

# Analisis Sentimen Terhadap *Cryptocurrency* Berbasis *Python TextBlob* Menggunakan Algoritma Naïve Bayes

Rizaldi Azhar<sup>1</sup>, Adi Surahman<sup>2</sup>, Christina Juliane<sup>3</sup>

<sup>1,2,3</sup>Program Studi Magister Sistem Informasi, STMIK LIKMI Bandung

Jl. Ir.H.Juanda N0.96 Bandung 40132, Jawa Barat-Indonesia

[rizaldi.azhar@bni.co.id](mailto:rizaldi.azhar@bni.co.id), [surahman325@asqi.co.id](mailto:surahman325@asqi.co.id), [christina.juliane@likmi.ac.id](mailto:christina.juliane@likmi.ac.id)

## Abstract

*Cryptocurrency users are now increasing as the market becomes more and more attractive. In 2019 recorded around 139 million account users verified id cryptocurrency. Recently, it was enlivened by the emergence of #crypto on Twitter and had become a world trending topic. This gives rise to many opinions and opinions from twitter users. With so many twitter users' opinions on the hashtag, it is very difficult to know whether positive, negative or neutral sentiments are manual. This requires machine learning to be able to automate labeling, be it positive, neutral or negative sentiments. Machine learning used is by utilizing Python TextBlob. The results of automatic labeling using Python TextBlob from a total of 1032 tweets obtained 632 tweets or 61.24% containing positive sentiments, 296 neutral sentiments or 28.68% tweets and 104 negative sentiments or 10.07%. The test results using the Naïve Bayes algorithm with each testing data and training data are 0.2 and 0.8. From this test, the accuracy value is 71.98%, precision is 83.04%, recall is 60.88% and f1\_score is 65.07%.*

**Keywords:** #crypto, twitter, Python TextBlob, Naïve Bayes

## Abstrak

*Pengguna cryptocurrency kini kian meningkat seiring pasar yang semakin menarik. Pada tahun 2019 mencatat sekitar 139 juta pengguna akun yang diverifikasi id cryptocurrency. Baru baru ini diramaikan dengan munculnya #crypto di twitter dan sempat menjadi trending topik dunia. Hal ini menimbulkan banyaknya opini maupun pendapat dari pengguna twitter. Dengan banyak nya opini pengguna twitter terhadap hashtag tersebut sangatlah sulit untuk diketahui sentimen positif, negatif ataupun netral secara manual. Hal ini dibutuhkan machine learning untuk bisa mengotomatisasi pelabelan, baik itu sentimen positif, netral ataupun negatif. Machine learning yang digunakan adalah dengan memanfaatkan Python TextBlob. Hasil pelabelan otomatis menggunakan Python TextBlob dari total 1032 tweet didapatkan 632 tweet atau 61.24% yang mengandung sentimen positif, sentimen netral sebanyak 296 atau tweet 28.68 % dan sentimen negatif sebanyak 104 tweet atau 10.07%. Hasil pengujian menggunakan algoritma Naïve Bayes dengan masing masing data testing dan data training adalah 0.2 dan 0.8. Dari pengujian ini menghasilkan nilai accuracy sebesar 71.98%, precision 83.04%, recall 60.88% dan f1\_score 65.07%.*

**Kata kunci:** #crypto, twitter, Python TextBlob, Naïve Bayes

## 1. PENDAHULUAN

Sosial media merupakan suatu media yang dapat digunakan untuk berekspresi oleh penggunanya [1]. Salah satu media sosial yang banyak dimanfaatkan masyarakat untuk beropini adalah *twitter* [2]. *Twitter* sebagai salah satu situs *microblogging* mengalami banyak perkembangan, terhitung pada 21 Maret 2016 *twitter* genap memasuki usianya yang ke 10. Media

sosial ini secara global memiliki 332 juta pengguna bulanan dengan 500 juta kicauan dikirim setiap hari dan 200 miliar kicauan dalam setahun [3].

*Cryptocurrency* adalah sebuah teknologi yang berbasis *blockchain* yang sering digunakan sebagai mata uang digital [4]. Berdasarkan Studi dari *Cambridge Center for Alternative Finance* pada tahun 2019 mencatat sekitar 139 juta pengguna akun yang diverifikasi dengan *ID cryptocurrency* dan dari jumlah tersebut ada sekitar 35 juta pengguna *cryptocurrency* di seluruh dunia yang dianggap sebagai pengguna pertukaran yang diatur [5]. Pengguna *cryptocurrency* ditemukan di hampir setiap bagian dunia seiring dengan meningkatnya minat pasar di seluruh dunia dan Secara eksplisit, tidak ada keraguan bahwa pengguna *cryptocurrency* akan terus meningkat seiring pasar yang semakin menarik.

Baru baru ini diramaikan dengan *#crypto*. *Hashtag* tersebut dimanfaatkan oleh para pengguna *twitter* untuk menyampaikan informasi mengenai mata uang kripto atau *cryptocurrency* bahkan menjadi *world trending topic* hingga bulan november 2021. *Hashtag crypto* menjadi permasalahan dalam penelitian ini untuk melihat sejauh mana sentimen masyarakat dunia terhadap *hashtag* tersebut. Dengan banyaknya opini masyarakat dunia terhadap *hashtag* tersebut sangat sulit untuk mengetahui sentimen positif, negatif ataupun netral secara manual [6]. Oleh sebab itu dibutuhkan *machine learning* dan metode yang tepat.

Diantara metode dan *machine learning* yang tepat dalam permasalahan ini adalah dengan melakukan analisis sentimen. Sentimen analisis atau *opinion mining* adalah jenis *natural language* yaitu pengolahan kata untuk melacak mood masyarakat tentang produk atau topik tertentu [7]. Untuk *machine learning* yang tepat dalam pelabelan adalah dengan menggunakan *Python TextBlob*. Selanjutnya dilakukan analisa dan pengujian text akurasi, presisi, *recall* dan *f1\_score* dengan algoritma *Naïve Bayes*.

Beberapa penelitian sebelumnya metode *Naïve Bayes* memiliki tingkat akurasi lebih tinggi dari *K-NN*. Akurasi *Naïve Bayes* sebesar 87.48% dan *k-NN* 85.40% dalam analisis sentimen tweet pornografi kaum homoseksual indonesia di *twitter* [8]. Penelitian ini memberikan solusi bagi masyarakat dunia tentang perkembangan dunia *Cryptocurrency* sehingga sempat menjadi trending topik di media sosial *twitter*. Bagi masyarakat dunia yang mau terjun di dunia *cryptocurrency* agar selalu melihat dahulu opini-opini masyarakat dunia di media sosial, terutama media sosial *twitter* dengan memasukan kata kunci *#crypto*. Penelitian ini memberikan manfaat bagi masyarakat dunia agar tidak waswas dengan trending nya *hashtag crypto* (*#crypto*). Dari hasil penelitian ini bahwa *#crypto* lebih banyak sentiment positif dibanding sentiment netral ataupun negatif. Dalam suatu penelitian diperlukan dukungan hasil-hasil penelitian yang telah ada sebelumnya yang berkaitan dengan penelitian tersebut. Sebagai contoh yang pertama ada penelitian [9] yang melakukan Analisis Sentimen Pada Review Film Dengan Menggunakan Algoritma Klasifikasi *Naive Bayes* Berdasarkan *Term Objects Keywords*. Memberikan hasil akhir hasil pengujian metode *Naïve Bayes*

*classifier* sebesar 28% dimana sampel yang digunakan adalah 100 data sentimen positif dan 100 data sentimen negatif. Penelitian dari [10] Melakukan Analisis Sentimen Pengguna *Twitter* Terhadap Polemik Persepakbolaan Indonesia Menggunakan Pembobotan *TF-IDF* dan *K-Nearest Neighbor*. Sehingga memperoleh hasil akhir dari 2000 data *tweet* berbahasa indonesia didapatkan akurasi optimal pada nilai  $k=23$  sejumlah 79.99%. Penelitian dari [11] melakukan Analisis Sentimen Dokumen *Twitter* Mengenai Dampak Virus Corona Menggunakan Metode *Naive Bayes Classifier*. Didapatkan hasil akhir dengan metode *Naive Bayes* mendapatkan akurasi 67% dan *error rate* sebesar 33%. Kemudian ada Penelitian dari [12] yang melakukan Analisis Sentimen Pemindahan Ibu Kota Indonesia Pada *Twitter*. Memberikan hasil akhir dari klasifikasi *Bernoulli Naive Bayes* mendapatkan akurasi sebesar 68,10% dengan menggunakan skenario 11. Perbedaan penelitian yang penulis lakukan dengan penelitian sebelumnya dari *keyword* dan pelabelan. peneliti menggunakan *keyword crypto* , dan pelabelan sentimen menggunakan *Python TextBlob*.

## 2. METODOLOGI PENELITIAN

Metode dalam penelitian diperlukan agar penelitian lebih terstruktur, sehingga hasil yang akan diperoleh sesuai dengan tujuan pada penelitian [13]. Adapun tahapan metode penelitian seperti pada gambar 1.



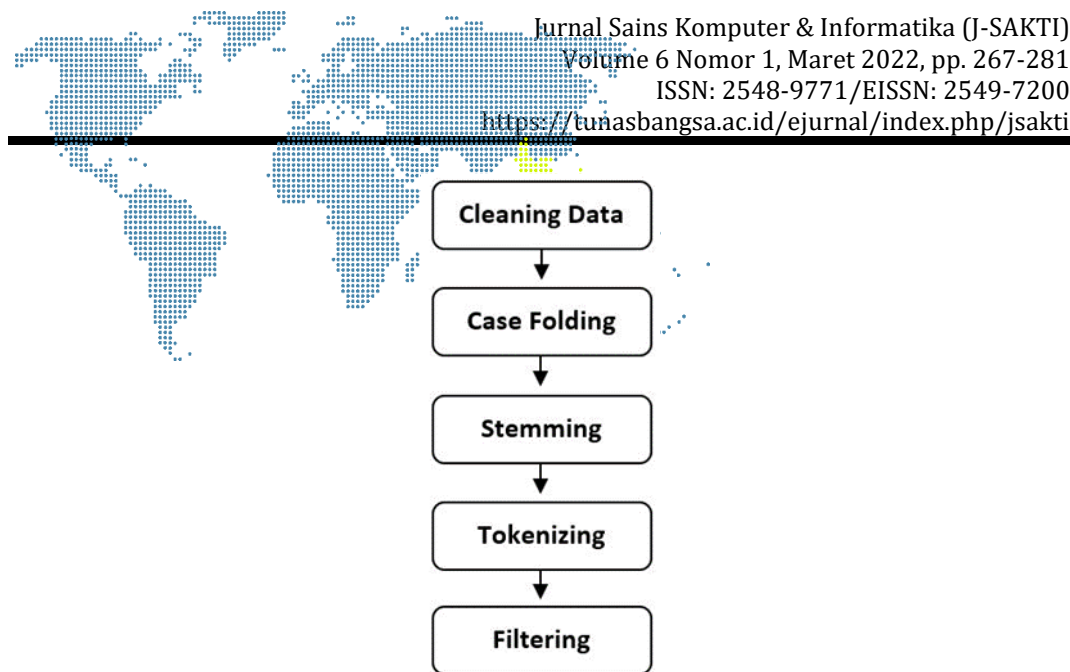
Gambar 1. Tahapan penelitian

### 2.1. Crawling Data Twitter

*Crawling* adalah aktifitas dalam membuat salinan dengan porsi yang relevan yang diambil dari sebuah *World Wide Web* [14]. Penelitian ini melakukan *crawling* di *twitter* melalui *API Twitter* dengan menggunakan modul *tweepy* pada bahasa pemrograman *Python*. Data yang dicrawl adalah *#Crypto*. Setelah itu dataset *tweet* yang masih mentah disimpan dalam bentuk *CSV (Comma Separated Values)* yang nantinya akan di proses lebih lanjut. *Crawling* data dilakukan pada tanggal 16 Desember 2021 dengan memasukan kata kunci *#crypto*. Data terkumpul sebanyak 1076 cuitan.

### 2.2. Preprocessing Data

Setelah proses *crawling* data dan disimpan dalam bentuk *CSV* maka langkah selanjutnya adalah *preprocessing data*, dikarenakan dataset tersebut belum terstruktur. Tugas utama *preprocessing data* adalah untuk menghilangkan serta mengatasi noise data agar hasil perhitungan optimal. Adapun tahapan dalam *preprocessing data* diantara lain, *cleaning data*, *case folding*, *stemming*, *tokenizing* dan *filtering*. Berikut ini adalah tahapan *preprocessing data* yang ditunjukkan pada gambar 2.



**Gambar 2.** Tahapan *preprocessing data*

### 2.2.1. *Cleaning*

Karena data masih terdapat noise maka dilakukan *cleaning* dengan menghilangkan karakter, tanda baca, *hashtag*, alamat *URL*, angka, *username* dan *retweet text*.

### 2.2.2. *Case Folding*

Setelah dilakukan *cleaning* selanjutnya adalah *case folding*. *Case folding* adalah penyeragaman semua bentuk huruf, baik itu huruf besar maupun huruf kecil [15].

### 2.2.3. *Stemming*

*Stemming* adalah tahap mencari akar kata dengan menghilangkan imbuhan pada sebuah kata [16].

### 2.2.4. *Tokenizing*

*Tokenizing* merupakan sebuah konsep yang mencakup proses sederhana dimana urutan teks dipecah menjadi bagian-bagian yang lebih kecil atau token dan kemudian dimasukkan sebagai input ke dalam model *Natural language processing (NLP)* [17].

### 2.2.5. *Filtering*

*Filtering* merupakan tahapan untuk menghilangkan kata yang tidak memiliki makna atau *stopword*. Sedangkan *stopword* adalah kosakata yang bukan ciri/unit dari suatu dokumen. *Stopwords* umumnya digunakan dalam pengambilan informasi salah satu contohnya adalah mesin pencari google. Pengurangan indeks dalam teks dengan penghilangan beberapa kata kerja, kata sifat, dan kata keterangan lainnya dapat dimasukkan ke dalam daftar *stopwords* [18].

## 2.2. Pelabelan Otomatis

Setelah dilakukan *preprocessing* langkah selanjutnya dan yang paling penting adalah melakukan pelabelan. Namun dalam penelitian ini dilakukan pelabelan otomatis dengan *Python TextBlob*. *Python TextBlob* merupakan *library Python* (versi 2 dan 3) yang digunakan untuk memproses data tekstual. *TextBlob* menyediakan API yang dapat digunakan untuk pemrosesan bahasa alami (*NLP*) seperti ekstraksi frase kata benda, analisis sentimen, klasifikasi, penerjemahan, dll [19]. Hasil objek *TextBlob* digunakan untuk memproses pembelajaran bahasa alami dan *library TextBlob* hanya dapat mengenali Bahasa Inggris.

## 2.3. Klasifikasi Metode *Naïve Bayes*

Klasifikasi metode *Naïve Bayes* adalah metode yang memanfaatkan statistika sederhana dengan dasar *teorema bayes* yang mengasumsikan ada atau tidaknya suatu kelas dengan fitur yang lain. Metode NBC digunakan untuk proses klasifikasi untuk menentukan *f1\_score accuracy*, *recall*, *precision* [20]. Adapun rumus nya bisa di lihat dibawah ini :

$$P(H | X) = \frac{P(X|H)P(H)}{P(X)} \quad (1)$$

Dimana	:
X	: Data dengan class yang belum diketahui
H	: Hipotesis data merupakan suatu class spesifik
P (H X)	: Probabilitas hipotesis H berdasar kondisi X (posteriori probabilitas)
P (H)	: Probabilitas hipotesis H (prior probabilitas)
P (X H)	: Probabilitas X berdasarkan kondisi pada hipotesis
H P(X)	: Probabilitas X

Pada penelitian ini, peneliti menggunakan fitur *NLTK* untuk mengklasifikasi *text mining* dengan metode *Naïve Bayes*. *NLTK* adalah salah satu *tools* yang sangat populer pada ilmu *Natural Language Processing (NLP)* dengan menggunakan bahasa pemrograman *Python*. *Natural Language Toolkit* ini sangat mendukung proses pengolahan bahasa natural seperti *classification*, *tokenization*, *stemming*, *tagging*, *parsing* dan *filtering*.

Setelah dilakukan *preprocessing* langkah selanjutnya dan yang paling penting adalah melakukan pelabelan. Namun dalam penelitian ini dilakukan pelabelan otomatis dengan *Python TextBlob*. *Python TextBlob* merupakan *library Python* (versi 2 dan 3) yang digunakan untuk memproses data tekstual. *TextBlob* menyediakan API yang dapat digunakan untuk pemrosesan bahasa alami (*NLP*) seperti ekstraksi frase kata benda, analisis sentimen, klasifikasi, penerjemahan, dll [19]. Hasil objek *TextBlob* digunakan untuk memproses pembelajaran bahasa alami dan *library TextBlob* hanya dapat mengenali Bahasa Inggris.

### 3. HASIL DAN PEMBAHASAN

#### 3.1. Hasil *Crawling Data*

Berikut adalah hasil *crawling data* yang dilaksanakan pada tanggal 16 Desember 2021 dengan memasukan kata kunci #crypto. Data terkumpul sebanyak 1076 cuitan.

	user	tanggal	tweet
0	Piku	2021-12-16 09:34:45	@CryptoTownEU Excellent project.I hope it will...
1	Stellar Australia	2021-12-16 09:34:38	This open-source wallet is Stellar's answer to...
2	Ripon Roy 🇮🇩	2021-12-16 09:34:28	@ErthaGame Best project and legit. \nThanks sir...
3	Ei Nge	2021-12-16 09:34:28	@DevWynngames The project has great prospects....
4	Purobi	2021-12-16 09:34:27	#xceldefi #xcelswap #BSC #XLD #DeFi #XcelLab #...
...	...	...	...
1071	Deuterium Labs.	2021-12-16 09:01:57	ethereum Price: \$4037.58 USD \n ethereum...
1072	Crypto Bro	2021-12-16 09:01:55	What #crypto will go 10x from here?
1073	Dhikawahyudi	2021-12-16 09:01:55	@CryptoTownEU @metaplacebsc The Best Airdrop J...
1074	cryptocurrencyliveprices.com	2021-12-16 09:01:49	The price of #Avalanche is currently \$103.28\n...
1075	cryptoon	2021-12-16 09:01:48	Hohohoholy shiiiiit! 🇮🇩\n\nhttps://t.co/fRTY...

1076 rows x 3 columns

Gambar 3. Hasil *crawling data*

#### 3.2. Hasil *Preprocessing*

Berikut adalah hasil *preprocessing* dengan mengambil satu sampel text. Dimulai dari data mentah dari hasil proses *crawling*, selanjutnya proses *cleaning*, *case folding*, *stemming*, *tokenizing* dan *filtering*.

Tabel 1. Hasil *preprocessing*

Preprocessing	Tweets
Data Mentah	@CryptoTownEU Excellent project.I hope it will be one of the best project in the airdrop history. good project and good teamâî, @Js83196841 @kmmanik01 @sajibjasim #Airdrop #Airdrops #Crypto
<i>Cleaning</i>	Excellent projectI hope it will be one of the best project in the airdrop history good project and good teamâî, Airdrop Airdrops Crypto
<i>Case Folding</i>	excellent projecti hope it will be one of the best project in the airdrop history good project and good teamâî, airdrop airdrops crypto
<i>Stemming</i>	excellent projecti hope it will be one of the best project in the airdrop history good project and good team airdrop airdrops crypto
<i>Tokenizing</i>	['excellent', 'projecti', 'hope', 'it', 'will', 'be', 'one', 'of', 'the', 'best', 'project', 'in', 'the', 'airdrop', 'history', 'good', 'project', 'and', 'good', 'team', 'airdrop', 'airdrops', 'crypto']
<i>Filtering</i>	['excellent', 'projecti', 'hope', 'one', 'best', 'project', 'airdrop', 'history', 'good', 'project', 'good', 'team', 'airdrop', 'airdrops', 'crypto']

### 3.3. Hasil Pelabelan Otomatis

Untuk pelabelan otomatis dilakukan menggunakan *Python TextBlob*. Berikut *script* python untuk melakukan pelabelan otomatis.

```
df=pd.read_excel('labeling.xlsx')
df['label'] = ''
for i,x in df.tweets.iteritems():
    label = TextBlob(x)
    df['label'][i] = label.sentiment.polarity
    print("Index: ", i, "label" , label.sentiment.polarity)
df
```

Script diatas merupakan pembobotan nilai sentimen text dengan *library TextBlob*. Hasil pembobotan text bisa dilihat pada gambar 4.

	tweets	label
0	excellent projecti hope one best project airdr...	0.85
1	opensource wallet stellars answer metamask xlm...	0.25
2	best project legit thanks sir sharing best cry...	0.733333
3	project great prospects certainly momenti beli...	0.671429
4	xceldefi xcelswap bsc xld defi xcellab dex cry...	0.35
...	...	...
1027	ethereum price usd ethereum market cap usd eth...	0
1028	crypto go x	0
1029	best airdrop join guys airdrop crypto airdrops...	1
1030	price avalanche currently crypto avax	0
1031	hohohoholy shiiiiit freakin cool visualisation...	0.35

**Gambar 4.** Hasil pembobotan *sentiment text*

Setelah itu membuat sebuah fungsi. Peneliti menyebutnya *polarity\_to\_label* dan itu akan memiliki *x* sebagai parameter. Fungsi ini akan memiliki kolom label berlalu ke dalamnya di mana ia akan memeriksa nilai: Jika itu antara -1 dan 0 maka itu berubah menjadi *neg* untuk negatif, jika itu sama dengan 0 maka akan berubah menjadi *netral*, dan jika itu antara 0 dan 1 kemudian diubah menjadi *pos* positif. Berikut *script* nya :

```
def polarity_to_label(x):
    if(x >= -1 and x < 0):
        return 'neg'
    if(x == 0):
        return 'netral'
    if(x > 0 and x <= 1):
        return 'pos'
    df.label = df.label.apply(polarity_to_label)
```

Lalu kita panggil fungsi tersebut dengan `df`. Setelah di `run` maka akan muncul hasil seperti pada gambar 5.

	tweets	label
0	excellent projecti hope one best project aidr...	pos
1	opensource wallet stellars answer metamask xlm...	pos
2	best project legit thanks sir sharing best cry...	pos
3	project great prospects certainly momenti beli...	pos
4	xceldefi xcelswap bsc xld defi xcellab dex cry...	pos
...	...	...
1027	ethereum price usd ethereum market cap usd eth...	neutral
1028	crypto go x	neutral
1029	best airdrop join guys airdrop crypto airdrops...	pos
1030	price avalanche currently crypto avax	neutral
1031	hohohoholy shiiiiit freakin cool visualisation...	pos

1032 rows x 2 columns

**Gambar 5.** Proses pembobotan menjadi teks

Selanjutnya kita akan melihat berapa total sentimen positif, netral dan negatif dengan menuliskan `df.label.value_counts()`. Hasil bisa dilihat di gambar 6.

```
pos      632
neutral  296
neg      104
Name: label, dtype: int64
```

**Gambar 6.** Hasil akhir pelabelan sentiment text

Setelah melakukan pelabelan otomatis menggunakan `Python TextBlob` total data menjadi 1032 `tweet`. Yang termasuk kedalam sentiment positif sebanyak 632 data, netral 296 data dan negatif 104 data. Perhitungan *Persentase* klasifikasi sentimen diperoleh berdasarkan rumus berikut:

$$\frac{\text{Jumlah komentar yang terklasifikasi}}{\text{Jumlah seluruh komentar}} \times 100\%$$

$$\text{Positif} = \frac{632}{1032} \times 100\% = 61.24\%$$

$$\text{Netral} = \frac{296}{1032} \times 100\% = 28.68\%$$

$$\text{Negatif} = \frac{104}{1032} \times 100\% = 10.07\%$$



### 3.4. Klasifikasi Dengan Metode Naïve Bayes

Proses klasifikasi teks diambil dari hasil filtering dengan tweet ['excellent', 'projecti', 'hope', 'one', 'best', 'project', 'airdrop', 'history', 'good', 'project', 'good', 'team', 'airdrop', 'airdrops', 'crypto']. Hasil *filtering* tersebut dimasukkan ke dalam *script Python*. *Script* bisa dilihat dibawah ini :

```
import pandas as pd
columns = ['tweets', 'sentimen']
rows = []
'''
rows = [['excellent projecti hope it will be one of the best project in the
airdrop history good project and good team airdrop airdrops crypto', 'positif']
'''
rows=[['excellent', 'positif'],
      ['projecti', 'positif'],
      ['hope', 'positif'],
      ['one', 'netral'],
      ['best', 'positif'],
      ['project', 'positif'],
      ['airdrop', 'netral'],
      ['history', 'netral'],
      ['good', 'positif'],
      ['project', 'positif'],
      ['good', 'positif'],
      ['team', 'positif'],
      ['airdrop', 'netral'],
      ['airdrops', 'netral'],
      ['crypto', 'netral']]
training_data = pd.DataFrame(rows, columns=columns)
training_data
```

Pelabelan per kata dilakukan secara manual sehingga di dapatkan label positif berjumlah 9, label netral 6 dan label negatif 0. Selanjut nya menghitung probabilitas dari sentimen positif dan sentimen netral. Untuk probabilitas sentimen negatif tidak dihitung dikarenakan di dalam teks tersebut tidak ada yang mengandung sentimen negatif. Untuk menghitung probabilitas dari sentimen positif dan netral digunakan fitur *NLTK (Natural Language Toolkit)* dengan bahasa pemrograman *Python*. Pertama menghitung probabilitas sentimen positif terhadap teks . *Script* bisa dilihat dibawah ini.

```
import nltk
nltk.download('punkt')
```

```

from nltk.tokenize import word_tokenize
new_sentence = 'best excelent good hope project projecti team'
new_word_list = word_tokenize(new_sentence)
print(new_word_list)

prob_s_with_ls = []
for word in new_word_list:
    if word in freq_s.keys():
        count = freq_s[word]
    else:
        count = 0
    prob_s_with_ls.append((count + 1)/(total_cnts_features_s + total_fea
tures))
dict(zip(new_word_list,prob_s_with_ls))
    
```

*Script* diatas merupakan perhitungan probabilitas sentimen positif. Hasil bisa dilihat pada tabel di bawah ini.

**Tabel 2.** Hasil perhitungan probabilitas positif

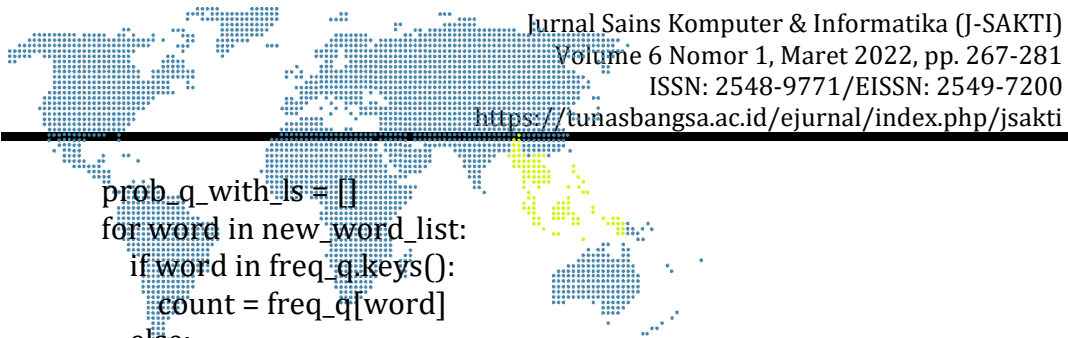
Tweets	Nilai Probabilitas Positif
Best	0.09523809523809523
Excelent	0.09523809523809523
Good	0.14285714285714285
Hope	0.09523809523809523
Project	0.14285714285714285
Projecti	0.09523809523809523
Team	0.09523809523809523

Setelah itu jumlahkan nilai masing-masing probabilitas *tweets* dengan menggunakan *script* di bawah ini.

```

i=0
prob_s_=1
for i in range(len(prob_s_with_ls)):
    prob_s_=prob_s_*prob_s_with_ls[i]
print (prob_s_)
print ("Nilai akhir positif ",prob_s_*(9/15))
    
```

Hasil penjumlahan probabilitas positif *tweets* di dapatkan nilai sebesar 0.0000015990329927927727. Untuk melihat hasil akhir probabilitas sentimen positif maka total sentiment positif dikali (Sentiment Positif /total text), sehingga memperoleh nilai sebesar 0.000000009594197956756635. Selanjutnya menghitung probabilitas netral terhadap sentimen positif. *Script* perhitungan bisa dilihat dibawah ini.



```
prob_q_with_ls = []  
for word in new_word_list:  
    if word in freq_q.keys():  
        count = freq_q[word]  
    else:  
        count = 0  
    prob_q_with_ls.append((count + 1)/(total_cnts_features_q + total_fe  
atures))  
dict(zip(new_word_list,prob_q_with_ls))
```

Hasil script di atas di dapatkan nilai sebagai berikut.

**Tabel 3.** Hasil perhitungan probabilitas netral

Tweets	Nilai Probabilitas Positif
Best	0.05555555555555555
Excelent	0.05555555555555555
Good	0.05555555555555555
Hope	0.05555555555555555
Project	0.05555555555555555
Projecti	0.05555555555555555
Team	0.05555555555555555

Setelah itu jumlahkan nilai masing-masing probabilitas *tweets* netral dengan menggunakan script di bawah ini.

```
i=0  
prob_q=1  
for i in range(len(prob_q_with_ls)):  
    prob_q=prob_q*prob_q_with_ls[i]  
print (prob_q_)  
print("Nilai akhir Netral=",prob_q_*(6/15))
```

Hasil penjumlahan probabilitas *tweets* netral di dapatkan nilai sebesar 0.000000001633399672881007. Untuk melihat hasil akhir probabilitas sentimen netral maka total sentiment netral dikali (sentiment netral : total text) dibagi dengan total sentiment positif dan sentiment netral, sehingga memperoleh nilai sebesar 0.0000000006533598691524029.

**Tabel 4.** Perbandingan nilai akhir sentiment positif dan sentimen netral

Nilai Akhir Sentiment Positif	Nilai Akhir Sentiment Netral
Best0.000000009594197956756635	0.0000000006533598691524029

Pengklasifikasian nilai probabilitas sentimen positif dan nilai probabilitas netral sudah selesai. Dari hasil pengklasifikasian dengan metode Naïve Bayes maka text ['excellent', 'projecti', 'hope', 'one', 'best', 'project',

'airdrop', 'history', 'good', 'project', 'good', 'team', 'airdrop', 'airdrops', 'crypto'] masuk ke dalam sentimen positif.

Pada bagian ini diberikan hasil penelitian yang dilakukan sekaligus dibahas secara komprehensif. Hasil bisa berupa gambar, grafik, tabel dan lain-lain yang mempermudah pembaca paham dan diacu di naskah. Jika bahasan terlalu panjang dapat dibuat sub-sub judul, seperti contoh berikut.

### 3.5. Evaluasi dan Pengujian

Setelah dilakukan *crawling*, *preprocessing*, pelabelan otomatis dengan *TextBlob*, dan pengklasifikasian dengan Naïve Bayes maka selanjutnya adalah mengukur *accuracy*, *precision*, *recall* dan *F1\_Score* menggunakan library Sklearn di dalam bahasa pemrograman Python dengan data testing sebesar 0.2 dan data training 0.8. adapun *script* nya bisa dilihat dibawah ini.

```
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix

x = MultinomialNB().fit(X_train, y_train)
predicted = x.predict(X_test)

print(confusion_matrix(y_test, predicted))
print('_____ \n')
print(classification_report(y_test, predicted, zero_division=0))
print("accuracy", accuracy_score(y_test, predicted))
print("precision", precision_score(y_test, predicted, average='macro'))
print("recall", recall_score(y_test, predicted, average='macro'))
print("f1_score", f1_score(y_test, predicted, average='macro'))
```

*Output script* di atas bisa dilihat pada gambar dibawah ini

```
[[ 11  5  7]
 [  0 25 45]
 [  0  1 113]]

precision  recall  f1-score  support
neg         1.00    0.48    0.65     23
neutral     0.81    0.36    0.50     70
pos         0.68    0.99    0.81    114

accuracy          0.72    207
macro avg         0.83    0.61    0.65    207
weighted avg     0.76    0.72    0.69    207

accuracy 0.7198067632850241
precision 0.8304333659172368
recall 0.6088772656278377
f1_score 0.6507147235912711
```

**Gambar 7.** Hasil pengujian menggunakan *Naïve Bayes*

Pada gambar 7, didapatkan nilai *accuracy* sebesar 0.7198, *precision* 0.8304, *recall* 0.6088 dan *f1\_Score* 0.6507. ketika dirubah ke bilangan persen maka hasil *accuracy* menjadi 71.98%, *precision* 83.04%, *recall* 60.88% dan *f1\_score* 65.07%.

#### 4. SIMPULAN

Dari penelitian yang telah dilakukan, maka dapat ditarik kesimpulan bahwa Analisis Sentimen dapat digunakan untuk mengetahui sentimen masyarakat dunia terhadap *hashtag* yang sempat trending di media sosial *twitter* yaitu *#crypto*. Tujuan nya untuk mengetahui seberapa positif,netral atau negatif terhadap *hashtag* tersebut. Setelah dilakukan pelabelan otomatis dengan *Python TextBlob* dengan total 1032 *tweet* maka jumlah sentimen positif sebanyak 61.24%, Sentimen netral sebanyak 26.68% dan sentimen negatif 10.07%. Dari hasil proses pengujian menggunakan algoritma Naïve Bayes dengan masing masing data testing dan data training adalah 0.2 dan 0.8 sehingga menghasilkan nilai *accuracy* sebesar 71.98%, *precision* 83.04%, *recall* 60.88% dan *f1\_score* 71.98%. Dengan nilai *accuracy*, *precision*, *recall* dan *f1\_score* tersebut, artinya analisis sentimen terhadap *#crypto* di media sosial *twitter* cukup baik.

#### DAFTAR PUSTAKA

- [1] B. W. Sari and F. F. Haranto, "Implementasi Support Vector Machine Untuk Analisis Sentimen Pengguna Twitter Terhadap Pelayanan Telkom Dan Biznet," *J. Pilar Nusa Mandiri*, vol. 15, no. 2, pp. 171–176, 2019, doi: 10.33480/pilar.v15i2.699.
- [2] S. Nurul, J. Fitriyyah, N. Safriadi, and E. E. Pratama, "Analisis Sentimen Calon Presiden Indonesia 2019 dari Media Sosial Twitter Menggunakan Metode Naive Bayes," vol. 5, no. 3, pp. 279–285, 2019.
- [3] Y. S. Mahardhika, E. Zuliarso, P. Studi, T. Informatika, F. T. Informasi, and U. Stikubank, "ANALISIS SENTIMEN TERHADAP PEMERINTAHAN JOKO WIDODO PADA MEDIA SOSIAL TWITTER MENGGUNAKAN ALGORITMA NAIVES BAYES," no. 2015, pp. 409–413, 2018.
- [4] I. Bagus and P. Bhiantara, "Teknologi Blockchain Cryptocurrency Di Era Revolusi Digital," *Jl. Udayana Kampus Teng.*, no. 0362, p. 27213, 2018, [Online]. Available: <http://pti.undiksha.ac.id/senapati>.
- [5] I. Danco, "Jumlah Pengguna Crypto di Seluruh Dunia Meningkat 16%, Mencapai 106 Juta," 2021. <https://id.bitcoinethereumnews.com/crypto/number-of-crypto-users-worldwide-strikes-increase-of-16-hits-106m/>.
- [6] F. I. Sulaiman, W. W. Winarno, and M. P. Kurniawan, "Perancangan Aplikasi Klasifikasi Sentimen Berbasis Web Terhadap Mata Uang Kripto," 1384.
- [7] A. Deviyanto and M. D. R. Wahyudi, "Penerapan Analisis Sentimen Pada Pengguna Twitter Menggunakan Metode K-Nearest Neighbor," *JISKA (Jurnal Inform. Sunan Kalijaga)*, vol. 3, no. 1, p. 1, 2018, doi:

- 10.14421/jjska.2018.31-01.
- [8] A. M. Pudjajana and D. Manongga, "Sentimen Analisis Tweet Pornografi Kaum Homoseksual Indonesia Di Twitter Dengan Naive Bayes," *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 9, no. 1, pp. 313–318, 2018, doi: 10.24176/simet.v9i1.1922.
- [9] I. W. S. Nirawana and I. G. Indrawan, "Analisis Sentimen Pada Review Film Dengan Menggunakan Algoritma Klasifikasi Naive Bayes Berdasarkan Term Objects Keywords," *Semin. Nas. Pendidik. Tek. Inform. (SENAPATI 2016)*, no. Senapati, pp. 172–174, 2016.
- [10] J. A. Septian, T. M. Fahrudin, and A. Nugroho, "Journal of Intelligent Systems and Computation 43," pp. 43–49, 2019, [Online]. Available: <https://t.co/9Wl0aWpfd5>.
- [11] N. M. A. J. Astari, Dewa Gede Hendra Divayana, and Gede Indrawan, "Analisis Sentimen Dokumen Twitter Mengenai Dampak Virus Corona Menggunakan Metode Naive Bayes Classifier," *J. Sist. dan Inform.*, vol. 15, no. 1, pp. 27–29, 2020, doi: 10.30864/jsi.v15i1.332.
- [12] E. Mas'udah, E. D. Wahyuni, and A. A. Arifiyanti, "Analisis Sentimen: Pindahan Ibu Kota Indonesia Pada Twitter," *J. Inform. dan Sist. Inf.*, vol. 1, no. 2, pp. 397–401, 2020.
- [13] R. Rosdiana, T. Eddy, S. Zawiyah, and N. Y. U. Muhammad, "Analisis Sentimen pada Twitter terhadap Pelayanan Pemerintah Kota Makassar," *Proceeding SNTEI*, pp. 87–93, 2019.
- [14] I. N. Husada, E. H. Fernando, H. Sagala, A. E. Budiman, and H. Toba, "Ekstraksi dan Analisis Produk di Marketplace Secara Otomatis dengan Memanfaatkan Teknologi Web Crawling," *J. Tek. Inform. dan Sist. Inf.*, vol. 5, no. 3, pp. 350–359, 2020, doi: 10.28932/jutisi.v5i3.1977.
- [15] A. Fathan Hidayatullah and A. Sn, "ISSN: 1979-2328 UPN "Veteran," *Semin. Nas. Inform.*, vol. 2014, no. semnasIF, pp. 115–122, 2014, [Online]. Available: <http://www.situs.com>.
- [16] N. Ruhyana, "Analisis Sentimen Terhadap Penerapan Sistem Plat Nomor Ganjil / Genap Pada Twitter Dengan Metode Klasifikasi Naive Bayes," *J. IKRA-ITH Inform.*, vol. 3, no. 1, pp. 94–99, 2019.
- [17] T. Jamaluddin, M. A. Bijaksana, and I. Asror, "Perbandingan Algoritma Sentencepiece BPE dan Unigram Pada Tokenisasi Artikel Bahasa Indonesia Pendahuluan Studi Terkait," *e-Proceeding Eng.*, vol. 7, no. 2, pp. 8323–8331, 2020.
- [18] R. Sari and R. Y. Hayuningtyas, "Penerapan Algoritma Naive Bayes Untuk Analisis Sentimen Pada Wisata TMII Berbasis Website," *Indones. J. Softw. Eng.*, vol. 5, no. 2, pp. 51–60, 2019, doi: 10.31294/ijse.v5i2.6957.
- [19] R. Parlita, S. I. Pradika, A. M. Hakim, and K. R. N. M., "Analisis Sentimen Twitter Terhadap Bitcoin dan Cryptocurrency Berbasis Python TextBlob," *J. Ilm. Teknol. Inf. dan Robot.*, vol. 2, pp. 33–37, 2020.
- [20] M. Tri Anjasmoros and Fitri Marisa, "Analisis Sentimen Aplikasi Go-Jek Menggunakan Metode Svm Dan Nbc (Studi Kasus: Komentar



Pada Play Store).” *Conf. Innov. Appl. Sci. Technol. (CIASTECH 2020)*, no. Ciastech, pp. 489-498, 2020.