

BAB 4

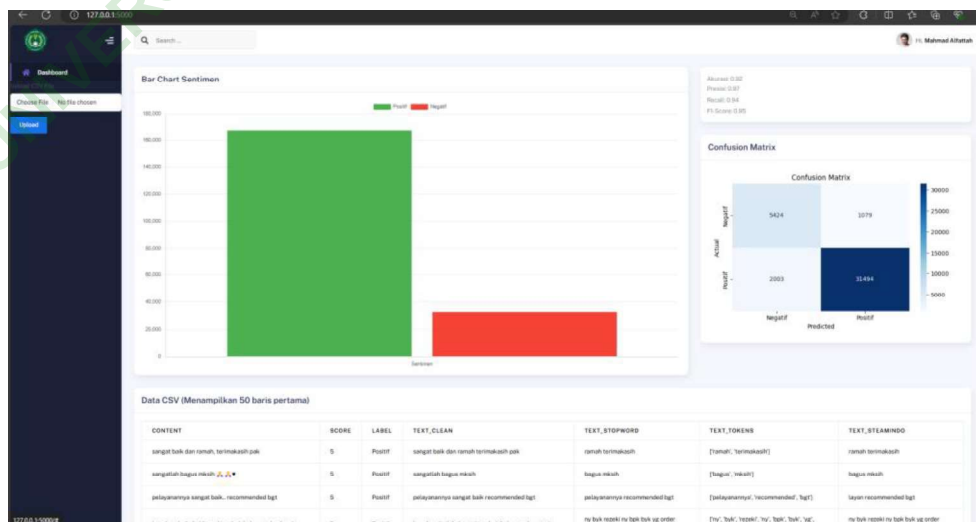
HASIL PENELITIAN

4.1 RINGKASAN HASIL PENELITIAN

Penelitian analisis sentimen terhadap ulasan pengguna aplikasi Maxim di Google Play Store, yang dilakukan dengan metode Naïve Bayes Classification, berhasil mengumpulkan 200.000 data ulasan pengguna. Data tersebut diperoleh dengan menggunakan Google Play Scraper. Pelabelan menggunakan model Bidirectional Encoder Representations from Transformers (BERT) berhasil mendapatkan hasil sentimen positif sebanyak 167.213 data dan sentimen negatif sebanyak 32.787 data.

4.2 IMPLEMENTASI DESAIN ANTARMUKA

Tampilan utama aplikasi, yang dapat dilihat pada Gambar 4.1, menunjukkan sebuah dashboard yang menyajikan hasil analisis sentimen dari ulasan pengguna di Google Play Store terhadap aplikasi Maxim. Pada halaman tersebut terdapat fitur untuk mengunggah file berformat csv yang telah melalui tahap pelabelan. Sistem akan menampilkan hasil visualisasi analisis sentimen dari data yang diunggah serta akan memberikan hasil evaluasi meliputi *accuracy*, *precision*, *recall*, *f1 score*, serta *confusion matrix*.



Gambar 4.1 Visualisasi Analisis Sentimen

4.3 PEMBAHASAN

Setelah di dapatkan 200.000 data serta dilakukan pengolahan data hingga diperoleh hasil sentimen positif sebanyak 167.213 data dan sentimen negatif sebanyak 32.787 data, selanjutnya adalah melakukan training dengan metode Naïve bayes Classification.

4.3.1 Training

Training merupakan suatu proses untuk membangun sebuah model klasifikasi dimana model tersebut dapat secara otomatis digunakan dalam proses klasifikasi data. Pada penelitian ini digunakan metode Naive Bayes Classification dalam melakukan *training data*. Pada tahap training ini digunakan komposisi 80% data training dan 20% data testing berikut kodenya:

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(
    data_clean['content'], data_clean['Label'],
    test_size = 0.20,
    random_state = 0)
```

Proses Splitting menghasilkan data yang ditunjukkan pada Tabel 4.1.

Tabel 4.1 Hasil *Splitting* Data

Variabel	Jumlah
X_Train	160000
Y_Train	160000
X_Test	40000
Y_Test	40000

4.3.2 TF-IDF

Term *frequency-inverse document frequency* atau TF-IDF adalah salah satu metode yang paling umum digunakan dalam pengolahan teks untuk menilai pentingnya suatu kata dalam sebuah dokumen. Tahap ini dilakukan pembobotan

disetiap kata dan pemecahan kata, berikut proses yang dilakukan dalam TF-IDF. Dokumen yang digunakan dapat dilihat pada Tabel 4.2

Tabel 4.2 Data TF-IDF

No	Dokumen
doc1	ramah terimakasih
doc2	bagus mksih
doc3	layan recommended bgt
doc4	layan bagus
doc5	mantap

Tabel 4.2 menunjukkan data yang digunakan untuk menghitung TF-IDF secara manual, dengan melibatkan 5 dokumen, yaitu doc1, doc2, doc3, doc4, dan doc5. Perhitungan TF memerlukan beberapa komponen, seperti kata, doc (dokumen), dan jumlah dokumen yang mengandung kata tersebut. Hasil dari perhitungan TF ini dapat dilihat pada Tabel 4.3.

Tabel 4.3 Perhitungan TF

Kata	doc1	doc2	doc3	doc4	doc5	Jumlah Dokumen
ramah	1					1
terimakasih	1					1
bagus		1		1		2
mksih		1				1
layan			1	1		2
recommended			1			1
bgt			1			1
mantap					1	1

Tabel 4.3 menguraikan kata-kata yang ada dalam dokumen. Perhitungan pada IDF ini memerlukan beberapa komponen, diantaranya yaitu kata, jumlah

dokumen yang mengandung kata tersebut), dan N (total jumlah dokumen). Hasil dari perhitungan IDF dapat dilihat pada Tabel 4.4

Tabel 4.4 Perhitungan IDF

Kata	Jumlah Dokumen	Idf	idf(N=5)
ramah	1	1	0,69897
terimakasih	1	1	0,69897
bagus	2	0,5	0,39794
mksih	1	1	0,69897
layan	2	0,5	0,39794
recomended	1	1	0,69897
bgt	1	1	0,69897
mantap	1	1	0,69897

Tabel 4.4 menunjukkan perhitungan IDF menggunakan rumus $idf = (\frac{N}{df})$ yang dilakukan secara manual. Untuk menghitung nilai TF-IDF ditunjukkan pada Tabel 4.5

Tabel 4.5 Perhitungan TF-IDF

Kata	doc1	doc2	doc3	doc4	doc5
ramah	0,69897				
terimakasih	0,69897				
bagus		0,39794		0,39794	
mksih		0,69897			
layan			0,39794	0,39794	
recomended			0,69897		
bgt			0,69897		
mantap					0,69897

Tabel 4.5 menunjukkan hasil perhitungan secara manual dengan mengalikan TF dan IDF. Dalam penelitian ini, klasifikasi dilakukan dengan menggunakan ekstraksi TF dan IDF yang memungkinkan dilakukan secara

otomatis pada dokumen *training*. Penghitungan otomatis ini dilakukan dengan bantuan *sklearn* dalam bahasa pemrograman Python, khususnya dengan menggunakan *TfidfVectorizer*.

Kemudian dilakukan pelatihan dengan model Naïve Bayes Classification pada data. Berikut cuplikan kodenya:

```
from sklearn.naive_bayes import MultinomialNB

nb = MultinomialNB()
nb.fit(x_train_res, y_train_res)
nb.score(x_train_res, y_train_res)

x_test_vect = vect.transform(x_test)
y_pred = nb.predict(x_test_vect)

y_pred
```

Hasil dari *y_pred* dapat digunakan untuk menghitung berbagai metrik evaluasi, seperti akurasi, presisi, recall, dan F1-score, yang berguna untuk menilai kinerja model pada data uji. Dapat dilihat pada Gambar 4.2.

```
array([-1,  1,  1, -1,  1, -1,  1,  1,  1,  1, -1, -1, -1,  1,  1,  1, -1,
       -1, -1,  1, -1,  1, -1, -1,  1,  1,  1, -1,  1,  1, -1,  1, -1,
        1,  1,  1, -1, -1,  1, -1, -1, -1,  1, -1, -1,  1, -1, -1,  1, -1,
        1,  1,  1,  1, -1,  1, -1, -1,  1,  1, -1,  1, -1, -1, -1, -1,
       -1,  1, -1,  1, -1,  1, -1,  1, -1,  1,  1, -1, -1, -1, -1,  1,
       -1, -1, -1,  1,  1,  1, -1,  1,  1,  1, -1, -1, -1, -1,  1, -1, -1,
       -1, -1,  1,  1, -1,  1, -1,  1,  1,  1, -1, -1, -1, -1, -1,  1, -1,
        1, -1,  1,  1, -1,  1,  1, -1, -1,  1,  1, -1, -1,  1, -1,  1, -1,
       -1, -1,  1])
```

Gambar 4.2 Hasil Pelatihan Model

4.3.3 Testing

Pada tahap ini, dilakukan evaluasi untuk mengukur akurasi pemodelan yang dibangun pada tahap pelatihan data. Pengujian ini menggunakan confusion matrix yang memungkinkan perhitungan akurasi, presisi, recall, dan F1-score.

1. Akurasi, mengukur proporsi dari prediksi yang benar terhadap total prediksi yang dibuat.
2. *Precision* mengukur proporsi dari prediksi positif yang benar terhadap seluruh prediksi positif.

3. *Recall* mengukur proporsi dari prediksi positif yang benar terhadap seluruh kasus positif yang sebenarnya.
4. F1-Score adalah rata-rata harmonis dari precision dan recall, memberikan keseimbangan antara keduanya.

Perhitungan testing dapat dilihat pada cuplikan kode berikut:

```
clf = MultinomialNB()
clf.fit(X_train, y_train)
predicted = clf.predict(X_test)
print("MultinomialNB Accuracy:",
      accuracy_score(y_test, predicted))
print("MultinomialNB Precision:",
      precision_score(y_test, predicted,
                      average="binary", pos_label="Negatif"))
print("MultinomialNB Recall:",
      recall_score(y_test, predicted,
                   average="binary", pos_label="Negatif"))
print("MultinomialNB f1_score:", f1_score(y_test,
                                           predicted, average="binary", pos_label="Negatif"))
```

Hasil dari pengujian data menggunakan model Naïve Bayes classification dengan `x_train` sebanyak 160000, `y_train` sebanyak 160000, `x_test` sebanyak 40000, serta `y_test` sebanyak 40000. Dari 40000 yang digunakan untuk pengujian di dapatkan data Negatif 6604 dengan *precision* 74% yang artinya dari semua data negatif data yang digunakan terdapat 74% benar-benar negatif. Data Positif 33396 dengan *precision* 0.94 yang artinya dari semua data positif 94% data yang benar-benar positif dari semua data positif.

Dari data pengujian didapatkan pada kelas negatif sebesar 85% dan kelas positif sebesar 94%. Untuk F1-Score pada kelas negatif sebesar 79% yang menunjukkan keseimbangan antara presisi dan recall pada kelas negatif, sedangkan untuk kelas positif F1-Score sebesar 96% yang menunjukkan model ini baik dalam mengidentifikasi kelas positif. Pada *Support* ini adalah jumlah data dalam kelas Negatif dan Positif. Dari proses pengujian yang sudah dilakukan didapatkan hasil akurasi pemodelan sebesar 93%, untuk lebih jelasnya dapat dilihat pada Gambar 4.4.

```

confusion_matrix:
[[ 5583  1021]
 [ 1927 31469]]
=====

              precision    recall  f1-score   support

   Negatif         0.74         0.85         0.79         6604
   Positif         0.97         0.94         0.96        33396

 accuracy              0.93         40000
 macro avg         0.86         0.89         0.87         40000
 weighted avg      0.93         0.93         0.93         40000

```

Gambar 4.3 Hasil Akurasi Testing

Kemudian dilakukan visualisasi confusion matrix yang dapat dilihat pada Tabel 4.6.

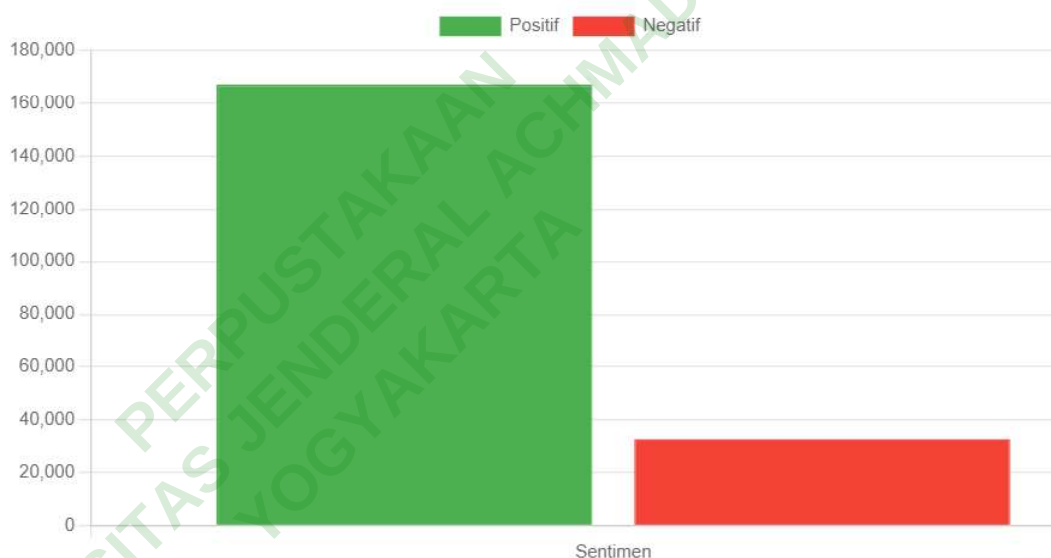
Tabel 4.6 Confusion Matriks

Actual	Prediksi	
	Positif	Negatif
Positif	31469	1927
Negatif	1021	5583

Confusion Matriks ini menunjukkan empat elemen penting. True Negatives (TN) sebesar 5583 mencerminkan jumlah contoh yang sebenarnya negatif dan berhasil diprediksi sebagai negatif oleh model. False Positives (FP) sebesar 1021 menunjukkan jumlah contoh yang sebenarnya negatif namun diprediksi sebagai positif oleh model, yang merupakan kesalahan tipe I. False Negatives (FN) sebesar 1927 menggambarkan jumlah contoh yang sebenarnya positif tetapi diprediksi sebagai negatif oleh model, yang merupakan kesalahan tipe II. Terakhir, True Positives (TP) sebesar 31.469 mencerminkan jumlah contoh yang benar-benar positif dan berhasil diprediksi sebagai positif oleh model. Elemen-elemen ini memberikan gambaran yang jelas mengenai kekuatan dan kelemahan model dalam melakukan klasifikasi serta jenis kesalahan yang mungkin terjadi.

4.4 HASIL ANALISIS SENTIMEN

Pada hasil analisis yaitu menghitung jumlah data keseluruhan data ulasan yang sudah terambil dengan total 200.000 data, kemudian dilakukan pelabelan secara otomatis dengan pemodelan BERT. Dari jumlah total 200.000 data dibagi menjadi 2 bagian untuk dilakukan training dan testing dengan proporsi 20% untuk testing yaitu sejumlah 40.000 data dan 80% untuk training yaitu sejumlah 160.000 data. Dari proses testing telah menghasilkan nilai akurasi yang cukup tinggi yaitu sebesar 93%. Hasil klasifikasi pada data ulasan keseluruhan yang sudah didapatkan dengan nilai positif dan negatif dapat dilihat pada Gambar 4.4



Gambar 4.4 Histogram Data Positif dan Negatif

Pada Gambar 4.1 telah di dapatkan hasil histogram dengan jumlah 167.213 Label Positif dan 32.787 Label Negatif. Dengan demikian hasil analisis sentimen mengenai ulasan pada aplikasi maxim berbanding lurus dengan nilai bintang yang tinggi yaitu 4,8 dari 5 bintang pada Google Play Store. Sehingga rating yang tinggi sesuai dengan sentimen pada ulasan yang diberikan pengguna Maxim