topLevelComment'][ 'snippet
38. ']['publishedAt']
```

```

29.                     comm_dict_returned.append(
    comm_dict)
30.
31.             if 'nextPageToken' in comment_resu
    lts:
32.                 nextPageToken = comment_resu
    lts['nextPageToken']
33.             else:
34.                 has_next_page = False
35.
36.         except requests.HTTPError as e:
37.             has_next_page = False
38.             pass
39.     return comm_dict_returned

```

Kode 5.13 Method crawlcomments

Kode diatas merupakan potongan kode untuk *method* crawlcomments yang bertujuan untuk mendapatkan komentar pada suatu video. Parameter yang digunakan yaitu comment_id, comment_message, dan comment_date. Parameter comment_id bertujuan untuk mendapatkan id dari suatu komentar, comment message bertujuan untuk mendapatkan teks komentar, dan comment_date bertujuan untuk mendapatkan tanggal komentar dikirim. Data akhir komentar akan disimpan ke dalam dictionary agar mengetahui kepemilikan komentar dalam data JSON.

```

1. if __name__ == '__main__':
2.
3.     start_row = 1
4.
5.     idpemda = spreadsheet.scrapingcolumn(3, start_
    row, None)
6.     namapemda = spreadsheet.scrapingcolumn(1, star
    t_row, None)
7.     idchannelpemda = spreadsheet.scrapingcolumn(9,
    start_row, -1)
8.     idchannelinfluencer = spreadsheet.scrapingcolu
    mn(12, start_row, -1)
9.
10.    api_key = <Youtube API Key>

```

```

11.
12. for satuanidpemda in idpemda:
13.
14.     pemda_dict = {}
15.     pemda_dict['pemda_id'] = satuanidpemda
16.     pemda_dict['pemda_name'] = namapemda[idpemda
       .index(satuanidpemda)]
17.     pemda_dict['resmi'] = {}
18.     pemda_dict['influencer'] = {}
19.
20.     satuanidchannelpemda = idchannelpemda[idpemda
       .index(satuanidpemda)]
21.     satuanidchannelinfluencer = idchannelinfluen
       cer[idpemda.index(satuanidpemda)]
22.
23.     print("\n===== No.{:^2.5} {:^5}".format(satuanidpemda, pemda_dict['pemda_name']))
24.
25.     if satuanidchannelpemda is not '':
26.         pemda_dict['resmi'] = crawlvideo(api_key,
       satuanidchannelpemda, pemda_dict['pemda_id'], pe
       mda_dict['pemda_name'], 'resmi')
27.
28.     elif satuanidchannelpemda is '':
29.         pemda_dict['resmi'][ 'video'] = []
30.         pemda_dict['resmi'][ 'video'].append({ 'vide
       o_comments': []})
31.
32.     if satuanidchannelinfluencer is not '':
33.         pemda_dict['influencer'] = crawlvideo(api_
       key, satuanidchannelinfluencer, pemda_dict['pemd
       a_id'], pemda_dict['pemda_name'], 'influencer')
34.
35.     elif satuanidchannelinfluencer is '':
36.         pemda_dict['influencer'][ 'video'] = []
37.         pemda_dict['influencer'][ 'video'].append({ 'vide
       o_comments': []})
38.
39.     if satuanidpemda is idpemda[0]:
40.         exported = '[' + str(json.dumps(pemda_dict
       , indent=4)) + ','
41.
42.         with open('youtube_lengkap_mei.json', 'a')
       as file:

```

```

43.         file.write(exported)
44.
45.     elif satuanidpemda is not idpemda[-1]:
46.         exported = str(json.dumps(pemda_dict, indent=4)) + ','
47.
48.     with open('youtube_lengkap_mei.json', 'a')
49.         as file:
50.             file.write(exported)
51.
52.     else:
53.         exported = str(json.dumps(pemda_dict, indent=4)) + ']'
54.
55.     with open('youtube_lengkap_mei.json', 'a')
56.         as file:
57.             file.write(exported)

```

Kode 5.14 Method Main

Kode diatas merupakan potongan kode untuk *method main* yang bertujuan untuk mendapatkan daftar akun yang telah didaftarkan pada *google spreadsheets*. *Crawler* akan berjalan secara berulang sejumlah akun yang telah terdaftar. Selain itu *method* ini juga melakukan pengecekan terhadap pemerintah daerah yang memiliki akun resmi namun tidak memiliki akun influencer dan begitu juga sebaliknya. Hasil akhir dari kode ini yaitu JSON lengkap keseluruhan akun pemerintah daerah yang telah terparsing dan dilakukan pengiriman data komentar melalui kafka. Hasil komentar JSON Youtube dapat dilihat pada Tabel 4.4 Tabel Field Youtube yang Digunakan

5.2.1.2 Pemisahan Hasil Crawling Postingan dan Komentar

Data yang didapatkan dari hasil crawling masih tergabung antara postingan oleh pemerintah daerah dan komentar netizen. Oleh karena itu, langkah selanjutnya adalah dengan melakukan pemisahan data hasil crawling antara postingan dan komentar.

Data hasil crawling sebelumnya telah terbagi berdasarkan pemerintah daerah beserta tipe akun pemerintah daerah baik

akun resmi dan akun *influencer* di daerah tersebut. Dalam melakukan pemisahan antara postingan dan komentar, parameter yang diambil yaitu id komentar atau tweet, tanggal dibuatnya komentar atau tweet, text komentar, dan id akun yang pemerintah daerah atau influencer. Setelah data komentar didapatkan maka langkah selanjutnya yaitu melakukan pengkategorian komentar. Dalam melakukan pemisahan hasil crawling postingan dan komentar dilakukan dua tahap yaitu parsing komentar dan mengubah struktur data menjadi rdd.

A. Parsing komentar

Parsing data komentar merupakan tahapan untuk mengambil data komentar yang diperlukan. Hal ini dikarenakan data komentar yang dikirimkan oleh kafka masih tergabung dengan data post. Berikut merupakan kode parsing komentar

```

1. def parsing_facebookComment(comment):
2.     comment_id = comment['_id']
3.     comment_createdDate = comment['comment_createdDate']
4.     page_id = comment['page_id']
5.     text = comment['comment_message']
6.
7.     return comment_id, comment_createdDate, page_id, text

```

Kode 5.15 Parsing facebookComment

Kode diatas merupakan parsing data komentar facebook yang terdiri atas parameter `_id` untuk id komentar, `comment_createdDate` untuk tanggal komentar, `page_id` untuk id halaman page, dan `comment_message` untuk isi teks komentar.

```

1. def parsing_tweetReply(tweet):
2.     tweet_id = ''
3.     tweet_createdDate = ''
4.     account_id = ''
5.     text = ''
6.

```

```

7.     if tweet['in_reply_to_screen_name'] is not None:
8.         if tweet['in_reply_to_screen_name'] in lists:
9.             if not tweet['truncated']:
10.                 text = tweet['text']
11.
12.             elif tweet['truncated']:
13.                 text = tweet['extended_tweet'][
14.                   'full_text']
15.             tweet_id = tweet['id_str']
16.             tweet_createdDate = eu.formatTwitter
17.               Time(tweet['created_at']))
18.             account_id = tweet['in_reply_to_screen_name'].lower()
19.         return tweet_id, tweet_createdDate, account_
    id, text

```

Kode 5.16 parsing_tweetReply

Kode diatas merupakan parsing untuk komentar pada twitter yang masuk. Pada kode diatas dilakukan pengecekan apakah tweet tersebut merupakan komentar dengan melihat parameter in_reply_to_screen_name dan apakah teks tweet terpotong atau tidak dengan melihat parameter truncated. Adapun data komentar yang diambil yaitu id_str untuk id tweet, text untuk teks tweet, account_id untuk melihat akun id yang di mention, dan tweet_createdDate untuk tanggal komentar.

```

1. def parsing_youtubeComment(comment):
2.     comment_id = comment['_id']
3.     comment_createdDate = comment['comment_createdDate']
4.     channel_id = comment['channel_id']
5.     text = comment['comment_message']
6.
7.     return comment_id, comment_createdDate, channel_id, text

```

Kode 5.17 parsing_youtubeComment

Kode diatas merupakan parsing untuk komentar youtube yang masuk. Parameter yang diambil yaitu _id untuk id komentar, comment_createdDate untuk tanggal komentar dibuat, channel_id untuk id channel, dan text untuk teks komentar.

Setelah dilakukan parsing data komentar maka langkah selanjutnya dilakukan perubahan struktur data menjadi RDD. Hal ini bertujuan agar data komentar nantinya dapat diubah menjadi data frame untuk dilakukan proses preprocessing data dan kategorisasi.

B. Perubahan struktur data RDD

Perubahan struktur data RDD menjadi data frame bertujuan agar proses preprocessing dan kategorisasi pada spark dapat berjalan. Berikut merupakan kode perubahan struktur data RDD

```

1. def processTwitter(rdd):
2.     if rdd.isEmpty():
3.         print('No Data Received from Twitter\n')

4.     else:
5.         schema = StructType([
6.             StructField('_id', StringType()),
7.             StructField('tweet_createdDate', Str
8.             ingType()),
9.             StructField('account_id', StringType
10.            ()),
11.             StructField('text', StringType()),
12.             ])
11.         df = spark.createDataFrame(rdd, schema)

12.         df = sparkDF_to_pd(df)
13.
14.         process_item(df)

```

Kode 5.18 Method processTwitter

Kode diatas merupakan potongan kode untuk proceessTwitter yang bertujuan untuk mengubah data parsing json menjadi struktur rdd. Adapun parameter yang diambil yaitu _id, tweet_createdDate, account_id, dan text.

```

1. def processFacebook(rdd):
2.     if rdd.isEmpty():
3.         print('No Data Received from Facebook\n')
4.     else:
5.         schema = StructType([
6.             StructField('_id', StringType()),
7.             StructField('comment_createdDate', S
8.             tringType()),
9.             StructField('page_id', StringType())
10.            ],
11.            df = spark.createDataFrame(rdd, schema)
12.            df = sparkDF_to_pd(df)
13.
14.            process_item(df)

```

Kode 5.19 Method processFacebook

Kode diatas merupakan potongan kode untuk proceessFacebook yang bertujuan untuk mengubah data parsing json menjadi struktur rdd. Adapun parameter yang diambil yaitu _id, comment_createdDate, page_id, dan text.

```

1. def processYoutube(rdd):
2.     if rdd.isEmpty():
3.         print('No Data Received from Youtube\n')
4.     else:
5.         schema = StructType([
6.             StructField('_id', StringType()),
7.             StructField('comment_createdDate', S
8.             tringType()),
9.             StructField('channel_id', StringType
10.            )),
11.            df = spark.createDataFrame(rdd, schema)
12.            df = sparkDF_to_pd(df)
13.

```

```
| 14. process_item(df)
```

Kode 5.20 Method processYoutube

Kode diatas merupakan potongan kode untuk proceessYoutube yang bertujuan untuk mengubah data parsing json menjadi struktur rdd. Adapun parameter yang diambil yaitu _id, comment_createdDate, channel_id, dan text.

5.2.1.3 Pengolahan Data

Setelah data komentar didapatkan, langkah selanjutnya yaitu melakukan pengolahan data. Pengolahan data yang dilakukan dalam langkah ini terbagi menjadi dua tahapan yaitu preprocessing data dan pengkategorian komentar.

a. Preprocessing data

Tahapan preprocessing data merupakan tahapan yang bertujuan untuk membersihkan data komentar yang masuk. Tahapan preprocessing data terbagi menjadi stopword removal, stemming, n-grams, dan document indexing.

A. Stopword removal

Pada tahapan ini, data komentar yang masuk akan dilakukan penghilangan kata-kata yang sering muncul dan tidak memiliki makna. Dalam melakukan stopword removal, digunakan library sastrawi. Berikut merupakan kode yang digunakan untuk melakukan stopword removal.

```
1. f = open("stopwords-id.txt", "r")
2. stopwords = f.read().split('\n')
3. filtered_features = [w for w in feature_names
   if not w in stopwords]
```

Kode 5.21 stopword removal

B. Stemming

Stemming merupakan tahapan untuk mengubah kata-kata dalam suatu kalimat menjadi kata dasar. Dalam pengerjaan

tugas akhir ini, tahapan stemming menggunakan library Sastrawi dengan menggunakan fungsi stem. Berikut merupakan kode yang digunakan untuk melakukan stemming.

```
1. df['text'] = df['text'].apply(stemmer.stem)
```

Kode 5.22 stemming

C. N-grams

N-grams merupakan tahapan untuk membagi kalimat pada komentar menjadi n bagian. Pada tahapan ini, n gram yang digunakan memiliki nilai maksimal sebesar dua sehingga terdapat satu dan dua kata yang dihasilkan dari kalimat. Hal ini bertujuan agar keyword rekomendasi tidak hanya sebuah kata saja namun bisa berupa dua kata. Dalam melakukan N-grams digunakan fungsi ngrams_range pada library TfIdfVectorizer. Berikut merupakan kode yang digunakan untuk melakukan N-grams.

```
1. tfidf_vectorizer = TfidfVectorizer(ngram_range=(1,2))
```

Kode 5.23 N-grams

b. Pengkategorian Komentar

Setelah data komentar yang masuk telah dilakukan preprocessing, maka langkah selanjutnya yaitu melakukan kategorisasi komentar. Secara umum, pengkategorian komentar bekerja dengan cara mencari kata kunci pada kumpulan komentar yang telah masuk. Pengkategorian komentar akan dijelaskan lebih lanjut pada sub bab selanjutnya.

5.2.1.4 Penyimpanan ke dalam Database

Langkah terakhir dalam proses crawling yaitu menyimpan data hasil crawling ke dalam database. Data yang disimpan ke dalam database merupakan data komentar yang telah terlabeli

berdasarkan kategorinya. Selain itu, data komentar yang telah dipisahkan oleh postingan akan disatukan kembali berdasarkan postinganya dan akun pemerintah daerah. Berikut merupakan kode untuk menyimpan data ke dalam database.

```
1. def insert_mongo(json_categorized):
2.
3.     if 'account_id' in json_categorized:
4.         twMongo.updateComment(json_categorized)
5.
6.     elif 'page_id' in json_categorized:
7.         fbMongo.updateComment(json_categorized)
8.
9.     elif 'channel_id' in json_category:
10.        ytMongo.updateComment(json_categorized)
```

Kode 5.24 Method insert_mongo

Konfigurasi database dengan menggunakan mongoDB dapat dilihat pada penelitian Dyaksa Hanindto[33]

5.2.2 Pengkategorian Komentar

Pengkategorian komentar bertujuan untuk mengelompokkan data komentar yang masuk berdasarkan kata kunci yang telah dimasukkan oleh pemerintah daerah berdasarkan SKPD. Dalam melakukan pengkategorian komentar, dilakukan pencarian komentar terlebih dahulu dilakukan pemecahan kalimat menjadi kata dengan fungsi bigram. Selanjutnya dilakukan pencarian kata berdasarkan kata kunci. Proses pencarian kata kunci dilakukan dengan menggunakan fungsi regular expression (regex) kepada kata kunci untuk mencari kata dengan menggunakan fungsi intersect pada kata dalam komentar dengan keyword yang telah didaftarkan. Apabila data komentar memiliki kata kunci yang dimaksud, maka komentar akan dilabeli berdasarkan SKPD terkait. Sedangkan apabila dalam komentar terdapat lebih dari satu keyword maka komentar akan dilabeli dengan duplicate. Dan terakhir apabila

keyword tidak ditemukan dalam komentar maka komentar akan dilabeli dengan uncategorized.

Dalam melakukan proses kategorisasi komentar digunakan Apache Spark sebagai pengolahan data. Konfigurasi Apache Spark dapat dilihat pada penelitian Muhammad Fikry Hazmi mengenai klasifikasi komentar[34]

Berikut merupakan kode yang digunakan untuk melakukan pengkategorian komentar.

```

1. FOR each comment
2.   GET account_id, text_comment
3.   GET dinas from account_id
4.   SET category as "uncategorized"
5.   FOR each dinas
6.     IF check text_comment intersect with keyword in dinas
7.       SET category as "dinas"
8.       count++
9.     ENDIF
10.    IF count > 1
11.      SET category as "duplicate"
12.    ENDFOR
13. ENDFOR

```

Kode 5.25 Pseudocode pengkategorian komentar

5.2.3 Rekomendasi Keyword

Setelah komentar telah dilabeli berdasarkan dinas, langkah selanjutnya yaitu mencari 5 keyword rekomendasi berdasarkan nilai IDF paling rendah. Setelah diketahui rekomendasi keyword maka keyword tersebut akan dimasukkan ke dalam database untuk ditunjukkan kepada user.

```

1. FOR each pemda
2.   GET dinas
3.   FOR each dinas
4.     GET comment where category is dinas and account_id is account.pemda.resmi
5.     IF comment in each dinas more than average of comment
6.       FOR each comment
7.         DO count IDF, sort IDF value

```

```

8.           GET word in comment where IDF val
     ue is low
9.           ENDFOR
10.          ENDIF
11.         ENDFOR
12.        ENDFOR

```

Kode 5.26 Pseudocode Rekomendasi Keyword

Kode diatas merupakan potongan pseudocode untuk mendapatkan 5 kata kunci yang memiliki nilai IDF terendah. Langkah pertama yang dilakukan yaitu mendapatkan nama dinas untuk setiap pemda. Kemudian dilakukan pencarian komentar yang telah terkategorisasi berdasarkan nama dinas masing-masing. Setelah data komentar didapatkan dilakukan pengecekan terlebih dahulu apakah jumlah komentar dalam dinas tersebut lebih besar dari rata-rata komentar dalam satu pemda, apabila lebih besar maka langkah selanjutnya yaitu menghitung nilai IDF yang terdapat dalam komentar dan kemudian kata tersebut dilakukan pengurutan dari nilai terkecil menjadi nilai terbesar. 5 Kata yang memiliki nilai IDF terendah akan dijadikan rekomendasi kepada pemerintah daerah.

5.2.4 Visualisasi Data

Dalam melakukan visualisasi, digunakan plugin highchart.js untuk menampilkan grafik. Aplikasi menampilkan dua jenis grafik yaitu grafik batang dan grafik garis. Grafik batang terdiri atas lima kategori dinas yang memiliki jumlah komentar terbanyak dan total komentar yang masuk untuk setiap pemerintah daerah. Sementara grafik garis menampilkan jumlah post yang masuk 10 hari terakhir Selain itu, aplikasi menampilkan data melalui table yang berisikan jumlah komentar setiap dinas untuk setiap social media.

5.2.4.1 Lima Kategori Dinas yang Memiliki Jumlah Komentar Terbanyak

Dalam menampilkan data, dipilih grafik batang untuk merepresentasikan jumlah komentar di masing-masing social media. Grafik batang yang digunakan merupakan grafik batang berkelompok yang mengelompokan social media akun resmi

dan influencer. Dalam melakukan visualisasi, dibutuhkan nama pemda, nama dinas, beserta jumlah komentar yang telah terkategorisasi. Karena ingin menampilkan 5 dinas yang memiliki komentar terbanyak maka terlebih dahulu dilakukan perhitungan total komentar dimasing-masing social media. Berikut merupakan potongan kode untuk visualisasi lima kategori dinas yang memiliki jumlah komentar terbanyak

```
1. $pemda = listPemdaModel::where('_id', (int)$id)->first();  
2. $dinases = dinasModel::where('idPemda', (int)$id)->get();  
3. $namaDinas = [];  
4. $jumlahKomentarFacebookResmi = [];  
5. $jumlahKomentarFacebookInfluencer = [];  
6. $jumlahKomentarTwitterResmi = [];  
7. $jumlahKomentarTwitterInfluencer = [];  
8. $jumlahKomentarYoutubeResmi = [];  
9. $jumlahKomentarYoutubeInfluencer = [];  
10.  
11. //facebook resmi  
12. $komentarFacebookResmi = facebookCommentsModel::where('page_id', $pemda->facebook_resmi)->get();  
13. //facebook influencer  
14. $komentarFacebookInfluencer = facebookCommentsModel::where('page_id', $pemda->facebook_influencer)->get();  
15. //twitter resmi  
16. $komentarTwitterResmi = twitter_replyModel::where('account_id', $pemda->twitter_resmi)->get();  
17. //twitter influencer  
18. $komentarTwitterInfluencer = twitter_replyModel::where('account_id', $pemda->twitter_influencer)->get();  
19. //youtube resmi  
20. $komentarYoutubeResmi = youtubeCommentsModel::where('channel_id', $pemda->youtube_resmi)->get();  
21. //youtube influencer  
22. $komentarYoutubeInfluencer = youtubeCommentsModel::where('channel_id', $pemda->youtube_influencer)->get();
```

```

23.
24.     foreach($dinases as $dinas) {
25.         $namaDinas[] = $dinas['nama_dinas'];
26.         $dinas['facebook_resmi'] = $komentarFacebookResmi-
>where('category', $dinas['nama_dinas'])-
>count();
27.         $dinas['facebook_influencer'] = $komentarFacebookInfluencer-
>where('category', $dinas['nama_dinas'])-
>count();
28.         $dinas['twitter_resmi'] = $komentarTwitterResmi->where('category', $dinas['nama_dinas'])-
>count();
29.         $dinas['twitter_influencer'] = $komentarTwitterInfluencer-
>where('category', $dinas['nama_dinas'])-
>count();
30.         $dinas['youtube_resmi'] = $komentarYoutubeResmi->where('category', $dinas['nama_dinas'])-
>count();
31.         $dinas['youtube_influencer'] = $komentarYoutubeInfluencer-
>where('category', $dinas['nama_dinas'])-
>count();
32.         $dinas['total_komentar'] = $dinas['facebook_resmi'] + $dinas['facebook_influencer'] + $dinas['twitter_resmi'] + $dinas['twitter_influencer'] + $dinas['youtube_resmi'] + $dinas['youtube_influencer'];
33.     }
34.
35.     $top5dinas = $dinases-
>sortByDesc('total_komentar')->take(5);
36.
37.     foreach($top5dinas as $td) {
38.         $namaDinas[] = $td['nama_dinas'];
39.         $jumlahKomentarFacebookResmi[] = $komentarFacebookResmi-
>where('category', $td['nama_dinas'])->count();
40.         $jumlahKomentarFacebookInfluencer[] = $komentarFacebookInfluencer-
>where('category', $td['nama_dinas'])->count();

```

```

41.           $jumlahKomentarTwitterResmi[] = $komentar
        TwitterResmi-
        >where('category', $td['nama_dinas'])->count();
42.           $jumlahKomentarTwitterInfluencer[] = $kom
        entarTwitterInfluencer-
        >where('category', $td['nama_dinas'])->count();
43.           $jumlahKomentarYoutubeResmi[] = $komentar
        YoutubeResmi-
        >where('category', $td['nama_dinas'])->count();
44.           $jumlahKomentarYoutubeInfluencer[] = $kom
        entarYoutubeInfluencer-
        >where('category', $td['nama_dinas'])->count();
45.       }
    
```

Kode 5.27 Query Lima Kategori Dinas Terbanyak

5.2.4.2 Jumlah komentar yang masuk dalam 10 hari terakhir

Dalam menampilkan data jumlah komentar yang masuk hari ini digunakan grafik garis yang menunjukkan jumlah komentar dalam 10 hari terakhir. Data yang dibutuhkan yaitu tanggal 10 hari kebelakang dan jumlah komentar yang masuk dimasing-masing social media. Berikut merupakan potongan kode untuk jumlah post yang masuk 10 hari terakhir.

```

1. for($i = 9; $i >= 0; $i --) {
2.     $tanggal[] = date('Y-m-d', strtotime('-
        '.$i.' days', strtotime(date("Y-m-d"))));
3. }
4.
5.     //komen
6.     foreach($tanggal as $tgl) {
7.         $hitungKomenFB[] = facebookCommentsModel
        ::where('comment_createdDate', $tgl)->count();
8.         $hitungKomenTW[] = twitter_replyModel::w
        here('tweet_createdDate', $tgl)->count();
9.         $hitungKomenYT[] = youtubeCommentsModel:
        ::where('comment_createdDate', $tgl)->count();
10.        $hitungPostFB[] = facebookPostsModel::wh
        ere('post_createdDate', $tgl)->count();
11.        $hitungPostTW[] = twitterPostsModel::whe
        re('tweet_createdDate', $tgl)->count();
12.        $hitungPostYT[] = youtubePostsModel::whe
        re('video_createdDate', $tgl)->count();
    
```

```
| 13. }
```

Kode 5.28 Query jumlah komentar 10 hari terakhir

5.2.4.3 Jumlah Komentar yang Masuk untuk Setiap Pemerintah Daerah

Visualisasi jumlah komentar yang masuk untuk setiap pemerintah daerah dilakukan dengan menggunakan grafik batang. Selain jumlah komentar yang masuk, pada grafik divisualisasikan garis yang menunjukkan rata-rata komentar yang masuk serta pengelompokan pemerintah daerah berdasarkan standard deviasi yang terbagi menjadi ± 1 SD, ± 2 SD, dan ± 3 SD. Parameter ± 1 SD menunjukkan bahwa nilai standard deviasi bernilai positif 1 atau negative 1 yang berarti bahwa pemerintah daerah tersebut memiliki jumlah komentar yang mendekati nilai rata-rata. Hal ini berkebalikan dengan parameter ± 3 SD yang berarti jumlah komentar pemerintah daerah tersebut jauh dari nilai rata-rata. Berikut merupakan potongan kode untuk menampilkan grafik batang jumlah komentar yang masuk untuk setiap pemerintah daerah

```
1. $listPemda = listPemdaModel::all();
2.     foreach($listPemda as $lp) {
3.         $namaPemda[] = $lp['name'];
4.
5.         $facebookResmiLower = strtolower($lp['face
book_resmi']);
6.         $hitungPostPemda = facebookAccountsResultM
odel::where('page_id', $facebookResmiLower)-
>first();
7.         $lp['commentCountFBResmi'] = $hitungPostPe
mda['result.statistics.commentCount'];
8.
9.         $facebookInfluencerLower = strtolower($lp[
'facebook_influencer']);
10.        $hitungPostPemda = facebookAccountsResultM
odel::where('page_id', $facebookInfluencerLower)-
>first();
11.        $lp['commentCountFBInfluencer'] = $hitungP
ostPemda['result.statistics.commentCount'];
```

```
12.
13.      $twitterResmiLower = strtolower($lp['twitt
   er_resmi']);
14.      $hitungTweetPemda = twitterAccountsResultM
   odel::where('account', $twitterResmiLower)->first();
15.      $lp['replyCountResmi'] = $hitungTweetPemda
   ['result.statistics.replyCount'];
16.
17.      $twitterInfluencerLower = strtolower($lp['
   twitter_influencer']);
18.      $hitungTweetPemda = twitterAccountsResultM
   odel::where('account', $twitterInfluencerLower)->first();
19.      $lp['replyCountInfluencer'] = $hitungTweet
   Pemda['result.statistics.replyCount'];
20.
21.      $youtubeResmiLower = strtolower($lp['youtu
   be_resmi']);
22.      $hitungVideoPemda = youtubeAccountsResultM
   odel::where('channel_id', $youtubeResmiLower)->first();
23.      $lp['commentCountYTResmi'] = $hitungVideoP
   emda['result.statistics.commentCount'];
24.
25.      $youtubeInfluencerLower = strtolower($lp['
   youtube_influencer']);
26.      $hitungVideoPemda = youtubeAccountsResultM
   odel::where('channel_id', $youtubeInfluencerLower)->first();
27.      $lp['commentCountYTInfluencer'] = $hitungV
   ideoPemda['result.statistics.commentCount'];
28.
29.      $lp['totalComment'] = $lp['commentCountFBR
   esmi'] + $lp['replyCountResmi'] + $lp['commentCou
   ntYTResmi'] + $lp['commentCountFBInfluencer'] + $l
   p['replyCountInfluencer'] + $lp['commentCountYTI
   nfluencer'];
30.
31. }
32.
33. $sortPemdaComment = $listPemda->sortByDesc('totalComment');
34.
35. foreach ($sortPemdaComment as $spc) {
```

```

36.      $namaPemdaComment[] = $spc['name'];
37.      $jumlahComment[] = $spc['totalComment'];
38.    }
39.
40.    $rata = $this->average($jumlahComment);
41.    $std = $this-
>Stand_Deviation($jumlahComment);

```

Kode 5.29 Query jumlah komentar setiap pemda

Dalam melakukan visualisasi dibutuhkan fungsi untuk menghitung standard deviasi dan rata-rata data komentar. Berikut merupakan potongan kode untuk menghitung standard deviasi dan rata-rata.

```

1.  public function Stand_Deviation($arr)
2.  {
3.      $num_of_elements = count($arr);
4.
5.      $variance = 0.0;
6.          // calculating mean using array_s
um() method
7.      $average = array_sum($arr)/$num_of_eleme
nts;
8.
9.      foreach($arr as $i)
10.     {
11.         // sum of squares of differences betw
een
12.             // all numbers and means.
13.
14.             $variance += pow(($i - $average), 2);
15.
16.     return (float)sqrt($variance/$num_of_ele
ments);
17.   }
18.
19.   public function average($arr)
20.   {
21.       $num_of_elements = count($arr);
22.
23.       $variance = 0.0;

```

```

24.      // calculating mean using array_sum() method
25.      $average = array_sum($arr)/$num_of_elements
26.      ;
27.      return round($average,2);
}

```

Kode 5.30 Method standar deviasi dan rata-rata

5.2.4.4 Tabel jumlah komentar setiap dinas

Visualisasi table ditunjukkan untuk menampilkan jumlah komentar yang masuk untuk setiap dinas pada suatu pemerintah kota. Dalam menampilkan table, digunakan dataTable untuk mempermudah pagination, pencarian data, dan sorting data. Adapun data yang digunakan yaitu nama dinas, jumlah komentar pada facebook, twitter, dan youtube yang telah memiliki label sesuai dinas, dan jumlah total keseluruhan komentar untuk masing-masing dinas. Berikut merupakan potongan kode untuk table jumlah komentar setiap dinas

```

1.      $pemda = listPemdaModel::where('_id', (int)
2.          $id)->first();
3.      $dinases = dinasModel::where('idPemda', (i
4.          nt)$id)->get();
5.      $namaDinas = [];
6.      $jumlahKomentarFacebookResmi = [];
7.      $jumlahKomentarFacebookInfluencer = [];
8.      $jumlahKomentarTwitterResmi = [];
9.      $jumlahKomentarTwitterInfluencer = [];
10.     $jumlahKomentarYoutubeResmi = [];
11.     $jumlahKomentarYoutubeInfluencer = [];
12.     //facebook resmi
13.     $komentarFacebookResmi = facebookCommentsM
14.         odel::where('page_id', $pemda->facebook_resmi)->get();
15.     //facebook influencer
16.     $komentarFacebookInfluencer = facebookComm
17.         entsModel::where('page_id', $pemda->facebook_influencer)->get();
18.     //twitter resmi
19.     $komentarTwitterResmi = twitter_replyModel
20.         ::where('account_id', $pemda->twitter_resmi)->get();

```

```
16.      //twitter influencer
17.      $komentarTwitterInfluencer = twitter_reply
Model::where('account_id', $pemda-
>twitter_influencer)->get();
18.      //youtube resmi
19.      $komentarYoutubeResmi = youtubeCommentsMod
el::where('channel_id', $pemda->youtube_resmi)-
>get();
20.      //youtube influencer
21.      $komentarYoutubeInfluencer = youtubeCommen
tsModel::where('channel_id', $pemda-
>youtube_influencer)->get();
22.
23.      foreach($dinases as $dinas) {
24.          $namaDinas[] = $dinas['nama_dinas'];
25.          $dinas['facebook_resmi'] = $komentarFace
bookResmi-
>where('category', $dinas['nama_dinas'])-
>count();
26.          $dinas['facebook_influencer'] = $komenta
rFacebookInfluencer-
>where('category', $dinas['nama_dinas'])-
>count();
27.          $dinas['twitter_resmi'] = $komentarTwitt
erResmi-
>where('category', $dinas['nama_dinas'])-
>count();
28.          $dinas['twitter_influencer'] = $komentar
TwitterInfluencer-
>where('category', $dinas['nama_dinas'])-
>count();
29.          $dinas['youtube_resmi'] = $komentarYoutu
beResmi-
>where('category', $dinas['nama_dinas'])-
>count();
30.          $dinas['youtube_influencer'] = $komentar
YoutubeInfluencer-
>where('category', $dinas['nama_dinas'])-
>count();
31.          $dinas['total_komentar'] = $dinas['faceb
ook_resmi'] + $dinas['facebook_influencer'] + $d
inas['twitter_resmi'] + $dinas['twitter_influenc
er'] + $dinas['youtube_resmi'] + $dinas['youtube
_influencer'];
```

```
|32. }
```

Kode 5.31 Query tabel jumlah komentar setiap dinas

5.2.5 Fitur pada Aplikasi

Pada bagian ini dijelaskan fitur-fitur yang terdapat pada aplikasi khususnya mengenai operasional data komentar

5.2.5.1 Mengubah kategori dinas pada komentar

Fitur ini bertujuan untuk mengubah kategori dinas pada komentar apabila hasil kategori yang diberikan tidak sesuai. Fitur ini dapat dilakukan untuk akun pemerintah daerah. Dalam melakukan pengambilan data komentar, terlebih dahulu didefinisikan nama dinas yang bertujuan untuk memanggil semua kategori dinas pada suatu pemerintah daerah. Kemudian pemerintah daerah memilih salah satu komentar untuk dilakukan edit dan memilih kategori yang baru sesuai dengan dinas pada pemerintah daerah. Penggantian kategori dinas dibedakan fungsinya berdasarkan sosial media. Hal ini disebabkan data komentar pada database disimpan dalam collection yang berbeda-beda sesuai sosial media. Pengambilan data komentar dilakukan pada function index() dan penggantian data komentar dilakukan pada function update().

Berikut merupakan potongan kode untuk mengubah kategori dinas pada komentar

```
1. public function index($id) {
2.     $user = userModel::where('idPemda', (int)$id)
->first();
3.     $dinases = $user->dinas;
4.     $pemda = listPemdaModel::where('_id', (int)$i
d)->first();
5.
6.     //ganti username ke lowercase
7.     $facebookResmi = strtolower($pemda-
>facebook_resmi);
8.     $facebookInfluencer = strtolower($pemda-
>facebook_influencer);
9.     $twitterResmi = strtolower($pemda-
>twitter_resmi);
```

```

10.      $twitterInfluencer = strtolower($pemda-
>twitter_influencer);
11.      $youtubeResmi = strtolower($pemda-
>youtube_resmi);
12.      $youtubeInfluencer = strtolower($pemda-
>youtube_influencer);
13.
14.      $namaDinas = [];
15.      foreach ($dinases as $dinas) {
16.          array_push($namaDinas, $dinas['nama_dinas'])
17.      };
18.
19. public function updateKomentar($id, $idKomentar,
Request $request) {
20.     $user = userModel::where('idPemda', (int)$id)
->first();
21.     $komentar = twitter_replyModel::where('_id',
$idKomentar)->first();
22.
23.     $komentar->category = $request->dinas;
24.
25.     $komentar->save();
26.
27.     return redirect()-
>route('kategorisasi.komentar.dinas', ['id' => $u
ser->idPemda]);
28. }

```

Kode 5.32 Mengubah Kategori Dinas

Tampilan dalam mengubah kategori dinas dapat dilihat pada Gambar 37 Tampilan Komentar Kategori Dinas dan Gambar 38 Edit Komentar Kategori Dinas

5.2.5.2 Mengubah kategori duplicate pada komentar

Fitur ini bertujuan untuk memberikan kategori dinas pada komentar yang terkategori sebagai duplicate. Pada fitur ini, pemerintah daerah diberikan pilihan untuk memilih salah satu dinas atau memilih semua dinas untuk dijadikan kategori dinas. Apabila pengguna memilih salah satu dinas, maka kategori akan digantikan menjadi nama dinas sementara apabila pengguna memilih semua dinas maka akan dibuat data komentar baru sejumlah dinas dengan kategori nama dinas dan menghapus data

komentar yang duplicate. Penggantian kategori duplicate dibedakan fungsinya berdasarkan sosial media. Hal ini disebabkan data komentar pada database disimpan dalam collection yang berbeda-beda sesuai sosial media. Pengambilan data komentar dilakukan pada function index dan penggantian data komentar dilakukan pada function update. Fungsi untuk memanggil data komentar dilakukan pada fungsi index(), fungsi untuk mengganti kategori duplicate menjadi dinas dilakukan pada fungsi gantiDuplicate(), dan fungsi untuk mengganti kategori duplicate menjadi semua dinas dilakukan pada fungsi gantiDuplicateInsert().

Berikut merupakan potongan kode untuk mengganti kategori duplicate pada komentar

```
1. public function index($id) {
2.     $user = userModel::where('idPemda', (int)$id)
3.         ->first();
4.     $pemda = listPemdaModel::where('_id', (int)$i
d)->first();
5.     //ganti username ke lowercase
6.     $facebookResmi = strtolower($pemda-
>facebook_resmi);
7.     $facebookInfluencer = strtolower($pemda-
>facebook_influencer);
8.     $twitterResmi = strtolower($pemda-
>twitter_resmi);
9.     $twitterInfluencer = strtolower($pemda-
>twitter_influencer);
10.    $youtubeResmi = strtolower($pemda-
>youtube_resmi);
11.    $youtubeInfluencer = strtolower($pemda-
>youtube_influencer);
12.
13.    //ambil komentar duplicate
14.    $facebookResmiComments = facebookCommentsMode
l::where('page_id', $facebookResmi)-
>where('category', 'duplicate')->get();
15.    $facebookInfluencerComments = facebookComment
sModel::where('page_id', $facebookInfluencer)-
>where('category', 'duplicate')->get();
```

```
16.      $twitterResmiComments = twitter_replyModel::w
here('account_id', $twitterResmi)-
>where('category', 'duplicate')->get();
17.      $twitterInfluencerComments = twitter_replyMod
el::where('account_id', $twitterInfluencer)-
>where('category', 'duplicate')->get();
18.      $youtubeResmiComments = youtubeCommentsModel:
where('channel_id', $youtubeResmi)-
>where('category', 'duplicate')->get();
19.      $youtubeInfluencerComments = twitter_replyMod
el::where('channel_id', $youtubeInfluencer)-
>where('category', 'duplicate')->get();
20.
21.
22.
23.      return view('komentarDuplicate', [
'facebookRe
smiComments' => $facebookResmiComments, 'facebook
InfluencerComments' => $facebookInfluencerComment
s, 'twitterResmiComments' => $twitterResmiComment
s, 'twitterInfluencerComments' => $twitterInfluen
cerComments, 'youtubeResmiComments' => $youtubeRe
smiComments, 'youtubeInfluencerComments' => $yout
ubeInfluencerComments, 'user' => $user, 'pemda' =
> $pemda]);
24.  }
25.
26. public function gantiDuplicate
($id, $idKomentar, Request $request) {
27.     $user = userModel::where('idPemda', (int)$id)
->first();
28.     $komentar = twitter_replyModel::where('_id',
$idKomentar)->first();
29.
30.     $komentar->category = $request->dinas_temp;
31.     $komentar->dinas_temp = null;
32.
33.     $komentar->save();
34.
35.     return redirect()-
>route('kategorisasi.komentar.duplicate', ['id' =
> $user->idPemda]);
36.  }
37.
38. public function gantiDuplicateInsert
($id, $idKomentar) {
```

```

39.      $user = userModel::where('idPemda', (int)$id)
        ->first();
40.      $komentar = twitter_replyModel::where('_id',
        $idKomentar)->first();
41.
42.      for ($i=0; $i<count($komentar['dinas_temp']); $i++) {
43.          $komentarBaru = new twitter_replyModel;
44.
45.          $komentarBaru->_id = $komentar-
        >_id.'_'.$komentar['dinas_temp'][$i];
46.          $komentarBaru-
        >tweet_createdDate = $komentar-
        >tweet_createdDate;
47.          $komentarBaru->reply_message = $komentar-
        >reply_message;
48.          $komentarBaru->account_id = $komentar-
        >account_id;
49.          $komentarBaru-
        >category = $komentar['dinas_temp'][$i];
50.          $komentarBaru->account_type = $komentar-
        >account_type;
51.          $komentarBaru->class = $komentar->class;
52.
53.          $komentarBaru->save();
54.      }
55.      $komentar->delete();
56.
57.
58.      return redirect()-
        >route('kategorisasi.komentar.duplicate', [ 'id' =
        > $user->idPemda]);
59.  }

```

Kode 5.33 Mengubah Kategori Duplicate

Tampilan dalam mengubah kategori dinas dapat dilihat pada Gambar 39 Komentar Kategori Duplicate

5.2.5.3 Mengubah kategori uncategorized pada komentar
 Fitur ini bertujuan untuk mengubah kategori uncategorized pada komentar oleh pemerintah daerah. Dengan adanya fungsi ini, pemerintah daerah dapat mengkategorikan komentar secara manual. Pertama-tama dilakukan pengambilan data komentar

dengan kategori Uncategorized, kemudian pemerintah daerah memilih salah satu komentar dan melakukan kategori berdasarkan dinas yang ada. Penggantian kategori Uncategorized dibedakan fungsinya berdasarkan sosial media. Hal ini disebabkan data komentar pada database disimpan dalam collection yang berbeda-beda sesuai sosial media. Pengambilan data komentar dilakukan pada function index() dan penggantian data komentar dilakukan pada function gantiUncategorized().

Berikut merupakan potongan kode untuk mengubah kategori Uncategorized pada komentar

```

1. public function index($id) {
2.     $user = userModel::where('idPemda', (int)$id)
    ->first();
3.     $pemda = listPemdaModel::where('_id', (int)$i
d)->first();
4.     $dinases = dinasModel::orderBy('nama_dinas',
    'asc')->where('idPemda', (int)$id)->get();
5.
6.     //ganti username ke lowercase
7.     $facebookResmi = strtolower($pemda-
    >facebook_resmi);
8.     $facebookInfluencer = strtolower($pemda-
    >facebook_influencer);
9.     $twitterResmi = strtolower($pemda-
    >twitter_resmi);
10.    $twitterInfluencer = strtolower($pemda-
    >twitter_influencer);
11.    $youtubeResmi = strtolower($pemda-
    >youtube_resmi);
12.    $youtubeInfluencer = strtolower($pemda-
    >youtube_influencer);
13.
14.    //ambil komentar duplicate
15.    $facebookResmiComments = facebookCommentsMode
    l::where('page_id', $facebookResmi)-
    >where('category', 'uncategorized')->get();
16.    $facebookInfluencerComments = facebookComment
    sModel::where('page_id', $facebookInfluencer)-
    >where('category', 'uncategorized')->get();

```

```

17.      $twitterResmiComments = twitter_replyModel::w
    here('account_id', $twitterResmi)->
    >where('category', 'uncategorized')->get();
18.      $twitterInfluencerComments = twitter_replyMod
    el::where('account_id', $twitterInfluencer)->
    >where('category', 'uncategorized')->get();
19.      $youtubeResmiComments = youtubeCommentsModel:
    :where('channel_id', $youtubeResmi)->
    >where('category', 'uncategorized')->get();
20.      $youtubeInfluencerComments = twitter_replyMod
    el::where('channel_id', $youtubeInfluencer)->
    >where('category', 'uncategorized')->get();
21.
22.      return view('komentarUncategorized', [
        'facebookResmiComments' => $facebookResmiComments,
        'facebookInfluencerComments' => $facebookInfluencerCom
        ments,
        'twitterResmiComments' => $twitterResmiCom
        ments,
        'twitterInfluencerComments' => $twitterInf
        luencerComments,
        'youtubeResmiComments' => $youtu
        beResmiComments,
        'youtubeInfluencerComments' => $yout
        beInfluencerComments,
        'user' => $user,
        'pemda' => $pemda,
        'dinases' => $dinases]);
23.  }
24.
25.
26.  public function gantiUncategorized($id, $idKome
    ntar, Request $request) {
27.      $user = userModel::where('idPemda', (int)$id)
    ->first();
28.      $komentar = twitter_replyModel::where('_id',
    $idKomentar)->first();
29.
30.      $komentar->category = $request->nama_dinas;
31.      $komentar->dinas_temp = null;
32.
33.      $komentar->save();
34.
35.      return redirect()->
    >route('kategorisasi.komentar.uncategorized', [
        'id' => $user->idPemda]);
36.  }

```

Kode 5.34 Mengubah Kategori Uncategorized

Tampilan dalam mengubah kategori dinas dapat dilihat pada Gambar 40 Komentar Kategori Uncategorized

5.2.5.4 Validasi kata kunci pada dinas

Fitur ini bertujuan untuk melakukan validasi pada saat pemerintah daerah membuat dinas baru atau mengedit dinas. Dengan adanya validasi ini, aplikasi akan melakukan pemeriksaan apakah kata kunci yang dimasukkan sudah ada atau belum pada database. Hal ini bertujuan untuk mencegah kata kunci yang sama pada suatu pemerintah daerah.

Proses validasi dilakukan dengan memanggil kata kunci dari database dan dilakukan pemeriksaan pada back-end melalui controller pada Laravel. Apabila kata kunci yang dimasukkan sudah ada dalam database maka aplikasi akan memberikan peringatan yaitu kata kunci sudah ada. Berikut merupakan potongan kode untuk validasi kata kunci pada dinas.

```
1. $messages = [
2.     'nama_dinas.not_in' => 'Nama Dinas Sudah Ada',
3.     'keyword_dinas.not_in' => 'Kata Kunci Sudah Digunakan'
4. ];
5.
6. foreach ($dinases as $dinas) {
7.     $validatedData = $request->validate([
8.         'nama_dinas' => 'not_in:'.$dinas['nama_dinas'],
9.         'keyword_dinas' => 'not_in:'.$dinas['keyword_dinas'],
10.    ], $messages);
11. }
12.
```

Kode 5.35 Validasi Kata Kunci

BAB VI

HASIL DAN PEMBAHASAN

Pada bab ini akan dijelaskan hasil dan pembahasan dalam pengujian dan pembuatan aplikasi.

6.1 Analisa Pengujian Use Case

Pada bagian ini akan menampilkan hasil dari uji coba fungsional. Berikut merupakan hasil dari pengujian Uji Coba Fungsional untuk user sesuai fitur yang telah dibuat pada bab 4.

Tabel 6.1 Use Case Pengujian Untuk User

Nama Use Case	Status
Membuat akun	Berhasil
Login akun	Berhasil
Melihat dashboard	Berhasil
Mengedit akun	Berhasil
Melihat dashboard	Berhasil
Melihat daftar dinas	Berhasil
Menghapus dinas	Berhasil
Mengedit dinas	Berhasil
Membuat dinas baru	Berhasil
Menambahkan keyword baru	Berhasil
Mengedit keyword	Berhasil
Menghapus keyword	Berhasil

Sementara berikut merupakan uji coba fungsional untuk admin

Tabel 6.2 Use Case Pengujian Untuk Admin

Nama Use Case	Status
Melihat daftar akun pemkot	Berhasil
Menambahkan akun pemkot	Berhasil
Mengedit akun pemkot	Berhasil
Menambahkan dinas	Berhasil
Mengedit dinas	Berhasil
Menghapus dinas	Berhasil
Menambahkan keyword	Berhasil
Mengedit keyword	Berhasil
Menghapus keyword	Berhasil

Hasil pengujian disesuaikan dengan pembuatan use case yang telah disepakati pada bab 4. Berdasarkan hasil pengujian, seluruh use case telah berhasil diimplementasi ke dalam aplikasi. Permasalahan yang ditemukan dalam implementasi use case yaitu request time yang cukup tinggi terutama saat mengakses halaman dashboard. Hal ini dikarenakan jumlah dashboard yang banyak pada satu halaman dan untuk satu dashboard membutuhkan query data yang banyak. Solusi dari permasalahan ini yaitu dengan melakukan pemisahan dashboard ke dalam beberapa halaman. Pengujian dilakukan dengan melakukan perhitungan berapa lama waktu yang dibutuhkan aplikasi untuk melakukan loading. Berikut total waktu yang dibutuhkan dalam melakukan loading halaman

Tabel 6.3 Lama Waktu Loading Halaman Dashboard

Kondisi	Lama Waktu yang Dibutuhkan (detik)
Grafik dijadikan dalam satu halaman dashboard	7.02
Grafik dipisah ke dalam beberapa halaman	3.23

Berdasarkan tabel 6.1 terjadi penurunan waktu yang dibutuhkan oleh aplikasi sebesar 3.79 detik untuk melakukan loading. Hal ini disebabkan terjadi penurunan jumlah query yang dibutuhkan oleh aplikasi dalam membuat grafik.

Permasalahan lain yang ditemukan yaitu kesalahan dalam memanggil data username facebook, twitter, dan youtube pemerintah kota dan influencer. Permasalahan ini disebabkan karena username social media disimpan tidak dalam bentuk huruf kecil sedangkan untuk melakukan visualisasi data username pada komentar sudah dilakukan perubahan huruf menjadi huruf kecil, sehingga apabila dilakukan query, data komentar tidak terpanggil. Username pada komentar yang sudah diubah menjadi huruf kecil bertujuan untuk mencegah kesalahan pemerintah daerah dalam memasukkan username social media. Solusi dari permasalahan ini yaitu dengan melakukan perubahan huruf pada username menjadi huruf kecil melalui PHP.

6.2 Hasil Kategorisasi

Dalam melakukan pengujian kategorisasi, dilakukan percobaan pada akun kota pemerintah kota Depok dengan data komentar untuk periode 1 Januari 2016 hingga 29 Mei 2018:

Tabel 6.4 Jumlah Komentar Kota Depok

Akun Kota Depok	Jumlah Komentar
Facebook Resmi	1.038

Facebook Influencer	3.473
Twitter Resmi	9
Twitter Influencer	0
Youtube Resmi	89
Youtube Influencer	0

Pada table diatas diketahui bahwa data pada twitter berjumlah lebih sedikit dibandingkan dengan social media lainnya. Hal ini disebabkan karena dalam melakukan pengambilan data pada Twitter yang menggunakan *library* tweepy tidak dapat melakukan pengambilan data yang bersifat dahulu (*past*) sehingga data yang digunakan dalam pengujian adalah data yang bersifat streaming. Data twitter yang digunakan adalah data streaming yang diambil pada tanggal 26 Juni 2018 pada jam 01.00 – 02.00.

Setelah data didapatkan dilakukan kategorisasi komentar berdasarkan dinas yang telah memiliki kata kunci yang telah ditentukan. Berikut nama dinas dan kata kunci untuk masing-masing dinas

Tabel 6.5 Dinas dan Keyword Dinas Kota Depok

Nama Dinas	Kata Kunci
Badan Keuangan Daerah	Anggaran
Dinas Kearsipan dan Perpustakaan	Buku
Dinas Kependudukan dan Pencatatan Sipil	Penduduk
Dinas Kesehatan	Puskesmas

Dinas Ketahanan Pangan, Pertanian dan Perikanan	Pangan
Dinas Komunikasi dan Informatika	Teknologi
Dinas Koperasi dan Usaha Mikro	Usaha
Dinas Lingkungan Hidup dan Kebersihan	Bersih
Dinas Pekerjaan Umum dan Penataan Ruang	Infrastruktur
Dinas Pemadam Kebakaran dan Penyelamatan	Kebakaran
Dinas Pemuda, Olahraga, Kebudayaan, dan Pariwisata	Wisata
Dinas Penanaman Modal dan Pelayanan Terpadu Satu Pintu	Portal
Dinas Pendidikan	Sekolah
Dinas Perdagangan dan Perindustrian	Perdagangan
Dinas Perhubungan	Jalan
Dinas Perlindungan Anak, Pemberdayaan Masyarakat dan Keluarga	Anak
Dinas Perumahan dan Permukiman	Perumahan
Dinas Sosial	Masyarakat
Dinas Tenaga Kerja	Pekerjaan

Setelah data komentar didapatkan, langkah selanjutnya yaitu melakukan preprocessing data. Proses preprocessing yang dilakukan yaitu data cleaning, case folding, stemming, n-grams, dan stopwords removal. Pada tabel 6.5 dijelaskan jumlah komentar untuk setiap akun kota depok.

Tabel 6.6 Jumlah Komentar Setelah Preprocessing

Akun Kota Depok	Jumlah Komentar Sebelum Preprocessing	Jumlah Komentar Setelah Preprocessing
Facebook Resmi	1.038	1.002
Facebook Influencer	3.473	3.225
Twitter Resmi	9	9
Twitter Influencer	0	0
Youtube Resmi	89	89
Youtube Influencer	0	0

Data komentar pada tabel diatas menunjukkan adanya perubahan jumlah komentar pada akun Facebook resmi dan Facebook influencer. Pengurangan jumlah komentar pada akun Facebook disebabkan karena komentar yang masuk tidak memiliki teks setelah dilakukan preprocessing.

Setelah dilakukan preprocessing, maka tahapan selanjutnya yaitu melakukan pengkategorian komentar. Komentar yang memiliki kata kunci dalam teks akan diberi label sesuai dengan dinas, apabila dalam komentar ditemukan kata kunci lebih dari satu maka akan diberi label “duplicate” dan apabila tidak ditemukan kata kunci maka akan diberi label “uncategorized”. Berikut merupakan tabel hasil pelabelan komentar

Tabel 6.7 Jumlah Komentar Terkategorisasi

Akun Depok	Kota	Label Dinas	Label Duplicate	Label Uncategorized
Facebook Resmi	224	34	744	
Facebook Influencer	209	13	3003	
Twitter Resmi	7	0	2	
Twitter Influencer	0	0	0	
Youtube Resmi	8	1	80	
Youtube Influencer	0	0	0	

Berdasarkan tabel diketahui bahwa jumlah uncategorized pada akun social media facebook sangat signifikan. Penyebab hal ini yaitu kata keyword yang dimasukkan ke dalam dinas belum bisa merepresentasikan data komentar yang masuk. Hal tersebut berlaku pada akun social media Youtube dimana jumlah komentar yang terlabel uncategorized masih dominan dalam pelabelan. Pada akun social media twitter, komentar yang masuk sudah dominan dengan label dinas karena data komentar memiliki kata kunci yang sesuai dengan dinas.

Label	Persen
Dinas	10,35
Duplicate	1,11
Uncategorized	88,54

Apabila dilakukan persentase, jumlah komentar yang memiliki label dinas sebesar 10,35%, komentar yang memiliki label duplicate sebesar 1,11%, dan komentar yang memiliki label uncategorized sebesar 88,54%.

Melalui angka 10,35% pada komentar yang terkategorikan sebagai dinas, penulis tidak dapat memberikan justifikasi apakah angka tersebut bernilai bagus atau tidak. Hal ini disebabkan penamaan dinas yang berbeda-beda untuk setiap pemerintah daerah dan nantinya akan berpengaruh terhadap kata kunci untuk dinas tersebut. Sebagai contoh, Dinas Perhubungan dan Dinas Komunikasi dan Informatika pada kota Surabaya merupakan dua dinas yang terpisah sementara pada provinsi Bali Dinas Perhubungan dan Dinas Komunikasi dan Informatika dijadikan satu dinas dengan nama Dinas Perhubungan, Informasi, dan Komunikasi. Hal ini berakibat pada pemilihan kata kunci yang berbeda-beda nantinya.

Tabel 6.8 Jumlah Komentar Untuk Dinas Kota Depok

Nama Dinas	Facebook Resmi	Facebook Influencer	Twitter Resmi	Twitter Influencer	Youtube Resmi	Youtube Influencer
Badan Keuangan Daerah	0	0	0	0	0	0
Dinas Kearsipan dan Perpustakaan	0	4	1	0	0	0
Dinas Kependudukan dan Pencatatan Sipil	0	0	0	0	0	0
Dinas Kesehatan	8	1	1	0	0	0
Dinas Ketahanan Pangan, Pertanian dan Perikanan	1	0	0	0	0	0
Dinas Komunikasi dan Informatika	1	0	1	0	0	0

Dinas Koperasi dan Usaha Mikro	4	8	1	0	0	0
Dinas Lingkungan Hidup dan Kebersihan	7	9	0	0	1	0
Dinas Pekerjaan Umum dan Penataan Ruang	1	0	0	0	0	0
Dinas Pemadam Kebakaran dan Penyelamatan	0	0	0	0	0	0
Dinas Pemuda, Olahraga, Kebudayaan, dan Pariwisata	2	4	1	0	0	0
Dinas Penanaman Modal	0	0	0	0	0	0

Pelayanan Terpadu Satu Pintu						
Dinas Pendidikan	1	8	0	0	2	0
Dinas Perdagangan dan Perindustrian	0	0	0	0	0	0
Dinas Perhubungan	143	79	0	0	5	0
Dinas Perlindungan Anak, Pemberdayaan Masyarakat dan Keluarga	44	81	1	0	0	0
Dinas Perumahan dan Permukiman	0	0	0	0	0	0
Dinas Sosial	12	15	1	0	0	0
Dinas Tenaga Kerja	0	0	0	0	0	0

Berdasarkan tabel diatas diketahui bahwa Dinas Perhubungan memiliki jumlah komentar terbanyak yang disusul dengan Dinas Perlindungan Anak, Pemberdayaan Masyarakat, dan Keluarga. Hal ini menandakan bahwa keyword yang digunakan mewakili dari dinas terkait.

Komentar yang telah terkategorisasi akan disimpan ke dalam database yang berisikan id komentar, teks komentar, tanggal komentar dibuat, nama akun resmi / influencer, tipe akun, label kategori.

```
_id: "10155393094963506_10155393218693506"
comment_createddate: "2018-01-18"
comment_message: "jalan kartini tiap malam macet di area sekitar stasiun depok lama lamp..."
page_id: "pemkotdepok"
category: "Dinas Perhubungan"
account_type: "resmi"
class: "Komplain Layanan Pemerintahan"
```

Gambar 6.1 Komentar yang Masuk ke Database dengan Label Dinas

Berdasarkan gambar 6.1, terdapat komentar “jalan kartini tiap malam macet di area sekitar stasiun depok lama lamp...” kepada akun Facebook resmi pemerintah kota Depok. Karena di dalam komentar terdapat kata kunci “jalan” yang sudah diberikan oleh Dinas Perhubungan maka label kategori akan diberikan “Dinas Perhubungan”

```
_id: "10155126895313506_10155126967928506"
comment_createddate: "2017-10-18"
comment_message: "tlong anak pank di aman pak karna sngat mresahkn kami sbgai guna jalan"
page_id: "pemkotdepok"
category: "duplicate"
account_type: "resmi"
class: "Komplain Layanan Pemerintahan"
```

Gambar 6.2 Komentar yang Masuk ke Database dengan Label Duplicate

Berdasarkan gambar 6.2, terdapat komentar “tlong anak pank di aman pak karna sngat mresahkn kami sbgai guna jalan” kepada akun Facebook resmi pemerintah kota Depok. Dalam komentar tersebut terdapat dua kata kunci yaitu “anak” untuk Dinas Perlindungan Anak, Pemberdayaan Masyarakat dan Keluarga dan “jalan” untuk Dinas Perhubungan. Oleh karena komentar

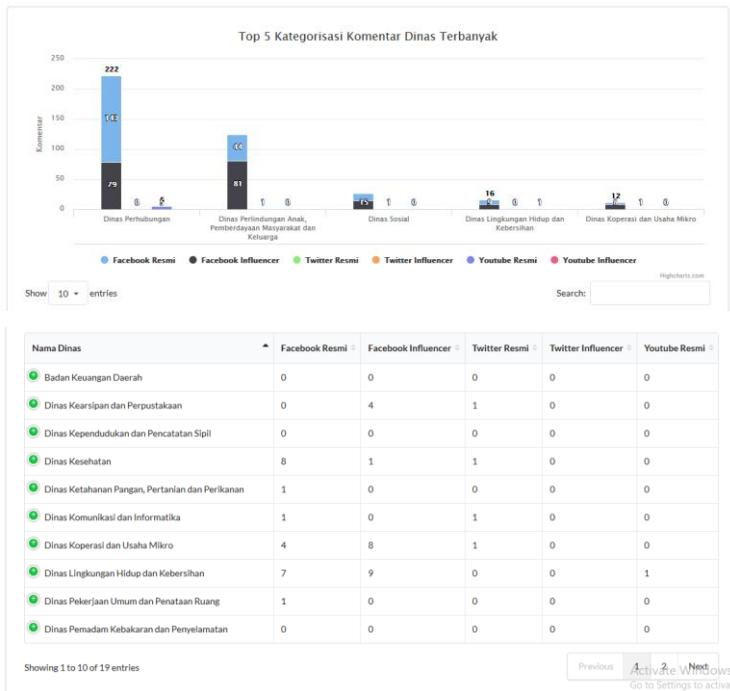
memiliki lebih dari satu kata kunci maka komentar dikategorikan sebagai “duplicate”.

```
_id: "10155415545688506_10155661365178506"  
comment_createdDate: "2018-04-24"  
comment_message: "kota depok jawa barat emang hebat"  
page_id: "pemkotdepok"  
category: "uncategorized"  
account_type: "resmi"  
class: "Apresiasi"
```

Gambar 6.3 Komentar yang Masuk ke Database dengan Label Uncategorized

Berdasarkan gambar 6.3, komentar dengan teks “kota depok jawa barat emang hebat” tidak memiliki kata kunci untuk dinas manapun. Sehingga komentar akan dikategorikan label “uncategorized”.

Setelah data komentar untuk masing-masing dinas didapatkan langkah selanjutnya yaitu melakukan visualisasi kepada user. Berikut merupakan tampilan visualisasi untuk user.



Gambar 6.4 Dashboard Kategorisasi Komentar

6.3 Rekomendasi Kata Kunci

Setelah didapatkan kata kunci maka langkah selanjutnya dilakukan pengujian terhadap rekomendasi kata kunci. Pengambilan kata kunci dilakukan dengan mengambil kata dengan nilai IDF paling rendah untuk setiap kata dalam komentar antar dinas.

Berikut merupakan kata kunci untuk dinas yang memiliki rekomendasi

Tabel 6.9 Rekomendasi Kata Kunci

Nama Dinas	Kata Kunci Rekomendasi	Relevan / Tidak	Nilai IDF
------------	------------------------	-----------------	-----------

Dinas Perhubungan	traffic light	Relevan	1.0
Dinas Perlindungan Anak, Pemberdayaan Masyarakat dan Keluarga	baca	Tidak Relevan	1.0
Dinas Sosial	nik	Tidak Relevan	1.728

Berdasarkan table diatas diketahui bahwa terdapat tiga dinas yang memiliki rekomendasi kata dari komentar yang masuk. Dinas yang tidak memiliki rekomendasi kata disebabkan karena jumlah komentar di dinas tersebut berjumlah dibawah rata-rata dari komentar yang telah terkategorisasi.

Adapun pada Dinas Perhubungan rekomendasi kata yang diberikan adalah “traffic light” dengan nilai IDF 1.0. Berdasarkan komentar yang didapatkan, kata “traffic light” relevan dengan SKPD yang ada karena komentar berupa saran masyarakat terhadap kondisi *traffic light* di kota Depok yang harus diubah. Pada Dinas Perlindungan Anak, Pemberdayaan Masyarakat, dan Keluarga rekomendasi kata yang diberikan yaitu “baca” dengan nilai IDF 1.0. Kata “baca” dalam komentar tidak relevan karena komentar menyatakan keluhan kepada pemerintah kota Depok oleh masyarakat yang membaca *billboard* di kota Depok mengatakan kalau kota Depok adalah kota ramah anak tetapi kenyataannya tidak. Terakhir pada Dinas Sosial memiliki rekomendasi kata “nik” dengan nilai IDF 3,6741. Kata kunci “nik” tidak relevan dengan SKPD karena kata ‘nik’ dalam komentar menyatakan keluhan masyarakat terhadap dibutuhkannya nik dalam proses registrasi sim *handphone*.

Ketidakrelevanan kata kunci dipengaruhi oleh kata kunci dalam melakukan kategorisasi. Hal ini menyebabkan jumlah komentar yang terkategorisasi menjadi sedikit dan mempengaruhi nilai IDF.

BAB VII

KESIMPULAN DAN SARAN

Pada bab ini dijelaskan kesimpulan dan saran dalam pengerjaan tugas akhir. Kesimpulan ditunjukkan untuk menjawab tujuan dari penelitian dan saran ditunjukkan untuk penelitian kedepannya.

7.1 Kesimpulan

Dari pelaksanaan tugas akhir didapatkan kesimpulan sebagai berikut:

1. Penelitian ini membuktikan bahwa metode streaming data dengan memanfaatkan Facebook Graph API, Twitter API, dan Youtube API mampu menjalankan proses pengumpulan data komentar pada akun resmi dan influencer di masing-masing sosial media.
2. Pengujian kategorisasi komentar yang menggunakan 4.325 komentar pada akun sosial media Kota Depok yang telah dilakukan preprocessing menghasilkan 10,35% komentar dikategorikan dinas, 1,11% komentar dikategorikan *duplicate*, dan 88,54% komentar dikategorikan *uncategorized* dengan kata kunci asumsi yang digunakan. Apabila kata kunci merepresentasikan komentar pada dinas, maka jumlah komentar yang terkategorikan dinas akan meningkat.
3. Pengujian rekomendasi kata kunci yang menggunakan 19 dinas menghasilkan 1 kata kunci untuk setiap 3 dinas. Ketiga dinas tersebut memiliki jumlah komentar diatas rata-rata komentar yang telah terkategorisasi. Adapun dari 3 kata kunci yang dihasilkan 1 diantaranya relevan sementara 2 sisanya tidak relevan. Hal ini disebabkan karena

pemilihan kata kunci dalam kategorisasi komentar yang masih umum dan belum spesifik terhadap suatu SKPD.

7.2 Saran

Pada bagian saran ini dijelaskan kekurangan dari aplikasi yang perlu diperbaiki dan dikembangkan lagi. Berikut ini saran untuk penelitian selanjutnya.

1. Perlu dilakukan pemilihan kata kunci yang lebih tepat untuk dinas agar jumlah komentar yang terkategori dinas dapat meningkat.
2. Perlu dilakukan pengujian lebih lanjut dengan menggunakan data komentar yang terkategori dinas yang lebih banyak agar nilai IDF dapat bervariasi sehingga rekomendasi kata kunci dapat semakin akurat.
3. Pemilihan kata kunci dengan menggunakan nilai IDF masih sederhana oleh sebab itu dapat dilakukan rekomendasi kata kunci dengan menggunakan *machine learning*.

DAFTAR PUSTAKA

- [1] Republik Indonesia, “UNDANG-UNDANG REPUBLIK INDONESIA NOMOR 32 TAHUN 2004 TENTANG PEMERINTAHAN DAERAH,” no. 1, pp. 1–7, 2004.
- [2] D. I. N. Indonesia, “Digital in India a Snapshot of the Country ’ S Key Digital Statistical Indicators,” *Hootsuite*, pp. 199–213, 2018.
- [3] TETRA PAK, “the Tetra Pak Index 2017,” no. 10, 2017.
- [4] S. Alam and A. Diamah, “Understanding user participation in Australian Government tourism Facebook page,” ... , *Locat. Locat. Proc. 23rd* ... , no. 1997, pp. 1–11, 2012.
- [5] E. Satriya, “PENTINGNYA REVITALISASI E-GOVERNMENT Asisten Deputi 5 / V Deputi Bidang Koordinasi Infrastruktur dan Pengembangan Wilayah Kementerian Koordinator Bidang Perekonomian Gedung Induk , Lt III Jl . Lapangan Banteng Timur 2-4 , Jakarta , 10710 , Indonesia,” pp. 38–43, 2006.
- [6] E. A. Sosiawan, “Evaluasi Implementasi E-Government Pada Situs Web Pemerintah Daerah Di Indonesia: Prespektif Content Dan Manajemen,” *Semin. Nas. Inform.*, vol. 2008, no. November 2007, pp. 88–98, 2008.
- [7] Menteri Komunikasi dan Informatika Republik Indonesia, ‘PERATURAN MENTERI KOMUNIKASI DAN INFORMATIKA REPUBLIK INDONESIA NOMOR 21 TAHUN 2016 TENTANG PERUBAHAN ATAS PERATURAN MENTERI KOMUNIKASI DAN INFORMATIKA NOMOR 22 TAHUN 2015 TENTANG RENCANA STRATEGIS KEMENTERIAN KOMUNIKASI DAN

- INFORMATIKA TAHUN 2015-2019.” 2016.
- [8] B. Supradono and A. N. Hanum, “Peran Sosial Media untuk Manajemen Hubungan dengan Pelanggan pada Layanan E-Commerce,” *Vallue Added*, vol. 7, no. 2, pp. 33–45, 2011.
 - [9] danah m. boyd and N. B. Ellison, “Social Network Sites: Definition, History, and Scholarship,” *J. Comput. Commun.*, vol. 13, no. 1, pp. 210–230, Oct. 2007.
 - [10] “Facebook: One Billion and Counting - WSJ.” [Online]. Available: <https://www.wsj.com/articles/SB10000872396390443635404578036164027386112>. [Accessed: 25-Jan-2018].
 - [11] N. Shah, P. Purayil Sneha, and S. Chattapadhyay, *Digital Activism in Asia Reader*. 2015.
 - [12] H. Kwak, C. Lee, H. Park, and S. Moon, “What is Twitter , a Social Network or a News Media ? Categories and Subject Descriptors,” *19th Int. Conf. World Wide Web*, pp. 591–600, 2010.
 - [13] A. N. Smith, E. Fischer, and C. Yongjian, “How Does Brand-related User-generated Content Differ across YouTube, Facebook, and Twitter?,” *J. Interact. Mark.*, vol. 26, no. 2, pp. 102–113, 2012.
 - [14] P. Gill, M. F. Arlitt, Z. Li, and A. Mahanti, “Youtube traffic characterization: a view from the edge,” *Proc. 7th {ACM} {SIGCOMM} Internet Meas. Conf. {IMC} 2007, San Diego, California, USA, Oct. 24-26, 2007*, pp. 15–28, 2007.
 - [15] “YouTube serves up 100 million videos a day online - USATODAY.com.” [Online]. Available: https://usatoday30.usatoday.com/tech/news/2006-07-16-youtube-views_x.htm. [Accessed: 25-Jan-2018].
 - [16] G. Kruitbosch and F. Nack, “Broadcast yourself on YouTube – Really?”

- [17] J. Burgess and J. Green, “Youtube (Digital Media and Society Series),” *Mediekultur*, vol. 25, no. 47, pp. 133–135, 2009.
- [18] D. R. Tobergte and S. Curtis, “Regular Expression Pocket Reference,” *Journal of Chemical Information and Modeling*, vol. 53, no. 9, pp. 1689–1699, 2013.
- [19] D. Jurafsky and J. H. Martin, “N-Grams,” *Speech Lang. Process.*, pp. 2–7, 2014.
- [20] R. Baeza-Yates, C. Castillo, M. Marin, and A. Rodriguez, “Crawling a country: better strategies than breadth-first for web page ordering,” *Spec. Interes. tracks posters 14th Int. Conf. World Wide Web*, pp. 864–872, 2005.
- [21] M. Thelwall, “A web crawler design for data mining,” *J. Inf. Sci.*, vol. 27, no. 5, pp. 319–325, 2001.
- [22] H. M. Kienle and H. A. Muiller, “Requirements of Software Visualization Tools: A Literature Survey.”
- [23] I. J. Asmara, E. Achelia, W. Maulana, R. Wijayanti, and Y. Rianto, “Teknik Visualisasi Grafik Berbasis Web Di Atas Platform Open Source,” *Semin. Nas. Apl. Teknol. Inf. 2009 (SNATI 2009)*, vol. 2009, no. Snati, pp. 44–47, 2009.
- [24] O. Karmayasa and I. B. Mahendra, “Implementasi VSM dan Notasi Metode TF-IDF pada SIstem Temu Kembali Informasi,” *J. Progr. Stud. Tek. Inform.*, p. 9, 2010.
- [25] M. Fitri, “PERANCANGAN SISTEM TEMU BALIK INFORMASI DENGAN METODE PEMBOBOTAN KOMBINASI TF-IDF UNTUK PENCARIAN DOKUMEN BERBAHASA INDONESIA,” *Peranc. Sist. Temu Balik Inf. Dengan Metod. Pembobotan Komb. Tf-Idf Untuk Pencarian Dok. Berbahasa Indones.*, 2013.
- [26] V. Srividhya and R. Anitha, “Evaluating preprocessing

- techniques in text categorization,” *Int. J. Comput. Sci. Appl.*, no. 2010, pp. 49–51, 2010.
- [27] X. B. Xue and Z. H. Zhou, “Distributional features for text categorization,” *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 3, pp. 428–442, 2009.
 - [28] K. pendayagunaan aparatur negara dan reformasi birokrasi republik Indonesia, “PEDOMAN PEMANFAATAN MEDIA SOSIAL INSTANSI PEMERINTAH,” 2012.
 - [29] G. Wang *et al.*, “Building a Replicated Logging System with Apache Kafka,” *Proc. VLDB Endow.*, vol. 8, no. 12, pp. 1654–1655, 2015.
 - [30] N. Garg, *Apache Kafka*. 2013.
 - [31] P. DEGANANDA FERDIAN, *RANCANG BANGUN APLIKASI BACKEND DAN WEBSERVICES PENYAMPAIAN LAPORAN MASYARAKAT BERBASIS CROWDSOURCING MELALUI JEJARING SOSIAL TWITTER*. 2016.
 - [32] G. K. Prakoso, “RANCANG BANGUN APLIKASI UNTUK KLASIFIKASI POST PADA SOSIAL MEDIA PEMERINTAH DAERAH DI INDONESIA MENGGUNAKAN SUPPORT VECTOR MACHINE (SVM),” 2018.
 - [33] D. Hanindito, “Rancang Bangun Sistem Penilaian Reaksi Masyarakat terhadap Akun Sosial Media Pemerintah Daerah,” 2018.
 - [34] M. F. HAZMI, “RANCANG BANGUN APLIKASI UNTUK KLASIFIKASI KOMENTAR NETIZEN PADA MEDIA SOSIAL PEMERINTAH DAERAH DI INDONESIA MENGGUNAKAN ALGORITMA RANDOM FOREST,” 2018.

LAMPIRAN A

TAMPILAN ANTAR MUKA APLIKASI

The screenshot shows a web-based application interface for user registration. At the top left is a navigation bar labeled 'Kategorisasi Admin'. On the top right, there is an email address 'admin@admin.com' and a dropdown arrow. The main title 'Register' is centered above the form fields. The form itself consists of several input fields:

- A dropdown menu set to 'KOTA SURABAYA'.
- An email input field containing 'surabaya@gmail.com'.
- A password input field showing '*****'.
- A confirmation password input field also showing '*****'.
- An 'Upload Surat Tugas' section with a 'Browse...' button followed by the file path '161-342-1-PB.pdf'.
- A text input field containing 'Suryondi'.
- A large teal-colored 'Register' button at the bottom of the form area.

At the bottom of the page, there is a link 'Sudah Mendaftar? Login' and a watermark 'Activate Windows'.

Gambar 9.1 Tampilan Regier Akun Pemda oleh Admin

The screenshot shows a web-based administrative interface. At the top left is a 'Kategorisasi Admin' button. On the top right, there is an email address 'admin@admin.com' followed by a dropdown arrow. Below the header, a breadcrumb navigation shows 'Admin/Pemda / KOTA SURABAYA'. The main title 'Edit Pemerintah Daerah' is centered above a form. The form contains a dropdown menu set to 'KOTA SURABAYA', an input field containing the email 'kotasurabaya@gmail.com', and a large teal 'Edit' button. At the bottom of the page, a footer reads 'E-Gov Benchmark Departemen Sistem Informasi 2018'.

Gambar 9.2 Tampilan Edit Akun Pemerintah Daerah Oleh Admin

Kategorisasi Admin					admin@admin.com	
					Deleted Pemda	Tambah Pemda
					Search:	
Admin / Pemda						
Show	10	entries				
Nama Pemda	Email Pemda	Nama Petugas	Surat Tugas	Action		
KOTA AMBON	ambon@gmail.com	nody	Download			
KOTA DEPOK	nodyriskypriatomo@gmail.com	nody	Download			
KOTA SURABAYA	kotasurabaya@gmail.com	Pegawai 1	Download			
PROVINSI NAD	nad@gmail.com	nody	Download			
Showing 1 to 4 of 4 entries					Previous	1
E-Gov Benchmark Departemen Sistem Informasi 2018						

Gambar 9.3 Tampilan Menu Lihat Dinas, Edit Pemda, dan Hapus Akun Pemda oleh Admin

Kategorisasi Admin		admin@admin.com	
Admin / Pemda / Deleted			
Show	10	entries	Search:
Nama Pemda	Email Pemda	Action	
KOTA AMBON	ambon@gmail.com	 	

Showing 1 to 1 of 1 entries

Previous 1 Next

E-Gov Benchmark Departemen Sistem Informasi 2016

Gambar 9.4 Tampilan Menu Restore Akun dan Hapus Permanen Pemerintah Daerah oleh Admin

Tambah Dinas

Nama Dinas *

Dinas Kebersihan

Deskripsi Dinas *

Dinas yang mengatur kebersihan kota di kota Depok

Keyword *

Masukkan satu kata kunci untuk dinas ini. Contoh : jalan

sampah

cancel

submit

Active
Go to S...

The screenshot shows a web-based administrative interface for adding a new agency. The main title is 'Tambah Dinas'. The first field is 'Nama Dinas' (Agency Name), which is required (*). The input field contains 'Dinas Kebersihan'. The second field is 'Deskripsi Dinas' (Description), also required (*), with the input 'Dinas yang mengatur kebersihan kota di kota Depok'. The third field is 'Keyword', which is also required (*), with the input 'sampah'. At the bottom right, there are two buttons: 'cancel' and 'submit'. A small green circular icon with a white letter 'G' is located on the right side of the input field for 'Deskripsi Dinas'.

Gambar 9.5 Tampilan Tambah Dinas oleh Admin

Kategorisasi Admin

admin@admin.com

Edit Dinas

Nama Dinas *

Deskripsi Dinas *

Keyword *

Edit Dinas

E-Gov Benchmark Departemen Sistem Informasi 2018

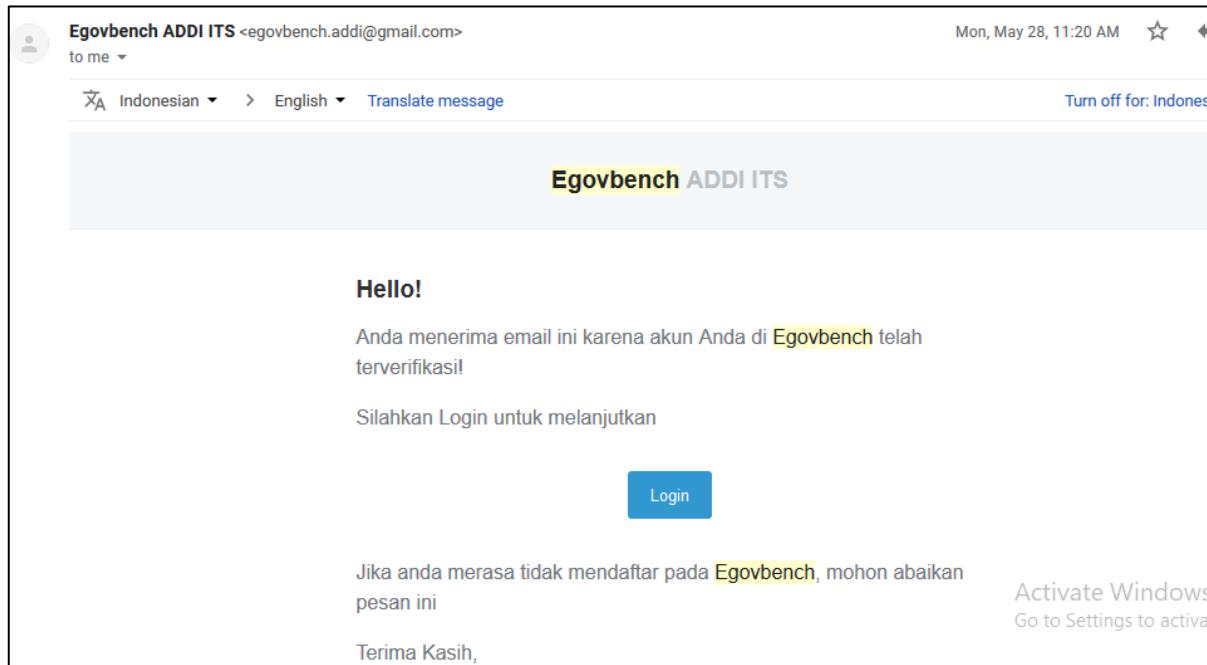
Activate Windows

A screenshot of a web-based administrative interface. At the top left is the text "Kategorisasi Admin". On the top right is an email address "admin@admin.com" followed by a dropdown arrow. The main title "Edit Dinas" is centered above a form. The form contains three input fields: "Nama Dinas" with the value "Badan Keuangan Daerah", "Deskripsi Dinas" with the value "Badan yang menaungi urusan keuangan kota Depok", and "Keyword" with the value "anggaran". Below the form is a large teal button labeled "Edit Dinas". At the bottom of the page, there is footer text "E-Gov Benchmark Departemen Sistem Informasi 2018" and a watermark "Activate Windows".

Gambar 9.6 Tampilan Edit Dinas oleh Admin



Gambar 9.7 Tampilan Menu Hapus Dinas oleh Admin



Gambar 9.8 Tampilan Email Akun Pemda Telah Diverifikasi

 Egovbench ADDI ITS <egovbench.addi@gmail.com>
to me ▾

Indonesian ▾ English ▾ Translate message Turn off for: Indonesian ×

Egovbench ADDI ITS

Hello!

Anda menerima email ini karena akun Anda di **Egovbench** tidak terverifikasi karena ketidakvalidan berkas.

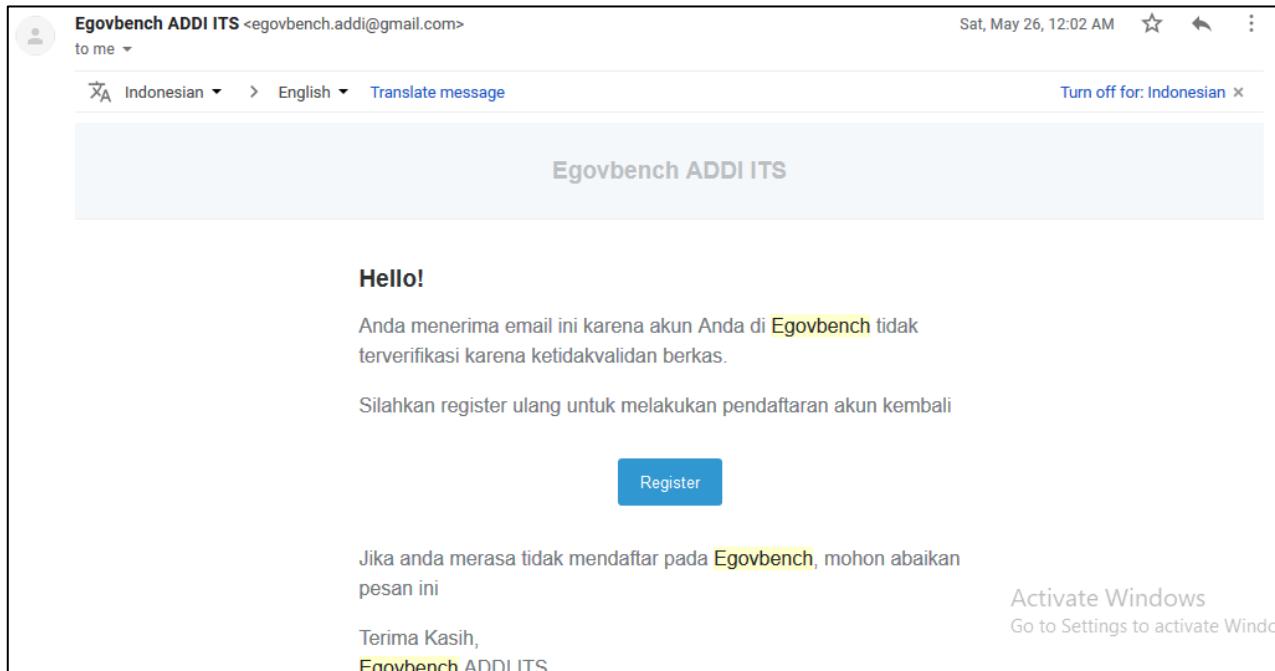
Silahkan register ulang untuk melakukan pendaftaran akun kembali

[Register](#)

Jika anda merasa tidak mendaftar pada **Egovbench**, mohon abaikan pesan ini

Terima Kasih,
Egovbench ADDI ITS

Activate Windows
Go to Settings to activate Windo



Gambar 9.9 Tampilan Email Akun ditolak

The screenshot shows a registration page with the following fields:

- A dropdown menu labeled "KOTA DEPOK" containing "KOTA DEPOK".
- An email input field containing "nodyriskypatomo@gmail.com".
- A password input field showing masked text.
- A confirmation password input field showing masked text.
- An "Upload Surat Tugas" section with a "Browse..." button and a file path "161-342-1-PB.pdf".
- A name input field containing "Nody".
- A large teal "Register" button.

Below the form, there is a link "Sudah Mendaftar? [Login](#)". On the right side of the page, there is a watermark for "Activate Windows" and a footer note "E-Gov Benchmark Departemen Sistem Informasi 2018".

Gambar 9.10 Tampilan Register Pemerintah Daerah



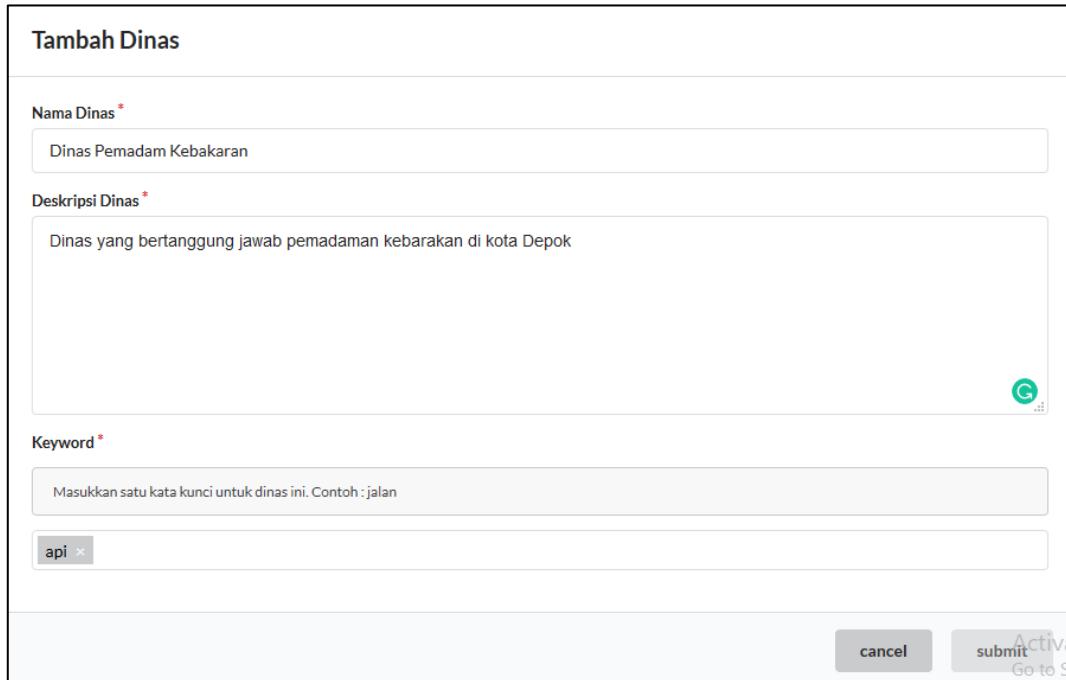
Gambar 9.11 Tampilan Edit Akun Pemerintah Daerah Oleh User

Tambah Dinas

Nama Dinas *
Dinas Pemadam Kebakaran

Deskripsi Dinas *
Dinas yang bertanggung jawab pemadaman kebakaran di kota Depok

Keyword *
Masukkan satu kata kunci untuk dinas ini. Contoh : jalan
api



Gambar 9.12 Tampilan Tambah Dinas Oleh User

Beranda About Peringkat ▾ Data ▾ Metodologi ▾ Klasifikasi Kategorisasi

nodyriskypatomo@gmail.com ▾

Edit Dinas

Nama Dinas *

Badan Keuangan Daerah

Deskripsi Dinas *

Badan yang menaungi urusan keuangan kota Depok

Keyword *

anggaran ×

Edit Dinas

E-Gov Benchmark Departemen Sistem Informasi 2018

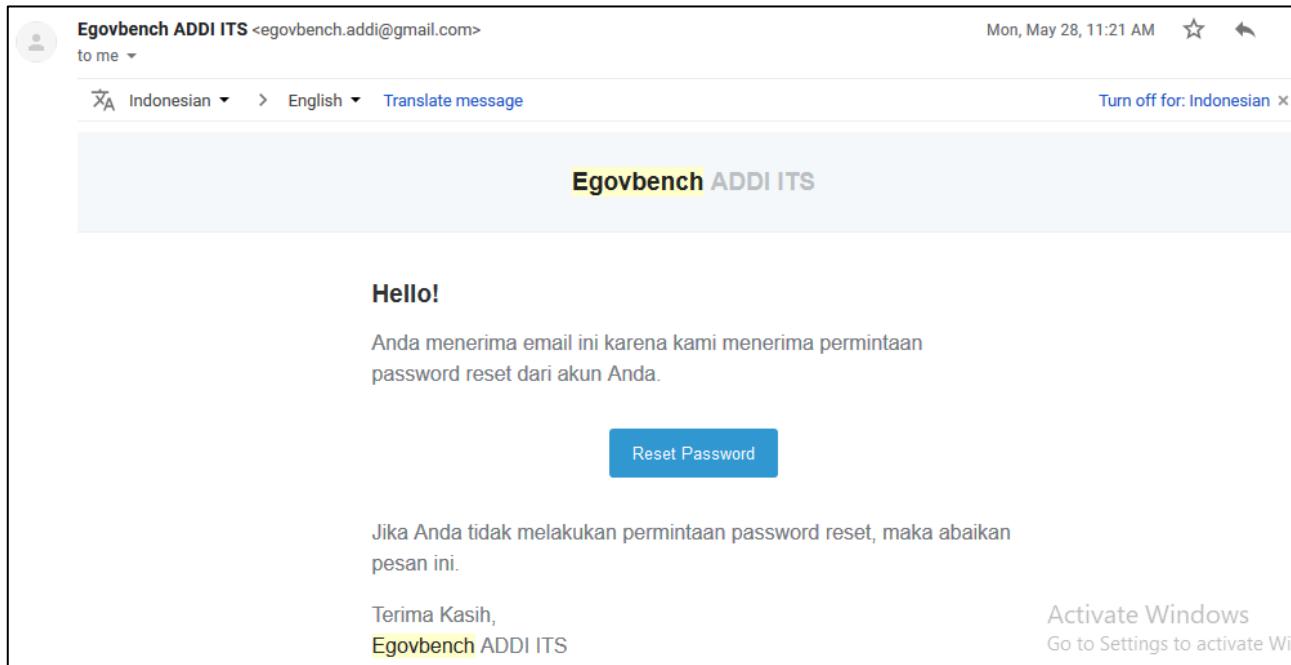
Activate Windows

A screenshot of a web-based application titled 'Edit Dinas'. The interface includes a navigation bar with links for Beranda, About, Peringkat, Data, Metodologi, Klasifikasi, and Kategorisasi. A user email, nodyriskypatomo@gmail.com, is displayed in the top right corner. The main content area features three input fields: 'Nama Dinas' containing 'Badan Keuangan Daerah', 'Deskripsi Dinas' containing 'Badan yang menaungi urusan keuangan kota Depok', and 'Keyword' containing 'anggaran'. A large teal button at the bottom right is labeled 'Edit Dinas'. At the bottom of the page, there is footer text: 'E-Gov Benchmark Departemen Sistem Informasi 2018' and a watermark for 'Activate Windows'.

Gambar 9.13 Tampilan Edit Dinas Oleh User

Nama Dinas	Deskripsi Dinas	Keyword	Action
Badan Keuangan Daerah	Badan yang menaungi urusan keuangan kota Depok	anggaran	
Dinas Kearsipan dan Perpustakaan	Dinas yang menaungi sektor karsipan dan pustaka kota Depok	buku	
Dinas Kependudukan dan Pencatatan Sipil	Dinas yang menaungi sektor pencatatan penduduk kota Depok	penduduk	

Gambar 9.14 Tampilan Menu Hapus dan Edit Dinas



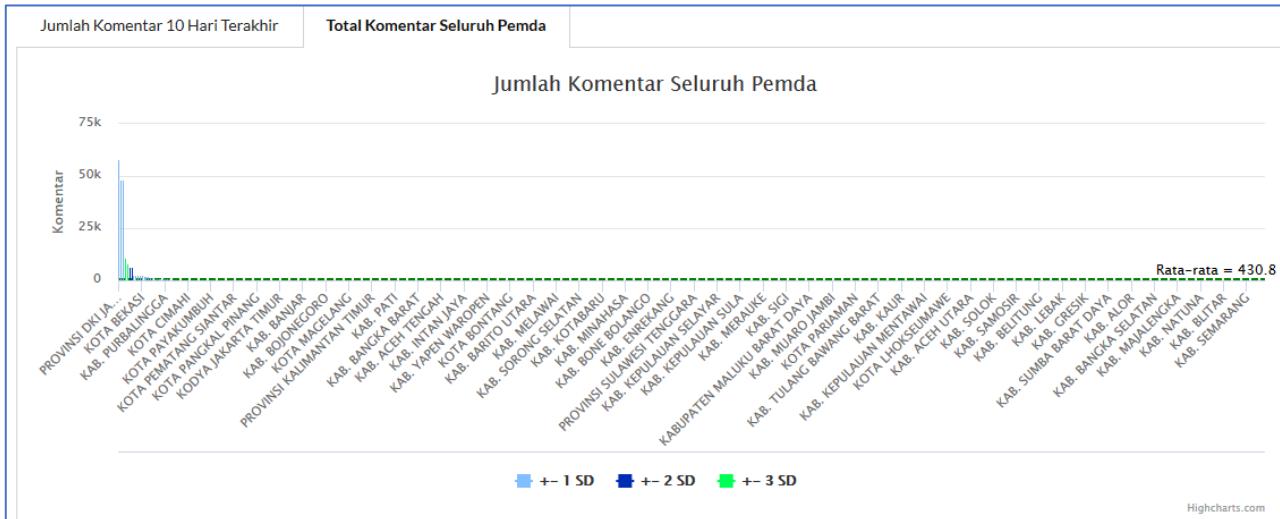
Gambar 9.15 Tampilan Email Permintaan Reset Password

The screenshot shows a web page with a light gray header bar. On the left is a 'Beranda' link, and on the right are 'Login' and 'Register' buttons. Below the header, the main content area has a title 'Reset Password' in bold teal text. Underneath the title is a form field with a placeholder 'Email' and a small user icon. At the bottom of the form is a large teal button with the white text 'Send Password Reset Link'.

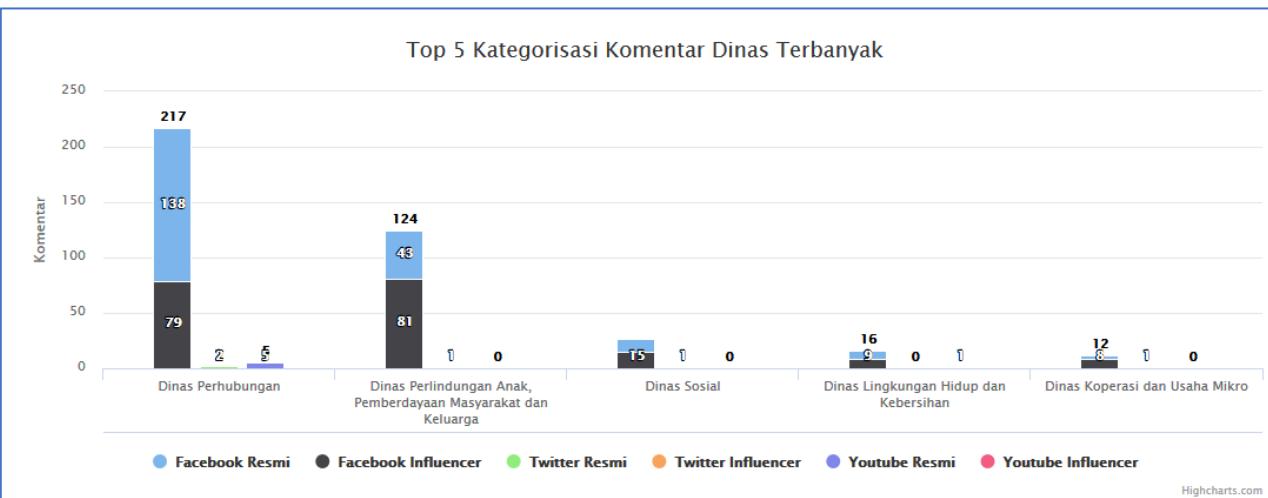
Gambar 9.16 Tampilan Reset Password



Gambar 9.17 Grafik Jumlah Komentar 10 Hari Terakhir



Gambar 9.18 Jumlah Komentar Seluruh Pemda



Gambar 9.19 Top 5 Kategorisasi Komentar Dinas Terbanyak

Nama Dinas	Facebook Resmi	Facebook Influencer	Twitter Resmi	Twitter Influencer	Youtube Resmi	
+	Dinas Perhubungan	138	79	2	0	5
+	Dinas Perlindungan Anak, Pemberdayaan Masyarakat dan Keluarga	43	81	1	0	0
+	Dinas Sosial	12	15	1	0	0
+	Dinas Kesehatan	8	1	2	0	0
+	Dinas Lingkungan Hidup dan Kebersihan	7	9	0	0	1
+	Dinas Koperasi dan Usaha Mikro	4	8	1	0	0
+	Dinas Ketahanan Pangan, Pertanian dan Perikanan	1	0	0	0	0
+	Dinas Komunikasi dan Informatika	1	0	2	0	0
+	Dinas Pekerjaan Umum dan Penataan Ruang	1	0	0	0	0
+	Dinas Pendidikan	1	8	0	0	2

Showing 1 to 10 of 19 entries

[Previous](#)
1
2
[Next](#)

Gambar 9.20 Tabel Jumlah Komentar Setiap Dinas

Beranda	About	Peringkat ▾	Data ▾	Metodologi ▾	Klasifikasi	Kategorisasi	Open Data	kotadepok@gmail.com ▾
KOTA DEPOK / Komentar Kategorisasi Dinas								
Facebook Resmi	Facebook Influencer	Twitter Resmi	Twitter Influencer	Youtube Resmi	Youtube Influencer			
Show <input type="button" value="10"/> entries						Search:		
Komentar	Dinas	Action						
aahhh aq prnh wa beliau lsg trya soal kia blgnya mo d cek mo d cek knyataannya smpe skg g da kbr berita tuh smpe anak skg ud th kaga ada jelas tuh program pcma buka hotline adu jd	Dinas Perlindungan Anak, Pemberdayaan Masyarakat dan Keluarga							
acara tarik moga anak juga senang sesuai visi sbg kota layak anak	Dinas Perlindungan Anak, Pemberdayaan Masyarakat dan Keluarga							
ada informasi lowong kah utk dokter di puskesmas sebut	Dinas Kesehatan							
	Dinas Perlindungan							

Gambar 37 Tampilan Komentar Kategori Dinas

The screenshot shows a web application interface with a navigation bar at the top. The navigation bar includes links for Beranda, About, Peringkat, Data, Metodologi, Klasifikasi, Kategorisasi, Open Data, and an email address kotadepok@gmail.com. The 'Kategorisasi' link is highlighted in grey, indicating it is the active page.

The main content area features a title 'Edit Komentar Facebook' in blue text. Below the title is a form field labeled 'Komentar*' containing a text area with the following text:

aahhh aq prnh wa beliau lsg tnya soal kia blgnya mo d cek mo d cek knyataannya smpe skg g da
kbr berita tuh smpe anak skg ud th kaga ada jelas tuh program pcma buka hotline adu jd

Below the comment field is another form field labeled 'Dinas*' with a dropdown menu showing the option 'Dinas Perlindungan Anak, Pemberdayaan Masyarakat dan Keluarga'.

A large teal button at the bottom of the form contains the text 'Edit Komentar'.

At the very bottom of the page, there is a footer note: 'E-Gov Benchmark Departemen Sistem Informasi 2018'.

Gambar 38 Edit Komentar Kategori Dinas

Beranda About Peringkat ▾ Data ▾ Metodologi ▾ Klasifikasi Kategorisasi Open Data kotadepok@gmail.com ▾

⌂ KOTA DEPOK / Komentar Duplicate

Facebook Resmi Facebook Influencer Twitter Resmi Twitter Influencer Youtube Resmi Youtube Influencer

Show 10 entries Search:

Komentar	Pilih Dinas	Pilih Semua Dinas
terjadi penertiban di jalan ITS	<button>Dinas Perhubungan</button> <button>Satpol PP</button>	<button>Semua Dinas</button>

Showing 1 to 1 of 1 entries Previous 1 Next

E-Gov Benchmark Departemen Sistem Informasi 2018

Gambar 39 Komentar Kategori Duplicate

Beranda	About	Peringkat ▾	Data ▾	Metodologi ▾	Klasifikasi	Kategorisasi	Open Data	kotadepok@gmail.com ▾		
KOTA DEPOK / Komentar Uncategorized										
Facebook Resmi Facebook Influencer Twitter Resmi Twitter Influencer Youtube Resmi Youtube Influencer										
Show	10 ▾	entries								
						Search:				
Komentar	Action									
a at ruhiyat tamlikho tam bunda nicoaurel teguh kartono rifka aulia al jupri nur haqmaliawan handy liauw indrian handi apif kahpiyudin arief herlambang mohon maaf atas ketidaknyamanannya kritik dan saran sdr sdri akan kami bantu sampai kepada dinas kait dalam hal ini dishub terima kasih	Pilih Dinas	✓								
abid tsabit ada bokap ente sob cc muhammad doang	Pilih Dinas	✓								
acara tarik banget nih gratis ada merchandise juga lho yuk datang ramai acara ini sumber	Pilih Dinas	✓								
ada info tentang mqk kemarin	Pilih Dinas	✓								

Gambar 40 Komentar Kategori Uncategorized

BIODATA PENULIS



Penulis lahir di kota Depok pada tanggal 23 April 1996. Anak pertama dari dua bersaudara ini telah menempuh pendidikan formal yaitu: SD Tugu Ibu 1 Depok, SMP Negeri 3 Depok, dan SMA Negeri 1 Depok.

Pada tahun 2014, penulis melanjutkan pendidikan ke jenjang yang lebih tinggi, yaitu di Institut Teknologi Sepuluh Nopember (ITS) Surabaya, sebagai mahasiswa departemen Sistem Informasi, Fakultas Teknologi Informasi dan Komunikasi (FTIK) dan terdaftar dengan NRP 05211440000166. Selama menjadi mahasiswa, penulis mengikuti kegiatan kemahasiswaan, antara lain seminar, organisasi dan asisten dosen. Penulis diberikan tanggung jawab oleh organisasi BEM Fakultas Teknologi Informasi dan Komunikasi (BEM FTIK) sebagai Kepala Departemen Entrepreneurship pada tahun kepengurusan 2016/2017. Disamping aktif dalam kegiatan kemahasiswaan, penulis juga pernah menjadi Asisten Praktikum mata kuliah Perencanaan Sumber Daya Perusahaan dan Desain Basis Data.

Pada tahun ke-4, penulis tertarik dengan bidang Social Media Analysis, sehingga mengambil bidang minat laboratorium Akuisisi Data dan Diseminasi Informasi (ADDI) dan lulus dalam waktu 4 tahun atau 8 semester. Penulis dapat dihubungi melalui email nodirskypratomo@gmail.com

