

BAB III

HASIL DAN PEMBAHASAN

3.1 Hasil

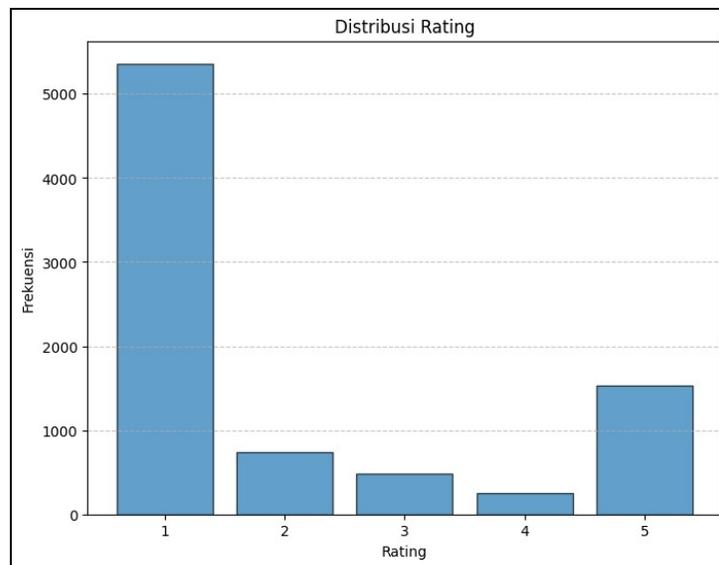
3.1.1 Pengambilan Data

Tahapan pertama yaitu pengambilan data melalui proses *scraping*. Pada prosesnya data *scraping* dimasukan ke *dataframe pandas*. Gambar 3.1 merupakan hasil dari *scraping* yang telah dilakukan proses *filtering* kolom, sehingga terdapat 5 kolom yang digunakan.

	userName	score	at	content	thumbsUpCount
6671	Nency Miranda	5	2024-02-05 14:07:12	Mantap	0
5783	ichal sabiel	1	2024-02-05 14:06:17	Sudah tau pemakainya masyarakat biasa Malah sp...	0
7909	Evandra Aditya	1	2024-02-05 14:06:16	Apk nya nggk bisa buat log in	0
6307	Ganesh insann	1	2024-02-05 14:05:38	Susah masuk dih	0
8008	Yohana Frediana	1	2024-02-05 14:04:32	Tidak bisa masuk inisiliasi	0

Gambar 3. 1 Hasil *Scraping* Data

Pada 5 kolom diatas terdapat nama pengguna, rating, waktu, komentar, dan jumlah like. Selanjutnya, gambar 3.2 merupakan hasil dari distribusi rating yang digambarkan dengan visualisasi *chart*. Data yang didapatkan dari proses *scraping* sebelumnya yaitu 8358 *dataset*. kemudian, dari hasil tersebut terdapat frekuensi data terbanyak pada *rating* 1, yaitu sebanyak 5361 ulasan. Dan frekuensi data yang paling sedikit terdapat pada *rating* 4, sebanyak 255 ulasan.



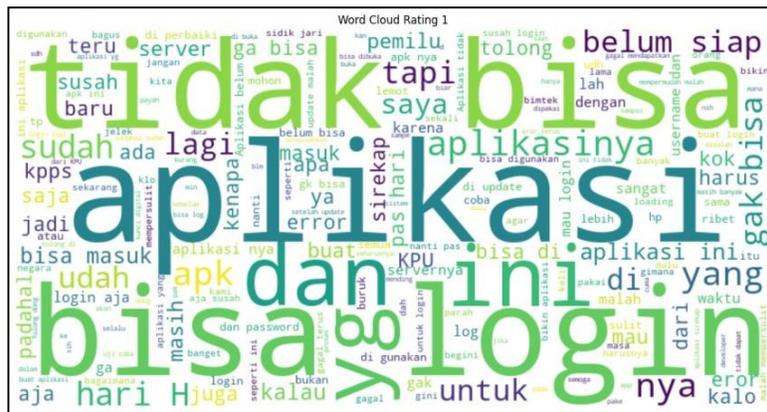
Gambar 3. 2 Hasil Distribusi *Rating*

Penjelasan hasil frekuensi setiap *rating* terdapat pada tabel 3.1 dibawah ini. Dari tabel tersebut dapat dilihat hasil frekuensi setiap *rating* mulai dari yang terbanyak, sedang dan yang paling sedikit.

Tabel 3. 1 Frekuensi Masing-Masing *Rating*

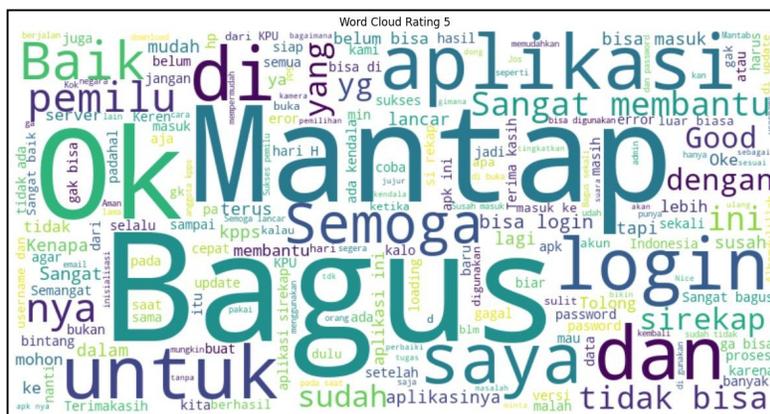
<i>Rating</i>	Frekuensi
1	5361
2	736
3	478
4	255
5	1528
Total	8358

Dari tabel diatas, terdapat frekuensi data ulasan dari setiap *rating* memiliki jumlah yang berbeda. *Rating* 1 memiliki frekuensi 5361 data ulasan, kemudian pada *rating* 2 terdapat 736 ulasan, *rating* 3 terdapat 478 ulasan, *rating* 4 terdapat 255 ulasan, dan *rating* 5 yaitu sebanyak 1528 ulasan.



Gambar 3. 3 Hasil *Wordcloud rating 1*

Selanjutnya, untuk memvisualisasikan frekuensi kata yang sering muncul pada ulasan *rating* 1, dengan menggunakan teknik “wordcloud”. Pada gambar 3.3 diatas, frekuensi kata yang sering muncul dan yang paling banyak dibahas pada *rating* 1 adalah “aplikasi”, “bisa”, “login”, “tidak”, “bisa” maka konteks yang sering muncul pada *rating* 1 yaitu “tidak bisa login aplikasi”.



Gambar 3. 4 Hasil *Wordcloud rating 5*

Sedangkan untuk frekuensi kata yang paling sering muncul pada ulasan *rating* 5 yaitu “Ok”, “Bagus” dan “Mantap”. Maka konteks yang sering muncul dan yang paling banyak dibahas pada *rating* 5 yaitu “aplikasinya bagus”.

3.1.2 Pra Proses Data

Tahap pertama pra proses yaitu *add id*. Proses ini memberikan identifikasi unik pada setiap baris. Panjang *id* ditentukan berdasarkan jumlah digit dari total jumlah baris. Berdasarkan gambar dibawah, jika data memiliki baris 8358 baris, maka *ID* nya ‘d0001’, ‘d0002’ hingga ‘d8357’. Proses ini memastikan bahwa setiap baris memiliki *id* yang mudah diidentifikasi dan terstruktur secara konsisten.

ID	Nama	Rating	Waktu	Komentar	Like
0	d0001	Nency Miranda	5 05/02/2024 14:07		
1	d0002	ichal sabiel	1 05/02/2024 14:06		
2	d0003	Evandra Aditya	1 05/02/2024 14:06		
3	d0004	Ganesh insann	1 05/02/2024 14:05		
4	d0005	Yohana Frediana	1 05/02/2024 14:04		
...		
8353	d8354	Tn A Muntaha	5 23/01/2024 09:05		
8354	d8355	Adhi Nugroho	1 23/01/2024 08:51		
8355	d8356	TATANG RUSDIANA	5 23/01/2024 07:03		
8356	d8357	vie	5 23/01/2024 06:50		
8357	d8358	Ahmad ferdiansyah	5 23/01/2024 04:26		
0				Mantap	0.0
1				Sudah tau pemakainya masyarakat biasa Malah sp...	0.0
2				Apk nya nggk bisa buat log in	0.0
3				Susah masuk dih	0.0
4				Tidak bisa masuk inisiliasi	0.0
...			
8353				Semoga Tambah Baik,Mudah Dalam Input Dan Share...	21.0
8354				Cara masuk nya bagaimana? Pakai email kok gag ...	448.0
8355				Mbuh	6.0
8356				Semoga tidak seperti sirekap 2020 yang saat di...	42.0
8357				Semoga dengan adanya aplikasi ini pemilu akan ...	186.0

[8358 rows x 6 columns]

Gambar 3. 5 Hasil *Add Id*

Gambar 3.5 diatas merupakan hasil dari *add id*, dari 8358 baris, dan 6 kolom. Pelabelan berdasarkan komentar dari pengguna “Sirekap 2024”. Label *id* yang dimulai dari baris 0 hingga 8357.

	Mantap	komentar_lowercase
0		mantap
1	Sudah tau pemakainya masyarakat biasa Malah sp...	sudah tau pemakainya masyarakat biasa malah sp...
2	Apk nya nggk bisa buat log in	apk nya nggk bisa buat log in
3	Susah masuk dih	susah masuk dih
4	Tidak bisa masuk inisiliasi	tidak bisa masuk inisiliasi
...
8353	Semoga Tambah Baik,Mudah Dalam Input Dan Share...	semoga tambah baik,mudah dalam input dan share...
8354	Cara masuk nya bagaimana? Pakai email kok gag ...	cara masuk nya bagaimana? pakai email kok gag ...
8355	Mbuh	mbuh
8356	Semoga tidak seperti sirekap 2020 yang saat di...	semoga tidak seperti sirekap 2020 yang saat di...
8357	Semoga dengan adanya aplikasi ini pemilu akan ...	semoga dengan adanya aplikasi ini pemilu akan ...

[8358 rows x 2 columns]
Waktu proses: 0.0249330997467041 detik

Gambar 3. 6 Sebelum dan Setelah *Lowercase*

Tahap berikutnya mengubah semua data di kolom ‘komentar’ menjadi huruf kecil. Seperti pada komentar dengan kata “Semoga” diubah menjadi “semoga”, huruf “S” yang awalnya huruf kapital berubah menjadi huruf kecil “s”. Pada penelitian ini hanya menggunakan kolom ‘komentar’ untuk proses klasifikasi teks dari ulasan pengguna. Gambar 3.6 gambar sebelah kiri merupakan sebelum *lowercase dataset* dan sebelah kanan merupakan hasil setelah proses *lowercase*. Proses ini membutuhkan durasi waktu untuk merubah data teks komentar menjadi huruf kecil selama 0,009 detik.

<pre> komentar_lowercase \ 0 mantap 1 sudah tau pemakainya masyarakat biasa malah sp... 2 apk nya nggk bisa buat log in 3 susah masuk dih 4 tidak bisa masuk inisiliasi ... 8353 semoga tambah baik,mudah dalam input dan share... 8354 cara masuk nya bagaimana? pakai email kok gag ... 8355 mbuh 8356 semoga tidak seperti sirekap 2020 yang saat di... 8357 semoga dengan adanya aplikasi ini pemilu akan ... </pre>	<pre> komentar_remove_char 0 mantap 1 sudah tau pemakainya masyarakat biasa malah sp... 2 apk nya nggk bisa buat log in 3 susah masuk dih 4 tidak bisa masuk inisiliasi ... 8353 semoga tambah baik mudah dalam input dan share... 8354 cara masuk nya bagaimana pakai email kok gag b... 8355 mbuh 8356 semoga tidak seperti sirekap 2020 yang saat di... 8357 semoga dengan adanya aplikasi ini pemilu akan ... [8358 rows x 2 columns] Waktu proses: 0.4594733715057373 detik </pre>
--	--

Gambar 3. 7 Hasil Sebelum dan Setelah *Remove Char*

Tahap berikutnya yaitu membersihkan data dari karakter-karakter yang tidak diperlukan seperti simbol, emoji, angka dengan menggunakan fungsi *'remove_unnecessary_char'*. Dari gambar 3.7 diatas pada baris 8353 yang sebelumnya terdapat tanda baca ',' dan pada baris 8354 yang sebelumnya terdapat tanda baca '?' setelah proses *remove char* tanda tersebut hilang. Gambar 3.7 sebelah kiri merupakan sebelum *remove char* dan gambar sebelah kanan merupakan hasil dari *remove char*. Durasi proses ini selama 0,459 detik.

<pre> komentar_remove_char \ 0 mantap 1 sudah tau pemakainya masyarakat biasa malah sp... 2 apk nya nggk bisa buat log in 3 susah masuk dih 4 tidak bisa masuk inisiliasi ... 8353 semoga tambah baik mudah dalam input dan share... 8354 cara masuk nya bagaimana pakai email kok gag b... 8355 mbuh 8356 semoga tidak seperti sirekap 2020 yang saat di... 8357 semoga dengan adanya aplikasi ini pemilu akan ... </pre>	<pre> komentar_spellchecker 0 mantap 1 sudah tahu pemakainya masyarakat biasa malah s... 2 aplikasi nya nggk bisa buat masuk in 3 susah masuk dih 4 tidak bisa masuk inisialisasi ... 8353 semoga tambah baik mudah dalam memasukkan dan ... 8354 cara masuk nya bagaimana pakai email kok gag b... 8355 mbuh 8356 semoga tidak seperti sirekap 2020 yang saat di... 8357 semoga dengan adanya aplikasi ini pemilu akan ... [8358 rows x 2 columns] Waktu proses: 0.07879090309143066 detik </pre>
--	---

Gambar 3. 8 Hasil Sebelum dan Setelah *Spellchecker*

Selanjutnya pada tahap ini sebelum masuk tahap *spellchecker*, terlebih dahulu membuat kamus tidak baku yang berlandaskan dari Kamus Bahasa Indonesia (KBI) yang kemudian dimasukkan kedalam proses *spellchecker*. Sebagai contoh pada salah satu ulan pada gambar 3.8, sebelah kiri kata yang sebelumnya "apk" berubah menjadi "aplikasi". Proses *spellchecker* membutuhkan waktu proses kurang lebih 0,078 detik.

<pre> komentar_remove_char \ 0 mantap 1 sudah tau pemakainya masyarakat biasa malah sp... 2 apk nya nggk bisa buat log in 3 susah masuk dih 4 tidak bisa masuk inisiliasi ... 8353 semoga tambah baik mudah dalam input dan share... 8354 cara masuk nya bagaimana pakai email kok gag ... 8355 mbuh 8356 semoga tidak seperti sirekap 2020 yang saat di... 8357 semoga dengan adanya aplikasi ini pemilu akan ... </pre>	<pre> komentar_stemming 0 mantap 1 sudah tahu maka masyarakat biasa malah spesifi... 2 aplikasi nya nggk bisa buat masuk in 3 susah masuk dih 4 tidak bisa masuk inisial ... 8353 moga tambah baik mudah dalam masuk dan share data 8354 cara masuk nya bagaimana pakai email kok gag b... 8355 mbuh 8356 moga tidak seperti sirekap 2020 yang saat guna... 8357 moga dengan ada aplikasi ini milu akan lebin b... [8358 rows x 2 columns] Waktu proses: 424.67806243896484 detik </pre>
---	---

Gambar 3. 9 Hasil Sebelum dan Setelah *Stemming*

Pada tahapan ini merubah setiap kata pada data menjadi bentuk dasar. Bentuk dasar kata yang diubah berlandaskan dari kamus tidak baku yang telah dibuat dari proses *spellchecker*. Proses *stemming* ini guna menormalisasi data teks. Pada gambar 3.9 sebelah kanan merupakan hasil

setelah *stemming*, terdapat pada ulasan yang awalnya kata “semoga” berubah menjadi kata dasarnya yaitu “moga”. Proses ini memakan waktu kurang lebih selama 7 menit.

3.1.3 WordNet

Pada tahapan ini peneliti memasukkan *library`deep_translator`* untuk menerjemahkan teks dari bahasa Indonesia ke bahasa Inggris. Karena *WordNet* hanya dapat memproses teks dalam bahasa Inggris.

<pre> 100% ██████████ 8302/8302 [1:09:46<00:00, 1.98it/s] komentar_stemming \ 0 mantap 1 sudah tahu maka masyarakat biasa malah spesifi... 2 aplikasi nya nggk bisa buat masuk in 3 susah masuk dih 4 tidak bisa masuk inisial 8297 moga tambah baik mudah dalam masuk dan share data 8298 cara masuk nya bagaimana pakai email kok gag b... 8299 mbuh 8300 moga tidak seperti sirekap 2020 yang saat guna... 8301 moga dengan ada aplikasi ini milu akan lebih b... </pre>	<pre> Translated Excellent 0 1 You already know that ordinary people even hav... 2 the application can't log in 3 it's hard to get in 4 Can't enter initials 8297 Hopefully it will be easier to enter and share... 8298 How do I log in using email? How come it doesn... 8299 Mbuh 8300 Hopefully it won't be like the 2020 recap wher... 8301 Hopefully with this application Milu will be m... [8302 rows x 2 columns] Waktu proses: 4189.570982217789 detik </pre>
--	--

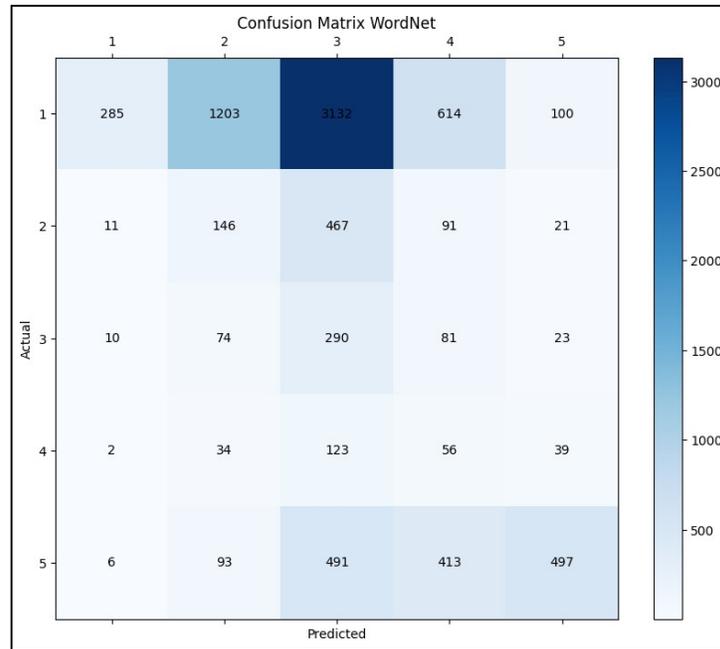
Gambar 3. 10 Hasil Sebelum dan Setelah *Translated*

Pada Gambar 3.10 sebelah kiri merupakan proses sebelum *translated* sedangkan gambar sebelah kanan merupakan hasil setelah *translated* dari bahasa Indonesia ke bahasa Inggris. Proses ini memakan waktu yang paling lama kurang lebih selama 1 jam 9 menit.

ID	Rating	komentar_stemming	Translated	Score_Wordnet	Sentiment_Score	
0	d0001	5	mantap	Excellent	5	1.000000
1	d0002	1	sudah tahu maka masyarakat biasa malah spesifi...	You already know that ordinary people even hav...	3	-0.045000
2	d0003	1	aplikasi nya nggk bisa buat masuk in	the application can't log in	3	0.000000
3	d0004	1	susah masuk dih	it's hard to get in	2	-0.291667
4	d0005	1	tidak bisa masuk inisial	Can't enter initials	3	0.000000

Gambar 3. 11 Hasil Hitung *Fungsi Score*

Selanjutnya pada hitung *score*, kolom '*komentar_stemming*' berisi ulasan asli pengguna yang telah melalui proses *stemming* sebelumnya. Dari gambar 3.11 diatas pada *id* 'd0001' dengan '*Score_Wordnet*' 5 dan '*Sentiment_Score*' '1,000000' yang berarti ulasan tersebut positif, dan pada *id* 'd0003' dengan '*Score_Wordnet*' 2 dan '*Sentiment_Scorre*' '-0,291667' yang berarti ulasan tersebut negatif, dan *id* 'd0005' dengan '*Score_Wordnet*' 3 dan '*Sentiment_Scorre*' '0,000000' yang berarti ulasan tersebut netral.



Gambar 3. 12 Hasil *Confusion Matrix Wordnet*

Selanjutnya, pada gambar 3.12 klasifikasi tertinggi terdapat pada warna biru gelap (*Navy*) yaitu sebanyak 3132 data *test* dengan nilai aktual *rating* 1, yang diklasifikasikan kedalam *rating* 3. Untuk klasifikasi paling rendah terdapat pada warna yang paling cerah (*light blue*) yaitu 2 data *test* dengan nilai aktual *rating* 4, yang diklasifikasikan pada *rating* 1.

Dari hasil gambar diatas, *rating* 1 yang berhasil diklasifikasikan benar sebanyak 285 data *test*, *rating* 2 yang berhasil diklasifikasikan benar sebanyak 146 data *test*, *rating* 3 berhasil diklasifikasikan benar sebanyak 290 data *test*, pada *rating* 4 yang berhasil diklasifikasikan benar 56 data *test*, dan yang terakhir pada *rating* 5 yang berhasil diklasifikasikan benar yaitu sebanyak 497 data *test*. Maka disimpulkan dari klasifikasi tersebut, menunjukkan banyak kesalahan besar, terutama dalam memprediksi *rating* 1 yang diklasifikasikan kedalam *rating* 3 yaitu sebanyak 3132 data *test*, sehingga hasil nilai performa menggunakan *F1-Score* sebesar 17,50%.

3.1.4 Klasifikasi Algoritma *K-Nearest Neighbor*

Pada tahap pertama klasifikasi ini, yaitu ekstraksi fitur TF-ID, proses dilakukan menggunakan library '*TfidfVectorizer*' yang berfungsi untuk mengubah teks menjadi fitur numerik berbasis TF-IDF dan untuk memasukkan nilai X.

```

Output exceeds the size limit. Open the
(0, 2046) 1.0
(1, 337) 0.6621262182383661
(1, 1350) 0.10794256266980186
(1, 266) 0.07468926169164286
(1, 2502) 0.1853482486775841
(1, 3561) 0.13356755666103043
(1, 3421) 0.2727733363721146
(1, 1250) 0.16099380470631008
(1, 3199) 0.3079382129849356
(1, 2018) 0.13570334989730748
(1, 519) 0.2437544763551553
(1, 2074) 0.282916933920367
(1, 2003) 0.2677440738604361
(1, 3269) 0.22400101010037562
(1, 3225) 0.1314431617042291
(2, 1335) 0.4310747504702509
(2, 2071) 0.2032426370354838
(2, 635) 0.35660872034218255
(2, 543) 0.2183617796441874
(2, 2348) 0.6835613818816532
(2, 2419) 0.31252212486134207
(2, 266) 0.18190410230834322
(3, 858) 0.9228114853920112
(3, 3249) 0.3231334539112743
(3, 2071) 0.2097706685691787
...
(8301, 1590) 0.2673947794830787
(8301, 1350) 0.16534233571085172
(8301, 266) 0.11440618672721325
(8301, 2074) 0.43336146103591516

```

Gambar 3. 13 Hasil Ekstraksi Fitur TF-IDF

Dari gambar 3.13 setiap baris menampilkan indeks, *term*, kemudian nilai TF-IDF, seperti pada baris awal terdapat *index* 0 dengan kata ‘mantap’ berada pada *term* 2046 dengan nilai TF-IDF 1,0. Pada baris kedua terdapat *index* 1 dengan kata ‘badut’, berada pada *term* 337 dan nilai TF-IDF 0,6621.

Selanjutnya yaitu tahap *cross validation* yang menampilkan isi data setiap pembagian *fold* yang berisi *train data*, *test data*, *train index* dan *test index*. Terdapat perbedaan hasil pada *test data* di beberapa *fold*. *Fold* 1-2 memiliki *test data* 831 *samples*, sedangkan *fold* 3-10 memiliki *test data* 830 *samples*. Hasil *test index* yang berbeda karena pembagian data ke *fold* yang berbeda-beda.

```

Fold 1:
- Train data: 7471 samples
- Test data: 831 samples
- Train Index: [ 831 832 833 ... 8299 8300 8301]
- Test Index: [ 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35
36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53
54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71
72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89
90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107
108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125
126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143
144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161
162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179
180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197
198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215
216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233
234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251
252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269
270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287
288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305
306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323
324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341
342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359
360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377
...
8256 8257 8258 8259 8260 8261 8262 8263 8264 8265 8266 8267 8268 8269
8270 8271 8272 8273 8274 8275 8276 8277 8278 8279 8280 8281 8282 8283
8284 8285 8286 8287 8288 8289 8290 8291 8292 8293 8294 8295 8296 8297
8298 8299 8300 8301]

```

Gambar 3. 14 Hasil Nilai *K-Fold 1*

Dari gambar 3.14 yang merupakan hasil dari nilai *K-fold 1* terdapat *train data* sebanyak 7471 *samples* sebagai data latih model, *test data* sebanyak 831 *samples* sebagai data uji model, untuk *test data* yang digunakan pada *fold 1* dari *index 0* sampai 8301.

```

Output exceeds the size limit. Open the full output data in a text editor
Fold 10:
- Train data: 7472 samples
- Test data: 830 samples
- Train Index: [ 0 1 2 ... 7469 7470 7471]
- Test Index: [7472 7473 7474 7475 7476 7477 7478 7479 7480 7481 7482 7483 7484 7485
7486 7487 7488 7489 7490 7491 7492 7493 7494 7495 7496 7497 7498 7499
7500 7501 7502 7503 7504 7505 7506 7507 7508 7509 7510 7511 7512 7513
7514 7515 7516 7517 7518 7519 7520 7521 7522 7523 7524 7525 7526 7527
7528 7529 7530 7531 7532 7533 7534 7535 7536 7537 7538 7539 7540 7541
7542 7543 7544 7545 7546 7547 7548 7549 7550 7551 7552 7553 7554 7555
7556 7557 7558 7559 7560 7561 7562 7563 7564 7565 7566 7567 7568 7569
7570 7571 7572 7573 7574 7575 7576 7577 7578 7579 7580 7581 7582 7583
7584 7585 7586 7587 7588 7589 7590 7591 7592 7593 7594 7595 7596 7597
7598 7599 7600 7601 7602 7603 7604 7605 7606 7607 7608 7609 7610 7611
7612 7613 7614 7615 7616 7617 7618 7619 7620 7621 7622 7623 7624 7625
7626 7627 7628 7629 7630 7631 7632 7633 7634 7635 7636 7637 7638 7639
7640 7641 7642 7643 7644 7645 7646 7647 7648 7649 7650 7651 7652 7653
7654 7655 7656 7657 7658 7659 7660 7661 7662 7663 7664 7665 7666 7667
7668 7669 7670 7671 7672 7673 7674 7675 7676 7677 7678 7679 7680 7681
7682 7683 7684 7685 7686 7687 7688 7689 7690 7691 7692 7693 7694 7695
7696 7697 7698 7699 7700 7701 7702 7703 7704 7705 7706 7707 7708 7709
7710 7711 7712 7713 7714 7715 7716 7717 7718 7719 7720 7721 7722 7723
7724 7725 7726 7727 7728 7729 7730 7731 7732 7733 7734 7735 7736 7737
7738 7739 7740 7741 7742 7743 7744 7745 7746 7747 7748 7749 7750 7751
7752 7753 7754 7755 7756 7757 7758 7759 7760 7761 7762 7763 7764 7765
...
8256 8257 8258 8259 8260 8261 8262 8263 8264 8265 8266 8267 8268 8269
8270 8271 8272 8273 8274 8275 8276 8277 8278 8279 8280 8281 8282 8283
8284 8285 8286 8287 8288 8289 8290 8291 8292 8293 8294 8295 8296 8297
8298 8299 8300 8301]

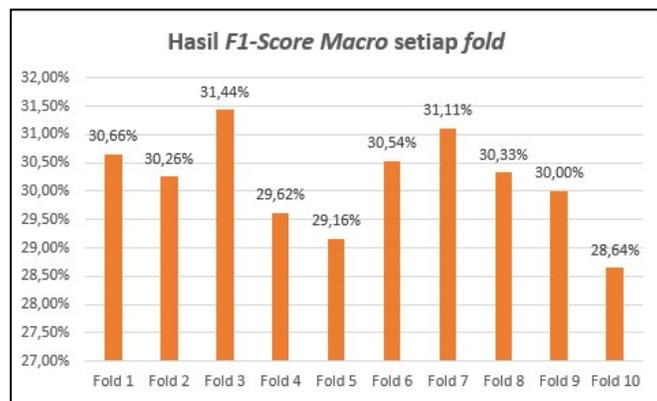
```

Gambar 3. 15 Hasil Nilai *K-Fold 10*

Sedangkan hasil dari *K-fold 10* pada gambar 3.15, *train data 7472 samples* sebagai data latih model, *test data 830 samples* sebagai data uji model, untuk *test data* yang digunakan pada *fold 10* dari *index 7472* sampai 8301. Fungsi dari proses *cross validation* yaitu untuk mengevaluasi performa model klasifikasi yang digunakan untuk mengklasifikasikan sentimen dari teks (ulasan “Sirekap 2024”).

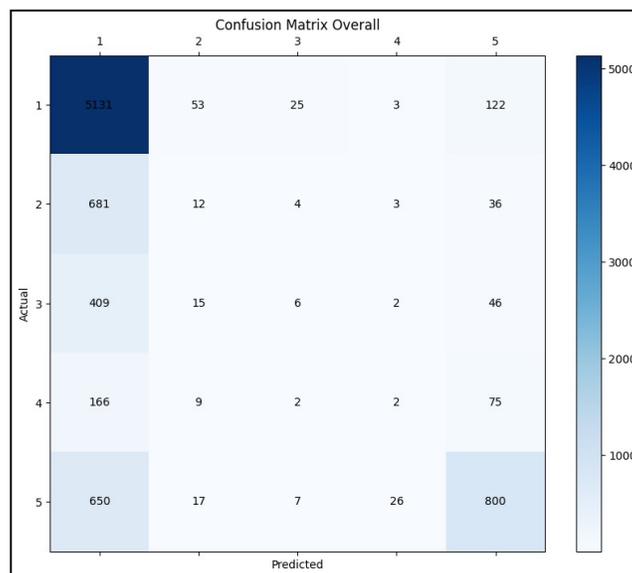
Tabel 3. 2 Hasil *F1 Score* Setiap *Fold*

<i>Fold</i>	<i>F1-Score</i>
<i>Fold-1</i>	30,66%
<i>Fold-2</i>	30,26%
<i>Fold-3</i>	31,44%
<i>Fold-4</i>	29,62%
<i>Fold-5</i>	29,16%
<i>Fold-6</i>	30,54%
<i>Fold-7</i>	31,11%
<i>Fold-8</i>	30,33%
<i>Fold-9</i>	30,00%
<i>Fold-10</i>	28,64%



Gambar 3. 16 Hasil *F1-Score Macro* Setiap *Fold*

Dari tabel 3.2 dan gambar 3.16 merupakan hasil dari setiap *fold*, hasil yang paling tinggi terdapat pada *fold 3* yaitu sebesar 31,44%, sedangkan hasil yang paling rendah yaitu pada *fold 10* sebesar 28,64%,



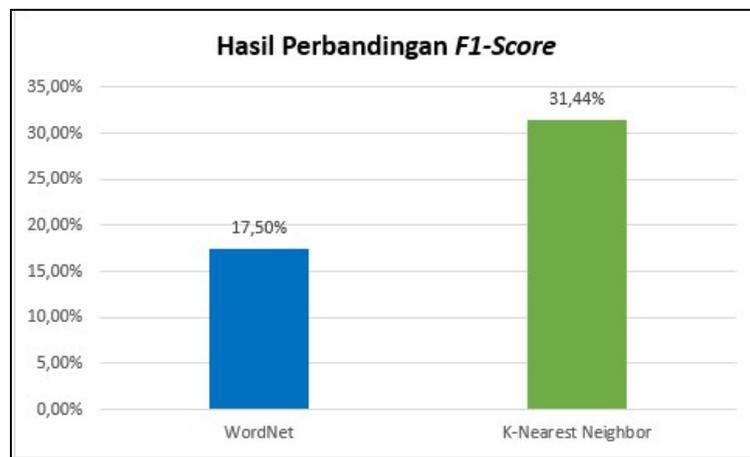
Gambar 3. 17 Hasil *Confusion Matrix Overall*

Selanjutnya, pada gambar 3.17 klasifikasi tertinggi terdapat pada warna biru gelap (*Navy*) yaitu sebanyak 5131 data *test* dengan nilai aktual *rating* 1, yang diklasifikasikan kedalam *rating* 1. Untuk klasifikasi paling rendah terdapat pada warna yang paling cerah (*light blue*) yaitu 2 data *test* dengan nilai aktual *rating* 4, yang diklasifikasikan pada *rating* 3.

Dari hasil gambar diatas, *rating* 1 yang berhasil diklasifikasikan benar sebanyak 5131 data *test*, *rating* 2 yang berhasil diklasifikasikan benar sebanyak 12 data *test*, *rating* 3 berhasil diklasifikasikan benar sebanyak 6 data *test*, pada *rating* 4 yang berhasil diklasifikasikan benar 2 data *test*, dan yang terakhir pada *rating* 5 yang berhasil diklasifikasikan benar yaitu sebanyak 800 data *test*. Maka disimpulkan dari klasifikasi tersebut, menunjukan jumlah prediksi benar pada diagonal relatif tinggi. terutama dalam memprediksi *rating* 1 yang diklasifikasikan kedalam *rating* 1 yaitu sebanyak 5131 data *test*. Sehingga hasil nilai performa menggunakan *F1-Score* sebesar 31,44%.

Tabel 3. 3 Hasil Perbandingan 2 Metode

Metode	Hasil
<i>WordNet</i>	17,50%
<i>K-Nearest Neighbor</i>	31,44%



Gambar 3. 18 Hasil Perbandingan 2 Metode

Dari tabel 3.3 dan gambar 3.18 hasil yang diperoleh dari perbandingan kedua metode tersebut dalam melakukan klasifikasi data ulasan aplikasi “Sirekap 2024” menggunakan evaluasi *f1-score* menunjukkan, hasil evaluasi metode *K-Nearest Neighbor* lebih baik dibandingkan dengan hasil evaluasi metode *WordNet*.

3.2 Pembahasan

Berdasarkan penyajian data dan hasil analisis data, maka pada bab ini akan di deskripsikan temuan penelitian dan hasil pengujian yang telah diuji pada bab sebelumnya guna menjawab rumusan masalah yaitu, bagaimana hasil komparasi metode *WordNet* dan klasifikasi *K-Nearest Neighbor* menggunakan evaluasi *f1-score* pada ulasan aplikasi sirekap 2024. Pengambilan data pada penelitian ini yang berhasil di *scraping* sebanyak 8358 dataset dan menggunakan 5 kolom, namun data yang *discraping* tidak seimbang, seperti yang dapat dilihat pada distribusi *rating* gambar 3.2,

dimana *rating* satu memiliki dataset yang paling banyak yang berarti pengguna sangat tidak puas dengan aplikasi atau pelayanannya.

Pada tahap pra proses dalam penelitian ini untuk mempersiapkan data agar siap digunakan dalam analisis dan modeling. *Spellchecker* pada penelitian ini untuk memperbaiki kesalahan pengejaan dalam teks, namun pada hasil *spellchecker* terdapat kata yang tidak berhasil berubah sesuai dengan kamus yang telah dibuat, terdapat pada gambar 3.8 baris 2 dengan kata ‘nggk’ dan 8335 dengan kata ‘mbuh’ . Kemudian *stemming* yang berfungsi untuk merubah setiap kata pada data menjadi bentuk dasar, tetapi karena memotong kata ke bentuk dasarnya tanpa memperhatikan konteks, bisa menyebabkan kehilangan makna spesifik dari kata tersebut, terdapat pada gambar 3.9 baris 8356 dengan kata ‘semoga’ kemudian menjadi kata ‘moga’ dan baris 8357 dengan awalnya kata ‘pemilu’ menjadi kata ‘milu’.

Pada proses *WordNet*, dalam mengklasifikasikan hanya dengan melakukan perhitungan menggunakan nilai polaritas, dan dari nilai tersebut ulasan diklasifikasikan kedalam peringkat sesuai nilai yang dihasilkan, namun pada proses *translated WordNet* termasuk salah satu tahap yang memakan waktu cukup lama yaitu selama 1 jam 9 menit. Hasil performa wordnet menggunakan *f1-score* sangat rendah yaitu hanya sebesar 17,50% karena dalam mengklasifikasikan menggunakan evaluasi *f1-score* terdapat banyak kesalahan dalam memprediksi, yang dapat dilihat pada *confusion matriks* gambar 3.12, yaitu terutama dalam memprediksi *rating* 1 yang diklasifikasikan kedalam *rating* 3 yaitu sebanyak 3132 data *test* sedangkan yang diprediksi benar dalam *rating* 1 sampai 5 hanya sebanyak 1274 data *test*, dan yang paling banyak diklasifikasikan benar yaitu pada *rating* 1, sebanyak 285 data *test*.

Pada tahap KNN dalam terdapat tahap *cross validation* dengan nilai $K=10$, karena pada penelitian ini data yang digunakan tidak seimbang sehingga tahap *cross validation* untuk mengurangi resiko *overfitting*. hasil performa klasifikasi knn menggunakan *f1-score* lebih baik dari *WordNet* yaitu sebesar 31,44% karena dalam mengklasifikasikan menggunakan evaluasi *f1-score* lebih banyak data yang diprediksi benar, terutama dalam memprediksi *rating* 1 yang diklasifikasikan kedalam *rating* 1, yaitu sebanyak 5131 data *test*, sedangkan yang prediksi salah paling banyak pada *rating* 5 yang diklasifikasikan kedalam *rating* 1 yaitu sebanyak 650 data *test* . Hasil klasifikasi dapat dilihat pada *confusion matriks* gambar 3.17, untuk melihat hasil perbandingan dari dua metode terdapat pada tabel 3.3 dan gambar 3.18.