

PROGRAM STUDI SARJANA SISTEM INFORMASI
PROPOSAL
PROYEK PEMROGRAMAN BAHASA ALAMI (11S4037)



Sentiment Analysis for Amazon Food Review Using Naïve Bayes

Disusun Oleh :

12S17012 – Reza Oktovian Siregar

12S17032 – Angelia Regina Ginting

12S17047 – Christina Clara

PROGRAM STUDI SARJANA SISTEM INFORMASI
FAKULTAS INFORMATIKA DAN TEKNIK ELEKTRO
INSTITUT TEKNOLOGI DEL
JANUARI 2021

DAFTAR ISI

BAB I. PENDAHULUAN	6
1.2 Rumusan Masalah.....	8
1.3 Tujuan	8
1.4 Manfaat	8
1.5 Ruang Lingkup	8
BAB II. ISI	9
2.1 Analisis Data	9
2.1.1 Analisa Data.....	9
2.1.2 Analisa Metode	11
2.2 Desain	13
BAB III. IMPLEMENTASI	18
3.1 Implementasi <i>Collecting Data</i>	18
3.2 Implementasi <i>Load data</i> dan <i>Review Data</i>	18
3.3 Implementasi <i>Explore Data</i>	20
3.4 Implementasi <i>Text Pre-Processing</i>	24
3.4.1 Implementasi <i>Convert Into Lower Cases</i>	24
3.4.2 Implementasi <i>Remove Stopwords</i>	24
3.4.3 Implementasi <i>Remove HTML Tags</i>	25
3.4.4 Implementasi <i>Remove Special Characters</i>	25
3.4.5 Implementasi <i>Lemmatization</i>	26
3.4.6 Implementasi <i>Stemming</i>	26
3.5 Implementasi <i>Feature Extraction</i>	27
3.6 Implementasi <i>Modelling</i>	28

3.7 Implementasi Evaluation	30
BAB IV. PENUTUP	31
4.1 Kesimpulan.....	31
4.2 Saran	31
4.3 Pembagian Tugas.....	31
Referensi.....	32

DAFTAR TABEL

Tabel 1. Variabel pada Metode Evaluasi.....	17
Tabel 2. Pembagian Tugas	31

DAFTAR GAMBAR

Gambar 1. Kode prgram read data.....	10
Gambar 2. Rumus Naïve Bayes.....	12
Gambar 3. Tahapan Pemrosesan Bahasa Alami	13
Gambar 4. Kode program load data.....	18
Gambar 5. Kode program menampilkan informasi data.....	19
Gambar 6. Kode program drop kolom	19
Gambar 7. Kode Program melakukanscale nilai pada produk	20
Gambar 8. Kode program split data.....	20
Gambar 9. Kode program wordcloud.....	21
Gambar 10. Wordcloud review buruk.....	22
Gambar 11. Wordcloud review baik	22
Gambar 12. Kode program analisis produk dengan pengguna.....	23
Gambar 13. Visualisasi review pengguna 2000-2012.....	23
Gambar 14. Kode program lower case.....	24
Gambar 15. Kode program remove stopwords.....	24
Gambar 16. Kode program Remove HTML Tags	25
Gambar 17. Kode program remove special characters	25
Gambar 18. Kode program lematisasi.....	26
Gambar 19. Kode program stemming	27
Gambar 20. Kode program feautre extraction	28
Gambar 21. Kode program implementasi klasifikasi.....	28
Gambar 22. Kode program kalkulasi akurasi	29
Gambar 23. Implementasi heatmap	29
Gambar 24. Kode program evaluasi	30

BAB I. PENDAHULUAN

Pada bab ini akan dijelaskan mengenai latar belakang, tujuan, manfaat dan ruang lingkup proyek.

1.1 Latar Belakang

Diera digital di waktu sekarang membuat masyarakat memasuki gaya kehidupan yang baru dimana hampir keseluruhan kaktivitas menggunakan media elektronik. Salah satu gaya kehidupan yang cukup banyak mengalami perubahan yakni dalam kegiatan jual-beli dimana belanja secara *online* sudah menjadi kegiatan yang sudah umum dimana sebelumnya kegiatan jual-beli dengan masih konvensional dan kini sudah menjadi modern dikarenakan kehadiran internet.

Didalam *e-commerce*, *review* dari sesama pengguna menjadi sebuah acuan untuk mencari saran maupun kualitas terkait layanan dan produk sebelum melakukan pembelian. Amazon merupakan website *e-commerce* yang menyediakan berbagai kebutuhan pelanggan seperti produk makanan, produk elektronik, produk kesehatan, produk kecantikan, dan berbagai kebutuhan lainnya. Namun pada penelitian ini akan berfokus pada produk makanan. Didalam website Amazon menyediakan sebuah kolom komentar atau disebut *review* yang dapat dimanfaatkan oleh pelanggan untuk membagikan pengalaman terkait dengan produk yang dibeli. Melaui kolom komentar tersebut dapat menjadi masukan untuk pihak Amazon maupun pelanggan yang ingin melakukan pembelian terhadap produk (makanan) tersebut. Namun dikarenakan informasi yang diberikan cukup banyak maka pelanggan hanya memperhatikan *review* teratas. Didalam kolom *review* tersebut juga banyak kata-kata yang dapat menjadi masukan untuk pihak amazon maupun penjual barang yang terkait dengan kualitas barang.

Pengguna yang memanfaatkan *e-commerce* untuk transaksi jual-beli semakin bertumbuh setiap waktu ditambah juga produk yang dijual semakin beragam sehingga pada akhirnya akan menghasilkan *review* yang berjumlah banyak pada setiap produk. Dikarenakan jumlah data yang sangat besar dan sangat banyak sehingga tidak mungkin untuk melakukan analisis dan menghasilkan kesimpulan dengan cara yang manual [1]. Selain dikarenakan jumlah data yang besar, masalah yang lain yakni informasi pada *review* adalah berupa opini. Opini berisikan informasi yang bersifat tidak lengkap, tidak konsisten, atau informasi yang berperilaku pada tren tertentu [2]. Banyak komentar juga menggunakan kaidah bahasa yang tidak formal. Melalui proses ekstraksi informasi dan mengembangkan sistem analisis sentimen, maka sistem tersebut

nantinya dapat dimanfaatkan untuk menentukan kualitas produk dengan penilaian secara objektif dan dapat juga dimanfaatkan untuk mengatasi masalah informasi yang jumlahnya besar. Sehingga analisis sentimen ini dapat digunakan untuk menemukan polaritas sentimen dari suatu pernyataan atau kalimat dan mengkategorikan menjadi kategori positif, netral serta negatif.

Analisis sentimen merupakan sebuah bidang yang mempelajari terkait opini, tanggapan, dan emosi dari seseorang terhadap sebuah entitas yakni salah satunya *review* produk [1]. Sentimen didalam bentuk teks mengandung informasi yang bersifat penting dan dapat diambil melalui *text mining*. *Text mining* merupakan tahapan awal dari proses analisis sentiment analisis dan sangat penting karena dimanfaatkan untuk mengidentifikasi emosional melalui pernyataan. [3]. Lalu proses selanjutnya dilakukannya analisis sentimen.

Berbagai penelitian juga semakin berkembang dalam bidang analisis sentimen yang dimanfaatkan sebagai solusi yang sesuai karena sistem yang dirancang sudah dapat secara otomatis melakukan analisis terhadap *review* yang berada diinternet dan mengekstrak informasi yang memungkinkan paling relevan terhadap pengguna. Analisis sentimen dimanfaatkan untuk dapat memantau pendapat atau kecenderungan opini terhadap sebuah masalah atau objek oleh seseorang, apakah cenderung berpandangan atau beropini negatif atau positif (Bo Pang, 2008) Berdasarkan pendapat Azam & Zao (2012) bahwa dengan mendapatkan informasi yang relevan dan tepat waktu dari keseluruhan *review* itu menjadi bagian yang sangat penting. Terdapat berbagai metode untuk melakukan analisis sentiment yakni Naïve Bayes sebagai salah satu pilihan pendekatan yang tepat. Pada penelitian ini akan menggunakan metode Naive Bayes. Naïve Bayes merupakan sebuah pendekatan yang berbasis probalistik. Metode Naïve Bayes Classifier merupakan metode sederhana namun disamping itu metode ini mempunyai nilai akurasi dan performansi yang tinggi dalam mengklasifikasikan sebuah teks [4]. Disamping itu terdapat juga dukungan pendapat dalam pemilihan metode Naïve Bayes yakni pada penelitian sebelumnya menyatakan bahwa naive bayes classifier cukup membutuhkan sejumlah kecil data pelatihan untuk mengestimasi parameter (varian dari kelas) yang diperlukan untuk klasifikasi (Chandra, et al., 2016).

Hasil akhir yang diharapkan pada penelitian analisis sentimen ini untuk mengidentifikasi sentiment positif dan negatif dari para *reviewer* pada *Amazon* dengan divisi *item food*. Klasifikasi pada label sentimen analisis yakni memakai penerapan dari ilmu linguistik *Natural*

Language Processing dengan menerapkan tahapan *text preprocessing*, *text classification*, dan *language modeling*. Hasil dari analisis ini dapat dimanfaatkan sebagai media yang dapat memberikan pertimbangan pada pelanggan maupun penjual untuk mengambil keputusan dimasa mendatang berdasarkan pengelompokkan opini yang telah dilakukan.

1.2 Rumusan Masalah

Berdasarkan pemaparan latar belakang dapat dirumuskan rumusan masalah didalam proyek Pemrograman Bahasa Alami ini adalah untuk bagaimana mengidentifikasi klasifikasi *review* pada amazon dengan kategori makanan (*food*) sehingga dapat memperoleh informasi mengenai kecenderungan opini pembeli melalui pendekatan Naïve Bayes .

1.3 Tujuan

Tujuan proyek Pemrograman Bahasa Alami ini adalah mengimplementasikan pendekatan Naïve Bayes untuk mengidentifikasi *review* pengguna amazon didalam produk kategori *Food*.

1.4 Manfaat

Manfaat yang diharapkan dari penelitian ini yakni membantu pihak Amazon dan penjual produk untuk mendapatkan *feedback* penting yakni yang sudah diklasifikasikan sehingga dapat langsung mendapatkan penilaian terhadap kualitas produk serta juga pengguna juga mendapatkan informasi penting dan relevan mengenai kualitas produk berdasarkan opini pengalaman pembeli sebelumnya sehingga pihak-pihak yang terkait didalam amazon dapat mengambil keputusan yang tepat

1.5 Ruang Lingkup

Ruang lingkup dalam pengerjaan proyek ini adalah menggunakan metode Naïve Bayes dengan data sampel yang diperoleh dari kaggle.com. Dataset yang digunakan adalah data komentar pengguna pada Amazon pada kategori *Food*.

BAB II. ISI

2.1 Analisis Data

Pada subbab ini akan menjabarkan hasil analisis yang telah dilakukan terhadap data serta metode yang akan dimanfaatkan pada proses implementasi pemrosesan bahasa alami. Data set yang digunakan didalam penelitian berasal dari *Amazon Fine Food review* yang diperoleh dari kaggle (<https://www.kaggle.com/snap/amazon-fine-food-reviews>).

2.1.1 Analisa Data

Data ini akan berisi tentang kumpulan opini pembeli mengenai makanan sedap yang ada pada Amazon dengan jangka waktu lebih dari 10 tahun, termasuk 568.454 ulasan hingga Oktober 2012. Ulasan termasuk peringkat, informasi produk dan pengguna, dan ulasan teks biasa. Ini juga mencakup ulasan dari semua kategori Amazon lainnya.

Adapun attribut yang berada pada dataset ini yakni :

1. ProductId: Pengidentifikasi unik untuk produk
2. Id: pengenalan unik untuk pengguna
3. ProfileName: Nama profil pengguna
4. HelpfulnessNumerator: Jumlah pengguna yang menganggap review bermanfaat
5. HelpfulnessDenominator: Jumlah pengguna yang menunjukkan apakah ulasan tersebut bermanfaat atau tidak
6. Score: Peringkat antara 1 dan 5
7. Time: Stempel waktu
8. Summary: Ringkasan review
9. Text : Review

```
In [10]: 1 dataset.head()
```

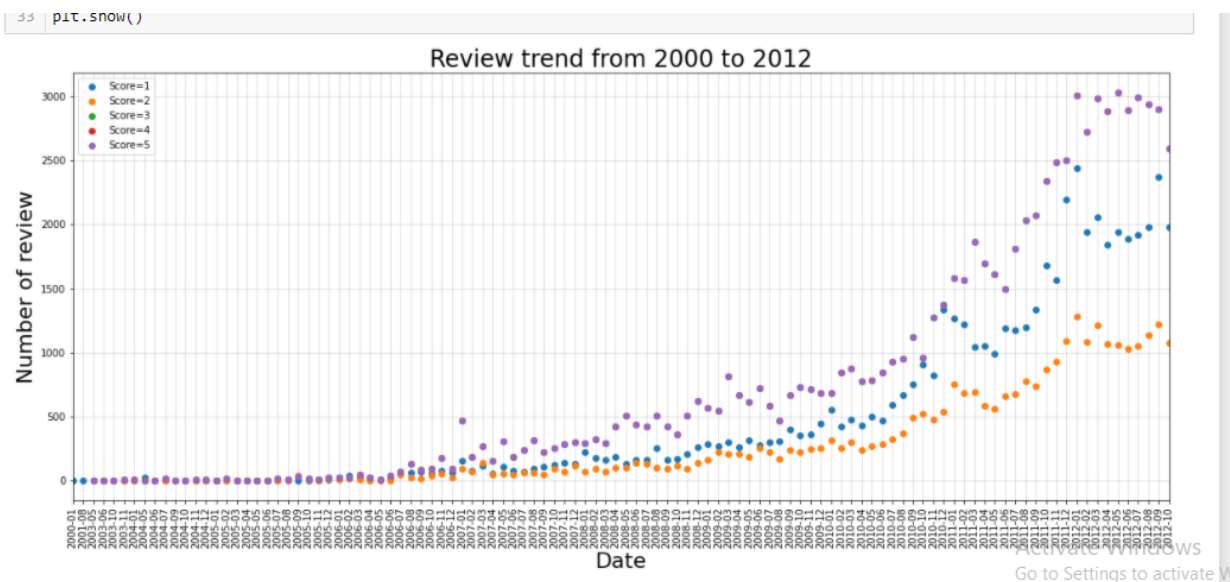
```
Out[10]:
```

	Id	ProductId	UserId	ProfileName	HelpfulnessNumerator	HelpfulnessDenominator	Score	Time	Summary	Text
0	1	B001E4KFG0	A3SGXH7AUHU8GW	delmartian	1	1	5	1303862400	Good Quality Dog Food	I have bought several of the Vitality canned dog food products and have found them all to be of good quality. The product looks more like a stew than a processed meat and it smells better. My Labr...
1	2	B00813GRG4	A1D87F6ZCVE5NK	dll pa	0	0	1	1346976000	Not as Advertised	Product arrived labeled as Jumbo Salted Peanuts...the peanuts were actually small sized unsalted. Not sure if this was an error or if the vendor intended to represent the product as "Jumbo".
2	3	B000LQOCH0	ABXLMWJIXXAIN	Natalia Corres "Natalia Corres"	1	1	4	1219017600	"Delight" says it all	This is a confection that has been around a few centuries. It is a light, pillowy citrus gelatin with nuts - in this case Filberts. And it is cut into tiny squares and then liberally coated with If you are looking for the secret ingredient in

Gambar 1. Kode program read data

Opini-Opini pengguna tersebut akan ditentukan positif dan negative menggunakan Score. Sebagai contoh dimana opini yang bernilai score 1 hingga 3 dikategorikan sebagai opini buruk dan 4 hingga 5 sebagai opini baik.

Untuk membantu analisis dapat ditinjau melalui tahapan eksplorasi data seperti melihat menganalisis tren ulasan.



Dari visualisasi diatas , peneliti dapat mengetahui dari tahun 2000 hingga 2006 jumlah tinjauan konsisten. Namun setelah itu, jumlah review mulai meningkat. Dari jumlah tersebut, sejumlah review dengan rating 5 bintang tergolong tinggi.



Terlihat bahwa adanya perbedaan ukuran, dan pendapat yang paling dominan yakni *good* dan *great*.

Naïve Bayes adalah teknik klasifikasi yang berlandaskan pada teorema Bayes. Metode pengklasifikasian dengan menggunakan metode probabilitas dan statistik yakni melakukan prediksi peluang berdasarkan pengalaman di masa sebelumnya (Teorema Bayes) dengan karakteristik utama yakni asumsi yang paling kuat akan ketergantungan dari masing-masing kondisi.

```

1 total_size=len(dataset)
2
3 train_size=int(0.70*total_size)
4
5 #untuk training dataset
6 train=dataset.head(train_size)
7 #untuk test dataset
8 test=dataset.tail(total_size - train_size)

```

Naive Bayes menerapkan fungsi statistik sederhana berdasarkan teorema bayes dengan asumsi keberadaan dari suatu fitur tertentu terhadap suatu kelas yang tidak berhubungan dengan fitur lainnya (Wilianto, Pudjiantoro, & Umbara, 2017).

Rumus Naïve Bayes :

Persamaan dari probabilitas prior:

$$P(H) = \frac{N_j}{N}$$

N_j : J umlah data pada suatu class

N : J umlah total data

Persamaan dari teorema bayes:

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

X : Data class belum diketahui

H : Hipotesis data class spesifik

$P(H|X)$: Probabilitas Hipotesis H terhadap kondisi X

$P(H)$: Probabilitas Hipotesis H

$P(X|H)$: Probabilitas X terhadap kondisi hipotesis H

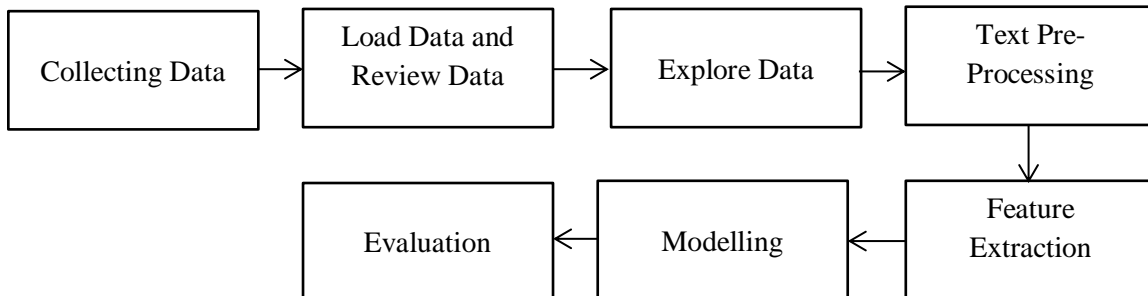
$P(X)$: Probabilitas X

Gambar 2. Rumus Naïve Bayes

Sumber([https://www.researchgate.net/publication/339234570 Penerapan Algoritma Naive Bayes Untuk Analisis Sentimen Pada Wisata TMII Berbasis Website](https://www.researchgate.net/publication/339234570_Penerapan_Algoritma_Naive_Bayes_Untuk_Analisis_Sentimen_Pada_Wisata_TMII_Berbasis_Website))

2.2 Desain

Pada bab ini akan dijabarkan mengenai tahapan pemrosesan bahasa alami yang akan diterapkan pada proyek. Adapun tahapan pemrosesan dapat dilihat pada aliran diagram pada Gambar 1.



Gambar 3. Tahapan Pemrosesan Bahasa Alami

1. Collecting Data

Pada tahap ini dilakukan pengumpulan data yang merupakan proses mengumpulkan dan memastikan informasi pada subjek yang akan dilakukan uji coba yang bertujuan untuk mendapatkan jawaban pertanyaan dari uji coba yang dilakukan, uji hipotesis dan mengevaluasi hasil. Proyek Sentimen Analisis akan dilakukan dengan menggunakan *dataset* yang berasal dari *kaggle* yang sudah di *download* dalam bentuk *file .csv* yaitu *Amazon Fine Food Review Sentiment Analysis* dengan 10 kolom dan 568454 baris. *Dataset* yang digunakan berisikan atribut mengenai Id, ProductId, UserId, ProfileName, HelpfulnessNumerator, HelpfulnessDenomitor, Score, Time, Summary, dan Text.

2. Load Data

Pada tahap ini dilakukan *load data* untuk menampilkan atribut dan isi dari *dataset* tersebut. Selain itu juga, tim proyek dapat memilih isi data dari atribut mana yang akan ditampilkan serta jumlah data yang ingin ditampilkan.

3. Explore Data

Pada tahap ini dilakukan eksplorasi data untuk memungkinkan tim proyek memahami isi data yang digunakan. *Explore data* yang dilakukan menggunakan *Word Cloud* atau *Tag Cloud*, dimana *Word Cloud* ini merupakan representasi visual dari data teks dengan menggambarkan

metadata atau dengan kata lain memvisualisasikan suatu bentuk teks secara bebas dengan berbagai ukuran (*font*) atau warna.

4. Text Pre-Processing

Pada tahap ini dilakukan *preprocessing data* dengan tujuan untuk menghilangkan *noise*, menyeragamkan bentuk kata, dan mengurangi volume kosakata. Adapun tahapan *preprocessing* ini meliputi:

1. *Convert into lower cases*

Proses ini dilakukan dengan mengkonversi semua huruf menjadi *lower case* atau huruf kecil. Hal ini dilakukan agar semua data yang masuk pada tahap berikutnya menjadi sama dan standar. Contohnya, kata “Pemrosesan Bahasa Alami” akan dikonversi menjadi kata “pemrosesan bahasa alami”. [5]

2. *Remove punctuations*

Proses ini dilakukan dengan menghapus tanda baca pada suatu kalimat. Contohnya, kata “this taffy is so good. it is very soft and chewy” menjadi “this taffy is so good it is very soft and chewy”.

3. *Remove stop words*

Proses ini dilakukan untuk memeriksa *stop word list* atau daftar kata-kata yang seharusnya dihilangkan. Proses ini dilakukan dengan membuat *token* atau *tokenize* dari kumpulan data terlebih dahulu, dimana kata yang kurang memiliki arti atau tidak memenuhi syarat sebagai kata akan dibuang. Namun, apabila kata-kata tersebut memang menjadi bagian dari sebuah kalimat, maka kata tersebut tidak dapat digunakan sebagai *token* karena kata tersebut berupa kata sambung atau keterangan. Misalnya “the”, “this”, “that”, “these”, “those”, “his”, “her”, “its”, “your”, “my”, dan lain-lain. [5] [6]

4. *Remove HTML tags*

Proses ini dilakukan untuk menghapus semua *tag* HTML yang ada pada daftar kata. Misalnya *tag* seperti “<a>”, “”, “
”, “</br>”, dan lain-lain akan dihapus.

5. *Remove special characters*

Proses ini dilakukan untuk menghapus semua *spescial character* dari setiap kalimat. Misalnya “he’s learned about whales, india, drooping roses” menjadi “he s learned about whales india drooping roses”.

6. *Lemmatization*

Proses ini dilakukan untuk mengubah setiap *token* ke bentuk dasar umum dari kata tersebut. Pada proses ini, setiap kata akan diubah menjadi kata dasar dan membuat peningkatan fitur benar pada setiap dokumen yang mengakibatkan peningkatan pengambilan fitur yang terekstrak. Dalam kata lain, *lemmatization* adalah suatu proses untuk menemukan bentuk dasar dari sebuah kata [7] [8].

7. *Stemming*

Proses ini dilakukan dengan menghilangkan semua imbuhan, baik imbuhan awalan, sisipan, akhiran, maupun gabungan awalan dan akhiran. Proses *stemming* dilakukan juga karena pada proses sebelumnya masih terdapat kata yang sebenarnya memiliki arti yang sama namun berbeda bentuk kata. Hal ini dikarenakan pemakaian bentuk kata yang berbeda dalam setiap kalimat. Misalnya kata “disease”, “diseases”, “diseased”, akan diubah menjadi bentuk “disease” [5] [8]

5. Feature Extraction

Feature extraction merupakan sebuah proses mengekstrak informasi-informasi yang paling relevan untuk dikategorikan dengan meminimalkan variasi antar pola dalam suatu kategori dan sekaligus memaksimalkan variasi antar kategori yang berbeda dari data mentah yang tersedia [9].

Feature extraction yang dapat digunakan pada penelitian ini yaitu menggunakan metode *Bag of Words*.

Metode *Bag of Words* (BoW) merupakan representasi sederhana yang digunakan pada pemrosesan bahasa alami, dimana sebuah teks yang berupa kalimat atau dokumen diwakili sebagai kantung (*bag*) *multiset* dari kata-kata yang terkandung didalamnya tanpa memandang urutan kata dan tata bahasa, namun tetap mempertahankan keberagamannya. Pengertian lain dari BoW adalah sebuah model yang mempelajari sebuah kosa kata dari seluruh dokumen untuk kemudian memodelkan tiap dokumen dengan menghitung jumlah kemunculan setiap kata.

6. Modelling

Pada tahap ini dilakukan pembangunan sentimen analisis untuk *food review amazon* menggunakan *Naïve Bayes*. Metode *Naïve Bayes* digunakan untuk mengklasifikasikan sebuah teks, dimana metode ini menggunakan perhitungan probabilistik dengan cara melakukan

penjumlahan terhadap frekuensi dan kombinasi dari sebuah dataset. Hasil dari proses klasifikasi ini kemudian akan diuji untuk menentukan tingkat akurasi sistem dalam melakukan proses klasifikasi.

7. Evaluation

Evaluation atau pengujian dilakukan dengan membagi data ke dalam data *training* dan data *testing*. Metode pengujian merupakan penilaian seberapa baik proses identifikasi pada sistem dengan menggunakan rumus umum perhitungan *accuracy*, *precision*, *recall*, dan *F-score* yang diperkenalkan oleh Baeza-Yates & Ribeiro-Neto [10].

- *Accuracy* adalah tingkat kedekatan antara nilai prediksi dengan nilai aktual.

Rumus yang digunakan untuk perhitungan *accuracy* dapat dilihat pada persamaan (1) dan (2) berikut.

$$Akurasi = \frac{\text{Jumlah Prediksi yang Benar}}{\text{Jumlah Keseluruhan}} \quad (1)$$

$$Akurasi = \frac{(TP+TN+Tnt)}{TP+TN+Tnt+FN+FNt+FP} \quad (2)$$

- *Precision* adalah tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem

Rumus yang digunakan untuk perhitungan *precicion* dapat dilihat pada persamaan (3) berikut.

$$Precision = \frac{TP}{TP+FP} \quad (3)$$

- *Recall* adalah tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi.

Rumus yang digunakan untuk perhitungan *recall* dapat dilihat pada persamaan (4) berikut.

$$Recall = \frac{TP}{TP+FN} \quad (4)$$

- *F1-Score* adalah *harmonic mean* dari *precision* dan *recall*.

Rumus yang digunakan untuk perhitungan *f1-score* dapat dilihat pada persamaan (5) berikut.

$$\frac{1}{F1} = \frac{1}{2} \left(\frac{1}{precision} + \frac{1}{recall} \right) \quad (5)$$

Tabel 1. Variabel pada Metode Evaluasi

		Label Manual		
		Positif	Negatif	Netral
Hasil Identifikasi	Positif	True Positif (TP)	False Positif (FP)	False Positif (FP)
	Negatif	False Negatif (FN)	True Negatif (TN)	False Negatif (FN)
	Netral	False Netral (FN)	False Netral (FNt)	True Netral (TNt)

Keterangan:

- True Positif (TP) adalah kondisi data dengan label positif berhasil diidentifikasi sebagai kalimat positif
- False Positif (FP) adalah kondisi data dengan label positif gagal diidentifikasi sebagai kalimat positif
- True Negatif (TN) adalah kondisi data dengan label negatif berhasil diidentifikasi sebagai kalimat negatif
- False Negatif (FN) adalah kondisi data dengan label negatif gagal diidentifikasi sebagai kalimat negatif
- True Netral (TN) adalah kondisi data dengan label netral berhasil diidentifikasi sebagai kalimat netral
- False Netral (FN) adalah kondisi data dengan label netral gagal diidentifikasi sebagai kalimat netral

BAB III. IMPLEMENTASI

Pada bab ini akan dijabarkan mengenai implementasi proyek sentiment analisis menggunakan Naïve Bayes yang diimplementasikan pada jupyter notebook. Implementasi yang dilakukan pada proyek ini berdasarkan desain yang telah dijabarkan pada bab 2.

3.1 Implementasi *Collecting Data*

Data yang digunakan merupakan kumpulan *review* makanan dari pengguna Amazon, yang diperoleh melalui Kaggle dan dapat diakses pada situs berikut <https://www.kaggle.com/snap/amazon-fine-food-reviews> sebanyak 568,454 *food reviews*.

3.2 Implementasi *Load data* dan *Review Data*

Pada *jupyter notebook* untuk melakukan *load* data dapat di tuliskan dengan kode program berikut, untuk melihat sebagai data yang sudah di*load* dapat menggunakan perintah `.head()` yang selanjutnya akan menampilkan beberapa data yang berasal dari *csv*.

```
dataset = pd.read_csv("../Reviews.csv", error_bad_lines=False)
```

```
dataset.head()
```

	ID	ProductId	UserId	ProfileName	HelpfulnessNumerator	HelpfulnessDenominator	Score	Time	Summary	Text
0	1	B001E4KFG0	A3SGXH7AUHU8GW	dellmartian	1	1	5	1303862400	Good Quality Dog Food	I have bought several of the Vitality canned dog food products and have found them all to be of good quality. The product looks more like a stew than a processed meat and it smells better. My Labr...
1	2	B00813GRG4	A1D87F6ZCVE5NK	dll pa	0	0	1	1346976000	Not as Advertised	Product arrived labeled as Jumbo Salted Peanuts...the peanuts were actually small sized unsalted. Not sure if this was an error or if the vendor intended to represent the product as "Jumbo".
2	3	B000LQOCH0	ABXLMWJDXAIN	Natalia Corres "Natalia Corres"	1	1	4	1219017600	"Delight" says it all	This is a confection that has been around a few centuries. It is a light, pillowy citrus gelatin with nuts - in this case Filberts. And it is cut into tiny squares and then liberally coated with ...

Gambar 4. Kode program load data

Perintah `.shape` dan `.info` akan menampilkan berapa banyak kolom dan baris pada dataset dan memberikan informasi mengenai dataset itu sendiri. Adapun kode program dari perintah tersebut dapat dilihat pada kode program dibawah ini.

dataset.shape									
(568454, 10)									
dataset.info									
<bound method DataFrame.info of									
0	1	B001E4KFG0	A3SGXH7AUHU8GW		Id	ProductId	UserId	ProfileName	\
1	2	B00813GRG4	A1D87F6ZCVE5NK				delmartian		
2	3	B000LQOCH0	ABXLMWJIXXAIN	Natalia Corres		"Natalia Corres"			
3	4	B000UA0QIQ	A395BORC6FGVXV				Karl		
4	5	B006K2ZZ7K	A1UQRSCLF8GW1T	Michael D. Bigham		"M. Wassir"			
...		
568449	568450	B001E07N10	A28KG5XOR054AY				Lettie D. Carter		
568450	568451	B003S1WTCU	A3I8AFVPEE8KI5				R. Sawyer		
568451	568452	B004I613EE	A121AA1GQV751Z				pkds "pk_007"		
568452	568453	B004I613EE	A3IBEVCTXKNOH	Kathy A. Welch		"katwel"			
568453	568454	B001LR2CU2	A3LGQPJCZVL9UC				srfell17		
HelpfulnessNumerator HelpfulnessDenominator Score Time \									
0		1	1	5	1303862400				
1		0	0	1	1346976000				
2		1	1	4	1219017600				
3		3	3	2	1307923200				
4		0	0	5	1350777600				
...				
568449				

Gambar 5. Kode program menampilkan informasi data

Pada implementasi ini tim proyek melakukan *drop* pada kolom yang tidak diperlukan Id, ProductId, HelpfulnessNumerator, HelpfulnessDenominator, dan Time.

#Dropping the unwanted columns from our data frame.				
dataset.drop("Id", inplace=True, axis=1)				
dataset.drop("ProductId", inplace=True, axis=1)				
dataset.drop("ProfileName", inplace=True, axis=1)				
dataset.drop("HelpfulnessNumerator", inplace=True, axis=1)				
dataset.drop("HelpfulnessDenominator", inplace=True, axis=1)				
dataset.drop("Time", inplace=True, axis=1)				
dataset.head()				
	UserId	Score	Summary	Text
0	A3SGXH7AUHU8GW	5	Good Quality Dog Food	I have bought several of the Vitality canned dog food products and have found them all to be of good quality. The product looks more like a stew than a processed meat and it smells better. My Labr...
1	A1D87F6ZCVE5NK	1	Not as Advertised	Product arrived labeled as Jumbo Salted Peanuts...the peanuts were actually small sized unsalted. Not sure if this was an error or if the vendor intended to represent the product as "Jumbo".
2	ABXLMWJIXXAIN	4	"Delight" says it all	This is a confection that has been around a few centuries. It is a light, pillowy citrus gelatin with nuts - in this case Filberts. And it is cut into tiny squares and then liberally coated with ...
3	A395BORC6FGVXV	2	Cough Medicine	If you are looking for the secret ingredient in Robitussin I believe I have found it. I got this in addition to the Root Beer Extract I ordered (which was good) and made some cherry soda. The fl...
4	A1UQRSCLF8GW1T	5	Great taffy	Great taffy at a great price. There was a wide assortment of yummy taffy. Delivery was very quick. If your a taffy lover, this is a deal.

Gambar 6. Kode program *drop* kolom

Score yang diberikan pengguna memiliki ragam variasi disini dilakukan penskalaan dimana untuk produk yang memiliki nilai lebih atau sama dengan 3 akan mendapatkan nilai 1 dan untuk produk yang bernilai kurang dari 3 akan mendapatkan nilai 0.

```
#Make all 'Score' less than 3 equal to -ve class and
# 'Score' greater than 3 equal to +ve class.
dataset.loc[dataset['Score']<3, 'Score'] = [0]
dataset.loc[dataset['Score']>3, 'Score'] = [1]
```

```
dataset.head()
```

	UserId	Score	Summary	Text
0	A3SGXH7AUHU8GW	1	Good Quality Dog Food	I have bought several of the Vitality canned dog food products and have found them all to be of good quality. The product looks more like a stew than a processed meat and it smells better. My Labr...
1	A1D87F6ZCVE5NK	0	Not as Advertised	Product arrived labeled as Jumbo Salted Peanuts...the peanuts were actually small sized unsalted. Not sure if this was an error or if the vendor intended to represent the product as "Jumbo".
2	ABXLMWJIXXAIN	1	"Delight" says it all	This is a confection that has been around a few centuries. It is a light, pillowy citrus gelatin with nuts - in this case Filberts. And it is cut into tiny squares and then liberally coated with ...
3	A395BORC8FGVXV	0	Cough Medicine	If you are looking for the secret ingredient in Robitussin I believe I have found it. I got this in addition to the Root Beer Extract I ordered (which was good) and made some cherry soda. The fl...
4	A1UQRSCLF8GW1T	1	Great taffy	Great taffy at a great price. There was a wide assortment of yummy taffy. Delivery was very quick. If your a taffy lover, this is a deal.

Gambar 7. Kode Program melakukan *scale* nilai pada produk

Implementasi yang dilakukan melakukan *split* data untuk *trainee data* dan *test data* dimana pada *train data* digunakan 0.70 dari ukuran dataset sisanya akan digunakan untuk *test data*.

```
total_size=len(dataset)
train_size=int(0.70*total_size)

#untuk training dataset
train=dataset.head(train_size)
#untuk test dataset
test=dataset.tail(total_size - train_size)
```

Gambar 8. Kode program *split* data

3.3 Implementasi Explore Data

Pada tahap eksplorasi data akan dilakukan visualisasi review customer menggunakan Word Cloud. Word cloud (atau disebut juga tag cloud) adalah representasi visual dari data teks, biasanya digunakan untuk menggambarkan metadata atau untuk memvisualisasikan suatu bentuk teks secara bebas. Wordcloud (atau Tag cloud) adalah representasi visual dari data teks ini. Ini menampilkan daftar kata dengan berbagai ukuran font atau warna yang berguna untuk

Pada pengimplementasian *WordCloud* membutuhkan inputan *single string* dari teks, ringkasan *review* akan digabungkan menjadi *single string*, similarly akan dibangun melalui atribut *Text*.

Pada pengimplementasian review dengan score buruk akan digunakan *background* berwarna putih dan pengimplementasian review dengan score baik akan digunakan *background* berwarna hitam.

```

wordcloud_bad = WordCloud(background_color='white').generate(bad_reviews_str)
wordcloud_good = WordCloud(background_color='black').generate(good_reviews_str)
# Plot
fig = plt.figure(figsize=(10,10))
ax1 = fig.add_subplot(211)
ax1.imshow(wordcloud_bad,interpolation='bilinear')
ax1.axis("off")
ax1.set_title('Review dengan Score Buruk',fontsize=20)

```

Text(0.5, 1.0, 'Review dengan Score Buruk')



Gambar 10. Wordcloud review buruk

```

fig = plt.figure(figsize=(10,10))
ax2 = fig.add_subplot(212)
ax2.imshow(wordcloud_good,interpolation='bilinear')
ax2.axis("off")
ax2.set_title('Review dengan Score Baik',fontsize=20)
plt.show()

```



Gambar 11. Wordcloud review baik

Analisis produk dengan pengguna juga dilakukan pada implementasi ini dimana akan ditampilkan id dari pengguna dan barang yang sudah dibeli.

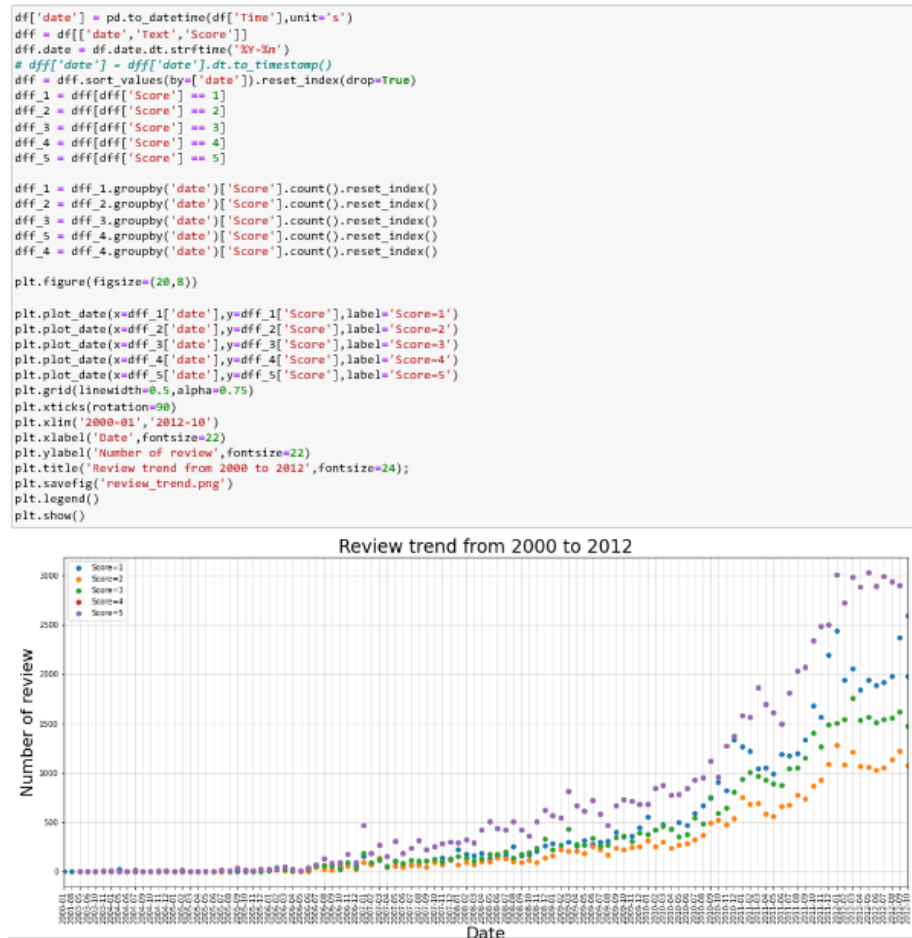
```
# we will see the products brought by each customer

purchases = df[['ProductId','UserId']].groupby('UserId').agg({'ProductId': ['count']})
purchases.columns = ['No_of_products_purchased']
purchases = purchases.reset_index()
purchases.head(2)
```

	UserId	No_of_products_purchased
0	#oc-R103C0QSV1DF5E	1
1	#oc-R109MU50BBZ59U	1

Gambar 12. Kode program analisis produk dengan pengguna

Tim proyek melakukan analisis pada *review* pengguna dari tahun ketahun dimana rentang waktu nya adalah dari tahun 2000 sampai 2012 pada grafik dapat dilihat bahwa review pengguna dengan score 5 semkain meningkat setiap tahunnya.



Gambar 13. Visualisasi review pengguna 2000-2012

3.4 Implementasi *Text Pre-Processing*

Pada bab ini akan dijabarkan mengenai tahapan pemrosesan bahasa alami yang akan diterapkan seperti *convert into lower cases*, *remove punctuations*, *remove stop words*, *remove HTML tags*, *remove special characters*, *lemmatization*, dan *stemming*.

3.4.1 Implementasi *Convert Into Lower Cases*

Proses *case folding* bertujuan untuk mengubah seluruh huruf kapital menjadi huruf kecil. Terdapat beberapa cara yang dapat dilakukan pada tahap *case folding*, seperti mengubah teks menjadi *lowercase*, menghapus angka, menghapus karakter kosong.

```
#digunakan untuk mengkonversi semua daftar ke lowercase
lst_text = [str(item).lower() for item in lst_text]
lst_summary = [str(item).lower() for item in lst_summary]

test_text = [str(item).lower() for item in test_text]

# lower casing and removing punctuations
df['Text'] = df['Text'].apply(lambda x: " ".join(x.lower() for
x in x.split()))
```

Gambar 14. Kode program *lower case*

3.4.2 Implementasi *Remove Stopwords*

Kinerja klasifikasi akan menjadi lebih optimal dengan menghapus kata-kata yang jarang muncul dan akan mengurangi data sparsial dan menyusutkan ruang fitur secara substansial.

```
#Melakukan penghapusan semua Stop Words
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
#word_tokenize menerima sebuah string sebagai masukan bukan sebuah file
stop_words = set(stopwords.words('english'))
for i in range(len(lst_text)):
    text_filtered = []
    summary_filtered = []
    text_word_tokens = []
    summary_word_tokens = []
    text_word_tokens = lst_text[i].split()
    summary_word_tokens = lst_summary[i].split()
    for r in text_word_tokens:
        if not r in stop_words:
            text_filtered.append(r)
    lst_text[i] = ' '.join(text_filtered)
    for r in summary_word_tokens:
        if not r in stop_words:
            summary_filtered.append(r)
    lst_summary[i] = ' '.join(summary_filtered)

for i in range(len(test_text)):
    text_filtered = []
    text_word_tokens = []
    text_word_tokens = test_text[i].split()
    for r in text_word_tokens:
        if not r in stop_words:
            text_filtered.append(r)
    test_text[i] = ' '.join(text_filtered)
```

Gambar 15. Kode program *remove stopwords*

3.4.3 Implementasi Remove HTML Tags

Pada implementasi juga dilakukan penghapusan pada tag html yang terdapat pada setiap daftar kata yang ada.

```
#dilakukan untuk menghapus semua HTML Tags yang ada pada list
import re
def striphtml(data):
    p = re.compile(r'<.*>')
    return p.sub('', data)

for i in range(len(lst_text)):
    lst_text[i] = striphtml(lst_text[i])
    lst_summary[i] = striphtml(lst_summary[i])

for i in range(len(test_text)):
    test_text[i] = striphtml(test_text[i])

lst_text[0:5]

['bought several vitality canned dog food products found good quality. product looks like stew proces
sed meat smells better. labrador finicky appreciates product better most.',
'product arrived labeled jumbo salted peanuts...the peanuts actually small sized unsalted. sure erro
r vendor intended represent product "jumbo".',
'confection around centuries. light, pillowy citrus gelatin nuts - case filberts. cut tiny squares l
iberally coated powdered sugar. tiny mouthful heaven. chewy, flavorful. highly recommend yummy treat.
familiar story c.s. lewis\' "the lion, witch, wardrobe" - treat seduces edmund selling brother sister
s witch.',
'looking secret ingredient robitussin believe found it. got addition root beer extract ordered (whic
h good) made cherry soda. flavor medicinal.',
'great taffy great price. wide assortment yummy taffy. delivery quick. taffy lover, deal.']
```

Gambar 16. Kode program Remove HTML Tags

3.4.4 Implementasi Remove Special Characters

Remove Special Characters bertujuan untuk menghilangkan karakter seperti angka, tanda baca dan hal lain yang bukan merupakan alfabet a-z karena tanda baca yang terdapat didalam suatu kata nantinya akan sulit untuk diklasifikasikan karena tidak sesuai dengan kata yang terdapat pada kumpulan teks.

```
#Dilakukan penghapusan spesial karakter dari setiap kalimat
for i in range(len(lst_text)):
    lst_text[i] = re.sub(r'^A-Za-z+', ' ', lst_text[i])
    lst_summary[i] = re.sub(r'^A-Za-z+', ' ', lst_summary[i])

for i in range(len(test_text)):
    test_text[i] = re.sub(r'^A-Za-z+', ' ', test_text[i])

lst_text[0:5]

['bought several vitality canned dog food products found good quality product looks like stew process
ed meat smells better labrador finicky appreciates product better most ',
'product arrived labeled jumbo salted peanuts the peanuts actually small sized unsalted sure error v
endor intended represent product jumbo ',
'confection around centuries light pillowy citrus gelatin nuts case filberts cut tiny squares libera
lly coated powdered sugar tiny mouthful heaven chewy flavorful highly recommend yummy treat familiar
story c s lewis the lion witch wardrobe treat seduces edmund selling brother sisters witch ',
'looking secret ingredient robitussin believe found it got addition root beer extract ordered which
good made cherry soda flavor medicinal ',
'great taffy great price wide assortment yummy taffy delivery quick taffy lover deal ']
```

Gambar 17. Kode program remove special characters

3.4.5 Implementasi *Lemmatization*

Pada implementasi dilakukan lematisasi yang digunakan untuk mengubah kata yang memiliki imbuhan menjadi kata dasarnya.

```
#Lemmatization
df['Text'] = df['Text'].apply(lambda x: " ".join([Word(word).
lemmatize() for word in x.split()]))
df.Text.head()

0      bought several vitality canned dog food product found good
quality product look like stew processed meat smell better labrador finicky appreciates product bette
r
1      product arrived labeled jumbo sa
ltd peanutsthe peanut actually small sized unsalted sure error vendor intended represent product jum
bo
2      confection around century light pillowy citrus gelatin nut case filbert cut tiny square liberall
y coated powdered sugar tiny mouthful heaven chewy flavorful highly recommend yummy treat faniliar
s...
3      looking secret ingredient
robitussin believe found got addition root beer extract ordered good made cherry soda flavor medicina
l
4      great taffy great price wide assortment yummy taffy delivery quick taffy lover deal
Name: Text, dtype: object
```

Gambar 18. Kode program lematisasi

3.4.6 Implementasi *Stemming*

Proses *stemming* dilakukan untuk merubah kata-kata dalam dokumen menjadi kata dasarnya. Pada teks yang menggunakan bahasa inggris proses yang dilakukan hanya menghilangkan sufiks. Implemnetasi stemming pada proyek ini menggunakan stemming snowball.

#pada bagian ini akan dilakukan stem pada setiap kata.

```
from nltk.stem.snowball import SnowballStemmer
stemmer = SnowballStemmer("english")
for i in range(len(lst_text)):
    text_filtered = []
    summary_filtered = []
    text_word_tokens = []
    summary_word_tokens = []
    text_word_tokens = lst_text[i].split()
    summary_word_tokens = lst_summary[i].split()
    for r in text_word_tokens:
        text_filtered.append(str(stemmer.stem(r)))
    lst_text[i] = ' '.join(text_filtered)
    for r in summary_word_tokens:
        summary_filtered.append(str(stemmer.stem(r)))
    lst_summary[i] = ' '.join(summary_filtered)
```

```
for i in range(len(test_text)):
    text_filtered = []
    text_word_tokens = []
    text_word_tokens = test_text[i].split()
    for r in text_word_tokens:
        if not r in stop_words:
            text_filtered.append(str(stemmer.stem(r)))
    test_text[i] = ' '.join(text_filtered)
```

```
lst_text[0:5]
```

```
['bought sever vital can dog food product found good qualiti product look like
stew process meat smell better labrador finicki appreci product better most',
'product arriv label jumbo salt peanut the peanut actual small size unsalt sur
e error vendor intend repr product jumbo',
'confect around centuri light pillowi citrus gelatin nut case filbert cut tini
squar liber coat powder sugar tini mouth heaven chewi flavor high recommend yum
mi treat familiar stori c s lewi the lion witch wardrob treat seduc edmund sell
brother sister witch',
'look secret ingredi robitussin believ found it got addit root beer extract or
der which good made cherri soda flavor medicin',
```

Gambar 19. Kode program stemming

3.5 Implementasi Feature Extraction

Feature extraction merupakan sebuah proses mengekstrak informasi-informasi yang paling relevan untuk dikategorikan dengan meminimalkan variasi antar pola dalam suatu kategori dan sekaligus memaksimalkan variasi antar kategori yang berbeda dari data mentah yang tersedia. *Feature extraction* yang digunakan pada penelitian ini yaitu menggunakan metode *Bag of words*.

```
from sklearn.feature_extraction.text import CountVectorizer
vect = CountVectorizer()
X_train_dtm = vect.fit_transform(lst_text)
```

```
X_train_dtm
```

```
<368013x73328 sparse matrix of type '<class 'numpy.int64'>'
  with 12433945 stored elements in Compressed Sparse Row format>
```

Untuk membuat sebuah prediksi observasi baru harus memiliki fitur yang sama dengan observasi pelatihan.

```
# melakukan transformasi pada testing data (using fitted vocabulary) kedalam dokumen-term matriks
X_test_dtm = vect.transform(test_text)
X_test_dtm
```

```
<157801x73328 sparse matrix of type '<class 'numpy.int64'>'
  with 4913036 stored elements in Compressed Sparse Row format>
```

Gambar 20. Kode program *feature extraction*

Keterangan :

1. `vect.fit(lst_text)` belajar dengan kosakata dari training data
2. `vect.transform(lst_text)` belajar dengan kosakata yang pas untuk membangun document-term matrix dari training data
3. `vect.transform(test_text)` belajar dengan kosakata yang pas to build a document-term matrix dari testing data (dan mengacuhkan tokens yang tidak pernah dilihat sebelumnya).

3.6 Implementasi *Modelling*

Pembangunan sentimen analisis untuk *food review amazon* menggunakan *Naïve Bayes*.

Implementasi ini menggunakan *naïve bayes* dimana akan dilakukan import library `MultinomialNB` untuk mengimport library Multinomial Naïve Bayes.

Pada implementasi akan dilakukan pembuatan kelas prediksi untuk `X_test_dtm`

```
# train model menggunakan X_train_dtm
%time nb.fit(X_train_dtm, train.Score)
```

```
Wall time: 344 ms
```

```
MultinomialNB()
```

```
# dilakukan pembuatan kelas prediksi untuk X_test_dtm
y_pred_class_nb = nb.predict(X_test_dtm)
```

Gambar 21. Kode program implementasi klasifikasi

Pada implementasi akan dilakukan kalkulasi untuk melihat nilai akurasi dan akan dilakukan pencetakan *confusion matrix*.

```
# lakukan kalkulasi
from sklearn import metrics
metrics.accuracy_score(test.Score, y_pred_class_nb)

0.9075734627790698

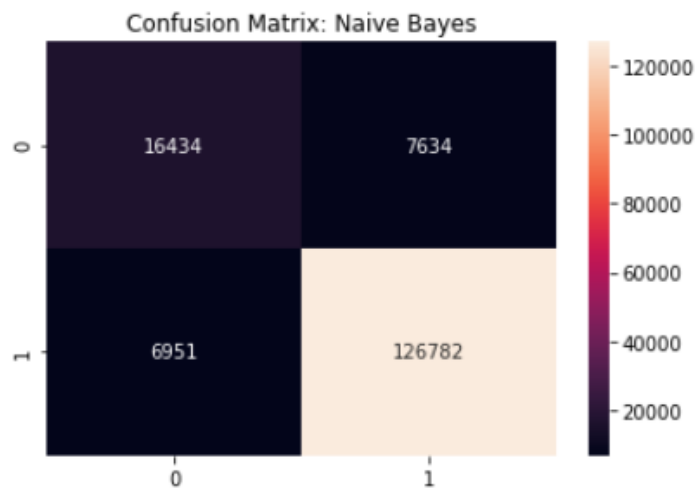
# lakukan pencetakan confusion matrix
con_metrics_nb = metrics.confusion_matrix(test.Score, y_pre
con_metrics_nb

array([[ 16434,   7634],
       [  6951, 126782]])
```

Gambar 22. Kode program kalkulasi akurasi

Implementasi ini menggunakan heatmap untuk *confusion matrix* dimana pembuatannya dapat dilihat pada implementasi dibawah ini.

```
#ploting heatmap untuk confusion matrix
import seaborn as sns
import matplotlib.pyplot as plt
sns.heatmap(con_metrics_nb, annot=True, fmt='d')
plt.title("Confusion Matrix: Naive Bayes")
plt.show()
```



Gambar 23. Implementasi heatmap

3.7 Implementasi Evaluation

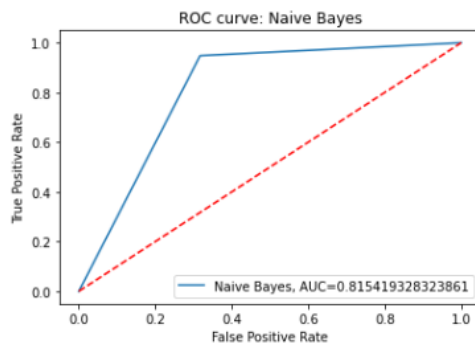
Evaluation atau pengujian dilakukan dengan membagi data ke dalam data *training* dan data *testing*. Metode pengujian merupakan penilaian seberapa baik proses identifikasi pada sistem dengan menggunakan rumus umum perhitungan *accuracy*, *precision*, *recall*, dan *F-score*.

```
#Lakukan pengecekan pada Precision, Recall and F1 Score
from sklearn.metrics import classification_report
print (classification_report(test.Score, y_pred_class_nb))
```

	precision	recall	f1-score	support
0	0.70	0.68	0.69	24068
1	0.94	0.95	0.95	133733
accuracy			0.91	157801
macro avg	0.82	0.82	0.82	157801
weighted avg	0.91	0.91	0.91	157801

Gambar 24. Kode program evaluasi

```
#Plotting Area Under the Curve
plt.plot(false_positive_rate,true_positive_rate,label="Naive Bayes, AUC="+str(auc_nb))
plt.plot([0,1],[0,1],'r--')
plt.title('ROC curve: Naive Bayes')
plt.legend(loc='lower right')
plt.ylabel('True Positive Rate')
plt.xlabel('False Positive Rate')
plt.show()
```



- *Accuracy* adalah tingkat kedekatan antara nilai prediksi dengan nilai aktual.
Rumus yang digunakan untuk perhitungan *accuracy*.
- *Precision* adalah tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem
- *Recall* adalah tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi.
Rumus yang digunakan untuk perhitungan *recall*
- *F1-Score* adalah *harmonic mean* dari *precision* dan *recall*.

BAB IV. PENUTUP

Pada bab ini dijelaskan mengenai pembagian tugas dan tanggung jawab, kesimpulan, dan saran pada kelanjutan pengembangan proyek yang dilakukan.

4.1 Kesimpulan

Dalam Analisis sentimen review restoran Fish Streat terdapat 2 proses, yaitu proses training dan proses testing. Proses training bertujuan untuk membangun model analisis sentimen dari data latih dengan mengitung probabilitas. Proses testing bertujuan untuk menentukan klasifikasi sentimen terhadap data yang akan di uji cobakan. Pengukuran akurasi dari sistem yang dibuat menggunakan confusion matrix. Berdasarkan hasil implementasi dan pengujian yang telah dilakukan pada analisis sentiment didapatkan tingkat akurasi sebesar 0,91%. Berdasarkan penelitian yang telah dilakukan, dalam analisis sentimen penggunaan metode Naïve Bayes Classifier akan menimbulkan tingkat akurasi yang tinggi. Semakin banyak data yang diujikan maka tingkat akurasi dari hasil klasifikasi akan semakin tinggi. Hal ini juga dipengaruhi oleh jumlah data latih yang digunakan pada metode Naïve Bayes Classifier. Kedepannya diharapkan dalam penelitian ini dapat menggunakan metode lain sebagai pembanding dari metode Naïve Bayes Classifier.

4.2 Saran

Berdasarkan pengerjaan proyek yang telah dilakukan, masih terdapat beberapa kekurangan yang perlu diperbaiki. Saran yang diberikan adalah dengan menyediakan antarmuka pengguna agar dapat digunakan oleh perusahaan untuk memudahkan memahami informasi yang diberikan oleh perusahaan atau organisasi.

4.3 Pembagian Tugas

Pada subab ini akan dijelaskan mengenai pembagian tugas tiap anggota kelompok.

Tabel 2. Pembagian Tugas

Team Members	Roles	Task
Reza Oktovian Siregar	Data Engineer	Bertanggung jawab untuk membersihkan, memproses data yang sudah dikumpulkan.
Angelia Regina Ginting	Data Engineer	Bertanggung jawab untuk membersihkan, memproses data yang sudah dikumpulkan.
Christina Clara	Team Leader	Bertanggung jawab untuk membersihkan, memproses data yang sudah dikumpulkan.

Referensi

- [1] S. K. Hendry Ardian, "Analisis Sentimen Pada Review Produk Kosmetik Bahasa Indonesia Dengan Metode Naive Bayes," Jurnal ENTER, Pontianak.
- [2] K. A. Mohamad Syahrul Mubarak, "Aspect-based sentiment analysis to review products using Naïve Bayes, Muhammad Dwi Aldhi," International Conference on Mathematics: Pure, Applied and Computation, 2017.
- [3] R. G. M. D. M. H. Lei Zhang, "Combining Lexicon-based and Learning-based Methods for Twitter Sentiment Analysis," Hewlett-Packard Development Company, Chicago, 2011.
- [4] L. K. O. S. Ni Putu Sri Merta Suryani, Penggunaan Metode Naïve Bayes Classifier pada Analisis Sentimen Facebook Berbahasa Indonesia, Majalah Ilmiah Teknologi Elektro, 2019.
- [5] E. Junianto and R. Rachman, "Penerapan Metode Naive Bayes Classifier Untuk Mendeteksi Emosi pada Komentar Media Sosial," *Jurnal Responsif*, vol. 2, no. 1, pp. 1-8, 2020.
- [6] A. P. Wijaya and H. A. Santoso, "Naive Bayes Classification pada Klasifikasi Dokumen Untuk Identifikasi Konten E-Government," *Journal of Applied Intelligent System*, vol. 1, no. 1, pp. 48-55, 2016.
- [7] S. M. Derwin Suhartono, "Natural Language Processing," Binus University, [Online]. Available: [https://socs.binus.ac.id/2013/06/22/natural-language-processing/#:~:text=Ingason%20dkk.%20\(2008\)%20mengemukakan,bentuk%20dasar%20dari%20sebuah%20kata.&text=Lemma%20adalah%20bentuk%20dasar%20dari,arti%20tertentu%20berdasar%20pada%20kamus..](https://socs.binus.ac.id/2013/06/22/natural-language-processing/#:~:text=Ingason%20dkk.%20(2008)%20mengemukakan,bentuk%20dasar%20dari%20sebuah%20kata.&text=Lemma%20adalah%20bentuk%20dasar%20dari,arti%20tertentu%20berdasar%20pada%20kamus..) [Accessed 08 Januari 2020].
- [8] I. M. Yulietha, S. A. Faraby and Adiwijaya, "Klasifikasi Sentimen Review Film Menggunakan Algoritma Support Vector Machine," in *e-Proceeding of Engineering*, Bandung, 2017.
- [9] E. P. P. D. Puspitaningrum and A. Mirfen, "Identifikasi Tanda Tangan dengan Pendekatan Support Vector Machine," *Jurnal Sains, Teknologi dan Industri*, vol. 12, no. 2, pp. 225-231, 2015.
- [10] F. Z. Alif, *Ekstraksi Fitur untuk Pemilihan Topik Spesifik Review Film dalam Menghasilkan Aspect-Based Sentiment Analysis*, Medan, 2020.
- [11] R. Putra, "PENERAPAN ASSOCIATION RULE DENGAN ALGORITMA A-PRIORI UNTUK SISTEM REKOMENDASI PADA PERUSAHAAN E-COMMERCE X," UNIVERSITAS KATOLIK PARAHYANGAN, Bandung, 2018.
- [12] W. A. Triyanto, "ASSOCIATION RULE MINING UNTUK PENENTUAN REKOMENDASI PROMOSI PRODUK," Universitas Muria Kudus, 2014.
- [13] R. A. H. A. M. Nugroho Wand, "Pengembangan Sistem Rekomendasi Penelusuran Buku Dengan Penggalan Association Rule Menggunakan Algoritma Apriori (Studi Kasus Badan Perpustakaan

- Dan Kearsipan Provinsi Jawa Timur)," Institut Teknologi Sepuluh Nopember (ITS), Surabaya, 2012.
- [14] S. Mujilahwati, "PRE-PROCESSING TEXT MINING PADA DATA TWITTER," *SENTIKA 2016*, pp. 18-19, 2016.
- [15] A. . M. Pravina, . I. Cholissodin and . P. . P. Adikara, "Analisis Sentimen Tentang Opini Maskapai Penerbangan pada Dokumen Twitter Menggunakan Algoritme Support Vector Machine (SVM)," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. III, no. 3, p. 4, 2019.
- [16] R. M. Reese, in *Natural Language Processing with Java*, 2015.
- [17] W. . A. Luqyana, . I. Cholissodin and . R. . S. Perdana, "Analisis Sentimen Cyberbullying pada Komentar Instagram dengan Metode Klasifikasi Support Vector Machine," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. II, no. 11, p. 4, 2018.
- [18] P. Bajaj, "Creating linear kernel SVM in Python," GeeksforGeeks, 20 June 2018. [Online]. Available: <https://www.geeksforgeeks.org/creating-linear-kernel-svm-in-python/>. [Accessed 22 November 2020].
- [19] F. Rahutomo, D. S. E. Ikawati and O. A. Rohman, "Evaluasi Fitur Word2Vec pada Sistem Ujian Esai Online," *JUPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 04, no. 01, pp. 36-45, 2019.
- [20] A. F. Niasita, P. P. Adikara and S. Adinugroho, "Analisis Sentimen Pembangunan Infrastruktur di Indonesia dengan Automated Lexicon Word2Vec dan Naive-Bayes," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, vol. 3, no. 3, pp. 2673-2679, 2019.