Analisis Sentimen Berbasis Aspek Terhadap *Richeese Factory* pada Platform *Twitter* Menggunakan Algoritme *Naïve Bayes* dan Menggunakan *Marketing Mix* 4P

1st Heidea Yulia Firzania
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
deafirzania@student.telkomuniversity.ac.id

2nd Oktariani Nurul Pratiwi
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
onurulp@telkomuniversity.ac.id

3rd Faqih Hamami
Fakultas Rekayasa Industri
Universitas Telkom
Bandung, Indonesia
faqihhamami@telkomuniversity.ac.id

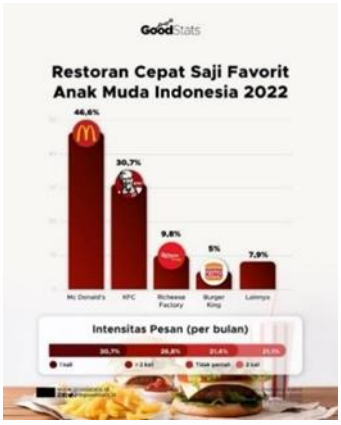
Abstrak— **Pembatasan pengunjung di gerai makanan membuatmasyarakat memesan makanan dan minuman dari rumah sehingga popularitas *brand* tersebut meningkat. Maka banyak *brand* bersaing terutama *brand* makanan cepat saji. *Richeese Factory* adalah *brand* makanan cepat saji Indonesia yang menyajikan makanan pedas, saus keju dan satu-satunya *brand* makanan yang bersaing dengan *brand* luar negeri, tetapi *brand*tersebut masih kalah dengan *brand* luar negeri. Dengan menggunakan analisis sentimen dan aspek teori *Marketing Mix* 4p yang didapat dari ulasan pelanggan pada *Twitter*, dapat meningkatkan minat masyarakat kepada *brand Richeese Factory*. Sistematika penyelesaian menggunakan Knowledge Discovery in Database, tahapan pertama adalah data selection, pada tahap tersebut, peneliti menentukan keyword pengambilan data dari *Twitter*, yang kemudian data diambil dengan cara crawling data. Hasil dari implementasi Algoritme Naïve Bayes perbedaan *max features* dan *test size* memiliki peran penting pada hasil akurasi. Algoritme Gaussian dan Multinomial Naïve Bayes memiliki hasil akurasi yang lebih tinggi daripada Algoritme Bernoulli Naïve Bayes, akan tetapi sebagian besar akurasi tertinggi pada Algoritme Multinomial Naïve Bayes dengan nilai 84%. Berdasarkan tingkat akurasi dari implementasi Algoritme Naïve Bayes jika nilai *max features* semakin tinggi nilai akurasi juga semakin tinggi.**

***Kata kunci*— Richeese Factory, Analisis Sentiment, Marketing Mix, Naïve Bayes**

I. PENDAHULUAN

Dalam rangka menghentikan penyebaran virus Covid-19, pembatasan dilakukan pada tempat ibadah, sarana olahraga, pusatperbelanjaan, bahkan pada restoran dan gerai [2]. Pembatasan tersebut membuat pola belanja masyarakat Indonesia mengalami perubahan secara signifikan. Perubahan yang dimaksud adalah masyarakat yang terbiasa membeli makan di restoran lebih memilih untuk membeli makan melalui daring. Dari perubahan pola memesan makanan dan minuman dari rumah oleh masyarakat Indonesia, tidak dipungkiri bahwa peminat dan popularitas makanan serta minuman di Indonesia meningkat. Pernyataan tersebutdibuktikan pada riset *Populix* pada tahun 2020 [3]. Maka dari itu, banyak *brand* makanan dan minuman yang bersaing satusama lain. Persaingan *brand* makanan cepat saji di Indonesia dikuasai oleh *brand* makanan dari luar negeri. Seperti pada riset yang dilakukan oleh *GoodStats*, restoran cepat saji pilihan masyarakat Indonesia *Richeese Factory* berada pada urutan ke 5 berdasarkan Gambar 1 [4] dan restoran cepat saji favorit anak muda, *Richeese Factory* berada pada urutan ke 3 berdasarkan Gambar 2 [5]. Dari riset tersebut hanya ada satu *brand* yang merupakan *brand* dari Indonesia, yaitu *Richeese Factory.*

Gambar 1



Gambar 2

Dapat dilihat dari Gambar 1 dan 2, kepopuleran *Richeese Factory* masih kurang menonjol apabila dibandingkandengan *Brand* dari luar negeri. Sehingga untuk meningkatkan nilai atau value serta jumlah pelanggan *Richeese Factory* dibutuhkan Analisis Sentimen berbasis Aspek menggunakanAlgoritme *Naive Bayes* dan *Marketing Mix* 4P yang diambil dari proses *crawling data* pada platform *Twitter*.

II. KAJIAN TEORI

A. *Richeese Factory*

Richeese Factory merupakan sebuah restoran siap saji yang berasal dari Indonesia, menu utama dari restoran tersebut adalah ayam goreng dengan saus pedas dan saus keju. *Richeese Factory* adalah restoran yang dimiliki oleh PT. Richeese Kuliner Indonesia yang merupakan anak usaha dari Kaldu Sari Nabati. Gerai pertama *Richeese Factory* berada di pusat perbelanjaan *Paris Van Java*, Bandung, tepatnya pada 8 Februari 2011 [6].

Kepopuleran *Richeese Factory* selama beberapa tahun kebelakang dapat dilihat dari bertambahnya jumlah gerai. Sumber terbaru menyatakan jumlah gerai *Richeese Factory* berjumlah sebanyak 177 [7]. Selain dari faktor bertambahnya gerai *Richeese Factory*, kepopuleran *Richeese Factory* tidak luput dari promo yang diberikan oleh *brand* makanan cepat saji tersebut. Seperti dikutip dari *Instagram Richeese Factory*, pada sepanjang tahun 2022 *Richeese* selalu memberikan promo untuk dapat menarik pelanggan [8].

B. *Twitter*

Twitter adalah sebuah media sosial yang menarik perhatian pengguna internet di seluruh dunia, karena *Twitter* merupakan *platform* yang mudah digunakan dalam bertukar informasi dengan setiap individu di dunia secaralangsung. Pada awalnya, *Twitter* berbentuk *micro-blogging* yang dibuatpada tahun 2006 oleh Jack Dorsey, Biz Stone, dan Evan Williams. Nama awal *Twitter a*dalah *Twttr* yang dibuat untuk sebuah layanan SMS (*short message service*) dan digunakan untuk berkomunikasi dalam satuan kelompok kecil. Dengan perkembangan teknologi, *Twitter* dapat diakses oleh setiap individu di seluruh dunia [9].

C. *Marketing Mix* 4P

Marketing Mix 4P merupakan sebuah alat pemasaran yang digunakan perusahaan dalam mencapai tujuan pemasaran sehingga dapat memenuhi target pasar. Faktor-faktor yang mempengaruhi keberhasilan pemasaran sebuah barang atau jasamenurut *Marketing Mix* 4P adalah *Product* (Produk), *Price* (Harga), *Place* (Tempat) dan *Promotion* (Promosi). *Product* merupakan sebuah barang atau jasayang menjadi fokus utama untuk dipasarkan kepada pelanggan dengan harga tertentu. *Price* merupakan nilai dari sebuah barang ataujasa. *Place* merupakan dimana barang atau jasa dipindahkan dari produsen ke pelanggan. *Promotion* merupakan cara efektif untuk mendorong pelanggan dalam membeli produk atau jasa [10].

D. *Text Mining*

Text Mining merupakan suatu proses yang bertujuan untuk menemukan sebuah informasi di dalam kumpulan teks yang besar, serta digunakan untuk mengidentifikasi hubungan yang terdapat pada teks. Ketika melakukan *text mining* dibutuhkan pengetahuan di bidang *Data Mining*, *natural language processing*, *machine Learning*, dan pencarian informasi, karena *text mining* adalah ilmu yang berkaitan dengan bidang-bidang tersebut. *Text mining* adalah disiplin ilmu yang saling berkaitan, karena kedua disiplin ilmu ini memiliki tujuan yang sama, yaitu mencari adanya hubungan dalam sebuah data. Perbedaan dari kedua disiplin ilmu tersebut adalah sumber data yang digunakan, *text mining* berupa teks, sedangkan *Data Mining* berupa angka, sehinggamenyebabkan *text mining* cenderung lebih sulit [10].

E. *Text Preprocessing*

Text Preprocessing adalah klasifikasi teks yang merupakan tahapan awal dari *text mining*. Tahap *text preprocessing* bertujuan untuk mengurangi adanya kesalahan pada *data raw*, tahapan tersebut dilakukan sebelum analisis. Dalam *text mining* umumnya menggunakan data dari internet, biasanya data tersebut tidak lengkap, berantakan, dan tidak konsisten, sehingga dibutuhkan *text preprocessing*agar data yang diambil lebih rapi. Maka dari itu tahapan *text preprocessing* adalah tahapan yang cukup penting dalamprosesnya, karena data yang dihasilkan dari tahapan *text preprocessing* dapat meningkatkan kualitas data tersebut.

Saat melakukan *text preprocessing* faktor utama dalam memengaruhi keberhasilan adalah representasi dan kualitas pada data yang dimiliki. Ketika pada data yang dimiliki, terdapat data banyak data yang tidak relevan, memiliki banyak *noise*, adanya redudansi data, serta data yang tidak handal, menyebabkan akan sulit ditemukan informasi saat *process training* [11]. Pada tahap *text preprocessing*, terdapat

4 tahapan antara lain *remove punctuation*, *case folding*, *stemming*, dan *tokenizing*, penjelasan sebagai berikut.

a) Remove Punctuation

Remove Punctuation merupakan proses untuk menghilangkan tanda baca pada teks, tanda baca yang dimaksud merupakan karakter dalam membedakan kalimat dan bagian penyusunnya, serta memperjelas makna dari kalimat tersebut [12].

b) Case Folding

Case Folding merupakan tahapan setelah dilakukan penghilangan tandabaca, pada tahap *case folding* semua huruf kapital diubah menjadi huruf kecil [12].

c) Stemming

Stemming merupakan proses pemecahan kata dan diubah menjadi kata dasar atau dengan kata lain, menghilangkan awalan dan akhiran dari sebuah kalimat [12].

d) Tokenizing

Tokenizing merupakan proses terakhir dalam *text preprocessing*, pada *tokenizing* kalimat yang telah dilakukan *stemming* akan dipecah menjadi beberapa bagian [12].

F. Analisis Sentimen

Analisis Sentimen yang juga dikenal sebagai *opinion mining* merupakan proses *automated* dalam mengidentifikasi dan melakukan ekstraksi informasi yang subjektif dari sebuah teks. Informasi tersebut dapat berupa pendapat, penilaian, atau perasaan mengenai topik atau subjek tertentu. Analisis Sentimen membantu memastikan maksud dari pembicara atau penulis terhadap suatu topik atau objek tertentu. Secara luas analisis sentimen dibagi menjadi dua jenis, analisis sentimen berbasis aspek dan analisis sentimen berbasis objektivitas [13].

Analisis Sentimen berbasis aspek sering digunakan untuk membantu sebuah*brand*, produk atau jasa untuk mengetahuipenilaian dari barang tersebut atau layanan yang diberikan, agar dapat dilakukan evaluasi kembali sehingga dapat meningkatkan performa *brand*, produk atau jasa tersebut. Jenis analisis sentimen yang paling umum adalah *polaritydetection*, analisis tersebut melibatkan klasifikasi pernyataan positif. negatif, atau netral [10], [13].

Analisis Sentimen yang bersifat positif mengandung kalimat dengan makna yang positif seperti; senang, bagus, enak, nyaman, sedangkan *sentiment* yang bersifat negatif mengandung makna yang negatif seperti; sedih, mahal, sedikit, dan lain sebagainya. Analisis Sentimen juga dapat mengandung kalimat dengan makna netral, kalimat tersebut mengekspresikan subjektivitas yang berbeda seperti spekulasi dan yang tidak memiliki polaritas *positive* atau *negative* [10].

A. Multi Output Classification

Multi Output Classification adalah sebuah *machine Learning* yang dapat melakukan prediksi dengan *multiple output simultaneously*. Di dalam *Multi Output Classification*, model yang didapatkan adalah dua atau lebih *outputs* setelah dibuat prediksinya [14]. *Multi Output Classification* memiliki pendekatan yang cukup kompleks dari *Multilabel Classification*, karena *Multi Output Classification* dapat mensupport prediksi pendekatan untuk *Multi Class* dan *Multilabel Classification* [15].

B. TF-IDF (*Term Frequency-Inverse Document Frequency*)

TF-IDF merupakan cara untuk menilai pada sebuah hubungan antara kata (*term*) dan sebuah dokumen. Metode tersebut merupakan gabungan dari dua konsep dalam menghitung nilai, yaitu frekuensi kemunculan suatu kata dalam suatu dokumen tertentu, serta frekuensi kebalikan dari dokumen yang mengandung katatersebut. Kepentingan kata pada sebuah dokumen dapat dilihat dari frekuensi kemunculan sebuah kata di dalam dokumen tersebut. Frekuensi dokumen dengan sebuah kata menjadikan seberapa umum kata tersebut. Sehingga nilai hubungan antara sebuah kata dan dokumen menjadi tinggi jika frekuensi kata di dalam dokumen juga tinggi, serta frekuensi keseluruhan dokumen yang mengandung kata rendah dalam dokumen.

Pada proses perhitungan TF-IDF banyak penelitian yang membatasi jumlah kata yang digunakan, menurut Pattnaik pembatasan tersebut dapat menggunakan *function min_df*, *max_df* atau *max_features. Function min_df berguna untuk menghapus kata yang jarang muncul pada dokumen, max_df berguna untuk menghapus kata yang terlalu sering muncul pada dokumen, sedangkan max_features berguna untuk mengambil kata teratas dengan frekuensi kemunculan terbesar* [16] [17].

Secara matematis nilai TF-IDF dapat didefinisikan menggunakan persamaan (II-2).

$$TF \cdot IDF_{std}(t) = tf_d^t \times \log \frac{N}{df^t} \quad (1)$$

ada persamaan di atas, tf_d^t merupakan jumlah istilah t muncul dalam dokumen d,

N merupakan jumlah total dokumen, df^t merupakan jumlah dokumen di mana istilah t terjadi.

C. Naïve Bayes

Naïve Bayes merupakan kumpulan dari algoritme klasifikasi yang menggunakan sifat-sifat teorema Bayes. *Naïve Bayes* merupakan algoritme yang sangat populerdalam analisis sentimen, klasifikasi *Naïve Bayes* merupakan klasifikasi probabilistik [12].

Keuntungan utama dari penggunaan Algoritme *Naïve Bayes* adalah hanya membutuhkan sedikit *datatraining* dalam penghitungan parameter untuk *prediction phase* [13]. Klasifikasi probabilitas pada *Naïve Bayes* dapat didefinisikan seperti persamaan berikut.

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)} \quad (2)$$

$P(H|X)$ merupakan peluang hipotesa H yang diperoleh dari kondisi X, X merupakan *data training* dengan *class* (label) yang telah diketahui, H merupakandata dengan *class* (label), $P(X|H)$ merupakan peluang yang diperoleh dari kondisihipotesa H, $P(H)$ merupakan peluang dari hipotesa X, serta $P(X)$ merupakan peluang yang diperoleh dari X yang telah dilakukan pengamatan.

Algoritme *Naïve Bayes* memiliki tiga tipe algoritme yang diterapkan pada *machine Learning*, berikut merupakan penjelasan dari masing-masing Algoritme *Naïve Bayes*.

a. Algoritme *Bernoulli Naïve Bayes*

Algoritme *Bernoulli Naïve Bayes* merupakan salah satu tipe Algoritma *Naïve Bayes* yang digunakan dalam penyelesaian masalah klasifikasi dokumen. Algoritma tersebut dapat membantu untuk melakukan klasifikasi dokumen ke dalam beberapa kategori, sesuai dengan kebutuhan peneliti [18]. Meskipun data yang digunakan adalah *Boolean,* akan tetapi terdapat penelitian yang

menggunakan Algoritme *Bernoulli* dengan ekstraksi fitur TF-IDF, dan ekstraksi tersebut dapat meningkatkan tingkat akurasi, *precision*, *recall*, dan F1-*score* [19].

Klasifikasi yang dilakukan pada Algoritme *Bernoulli Naïve Bayes* merupakan klasifikasi yang berfokus kepada parameter ada atau tidak ada kata pada suatu dokumen. Ketika kata yang termasuk dalam kategori tertentu ada pada sebuah dokumen, maka dokumen tersebut masuk kedalam kategori tersebut [20].

b. Algoritme *Gaussian Naïve Bayes*

Algoritme *Gaussian Naïve Bayes* merupakan algoritme yang memiliki pendistribusian yang cukup berbeda dengan Algoritme *Naïve Bayes* yang lain, yaitu berupa klasifikasi asumsi pada nilai kontinu yang memiliki keterkaitan dengan setiap fitur yang berisi nilai numerik [20]. Akan tetapi tipe data yang dapat digunakan Algoritme *Gaussian* juga menerapkan penyamaan tipe data seperti Algoritme *Bernoulli*, pada Algoritme *Gaussian* tipe data yang digunakan adalah numerik, apabila terdapat tipe data yang berbeda pada dokumen [18].

c. Algoritme *Multinomial Naïve Bayes*

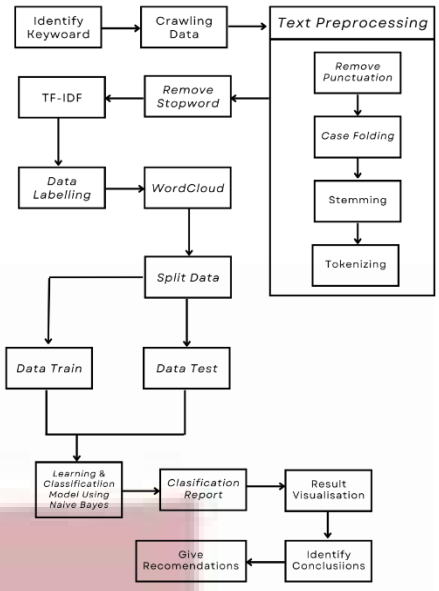
Algoritme *Multinomial Naïve Bayes* merupakan tipe yang mirip dengan Algoritme *Bernoulli Naïve Bayes*, namun klasifikasi pada Algoritme *Multinomial* berfokus pada frekuensi kata-kata yang muncul dalam dokumen yang digunakan peneliti [20]. Jika menggunakan Algoritme *Multinomial* tipe data yang digunakan akan disamakan juga seperti tipe algoritmeyang lain, tipe data yang digunakan adalah pecahan [18].

Cara kerja Algoritme *Multinomial* hampir mirip dengan Algoritme *Bernoulli* yaitu mengelompokkan dokumen berdasarkan kategori dokumen tersebut, akan tetapi pada Algoritme *Multinomial*, klasifikasi yang dilakukan berdasar frekuensi kata yang ada pada dokumen [20].

III. METODE

Pada tahapan awal, peneliti menentukan *keyword* yang akan digunakan untuk pengambilan data dari *Twitter*, yang kemudian data dari *Twitter* diambil dengan cara *Crawling Data*. Tahapan selanjutnya adalah *text processing* berupa *remove punctuation*, *case folding*, *stemming*, dan *tokenizing*. Selanjutnya dilakukan *labelling* data, dan penampilan data pada setiap sentime menggunakan *Word Cloud*.

Tahapan berikutnya adalah *data* di-*split* menjadi *data training* dan *data test*, dari kedua data tersebut dilakukan penilaian bobot menggunakan TF-IDF, selanjutnya *Learning* dan *Classification* menggunakan *Naïve Bayes*, dari klasifikasi yang telah dilakukan tersebut data diolah sehingga dapat menampilkan *Classification Report*. Tahapan terakhir pada penelitian adalah penampilan atau visualisasi hasil dari data analisis yang telah peneliti lakukan, kemudian peneliti akan menentukan kesimpulan serta memberikan saran terkait dengan penelitian yang telah dilakukan. Rangkaian tersebut dapat diilustrasikan pada Gambar III.



Gambar 1

A. Analisis Studi Kasus

Dalam penelitian ini kasus atau fenomena yang diteliti oleh peneliti berkaitan dengan review pelanggan Richeese Factory *pada platform* Twitter. Review *atau* Tweet *pelanggan* Richeese Factory *merupakan kumpulan komentar yang* ditinggalkan oleh pelanggan *Richeese Factory*. Pelanggan *Richeese Factory* dapat memposting komentar positif atau negatif pada platform *Twitter*. Oleh karena itu, diperlukan proses analisis sentimen berdasarkan *aspect-based* untuk mengetahui tren nilai dari kumpulan komentar positif atau negatif pelanggan *Richeese Factory* tersebut.

Aspect-based pada penelitian diambil dari teori *Marketing Mix*, yaitu produk (*Product*), harga (*Price*), tempat (*Place*), dan promosi (*Promotion*). Kalimat yang termasuk *Aspect-based Product* merupakan kalimat yang membahas mengenai produk *Richeese Factory*. Kalimat yang termasuk *Aspect-based Price* merupakan kalimat yang membahas mengenai harga dari produk-produk yang dijual atau diproduksi oleh *Richeese Factory*. Kalimat yang termasuk *Aspect-based Place* merupakan kalimat yang membahas mengenai tempat atau *platform* penjualan produk-produk *Richeese Factory* tersebut. Kalimat yang termasuk *Aspect-based Promotion* merupakan kalimat yang membahas mengenai promosi yang ada di *Richeese Factory*. Dari *Aspect-based* tersebut kalimat dikategorikan kembali menjadi lebih spesifik berdasarkan sentimennya.

Sentimen yang digunakan adalah positif (1), negatif (2), dan netral (0). Kalimat yang termasuk kalimat positif adalah kalimat yang bermakna positif, kalimat negatif adalah kalimat yang bermakna negatif, kalimat netral adalah kalimat yang tidak positif ataupun negatif. Sehingga kalimat yang tidak menjelaskan *Aspect- based* tersebut dapat dikategorikan netral.

B. Data Selection

Dataset yang digunakan dalam penelitian merupakan *tweet* atau *review* dari pelanggan *Richeese Factory* pada *platform Twitter* yang diperoleh menggunakan *Twitter* API atau *Tweepy*. *Tweepy* merupakan sebuah *library* pada *python* yang berguna untuk mengakses *tweet* secara langsung, proses

tersebut dinamakan *Crawling Data.* Proses *Crawling Data* menggunakan *library Tweepy* harus melalui proses *Authentication* melalui OAuth 2.0 dengan menggunakan *Bearer Token* yang tersedia di *Twitter* API *Developer* [21]. Setelah melakukan *Authentication*, dilakukan penentuan *keyword*, pada penelitian digunakan dua *keyword* sebagai berikut.

Tabel 1

No.	*Keyword*
1.	*Richeese Factory*
2.	*Richeese*

Setelah melakukan penentuan *keyword*, langkah selanjutnya yang dilakukan oleh peneliti adalah membuat aturan-aturan Tweet yang akan di Crawling, seperti batas maksimal tweet, jenis Tweet yang akan dicari, tanggal dari tweet yang akan dicari jika tidak mencari Tweet terbaru, sehingga lebih spesifik, dan aturan lain sebagainya. Dalam penelitian maksimal data yang akan di Crawling adalah 5.000 Tweet pada masing-masing *keyword* dan jenis Tweet yang dicari adalah Tweet terbaru. Hasil dari *Crawling Data* yang dilakukan pada 11 Januari 2023 adalah sebanyak 1.007 Tweet dan dilakukan penyimpanan ke sebuah file CSV. Berikut adalah dataset yang diperoleh peneliti pada tabel berikut.

Tabel 2

No	Sentiment Text
1	b'the only ricis i could trust is richeese factory level 1, aaak mau bgt ingin’
2	@FOOD_FESS Emg gak worth it richeese tuh, TAPI\n\nSaus keju &pink lavanya enak banget sih juaraaa \xf0\x9f\x98\xad
3	b'baru tau di banda aceh ada richeese factory'
4	b’@MartonMelody @FOOD_FESS aku ke richeese cuma pas promo kartupelajar aja'
5	b'cita cita pengen makan richeese factory tapi di pati not found, ngeod'

C. *Data Preprocessing*

Setelah dilakukan *Crawling Data*, langkah selanjutnya adalah *data preprocessing*. Pada *data preprocessing* merupakan proses yang penting dalam *data mining*, karena dengan proses tersebut maka dapat dijaga kualitas dari data, serta data dapat diolah dahulu sebelum ke proses analisis. Data yang tidak melalui *data preprocessing* cenderung tidak terkontrol dengan baik dan hasil dari analisis yang dilakukan akan kurang tepat, sehingga untuk mencapai tahap penemuan pengetahuan yang baik maka proses tersebut sangat diperlukan pada *Data Mining* [21].

a. *Remove Punctuation* dan *Case Folding*

Tahapan *Remove Punctuation* merupakan tahapan untuk menghapus *link*, emoji, serta tanda baca pada *dataset*. Pada tahapan tersebut digunakan *function re.sub()*, *function* tersebut bertujuan untuk mencari *string* atau *pattern* kalimat pada *dataset* dan menghapusnya atau mengganti dengan spesifik *string* [22].

Tahapan kedua adalah *Case Folding*, *Case Folding* merupakan tahapan untuk merubah semua huruf kapital pada *dataset* menjadi seragam ke huruf kecil. Pada tahapan tersebut digunakan *function .lower()*, *function* tersebut memiliki kinerja merubah huruf kapital menjadi huruf kecil [23].

Pada tabel di bawah merupakan hasil dari *Remove Punctuation* dan *Case Folding*.

Tabel 3

No	Sentiment Text	*remove_punctuation*
1	b'the only ricis i could trust is richeese factory level 1, aaak mau bgt ingin’	the only ricis could trust is richeese factory level mau banget ingin
2	@FOOD_FESS Emg gak worth it richeese tuh, TAPI\n\nSaus keju &pink lavanya enak banget sih juaraaa \xf0\x9f\x98\xad	emg gak worth it richeese tuh tapi saus keju amp pink lavanya enak banget sih juaraaa
3	b'baru tau di banda aceh ada richeese factory’	b'baru tau di banda aceh ada richeese factory'
4	b’@MartonMelody @FOOD_FESS aku ke richeese cuma pas promo kartu pelajar aja'	aku ke richeese cuma pas promokartu pelajar aja
5	b'cita cita pengen makan richeese factory tapi di pati not found, ngeod'	cita cita pengen makan richeesefactory tapi di pati not found ngeod

b. *Stemming*

Tahapan selanjutnya adalah Tahapan Stemming, Stemming *merupakan tahapan untuk menghilangkan* imbuhan yang ada di dalam kalimat pada *dataset* dan menjadikan kata pada tersebut menjadi kata dasar. Misalkan kata enaknya, setelah dilakukan *stemming* menjadi enak. Pada tahapan tersebut peneliti menggunakan library *Stemmer* Sastrawi, *Stemmer* Bahasa Indonesia karena *dataset* yang digunakan adalah Bahasa Indonesia [24].

Pada tabel di bawah merupakan hasil tahapan *Stemming* yang dilakukan oleh peneliti.

Tabel 4

No	*remove_punctuation*	*stemming_tweet*
1	the only ricis could trust is richeese factory level mau banget ingin	the only ricis could trust isricheese factory level mau banget ingin
2	emg gak worth it richeese tuh tapisaus keju amp pink lavanya enak banget sih juaraaa	emg gak worth it richeese tuhtapi saus keju amp pink lava enak banget sih juara
3	baru tau di banda aceh adaricheese factory	baru tau di banda aceh adaricheese factory
4	aku ke richeese cuma pas promokartu pelajar aja	aku ke richeese Cuma pas promo kartu pelajar aja
5	cita cita pengen makan richeese factory tapi di pati not found ngeod	cita cita ingin makan richeese factory tapi di pati not found ngeod

c. Tokenizing

Tahapan terakhir pada *Data Preprocessing* adalah *Tokenizing*, *Tokenizing* merupakan tahapan untuk merubah *string* pada *dataset* ke *text data* agar dapat

dengan mudah menafsirkan makna dari setiap kata pada *dataset*. *Function* yang digunakan oleh peneliti adalah *.split()*, *function* tersebut merupakan *tokenize* paling dasar untuk memecah *string* di setiap spasi [25].

Hasil dari *Tokenizing* yang dilakukan dapat dilihat pada tabel di bawah.

Tabel 5

No	*stemming_tweet*	preprocessed_tweet
1	the only ricis could trust isricheese factory level mau banget ingin	['the', 'only', 'ricis', 'could', 'trust', 'is', 'richeese', 'factory', 'level', 'mau', 'banget', 'ingin']
2	emg gak worth it richeese tuhtapi saus keju amp pink lava enak banget sih juara	['emg', 'gak', 'worth', 'it', 'richeese', 'tuh', 'tapi', 'saus', 'keju', 'amp', 'pink', 'lava', 'enak', 'banget', 'sih', 'juara']
3	baru tau di banda aceh adaricheese factory	['baru', 'tau', 'di', 'banda', 'aceh', 'ada','richeese', 'factory']
4	aku ke richeese Cuma pas promo kartu pelajar aja	['aku', 'ke', 'richeese', 'cuma', 'pas', 'promo', 'kartu', 'ajar', 'aja']
No	***stemming_tweet***	**preprocessed_tweet**
5	cita cita ingin makan richeese factory tapi di pati not found ngeod	['cita', 'cita', 'ingin', 'makan', 'richeese', 'factory', 'tapi', 'di', 'pati', 'not', 'found','ngeod']

D. *Data Labelling*

Tahap selanjutnya adalah *Data Labelling*, pada tahap tersebut peneliti melakukan *labelling* secara manual agar *dataset* yang dihasilkan lebih akurat karena peneliti membaca maksud dari kalimat yang ada dalam *dataset* tersebut secara langsung [26]. *Class* atau *Aspect* yang digunakan pada penelitian adalah *Product*, *Price*, *Place*, dan *Promotion*. Kalimat yang termasuk *Product* merupakan kalimat yang membahas mengenai produk dari *Richeese Factory*, *Price* merupakan kalimat yang membahas mengenai harga dari produk *Richeese Factory*, *Place* merupakan kalimat yang membahas mengenai tempat atau *platform* yang digunakan dalam penjualan produk *Richeese Factory*, dan *Promotion* merupakan kalimat yang membahas mengenai promosi pada produk *Richeese Factory*.

Label Sentiment yang digunakan adalah *Positive* (1), *Negative* (2), dan *Neutral* serta *None* (0), di mana kalimat yang ditandai positif adalah kalimat yang mempunyai makna positif, kalimat yang mempunyai makna negatif ditandai dengan negatif, sedangkan kalimat yang tidak menandai keduanya atau tidak termasuk dalam *aspect* tertentu maka ditandai dengan netral.

Pada tabel di di bawah merupakan hasil *labelling* manual yang dilakukan oleh peneliti.

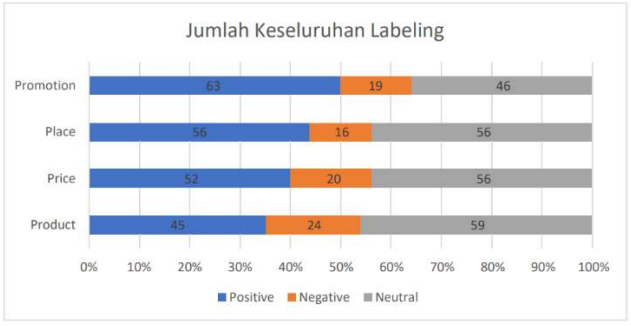
Tabel 6

No	preprocessed_tweet	*Product*	*Price*	*Place*	*Promotion*
1	['the', 'only', 'ricis', 'could', 'trust', 'is', 'richeese', 'factory', 'level', 'mau', 'banget', 'ingin']	1	0	1	0
2	['emg', 'gak', 'worth', 'it', 'richeese', 'tuh', 'tapi', 'saus', 'keju', 'amp', 'pink', 'lava', 'enak', 'banget', 'sih', 'juara']	1	2	0	0
3	['baru', 'tau', 'di', 'banda', 'aceh', 'ada','richeese', 'factory']	0	0	1	0
4	['aku', 'ke', 'richeese', 'cuma', 'pas', 'promo', 'kartu', 'ajar', 'aja']	0	1	0	1
5	['cita', 'cita', 'ingin', 'makan', 'richeese', 'factory', 'tapi', 'di', 'pati', 'not', 'found','ngeod']	1	0	2	0

Pada tabel di bawah menunjukkan jumlah keseluruhan dari *labelling* manualyang dilakukan oleh peneliti.

Tabel 7

Category	*Product*	*Price*	*Place*	*Promotion*
Positive	45	52	56	63
Negative	24	20	16	19
Neutral	59	56	56	46



Gambar 2

E. *Word Cloud*

Langkah selanjutnya adalah *word cloud*, *word cloud* adalah salah satu metode dari *Text Mining* yang digunakan untuk menampilkan kata-kata yang seringdigunakan, *Word Cloud* biasa dimanfaatkan dalam menentukan istilah *trend* berdasarkan frekuensi kata yang sering muncul dari pengguna. *Word cloud* dapat memunculkan kata populer dalam *text mining* dengan bentuk yang menarik namun tetap informatif [27]. Pada penelitian peneliti menggunakan *word cloud* untuk menampilkan kata-kata yang paling muncul di dalam teks baik kata-kata positif, negatif, maupun netral berdasarkan masing-masing *aspect*.

F. *Data Mining*

Proses selanjutnya dalam penelitian adalah *Data Mining*, *Data Mining* merupakanproses ekstraksi atau pengambilan informasi yang berguna dari data yang besar, beragam, dan kompleks. Proses *Data Mining* bertujuan untuk menemukan

pola atau hubungan yang tersembunyi di dalam *dataset*, sehingga dapat digunakan dalam membuat keputusan yang lebih baik [28]. Pada penelitian proses *Data Mining* dimulai dengan *Split Data* kalimat dan *aspect*, kemudian *Split* ke dalam *Train Data* dan *Test Data*, dilanjutkan dengan perhitungan TF-IDF, lalu *Learning and Classification* menggunakan *Naïve Bayes*, dan terakhir menampilkan *Classification Report* dari masing-masing *aspect*.

G. *Split Data*

Tahap pertama dalam *Data Mining* pada penelitian adalah *split data*, *split data* dilakukan untuk memisahkan kalimat atau *source* dan *aspect.* Tahap *Split Data* dilakukan untuk memudahkan peneliti dalam mengolah *dataset* yang ada. *Split Data* yang dilakukan adalah *Source* ditandai dengan X dan *aspect Product*, *Price*,*Place*, serta *Promotion* ditandai dengan y sebagai variabel pada penelitian.

H. *Data Train* dan *Data Test*

Tahap kedua dalam *Data Mining* pada penelitian adalah *Train Data* dan *Test Data*. Setelah *dataset* dipisahkan antara kalimat dan *aspect*, selanjutnya adalah pemisahan antara *Train Data* dan *Test Data*. *Train Data* merupakan data yang digunakan dalam melatih sistem mengenali pola yang akan dicari, sedangkan *TestData* merupakan data yang digunakan dalam menguji hasil latihan yang dilakukan [29].

Split Train Data dan *Test Data* yang dilakukan oleh peneliti dibagi ke dalam masing-masing *Source* (X) dan *aspect* (y), begitupun juga dilakukan pembagian yang sama untuk *Test Data*. Berikut pembagian yang dilakukan oleh peneliti, X_train sebagai sumber data yang akan dilatih (*train*), X_*test* sebagai sumber datayang akan diuji, y_train sebagai variabel yang akan dilatih, y_*test* sebagai variabel yang akan diuji. Dalam proses pengujian digunakan *test*_size sebesar 20% dan random_state sebesar 42.

I. TF-IDF

TF-IDF atau *Term Frequency-Inverse Document Frequency* merupakan tahap selanjutnya pada proses *Data Mining*. TF-IDF sendiri digunakan untuk melakukan pembobotan pada sebuah teks yang akan dilakukan analisis pada tahapselanjutnya [21]. Pada *dataset* terdapat nilai TF yang lebih tinggi dibandingkan dengan nilai TF yang lain, sehingga dilakukan pembatasan nilai *features* menggunakan *max features*. Dalam menghitung TF-IDF diperlukan persamaan sebagai berikut.

$$TF \cdot IDF_{std}(t) = tf_d^t \times \log \frac{N}{df^t} \quad (3)$$

Pada persamaan (IV-1), tf_d^t merupakan jumlah istilah t muncul dalam dokumen d, N merupakan jumlah total dokumen, df^t merupakan jumlah dokumen di mana istilah t terjadi. Pada penelitian tf_d^t adalah kata yang terdapat pada Sentiment Text, df^tadalah jumlah dokumen jika kata yang terpilih muncul.

Peneliti akan menghitung simulasi TF-IDF pada kalimat berikut.

Tabel 8

No	*Type*	*Sentiment Text*	*Product*	*Price*	*Place*	*Promotion*
1	*Data Train*	['emg', 'gak', 'worth', 'it', 'richeese', 'tuh', 'tapi', 'saus', 'keju', 'amp', 'pink', 'lava', 'enak', 'banget', 'sih', 'juara']	1	2	0	0
2		['baru', 'tau', 'di','banda', 'aceh', 'ada', 'richeese', 'factory']	0	0	1	0
3	*Data Test*	['ayam', 'richeese', 'factory', 'dan', 'saus', 'keju', 'enak', 'harga', 'worth', 'it', 'ada','di', 'grabfood']	?	?	?	?

Data disimulasikan menggunakan aplikasi *Ms. Excel* dengan rumus TF-IDF secara manual. Simulasi dilakukan dengan menggunakan 2 *Data Train* dan 1 *Data Test* yang belum diketahui label sentimennya pada setiap *aspect*. Berikut merupakan hasil dari simulasi TF-IDF

Tabel 9

Term	**Tf**			**df**	**N/df**
	d1	***d2***	***d3***		
emg	1			1	3
gak	1			1	3
worth	1		1	2	1,5
it	1		1	2	1,5
Term	**Tf**			**df**	**N/df**
	d1	***d2***	***d3***		
richeese	1	1	1	3	1
tuh	1			1	3
tapi	1			1	3
saus	1		1	2	1,5
keju	1		1	2	1,5
amp	1			1	3
pink	1			1	3
lava	1			1	3
enak	1		1	2	1,5
banget	1			1	3
sih	1			1	3
juara	1			1	3
baru		1		1	3
tau		1		1	3
di		1	1	2	1,5
banda		1		1	3
aceh		1		1	3
ada		1	1	2	1,5
factory		1	1	2	1,5
ayam			1	1	3
dan			1	1	3
harga			1	1	3
grabfood			1	1	3

Tabel 10

Term	TF		
	d1	**d2**	**d3**
emg	0,477121	0	0

gak	0,477121	0	0
worth	0,176091	0	0,176091
it	0,176091	0	0,176091
richeese	0	0	0
tuh	0,477121	0	0
tapi	0,477121	0	0
saus	0,176091	0	0,176091
keju	0,176091	0	0,176091
amp	0,477121	0	0
pink	0,477121	0	0
lava	0,477121	0	0
enak	0,176091	0	0,176091
banget	0,477121	0	0
sih	0,477121	0	0
juara	0,477121	0	0
baru	0	0,477121	0
tau	0	0,477121	0
di	0	0,176091	0,176091
banda	0	0,477121	0
aceh	0	0,477121	0
ada	0	0,176091	0,176091
factory	0	0,176091	0,176091
Term	TF		
	d1	**d2**	**d3**
ayam	0	0	0,477121
dan	0	0	0,477121
harga	0	0	0,477121
grabfood	0	0	0,477121
Bobot	5,651669	2,436759	3,317215

Simulasi yang dilakukan oleh peneliti pada tabel di atas menggunakan 2 *Data Train* dan 1 *Data Test*, data tersebut telah melewati proses *Data Preprocessing*, sehingga kata yang dihasilkan telah bersih. Hasil simulasi manual TF-IDF pada dokumen 1 memiliki nilai bobot yang lebih tinggi dari dokumen 2 dan 3, karena dokumen 1 mengandung lebih banyak kata daripada dokumen yang lain. Dari hasil simulasi tersebut dapat disimpulkan tingkat relevan tertinggi yang dihasilkandari perhitungan manual TF-IDF adalah dokumen 1 dengan nilai 5,651669 dan dokumen 2 dengan nilai relevan paling rendah sebesar 2,436759.

J. Naïve Bayes

Tahap selanjutnya adalah *Learning and Classification* menggunakan Algoritme *Naïve Bayes*, *Naïve Bayes* merupakan salah satu algoritme klasifikasi untuk menggabungkan kombinasi nilai frekuensi basis data berdasarkan teorema *Bayes* dalam menghitung semua probabilitas. Algoritme tersebut sering digunakan karena kecepatan dan keakuratan prediksi yang cukup tinggi. Algoritme *Naïve Bayes* sendiri terdapat tiga metode yang berbeda yaitu *Bernoulli*, *Gaussian*, dan *Multinomial Naïve Bayes.* Algoritme tersebut sering digunakan karena kecepatan dan keakuratan prediksi yang cukup tinggi. Algoritme *Naïve Bayes* sendiri terdapat tiga metode yang berbeda yaitu *Bernoulli*, *Gaussian*, dan *Multinomial Naïve Bayes.* Sehingga peneliti ingin membandingkan keakuratan dari ketiga metode tersebut.

Klasifikasi probabilitas pada *Naïve Bayes* dapat didefinisikan seperti persamaan berikut.

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)} \quad (3)$$

$P(H|X)$ pada persamaan (3) merupakan probabilitas kalimat H yang terjadi dengan bukti bahwa kalimat X telah terjadi atau disebut dengan *probabilitas superior*, kemudian $P(X|H)$ merupakan probabilitas X yang terjadi dengan bukti bahwa H telah terjadi. Pada penelitian H yang dimaksud adalah kata pada kalimat *sentiment* dan X adalah *Label Sentiment*. $P(H)$ merupakan peluang terjadinya H atau kata pada kalimat *sentiment* dan $P(X)$ merupakan peluang terjadinya X atau *Label Sentiment*.

Berikut pada tabel di bawah merupakan kalimat yang digunakan peneliti untuk melakukan simulasi *Naïve Bayes*.

Tabel 11

No	***Type***	***Sentiment Text***	***Product***	***Price***	***Place***	***Promotion***
1	***Data Train***	['emg', 'gak', 'worth', 'it','richeese', 'tuh', 'tapi', 'saus', 'keju', 'amp', 'pink', 'lava', 'enak', 'banget', 'sih', 'juara']	1	2	0	0
2		['baru', 'tau', 'di', 'banda', 'aceh', 'ada', 'richeese', 'factory']	0	0	1	0

Peneliti mencari probabilitas *Naïve Bayes* dari setiap kata pada tabel di atas menggunakan persamaan (3). Berikut pada tabel di bawah merupakan hasil dari simulasi yang dilakukan.

Tabel 12

Kata	**1**	**2**	**0**	**Total (X)**	**P(X)**	**P(X|H) Positive**	**P(X|H) Negative**	**P(X|H) Neutral**
emg	1	1	2	4	0,04167	0,04167	0,0625	0,03571
gak	1	1	2	4	0,04167	0,04167	0,0625	0,03571
worth	1	1	2	4	0,04167	0,04167	0,0625	0,03571
it	1	1	2	4	0,04167	0,04167	0,0625	0,03571
richeese	2	1	5	8	0,08333	0,08333	0,0625	0,08928
tuh	1	1	2	4	0,04167	0,04167	0,0625	0,03571
tapi	1	1	2	4	0,04167	0,04167	0,0625	0,03571
saus	1	1	2	4	0,04167	0,04167	0,0625	0,03571
keju	1	1	2	4	0,04167	0,04167	0,0625	0,03571
amp	1	1	2	4	0,04167	0,04167	0,0625	0,03571
pink	1	1	2	4	0,04167	0,04167	0,0625	0,03571
lava	1	1	2	4	0,04167	0,04167	0,0625	0,03571
enak	1	1	2	4	0,04167	0,04167	0,0625	0,03571
banget	1	1	2	4	0,04167	0,04167	0,0625	0,03571
sih	1	1	2	4	0,04167	0,04167	0,0625	0,03571

juara	1	1	2	4	0,04167	0,04167	0,0625	0,03571
baru	1	0	3	4	0,04167	0,04167	0	0,05357
tau	1	0	3	4	0,04167	0,04167	0	0,05357
di	1	0	3	4	0,04167	0,04167	0	0,05357
banda	1	0	3	4	0,04167	0,04167	0	0,05357
aceh	1	0	3	4	0,04167	0,04167	0	0,05357
ada	1	0	3	4	0,04167	0,04167	0	0,05357
factory	1	0	3	4	0,04167	0,04167	0	0,05357
Total P(H)	24	16	56		**P(H)**	0,25	0,167	0,583

Tabel 13

Kata	**P(H|X)**		
	Positive	**Negative**	**Neutral**
emg	0,24	0,24	0,48
gak	0,24	0,24	0,48
worth	0,24	0,24	0,48
it	0,24	0,24	0,48
richeese	0,24	0,12	0,60
tuh	0,24	0,24	0,48
tapi	0,24	0,24	0,48
saus	0,24	0,24	0,48
keju	0,24	0,24	0,48
Kata	**P(H|X)**		
	Positive	**Negative**	**Neutral**
amp	0,24	0,24	0,48
pink	0,24	0,24	0,48
lava	0,24	0,24	0,48
enak	0,24	0,24	0,48
banget	0,24	0,24	0,48
sih	0,24	0,24	0,48
juara	0,24	0,24	0,48
baru	0,24	0	0,72
tau	0,24	0	0,72
di	0,24	0	0,72
banda	0,24	0	0,72
aceh	0,24	0	0,72
ada	0,24	0	0,72
factory	0,24	0	0,72

Pada Tabel 12 dan 13 merupakan hasil simulasi Algoritme *Naïve Bayes* secara manual menggunakan aplikasi *Ms. Excel* sesuai dengan persamaan (3). Dan dapat dilihat bahwa perhitungan terbanyak ada pada *Neutral*, karena pada 2 kalimat yang memiliki label sentimen terbanyak pada sentimen neutral. Dan Label sentimen paling sedikit adalah *negative*, karena hanya hanya ada 1 aspect yang memiliki label *sentiment negative*.

IV. HASIL DAN PEMBAHASAN

A. *Dataset*

Pada penelitian *dataset* yang digunakan adalah data hasil dari proses *Crawling Data* menggunakan *library Tweepy*. *Tweet* yang dikumpulkan merupakan ulasan pelanggan *Richeese Factory*. Data tersebut kemudian disimpan dalam bentuk FileCSV agar memudahkan untuk mengakses kembali hasil dari *Crawling Data* tersebut. Pada tabel berikut merupakan jumlah *Crawling Data* yang dilakukan oleh peneliti.

Tabel 14

Keyword	**Jumlah**
Richeese Factory	86
Richeese	941
Total	1.027

Berdasarkan tabel di atas dapat dilihat bahwa total *Crawling Data* yang akan digunakan dalam penelitian adalah 1.027 *Tweet*, dengan 86 *Tweet* dengan *keyword Richeese Factory*, dan 941 *Tweet* dengan *keyword Richeese*. Kemudian peneliti melakukan proses *Data Processing* dan melakukan *Labelling* secara manual dengan *Aspect Product*, *Price*, *Place*, dan *Promotion,* serta *label sentiment Positive*, *Negative*, dan *Neutral*. Pada tabel berikut merupakan hasil dari proses *Data Processing* dan *Labelling manual*.

Tabel 15

	Positive	***Negative***	***Neutral***	**Total**
Product	45	24	59	128
Price	52	20	56	128
	Positive	***Negative***	***Neutral***	**Total**
Place	56	16	56	128
Promotion	63	19	46	128
Total	216	79	217	512

Berdasarkan di atas, dapat dilihat bahwa total *dataset* yang akan diproses ke tahap selanjutnya adalah 512 *Tweet*, dengan jumlah *Label Positive* sebanyak 216, *Label Negative* sebanyak 79, dan *Label Neutral* sebanyak 217.

Pada penelitian implementasi Algoritme yang dilakukan oleh peneliti dibagimenjadi dua data yaitu *Data Train* dan *Data Test*. Penelitian dilakukan untuk melakukan perbandingan antara tiga Algoritme *Naïve Bayes* yaitu *BernoulliGaussian*, dan *Multinomial Naïve Bayes* bertujuan untuk melihat tingkat akurasi yang paling tinggi pada masing-masing algoritme. Peneliti juga melakukan perbandingan pada perbedaan jumlah *Max Features* pada tahap TF-IDF untuk melihat keterkaitan tingkat akurasi dengan perbedaan jumlah tersebut.

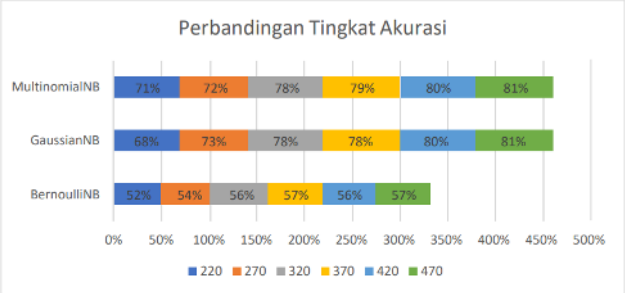
Pada penelitian juga dilakukan perbandingan antara tiga Algoritme *Naïve Bayes* dengan perbedaan ukuran *Data Train* dan *Data Test*. Penelitian bertujuan untuk membandingkan tingkat akurasi berdasarkan perbedaan ukuran tersebut.

B. Analisis Implementasi Algoritme *Naïve Bayes* dan Perbedaan *Max Features* TF-IDF

Implementasi Algoritme *Naïve Bayes* dan perbedaan *Max Features* TF-IDF pada penelitian merupakan implementasi yang bertujuan melihat perbandingan tingkatakurasi pada tiga Algoritme *Naïve Bayes* dan juga perbandingan pada jumlah *Max Features* TF-IDF yang berbeda.

Tabel 16

Max Features	Tingkat Akurasi *BernoulliNB*	Tingkat Akurasi *GaussianNB*	Tingkat Akurasi *MultinomialNB*
220	52%	68%	71%
270	54%	73%	72%
320	56%	78%	78%
370	57%	78%	79%



Gambar 3

Berdasarkan Implementasi ketiga Algoritme *Naïve Bayes* yang telah dilakukan oleh peneliti dapat dilihat perbedaan hasil akurasi pada masing-masing algoritmedan perbedaan akurasi dari *Max Features* TF-IDF. Berikut rangkuman perbedaan hasil akurasi pada penelitian.

Pada Gambar V-1 menunjukkan hasil perbandingan dari ketiga implementasi Algoritme *Naïve Bayes* dan *Max Features* TF-IDF yang telah dilakukan oleh peneliti. Pada penelitian peneliti menggunakan *Pipeline* pada masing-masing Algoritme *Naïve Bayes* untuk menghilangkan kesalahan serta menetralisir hambatan pada proses implementasi. Peneliti juga menggunakan *function Multi Output Classifier* pada masing-masing Algoritme *Naïve Bayes* untuk melakukan prediksi *Multi Label* (*Positive*, *Negative*, dan *Neutral*) pada *Multi Class* (*Aspect Product*, *Price*, *Place*, dan *Promotion*).

Hasil tingkat akurasi dari tiga implementasi Algoritme *Naïve Bayes* menunjukkan perbedaan nilai tingkat akurasi. Implementasi Algoritme *Bernoulli Naïve Bayes* memiliki tingkat akurasi paling rendah daripada Algoritme *Naïve Bayes* yang lain. Algoritme *Gaussian* dan *Multinomial Naïve Bayes* memiliki nilai akurasi yang hampir serupa pada semua percobaan penelitian.

Hasil tingkat akurasi dari perbedaan jumlah *Max Features* TF-IDF menunjukkan bahwa semakin tinggi jumlah *Max Features* yang digunakan maka tingkat akurasi akan meningkat. Perbedaan tingkatan tersebut dapat disimpulkan oleh peneliti terjadi karena semakin banyaknya kata yang digunakan untuk dihitung pada tahap TF-IDF maka akan banyak sampel pada *dataset* yang digunakan sehingga tingkat akurasi yang dihasilkan akan meningkat.

C. Analisis Implementasi Algoritme *Naïve Bayes* dan Perbedaan *Test Size*

Analisis Implementasi Algoritme *Naïve Bayes* dan perbedaan *Test Size* merupakan implementasi yang bertujuan melihat perbedaan hasil akurasi jika ukuran Data Train dan Data *Test* berbeda. Pada implementasi data dibagi menjadi dua bagian yaitu Data Train dan Data *Test* dengan nilai perbandingan yang digunakan adalah 60:40, 80:20, dan 80:20 dan menggunakan *Max Features* TF- IDF sebesar 900. Jumlah data dari setiap perbandingan dapat dilihat pada tabel berikut.

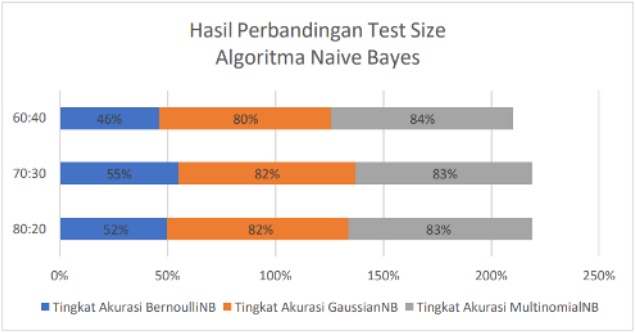
Tabel 17

Perbandingan	*DataTrain*	*Data Test*	Total
80:20	102	26	128
70:30	89	39	128
60:40	76	52	128

Perbandingan perbedaan *Test Size* menggunakan tiga Algoritme *Naïve Bayes* memiliki tujuan untuk melakukan pencarian tingkat akurasi pada masing-masing algoritme dan perbandingan *Test Size* tersebut. Adapun hasil tingkat akurasi dari perbandingan menggunakan tiga Algoritme *Naïve Bayes* dengan perbedaan perbandingan *Test Size* adalah sebagai berikut.

Tabel 18

Perbandingan	Tingkat Akurasi *BernoulliNB*	Tingkat Akurasi *GaussianNB*	Tingkat Akurasi *MultinomialNB*
80:20	52%	82%	83%
70:30	55%	82%	83%
60:40	46%	80%	84%



Gambar 4

Berdasarkan grafik pada Gambar V-2 menunjukkan tingkat akurasi tertinggi pada perbandingan *Test Size* Algoritme *Naïve Bayes* adalah 70:30 dan perbandingan yang memiliki tingkat akurasi paling rendah adalah 60:40. Pada Algoritme *Bernoulli Naïve Bayes* perbedaan perbandingan *Test Size* sangat berpengaruh kepada hasil akurasi implementasi, sedangkan Algoritme *Gaussian* dan *Multinomial Naïve Bayes* tidak terlalu berpengaruh. Tidak berpengaruhnya algoritme tersebut dengan perbandingan *Test Size* karena hasil akurasi pada grafik menunjukkan bahwa hasil akurasi kedua algoritme hanya berselisih 1% hingga 2%.

D. Hasil Analisis Implementasi Algoritme *Naïve Bayes*

Hasil dari implementasi Algoritme *Naïve Bayes* yang telah dilakukan pada poin B dan C perbedaan *Max Features* dan *Test Size* memiliki peran penting pada hasil akurasi penelitian. Pada kedua implementasi Algoritme *Naïve Bayes*, implementasi algoritme dilakukan pada *Pipeline* yang

berfungsi menghubungkanAlgoritme *Naïve Bayes* dengan *function Multi Output Classification*, serta digunakan *function Min Max Scaller* yang bertujuan untuk menormalisasi jumlahdata pada *dataset*.

Hasil tingkat akurasi dari kedua implementasi Algoritme *Naïve Bayes* yang telahdilakukan peneliti, terdapat perbedaan nilai tingkat akurasi pada ketiga algoritmetersebut. Algoritme *Bernoulli Naïve Bayes* selalu memiliki hasil akurasi yang rendah pada kedua implementasi tersebut. Algoritme *Gaussian* dan *Multinomial Naïve Bayes* memiliki selisih hasil akurasi sekitar 0% hingga 4% sehingga, akan tetapi Sebagian besar akurasi tertinggi pada Algoritme *Multinomial.*

Berdasarkan tingkat akurasi dari implementasi Algoritme *Naïve Bayes* pada poin jika nilai *Max Features* pada penelitian semakin tinggi akan menghasilkan nilai akurasi yang semakin tinggi pula, karena semakin banyaknya kata yang digunakan untuk dihitung pada tahap TF-IDF maka akan banyak sampel pada *dataset* yang digunakan sehingga tingkat akurasi yang dihasilkan akan meningkat.

Berdasarkan tingkat akurasi dari implementasi Algoritme *Naïve Bayes* pada poin jika perbandingan pada *Data Train* yang digunakan semakin tinggi akan menghasilkan nilai akurasi yang semakin tinggi pula, karena semakin banyak jumlah data yang dilatih akan menghasilkan data yang lebih mengerti mengenai keseluruhan data.

E. Evaluasi Model

Pada hasil akurasi yang telah diperoleh pada Algoritme *Multinomial Naïve Bayes*cukup tinggi dibandingkan dengan Algoritme *Naïve Bayes* lainnya, sehingga peneliti menggunakan hasil akurasi dari algoritme tersebut pada *Max Features* TF- IDF 900 dan perbandingan *Test Size* 70:30 untuk dilakukan evaluasi model menggunakan *confusion matrix*. Berikut pada tabel di bawah merupakan hasil pengujian *Confusion Matrix* yang telah dilakukan peneliti pada *Aspect Product, Price, Place* dan *Promotion*.

Tabel 19

	Predicted Positive	***Predicted Negative***	***Predicted Neutral***
Actual Positive	16	5	0
Actual Negative	4	7	3
Actual Neutral	0	0	4

Tabel 20

	Predicted Positive	***Predicted Negative***	***Predicted Neutral***
Actual Positive	9	5	1
Actual Negative	3	13	1
Actual Neutral	2	2	3

Tabel 21

	Predicted Positive	***Predicted Negative***	***Predicted Neutral***
Actual Positive	7	4	1
Actual Negative	3	19	1
Actual Neutral	0	2	2

Tabel 22

	Predicted Positive	***Predicted Negative***	***Predicted Neutral***
Actual Positive	8	2	2
Actual Negative	0	21	1
Actual Neutral	4	1	0

Berdasarkan Tabel 19 hingga 22 menunjukkan *Confusion Matrix* pada seluruh Aspect, pada Tabel 19 dapat dilihat bahwa pada *Aspect Product* jumlah *Tweet* bernilai *Positive* yang benar diprediksi oleh sistem adalah sebanyak 16, sedangkan*Tweet* bernilai *Negative* yang benar diprediksi oleh sistem adalah sebanyak 7, dan4 *Tweet Neutral* yang benar diprediksi oleh sistem adalah sebanyak 4.

Berdasarkan hasil dari data yang telah didapat pada *confusion matrix* dapat dihitung tingkat akurasinya secara manual menggunakan rumus sebagai berikut.

$$Accuracy(AC) = \frac{TP+TN+TNeu}{\text{TP+FP+TNeg +TNeu+FNeg+FNeu}} \quad (4)$$
$$= \frac{16+7+4}{16+5+7+4+7+0} = \frac{27}{39} = 0{,}69$$

Hasil akurasi yang dihitung secara manual adalah sebesar 0,69 atau jika dikonversikan ke dalam bentuk persentase menjadi 69%, hasil tersebut sama dengan hasil implementasi Algoritme *Naïve Bayes* yang telah dilakukan menghitung presisi secara manual menggunakan rumus sebagai berikut.

$$Precision(Positive) = \frac{TP}{\text{TP+FP}} \quad (5)$$
$$= \frac{16}{16+4} = \frac{16}{20} = 0{,}8$$

$$Precision(Negative) = \frac{TP}{\text{TP+FP}} \quad (6)$$
$$= \frac{7}{7+5} = \frac{7}{12} = 0{,}58$$

$$Precision(Neutral) = \frac{TP}{\text{TP+FP}} \quad (7)$$
$$= \frac{14}{4+3} = \frac{4}{7} = 0{,}57$$

Hasil *Precision* pada *Aspect Product* yang dihitung secara manual oleh peneliti menghasilkan 0,8 untuk *Precision Positive*, 0,58 *Precision Negative*, dan 0,57 *Precision Neutral*, yang masing-masing jika dikonversikan ke dalam bentukpersentase adalah sebesar 80%, 58%, dan 57%, hasil tersebut sama dengan hasil implementasi Algoritme *Naïve Bayes* yang telah dilakukan oleh peneliti sebelumnya. Langkah selanjutnya adalah menghitung *Recall* dengan rumus sebagai berikut.

$$Recall(Positive) = \frac{TP}{\text{TP+FP}} \quad (8)$$
$$= \frac{16}{16+5} = \frac{16}{21} = 0{,}76$$

$$Recall(Negative) = \frac{TP}{\text{TP+FP}} \quad (9)$$
$$= \frac{7}{7+7} = \frac{7}{14} = 0{,}5$$

$$Recall(Neutral) = \frac{TP}{\text{TP+FP}} \quad (10)$$

$$= \frac{4}{4+0} = \frac{4}{4} = 1$$

Hasil *Recall* pada *Aspect Product* yang dihitung secara manual oleh peneliti menghasilkan 0,76 untuk *Precision Positive*, 0,5 *Precision Negative*, dan 1 *Precision Neutral*, yang masing-masing jika dikonversikan ke dalam bentuk persentase adalah sebesar 76%, 50%, dan 100%, hasil tersebut sama dengan hasil implementasi Algoritme *Naïve Bayes* yang telah dilakukan oleh peneliti sebelumnya. Langkah selanjutnya adalah menghitung F1-*Score* dengan menggunakan rumus sebagai berikut.

$$F1 - Score(Positive) = \frac{(\text{precision} \times \text{recall})}{(\text{precision} + \text{recall})} \quad (11)$$
$$= 2 \times \frac{0{,}8 \times 0{,}76}{0{,}8 + 0{,}76} = 2 \times \frac{0{,}61}{1{,}56} = 0{,}78$$

$$F1 - Score(Negative) = \frac{(\text{precision} \times \text{recall})}{(\text{precision} + \text{recall})} \quad (12)$$
$$= 2 \times \frac{0{,}58 \times 0{,}5}{0{,}58 + 0{,}5} = 2 \times \frac{0{,}29}{1{,}08} = 0{,}54$$

$$F1 - Score(Neutral) = \frac{(\text{precision} \times \text{recall})}{(\text{precision} + \text{recall})} \quad (13)$$
$$= 2 \times \frac{0{,}57 \times 1}{0{,}57 + 1} = 2 \times \frac{0{,}57}{1{,}57} = 0{,}73$$

Hasil F1-*Score* pada *Aspect Product* yang dihitung secara manual oleh peneliti menghasilkan 0,78 untuk *Precision Positive*, 0,54 *Precision Negative*, dan 0,73 *Precision Neutral*, yang masing-masing jika dikonversikan ke dalam bentukpersentase adalah sebesar 78%, 54%, dan 73%, hasil tersebut sama dengan hasil implementasi Algoritme *Naïve Bayes* yang telah dilakukan oleh peneliti sebelumnya. Langkah selanjutnya adalah menghitung *True Positive, False Positive*, *True Negative* dan *False Negative* pada masing-masing *Aspect* dan *Label*, berikut merupakan hasil dari perhitungan manual menggunakan *Ms. Excel*.

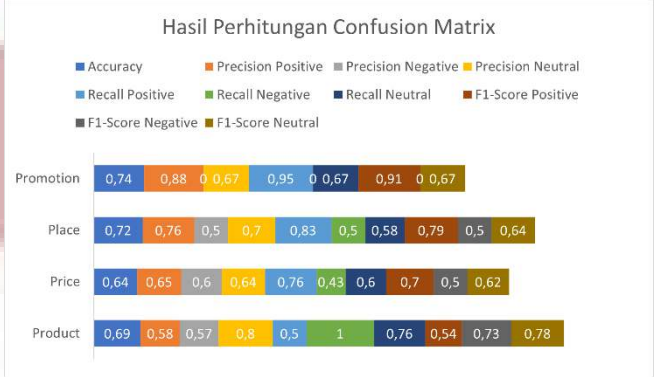
Tabel 23

		Product	*Price*	*Place*	*Promotion*
Positive	***TP***	7	13	19	21
	FP	5	7	6	3
	TN	7	4	4	1
	FN	7	4	4	1
Negative	***TP***	4	3	2	0
	FP	3	2	2	3
	TN	0	4	2	5
	FN	0	4	2	5
Neutral	***TP***	16	9	7	8
	FP	4	5	3	4
	TN	5	6	5	4
	FN	5	6	5	4

Setelah mendapat *True Positive, False Positive*, dan *False Positive* pada masing-masing *Aspect* dan *Label*, langkah selanjutnya adalah menghitung *Accuracy Score*, *Precision*, *Recall*, dan F1-*Score* pada masing-masing *Aspect* dan *Label*. Berikut merupakan hasil perhitungan manual yang dilakukan oleh peneliti menggunakan *Ms. Excel*.

Tabel 24

		Product	*Price*	*Place*	*Promotion*
Accuracy		0,69	0,64	0,72	0,74
Precision	***1***	0,58	0,65	0,76	0,88
	2	0,57	0,60	0,50	0,00
	0	0,80	0,64	0,70	0,67
Recall	***1***	0,50	0,76	0,83	0,95
	2	1,00	0,43	0,50	0,00
	0	0,76	0,60	0,58	0,67
F1-Score	***1***	0,54	0,70	0,79	0,91
	2	0,73	0,50	0,50	0,00
	0	0,78	0,62	0,64	0,67



Gambar 5

Hasil *Accuracy Score*, *Precision*, *Recall*, dan F1-*Score* pada seluruh *Aspect* dan *Label* yang dihitung secara manual oleh peneliti menghasilkan nilai yang sama dengan hasil implementasi Algoritme *Naïve Bayes* yang telah dilakukan oleh peneliti sebelumnya. Pada Tabel 24 dapat dilihat bahwa dalam prediksi yang dihasilkan pada *Aspect Product* memiliki nilai yang prediksi lebih tinggi dari pada *Aspect* yang lain. Sehingga *Aspect Product* merupakan *aspect* yang paling utama dalam menentukan peningkatan pelanggan *Richeese Factory* karena kata pada *aspect* tersebut sering muncul pada *dataset*.

F. Visualisasi *Word Cloud*

Visualisasi yang dilakukan dalam penelitian adalah visualisasi menggunakan *Word Cloud*, *Word Cloud* adalah tahap akhir dari Proses *Data Processing* yang dilakukan karena bertujuan untuk menampilkan kata-kata yang paling sering muncul dari masing-masing *Aspect* dan *Label Sentiment* pada *dataset*.

Pada Gambar 8 hingga 10 di bawah merupakan kalimat positif, negatif dan netral yang ada pada *aspect Product*.

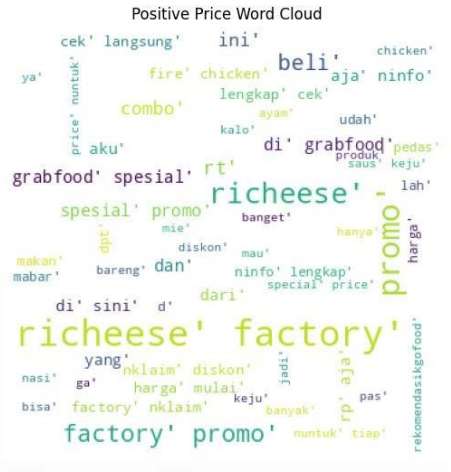
Gambar 6

Gambar 7

Gambar 8

Hasil *Word Cloud* di atas menunjukkan bahwa pada *Aspect Product Positive* kata yang sering muncul adalah beli, ayam, enak. Kata enak banyak digunakan dalam *Label Positive* karena pengguna ingin menunjukkan bahwa *Product* dari *Richeese Factory* rasanya enak. Sedangkan kata yang sering muncul pada *Aspect Product Negative* adalah nasi, kecil. Kata kecil banyak digunakan dalam *Label Negative* karena pengguna ingin menunjukkan bahwa porsi nasi pada *Product Richeese Factory* sedikit.

Pada Gambar di bawah merupakan kalimat positif, negatif dan netral yang ada pada *aspect Price*.



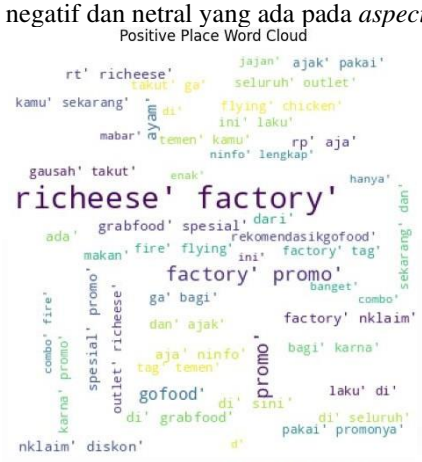
Gambar 9

Gambar 10

Gambar 11

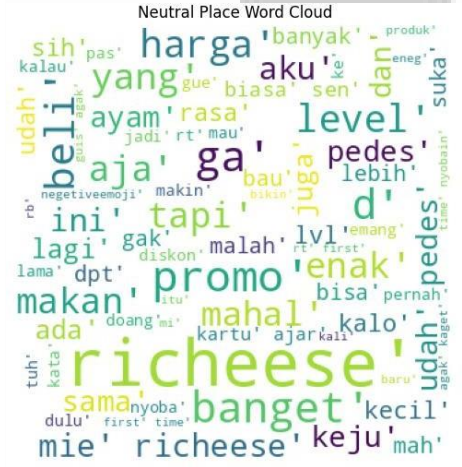
Hasil *Word Cloud* di atas menunjukkan bahwa pada *Aspect Price Positive* kata yang sering muncul adalah *special price*. Kata *special price* banyak digunakan dalam *Label Positive* karena pengguna ingin menunjukkan bahwa harga dari *Richeese Factory* sesuai dengan rasa. Sedangkan kata yang sering muncul pada *Aspect Product Negative* adalah mahal. Kata mahal banyak digunakan dalam *Label Negative* karena pengguna ingin menunjukkan bahwa harga *Richeese Factory* mahal.

Pada Gambar 14 hingga 16 di bawah merupakan kalimat positif, negatif dan netral yang ada pada *aspect Place*.



Gambar 12

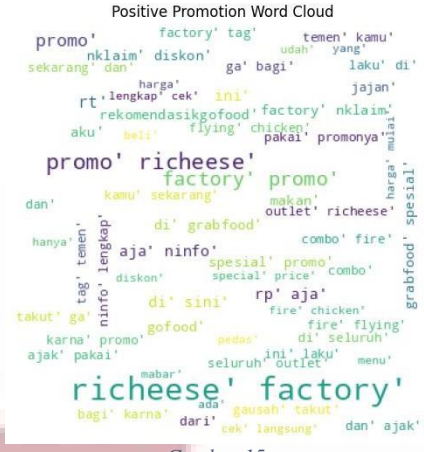
Gambar 13



Gambar 14

Hasil *Word Cloud* di atas menunjukkan bahwa pada *Aspect Place Positive* kata yang sering muncul adalah *grab food*, *gofood*. Kata tersebut banyak digunakan dalam *Label Positive* karena pengguna ingin menunjukkan bahwa *Place* dari *Richeese Factory* mudah di order di *platform* tersebut. Sedangkan kata yang sering muncul pada *Aspect Product Negative* adalah antri. Kata antri banyak digunakan dalam *Label Negative* karena pengguna ingin menunjukkan bahwa *Place RicheeseFactory* sering mengantri.

Pada Gambar 17 hingga 19 di bawah merupakan kalimat positif, negatif dannetral yang ada pada *aspect Place*.



Gambar 15

Gambar 16

Gambar 17

Hasil *Word Cloud* di atas menunjukkan bahwa pada *Aspect Promotion Positive* kata yang sering muncul adalah *special promo*. Kata tersebut banyak digunakan dalam *Label Positive* karena pengguna ingin menunjukkan bahwa *Richeese Factory* sering memberikan *promo*. Sedangkan kata yang sering muncul pada *Aspect Product Negative* adalah mahal. Kata mahal banyak digunakan dalam *Label Negative* karena pengguna ingin menunjukkan bahwa harga *Richeese Factory* tetap mahal walaupun ada promo.

Kata-kata dalam hasil *Word Cloud* pada masing-masing *Aspect* dan *Label* memiliki banyak kata yang sama. Persamaan tersebut terjadi ketika saat proses klasifikasi *machine Learning* menggunakan Algoritme *Naïve Bayes* untuk membuat nilai keseluruhan, sehingga tidak seperti klasifikasi manual, serta masih banyak kata yang seharusnya memiliki *Sentiment Positive* pada *Sentiment Negative* ataupun *Sentiment Neutral*, begitu pula sebaliknya..
=

V. KESIMPULAN

Pada implementasi *Marketing Mix 4*P pelanggan *Richeese Factory* memiliki penilaian *Positive* dan *Neutral* yang lebih banyak dari pada penilaian *Negative* karena pada saat proses *crawling data* dilakukan, *Richeese Factory* banyak memberi promo kepada pelanggan. Pernyataan tersebut dapat dibuktikan pada poin II.A. Lalu pada keempat *Aspect* yang memiliki penilaian *Positive* terbanyak ada pada *Aspect Promotion* dan *Place* senilai 63 dan 54, *Negative* terbanyak terdapat pada *Aspect Product* dan *Price* senilai 24 dan 20, serta *Neutral* terbanyak terdapat pada *Aspect Product* sebanyak 59. Pernyataan tersebut dapat dibuktikan pada Gambar 4.

Pada implementasi Algoritme *Naïve Bayes* pada masing-masing algoritme peneliti mengetahui bahwa tingkat akurasi dapat dipengaruhi oleh perbedaan *Max Features* dan ukuran *Test Size*. Pada implementasi Algoritme *Naïve Bayes* perbedaan *Max Features* sangat berpengaruh pada hasil akurasi, semakin besar *Max Features* maka semakin besar hasil akurasi, pernyataan tersebut dibuktikan pada poin V.B, hasil tingkat akurasi yang memiliki nilai terbesar senilai 81% adalah implementasi yang memiliki nilai *Max Features* tertinggi. Kemudian pada implementasi Algoritme *Naïve Bayes* perbedaan ukuran *Test Size* memiliki pengaruh terhadap hasil akurasi yang dihasilkan, terutama pada Algoritme *Bernoulli*, pernyataan tersebut dibuktikan pada poin V.C, hasil tingkat akurasi terjadi signifikan perbedaan sekitar 9%, tetapi pada algoritme tersebut tinggi rendahnya perbandingan *dataset* tidak berpengaruh terhadap besar dan kecilnya nilai akurasi yang dihasilkan. Sedangkan pada Algoritme *Gaussian* dan *Multinomial* perbedaan yang dihasilkan hanya sekitar 1% hingga 4%.

Pada kedua implementasi Algoritme *Naïve Bayes*, Algoritme *Bernoulli* selalu memiliki nilai akurasi paling rendah, sedangkan Algoritme *Gaussian* dan *Multinomial* nilai akurasi yang dihasilkan cukup besar bernilai sekitar 68% hingga 84%, dan jika diteliti kembali antara kedua algoritme tersebut Algoritme *Multinomial* memiliki nilai akurasi yang paling besar.

Pada implementasi *Marketing Mix* 4P, *Sentiment Negative* yang memiliki nilai cukup tinggi ada pada *Aspect Product* sebanyak 24, pernyataan tersebut dapat dibuktikan pada Gambar 4. Pada implementasi Algoritme *Naïve Bayes*, hasil *confusion matrix* pada *negative sentiment* memiliki nilai *true positive* tertinggi pada *Aspect Product* karena frekuensi kemunculan kata *sentiment negative* sering muncul pada *aspect* tersebut, pernyataan tersebut dapat dibuktikan pada Tabel 23. Sehingga untuk meningkatkan pelanggan *Richeese Factory*, *Product* pada *brand* tersebut merupakan *aspect* yang penting dan harus ditingkatkan.

Saran dari peneliti untuk *Richeese Factory,* tingkatkan kualitas Product Richeese Factory yang sudah ada, misalnya banyak pelanggan yang complaint dengan porsi nasi yang sedikit sehingga bisa lebih diperbanyak, dan banyak pelanggan yang complaint rasa tingkat kepedasan tidak konsisten maka harus dibuat *training* lebih dalam mengenai takaran tingkat kepedasan product agar lebih konsisten.

Saran dari peneliti untuk penelitian selanjutnya, gunakan Algoritme Gaussian atau Multinomial Naïve Bayes pada penelitian Analisis Sentimen berdasarkan aspek dan gunakan *Max Features* TF-IDF yang tinggi sehingga menghasilkan akurasi yang tinggi jika kedua metode tersebut digabungkan

REFERENSI

Electronic References

[1] Ariyanto, “Asal Mula dan Penyebaran Virus Corona dari Wuhan ke Seluruh Dunia,” Indonesia, 2020.

[2] Farid Kusuma, “Detail Cakupan Pengetatan Aktivitas Saat PPKM Darurat Jawa-Bali,” Suara Surabaya, Surabaya, Juli 2021.

[3] Populix, “10 Brand Minuman kekinian yang Paling Digemari Masyarakat,” Populix, September 2020.

[4] GoodStats, “Top 5 Restoran Cepat Saji Terfavorit Masyarakat Indonesia 2022,” GoodStats, 2022.

[5] GoodStats, “Restoran Cepat Saji Favorit Anak Muda Indonesia 2022,” GoodStats, 2022.

[6] Ide Bagus Gede Ari Suyoga dan I Wayan Santik, “Peran Brand Image Dalam Memediasi Pengaruh Electronic Word of Moth Terhadap Niat Beli,” E- Journal Manajemen Unud, hlm. 3230, 2018.

[7] Richeese Factory, “Outlet,” Richeese Factory, 2023.

[8] Richeese Factory, “Richeese Factory,” Instagram Richeese Factory, 2022.

[9] Nadia Araditya Paramastri dan Gumgum Gumilar, “Penggunaan Twitter Sebagai Medium Distribusi Berita dan Newsgathering oleh Tirto.id,” Kajian Jurnalisme, hlm. 18, 2019.

[10] Rifqy Mikoriza Turjaman dan Indra Budi, “Analisis Sentimen Berbasis Aspek Marketing Mix Terhadap Ulasan Aplikasi Dompet Digital (Studi Kasus: Aplikasi Linkaja Pada Twitter),” Jurnal Darma Agung, hlm. 266, 2022.

[11] Auliya Khanza Qorita, “Analisis Sentimen Berbasis Aspek Pada Ulasan Tempat Wisata DIY,” Jurnal Universitas Islam Indonesia, 2022.

[12] Merinda Lestatndy, Abdurahim, dan Lailis Syafa’ah, “Analisis Sentimen Tweet Vaksin COVID-19 Menggunakan Recurrent Neural Network dan Naive Bayes,” Jurnal Resti (Rekayasa Sistem dan Teknologi Informasi), hlm. 802–808, 2021.

[13] Saurabh Shrikant Kulkarni, “An overview of Sentiment Analysis of Twitter Data,” Diposit Digital de Documents de la UAB, 2020.

[14] Willy Ngashu, “Multi-Output Classification with Machine Learning,” Section, 2022.

[15] Xuan Liu dkk., “Emotion classification for short texts: an improved,” Humanit Soc Sci Commun, 2023.

[16] Saurav Pattnaik, “How to correctly use TF-IDF with imbalanced data,” Deepwiz AI, 2021.

[17] Vincentius Westley DImitrius Thomas dan Fitrah Rumaisa, “Analisis Sentimen Ulasan Hotel Bahasa Indonesia Menggunakan Support Vector Machine dan TF-IDF,” Jurnal Media Informatika Budidarma, 2022.

[18] Algoritma Data Indonesia, “Apa itu Naive Bayes?,” Algoritma Data Indonesia, 2022.

[19] Ali Akbar Septiandri, “Modul Naive Bayes Tambahan,” Universitas Al Azhar Indonesia Ali Akbar, 2020.

[20] Agri Yodi Prayoga, Asep Id Hadiana, dan Fajri Rakhmat Umbara, “Deteksi Hoax Pada Berita Online Bahasa Iggris

Menggunakan Bernoulli Naive Bayes Dengan Ekstraksi Fitur TF-IDF," Jurnal Syntax Admiration, 2021.

[21] Yesaya Sergio Vito Putranta, Bayu Rahayudi, dan Welly Purnomo, "Analisis Sentimen Masyarakat terhadap Kebijakan Penghapusan Subsidi BBMpada Media Sosial Twitter menggunakan Algoritma Naive Bayes Classifierdengan Ekstraksi Fitur N-Gram TF- IDF," Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, 2023.

[22] Haroon Javed, "Remove Punctuation from String Python," Linux Hint, 2023.

[23] Chris Cardilo, "Lowercase in Python," Data Camp, 2020.

[24] Muhammad Yasir dan Robertus Suraji, "Perbandingan Metode Klasifikasi Naive Bayes, Decision Tree, Random Forest Terhadap Analisis Sentimen Kenaikan Biaya Haji 2023 Pada Media Sosial Youtube," Jurnal Cahaya Mandalika, 2023.

[25] Shubham Singh, "How to Get Started with NLP – 6 Unique Methods to Perform Tokenization," Analytics Vidhya, 2023.

[26] Styawati Suryati Emi dan Ahmad Ari Aldino, "96Analisis Sentimen Transportasi Online Menggunakan Ekstraksi Fitur Model Word2vec Text Embedding Dan Algoritma Support Vector Machine (SVM)," Jurnal Teknologi dan Sistem Informasi, 2023.

[27] Wilya Kurnia, "Sentimen Analisis Aplikasi E- Commerce Berdasarkan Ulasan Pengguna Menggunakan Algoritma Stochastic Gradient," Jurnal Teknologi dan Sistem Informasi, 2023.

[28] Ika Amelia, Adinda Mutiara, dan Imam Santoso, "Analisis Sentimen Opini Publik Terhadap Pengambil Alihan TMII Oleh Pemerintah dengan Algoritma Naive Bayes," Jurnal Ikraith-Informatika, 2023.

[29] Arie Wijaya dan Prihandoko, "Analisis Sentimen Review Pengguna Aplikasi Depok," Jurnal Ilmiah Informatika Komputer, 2023.