

Analisis Sentimen Masyarakat pada Media Sosial Twitter Terhadap Kebijakan Larangan Mudik dengan Algoritma Boyer-Moore dan *Regular Expression*

Dionisius Darryl Hermansyah 13519058
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail: dionisius.dh28@gmail.com

Abstrak—Pandemi COVID-19 berdampak besar pada aktivitas manusia di seluruh dunia. Di masa pandemi ini, berbagai upaya telah dilakukan oleh pemerintah Indonesia dalam memutus rantai penyebaran COVID-19. Hal ini tidak terlepas dari larangan mudik yang diberlakukan pemerintah berturut-turut pada tahun 2020 dan 2021. Larangan mudik ini menuai berbagai opini yang sangat bervariasi dari berbagai kalangan masyarakat. Oleh karena itu, kajian kebijakan pemerintah menggunakan opini dari masyarakat luas sangatlah krusial dalam keberhasilan aturan terkait. Dengan hadirnya media sosial seperti Twitter, maka data-data opini masyarakat dalam bentuk *tweet* dapat diambil dan dianalisis sentimennya menggunakan algoritma pencocokan *string* Boyer-Moore dan *regular expression* sebagai bahan kajian dan peninjauan bagi pemerintah. Tahapan-tahapan analisis yang dilaksanakan adalah *data scraping* menggunakan kaskas Twint, pembersihan data menggunakan *regular expression*, dan analisis sentimen dengan memanfaatkan algoritma Boyer-Moore. Hasil analisis sentimen menunjukkan bahwa mayoritas *tweet* memiliki sentimen negatif terhadap kebijakan larangan mudik, dimana, ditemukan fakta bahwa 32% *tweet* memiliki sentimen positif, 48% *tweet* memiliki sentimen negatif, dan 20% *tweet* memiliki sentimen netral. Temuan ini diharapkan dapat membantu pemerintah dalam mengkaji kebijakan larangan mudik yang telah diterapkan berdasarkan respons masyarakat di media sosial Twitter.

Keywords—Mudik, Twitter, Analisis Sentimen, Pencocokan String, Regular Expression

I. LATAR BELAKANG

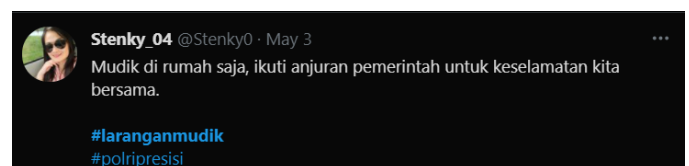
Pandemi COVID-19 (*Coronavirus Disease*) yang disebabkan oleh virus SARS-CoV-2 berdampak besar pada setiap aktivitas yang dilakukan oleh manusia di seluruh penjuru dunia, tak terkecuali negara Indonesia. Di masa pandemi ini, berbagai upaya telah dilakukan oleh pemerintah Indonesia dalam memutus rantai penyebaran COVID-19. Salah satu caranya adalah dengan melakukan *physical distancing*, *lockdown*, dan himbauan bagi masyarakat untuk meminimalisir aktivitas di luar rumah. Hal ini tidak terlepas dari larangan mudik yang diberlakukan pemerintah berturut-turut pada tahun 2020 dan 2021.

Mudik yang secara harafiah berarti “pulang kampung” merupakan sebuah tradisi yang selalu dilakukan oleh masyarakat Indonesia menjelang perayaan hari raya keagamaan

atau peringatan besar lainnya. Pada umumnya, salah satu hari raya yang melibatkan kegiatan mudik dalam skala yang besar adalah perayaan Idul Fitri. Kegiatan ini seringkali juga disebut sebagai mudik lebaran.

Larangan mudik ini menuai berbagai opini yang sangat bervariasi dari berbagai kalangan masyarakat. Beberapa bagian masyarakat setuju dengan larangan mudik ini demi meminimalisir resiko penyebaran virus COVID-19, namun tak sedikit pula yang menentang kebijakan pemerintah ini.

Opini-opini tersebut disampaikan melalui berbagai media publik, namun salah satu sumber penyumbang opini terbesar adalah melalui jaringan media sosial. Twitter, merupakan salah satu media sosial dengan pengguna terbesar di Indonesia. Pengguna Twitter dapat menyampaikan isi pikiran mereka secara publik melalui pesan teks, gambar, atau video yang disebut juga dengan *tweet*. Jumlah *tweet* yang dibuat setiap harinya memiliki jumlah yang sangat masif yaitu lebih dari 500.000.000 *tweet* di seluruh dunia.



(a)



(b)

Gambar 1. Berbagai opini (a) positif dan (b) negatif mengenai kebijakan larangan mudik di media sosial Twitter (Sumber: Twitter)

Jumlah data tak beraturan dalam bentuk *tweet* yang besar ini dapat dimanfaatkan untuk melakukan analisis sentimen masyarakat terhadap kebijakan larangan mudik oleh pemerintah di masa pandemi COVID-19 ini. Dengan

menerapkan dan memanfaatkan algoritma pencocokan *string* seperti algoritma Boyer-Moore dan *regular expression*, data *tweet* yang ada dapat diolah dan dianalisis sentimennya untuk menghasilkan sebuah data yang dapat digunakan oleh pemerintah dalam meninjau kebijakan larangan mudik yang telah diterapkan berdasarkan respon dari masyarakat pada media sosial Twitter.

II. DASAR TEORI

A. Mudik

Penduduk di sebuah negara dapat melakukan sebuah kegiatan mobilitas yang melintasi batas wilayah tertentu dalam waktu tertentu baik bersifat horizontal atau geografis. Salah satu contoh mobilitas masyarakat di negara Indonesia adalah mudik.



Gambar 2. Pemudik tahun 2019 mencapai 23 juta jiwa (Sumber: Lokadata)

Mudik atau “pulang kampung” merupakan sebuah tradisi yang selalu dilakukan oleh masyarakat Indonesia menjelang perayaan hari raya keagamaan atau peringatan besar lainnya. Pada umumnya, salah satu hari raya yang melibatkan kegiatan mudik dalam skala yang besar adalah perayaan Idul Fitri. Kegiatan ini seringkali juga disebut sebagai mudik lebaran. Jangka waktu mudik seseorang ke tempat asalnya dapat sangat bervariasi tergantung pada masa libur yang diberlakukan oleh perusahaan tempat bekerja dan institusi pemerintahan setempat.

Pemudik atau orang yang melakukan mudik memiliki jumlah yang sangat besar di negara Indonesia. Pada tahun sebelum terjadinya pandemi, yaitu tahun 2019, jumlah pemudik mencapai angka hingga 23 juta jiwa, sedangkan di masa pandemi 2020, angka jumlah pemudik menurun hingga 465 ribu pemudik.

B. Twitter

Twitter merupakan sebuah media sosial yang memberikan layanan bagi penggunanya untuk mengirim, membaca, dan melihat pesan berbasis teks yang disebut juga dengan *tweet*. Pesan berbasis teks ini umumnya juga dapat disertai dengan file audio-visual seperti gambar atau video. *Tweet* yang dikirimkan oleh sebuah pengguna Twitter akan bersifat publik dan dapat dilihat oleh pengguna lainnya. Oleh karena itu, Twitter sebagai sebuah media sosial umumnya dimanfaatkan oleh masyarakat dalam memberikan pendapat dan respons

terkait hal-hal yang sedang menjadi topik hangat di dalam jaringan (*trending topic*). Salah satu negara pengguna harian Twitter terbanyak adalah negara Indonesia.

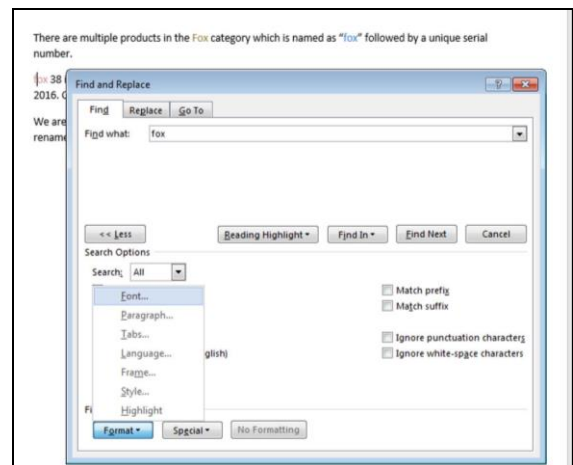
C. Analisis Sentimen

Analisis sentimen atau *sentiment analysis* merupakan sebuah teknik dalam mengidentifikasi ekspresi sebuah sentimen yang digunakan di dalam sebuah teks. Proses identifikasi ini juga menyangkut pengkategorian sentimen ke dalam 3 kelas yaitu, sentimen positif, negatif, dan netral. Sebuah teks yang memiliki sentimen positif berarti, sifat emosi di dalam teks tersebut cenderung bersifat positif seperti menyukai, mendukung, dan memuji. Sebuah teks dikatakan memiliki sentimen negatif jika sifat emosi yang diidentifikasi adalah menolak, memaki, marah, dan sebagainya. Jika teks tersebut tidak memiliki emosi atau memiliki kestabilan emosi positif dan negatif yang berimbang maka teks tersebut dapat dikategorikan memiliki sentimen netral.

Analisis sentimen masuk ke dalam cabang ilmu pengolahan bahasa alami, linguistik komputer, dan penambangan teks yang memiliki tujuan utama untuk menganalisa pendapat, sikap, penilaian, dan emosi seseorang terhadap sebuah topik tertentu. Tentunya, hal ini akan sangat berguna untuk mengkonversi data tak beraturan seperti teks menjadi sesuatu yang lebih bermakna, contohnya, konversi komentar sebuah produk menjadi berbentuk *rating*. Selain itu, dengan mengetahui sebuah sentimen dari seseorang atau sekelompok orang, maka pihak yang berkepentingan dapat melakukan kajian terhadap produk, jasa, atau topik yang mereka tawarkan.

D. Pencocokan String

Sebuah *string* merupakan urutan dari simbol-simbol yang terdefinisi misalnya simbol karakter alfabet dan simbol biner. *String* dapat dibagi menjadi beberapa bagian yang disebut sebagai *substring*. *String* juga memiliki beberapa istilah penting seperti *prefix* dan *suffix*. Sebuah *prefix* merupakan *substring* dengan indeks dari 0 hingga k, sedangkan *suffix* merupakan *substring* dengan indeks k hingga m-1, dengan m adalah panjang *string* dan k adalah indeks antara 0 hingga m-1.



Gambar 3. Fitur *find and replace* pada Microsoft Word yang memanfaatkan *string matching* (Sumber: Mr. Excel)

Pencocokan *string* atau *string matching* merupakan sebuah proses untuk memanfaatkan suatu algoritma untuk melakukan pencarian kemunculan dari sebuah pola atau *pattern* di dalam sebuah teks. Panjang dari sebuah *pattern* misalnya m , harus lebih kecil atau sama dengan panjang dari teks atau *string* yaitu n , sehingga $m \leq n$, atau pada umumnya m sangat kecil dibanding n ($m \ll n$).

Contoh persoalan pencocokan *string* adalah sebagai berikut:

Carilah indeks pertama di dalam teks (T) "hari sudah pagi" yang bersesuaian dengan *pattern* (P) "sudah"!

Dalam hal ini:
T : hari **sudah** pagi
P : **sudah**

Diperoleh bahwa *pattern* P ditemukan pada indeks ke-5 dari *string* T.

Pencocokan *string* sudah banyak diaplikasikan untuk membantu manusia dalam menyelesaikan permasalahan di dunia nyata. Beberapa contoh aplikasi pencocokan *string* adalah seperti *search engine* pada Google, pencocokan rantai asam amino pada bidang ilmu bioinformatika, dan fitur pencarian teks pada aplikasi desktop Notepad dan Microsoft Word.

E. Algoritma Boyer-Moore

Algoritma Boyer-Moore ditemukan oleh Robert S. Boyer, dan J. Strother Moore pada tahun 1977. Algoritma Boyer-Moore merupakan salah satu algoritma pencocokan *string* terbaik saat ini. Algoritma ini merupakan sebuah algoritma pencocokan *string* yang didasari oleh dua teknik utama. Teknik-teknik tersebut adalah:

- Teknik *looking-glass*

Merupakan sebuah teknik dengan menjalankan pencarian *pattern* P pada teks T dengan berjalan mundur pada P mulai dari akhir P.

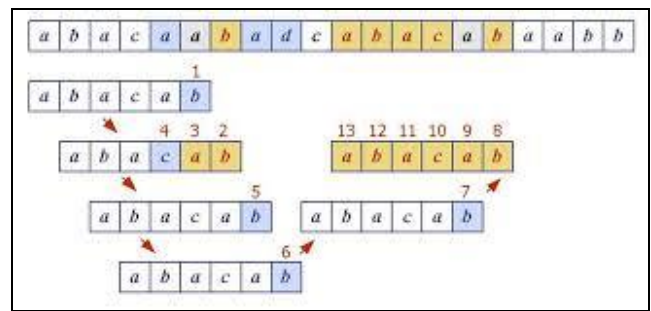
- Teknik *character jump*

Sesuai dengan namanya, teknik *character jump* akan mengatur lompatan yang akan terjadi ketika ditemukan sebuah kasus khusus dalam proses pencarian *string*. Teknik ini dilakukan ketika:

- Saat terjadi mismatch pada $T[i] == x$
- Saat $P[j]$ tidak sama dengan $T[i]$

Ada beberapa kemungkinan 'lompatan' yang dapat dilakukan:

- Jika P mengandung karakter x , maka geser P ke kanan agar sesuai dengan last occurrence dari x di P dengan $T[i]$
- Jika P mengandung karakter x , namun pergeseran P ke kanan sesuai dengan last occurrence tidak dimungkinkan, maka geser P ke kanan sebanyak 1 karakter dari $T[i+1]$
- Jika kasus (a) dan (b) tidak dihadapi maka geser P agar sesuai dengan $P[0]$ pada $T[i+1]$



Gambar 4. Ilustrasi kasus pencocokan *string* dengan algoritma Boyer-Moore (Sumber: StackOverflow)

Kompleksitas waktu algoritma Boyer-Moore memiliki *worst case* sebesar $O(nm + A)$ dimana n adalah panjang *string*, m adalah panjang *pattern*, dan A adalah jumlah dari alphabet yang ada. Kompleksitas ini tentunya jauh lebih cepat jika dibandingkan dengan algoritma *brute force* dalam pencocokan *string* dalam kasus jumlah alphabet yang besar seperti teks berbahasa Inggris pada umumnya. Namun, algoritma ini kurang bekerja dengan baik jika jumlah alphabet sedikit seperti pada teks binari.

Algoritma Boyer-Moore memiliki banyak aplikasi nyata di dunia modern. Contoh penerapan algoritma ini seperti pencarian nama berdasarkan kesamaan fonetik dan pencarian kata pada dokumen literatur.

F. Regular Expression

Eksresi reguler atau *regular expression* yang biasanya disingkat sebagai regex merupakan serangkaian karakter yang mendefinisikan sebuah pola pencarian dan pencocokan *string*. Regex sangat berkaitan dengan ilmu teori komputer dan bahasa formal. Ada banyak ketentuan umum pola dalam penulisan regex, beberapa contoh sederhana yang umumnya digunakan dapat dilihat pada gambar 5.

<code>\d</code>	Match a digital number or as [0-9]
<code>\w</code>	Match any letter of a-z or A-Z or 0-9 and _
<code>\s</code>	Match a white space
<code>[abc]</code>	Match any one of a, b, c
<code>(abc)</code>	Match abc exact once
<code>(abc)+</code>	Match abc one or more times
<code>(abc)?</code>	Match abc zero or one times
<code>(abc)*</code>	Match abc zero or more times
<code>[^abc]</code>	Match any character except of a, b, c
<code>\W\S\D</code>	Not \w not \s and not \d

Gambar 5. Contoh aturan sederhana pada *regular expression* (Sumber: SlidePlayer)

Regex dapat digunakan misalnya pada pencarian kata kunci dalam berita, membersihkan sebuah data teks, pengecekan kesalahan ketikan (*typo*) dan masih banyak lagi. Dalam implementasinya, banyak bahasa yang dapat digunakan untuk membuat ekspresi regex maupun menggunakannya secara langsung, misalnya pada Python, Java, JavaScript, dan PHP.

III. METODOLOGI

Metodologi dalam analisis sentimen menggunakan algoritma pencocokan *string* Boyer-Moore dan *regular expression* ini dibagi menjadi 3 tahapan besar yaitu *data scraping*, pembersihan data menggunakan *regular expression*, dan analisis sentimen menggunakan pencocokan *string*. Metode-metode tersebut akan dilakukan menggunakan bahasa pemrograman Python dan modul-modul yang ada di dalamnya.

A. Data scraping

Data *tweet* yang akan digunakan dalam analisis sentimen akan diperoleh menggunakan teknik *scraping* pada website Twitter. *Scraping* merupakan sebuah metode pengambilan data dari sumber digital, utamanya adalah sebuah situs web. *Scraping* dapat dilakukan secara manual dengan langsung melakukan *copy-paste* atau menggunakan otomasi tertentu. Di dalam pengambilan data *tweet* ini, akan digunakan sebuah kaskas untuk melakukan *scraping* secara otomatis yaitu Twint. Twint (*Twitter Intelligence Tool*) merupakan sebuah *scraper* yang tidak memerlukan akses pendaftaran *application programming interface* dari Twitter. Berikut ini merupakan parameter yang digunakan untuk melakukan *scraping* pada data *tweet*:

Tabel 1. Parameter *data scraping*

Parameter	Nilai
Search	Mudik
Start date	01/05/2021
End date	08/05/2021
Lang	id
Limit	1000
Store CSV	True

Parameter di atas memiliki arti “ambil seluruh *tweet* berbahasa Indonesia yang memiliki kata kunci ‘Mudik’ dari periode 01/05/2021 hingga 08/05/2021 sebanyak 1000 *tweet* dan simpan ke dalam sebuah file .csv”

B. Pembersihan data

Data yang telah diperoleh melalui proses *data scraping* tentunya harus dibersihkan sebelum dilakukan analisis sentimen. Pembersihan data akan dilakukan dengan memanfaatkan *regular expression* dengan membentuk ekspresi regex serta menjalankan substitusi pada kata-kata atau *pattern* yang ingin dihilangkan. Substitusi tersebut dilakukan menggunakan *library* regex pada Python, sedangkan untuk manipulasi data lainnya akan dilakukan dengan *library* Pandas.

```
import re
text = "Sample text"
text = re.sub(r'ekspresi_regex ', "", text)
```

Regular expression yang digunakan dalam pembersihan *tweet* ini berjumlah 4 ekspresi. Berikut merupakan ekspresi regex yang digunakan:

Tabel 2. Ekspresi regex dalam pembersihan data

Ekspresi regex	Keterangan
[#@][A-Za-z0-9./_]+	Menangani <i>mention</i> dan <i>hashtag</i> . Contoh: @pemerintah, #mudik
pic.twitter.com.[\w]+	Menangani pranala gambar. Contoh: pic.twitter.com.adzaoaqp
((www\.[^\s]+) (https?:\/\/[^\s]+) (http?:\/\/ [^\s]+))	Menangani pranala atau URL Contoh: www.google.com
[\^0-9a-zA-Z]+	Menangani karakter non- alfanumerik. Contoh: ¶, ?

Selain pembersihan kata menggunakan regex, data *tweet* juga akan dinormalisasi dengan melakukan *lowercase*, menangani *stopword* seperti ‘dan’, ‘yang’, ‘pada’, serta *slang* (kata gaul) seperti ‘gw’, ‘loe’, dan semacamnya.

C. Analisis sentimen

Analisis sentimen dilakukan dengan melakukan perhitungan skor frekuensi kemunculan kata yang bersifat positif dan negatif pada masing-masing *tweet*. Hal ini dilakukan dengan memanfaatkan algoritma pencocokan string Boyer-Moore dari setiap daftar kata positif dan negatif sebagai *pattern* dan data *tweet* sebagai *string* teks. Berikut merupakan aturan lebih detail dalam perhitungan nilai sentimen:

- Nilai sentimen awal dari setiap *tweet* adalah 0 yang menandakan *tweet* bersifat netral.
- Jika *tweet* mengandung sebuah kata yang berkonotasi positif seperti ‘bagus’, ‘baik’, dan semacamnya, maka nilai sentimen akan ditambah 1.
- Jika *tweet* mengandung sebuah kata yang berkonotasi negatif seperti ‘jelek’, ‘buruk’, dan semacamnya, maka nilai sentimen akan dikurangi 1.

Setelah melalui proses perhitungan tersebut, maka setiap *tweet* akan ditentukan sentimennya apakah bernilai positif (skor >0), negatif (skor <0), maupun netral (skor=0). Hasil distribusi label sentimen akan divisualisasikan menggunakan *library* Matplotlib. Untuk memperjelas langkah-langkah tersebut, berikut ini merupakan contoh dari desain proses analisis sentimen yang akan dilakukan:

Tweet: “larangan mudik bagus”

Kata positif : [“bagus”]

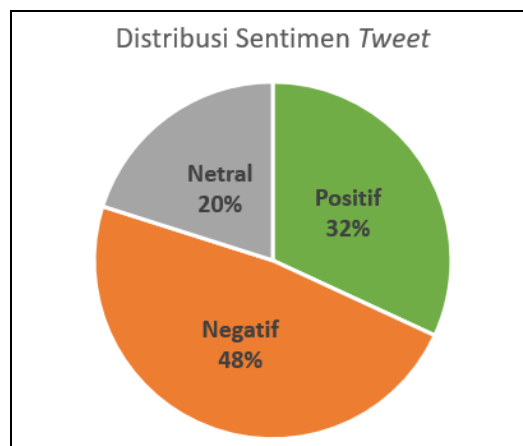
Kata negatif : [“jelek”]

Kata positif dan negatif yang diberikan hanya berupa contoh sederhana. Dalam praktiknya, kata-kata tersebut berbentuk senarai kata dengan jumlah yang tak sedikit.

Tabel 3. Sampel hasil pembersihan data

<i>Tweet awal</i>	<i>Tweet yang dibersihkan</i>
Berhenti mudik! Mari kita bantu pemerintah indonesia biar lebih amann dari covid	berhenti mudik mari bantu pemerintah indonesia amann covid
Itu bener WNA masuk pas mudik lagi dilarang? Pemudik suruh putar balik, eeh yg WNA dipersilahkan masuk.. pemerintah tutup kuping?	warga negara asing masuk pas mudik dilarang pemudik suruh putar eh warga negara asing dipersilakan masuk pemerintah tutup kuping
Giliran sibuk kerja tiket Pesawat banyak , Giliran dapat libur eh kagak dapat tiket mana ada larangan mudik lagi , nyusahin sih ☹️	giliran sibuk kerja tiket pesawat giliran libur eh tiket larangan mudik menyusahkan
#ayo0000 #nginap #mudik hotel di jl. Laksda adisucipto yogyakarta KLIK https://t.co/zrZq2CGei6	ayo0000 nginap mudik hotel jalan laksda adisucipto yogyakarta klik
@liputan6dotcom Bubarin Aja Indonesia...mudik lolak Aja dilarang...eh masih terima wna	liputan6dot bubarkan indonesia mudik lolak dilarang eh terima warga negara asing

Analisis sentimen yang dilakukan menggunakan perhitungan frekuensi konotasi kata dengan memanfaatkan algoritma Boyer-Moore menunjukkan hasil bahwa sebesar 319 *tweet* memiliki sentimen positif, 480 *tweet* memiliki sentimen negatif, dan 201 *tweet* memiliki sentimen netral. Hasil ini divisualisasikan dengan diagram lingkaran pada gambar 5. Algoritma Boyer-Moore dipilih karena memiliki performa paling baik jika digunakan pada data teks yang memiliki alfabet bervariasi seperti teks bahasa pada umumnya.



Gambar 6. Hasil distribusi sentimen *tweet*

Berdasarkan hasil tersebut, diperoleh bahwa sebagian besar *tweet* memiliki sentimen negatif, yaitu berjumlah hampir 50%

Inisiasi skor sentimen = 0

Asumsikan awalnya *tweet* bersifat netral. Jika tidak ada kata kunci (*pattern*) pada senarai kata positif dan negatif yang bersesuaian dengan teks, maka teks akan bersifat netral.

Ada 1 kata kunci positif dan 1 kata kunci negatif yang terdefinisi. Untuk setiap kata kunci (sebagai *pattern*), cek apakah *pattern* tersebut ada di dalam *string* yaitu *tweet* menggunakan algoritma Boyer-Moore.

Jalankan algoritma Boyer-Moore dengan:

Iterasi 1:

T = "larangan mudik bagus"

P = "bagus"

```

larangan mudik bagus
1 bagus
2      bagus
3      bagus
4      bagus
5      bagus [ Pattern ditemukan ]
    
```

Kata "bagus" terdapat di dalam *tweet*.

Skor sentimen ditambah 1.

Skor sentimen = 1

Iterasi 2:

T = "larangan mudik bagus"

P = "jelek"

```

larangan mudik bagus
1 jelek
2      jelek
3      jelek
4      jelek
    
```

Kata "jelek" tidak terdapat di dalam *tweet*.

Skor sentimen = 1

Akhir iterasi.

Skor sentimen akhir = 1

Jadi, *tweet* tersebut memiliki sentimen positif.

IV. HASIL DAN PEMBAHASAN

Data yang telah diperoleh menggunakan metode *scraping* dengan kaskas Twint memiliki format .csv dengan dimensi 1000 baris dan 36 atribut. Dari 36 atribut tersebut, hanya akan digunakan 1 atribut yaitu *tweet* yang berisi pesan utama dari pengirim. Dengan demikian, dimensi akhir dari data adalah 1000 baris dan 1 kolom.

Hasil pembersihan data menggunakan metode substitusi *regular expression*, normalisasi *stopword*, dan *slang* menunjukkan data *tweet* yang relatif lebih bersih dan teratur dibandingkan dengan data *tweet* pada awalnya. Berikut merupakan beberapa contoh sampel dari hasil pembersihan *tweet* yang dilakukan:

dari total *tweet* yang ada. *Tweet* yang memiliki sentimen positif memiliki proporsi dari 32% total data dan *tweet* netral 20% dari total data. Berikut ini merupakan contoh sampel *tweet* dari masing-masing label sentimen:

Tabel 4. Sampel *tweet* dari masing-masing label sentimen

Sentimen	Sampel <i>Tweet</i>
Positif	tunda mudik ikhlas hindari covid 19 patuhi protokol kesehatan mencuci tangan memakai masker menjaga jarak menghindari kerumunan mengurangi mobilitas tidakmudik
Negatif	diskriminasi warga negara indonesia mudik dilarang warga negara asing datang
Netral	atur jadwal meetup kakak satu mudik

Dari hasil analisis sentimen yang diperoleh, dapat ditarik kesimpulan bahwa berdasarkan sampel *tweet* yang diambil dari Twitter, sebagian besar sentimen masyarakat terhadap kebijakan larangan mudik pemerintah adalah negatif. Dengan demikian, pemerintah sebaiknya melakukan kajian terhadap kebijakan tersebut sesuai dengan respons masyarakat, atau juga dapat memperketat pengamanan saat mudik untuk mencegah sebagian besar masyarakat yang tidak setuju dengan larangan mudik agar tidak melanggar peraturan dari pemerintah tersebut.

Kompleksitas waktu dari algoritma Boyer-Moore pada setiap kata kunci (*keyword*) dan *tweet* yang didapatkan adalah $O(nm + A)$ dimana n merupakan panjang dari *tweet*, m merupakan panjang dari *keyword*, x merupakan jumlah *keyword* yang didefinisikan, serta A merupakan jumlah karakter alfabet unik yang terdefinisi.

V. KESIMPULAN DAN SARAN

Analisis sentimen masyarakat pada media sosial Twitter terhadap kebijakan larangan mudik dapat dilakukan dengan memanfaatkan algoritma Boyer-Moore dan *regular expression*. Hasil analisis sentimen menunjukkan bahwa mayoritas *tweet* memiliki sentimen negatif terhadap kebijakan tersebut, dimana, ditemukan fakta bahwa 32% *tweet* memiliki sentimen positif, 48% *tweet* memiliki sentimen negatif, dan 20% *tweet* memiliki sentimen netral. Temuan ini diharapkan dapat membantu pemerintah dalam mengkaji kebijakan larangan mudik yang telah diterapkan berdasarkan respons masyarakat di media sosial Twitter.

Dalam pengembangan selanjutnya, penulis menyarankan untuk melakukan analisis sentimen dengan jumlah data yang lebih besar serta data yang berasal dari berbagai sosial media lainnya seperti Instagram dan Facebook. Selain itu, penulis juga menyarankan untuk mengembangkan struktur

penyimpanan kata kunci agar dapat meningkatkan efektivitas dan efisiensi dari algoritma pencocokan *string* Boyer-Moore yang dijalankan pada tiap iterasi, seperti melakukan *indexing* atau mengimplementasikan sebuah *hash map*.

PRANALA VIDEO YOUTUBE

Berikut ini merupakan pranala video Youtube yang membahas topik makalah ini menggunakan media audio-visual:

https://youtu.be/njWfBWCY_8

UCAPAN TERIMA KASIH

Penulisan makalah ini tidak akan dapat diselesaikan tanpa berkat dari Tuhan Yang Maha Esa. Penulis ingin berterimakasih kepada Dr. Nur Ulfa Maulidevi, ST., M.Sc. sebagai dosen pengajar mata kuliah IF2211 Strategi Algoritma kelas 2, Dr. Ir. Rinaldi Munir, MT. yang menyediakan situs untuk mendapatkan inspirasi topik penulisan makalah, keluarga, dan teman-teman yang mendukung keberhasilan penulisan makalah ini.

DAFTAR PUSTAKA

- [1] A. Fathan Hidayatullah, dan Azhari, "Analisis Sentimen dan Klasifikasi Kategori Terhadap Tokoh Publik pada Twitter", Seminar Nasional Informatika (SEMNASIF). Vol. 1. No. 1, 2015.
- [2] A. Levitin, "Introduction to the Design & Analysis of Algorithms", Addison-Wesley, 2003.
- [3] B. Bemby Soebyakto, "Mudik Lebaran: Studi Kualitatif", Jurnal Ekonomi Pembangunan, Palembang: Universitas Sriwijaya, 2011.
- [4] E. Rahmanita, "Pencarian String Menggunakan Algoritma Boyer Moore pada Dokumen", Network Engineering Research Operation 1.1., 2016.
- [5] G. Asrofi Buntoro, "Analisis Sentimen Calon Gubernur DKI Jakarta 2017 di Twitter." INTEGER: Journal of Information Technology 2.1, 2017.
- [6] I. Diah. "Awesome Indonesia NLP", GitHub, 2019, diakses dari www.github.com pada 7 Mei 2021.
- [7] M. Leylia Khodra, "String Matching dengan Regular Expression", Bandung: Program Studi Teknik Informatika Institut Teknologi Bandung, 2019.
- [8] N.N., "Pencocokan String (*String/Pattern Matching*)", Bandung: Program Studi Teknik Informatika Institut Teknologi Bandung, 2020.
- [9] Y. Artha Uly, "Aturan Larangan Mudik Lebaran 2021", Kompas, 2021, diakses dari www.kompas.com pada 7 Mei 2021.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 8 Mei 2021



Dionisius Darryl Hermansyah 13519058