JURNAL FASILKOM ISSN: 2089-3353
Volume 11 No.2 | Agustus 2021: 97-103

Penerapan Lexicon Based Untuk Analisis Sentimen Pada Twiter Terhadap Isu Covid-19

Yanda Nooryuda Prasetya[1], Doni Winarso[2], Syahril[3]
[123]Sistem Informasi, Fakultas Ilmu Komputer, Universitas Muhammadiyah Riau
yandanooryudaprasetya@student.umri.ac.id, doniwinarso@umri.ac.id *, syahril@umri.ac.id

Abstract

Currently, the use of social media such as Twitter has become a necessity in this modern era. Twitter is enabling users to provide feedback on various hot topics and issues happening. The current trending issue is Covid-19. It is spread by some people that Covid-19 is a conspiracy and it is not uncommon to believe that Covid-19 is something real. Therefore, an analysis is needed to get the truth about whether Covid-19 is a conspiracy or real from the point of view and public opinion contained in Twitter, Lexicon based is a method used to classify public opinion into 3 classes, namely, positive, negative or negative sentiments. neutral. Research conducted shows that public opinion on Twitter who believes that Covid-19 is something real is higher than that of community groups who believe the issue of Covid-19 is a conspiracy. The percentage of grouping can be seen from the positive sentiment category at 58.08%, the negative category sentiment opinion at 37.61%, and the neutral sentiment opinion category at 4.31%.
Keywords: Covid-19, Twitter, Sentiment Analysis, Lexicon Based

Abstrak

Saat ini penggunaan sosial media seperti twitter telah menjadi kebutuhan di era modern ini. Twitter merupakan memungkinkan pengguna untuk memberikan tanggapan terhadap berbagai topik dan isu hangat terjadi. Saat ini isu yang menjadi trending adalah Covid-19. Tersebar disebagian masyarakat bahwa Covid-19 adalah suatu konspirasi dan tidak jarang yang mempercayai bahwa Covid-19 adalah sesuatu yang nyata. Karena itu diperlukan sebuah analisis untuk mendapatkan kebenaran apakah Covid-19 sebuah konspirasi atau nyata dilihat dari sudut pandang dan opini masyarakat yang tertuang di dalam Twitter, Lexicon based merupakan metode yang digunakan untuk mengelompokkan opini masyarakat ke dalam 3 kelas yaitu, sentimen positif, negatif atau netral. Penelitian yang dilakukan menunjukkan bahwa opini masyarakat pada twitter yang percaya bahwa Covid-19 adalah sesuatu yang nyata lebih tinggi dibandingkan dengan kelompok masyarakat yang mempercayai isu Covid-19 adalah konspirasi. Persentase pengelompokan dapat dilihat dari sentimen kategori positif sebesar 58.08%, opini sentimen kategori negatif sebesar 37.61%, dan opini sentimen kategori netral sebesar 4.31%.

Kata kunci : Covid-19, Twitter, Analisis Sentimen, *Lexicon Based*

1. Pendahuluan

Twitter merupakan salah satu media komunikasi yang banyak diminati oleh masyarakat dunia yang memungkinkan penggunanya untuk menulis tentang berbagai topik dan membahas isu-isu yang sedang terjadi [1]. Pengguna twitter dapat mengemukakan opininya melalui tweet. Setiap tweet yang di posting pengguna beraneka ragam sesuai dengan keinginan pengguna. Tweet bisa berupa pendapat, saran, ataupun kritikan tentang topik-topik tertentu. Salah satu isu yang sedang menjadi trending topic saat ini adalah Covid-19.

Covid-19 atau disebut juga sebagai 2019 Novel Coronavirus (2019-nCoV) merupakan virus yang telah menyerang masyarakat dunia saat ini. Virus ini diketahui pertama kali muncul di kota Wuhan, China pada akhir Desember 2019. Covid-19 mulai masuk Indonesia awal Maret 2020 [2]. Informasi tentang virus ini terus menyebar secara masif di kalangan masyarakat.

Banyak kalangan di masyarakat tersebar isu bahwa Covid-19 ini merupakan konspirasi dan tidak sedikit juga yang mempercayai bahwa Covid-19 ini nyata. Hal ini dapat diamati dari *tweet* pengguna Twitter ada beberapa *tweet* yang seolah menggambarkan bahwa covid ini adalah konspirasi dan ada beberapa tweet yang mengarah kepada bahwa covid ini adalah benar benar nyata.

Selain itu keraguan ini juga dapat terlihat pada pemberitaan dibeberapa media digital nasional yang berisikan keraguan terkait isu Covid-19 ini. Diantaranya adalah menurut [3] menuliskan bahwa munculnya teori konspirasi yang menyesatkan tentang virus ini telah mengancam langkah dunia secara serius untuk menanggapi pandemi Covid-19. Ini karena hanya sedikit orang yang percaya teori konspirasi yang tersebar luas ini.

Berangkat dari itu untuk mengetahui apakah Covid-19 ini nyata atau konspirasi berdasarkan opini masyarakat di Twitter, maka penelitian ini akan melakukan analisis sentimen terhadap opini tersebut.

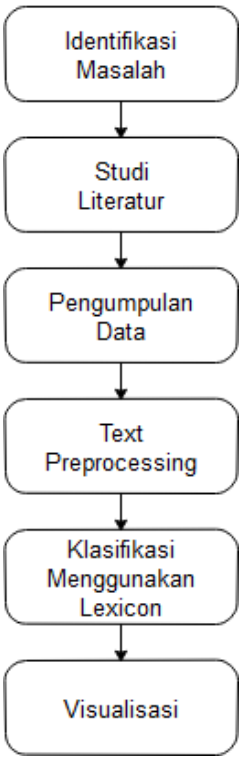
Analisis sentimen atau disebut juga *opinion mining* yang bertujuan untuk menganalisis, memahami, mengolah, dan mengestrak data tekstual yang berupa opini terhadap entitas seperti organisasi dan topik tertentu agar mendapatkan suatu informasi [4]. Analisis sentimen dilakukan untuk menentukan apakah opini atau komentar terhadap suatu permasalahan atau isu terkait topik tertentu memiliki kecenderungan positif, negatif, atau netral dan dapat dijadikan sebagai acuan dalam meningkatkan suatu pelayanan, ataupun meningkatkan kualitas produk.

Berbagai penelitian terkait analisis sentimen telah banyak dilakukan. Terdapat dua pendekatan untuk melakukan analisis sentimen, pendekatan yang pertama adalah berbasis *machine learning* yaitu dengan melatih data latih pada dataset yang telah dilabelkan secara manual. Pendekatan yang kedua adalah berbasis leksikal (*Lexicon Based*) yang tidak memerlukan pelatihan dataset untuk menemukan polaritas sentimen [5]. Penelitian yang dilakukan oleh [6] membandingkan beberapa metode analisis sentimen dalam topik pariwisata, kesimpulannya adalah bahwa menggunakan metode *lexicon* memiliki peningkatan hasil sentimen paling tinggi dibandingkan dengan metode lainnya. Pada penelitian ini penulis akan menggunakan sumber daya kamus leksikon yaitu InSet (*Indonesian Sentiment Lexicon*) yang dikembangkan oleh [7].

Berdasarkan latar belakang tersebut, maka akan dilakukan penelitian terkait analisis sentimen pada media sosial Twitter terhadap isu Covid-19 di Indonesia dengan metode *Lexicon Based*. Dengan tujuan penelitian ini adalah untuk mengelompokkan opini masyarakat ke dalam kategori sentimen positif, negatif atau netral menggunakan lexicon.

2. Metode Penelitian

Metodologi penelitian adalah langkah-langkah yang akan dilakukan dalam membuat dan menyelesaikan penelitian. Berikut adalah diagram metodologi pada penelitian ini.



Gambar 1. Tahapan Penelitian

2.1 Identifikasi Masalah

Langkah awal yang dilakukan pada penelitian ini adalah identifikasi masalah bagaimana melakukan analisis sentimen pada media sosial Twitter dengan menggunakan metode Lexicon Based. Pengguna Twitter dapat menuliskan pesan, komentar atau opini apapun pada Twitter. Namun dengan terbatasnya cangkupan penulisan yang hanya 280 karakter, membuat pengguna Twitter menuliskan pesan berupa singkatan. Hal ini menjadi permasalahan sendiri dalam menemukan sentimen pada data Twitter. Untuk itu diperlukan sebuah Algoritma yang dapat menyeleksi kata slang dan kemudian merubahnya menjadi kata yang baku sehingga dapat digunakan untuk analisis sentimen.

2.2 Studi Literatur

Pada tahap ini merupakan proses mencari, mempelajari, dan menggunakan berbagai macam literatur berupa buku, jurnal, paper, e-book, atau literatur lain yang berkaitan dengan metode Lexicon Based khususnya yang digunakan untuk analisis sentimen.

2.3 Pengumpulan Data

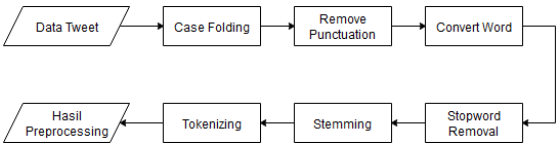
Pada tahapan ini dilakukan pengumpulan dan pencarian dataset twitter Bahasa Indonesia. Data yang dikumpulkan adalah data tweet masyarakat dari Twitter yang merupakan pesan, komentar atau opini dengan topik Covid-19 Indonesia. Data yang

diambil adalah data tweet dari bulan Maret sampai Agustus 2020.

Data diambil menggunakan proses crawling melalui Twitter API dengan memanfaatkan library GetOldTweet menggunakan bahasa pemograman Python.

2.4 Text Preprocessing

Data tweet yang telah dikumpulkan dari twitter, banyak memiliki noise sehingga perlu adanya tahapan proses menghilangkan noise agar proses analisis sentimen menjadi lebih akurat dan dapat digunakan secara general. Alur text preprocessing pada penelitian ini dapat dilihat pada gambar berikut:



Gambar 2. Text Preprocessing

2.5 Klasifikasi Menggunakan Metode Lexicon

Setelah membersihkan data pada proses preprocessing, tahap selanjutnya yaitu klasifikasi menggunakan metode lexicon. Proses klasisifikasi ini dilakukan dengan cara pengecekan kata yang ada pada dataset kemudian di cocokan dengan kata pada kamus lexicon yang sudah disiapkan sebelumnya.

2.6 Visualisasi

Setelah tahapan klasifikasi data menggunakan metode Lexicon selesai dilakukan maka tahap akhir adalah memvisualisasikan hasil dari klasifikasi analisis sentimen dalam bentuk diagram batang maupun wordcloud.

3. Hasil dan Pembahasan

Bab ini membahas hasil dari penelitian analisis sentimen masyarakat pada Twitter terhadap isu covid-19 menggunakan metode *Lexicon Based* sesuai dengan tahapan yang sudah dijelaskan pada bab sebelumnya.

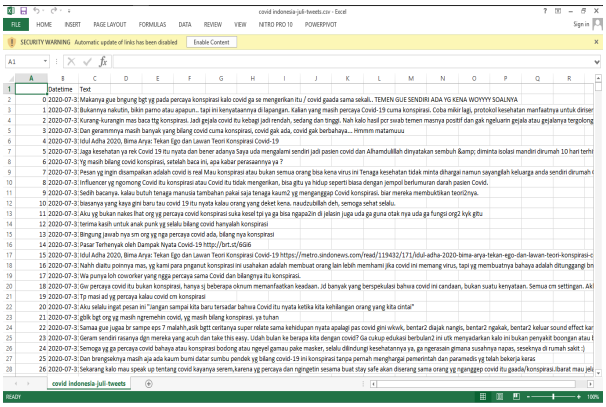
3.1 Pengumpulan Data

Pengumumpulan data yang digunakan pada penelitian ini adalah data Twitter dari bulan Maret 2020 sampai dengan Agustus 2020. Setelah dilakukan proses *crawling* maka didapatkan data sebanyak 151.618 data *tweet*. Kode untuk melakukan *crawling* data dapat dilihat pada gambar berikut.

```
In [1]: import GetOldTweets3 as got
        import pandas as pd
        # Function that pulls tweets based on a general search query and turns to csv file
        # Parameters: (text query you want to search), (max number of most recent tweets to pull from)
        def text_query_to_csv(text_query):
            # Creation of query object
            tweetCriteria = got.manager.TweetCriteria().setQuerySearch(text_query)\
                                                       .setSince("2020-03-01")\
                                                       .setUntil("2020-04-01")\
                                                       .setLang("id")
            # Creation of list that contains all tweets
            tweets = got.manager.TweetManager.getTweets(tweetCriteria)

            # Creating list of chosen tweet data
            text_tweets = [[tweet.date, tweet.text] for tweet in tweets]

            # Creation of dataframe from tweets
            tweets_df = pd.DataFrame(text_tweets, columns = ['Datetime', 'Text'])

            # Converting tweets dataframe to csv file
            tweets_df.to_csv('{}-tweets.csv'.format(text_query), sep=',')

        text_query = 'covid indonesia'
        # Calling function to query X amount of relevant tweets and create a CSV file
        text_query_to_csv(text_query)
```

Gambar 3. Kode Crawling Data Twitter

Hasil dari pengumpulan data yang merupakan sebagian dari data Twitter yang diambil menggunakan proses crawling. Data diambil dari Twitter dengan memanfaatkan library GetOldTweet menggunakan bahasa pemograman Python. Data *field* yang diambil hanya data *datetime,* dan *tweet*. Data tersebut kemudian disimpan kedalam file csv seperti pada gambar di bawah ini untuk memudahkan tahapan atau proses selanjutnya.



Gambar 4. Pengumpulan Data Kedalam File CSV

3.2 Tahapan *Text Preprocessing*

Tahapan *text preprocessing* dilakukan karena data hasil *crawling* dari twitter mengandung banyak *noise,* seperti duplikasi data karena pesan di *retweet* (RT) atau di *share* kembali oleh pengguna twitter, mengandung informasi yang tidak berguna untuk proses klasisfikasi sentimen seperti *hastag* (#), *mention* (@), link (http://www), tanda baca, angka, penggunaan bahasa slang atau singkatan.

3.2.1 Tahapan *Case Folding*

Tahapan *case folding* bertujuan untuk mengubah semua huruf yang ada pada dokumen kedalam bentuk yang sama menjadi *lower case* atau huruf kecil. Berikut adalah kode untuk melakukan tahapan *case folding*:

```
In [4]: import pandas as pd
        import string

        df = pd.read_csv('dataset/tesdata.csv')
        text = df['Text']
        case_folding = []
        for w in text:
            tweet = w.lower()
            case_folding.append(tweet)

        df['case_folding']  = case_folding
        df.head()
```

Gambar 5. Tahapan *Case Folding*

Gambar 5 merupakan kode untuk melakukan tahap *case folding.* Data yang sudah disimpan dalam sebuah file csv akan di *load* kemudian ditampung dalam sebuah variabel. Kemudian setiap *tweet* pada data akan dilakukan proses *case folding*. Berikut contoh data yang sudah melewati tahapan *case folding*:

3.2.2 Tahapan *Remove Punctuation*

Tahapan *remove punctuation* bertujuan untuk menghapus karakter yang tidak memberikan pengaruh terhadap proses klasifikasi sentimen setelah dilakukan proses *case folding,* seperti menghapus tanda baca koma (,), titik (.), hastag (#), mention (@), link, karakter selain huruf dan angka. Disamping itu, tahapan ini juga membersihkan duplikasi data yang berasal dari tindakan *retweet* (RT) pengguna twitter serta membuang kata yang dianggap tidak perlu terhadap penelitian ini seperti kata "covid" dan "indonesia" karena kata tersebut merupakan keyword dalam pencarian data. Berikut adalah kode untuk melakukan tahapan *remove punctuation*:

```
In [14]: import re
         from nltk.tokenize import word_tokenize

         def del_word(x,key_list):
             n = len(key_list)
             word_tokens = word_tokenize(x)
             new_x =''
             for word in word_tokens:
                 if word not in key_list:
                     new_x = new_x+word+' '
             return new_x

         def clean_tweets(tweet):
             tweet = re.sub("(&amp)"," ", tweet)
             #menghapus @
             tweet = re.sub("([@#][A-Za-z0-9]+)|(\w+:\/\/\S+)"," ", tweet)
             #or RT sign in the beginning of the tweet
             tweet = re.sub(r':', '', tweet)
             tweet = re.sub(r',Å¶', '', tweet)
             #replace consecutive non-ASCII characters with a space
             tweet = re.sub(r'[^\x00-\x7F]+',' ', tweet)
             #remove punctuation manually
             tweet = re.sub('[^a-zA-Z]', ' ', tweet)
             #remove tags
             tweet=re.sub("&lt;/?.*?&gt;","&lt;&gt;",tweet)
             #remove digits and special chars
             tweet=re.sub("(\\d|\\W)+"," ",tweet)
             #remove other symbol from tweet
             tweet = re.sub(r'â', '', tweet)
             tweet = re.sub(r'€', '', tweet)
             tweet = re.sub(r'¦', '', tweet)
             return tweet

         re_clean = df['case_folding'].apply(lambda x: clean_tweets(x))
         keyword = ['covid','indonesia','nya']
         clean_text_extra = re_clean.apply(lambda x: del_word(x,keyword))

         df['clean_text'] = clean_text_extra
         df.drop_duplicates(subset ="clean_text", keep = False, inplace = True)
         df = df.reset_index(drop=True)
```

Gambar 6. Tahapan *Remove Punctuation*

Gambar 6 merupakan kode untuk melakukan tahap *remove punctuation.* Data yang sudah melewati tahap *case folding* kemudian dilakukan tahap penghapusan karakter yang dianggap tidak berpengaruh terhadap analisis sentimen. Berikut contoh data yang sudah melewati tahapan *remove punctuation*:

3.2.3 Tahapan *Convert Word*

Pada tahapan ini dilakukan proses normalisasi kata yang tidak sesuai ke bentuk baku sesuai kamus besar bahasa Indonesia (KBBI). Tahapan ini dilakukan karena pada twitter pengguna dibatasi hanya dapat menggunakan 280 karakter. Hal tersebut membuat penggunanya menuliskan *tweet* dengan kata yang berupa singkatan ataupun kata yang tidak baku sehingga kata tersebut perlu di ubah kedalam bentuk baku nya. Kata-kata singkatan seperti "bgt", "yg", "ttg", "sdh" dan lain sebagainya akan di ubah kedalam bentuk normal nya. Sumber kamus kata slang tersebut didapat dari *repository* GitHub (GitHub - ramaprakoso) dengan penambahan beberapa kata yang dianggap belum ada pada *library* tersebut. Namun pada penerapannya, masih terdapat kata yang tidak bisa di *convert* karena pada beberapa data *tweet* terdapat kalimat *typo* atau salah pengetikan yang membuat kata tersebut sulit di normalkan.

```
In [62]: import ast

         file_2  = open("cleaning_source/update_combined_slang_words.txt", "r")
         content2 = file_2.read()
         slang_words = ast.literal_eval(content2)
         file_2.close()

         def convert_word(tweet):
             word_tokens = word_tokenize(tweet)
             for w in word_tokens:
                 if w in slang_words.keys():
                     word_tokens[word_tokens.index(w)] = slang_words[w]

             #filter using NLTK library append it to a string
             filtered_tweet = [w for w in word_tokens]
             filtered_tweet = []

             #looping through conditions
             for w in word_tokens:
                 filtered_tweet.append(w.lower())
             return ' '.join(filtered_tweet)

         convert_word = df['clean_text'].apply(lambda x: convert_word(x))
         df['convert_word'] = convert_word
```

Gambar 7. Tahap *Convert Word*

Gambar 7 merupakan kode untuk melakukan tahap *convert word*. Data yang telah dibersihkan pada proses *remove punctuation* kemudian akan dilakukan *convert word* agar kata dalam kalimat menjadi kata yang baku. Berikut contoh data yang telah dilakukan tahapan *convert word*.

3.2.4 Tahapan *Stopword Removal*

Tahapan *stopword removal* dilakukan untuk membuang kata-kata yang tidak berpengaruh dalam proses klasifikasi sentimen seperti waktu, penghubung, dan lainnya [8]. Kata-kata tersebut dihapus berdasarkan kata yang telah disiapkan pada kamus *stopword*. Sumber kamus *stopword* yang digunakan diambil dari *repository* GitHub [10]. Berikut merupakan beberapa contoh kata yang ada pada kamus *stopword*.

Tabel 1. Contoh Kata Pada *Stopword*

adalah	adapun	agaknya	akhirnya
aku	dia	saya	adalah

begitu	dan	dari	kapan
kenapa	siapa	atau	kok
makanya	mengapa	jadi	sudah

```
In [66]: my_file = open("cleaning_source/combined_stop_words.txt", "r")
content = my_file.read()
stop_words = content.split("\n")
my_file.close()

def stopword_remove(tweet):
    word_tokens = word_tokenize(tweet)

    #filter using NLTK library append it to a string
    filtered_tweet = [w for w in word_tokens if not w in stop_words]
    filtered_tweet = []

    #looping through conditions
    for w in word_tokens:
        #check tokens against stop words , emoticons and punctuations
        if w not in stop_words and w not in string.punctuation:
            filtered_tweet.append(w.lower())
    return ' '.join(filtered_tweet)
stopword_remove = df['convert_word'].apply(lambda x: stopword_remove(x))
df['stopword_remove'] = stopword_remove
```

Gambar 8. Tahapan *Stopword Removal*

Gambar 8 merupakan kode untuk melakukan tahap *stopword removal*. Data yang sudah di normalkan kemudian dilakukan penghapusan kata berdasarkan daftar kamus *Stopword* yang sudah disiapkan. Berikut contoh data yang telah dilakukan tahapan *stopword removal.*

3.2.5 Tahapan Stemming

Tahapan *stemming* bertujuan untuk mengembalikan kata ke bentuk dasarnya. Dalam penelitian ini akan digunakan *library* Sastrawi untuk melakukan proses *stemming*. Sastrawi merupakan *library* pada bahasa pemograman python yang dibangun dengan algoritma NA [9]

Pada penelitian ini tahapan *stemming* dilakukan bersamaan dengan proses klasifikasi *lexicon.* Berikut contoh kode untuk melakukan tahapan *stemming.*

```
In [5]: from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

factory = StemmerFactory()
stemmer = factory.create_stemmer()

def stem(tweet):
    word_tokens = word_tokenize(tweet)

    #filter using NLTK library append it to a string
#     filtered_tweet = [w for w in word_tokens]
    filtered_tweet = []

    #looping through conditions
    for w in word_tokens:
        kata_dasar = stemmer.stem(w)
        filtered_tweet.append(kata_dasar)
    return ' '.join(filtered_tweet)

stemming = df['stopword_remove'].apply(lambda x: stem(x))
```

Gambar 9. Tahap Stemming

3.2.6 Tahapan Tokenizing

Tahapan terakhir dari proses *preprocessing* adalah *tokenizing*, dimana data teks setelah dilakukan pembersihan kemudian dipecah ke dalam token-token berdasarkan delimiternya yaitu spasi (*space)*. Hasil proses *tokenizing* ini selanjutnya akan dilakukan klasifikasi dengan kamus *Lexicon*.

```
In [70]: token = []

for w in df.stopword_remove:
    word_tokens = word_tokenize(w)
    token.append(word_tokens)

df['tokenizing'] = token

In [71]: df['tokenizing']

Out[71]: 0    [bingung, banget, percaya, konspirasi, tidak, ...
1    [kurangin, mas, baca, konspirasi, gejala, keba...
2    [jaga, kesehatan, rek, nyata, benar, mengalami...
3       [bilang, konspirasi, baca, kabar, perasaannya]
4    [pesan, disampaikan, is, real, konspirasi, buk...
Name: tokenizing, dtype: object
```

Gambar 10. Tahapan Tokenizing

Gambar 10 merupakan kode untuk melakukan tahap *tokenizing.*

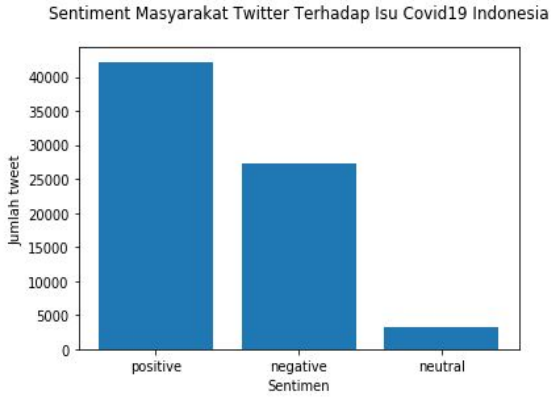
3.3 Proses Klasisfikasi Menggunakan Lexicon

Setelah semua tahapan *preprocessing* selesai dilakukan, didapat total jumlah data bersih sebanyak 72.686 data yang siap untuk dilakukan tahap klasifikasi sentimen menggunakan kamus *lexicon*. Tahapan ini memegang peranan penting dalam klasifikasi. Karena penelitian ini menggunakan pendekatan pada *word level*, dimana data yang diproses adalah kata untuk memperoleh skor sentimen. Kamus *Lexicon* yang digunakan pada penelitian ini adalah kamus InSet *Lexicon* dari penelitian sebelumnya yang dilakukan oleh Koto dan Rahmaningtyas [7] untuk mengklasifikasi sentimen terhadap data twitter.

Kamus InSet *Lexicon* ini berisi daftar kata yang mengandung sentimen positif maupun negatif serta sudah memiliki bobot nilai untuk kata nya. Kamus *lexicon* ini terdiri dari 3609 kata positif dan 6609 kata negatif. Bobot pada kamus *lexicon* ini memiliki nilai dengan rentang skor dari -5 sampai +5. Pada penelitian ini dilakukan beberapa penambahan kata yang berkaitan dengan topik Covid-19.

3.4 Visualisasi Data

Tahap terakhir dalam penelitian ini adalah memvisualisasikan hasil klasisfikasi metode *Lexicon Based* untuk analisis sentimen pada data twitter untuk kelas positif, negatif, dan netral. Hasil visualisasi dalam diagram batang disajikan pada gambar berikut.

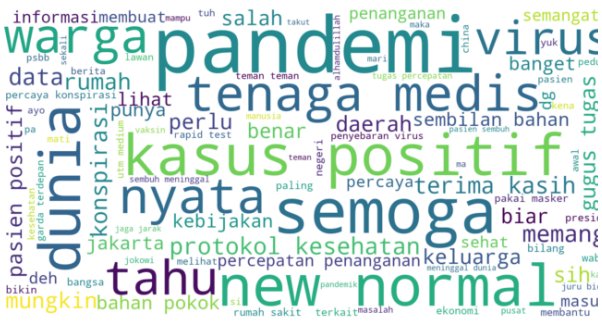


Gambar 11. Visualisasi sentimen masyarakat terhadap isu covid-19

Dari gambar di atas dapat dilihat bahwa opini masyarakat pada Twitter diperoleh frekuensi opini dengan sentimen positif lebih tinggi untuk pengguna yang mempercayai bahwa covid adalah nyata dibandingkan dengan pengguna yang mengatakan covid adalah konspirasi.

Hasil klasifikasi sentimen menggunakan metode *Lexicon Based* dengan melakukan perhitungan bobot setiap kata pada opini masyarakat Twitter terhadap isu Covid-19 di Indonesia dari bulan Maret 2020 sampai dengan Agustus 2020. Dari total 72.686 data twitter menunjukkan bahwa persentase opini masyarakat untuk kelas sentimen positif sebesar 58.08% yang mempercayai covid adalah nyata dan opini masyarakat untuk kelas sentimen negatif sebesar 37.61% yang mengatakan covid adalah konspirasi. Sisanya 4.31% opini masyarakat berada pada sentimen netral

Kemudian visualisasi dalam bentuk *wordcloud* yang menunjukkan kata-kata yang sering muncul dalam topik Covid-19 Indonesia pada kelas sentimen positif ditunjukkan pada gambar berikut ini.



Gambar 12. WordCloud Pada Sentimen Positif

Gambar 12 manampilkan visualisasi output kata yang paling banyak muncul pada sentimen positif terhadap isu covid-19. Pada gambar tersebut dapat terlihat kata yang paling sering muncul adalah kata "pandemi", "kasus positif", dan "nyata". Hal ini menunjukan opini masyarakat pada sentimen positif mengarah ke mempercayai covid adalah nyata. Namun dalam *wordcloud* bersentimen positif terlihat adanya kata "konspirasi" yang sebenarnya memiliki makna negatif. Hal tersebut dapat disebabkan kata "konspirasi" muncul beriringan dalam ulasan yang bersentimen positif.

Gambar 13. WordCloud Pada Sentimen Negatif

Gambar 13 menampilkan visualisasi output kata yang paling banyak muncul pada sentimen negatif terhadap isu Covid-19. Dapat dilihat pada gambar tersebut kata yang paling sering muncul adalah kata "konspirasi". Hal ini menunjukkan opini masyarakat pada sentimen negatif masih ada yang cenderung mengatakan bahwa covid adalah konspirasi

Dari proses analisis sentimen yang telah dilakukan, hasil yang didapati adalah sentimen positif untuk masyarakat pengguna Twitter yang percaya covid-19 adalah nyata cukup dominan dengan jumlah 42219 *tweets* atau 58.08%, kemudian disusul oleh sentimen negatif yang mengatakan bahwa covid adalah konspirasi berjumlah 27337 *tweets* atau 37.61%, dan sentimen netral dengan jumlah 3130 *tweets* atau 4.31%

4. Kesimpulan

Berdasarkan hasil yang telah dilakukan pada penelitian analisis sentimen masyarakat pada Twitter terhadap isu Covid-19 di Indonesia menggunakan metode *Lexicon Based* dengan melakukan perhitungan bobot setiap kata, dari 72.686 data twitter menunjukkan hasil opini masyarakat dengan sentimen kategori positif sebesar 58.08%, opini sentimen kategori negatif sebesar 37.61%, dan opini sentimen kategori netral sebesar 4.31%. Sehingga dapat disimpulkan dari hasil analisis sentimen menunjukkan bahwa opini masyarakat pada twitter yang percaya bahwa Covid-19 adalah nyata masih cukup tinggi dibandingkan dengan masyarakat yang mempercayai isu konspirasi Covid-19.

Daftar Rujukan

[1] Hamdan, H., Bellot, P., & Bechet, F. (2015). Lsislif: Feature Extraction and Label Weighting for Sentiment Analysis in Twitter. Proceedings ofthe 9th International Workshop on Semantic Evaluation, SemEval, 568–573. https://doi.org/10.18653/v1/s15-2095

[2] Febrian, D. A. (2020). Asal Mula dan Penyebaran Virus Corona dari Wuhan ke Seluruh Dunia. idntimes.com. https://bali.idntimes.com/health/medical/denny-adhietya/asal-muasal-dan-perjalanan-virus-corona-dari-wuhan-ke-seluruh-dunia-regional-bali/full

[3] Kompas.com. (2020). Ini Alasan Mengapa Orang Percaya pada Teori Konspirasi Virus Corona Halaman all - Kompas.com.https://www.kompas.com/sains/read/2020/04/20/180200923/ini-alasan-mengapa-orang-percaya-pada-teori-konspirasi-virus-corona?page=all

[4] Liu, B. (2010). Sentiment analysis and subjectivity. *Handbook of natural language processing*, *2*(2010), 627-666.

[5] Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams engineering journal*, *5*(4), 1093-1113.

[6] Alaei, A. R., Becken, S., & Stantic, B. (2019). Sentiment Analysis in Tourism: Capitalizing on Big Data. Journal of Travel Research, 58(2), 175–191. https://doi.org/10.1177/0047287517747753

[7] Koto, F., & Rahmaningtyas, G. Y. (2018). Inset lexicon: Evaluation of a word list for Indonesian sentiment analysis in microblogs. Proceedings of the 2017 International Conference on Asian Language Processing, IALP 2017, 2018

[8] Vijayarani, S., Ilamathi, M. J., & Nithya, M. (2015). Preprocessing techniques for text mining-an overview. *International Journal of Computer Science & Communication Networks*, *5*(1), 7-16.

[9] Adriani, M., Asian, J., Nazief, B., Tahaghoghi, S. M. M., & Williams, H. E. (2007). Stemming Indonesian. ACM Transactions on Asian Language Information Processing, 6(4), 1–33. https://doi.org/10.1145/1316457.1316459

[10] Masdevid, https://github.com/masdevid/ID-OpinionWords, diakses tgl. 27-7-2021.