

Klasifikasi Sentimen Terhadap Kualitas Aplikasi Bahan Ajar Digital Akademik Universitas Terbuka di Google Play

Rhini Fatmasari¹, Windu Gata², Nia Kusuma Wardhani³, *Kurnia Prayogi⁴, Modesta Binti Husna⁵

1) Fakultas Keguruan dan Ilmu Pendidikan, Universitas Terbuka

Jalan Cabe Raya, Pondok Cabe, Pamulang, Tangerang Selatan 15437

3) Fakultas Ekonomi dan Bisnis, Universitas Nusa Mandiri

Jl. Meruya Selatan No.1, Meruya Sel., Kembangan, Kota Jakarta Barat 11650

2,4,5) Ilmu Komputer, Fakultas Teknologi Informasi, Universitas Nusa Mandiri

Jl. Margonda No.545, Pondok Cina, Kecamatan Beji, Kota Depok, Jawa Barat 16424

Email: ¹riens@ecampus.ut.ac.id, ²windu@nusamandiri.ac.id, ³Nia.kusuma@mercubuana.ac.id, ⁴kurniaaprayogii1@gmail.com,

⁵desta001007@gmail.com

ABSTRACT

Terbuka University is a leading institution that implements the optimization of digital transformation, especially in distance learning systems. Terbuka University has developed the Terbuka University Digital Learning Materials application. This application offers several learning modules that can be accessed through the Google Play Store. This research aims to classify data using different labels related to reviews of the Terbuka University Digital Learning Materials application using the Long Short-Term Memory classification algorithm. Evaluation is conducted to find accuracy, f1-score, precision, and recall values. The research results show that classification with Long Short-Term Memory achieves an accuracy of 76.72% with the Vader label, and the accuracy with the TextBlob label reaches 74.21%. Confusion matrix evaluation shows precision results of 0.91 and recall of 0.78, with an f1-score of 0.84 for the Vader label. For the TextBlob label, the precision is 0.96, recall is 0.45, and the f1-score is 0.61. This research contributes positively to understanding the evaluation and classification of reviews of the Terbuka University Digital Learning application. Implementing the Long Short-Term Memory algorithm with the Vader label can be an effective choice to improve service and learning quality through the application.

Keywords: *terbuka university; long short-term memory; google play; vader; textblob*

ABSTRAK

Universitas Terbuka merupakan perguruan tinggi terkemuka yang menerapkan optimalisasi transformasi digitalisasi, khususnya dalam sistem belajar jarak jauh. Universitas Terbuka mengembangkan aplikasi Bahan Ajar Digital Universitas Terbuka. Aplikasi ini menyajikan beberapa modul pembelajaran yang dapat diakses melalui *Google Play Store*. Penelitian ini bertujuan untuk melakukan klasifikasi data menggunakan label yang berbeda terkait ulasan aplikasi Bahan Ajar Digital Universitas Terbuka yang menggunakan algoritma klasifikasi deep learning, yaitu *Long Short-Term Memory*. Evaluasi dilakukan untuk mencari nilai accuracy, f1-score, precision, dan recall. Hasil penelitian menunjukkan bahwa klasifikasi dengan *Long Short-Term Memory* memberikan performa dengan nilai accuracy sebesar 76.72% dengan label Vader, dan hasil accuracy dengan label TextBlob mencapai 74.21%. Evaluasi confusion matrix menunjukkan hasil precision sebesar 0.91, recall 0.78, dan f1-score 0.84 pada label Vader. Pada label TextBlob, nilai precision mencapai 0.96, recall sebesar 0.45, dan f1-score sebesar 0.61. Penelitian ini memberikan kontribusi positif terhadap pemahaman tentang evaluasi dan klasifikasi ulasan aplikasi Bahan Ajar Digital Universitas Terbuka. Implementasi algoritma *Long Short-Term Memory* dengan label Vader dapat menjadi pilihan yang efektif untuk meningkatkan pelayanan dan kualitas belajar melalui aplikasi tersebut.

Kata Kunci: universitas terbuka; long short-term memory; google play; vader; textblob

Diterima Redaksi: 1-1-2024 | Selesai Revisi: 17-3-2024 | Diterbitkan Online: 31-3-2024

1. PENDAHULUAN

Bahan Ajar Digital Akademik merupakan aplikasi pelayanan yang dikeluarkan oleh Universitas Terbuka untuk memenuhi kebutuhan mahasiswa dalam mendukung pembelajaran yang terdapat beberapa modul yang bisa dipelajari dimana dan kapan saja. Universitas Terbuka adalah lembaga pendidikan tinggi yang menekankan pada pembelajaran mandiri dan inisiatif. Proses pembelajaran dilakukan baik secara individu maupun dalam kelompok, dengan sumber belajar yang mencakup perpustakaan, panduan dari internet, siaran radio, televisi, tutorial komputer, serta program audio atau video (Fatmasari et al., 2023).

Aplikasi Bahan Ajar Digital Akademik Universitas Terbuka juga tak luput dari kekurangan, dengan rating 3.1 di *Google Play Store* membuat Universitas Terbuka berusaha memperbaiki kualitas aplikasi demi mendukung pelayanan terbaik kepada mahasiswa. *Google Play Store* adalah platform digital yang memungkinkan pengguna Android untuk mengakses dan smartphone mengunduh berbagai aplikasi serta produk digital lainnya, termasuk e-book, film, permainan, dan lain-lain, baik yang gratis maupun yang

berbayar (Aditiya et al., 2022). Dengan rating tersebut yang disertai dari beberapa kumpulan ulasan negatif dan positif yang mengartikan bahwa pelayanan yang diberikan oleh Universitas Terbuka belum sepenuhnya memenuhi harapan dari pengguna aplikasi tersebut. Oleh karena itu, upaya dalam mengoptimalkan aplikasi Bahan Ajar Digital Akademik Universitas Terbuka dapat diketahui dengan melakukan klasifikasi menggunakan metode klasifikasi *Long Short-Term Memory*, dalam bentuk analisis sentimen ulasan dari pengguna aplikasi di *Google Play Store*. Analisis sentimen merupakan suatu proses pengolahan atau klasifikasi sebuah kata atau text yang bersumber dari internet atau media sosial dengan tujuan mendapatkan respon atau opini masyarakat terkait media sosial (Kaburuan & Setiawan, 2023).

Penelitian ini juga mengambil dari beberapa penelitian terdahulu yang menggunakan algoritma klasifikasi *Long Short-Term Memory* seperti pada penelitian (Sri Widagdo et al., n.d.), dimana penelitian ini menggunakan metode klasifikasi *Long Short-Term Memory* dengan data testing 1640 data yang menghasilkan nilai akurasi 63%, semuanya di atas 60% dan nilai roc_auc

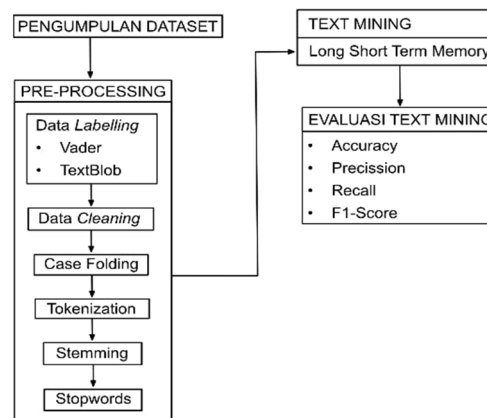
sebesar 81%. Berikutnya diambil dari penelitian (Dwi et al., n.d.), dimana pada penelitian ini juga menggunakan metode *Long Short-Term Memory*, dengan pengujian pertama menggunakan data sebanyak 3.184 kalimat mendapatkan nilai f1-Score pada kelas 1 sebesar 0.9331. Pada pengujian kedua terdapat penambahan data sebanyak 894 kalimat sehingga total data menjadi 4.078 kalimat dengan rata-rata f1-score 0.8181.

2. METODE

Pada penelitian ini akan menggunakan metode *kuantitatif* dimana metode tersebut menggunakan *filsafat positivisme*, untuk meneliti populasi atau sampel tertentu dengan pengumpulan data menggunakan instrumen penelitian, analisis data bersifat kuantitatif (Riany & Testiana, 2023). Dalam penelitian ini akan menerapkan metode kuantitatif yang memanfaatkan fungsi dari klasifikasi *Deep Learning Long Short-Term Memory* (LSTM) yang bertujuan untuk mengklasifikasikan data ulasan dari pengguna aplikasi Bahan Ajar Digital Universitas Terbuka dengan mencari nilai yang mengacu pada *accuracy*, *recall*, *f1-score*, dan *precision* sehingga dapat menarik kesimpulan

bagaimana Universitas Terbuka dalam meningkatkan aplikasi Bahan Ajar Digital bagi mahasiswa.

Penelitian ini terdapat beberapa tahapan, adapun tahapan tersebut dapat dilihat pada Gambar 1 berikut ini.



Gambar 1. Tahapan Penelitian

2.1 Pengumpulan Dataset

Pengumpulan data dilakukan dengan *Web Scraping* dengan cara *crawling data* menggunakan *library google-play-scraper*. Teknik ini dapat menemukan data komentar dari pengguna aplikasi Bahan Ajar Digital Akademik Universitas Terbuka secara otomatis.

Tabel 1. Hasil Seleksi Dataset Ulasan Aplikasi Bahan Ajar Digital Universitas Terbuka

	userName	score	at	content
0	reda D	2	09/11/2023 21:16:18	tolong di update lagi, masa udah ganti semester modul yang semester baru ga ada, adanya cuma modul di semester sebelumnya

1	Friska Dwi	1	03/11/2023 20:08:59	Ayo dong ditingkatin, udah lama gabisa buku ba digital loading terus.
...
800	Pengguna Google	4	24/02/2017 18:23:11	Lapor , 1. Saat selesai download BA, aplikasi force close . 2. Ada jeda yg agak lama saat berpindah halaman . 3. BA yg sudah di download tidak tersimpan di penyimpanan telepon. 4. Saat membaca apakah, harus selalu terhubung ke internet ? . Laporan selesai , terima kasih !

2.2 Pre-Processing

Pre-processing data merupakan tahapan penting dalam persiapan data untuk analisis. Proses ini dilakukan untuk mengubah data mentah menjadi data yang siap digunakan dalam analisis, sehingga dapat memperoleh hasil analisis yang akurat dan relevan (Khoiruddin et al., 2023). Dengan tahapan tersebut dibutuhkan tahapan pendukung seperti *Text mining* dimana menganalisis pencarian sebuah informasi yang dianggap penting untuk sebuah hasil spesifik. Dengan prinsip dan cara data mining dilakukan agar mengetahui pola yang ada di suatu teks (Aditiya et al., 2022).

Berdasarkan data yang sudah didapatkan, *pre-processing* memiliki beberapa tahapan seperti :

a. Data Selection

1) Translate Data

Tahapan ini dilakukan untuk mempermudah saat memberikan label pada data. Pada python terdapat suatu library dari google translate yaitu google trans. (Fatariska, 2023) *Google translate* merupakan Layanan terjemahan yang paling sering digunakan oleh penutur bahasa Indonesia, terutama pada terjemahan bahasa Inggris-Indonesia.

2) Labelling Data

Tahapan ini akan menggunakan dua model Teknik analisis yaitu label *Vader* dan label *TextBlob*. *Vader* (*Valence Aware Dictionary Sentiment Reasoner*) adalah alat analisis sentimen berbasis leksikon dan aturan (*lexicon- and rule-based*) yang secara khusus disesuaikan dengan sentimen yang diungkapkan di media sosial (Taufiq Anwar et al., 2023). Sedangkan untuk *TextBlob* adalah *library python* untuk memproses data tekstual. *Textblob* memiliki fitur-fitur dasar *Natural Language*

Processing (NLP), dalam penelitian kali ini fitur analisis sentiment digunakan untuk melakukan klasifikasi terhadap data testing (Lestari et al., 2020).

b. *Data Cleaning*

Tahap *Cleaning* bertujuan untuk mengurangi *noise* atau gangguan dalam data dengan menghilangkan kata-kata yang tidak relevan dan membersihkannya. contohnya alamat URL, tanda unik seperti pagar (#), nama pengguna (@username), tanda baca, dan alamat email (Kusuma & Nurramdhani Irmanda, 2022).

c. *Case Folding*

Setelah menyelesaikan tahapan *cleaning*, penulis melanjutkan ke tahap *case folding*. Pada langkah selanjutnya mengubah semua teks dalam data ke huruf kecil atau *lowercase*.

d. *Tokenization*

Tahap *tokenization*, juga dapat disebut sebagai *Term Frequency - Inverse Document Frequency* (TF-IDF), sebuah metode untuk mengubah data teks menjadi data numerik yang kemudian memberikan bobot pada setiap *term* atau kata (Husnina et al., 2023).

e. *Stemming*

Menyingkirkan imbuhan dari setiap kata yang akan menjadi istilah pokok, dengan maksud mengurangi penggunaan kata yang kurang tepat dalam kalimat (Nurhafida & Sembiring, 2022).

f. *Stopwords*

Pada tahap ini, kata-kata yang tidak memberikan kontribusi pada makna kalimat dihilangkan tanpa mengurangi informasi yang disampaikan oleh kalimat tersebut (Kulsum et al., 2022).

2.3 Long Short-Term Memory

Long Short-Term Memory (LSTM) merupakan sebuah evolusi dari arsitektur RNN, dimana pertama kali diperkenalkan oleh Hochreiter & Schmidhuber (1997). Hingga penelitian ini dilakukan banyak para peneliti yang terus mengembangkan arsitektur LSTM di berbagai bidang seperti dalam bidang speech recognition dan forecasting (ArFan & Lussiana ETP, 2019).

Pada penelitian (Tampubolon et al., 2023), Long Short-Term Memory (LSTM) merupakan algoritma Deep Learning yang populer untuk melakukan prediksi dan klasifikasi yang berkorelasi dengan waktu. Penggunaan metode algoritma ini dapat menyimpan data

informasi dalam jangka waktu yang panjang, selain itu juga dapat digunakan untuk mengolah, meramalkan, dan mengklasifikasikan suatu informasi berdasarkan deret waktu.

Pada *Long Short-Term Memory* (LSTM), terdapat kekhasan cell yang dikonfigurasi oleh empat buah gerbang utama: *input gate*, *input modulation gate*, *forget gate*, dan *output gate*. *Input gate* mengambil poin input baru dari luar dan proses data yang baru. *Memory cell input gate* mengambil input dari *output cell* LSTM pada iterasi terakhir (Anisa et al., 2023).

Suatu keadaan tersembunyi dari *memory cells* ditunjukkan oleh perhitungan formula berikut (Anisa et al., 2023)

Tabel 2. Formula LSTM (*Gate*)

Input	=	$\sigma(W_{ix}x_t + W_{ih}h_{t-1} + W_{ic}c_{t-1} + b_i)$
Forget	=	$\sigma(W_{fx}x_t + W_{fh}h_{t-1} + W_{fc}c_{t-1} + b_f)$
Output	=	$\sigma(W_{ox}x_t + W_{oh}h_{t-1} + W_{oc}c_{t-1} + b_o)$
Memory Cell (c_t)	=	$f_t * c_{t-1} + i_t * g(W_{cx}x_t + W_{ch}h_{t-1} + W_{cc}c_{t-1} + b_c)$

2.4 Evaluasi Model

Hasil dari pengujian akan dievaluasi untuk menguji performa kinerja dari model yang terbentuk menggunakan algoritma *Long Short Term Memory* dengan pengujian menghitung nilai dari *accuracy*, *precision*, *recall*, dan *specificity* pada *confusion matrix*.

- TP (*True Positive*) merupakan jumlah data yang bernilai positif dan diprediksi benar.
- TN (*True Negative*) merupakan jumlah data yang bernilai negatif dan diprediksi benar.
- FP (*False Positive*) merupakan data yang diprediksi memiliki hasil positif, akan tetapi kenyataannya negatif.
- FN (*False Negative*) merupakan jumlah data yang diprediksi memiliki hasil negatif, akan tetapi kenyataannya positif.

Tabel 3. Evaluasi Model Confusion Matrix

		Actual	
		Positive	Negative
Prediction	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

Dengan menggunakan *confusion matrix*, kita dapat menghitung nilai *accuracy*, *precision*, *recall*, dan *specificity* menggunakan persamaan (1), (2), (3), (4).

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

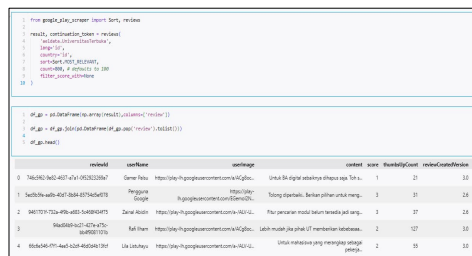
$$Specificity = \frac{TN}{TN + FP} \quad (4)$$

Accuracy adalah hasil dari perbandingan prediksi yang benar dengan total data yang ada, sementara itu untuk *precisson* adalah perbandingan prediksi yang benar positif dengan total prediksi positif yang dibuat. *Recall* adalah perbandingan prediksi yang benar positif dengan total data aktual yang positif, dan *specificity* adalah seimbang antara presisi dan recall (Kusuma & Nurramdhani Irmanda, 2022).

3. HASIL DAN PEMBAHASAN

3.1 Evaluasi Model

a. Scraping Data



```

1 from google_play_scraper import Scraper, reviews
2 results = scraper.get_reviews('com.example.app', 1)
3 scraper.get_reviews('com.example.app', 1)
4 scraper.get_reviews('com.example.app', 1)
5 scraper.get_reviews('com.example.app', 1)
6 scraper.get_reviews('com.example.app', 1)
7 scraper.get_reviews('com.example.app', 1)
8 scraper.get_reviews('com.example.app', 1)
9 scraper.get_reviews('com.example.app', 1)
10 scraper.get_reviews('com.example.app', 1)
11 scraper.get_reviews('com.example.app', 1)
12 scraper.get_reviews('com.example.app', 1)
13 scraper.get_reviews('com.example.app', 1)
14 scraper.get_reviews('com.example.app', 1)
15 scraper.get_reviews('com.example.app', 1)
16 scraper.get_reviews('com.example.app', 1)
17 scraper.get_reviews('com.example.app', 1)
18 scraper.get_reviews('com.example.app', 1)
19 scraper.get_reviews('com.example.app', 1)
20 scraper.get_reviews('com.example.app', 1)
21 scraper.get_reviews('com.example.app', 1)
22 scraper.get_reviews('com.example.app', 1)
23 scraper.get_reviews('com.example.app', 1)
24 scraper.get_reviews('com.example.app', 1)
25 scraper.get_reviews('com.example.app', 1)
26 scraper.get_reviews('com.example.app', 1)
27 scraper.get_reviews('com.example.app', 1)
28 scraper.get_reviews('com.example.app', 1)
29 scraper.get_reviews('com.example.app', 1)
30 scraper.get_reviews('com.example.app', 1)
31 scraper.get_reviews('com.example.app', 1)
32 scraper.get_reviews('com.example.app', 1)
33 scraper.get_reviews('com.example.app', 1)
34 scraper.get_reviews('com.example.app', 1)
35 scraper.get_reviews('com.example.app', 1)
36 scraper.get_reviews('com.example.app', 1)
37 scraper.get_reviews('com.example.app', 1)
38 scraper.get_reviews('com.example.app', 1)
39 scraper.get_reviews('com.example.app', 1)
40 scraper.get_reviews('com.example.app', 1)
41 scraper.get_reviews('com.example.app', 1)
42 scraper.get_reviews('com.example.app', 1)
43 scraper.get_reviews('com.example.app', 1)
44 scraper.get_reviews('com.example.app', 1)
45 scraper.get_reviews('com.example.app', 1)
46 scraper.get_reviews('com.example.app', 1)
47 scraper.get_reviews('com.example.app', 1)
48 scraper.get_reviews('com.example.app', 1)
49 scraper.get_reviews('com.example.app', 1)
50 scraper.get_reviews('com.example.app', 1)
51 scraper.get_reviews('com.example.app', 1)
52 scraper.get_reviews('com.example.app', 1)
53 scraper.get_reviews('com.example.app', 1)
54 scraper.get_reviews('com.example.app', 1)
55 scraper.get_reviews('com.example.app', 1)
56 scraper.get_reviews('com.example.app', 1)
57 scraper.get_reviews('com.example.app', 1)
58 scraper.get_reviews('com.example.app', 1)
59 scraper.get_reviews('com.example.app', 1)
60 scraper.get_reviews('com.example.app', 1)
61 scraper.get_reviews('com.example.app', 1)
62 scraper.get_reviews('com.example.app', 1)
63 scraper.get_reviews('com.example.app', 1)
64 scraper.get_reviews('com.example.app', 1)
65 scraper.get_reviews('com.example.app', 1)
66 scraper.get_reviews('com.example.app', 1)
67 scraper.get_reviews('com.example.app', 1)
68 scraper.get_reviews('com.example.app', 1)
69 scraper.get_reviews('com.example.app', 1)
70 scraper.get_reviews('com.example.app', 1)
71 scraper.get_reviews('com.example.app', 1)
72 scraper.get_reviews('com.example.app', 1)
73 scraper.get_reviews('com.example.app', 1)
74 scraper.get_reviews('com.example.app', 1)
75 scraper.get_reviews('com.example.app', 1)
76 scraper.get_reviews('com.example.app', 1)
77 scraper.get_reviews('com.example.app', 1)
78 scraper.get_reviews('com.example.app', 1)
79 scraper.get_reviews('com.example.app', 1)
80 scraper.get_reviews('com.example.app', 1)
81 scraper.get_reviews('com.example.app', 1)
82 scraper.get_reviews('com.example.app', 1)
83 scraper.get_reviews('com.example.app', 1)
84 scraper.get_reviews('com.example.app', 1)
85 scraper.get_reviews('com.example.app', 1)
86 scraper.get_reviews('com.example.app', 1)
87 scraper.get_reviews('com.example.app', 1)
88 scraper.get_reviews('com.example.app', 1)
89 scraper.get_reviews('com.example.app', 1)
90 scraper.get_reviews('com.example.app', 1)
91 scraper.get_reviews('com.example.app', 1)
92 scraper.get_reviews('com.example.app', 1)
93 scraper.get_reviews('com.example.app', 1)
94 scraper.get_reviews('com.example.app', 1)
95 scraper.get_reviews('com.example.app', 1)
96 scraper.get_reviews('com.example.app', 1)
97 scraper.get_reviews('com.example.app', 1)
98 scraper.get_reviews('com.example.app', 1)
99 scraper.get_reviews('com.example.app', 1)
100 scraper.get_reviews('com.example.app', 1)


```

review_id	username	user_avatar	content	score	timestamp	review_created_at
1	reza D		tolong di update lagi, masa udah ganti semester modul yang semester baru ga ada, adanya cuma modul di semester sebelumnya	2	2023-11-09 12:16:16	
2	Priska Dini		Ayo dong diinglatin, udah lama gabisa buka digital loading terus	1	2023-11-09 10:06:36	
3	hijup leh		penggunaannya tidak jelas	1	2023-10-21 07:42:26	
4	ivalkmaz teri		Aplikasi sering error, kalau di ada niat update apk apus aja ggama juga	1	2023-10-26 23:39:46	
5	Mandiri Info		App kurang bermanfaat ga bisa baca online mending bagi bagi pdf	1	2023-10-21 00:21:38	
6	Maria Yupita			5	2023-10-07 22:26:53	

Gambar 2. Scraping Data

Melakukan *Scraping* data dengan menggunakan *google-play-scraper* yang menghasilkan jumlah data sebanyak 800.

b. Translate Data



username	score	at	content	tlcontent
reza D	2	2023-11-09 12:16:16	tolong di update lagi, masa udah ganti semester modul yang semester baru ga ada, adanya cuma modul di semester sebelumnya	Please update again, after changing semesters, there are no modules for the new semester, there are only modules in the previous semester
Priska Dini	1	2023-11-09 10:06:36	Ayo dong diinglatin, udah lama gabisa buka digital loading terus	Come on, please improve it, it's been a long time since I could keep loading digital books
hijup leh	1	2023-10-21 07:42:26	penggunaannya tidak jelas	its use is unclear
ivalkmaz teri	1	2023-10-26 23:39:46	Aplikasi sering error, kalau di ada niat update apk apus aja ggama juga	The application often errors, if you don't have the intention to update the apk, just delete it
Mandiri Info	1	2023-10-21 00:21:38	App kurang bermanfaat ga bisa baca online mending bagi bagi pdf	The app is not very useful, it can't read online, it's better for pdfs
Maria Yupita	5	2023-10-07 22:26:53		

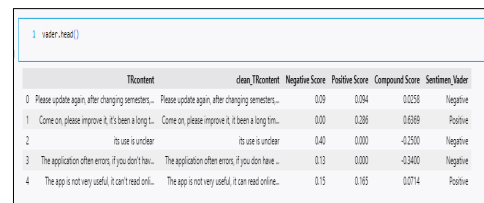
Gambar 3. Translate Data

Dataset yang sudah terkumpul, akan di terjemahkan menggunakan *library google-trans* dari bahasa Indonesia ke bahasa inggris untuk mempermudah dalam pemrosesan label *Vader* dan *TextBlob*.

Tabel 4. Hasil Translate Data

Content	Translate Content
tolong di update lagi, masa udah ganti semester modul yang semester baru ga ada, adanya cuma modul di semester sebelumnya	Please update again, after changing semesters, there are no modules for the new semester, there are only modules in the previous semester

c. Labelling Data



	tlcontent	clean_tlcontent	Negative Score	Positive Score	Compound Score	Sentimen_Vader
0	Please update again, after changing semesters...	Please update again, after changing semesters...	0.09	0.094	0.0238	Negative
1	Come on, please improve it, it's been a long time...	Come on, please improve it, it's been a long time...	0.00	0.286	0.6569	Positive
2	its use is unclear	its use is unclear	0.40	0.000	-0.2500	Negative
3	The application often errors, if you don't have...	The application often errors, if you don't have...	0.13	0.000	-0.3400	Negative
4	The app is not very useful, it can't read online...	The app is not very useful, it can't read online...	0.15	0.165	0.0714	Positive

Gambar 4. Labelling Data

Data yang sudah diterjemahkan ke bahasa inggris, selanjutnya akan dilakukan proses pelabelan. Pada proses ini, pelabelan menggunakan *Vader* dan *TextBlob* yang membagi menjadi dua kelas yaitu kelas negatif dan kelas positif.

Tabel 5. Hasil Labelling Data

Content	Sentimen
Ayo mohon diperbaiki, sudah lama sekali saya tidak bisa memuat buku digital terus menerus	Positive

Aplikasinya sering
error, kalau tidak
niat update apknya
hapus saja

Negative

3.2 Pre-Processing

a. Data Cleaning

Pada langkah ini, beberapa proses dilakukan untuk menghilangkan nilai yang hilang pada data, menghapus simbol karakter, karakter khusus, dan angka.

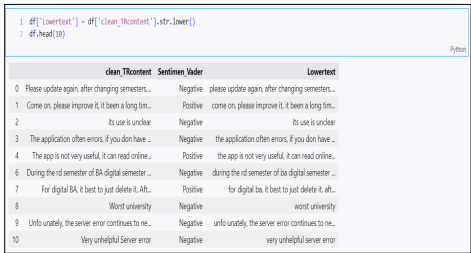


Gambar 5. Cleaning Data

Tabel 6. Hasil Cleaning Data

Sebelum Cleaning	Sesudah Cleaning
How come this new semester's courses haven't appeared on the BA DIGITAL UT account yet 🙄 Even though it's already session 2 🙄 Please fix it immediately	How come this new semester courses haven appeared on the BA DIGITAL UT account yet Even though it already session 2 Please fix it immediately

b. Case Folding



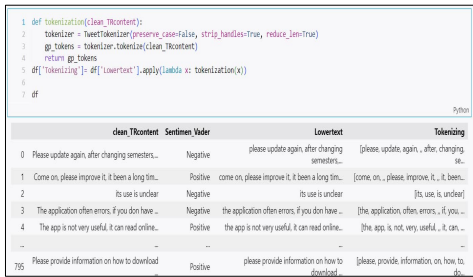
Gambar 6. Case Folding Data

Melakukan penormalan kata dengan mengubah semua huruf besar menjadi huruf kecil. Langkah ini bertujuan untuk mencegah tidak terdeteksinya kata yang sebenarnya sama tetapi memiliki perbedaan dalam bentuk huruf.

Content	Lowertext
Come on, please improve it, it been a long time since I could keep loading digital books	come on, please improve it, it been a long time since i could keep loading digital books

c. Tokenizing

Proses *tokenizing*, dilakukan untuk memisahkan kata-kata dari kalimat pada data tinjauan sebelumnya menjadi satuan kata atau kata penyusun.



Gambar 7. Tokenizing Data

Tabel 8. Hasil Tokenizing Data

Lowertext	Tokenizing
come on, please improve it, it been a long time since i could keep loading digital books	['come', 'on', ',', 'please', 'improve', 'it', ',', 'it', 'been', 'a', 'long', 'time', 'since', 'i', 'could', 'keep', 'loading', 'digital', 'books']

d. Stemming

```

1 factory = StemmerFactory()
2 stemmer = factory.create_stemmer()
3
4 def stemming(word):
5     stem_word = stemmer.stem(word)
6     return stem_word
7
8 df['stemming'] = df['tokenizing'].apply(lambda x: stemming(x))
9
10 df.head(5)

```

	clean_TfContent	Sentimen_Vader	Lowercase	Tokenizing	Stemming
0	Please update again, after changing semester...	Negative	please update again, after changing semester...	[please, update, again, after, changing, sem...	[please, update, again, after, changing, sem...
1	Come on, please improve it, it been a long time.	Positive	come on, please improve it, it been a long time.	[come on, please, improve, it, it, been...	[come on, please, improve, it, it, been...
2	It use is unclear	Negative	it use is unclear	[it, use, is, unclear]	[it, use, is, unclear]
3	The application often errors, if you don't have...	Negative	the application often errors, if you don't have...	[the, application, often, errors, if, you, don't, have...	[the, application, often, errors, if, you, don't, have...
4	The app is not very useful, it can read online.	Positive	the app is not very useful, it can read online.	[the, app, is, not, very, useful, it, can, read, online...	[the, app, is, not, very, useful, it, can, read, online...

Gambar 8. Stemming Data

Dalam tahap *Stemming*, terjadi proses pengubahan kata-kata pada data ulasan menjadi kata dasar dengan cara menghilangkan dari imbuhan, baik di awal, di tengah, maupun di akhir kata.

Tabel 9. Hasil Stemming Data

Tokenizing	Stemming
['can', 'use', ',', 'book', 'doesn', 'Terbuka']	['can', 'use', ',', 'book', 'doesn', 'Terbuka']
['it', 'very', 'difficult', 'to', 'log', 'in', ',', 'please', 'fix', 'it', 'immediately']	['it', 'very', 'difficult', 'to', 'log', 'in', ',', 'please', 'fix', 'it', 'immediately']

e. Stopwords Removal

```

1 stopwords_eng = stopwords.words('english')
2
3 def remove_stopwords(tokenizing):
4     output = ""
5     for i in range(1, len(tokenizing)):
6         if tokenizing[i] not in stopwords_eng:
7             output += tokenizing[i] + " "
8     return output
9
10 df['stopwords'] = df['tokenizing'].apply(lambda x: remove_stopwords(x))
11
12 df.head(5)

```

	clean_TfContent	Sentimen_Vader	Lowercase	Tokenizing	Stemming	Stopwords
0	Please update again, after changing semester...	Negative	please update again, after changing semester...	[please, update, again, after, changing, sem...	[please, update, again, after, changing, sem...	please update changing semester module new...
1	Come on, please improve it, it been a long time.	Positive	come on, please improve it, it been a long time.	[come on, please, improve, it, it, been...	[come on, please, improve, it, it, been...	come please improve long time since could fix...
2	It use is unclear	Negative	it use is unclear	[it, use, is, unclear]	[it, use, is, unclear]	use unclear
3	The application often errors, if you don't have...	Negative	the application often errors, if you don't have...	[the, application, often, errors, if, you, don't, have...	[the, application, often, errors, if, you, don't, have...	application often errors intention update app...
4	The app is not very useful, it can read online.	Positive	the app is not very useful, it can read online.	[the, app, is, not, very, useful, it, can, read, online...	[the, app, is, not, very, useful, it, can, read, online...	app useful read online better pdfs

Gambar 9. Stopwords Renova Data

Penulis melakukan penghilangan terhadap kata-kata yang tidak bermakna, namun tidak mengurangi makna dari informasi suatu kalimat tertentu.

Tabel 10. Hasil Stopwords Removal Data

Stemming	Stopwords
['the', 'app', 'is', 'not', 'very', 'useful', ',', 'it', 'can', 'read', 'online', 'if', 'you', 'don't', 'have', 'it', 'can', 'read', 'online', 'better', 'for', 'pdfs']	app useful read online better pdfs

3.3 Pembobotan Kata

Melanjutkan dengan tahap pembobotan kata, dimana untuk memboboti setiap kata menjadi nilai fitur.

```

1 #Training Testing - TF-IDF
2 vectorizer = TfidfVectorizer(min_df = 5,
3                             max_df = 0.8,
4                             sublinear_tf = True,
5                             use_idf = True)
6 vectorizer.fit_transform(df['clean_TfContent'])
7 Train_X_Tfidf = vectorizer.transform(Train_X)
8 Test_X_Tfidf = vectorizer.transform(Test_X)
9 print(Train_X_Tfidf)

```

(1, 331)	0.6543319062428389
(1, 147)	0.7562074824230534
(2, 298)	0.2273745531134242
(2, 169)	0.32528962221617336
(2, 136)	0.5007401097427103
(2, 56)	0.7692508152545495
(3, 338)	0.77920313673814958
(3, 64)	0.48098169099108544
(3, 31)	0.4741327046867287
(4, 319)	0.23746876227754155

Gambar 10. Pembobotan Kata Pada Data

3.4 Klasifikasi Deep Learning

Pengujian terakhir yaitu dengan menggunakan model *deep learning Long Short Term Memory (LSTM)*, dengan data yang sama tetapi dengan label yang berbeda. Pada pengujian kali ini melakukan *epoch* sebanyak 5 kali menggunakan data dengan label *Vader* mendapatkan akurasi 76.72% dan pada label *TextBlob* mendapatkan akurasi 74.21%. Pada Tabel perbandingan adalah hasil pengujian *deep learning* tersebut ditunjukkan pada Tabel dibawah ini.

Tabel 11. Hasil Pengujian Epoch Long Short Term Memory dengan label TextBlob

Uji Epoch	1	2	3	4	5
Step(ms)	295	221	222	207	197
Loss	0.67	0.64	0.60	0.55	0.45
Accuracy	0.58	0.59	0.64	0.76	0.83
Val-Loss	0.63	0.59	0.66	0.58	0.52
Val-Accuracy	0.65	0.65	0.54	0.68	0.74

Tabel 12. Hasil Pengujian Epoch Long Short Term Memory dengan label Vader

Uji Epoch	1	2	3	4	5
Step(ms)	256	193	212	207	214
Loss	0.68	0.63	0.54	0.39	0.28
Accuracy	0.60	0.66	0.78	0.87	0.90
Val-Loss	0.67	0.61	0.55	0.50	0.48
Val-Accuracy	0.65	0.66	0.75	0.78	0.76

Tabel 13. Hasil Akurasi Deep Learning (LSTM)

Algoritma	Nilai Akurasi Vader (%)	Nilai Akurasi TextBlob (%)
Deep Learning	76.72	74.21

3.5 Evaluasi Model Long Short Term Memory

Tahapan berikutnya dilakukan evaluasi terhadap model yang sudah dibuat pada tahap sebelumnya. Untuk Evaluasi model ini menggunakan metode *confusion matrix* yang berfungsi dalam melakukan pengecekan performa dari suatu model atau algoritma. Untuk dapat melihat tingkat kesalahan dari beberapa algoritma bisa dilihat melalui hasil *confusion matrix* yang menampilkan *accuracy*, *precision*, *recall*, dan *f1-score* sebagai berikut :

Tabel 14. Evaluasi Klasifikasi Long Short Term Memory

Algoritma	Precision	Recall	F1-Score
Vader	0.91	0.78	0.84
TextBlob	0.96	0.45	0.61

3.6 Analisis Hasil Klasifikasi Antar Label

Setelah melakukan klasifikasi ulasan pada aplikasi menggunakan teknik *text mining* dengan label *Vader* dan *TextBlob*, maka bisa dilanjutkan ke tahap analisis dimana dari hasil pengujian menggunakan beberapa algoritma salah satunya dengan algoritma *Long Short Term Memory* yang berhasil mencapai kinerja yang terbaik menggunakan label *Vader*, dengan *accuracy* sebesar 76.72%, *precision* 0.91, *recall* 0.78, dan *f1-score* 0.84 sedangkan dengan label *TextBlob* mendapatkan *accuracy* 74.21% *precision* 0.96, *recall* 0.45, dan *f1-score* 0.61 dengan label *TextBlob*. Namun, dalam menganalisis suatu kesalahan saat melakukan klasifikasi memiliki suatu hambatan, terutama pada saat menghadapi ulasan dengan bahasa informal, penggunaan singkatan dan *emoticon*. Contohnya, beberapa ulasan menggunakan singkatan seperti “plz” (*Please*) dan contoh lainnya yang dapat membingungkan algoritma tersebut dalam menentukan sentimen yang

sebenarnya. Selain itu, dari banyaknya ulasan yang mengandung bahasa informal tetapi menggunakan bahasa formal dalam konteks tertentu juga menjadi tantangan tersendiri. Melalui analisis lebih lanjut, ditemukan bahwa ulasan positif cenderung mengarah dalam penggunaan aplikasi dan efisiensi pemakaian, sementara ulasan negatif cenderung mencakup keluhan terkait dengan *bug* atau masalah teknis yang mengganggu.

Penggunaan label *Vader* dan *TextBlob* melalui parameter *Labelling* data, bahwa untuk mencari *accuracy* dari pengujian tersebut dengan label *Vader* lebih baik daripada label *TextBlob*, dimana label *Vader* dalam hal meningkatkan *accuracy* sentimen ulasan yang menjadi pilihan efektif dalam upaya meningkatkan pelayanan dan kualitas belajar melalui aplikasi.

4. KESIMPULAN

Berdasarkan hasil dari beberapa tahapan penelitian yang telah dilakukan dapat disimpulkan bahwa pada proses *deep learning* dapat diterapkan dalam uji sentimen dari aplikasi Bahan Ajar Digital Universitas Terbuka dengan metode klasifikasi *Long Short Term Memory* dalam menganalisis dan

mengelompokkan dari ulasan aplikasi. Sehingga dapat memberikan hasil yang bervariasi untuk memprediksi dan mengklasifikasi ulasan aplikasi serta dapat menentukan model *Labelling* data mana yang memberikan performa paling baik dalam mengklasifikasi data ulasan dari aplikasi Bahan Ajar Digital Universitas Terbuka. Salah satu metode klasifikasi menggunakan algoritma *Long Short Term Memory* yang memiliki nilai akurasi yang terbaik jika dibandingkan dengan algoritma lainnya.

Melalui analisis ini, tujuannya adalah untuk meningkatkan kualitas klasifikasi ulasan dalam aplikasi di masa mendatang, terutama dalam memahami variasi ulasan serta makna penggunaan aplikasi tersebut. Sehingga dapat melakukan peningkatan sistem dari hasil klasifikasi yang lebih efektif dan responsif terhadap kebutuhan pengguna. Dengan demikian, diharapkan pengguna dapat mendapatkan pengalaman yang lebih memuaskan dan relevan saat menggunakan aplikasi tersebut.

DAFTAR PUSTAKA

Aditiya, P., Enri, U., & Maulana, I. (2022). Analisis Sentimen Ulasan Pengguna Aplikasi Myim3 Pada Situs Google Play Menggunakan Support Vector Machine. *JURIKOM (Jurnal*

- Riset Komputer*), 9(4), 1020. <https://doi.org/10.30865/jurikom.v9i4.4673>
- Anisa, D. F. N., Mukhlash, I., & Iqbal, M. (2023). Deteksi Berita Online Hoax Covid-19 Di Indonesia Menggunakan Metode Hybrid Long Short Term Memory dan Support Vector Machine. *Jurnal Sains Dan Seni ITS*, 11(3). <https://doi.org/10.12962/j23373520.v11i3.83227>
- ArFan, A., & Lussiana ETP. (2019). Prediksi Harga Saham Di Indonesia Menggunakan Algoritma Long Short-Term Memory. *Seminar Nasional Teknologi Informasi Dan Komunikasi STIK (SeNTIK)*, 3(1), 2581–2327.
- Dwi, R., Santosa, W., Arif Bijaksana, M., & Romadhony, A. (n.d.). Implementasi Algoritma Long Short-Term Memory (LSTM) untuk Mendeteksi Penggunaan Kalimat Abusive Pada Teks Bahasa Indonesia. In *Jurnal Tugas Akhir Fakultas Informatika* (Vol. 12, Issue 2020).
- Fatariska, I. O. (2023). Teknik dan Kualitas Terjemahan dalam Artikel Berjudul “Batik, The Traditional Fabric of Indonesia” Menggunakan Penejermahan Online. *Jurnal El-Hamra: Kependidikan Dan Kemasyarakatan*, 8(1), 2721–6047.
- Husnina, D. N. N., Ratnawati, D. E., & Rahayudi, B. (2023). Analisis Sentimen Pengguna Aplikasi RedBus berdasarkan Ulasan di Google Play Store menggunakan Metode Naïve Bayes. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 7(2), 737–743. <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/12297%0Ahttps://j-ptiik.ub.ac.id/index.php/j-ptiik/article/download/12297/5593>
- Khoiruddin, Y., Fauzi, A., & Siregar, A. M. (2023). Analisis Sentimen Gojek Indonesia Pada Twitter Menggunakan Algoritme Naïve Bayes Dan Support Vector Machine. *Jurnal Ilmiah Komputer*, 19, 391–400.
- Kulsum, U., Jajuli, M., & Sulistiyowati, N. (2022). Analisis Sentimen Aplikasi WETV di Google Play Store Menggunakan Algoritma Support Vector Machine. *Journal of Applied Informatics and Computing*, 6(2), 205–212. <https://doi.org/10.30871/jaic.v6i2.4802>
- Kusuma, A., & Nurramdhani Irmanda, H. (2022). Analisis Sentimen Pada Ulasan Aplikasi Indodax di Google Play Store Menggunakan Metode Support Vector Machine.
- Lestari, N. A., Akhriza, M., & Yuniar, E. (2020). Metode Naïve Bayes Classifier Dengan Textblob Untuk Analisis Sentimen Terhadap Pelayanan Indihome Dan First Media. *Seminar Nasional Teknologi Informasi Dan Komunikasi STI&K (SeNTIK)*, 4(1), 283–288. <https://t.co/Ws2wOyU5kz>
- Nurhafida, S. I., & Sembiring, F. (2022). Analisis Sentimen Aplikasi Novel Online Di Google Play Store Menggunakan Algoritma Support Vector Machine (SVM). *Jurnal Sains Komputer & Informatika (J-SAKTI)*, 6(1), 317–327.

- Riany, A. F., & Testiana, G. (2023). Penerapan Data Mining untuk Klasifikasi Penyakit Stroke Menggunakan Algoritma Naïve Bayes. *Jurnal SAINTEKOM*, 13(1), 42–54. <https://doi.org/10.33020/saintekom.v13i1.352>
- Tampubolon, D. N. M., Hulu, V. V., Sipahutar, R. O., & Sihombing, O. (2023). Analisis Prediksi Genre Film Pada Internet Movie Database Indonesia Menggunakan Metode Long Short Term Memory. *Jurnal Tekinkom (Teknik Informasi Dan Komputer)*, 6(2), 290–298. <https://doi.org/10.37600/tekinkom.v6i2.925>
- Taufiq Anwar, M., Riandhita Arief Permana, D., STMI Jakarta, P., Sistem Informasi Industri Otomotif, P., Letjen Suprpto No, J., & Pusat, J. (2023). Analisis Sentimen Masyarakat Indonesia Terhadap Produk Kendaraan Listrik Menggunakan VADER. *Jurnal Teknik Informatika Dan Sistem Informasi*, 10(1), 783–792. <http://jurnal.mdp.ac.id>