

## **BAB III**

### **METODOLOGI PENELITIAN**

#### **3.1 Subjek dan Objek Penelitian**

Subjek penelitian yang digunakan dalam penelitian ini adalah *dataset* komentar *review e-wallet* DANA, OVO, dan GoPay dari Google PlayStore. Sedangkan objek penelitian yang digunakan pada penelitian ini adalah pelabelan sentimen menggunakan algoritma LSTM dan GRU.

#### **3.2 Bahan dan Alat Penelitian**

Bahan penelitian yang digunakan dalam penelitian ini adalah *dataset* komentar *review e-wallet* DANA, OVO, dan GoPay yang diperoleh dari *scraping* aplikasi Google PlayStore. Sedangkan alat yang digunakan dalam penelitian ini terdiri dari dua jenis yaitu perangkat keras (*hardware*) dan perangkat lunak (*software*), adapun perinciannya sebagai berikut:

1. Perangkat Keras (*Hardware*)

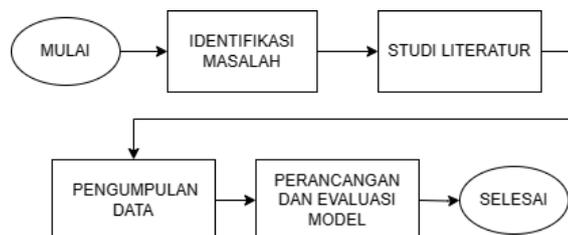
Device	: Dell Inspiron 5415
Processor	: AMD Hexa Core Ryzen 5 5500U Up To 4,0Ghz
RAM	: 8 GB

2. Perangkat Lunak (*Software*)

Sistem Operasi	: Windows 11 Enterprise 64-bit
Bahasa Pemrograman	: Python
Aplikasi	: Jupyter Notebook, Google Colaboratory, Microsoft Word, dan Microsoft Excel

#### **3.3 Diagram Alir Penelitian**

Secara umum, diagram alir penelitian dari awal hingga akhir ditunjukkan dalam Gambar 3.1.



Gambar 3. 1 Rancangan Alur Penelitian

Berdasarkan Gambar 3.1, penelitian akan dilakukan 5 tahap, yaitu dimulai dengan identifikasi masalah, studi literatur, pengumpulan *dataset*, perancangan dan evaluasi model, dan dokumentasi dan penulisan penelitian.

### 3.3.1. Identifikasi Masalah

Identifikasi masalah menjadi dasar dalam penelitian ini. Tahap pertama yang dilakukan dalam penelitian adalah identifikasi masalah dengan menentukan topik penelitian terlebih dahulu. Kelemahan anotasi sentimen oleh manusia dan mesin menjadi rumusan permasalahan, sehingga anotasi sentimen menggunakan metode *Semi-Supervised Learning* menjadi topik pada penelitian ini. Hal ini kemudian dapat menjadi garis dasar dalam penelitian. Sehingga dalam penelitian ini bertujuan mengimplementasikan *Semi-Supervised Learning* menggunakan algoritma LSTM, dan mengetahui tingkat akurasi.

### 3.3.2. Studi Literatur

Setelah ditentukan tema, topik, dan permasalahan, lalu dilakukan studi literatur. Dimulai dengan mencari berbagai referensi dari berbagai buku, jurnal, dan sumber lainnya terkait dengan metode dan topik penelitian. Selain mendapatkan informasi dari sumber referensi, studi literatur juga mendukung dalam pengolahan informasi (data), tahap pemrosesan data, dan pengembangan metode yang diterapkan pada penelitian. Keluaran pada tahap ini adalah mendapatkan berbagai jurnal yang digunakan dalam referensi selama penelitian, termasuk jurnal utama sebagai jurnal acuan dalam penelitian.

### 3.3.3. Pengumpulan Data

Data dikumpulkan melalui proses *scraping* aplikasi yang terdapat pada Google Play Store. Proses ini menggunakan library yang tersedia pada bahasa

pemrograman python, yaitu google-play-scraper. Untuk dapat menggunakan library, terlebih dahulu instalasi dengan perintah `pip install google-play-scraper`. Berikut merupakan *source code* pada proses pengumpulan data:

```
result, continuation_token = reviews(
    'com.gojek.gopay',
    lang='id',
    country='id',
    sort=Sort.MOST_RELEVANT,
    count=3500,
    filter_score_with= 5
)
```

Parameter yang digunakan dalam scraping pada penelitian ini, antara lain id aplikasi, negara, bahasa, sorting komentar, banyak data yang diinginkan, dan *filter score*. Id aplikasi menyesuaikan aplikasi yang akan digunakan dalam pengambilan data. Bahasa (*lang*) menggunakan 'id' yang memiliki arti Bahasa Indonesia. Negara (*country*) menggunakan 'id' yang memiliki arti Negara Indonesia. Sorting komentar yang digunakan yaitu berdasarkan komentar terbaru (*newest*), namun juga memungkinkan untuk menggunakan sorting komentar berdasarkan yang paling relevan (*most relevant*). Banyak data diatur sebanyak maksimal 3000 data, dan apabila membutuhkan lebih banyak data dapat diubah sesuai jumlah data yang dibutuhkan. *filter score* yaitu skor (*rating*) berapa saja yang dibutuhkan, yaitu dapat diisi dengan angka 1, 2, 3, 4, atau 5, 'None' apabila ingin mendapatkan seluruh *rating*. *filter score* menunjukkan kecenderungan komentar pengguna terhadap aplikasi, yaitu semakin tinggi angka pada *filter score*, maka opini yang diberikan pengguna aplikasi semakin mengarah pada komentar positif. Pada penelitian ini menggunakan *filter score* dengan angka 1, 2, 4, dan 5. Berlandaskan pada penelitian [64], kategori komentar negatif diperoleh dengan mengumpulkan *dataset* yang memiliki *filter score* angka 1 dan 2. Sedangkan, kategori komentar positif diperoleh dengan mengumpulkan *dataset* yang memiliki *filter score* angka 4 dan 5.

*Dataset* yang digunakan ada 3 macam, yaitu *dataset* komentar *review e-wallet* DANA, OVO, dan GoPay dari Google PlayStore. *Dataset* pertama, menggunakan data yang diambil dari PlayStore pada komentar aplikasi DANA

sebanyak 18353 data, terlihat pada Tabel 3.1. *Dataset* kedua diambil dari PlayStore pada komentar *review e-wallet* OVO sebanyak 23880 data, terlihat pada Tabel 3.2. *Dataset* ketiga diambil dari PlayStore pada komentar aplikasi GoPay sebanyak 1592 data, terlihat pada Tabel 3.3.

Tabel 3. 1 Dataset pertama (DANA)

<b>Komentar</b>	<b>Label</b>
Sangat kecewa dengan pihak DANA, saldo saya berkurang sendiri. Tapi tidak ada tindak lanjuti buat kembalikan saldo saya...malah suruh tunggu tunggu dan tunggu	1
Apk tambah lemot sering pembayaran tidak muncul di coba sampai relod hp begitu juga tdk muncul	2
lumayan membantu dengan instan	4
Sangat sangat puas dan bangga atas segala bentuk pelayanan aplikasi DANA, Terimakasih	5

Tabel 3. 2 Dataset Kedua (OVO)

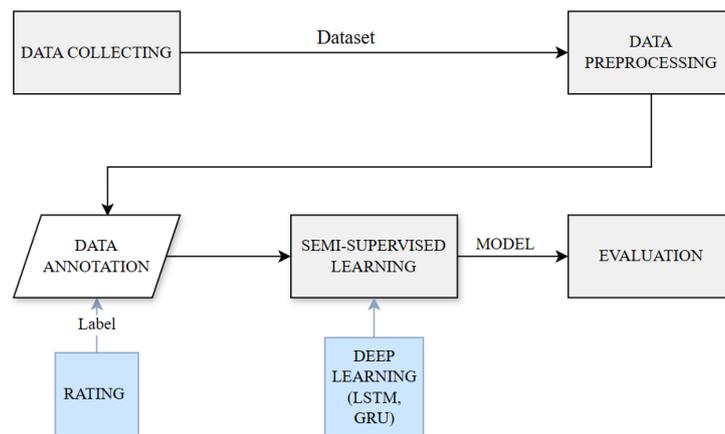
<b>Komentar</b>	<b>Label</b>
Susah mau masuk apl tiba tiba keluar sendiri trus	1
Aplikasi nya ga bisa di upgrate kaya di dimainkan di ulang" verifikasi gagal Tolong dong admin di perbaiki	2
Cukup puas dengan layanannya tapi masih ada kekurangannya	4
Mantapp buat top up game bisa lebihh murah dan cepatt	5

Tabel 3. 3 Dataset Ketiga (GoPay)

<b>Komentar</b>
Mau login ga bisa OTP udh dikirim tapi ga ada OTP yg masuk sama sekali. Gara gara di update malah ga jelas begini
Baru top up tiba" saldo langsung hilang keterangan transfer ke gopay tabungan , pas saya lihat tidak ada di saldonya di gopay tabungan, gak ada tanggapan sama sekali di email
masih ada bug bagian transfer

### 3.3.4. Perancangan dan Evaluasi Model

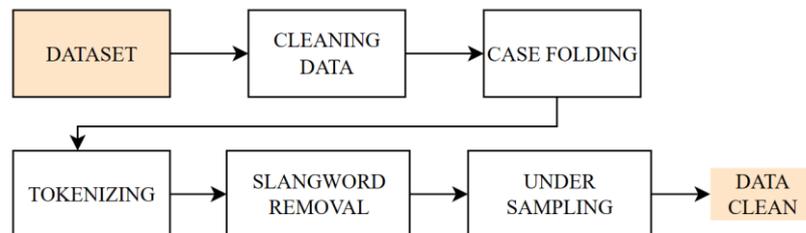
Skema tahap perancangan dan evaluasi model ini ditunjukkan oleh Gambar 3.2. Dimulai dari tahap pengumpulan data, *preprocessing*, *data annotation*, *modelling*, *testing classification*, *validation*, dan terakhir akan diperoleh akurasi, *recall*, *precision*, dan *F1-score*.



Gambar 3. 2 Perancangan Model Penelitian

#### 1. Pemrosesan Awal Data (*Data Preprocessing*)

Alur pemrosesan awal data dapat dilihat pada Gambar 3.3.



Gambar 3. 3 Alur Pemrosesan awal Data

##### A) *Drop Null dan Drop Duplicates Data*

Pada tahap menghilangkan data duplikat dan null, *dataset* komentar aplikasi OVO berkurang 217 baris data, *dataset* komentar aplikasi DANA berkurang 16 baris data, dan *dataset* komentar aplikasi GoPay tidak ada pengurangan data dikarenakan tidak terdapat data duplikat dan null. *Source code* untuk menghapus duplikat data dapat dilihat pada Gambar 3.4, sedangkan *source code* untuk menghapus data kosong dapat dilihat pada Gambar 3.5.

```
df = df.drop_duplicates()
```

Gambar 3. 4 *Source code* menghapus duplikat data

```
df = df.dropna()
```

Gambar 3. 5 *Source code* menghapus data kosong

Perbandingan banyak data sebelum dan setelah proses dapat dilihat pada Tabel 3.4. Pada Tabel 3.4, *dataset* OVO terdapat 217 data duplikat, dan tidak memiliki data kosong (null). *Dataset* DANA terdapat 16 data duplikat, dan tidak memiliki data kosong (null). Sedangkan *dataset* GoPay tidak terdapat data duplikat maupun data kosong, sehingga jumlah baris data tidak berkurang setelah melewati proses ini.

Tabel 3. 4 Perbandingan Banyak Data

	OVO					DANA				
	1	2	4	5	Total Data	1	2	4	5	Total Data
Data asli	5174	5572	4975	8159	23880	4776	4000	4577	5000	18353
Setelah drop duplicate	5171	5571	4951	7970	23663	4776	4000	4575	4986	18337
Setelah drop null	5171	5571	4951	7970	23663	4776	4000	4575	4986	18337

	GOPAY				
	1	2	4	5	Total Data
Data asli	398	398	597	597	1990
Setelah drop duplicate	398	398	597	597	1990
Setelah drop null	398	398	597	597	1990

### B) *Cleaning Data*

Menghapus elemen-elemen non-teks, seperti *hashtags*, URL, angka, simbol (titik koma, tanda petik, tanda tanya), simbol mata uang (Dollar \$). Hasil *cleaning* data dapat dilihat pada Tabel 3.5, yang menunjukkan perbedaan sebelum dan setelah proses pembersihan data.

Tabel 3. 5 Hasil cleaning data

Sebelum Cleaning	Setelah Cleaning
Woi aplikasi gak jelas lu , susah amat dibuka, gimana mau chat support orang baru masuk udah keluar lagi...aneh2 aja	Woi aplikasi gak jelas lu susah amat dibuka gimana mau chat suport orang baru masuk udah keluar lagi aneh aja

### C) Case Folding

Tahap *case folding* dilakukan menggunakan metode `.lower()` untuk mengubah seluruh huruf menjadi huruf kecil, atau metode `.upper()` untuk mengubah seluruh huruf menjadi huruf besar. Hasil *case folding* dapat dilihat pada Tabel 3.6, yang menunjukkan perbedaan sebelum dan setelah proses *case folding*.

Tabel 3. 6 Hasil case folding

Sebelum Case Folding	Setelah Case Folding
Woi aplikasi gak jelas lu susah amat dibuka gimana mau chat suport orang baru masuk udah keluar lagi aneh aja	woi aplikasi gak jelas lu susah amat dibuka gimana mau chat suport orang baru masuk udah keluar lagi aneh aja

### D) Tokenisasi

*Tokenizing* atau tokenisasi berfungsi memisahkan teks menjadi kata-kata individual. Tokenisasi dapat memudahkan identifikasi dan menghapus kata-kata slang atau kata tidak baku dari teks. Menggunakan `from nltk.tokenize import word_tokenize` untuk memisahkan kalimat menjadi per kata. Hasil *tokenizing* dapat dilihat pada Tabel 3.7, yang menunjukkan perbedaan sebelum dan setelah proses tokenisasi.

Tabel 3. 7 Hasil Tokenisasi

Sebelum <i>Tokenizing</i>	Setelah <i>Tokenizing</i>
woi aplikasi gak jelas lu susah amat dibuka gimana mau chat suport orang baru masuk udah keluar lagi aneh aja	['woi', 'aplikasi', 'gak', 'jelas', 'lu', 'susah', 'amat', 'dibuka', 'gimana', 'mau', 'chat', 'suport', 'orang', 'baru', 'masuk', 'udah',

	'keluar', 'lagi', 'aneh', 'aja']
--	----------------------------------

#### E) Slangword Removal

*Slangword* atau kata slang sering memiliki makna yang ambigu atau berubah-ubah tergantung pada konteksnya. Dengan menghapus atau mengganti dengan kata baku, dapat mengurangi kemungkinan ambiguitas atau variasi dalam pemahaman kata-kata. Kamus yang terdapat dalam nama file slangwords.txt berisi daftar kata-kata slang beserta dengan kata penggantinya atau mungkin tidak perlu digantikan (dihilangkan). Contoh daftar kata slang pada file tersebut dapat dilihat pada Tabel 3.8.

Tabel 3. 8 Contoh kamus kata slang

Tidak Baku	Baku	Tidak Baku	Baku	Tidak Baku	Baku
nglindur	mengigau	ngga	tidak	ngurus	mengurus
ngerampok	merampok	ngibul	berbohong	nih	ini
nie	ini	ngomong	bicara	ngiri	iri

Hasil slangword removal dapat dilihat pada Tabel 3.9, yang menunjukkan perbedaan sebelum dan setelah proses pemrosesan kata slang.

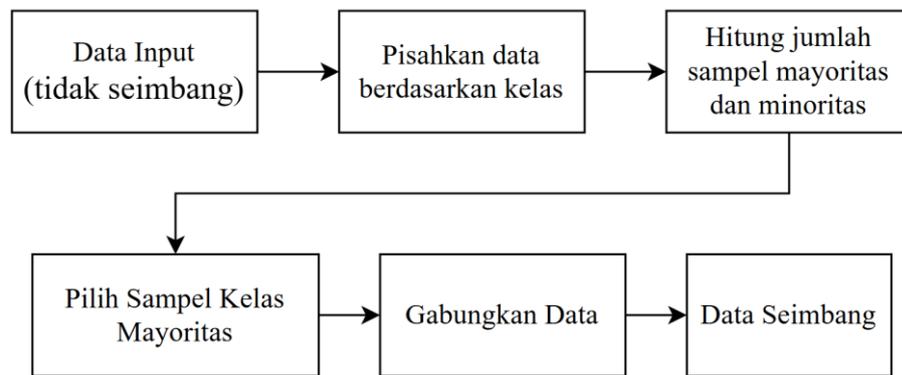
Tabel 3. 9 Hasil slangword removal

Sebelum Slangword Removal	Setelah Slangword Removal
['woi', 'aplikasi', 'gak', 'jelas', 'lu', 'susah', 'amat', 'dibuka', 'gimana', 'mau', 'chat', 'suport', 'orang', 'baru', 'masuk', 'udah', 'keluar', 'lagi', 'aneh', 'aja']	['', 'aplikasi', 'tidak', 'jelas', 'kamu', 'susah', 'amat', 'dibuka', 'bagaimana', 'mau', 'pesan', 'suport', 'orang', 'baru', 'masuk', 'sudah', 'keluar', 'lagi', 'aneh', 'saja']

#### F) Under sampling

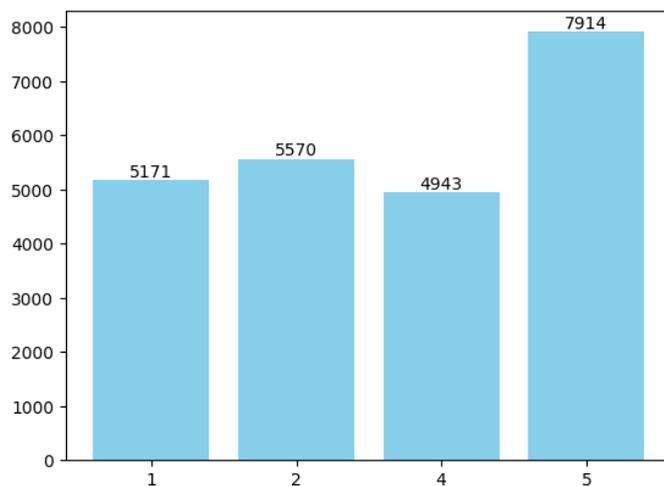
Tahapan yang dilakukan setelah preprocessing adalah *handling imbalance data*, yaitu menangani ketidakseimbangan kuantitas antarkelas pada *dataset*. Pada Gambar 3.6 dan 3.8 dapat diketahui bahwa kuantitas data tidak sama di setiap kelasnya, sehingga perlu dilakukan metode penanganan ketidakseimbangan jumlah data. Metode undersampling memiliki performa yang bagus ketika

digunakan untuk menangani ketidakseimbangan data pada jumlah data yang besar. Sebaliknya, metode oversampling memiliki performa yang bagus ketika digunakan untuk menangani ketidakseimbangan data pada jumlah data yang sedikit. Pemilihan metode yang tepat dapat memperbaiki kinerja model dalam melakukan tugas, dalam penelitian ini yaitu tugas klasifikasi. Hal ini selaras pada penelitian [65] dan [66]. Flowchart dari proses *under sampling* ditunjukkan pada Gambar 3.6.

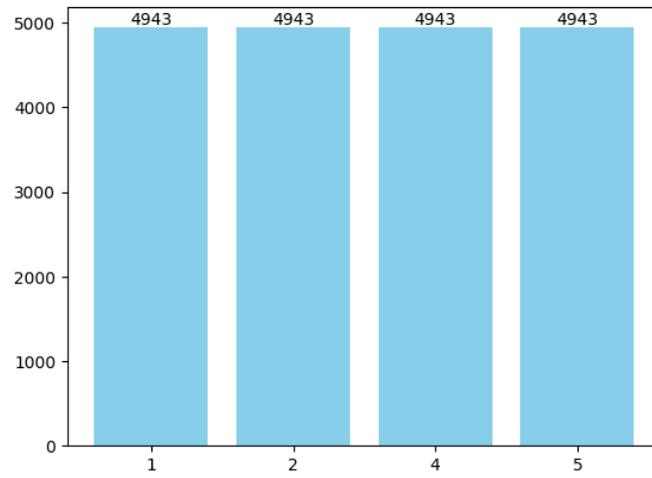


Gambar 3. 6 Proses *Under Sampling*

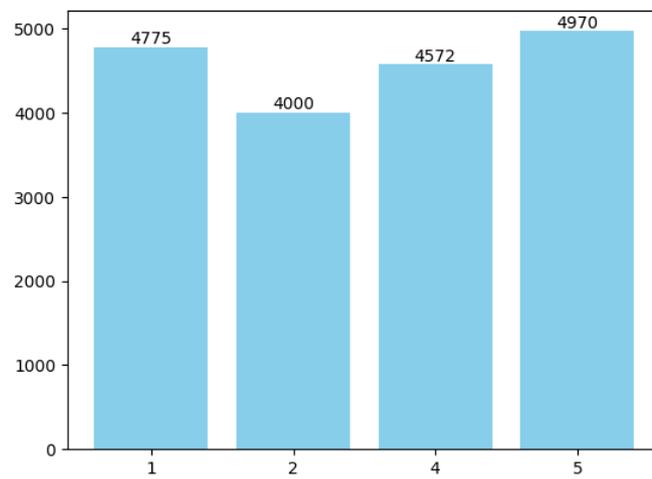
Grafik yang menunjukkan persebaran banyak data pada setiap kelas dapat dilihat pada Gambar 3.7, Gambar 3.8, Gambar 3.9, dan Gambar 3.10.



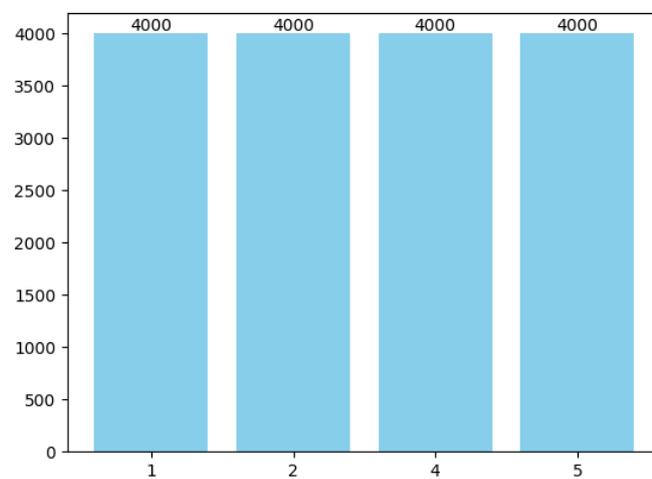
Gambar 3. 7 Banyak data OVO sebelum *Under Sampling*



Gambar 3. 8 Banyak data OVO setelah *Under Sampling*



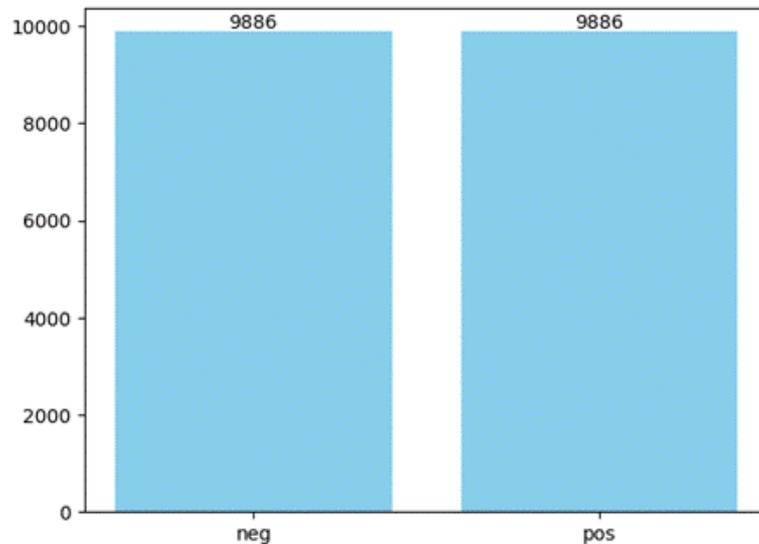
Gambar 3. 9 Banyak data DANA sebelum *Under Sampling*



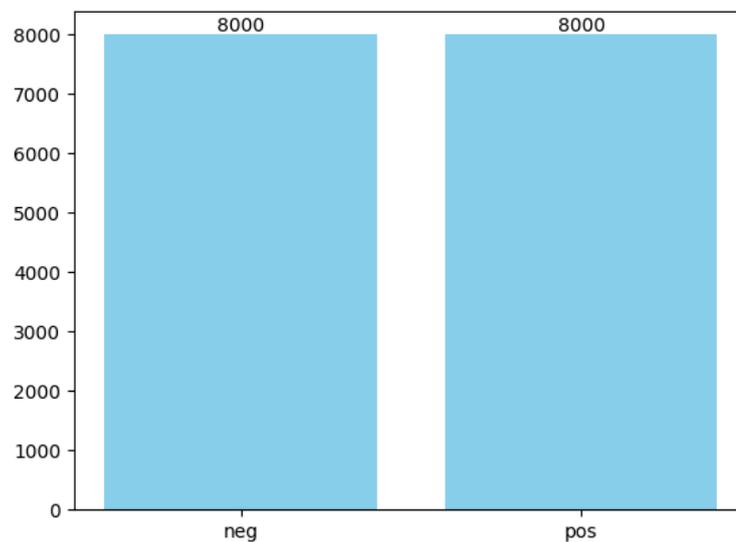
Gambar 3. 10 Banyak data DANA setelah *Under Sampling*

## 2. Pelabelan Data

*Dataset* OVO memiliki banyak data pada kelas positif dan negatif masing-masing sebanyak 9886 data, dapat dilihat pada Gambar 3.11. Sedangkan *dataset* DANA memiliki banyak data pada kelas positif dan negatif masing-masing sebanyak 8000 data, dapat dilihat pada Gambar 3.12.



Gambar 3. 11 Banyak data OVO setelah *labelling*



Gambar 3. 12 Banyak data DANA setelah *labeling*

Setelah dilakukan pengelompokan kelas menjadi kelas positif dan negatif, tahap selanjutnya adalah *dataset* OVO dan DANA digabungkan menjadi satu

*dataset* untuk menjadi data latih (*training data*). Tahap penggabungan *dataset* menggunakan perintah `concat`, seperti yang terdapat pada Gambar 3.13.

```
concat =pd.concat([ovo, dana])
```

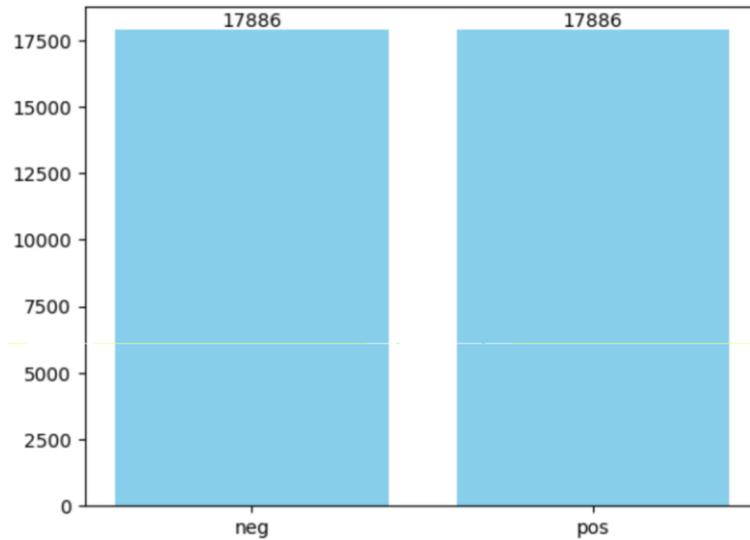
Gambar 3. 13 *Source code* penggabungan data

Hasil penggabungan *dataset* pertama dan kedua dapat dilihat pada Tabel 3.9, dengan jumlah data sebanyak 35.772 data.

Tabel 3. 10 Hasil penggabungan dataset

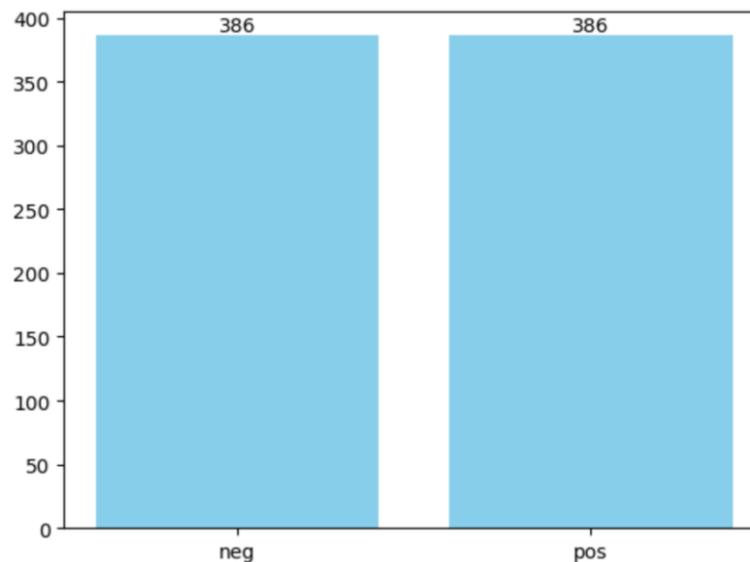
Data	score
saya baru unduh ovo terus disuruh meningkatkan ke ovo premier tapi mau di meningkatkan tidak bisa gagal terus	0
pelayanan buruk yang pernah saya temui di aplikasi transaksi uang masak saya transfer dana dari jam pagi jam sore tidak masuk pas ke lagi di proses terus tidak sudah aplikasi macam apa ini tolong main toko ditegur lah apa di baned sekali bukan dikit saya melakukan transaksi uang padahal itu kan uang saya sendiri dan transfer juga ke rekening saya sendiri	0
halo min biasanya transaksi dengan ovo sangat aman dan mudah tapi semalam mungkin ada gangguan jaringan dan saya ada coba isi pulsa tapi kok sampai sekarang belum masuk pulsanya tapi saldonya kepotong saya harus bagaimana iya mohon solusinya terima kasih	0
terima kasih buat aplikasi ovo tingkatkan layanan untuk kedepan yang lebih baik	1
awalnya suka banget pakai ovo tapi kesininya topup melalui oneklik keterangan tidak berhasil dan tidak bertambah tapi saldo di atm bahkan terpotong terus transfer ke rekening berhasil tapi di rekening bahkan tidak ada yang masuk uangnya	0

Pembagian *dataset* (*splitting data*) dilakukan setelah proses pelabelan data, yaitu dengan membagi *dataset* yang digunakan dengan perbandingan yang telah ditentukan. *Splitting data* dengan membagi data sebanyak 35772 ke dalam 3 (tiga) kategori, yaitu data latih, data validasi, dan data uji. Data yang digunakan memiliki persebaran data yang sama pada setiap kelas, yaitu sebanyak 17886 data seperti yang dapat dilihat pada Gambar 3.14.



Gambar 3. 14 Kuantitas Data pada Setiap Selas

Langkah pertama adalah mengambil 386 data pada masing-masing kelas untuk menjadi data uji, atau total sebanyak 772 data. Hasil pengambilan data uji dapat dilihat pada Gambar 3.15.

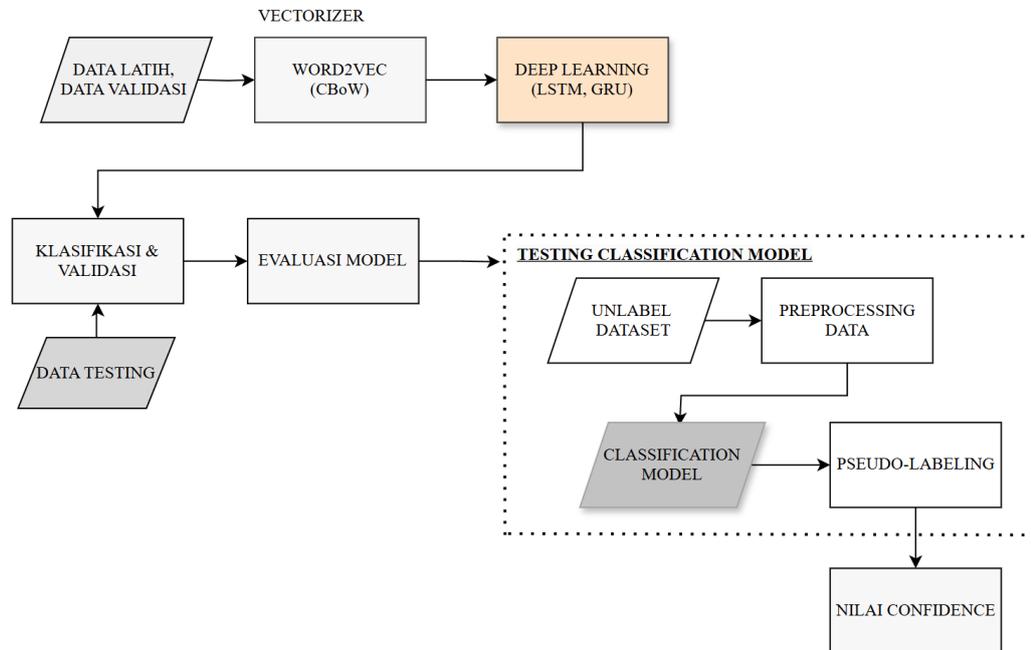


Gambar 3. 15 Hasil Pengambilan Data Uji

Kemudian sebanyak 35000 data dibagi menjadi data latih dan data validasi dengan perbandingan 80:20. Sebelum dilakukan pembagian ke data latih dan data validasi, data dilakukan cek terhadap data duplikat dan data null, serta menyeragamkan kuantitas data pada masing-masing kelas dengan undersampling. Diperoleh banyak data yang akan dibagi ke data latih dan data validasi sebanyak

34884 data. Dengan data tersebut, didapatkan data latih sebanyak 27907 data, dan data validasi sebanyak 6977 data.

### 3. SSL Model



Gambar 3. 16 Alur Metode *Semi-Supervised Learning*

Tahap *Semi-Supervised Learning* dimulai dengan menentukan dataset yang akan digunakan sebagai masukan (input). Dataset yang telah melewati tahap *preprocessing* dan tahap *binary labeling*, kemudian dibagi ke dalam data latih, data validasi, dan data uji. Ketiga kategori data tersebut, masing-masing masuk dalam tahap vektorisasi yang mengubah kata menjadi bentuk vektor numerik. Tahap ini menggunakan metode Word2Vec, dengan model Continuous Bag of Word (CBoW). Model CBoW dilatih dengan 120 epoch dengan hyperparameter `vector_size=100`, `window=5`, `min_count=1`, `sg=0`. Contoh Hasil Word Embedding dapat dilihat pada Tabel 3.11.

Tabel 3. 11 Contoh Hasil Word Embedding

"Menginformasikan"						
[-1.1146793	-2.036982	-1.0926849	1.181311	-2.7451162	0.47781125	-
1.391421	0.03949102	-1.603203	-4.4339194	0.34133235	-1.4076655	-
1.6707718	3.3340545	0.75184083	1.6315496	0.07985914	-1.3940139	-
1.9031693	-1.2049723	-2.0477335	-3.0716445	2.4264019	-0.9936188	-

1.3986042	2.2230315	-1.6504574	2.1592934	-0.5459352	-0.56055814	-
0.23404081	2.739908	0.6424688	3.2121453	2.5273411	2.6440668	-
2.7324064	0.5456563	-0.48760468	-0.5503142	1.7364691	-3.0896313	
0.65123075	-2.368174	-1.2880491	-3.5399575	-1.2917658	0.9785995	
-1.1625224	1.2963524	1.0063558	-0.9055597	0.7286115	1.3266513	
1.8085691	1.847696	-1.6065309	1.7667304	1.7213436	2.8100984	
-4.0163636	-1.858917	-2.4638767	-3.5802796	0.2284	-0.9511405	
3.7483032	-0.9219694	-1.5256355	-0.20528443	-1.5243223	0.30571774	
1.4553732	0.2808095	-1.3382924	-0.4455473	2.5675452	0.94929945	
-2.7980907	0.35942116	-0.9226401	0.17971843	-0.7181038	-1.984855	
-1.8403093	-0.2540716	-1.8947273	-0.38741162	-3.7220397	-3.2762003	
1.0757284	2.8561232	-0.16373113	0.39178407]			

Vektor yang telah dibentuk akan digunakan dalam membangun algoritma LSTM dan GRU. Parameter yang digunakan dalam pemodelan LSTM yaitu *learning rate* dari fungsi optimasi Adam, *epoch*, *batch size*, koefisien regularisasi ( $\lambda$ ), dan kernel inisialisasi. Penggunaan hyperparameter yang berbeda akan memengaruhi kinerja dan akurasi model dalam tugas klasifikasi. Susunan layer algoritma *Deep Learning* dapat dilihat pada Tabel 3.12.

Tabel 3. 12 Susunan Layer Algoritma *Deep Learning*

	LSTM	GRU
<i>activation</i>	Sigmoid	-
Jumlah Unit	128, 64	128, 64
Dropout	0.2	0.2
Recurrent Dropout	0.2	0.2
Kernel Inisialisasi	-	he-normal

Sedangkan susunan nilai hyperparameter yang digunakan selama percobaan penelitian, lebih lanjut terdapat pada Tabel 3.13.

Tabel 3. 13 Nilai Hyperparameter Model *Deep Learning*

Hyperparameter	Nilai
Learning Rate Optimasi	0.001 dan 0.002
Epoch	50 dan 100
Batch Size	128, 256, dan 512
Koefisien Regularisasi ( $\lambda$ )	0.001 dan 0.001, 0.005 dan 0.01

Lebih lanjut skenario pemodelan algoritma berdasarkan hyperparameter yang sudah ditentukan, yaitu untuk skenario pemodelan algoritma LSTM dapat dilihat pada Tabel 3.14, dan skenario pemodelan algoritma GRU pada Tabel 3.15.

Tabel 3. 14 Skenario Pemodelan Algoritma LSTM

Learning Rate (LR)	Koefisien Regularisasi ( $\lambda$ )	Epoch	Batch Size	Skenario
0.002	0.001 dan 0.001	50	128	<i>0.002-LSTM-50-128</i>
			256	<i>0.002- LSTM-50-256</i>
			512	<i>0.002- LSTM-50-512</i>
		100	128	<i>0.002- LSTM-100-128</i>
			256	<i>0.002- LSTM-100-256</i>
			512	<i>0.002- LSTM-100-512</i>
0.001	0.005 dan 0.01	50	128	<i>0.001- LSTM-50-128</i>
			256	<i>0.001- LSTM-50-256</i>
			512	<i>0.001- LSTM-50-512</i>
		100	128	<i>0.001- LSTM-100-128</i>
			256	<i>0.001-LSTM-100-256</i>
			512	<i>0.001-LSTM-100-512</i>

Tabel 3. 15 Skenario Pemodelan Algoritma GRU

Learning Rate (LR)	Koefisien Regularisasi ( $\lambda$ )	Epoch	Batch Size	Skenario
0.002	0.001 dan 0.001	50	128	<i>0.002-GRU-50-128</i>
			256	<i>0.002-GRU-50-256</i>
			512	<i>0.002-GRU-50-512</i>
		100	128	<i>0.002-GRU-100-128</i>
			256	<i>0.002-GRU-100-256</i>
			512	<i>0.002-GRU-100-512</i>
0.001	0.005 dan 0.01	50	128	<i>0.001-GRU-50-128</i>
			256	<i>0.001-GRU-50-256</i>
			512	<i>0.001-GRU-50-512</i>
		100	128	<i>0.001-GRU-100-128</i>
			256	<i>0.001-GRU-100-256</i>
			512	<i>0.001-GRU-100-512</i>

Setelah mendapatkan model klasifikasi (LSTM dan GRU) yang telah dilatih, kemudian dilakukan pengujian pada data uji yang telah divektorisasi. Evaluasi model menggunakan confusion matrix dimana dapat diketahui banyak data yang benar dan salah diprediksi oleh model.

Sedangkan tahap *testing classification model* adalah tahap pengujian model LSTM dan GRU menggunakan *dataset* yang berbeda. *Dataset* yang digunakan adalah didapatkan dari Google PlayStore *review* aplikasi GoPay. Dimulai dengan *preprocessing dataset* tanpa label, dilanjutkan dengan mendefinisikan model yang telah dilatih dalam tahap pelatihan yang akan diimplementasikan terhadap *unlabel dataset* untuk mendapatkan *pseudo-label*. Setelah *pseudo-label* didapatkan, dilakukan penghitungan nilai *confidence model* dan nilai akurasi menggunakan confusion matrix.