Peningkatan Performa Analisis Sentimen Dengan *Resampling* dan *Hyperparameter* pada Ulasan Aplikasi BNI Mobile

Wijanarto[1], Seviana Pungky Brilianti[2]
Program Studi Teknik Informatika
Universitas Dian Nuswantoro
Semarang, Indonesia
e-mail: [1]wijanarto@dsn.dinus.ac.id, [2]sevianapungkyb@gmail.com

Diajukan: 8 Januari 2020; Direvisi: 4 Maret 2020; Diterima: 11 Maret 2020

Abstrak

Penggunaan mobile banking meningkat seiring dengan kemajuan teknologi. Hampir setiap bank di Indonesia memiliki layanan mobile banking, termasuk BNI. Menurut survei dari Top Brand Award, BNI Mobile Banking menurun menjadi nomor 4 pada tahun 2016 dan 2017. Artinya terdapat relasi yang kuat antara ulasan pemakai aplikasi terhadap kinerja aplikasi. Dengan demikian membawa akibat pada pentingnya mempertahankan kualitas layanan serta kemampuan untuk bersaing dengan bank lain. Beberapa penelitian analisis sentimen sebelumnya belum melihat ketersediaan apakah dataset sudah dieksplorasi keseimbangannya atau tidak untuk meningkatkan performa model yang dipilih. Oleh karena itu, dalam artikel ini mencoba melakukan analisis sentimen pada ulasan pengguna aplikasi BNI Mobile Banking di Google Play sebanyak 6954 data terpilih dengan label positif dan negatif dan menggunakan 7 metode dasar sebagai baseline untuk dipilih satu yang mempunyai performa terbaik yitu Support Vector Classifier, setelah dilakukan resampling dataset dengan Repeated Edited Nearest Neighbours dan hyperparameter model C=1, degree=2 kernel poly didapatkan akurasi sebesar 98.54% pada data training dan akurasi 100% pada data uji. Selanjutnya dari 26 data mentah baru dilakukan eksperimen dan menghasilkan prediksi benar sebesar 19 sementara 7 salah dengan error rate sebesar 27%.

Kata kunci: *Analisis sentimen, BNI Mobile Banking, Resampling, Support vector classifier.*

Abstract

The use of mobile banking is increasing in line with technological advancements. Almost every bank in Indonesia has mobile banking services, including BNI. According to a survey from the Top Brand Award, BNI Mobile Banking dropped to number 4 in 2016 and 2017. This means that there is a strong relationship between application user reviews on application performance. As such it has an impact on the importance of maintaining service quality and the ability to compete with other banks. Some previous sentiment analysis studies have not looked at whether the dataset has been explored or not to improve the performance of the selected model. Therefore, in this article, we try to do sentiment analysis on the user reviews of BNI Mobile Banking applications on Google Play as many as 6954 selected data with positive and negative labels and use 7 basic methods as baselines to choose the one that has the best performance, Support Vector Classifier, after Resampling the dataset with Repeated Edited Nearest Neighbors and hyperparameter model C = 1, degree = 2 poly obtained accuracy of 98.54% in training data and 100% accuracy in test data. Furthermore, from 26 new raw data, the experiment was carried out and resulted in correct predictions of 19 while 7 were incorrect with an error rate of 27%.

Keywords*: Sentiment analysis, BNI Mobile Banking, Resampling, Support vector classifier.*

1. Pendahuluan

Pertumbuhan pengguna *mobile banking* naik dari empat bank (Mandiri, BCA, BNI, dan BRI) mencapai 23,65 juta pengguna pada tahun 2015 dan meningkat 25% dari 18,8 juta pengguna pada tahun 2014. Ini juga mempengaruhi salah satu bank yaitu bank BNI [1] . Selama tahun 2017, jumlah pengguna *mobile banking* BNI meningkat 170% dari 506 ribu pengguna ke 1.368 ribu pengguna. Serta dari jumlah transaksi mengalami peningkatan sebesar 207% dibanding tahun sebelumnya yaitu sebanyak 10,5 juta transaksi menjadi 32,3 juta transaksi [2]. Nilai transaksi pada BNI dengan menggunakan *mobile banking* BNI atau disebut BNI Mobile Banking sampai September 2018 mencapai Rp 90,7 triliun [3]. Ini merupakan

peningkatan dibanding pada tahun 2017. Namun seperti halnya usaha di mana perkembangan usaha ditentukan dengan menguasai pangsa pasar sebesar mungkin maka persaingan pasar perlu diperhatikan[4]. Menurut survei dari Top Brand Award, BNI Mobile Banking menjadi Top Brand nomor 3 pada tahun 2013 kemudian mengalami penurunan menjadi nomor 4 pada tahun 2014. Kemudian naik lagi menjadi Top Brand nomor 3 pada tahun 2015 dan mengalami penurunan menjadi nomor 4 ditahun 2016 dan 2017 [5]. Sehingga perlu dilakukan evaluasi terhadap layanan BNI Mobile Banking agar dapat selalu bersaing dengan pasar.

Ulasan pengguna atau konsumen sangat penting dalam meningkatkan suatu produk karena memiliki tujuan untuk meningkatkan dan mengevaluasi kualitas suatu produk [6]. Dalam hal ini ulasan pengguna dapat dijadikan sebagai analisis sentimen untuk mengetahui sentimen atau tanggapan dari pengguna. Ulasan pengguna BNI Mobile Banking dapat diambil dari Google Play maupun Appstore. Menurut survei Statcounter GlobalStats pada September 2018 pengguna sistem operasi pada perangkat bergerak di Indonesia didominasi oleh Android sebesar 92.27%, kemudian diikuti oleh iOS sebesar 4.87% dan sisanya merupakan pengguna yang lain [7]. Analisis sentimen telah banyak dilakukan baik pada bidang politik, berita, hiburan, maupun produk tertentu.

Dalam [8] dilakukan analisis sentimen *tweet* pada bidang politik dengan bahasa Indonesia pada pemilihan gubernur DKI 2017 dengan metode klasifikasi Support Vector Machine (SVM) dan Naïve Bayes Classifier (NBC) dengan akurasi tertinggi pada NBC sebesar 95% dan. Sementara dalam [9], analisis terhadap berita dilakukan untuk mendapatkan kesesuaian berita dengan minat pembacanya dengan Support Vector Regression menunjukkan tingkat akurasi dengan MAPE sebesar 0,8243075902233644%. Kemudian dalam [10] mencoba menganalisis *review* aplikasi *mobile* dengan algoritma klasifikasi Svm dan Naïve Bayes yang terbukti lebih baik dengan N-Gram=2 mendapatkan F-Test 0.967. Algoritma KNN dan SVM juga telah diaplikasikan untuk membandingkan analisa sentimen dalam [11] performa terbaik diperoleh algoritma KNN hingga 90% dan [12] membandingkan Logistic Regression, Stochastic Gradient Descent, Naive Bayes, dan Convolutional Neural Networks dengan Word2vec untuk ulasan di Amazon dengan akurasi terbaik 79.60% dan *lag loss*=0.52. Dalam [13] mencoba mengategorisasikan Google Mobile Application dengan Multinomial Naïve Bayes dengan akurasi 72%, sementara dalam [14] menghasilkan akurasi 82.92% dengan SVM dan [15] menganalisis berdasarkan seleksi fitur dengan LDA pada Google Apps Store dapat mereduksi 48% fitur yang tidak perlu. Sebagian besar penelitian di atas belum melihat ketersediaan apakah *dataset* sudah dieksplorasi keseimbangannya atau tidak untuk meningkatkan performa model yang dipilih. *Dataset* yang tidak seimbang, mengakibatkan pengukurannya semu, artinya tingginya akurasi yang dicapai sebenarnya tidak mewakili nilai data yang sebenarnya. Dengan demikian usaha peningkatan performa juga menjadi tidak berguna karena hasil pengukurannya terjebak pada akurasi yang semu. Untuk itulah perlu dilakukan *resampling* pada data yang tidak seimbang, sehingga kita dapat melakukan peningkatan performa (*tuning*) untuk mendapatkan model optimal yang dapat dilakukan dengan memberi parameter (*hyperparameter*) pada model terpilih. Dalam artikel ini akan di sajikan bagaimana melakukan peningkatan performa analisa sentimen untuk data yang tidak seimbang dengan *resampling* dan *hyperparameter* pada ulasan aplikasi *mobile* BNI yang ada di Google Play Store.

2. Metode Penelitian

Dalam penelitian ini akan diajukan kerangka kerja atau metode penelitian untuk menghasilkan peningkatan performa analisa sentimen dengan *resampling* dan *hyperparameter* pada ulasan Aplikasi BNI Mobile yang terdapat di Playstore dipaparkan sebagai berikut pada Gambar 1 yang terdiri dari beberapa langkah berurutan yang dijelaskan sebagai berikut.

2.1. *Dataset*

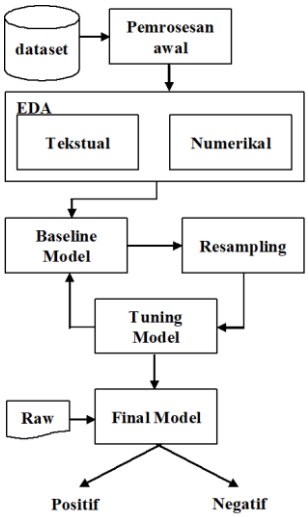
Data ulasan pengguna diambil dengan cara *webscrapping* menggunakan menulis secara *custom* berdasarkan pustaka Python yaitu Selenium dan BeautifulSoup dalam jangka waktu 5/28/2017 sampai dengan 11/26/2019 dan diperoleh data 8080 ulasan. Namun tidak semua digunakan, melainkan data yang terkumpul dipilih 6954 ulasan secara manual dengan bahasa Indonesia dan melakukan penerjemahan secara manual ke bahasa Indonesia jika mengandung bahasa Inggris maupun bahasa daerah, dan melakukan perbaikan kata-kata singkat dan kata-kata salah ketik bahasa Indonesia serta yang dipilih merupakan ulasan yang berisi cacian maupun pujian bukan ulasan bersifat netral.

Penentuan proporsi data latih dan data uji yang digunakan adalah 90% data latih dan data uji 10%. Pelabelan dalam penelitian ini ada 2 kategori label positif untuk ulasan yang lebih banyak mengandung pujian daripada keluhan ataupun cacian sedangkan label negatif untuk ulasan yang lebih banyak mengandung keluhan ataupun cacian daripada pujian.

2.2. Pemrosesan awal

Perangkat atau pustaka yang akan dipakai dalam penelitian ini adalah pustaka Python. Mulai dari Beautifulsoup, Regex, Pandas, dan Numpy untuk pemrosesan *text*. Scikit-Learn keras untuk klasifikasi. Matplotlib dan Seaborn untuk visualisasi. Serta Imblearn untuk *resampling dataset*. Beberapa langkah penting yang dilakukan dalam pemrosesan awal terhadap *dataset* adalah:

a. *Cleansing* dilakukan untuk mengurangi *noise* dalam ulasan pengguna dengan menghilangkan *emoji*, *emoticon*(:@, :*, :D), tanda baca seperti koma (,), titik (.), angka dan juga tanda baca lainnya.
b. *Case folding* merupakan proses mengubah huruf menjadi kecil semua agar huruf menjadi seragam.
c. *Stemming* merupakan proses pengubahan kata menjadi kata dalam bentuk kata dasar.
d. *Stopword removing* merupakan penyaringan kata-kata yang mewakili dokumen tersebut sehingga kata yang dianggap tidak penting dibuang. Contoh kata-kata tidak penting adalah "di", "dan", "ke", "dari", dan lain-lain.
e. *Tokenizing* merupakan proses pemisahan setiap kata yang menyusunnya hingga membentuk *vocabulary*.



Gambar 1. Kerangka kerja penelitian.

Berikut informasi dan 5 baris teratas akhir *dataset* yang sudah bersih dalam Table 1, dengan kolom *reviews* 6954 dan label 6954.

Tabel 1. *Sample dataset review* dan label.

	reviews	label
0	bermanfaat	pos
1	juni lancar	pos
2	memperbarui aplikasi trbaru	pos
3	bintang pembaruan	pos
4	abis sederhana	pos

2.3. Eksplorasi Data Analisis

Sebelum melakukan proses pemodelan perlu dilakukan eksplorasi data analisis di mana kita akan melihat dari 2 sudut pandang yaitu analisis *textual* dengan menggunakan *library* Wordcloud untuk melihat polarisasi dan subjektivitas kata, analisis frekuensi kata yang terdiri dari beberapa persamaan 1, 2, dan 3 sebagai berikut.

$$rerata\ pos = \frac{f(pos)}{f(pos)+f(neg)} \quad (1)$$

Rerata positif ulasan adalah frekuensi ulasan positif dibagi dengan jumlah frekuensi ulasan positif dan negatif, yang akan dihitung persentasenya dengan,

$$fpct(pos) = \frac{f(pos)}{\sum f(pos)} \quad (2)$$

Sementara untuk mengurangi dampak *outlier* kita menggunakan rerata *harmonic* [16] yang didefinisikan dengan,

$$H = \frac{n}{\sum_{i=1}^{n} \frac{1}{x_i}} \quad (3)$$

Serta akan digunakan Hukum Zipf [17] untuk melihat frekuensi kata tersering yang muncul dengan rumus pada persamaan 3 berikut,

$$f(r) \propto \frac{1}{r^{\alpha}}, \text{untuk } \alpha \approx 1 \quad (4)$$

Sementara analisa numerikal untuk melihat sebaran data, model vektorisasi dan menentukan model *resampling* yang tepat [18] [19] bagi *dataset* kita.

2.4. *Baseline Model*

Setelah dilakukan eksplorasi data analisis, kita akan menentukan model dengan *dataset* yang masih belum dilakukan *resampling* membuat *baseline* model berdasarkan 7 algoritma Klasifikasi Linier, Logistic Regression, Linear SVC dan Algoritma Non Linear , Multinomial dan Complement Naïve Bayes, Decision Tree, KNN, serta SVC. Di sini akan dilakukan *pipelinening* dan mengevaluasi dengan *cross validation* untuk mendapatkan hasil akurasi tertinggi yang akan dipilih sebagai model untuk dilakukan *resampling* pada tahap selanjutnya.

2.5. *Resampling*

Berdasarkan analisa *numerical* diperoleh model *resampling* yang optimal, kita akan menerapkan model *resampling* yang terpilih sehingga di sini terjadi transformasi *dataset* yang tidak seimbang menjadi *dataset* yang seimbang untuk evaluasi ulang tetapi hanya dengan 3 model dengan akurasi tinggi.

2.6. Tuning Model

Tuning model adalah langkah untuk memberikan parameter pada model yang memiliki akurasi tertinggi berdasarkan *dataset resampling* yang sudah dilakukan pada langkah sebelumnya. Hasil evaluasi dari *tuning* model dilakukan dengan *library* GridsearchCV dari Python untuk memilih parameter terbaik.

2.7. Finalisasi Model

Tahap ini adalah melakukan finalisisasi model yaitu dengan menyimpan model yang sudah dipilih berdasarkan *tuning* dengan *dataset resampling* dan mempunyai akurasi tertinggi. Model yang disimpan dapat dipanggil kembali untuk dilakukan eksperimen terhadap data ulasan baru dan akan menghasilkan prediksi sentimen negatif atau positif.

Semua tahap akan dilakukan evaluasi dengan menggunakan *cross-validation* untuk menghasilkan pengukuran akurasi, item frekuensi, *confusion matrix* dan *classification report*, mengukur presisi, *recall* dan skor f1.

3. Hasil dan Pembahasan

Berdasarkan *dataset* yang sudah diambil, dipilih dan dilabeli maka kita dapat memisahkan 6954 data bersih dari 8080 data mentah hasil *scrapping* dengan 90% atau 6258 data *training* dan 10% atau 696 data testing, berikut hasil pemrosesan awal *review* negatif dan positif dalam Table 2.

Tabel 2. Data hasil pemrosesan awal

Review Negatif

reviews	label
kesalahan fatal aplikasi mendaftarkan kartu ap...	neg
membuka aplikasi frustasi pergi mesin atm mban...	neg
perbarui terbaru terbuka mengetuk aplikasi fla...	neg
membuka aplikasi langsung menutup otomatis bag...	neg
menutup instan bosan menghapus menginstal kali...	neg

Review Positif

reviews	label
bermanfaat	pos
juni lancar	pos
memperbarui aplikasi trbaru	pos
bintang pembaruan	pos
abis sederhana	pos

3.1. Analisa Tekstual

Dalam analisa tekstual digunakan Wordcloud untuk melihat representasi penggunaan kata dalam dokumen dengan mengatur ulang ukuran kata secara proporsional sesuai frekuensinya dan menampilkannya secara *random*. Analisa tekstual sangat penting dalam ulasan pengguna aplikasi yang menyediakan ide kata apa yang sering muncul dalam korpus serta melihat polarisasi dan subjektivitas kata dalam sentimen yang ada dan Gambar 2 menyajikan hasil analisis tekstual dengan Wordcloud sebagai berikut.

a. Analisis ulasan negatif b. Analisis ulasan positif

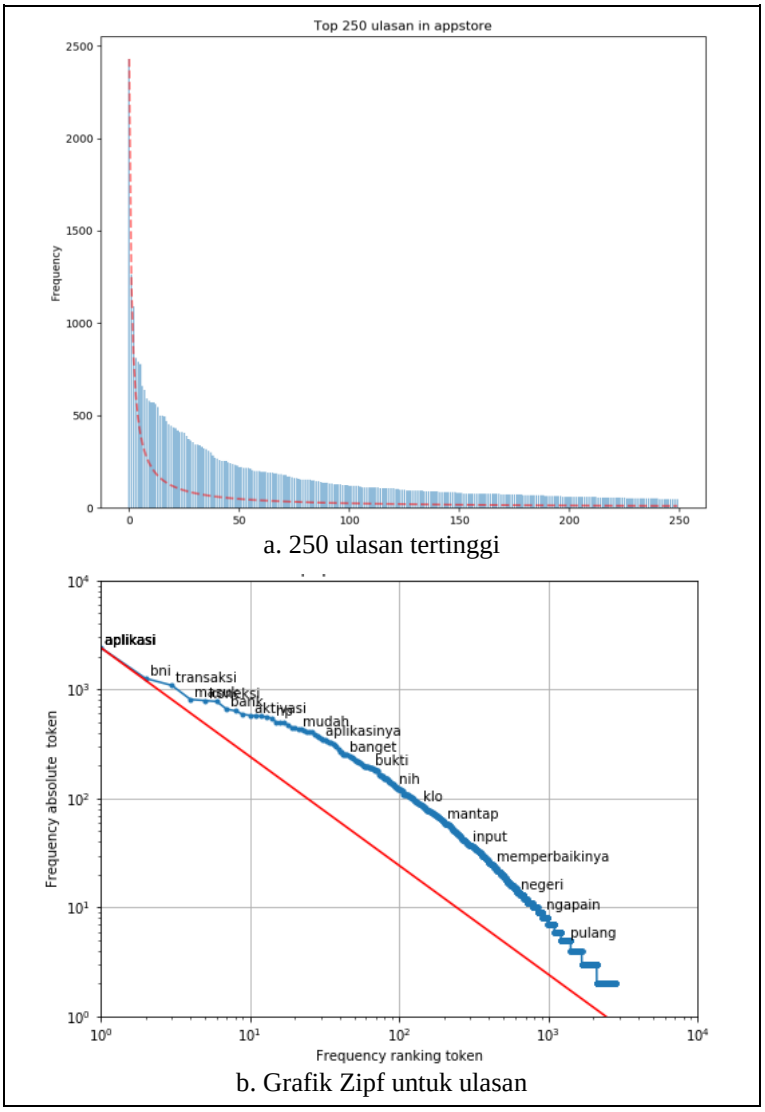
Gambar 2. Analisis ulasan

Gambar 2.a. menyajikan hasil ulasan negatif di mana frekuensi tertinggi adalah kata-kata yang tercetak besar, seperti “transaksi”, “aplikasi” yang bermakna cukup netral. Selain itu kita dapat melihat kata dalam ukuran kecil yang masuk akal untuk di kelompokan dalam ulasan negatif seperti, “buruk”, “salah”, “kecewa”, “ribet” dan sebagainya. Kita akan coba melihat kata yang positif yang muncul dalam ulasan negatif seperti kata “cinta” yang muncul 2 baris dan kita akan coba meneliti kenapa hal ini dapat terjadi, yaitu, “*meh memasang iklan keledai halaman mengerti iklan cinta perhatian menjengkelkan nyaman desain mengalahkan tujuan mobile banking perbaiki ui beruntung berfungsi normal*” dan “*cinta ui mudah saldo diperbarui otomatis transaksi memiliki tombol refresh memperbarui saldo masuk memperbarui saldo*”. Tampaknya kata “cinta” dimaknai secara hiperbolis dan kasar atau mungkin berupa kata sarkasme. Sementara dalam ulasan positif, beberapa kata yang netral muncul dalam ulasan negatif, muncul juga dalam ulasan positif, ini sebabnya kita katakan netral karena kata-kata tersebut muncul dalam kedua ulasan seperti dalam Gambar 2.b di atas. Kata-kata kecil yang masuk akal masuk dalam ulasan positif seperti “lancar”, “suka”, “oke”, “bagus”, dan sebagainya memang menunjukkan frekuensi tinggi, dan tampak tidak ada kata negatif yang muncul dalam ulasan positif ini. Lebih lanjut hasil analisa tekstual dapat dilihat dari tabel berikut yang menghitung frekuensi kata dalam korpus yang menggunakan pustaka Python CountVectorizer untuk mengekstraksi frekuensi kata. Ada beberapa opsi parameter yang tersedia untuk penghitung vektor, seperti menghapus *stopword* dan membatasi jumlah istilah maksimum. Namun, untuk mendapatkan gambaran lengkap dari *dataset* terlebih dahulu, kita menerapkannya dengan memasukkan *stopwords*, dan tidak membatasi jumlah term maksimum.

Tabel 3. Frekuensi kata negatif dan positif.

	negative	positive	total
aplikasi	1260	1170	2430
bni	599	666	1265
transaksi	524	567	1091
masuk	444	367	811
koneksi	181	610	791
tolong	314	462	776
bank	247	413	660
login	276	361	637
bagus	380	210	590
aktivasi	120	460	580

Terbukti analisa tekstual dengan Wordcloud membantu visualisasi frekuensi kata dalam korpus dan dari tabel di atas tampak urutan kata dengan frekuensi tertinggi yang muncul di kedua ulasan adalah kata “aplikasi”, sehingga wajar jika suatu ulasan yang mengandung kata tersebut dapat diprediksi sebagai ulasan negatif atau positif.

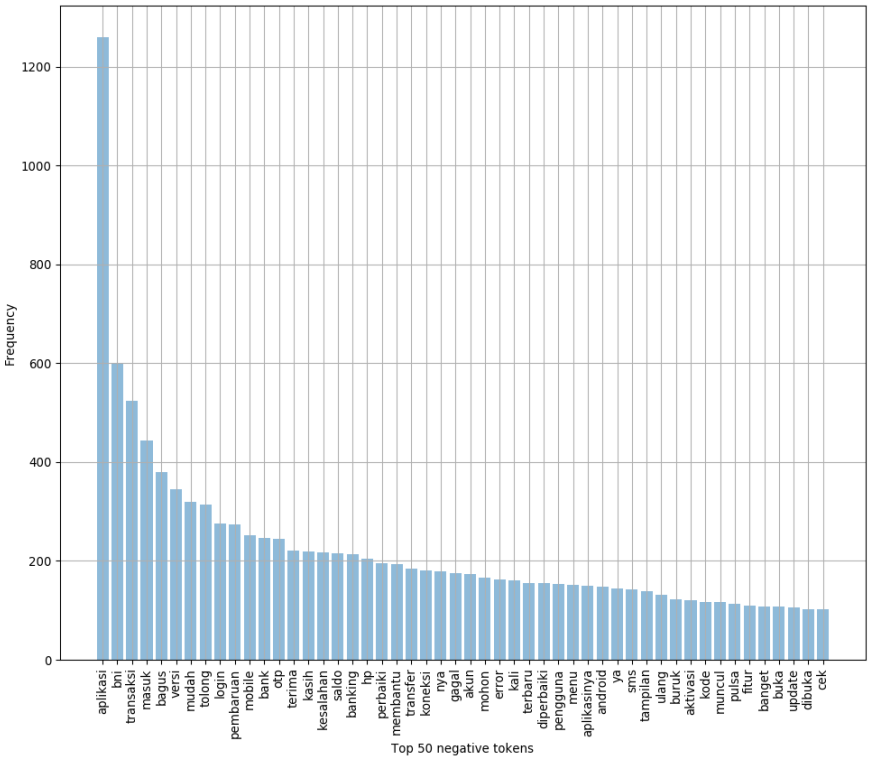


a. 250 ulasan tertinggi

b. Grafik Zipf untuk ulasan

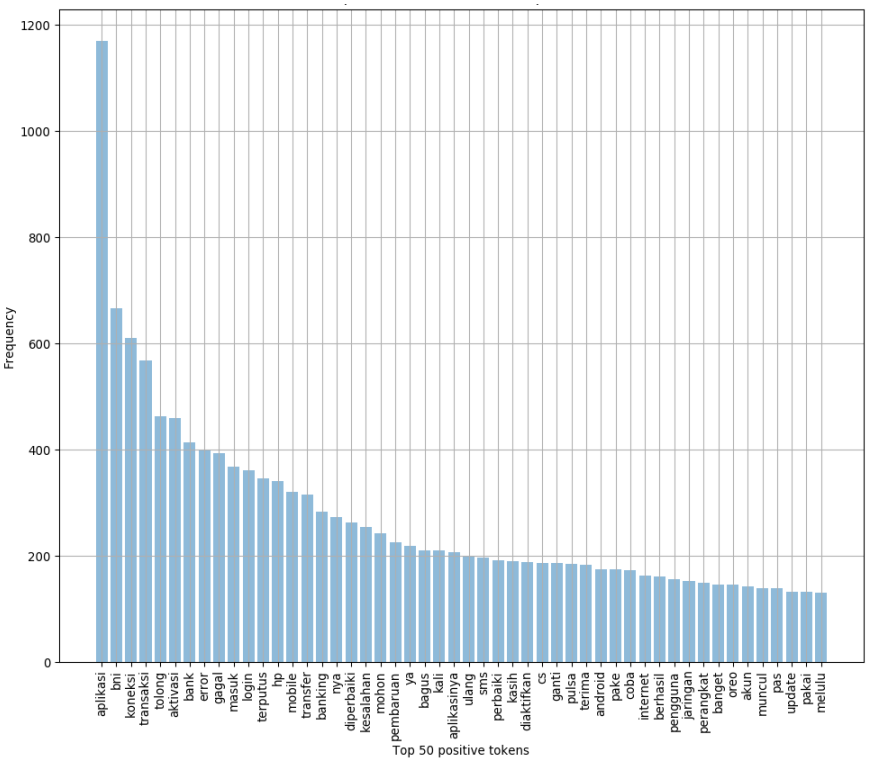
Gambar 3. 250 ulasan tertinggi dan Grafik Zipf untuk ulasan.

Selanjutnya dengan menggunakan hukum Zipf diharapkan kita lebih presisi dalam melihat kata dalam korpus untuk mengestimasi suatu ulasan masuk dalam sentimen negatif atau positif, berikut grafik 250 ulasan tertinggi dan grafik Zipf dalam Gambar 3 a dan b. Pada Gambar 3.a, sumbu *X* adalah peringkat frekuensi tertinggi dari kiri ke peringkat 250 di sebelah kanan. Sumbu *Y* adalah frekuensi yang diamati dalam korpus (dalam hal ini, *dataset* "review bni"). Satu hal yang perlu diperhatikan adalah bahwa pengamatan aktual dalam kebanyakan kasus sepenuhnya mengikuti distribusi Zipf pada Gambar 3.b, sehingga mengikuti tren distribusi "near-Zipfian". Meskipun kita dapat melihat plotnya mengikuti tren hukum Zipf, tetapi sepertinya plot tersebut memiliki lebih banyak area di atas kurva Zipf yang diharapkan dengan kata-kata yang berperingkat lebih tinggi. Cara lain untuk mem-*plot* ini adalah pada grafik *log-log*, dengan sumbu-*X* menjadi *log* (peringkat), sumbu-*Y* adalah *log* (frekuensi). Dengan mem-*plot* pada skala *log*, hasilnya akan menghasilkan garis linier kasar pada grafik. Setidaknya, kita membuktikan bahwa bahkan *review* mengikuti distribusi "near-Zipfian", tetapi ini membuat kita penasaran tentang penyimpangan dari hukum Zipf. Berikut juga disajikan 50 *review* negatif dan positif serta sebaran keduanya yang menggambarkan informasi ulasan pengguna aplikasi pada Gambar 4a, b, dan c.



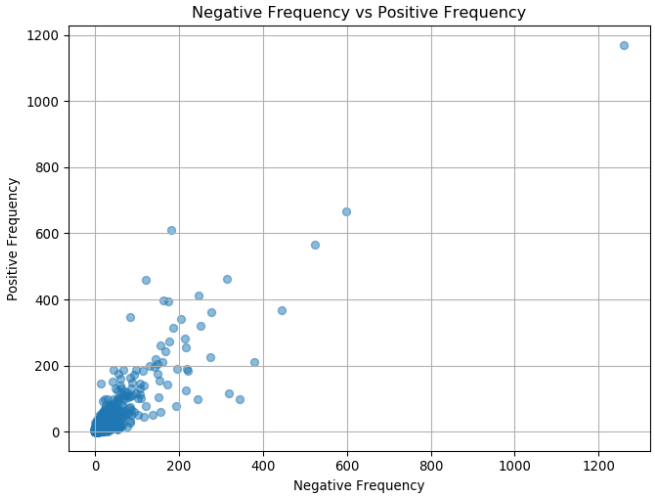
Gambar 4.a. 50 ulasan negatif.

Terlihat Gambar 5a. kata “aplikasi”, “bni” dan seterusnya menempati frekuensi tertinggi walaupun bermakna netral, sehingga polarisasi kata di sini menunjukkan bukan makna sebenarnya untuk kasus *unigram*. Sementara pada Gambar 5.b kata atau *token* yang sama menempati *ranking* tertinggi seperti pada Gambar 5.a sehingga disimpulkan untuk kasus *unigram* kata tersebut bermakna netral.



Gambar 4.b. 50 ulasan positif.

Pada Gambar 4.c sebagian besar kata di bawah 400 pada sumbu *X* dan sumbu *Y*, dan kita tidak dapat melihat hubungan yang bermakna antara frekuensi negatif dan positif



Gambar 4.c. sebaran ulasan negatif dan positif

Selanjutnya kita akan melihat rerata ulasan positif, persentase rerata dan rerata *harmonic* yang akan disajikan dalam Tabel 4 a dan b di bawah ini.

Tabel 4. Rerata dan Persentase ulasan positif

a. Rerata ulasan positif

	(-)	(+)	total	rerata
aaja	0	2	2	1
muluu	0	2	2	1
keluarnya	0	2	2	1
kemakan	0	2	2	1
kesekian	0	5	5	1
bertanggung	0	3	3	1
keselamatan	0	2	2	1
keseringan	0	3	3	1
ket	0	2	2	1
bersamaan	0	3	3	1

b. Persentase rerata ulasan positif

	(-)	(+)	total	rate	Freq(%)
aplikasi	1260	1170	2430	0.481481	0.034
bni	599	666	1265	0.526482	0.019
koneksi	181	610	791	0.771176	0.018
transaksi	524	567	1091	0.519707	0.016
tolong	314	462	776	0.595361	0.013
aktivasi	120	460	580	0.793103	0.013
bank	247	413	660	0.625758	0.012
error	162	398	560	0.710714	0.012
gagal	175	394	569	0.692443	0.011
masuk	444	367	811	0.452528	0.011

Pada Tabel 4.a kita melihat frekuensi masing-masing ulasan. Secara intuitif, jika sebuah kata muncul lebih sering di satu kelas dibandingkan dengan yang lain, ini bisa menjadi ukuran yang baik dari seberapa banyak kata tersebut bermakna untuk menggambarkan kelas tersebut. Sementara sajian pada Tabel 4.b, kata-kata dengan pos_rate tertinggi memiliki frekuensi nol di *review* negatif, tetapi frekuensi keseluruhan dari kata-kata ini terlalu rendah untuk menganggapnya sebagai pedoman untuk *review* positif. Metrik lain adalah frekuensi kata muncul di kelas tersebut.

Tetapi karena pos_freq_pct hanyalah frekuensi yang diskalakan dari jumlah total frekuensi, pangkat pos_freq_pct persis sama dengan frekuensi positif. Apa yang dapat kita lakukan sekarang adalah menggabungkan pos_rate, pos_freq_pct bersama-sama untuk menghasilkan metrik yang mencerminkan pos_rate dan pos_freq_pct. Meskipun kedua hal ini dapat mengambil nilai mulai dari 0 hingga 1, pos_rate memiliki rentang yang jauh lebih luas sebenarnya berkisar dari 0 hingga 1, sementara semua nilai pos_freq_pct terjepit dalam rentang yang lebih kecil dari 0,07. Jika kita meratakan dua angka ini, pos_rate akan terlalu dominan, dan tidak akan mencerminkan kedua metrik secara efektif. Kedua rerata diatas masih mempunyai kelemahan dalam menghasilkan kesimpulan, maka solusi untuk melihat secara adil adalah dengan menggunakan rerata *harmonic* seperti pada Tabel 5 berikut.

Tabel 5. Rerata *harmonic*.

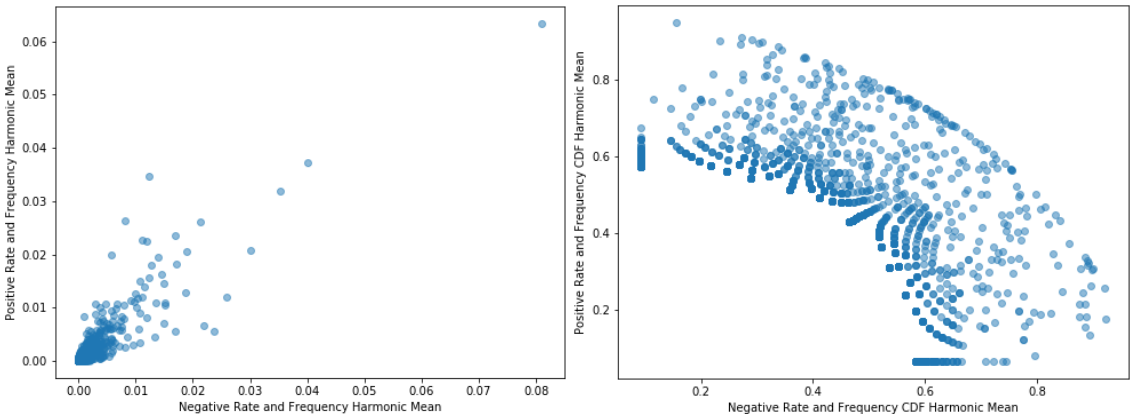
	negative	positive	total	pos_rate	pos_freq_pct	pos_hmean
aplikasi	1260	1170	2430	0.481481	0.034	0.063399
bni	599	666	1265	0.526482	0.019	0.037265
koneksi	181	610	791	0.771176	0.018	0.03459
transaksi	524	567	1091	0.519707	0.016	0.031881
aktivasi	120	460	580	0.793103	0.013	0.026241
tolong	314	462	776	0.595361	0.013	0.026209
bank	247	413	660	0.625758	0.012	0.023507
error	162	398	560	0.710714	0.012	0.022718
gagal	175	394	569	0.692443	0.011	0.022483
masuk	444	367	811	0.452528	0.011	0.020799

Peringkat rata-rata harmonis sepertinya sama dengan pos_freq_pct. Dengan menghitung rata-rata harmonik, dampak nilai kecil (dalam hal ini, pos_freq_pct) terlalu diperparah dan akhirnya mendominasi nilai rata-rata. Sekali lagi ini sama persis dengan hanya peringkat nilai frekuensi dan tidak memberikan hasil yang jauh lebih berarti. Selanjutnya kita akan menghitung nilai CDF (Cumulative Distribution Function) dari pos_rate dan pos_freq_pct. CDF dapat dijelaskan sebagai "fungsi distribusi X, dievaluasi pada x, adalah probabilitas bahwa X akan mengambil nilai kurang dari atau sama dengan x". Dengan menghitung nilai CDF seperti pada table 7, kita dapat melihat di mana nilai pos_rate atau pos_freq_pct terletak pada distribusi dalam hal kumulatif. Pada hasil kode di bawah ini, kita dapat melihat kata "oreo" dengan pos_rate_normcdf dari 0. 908881, dan pos_freq_pct_normcdf dari 0.998243Ini berarti sekitar 91% dari review akan mengambil nilai pos_rate kurang dari atau sama dengan 0.908881, dan 99,82% akan mengambil nilai pos_freq_pct kurang dari atau sama dengan 0.004.

Tabel 6. Distribusi frekuensi kumulatif ulasan positif.

	-	+	∑	rate	pct	hmean	normcdf	normcdf	normcdf hmean
oreo	13	145	158	0.917722	0.004	0.008373	0.908881	0.998243	0.951469
diaktifkan	43	188	231	0.813853	0.005	0.010833	0.835767	0.999944	0.910514
terputus	83	346	429	0.806527	0.01	0.019824	0.829465	1	0.906784
aktifasi	18	91	109	0.834862	0.003	0.005262	0.852993	0.958256	0.902566
aktivasi	120	460	580	0.793103	0.013	0.026241	0.81752	1	0.8996
jaringan	41	152	193	0.787565	0.004	0.008768	0.812443	0.99894	0.896091
email	24	99	123	0.804878	0.003	0.005722	0.828025	0.971727	0.894139
koneksi	181	610	791	0.771176	0.018	0.03459	0.796916	1	0.886982
pake	55	174	229	0.759825	0.005	0.010027	0.785725	0.999812	0.879934
nih	31	98	129	0.75969	0.003	0.005663	0.78559	0.970272	0.868218

Jika dilakukan perhitungan pada ulasan negatif, maka kita dapat memperoleh hasil *plotting* dalam Gambar 6 a dan b berikut di bawah ini yang terdiri grafik *harmonic* rerata negatif dan positif serta grafik CDF *harmonic*.



a. Frekuensi harmonik b. CDF Harmonik

Gambar 5. Grafik frekuensi dan CDF Harmonik ulasan negatif dan positif.

Sepertinya rata-rata harmonis dari tingkat CDF dan frekuensi CDF telah menciptakan pola yang menarik pada plot. Jika titik data dekat dengan sudut kiri atas, itu lebih positif, dan jika lebih dekat ke sudut kanan bawah, itu lebih negatif. Adalah baik bahwa metrik telah menciptakan beberapa wawasan yang berarti dari frekuensi, tetapi dengan data teks, menunjukkan setiap *token* hanya sebagai titik kekurangan informasi penting yang diwakili *token* setiap titik data. Dengan 8.000 poin, sulit untuk membubuhi keterangan semua poin dalam plot. Untuk bagian ini, kita telah mencoba beberapa metode dan sampai pada kesimpulan bahwa itu tidak terlalu praktis atau layak untuk secara langsung menjelaskan poin data pada plot.

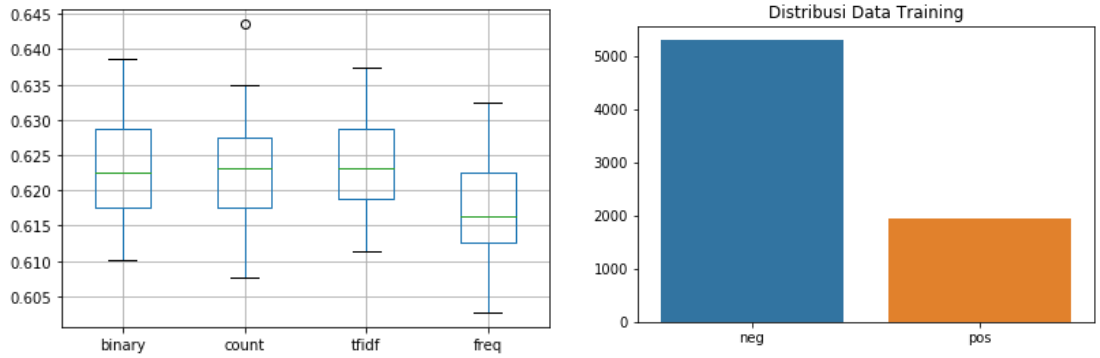
3.2. Analisa Numerikal

Dalam bagian ini kita cenderung melakukan analisa terhadap ekstraksi fitur dan *vector*, dengan mencari mode *tokenizer* terbaik (*bag of words*) untuk dilakukan *resampling* pada *dataset*. Implementasinya menggunakan pustaka keras Sequential untuk mendapatkan mode terbaik vektorisasi dengan melakukan 30000 kali perulangan dengan 50 neuron dan *epoch* sebesar 50 saat *training* model dan berikut hasilnya pada Tabel 7 di bawah ini.

Tabel 7. Deskripsi Statistik mode vektorisasi terbaik.

	binary	**count**	**tfidf**	**freq**
count	30000	30000	30000	30000
mean	0.548851	0.55067	0.548	0.516
std	0.010484	0.00674	0.009	0.005
min	0.528736	0.54023	0.533	0.507
25%	0.543103	0.54598	0.543	0.511
50%	0.547414	0.54813	0.55	0.516
75%	0.556753	0.5546	0.553	0.518
max	0.573276	0.56753	0.566	0.527

Terlihat mode *count* tampak tinggi dibanding lainnya, namun untuk lebih jelasnya kita akan plot kotak untuk melihat sebaran serta *outlier* jika ada seperti disajikan pada Gambar 7.a berikut di bawah. Berdasarkan gambar di atas maka dipilih mode *binary* dengan sebaran merata dan tidak terdapat *outlier*. Setelah dilakukan pemisahan pada data *training* kita akan melakukan analisa ekstraksi fitur.



a. Perbandingan mode vektorisasi. b. Sebaran data *training* kelas ulasan.

Gambar 6. Perbandingan mode vektorisasi dan sebaran data *training* kelas

Pertama kita lihat sebaran datanya seperti pada Gambar 7.b di mana data tampak tidak seimbang dengan perbandingan data negatif dan positif sebesar 84%:16% atau 5818:1136 atau 5:1 seperti pada Gambar 7.b di atas. Untuk meyakinkan dilakukan evaluasi dengan melihat akurasi dengan model Multinomial Naïve Bayes pada data testing didapat akurasi sebesar 88.5%, sementara untuk fitur tunggal akurasi turun menjadi 82.75%, dan ini merupakan jebakan pengukuran tanpa *cross validation*, karena saat dilihat jumlah item frekuensi kelasnya dan *confusion* matriksnya pada saat 1 fitur hampir seluruh prediksinya berada dalam kelas negatif diperoleh hasil seperti Tabel 8 di bawah ini.

Tabel 8. Frekuensi kelas dan confusion matrik..

Keterangan	Data testing semua fitur	Data testing 1 fitur
Akurasi	88.50%	82.75%
Frekuensi kelas	[[0 574] [1 52]]	[[0 626]]
Confusion matrix	[[510 8] [64 44]]	[[518 0] [108 0]]

Terlihat selisih kelas frekuensi terlalu lebar sebesar, juga distribusi prediksi dan kenyataan yang menunjukkan angka tidak berimbang, demikian juga *error rate* sebesar % meyakinkan perlunya melakukan *resampling dataset*. Model *resampling* diperoleh dengan melakukan perbandingan metode Random Over Sampling, Random Under Sampling, dan kombinasi keduanya dan hasilnya seperti Tabel 9 berikut ini.

Tabel 9. Hasil akurasi model *resampling*.

Model	Akurasi	Selisih kelas
Ros	93.40%	
Smote	93.25%	
Adasysn	89.68%	
Smote-nc	**96.93%**	[[0 4634][1 4802]]= |168|
Bl-smote	87.99%	
Rus	93.71%	
Nearmiss 1, 2, 3	88.29%, 94.04%, 78.64%	
Tomek-link	92.68%	
Editednearestneighbours	95.92%	
Repeatededitednearestneighbours	**98.22%**	[[0 902][1 896]]= |6|
Allknn	**96.30%**	[[0 1009][1 857]]= |157|
Onesidedselection	92.68%	
Neighbourhoodcleaningrule	94.35%	
Smoteenn	**98.89%**	[[0 791][1 4631]]= |3840|
Smotetomek	93.58%	

Model *resampling* yang dipilih adalah Repeated Edited Nearest Neighbours karena memiliki akurasi tinggi 98.22% dengan selisih frekuensi kelas terendah 6.

3.3. *Baseline Model*

Untuk menentukan model klasifikasi, dalam penelitian ini akan dipilih 7 algoritma, Logistic Regression dan Linear SVC yang mewakili model linear, Multinomial dan Complement Naïve Bayes, Decision Tree, K-Nearest Neighbor dan SVC yang mewakili model non-linear akan diterapkan menggunakan pustaka Python Scikit-Learn. Tiap model dipisahkan dengan *cross validation* sebesar 10 dan diukur menggunakan *metric* akurasi dan hasilnya seperti pada Tabel 10.a.

3.4. Aplikasi *Resampling* untuk Model *Baseline*

Setelah dilakukan *resampling* berdasarkan pada Tabel 10, yaitu model Repeated Edited Nearest Neighbours (RENN) terpilih untuk model *baseline* dan berikut hasilnya pada Tabel 10.b di bawah ini. Di mana SVC menduduki peringkat tertinggi diikuti oleh Linear SVC dan *logistic regression*.

Tabel 10. *Baseline model* dengan *resampling*.

a. Akurasi sebelum *resampling*

Model	Mean	STD
Logistic Regression	**0.913912**	0.010253
Linear SVC	**0.93288**	0.012505
Multinomial NB	0.899217	0.01305
Complement NB	0.904325	0.015028
Decision Tree	0.901557	0.012944
K-Nearest Neighbor	0.686124	0.030821
SVC	**0.928618**	0.010271

b. Akurasi setelah *resampling*

Model	Mean	STD
Logistic Regression	**0.964607**	0.027683
Linear SVC	**0.977783**	0.016379
Multinomial NB	0.944454	0.040858
Complement NB	0.955565	0.034340
Decision Tree	0.945187	0.039688
K-Nearest Neighbor	0.913980	0.087088
SVC	**0.980570**	0.011535

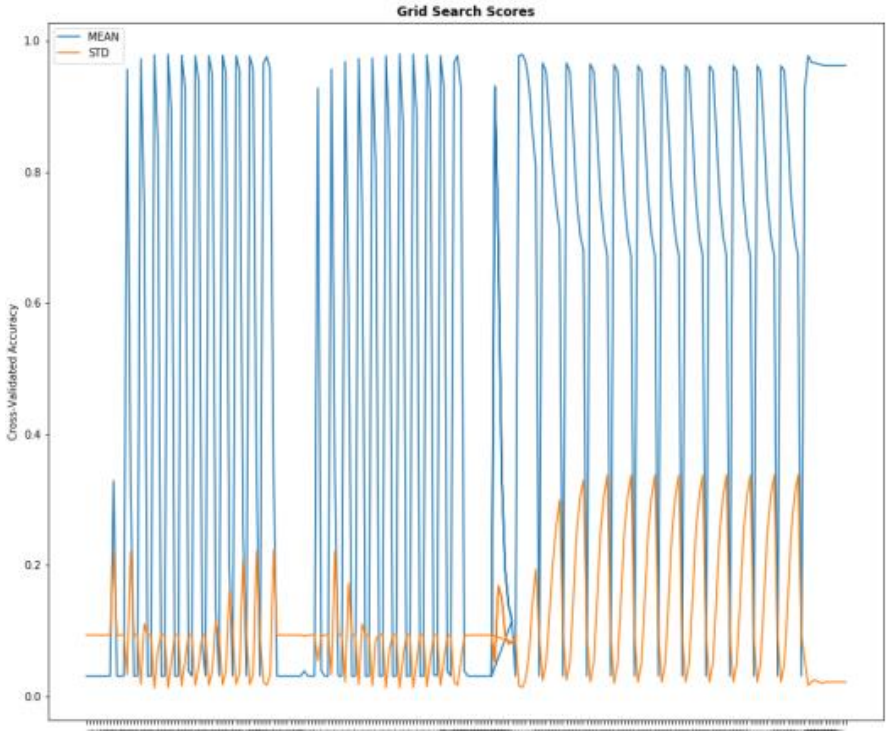
3.5. Tuning Model

Setelah didapatkan model setelah *resampling* kemudian dilakukan *tuning* parameter model terpilih dengan pustaka Python GridSearchCV sesuai dengan Tabel 11 di bawah ini.

Tabel 11. Parameter untuk *tuning*.

kernel	rbf	sigmoid	poly	linear
gamma	[1e-2, 1e-3, 1e-4, 1e-5]			
C	[0.001, 0.10, 0.1,1, 10, 20,30,40, 50,60,70,80,90,100, 1000]			
degree	-	-	[0, 1, 2, 3, 4, 5, 6]	-

Didapatkan hasil akurasi terbaik sebesar: 0.98543 dengan parameter: {'C': 1, *'degree'*: 2, *'kernel'*: *'poly'*}. Berikut Gambar 7 kurva *elbow* hasil dari *gridsearch*.



Gambar 7. Kurva *elbow tuning* parameter model.

3.6. Finalisasi Model dan Eksperimen Data Baru

Hasil finalisasi model adalah menyimpan model untuk dapat dipakai kembali untuk prediksi terhadap data baru dan berikut hasil eksperimen data baru pada Gambar 8.

reviews	label	predicted	result	reviews	label	predicted	result
masih ganggu mau masuk enggak bisa akun aktip perangkat lain padahal kartu sim internet pakai no daftar aduh tolong baik segera terimakasih	0	0	True	coba moga bagus	1	1	True
kenapa mau masuk susah ya padahal pulsa ada kartu	0	0	True	server kelas payah terlalu eror kampret	0	0	True
pernah daftar bank ganti hp unduh aplikasi mau masuk kenapa harus registrasi	0	0	True	koneksi putus padahal pakai jaring stabil pakai aplikasi lain lancar apa	0	0	True
depan tolong tambah bayar ukt universitas terimakasih	0	1	False	tolong baik	1	0	False
apakah sedang sulit hari	0	0	True	bantu	1	1	True
kenapa tidak bisa masuk koneksi putus padahal koneksi sedang ada baik	0	0	True	eror payah ah susah mau transfer	0	0	True
terlalu lola	0	1	False	kenapa enggak bisa transfer ya padahal sinyal bagus kuota masih banyak	0	1	False
aplikasi enggak bisa padahal sinyal stabil kalo gin males pakai masa transfer musti atm enggak praktis	0	0	True	kenapa mau transfer jaring putus sampai hapus aktivasi susah minta ampun sampai	0	0	True
mesti ada notifikasi sistem kalau ada ganggu transaksi jangan bilang jaring putus padahal jaring baik obrol	0	1	False	tidak dukung hp	0	1	False
eror kecewa darurat tolong direspon	0	0	True	mau aktivasi repot harus ganti pasang sim koneksi harus stabil kasihan kurang erti teknologi malah makin sulit padahal dulu	0	0	True
ganggu muter tapi lumayan mudah	0	1	False	jaring tidak stabil jaring tidak stabil kali oke hari oke bulan tiap hari coba jaring tidak stabil aduh aduh mana atm eror sungguh tidak bisa bangga kalo gaji enggak lewat	0	0	True
jaring lancar masuk lancar tapi transfer enggak bisa	0	0	True	bantu kerja bagus	1	1	True
moga kembang	1	1	True	bagus	1	1	True

Gambar 8. Eksperimen data baru

Untuk data sebanyak 26 ulasan dengan label model dapat melakukan prediksi benar sebesar 19 ulasan dengan 7 ulasan yang salah, artinya model untuk data aktual mempunyai *error rate* sebesar 27% dan ini tidak buruk.

4. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan dapat disimpulkan beberapa hal sebagai berikut:

a. Dari 6258 data *train* dan 696 data *test* hasil analisis tekstual menunjukkan kecenderungan polaritas netral pada kata dengan frekuensi tinggi yaitu "aplikasi", sebesar total 2430, dengan 1260 *review* negatif dan 1170 positif . Sementara berdasarkan hukum Zipf semua kata berada di atas garis linear yang diharapkan. Terakhir berdasarkan rerata frekuensi, *harmonic*, dan CDF (Cumulative Distribution Frequency) didapatkan, kata "oreo" mempunyai rerata sebesar 0.917722, dan persentase rerata 0.908881. Ini berarti sekitar 91,77% dari *review* akan mengambil nilai rerata kurang dari atau sama dengan 0.908881, dan rerata *harmonic* sebesar 95,14% persen kurang dari rerata persentase normal CDF sebesar 99.82 yang artinya 4.68 lebih besar frekuensinya katanya dibanding rerata *harmonic*.
b. Dari analisa numerik pada data *training* dengan kelas negatif 5236 dan positif 1022, diperoleh model vektorisasi terbaik adalah *binary* dengan akurasi sebesar 54.88% dan model *resampling* terbaik Repeated Edited Nearest Neighbours (RENN) dengan akurasi sebesar 98.22%.
c. Berdasarkan 7 *baseline model* didapatkan 3 model dengan akurasi tertinggi regresi *logistic* 91%, SVC 92%, dan Linear SVC 93%, dan setelah dilakukan *resampling* RENN didapatkan model terbaik yaitu SVC dengan akurasi sebesar 98.05%, artinya terdapat kenaikan sebesar 6%.
d. Tuning parameter model terpilih menghasilkan akurasi baru sebesar 98.54% dengan parameter : {'C': 1, *'degree'*: 2, *'kernel'*: *'poly'*}, artinya terdapat kenaikan sebesar 0.5%.
e. Hasil eksperimen model terhadap data baru sebanyak 26 *review* menghasilkan prediksi benar 19 dan salah sebesar 7 atau menghasilkan *error rate* 27%.

Daftar Pustaka

[1] Adhi, "Mobile Banking, Kemudahan Transaksi Finansial di Ujung Jari Anda," *https://www.money.id/*, 2016. [Online]. Available: https://www.money.id/finance/mobile-banking-kemudahan-transaksi-finansial-di-ujung-jari-anda-1605176/pengguna-mobile-banking-tumbuh-pesat.html. [Accessed: 17-Apr-2019].

[2] P. B. Tbk, "Memperkokoh keunggulan kompetitif," 2017.

[3] G. Yudistira, "Transaksi mobile banking BNI tembus Rp 90 triliun hingga akhir September," *KONTAN.CO.ID*, 2018. [Online]. Available: https://keuangan.kontan.co.id/news/transaksi-mobile-banking-bni-tembus-rp-90-triliun-hingga-akhir-september.

[4] Sindo, "Begini Cara Menyusun Strategi Persaingan Usaha," *okezone.com*, 2017. [Online]. Available: https://economy.okezone.com/read/2017/02/10/320/1614693/begini-cara-menyusun-strategi-persaingan-usaha.

[5] topbrand-award, "Index Top Brand Award," 2018. [Online]. Available: http://www.topbrand-award.com/top-brand-survey/survey-result.

[6] F. Gunawan, M. A. Fauzi, and P. P. Adikara, "Analisis Sentimen Pada Ulasan Aplikasi Mobile Menggunakan Naive Bayes dan Normalisasi Kata Berbasis Levenshtein Distance (Studi Kasus Aplikasi BCA Mobile)," *Syst. Inf. Syst. Informatics J.*, vol. 3, no. 2, pp. 1–6, 2017, doi: 10.29080/systemic.v3i2.234.

[7] Statcounter, "Mobile Operating System Market Share Indonesia," *https://gs.statcounter.com/*, 2018. [Online]. Available: http://gs.statcounter.com/os-market-share/mobile/indonesia.

[8] G. A. Buntoro, "Analisis Sentimen Calon Gubernur DKI Jakarta 2017 Di Twitter," *Integer J. Maret*, vol. 1, no. 1, pp. 32–41, 2017.

[9] M. W. Ningrum and W. Wijanarto, "Implicit Social Trust Dan Support Vector Regression Untuk Sistem Rekomendasi Berita," *CogITo Smart J.*, vol. 3, no. 2, p. 275, 2018, doi: 10.31154/cogito.v3i2.77.275-285.

[10] L. Zhang, K. Hua, H. Wang, G. Qian, and L. Zheng, "Sentiment analysis on reviews of mobile users," *Procedia Comput. Sci.*, vol. 34, pp. 458–465, 2014, doi: 10.1016/j.procs.2014.07.013.

[11] M. Rezwanul, A. Ali, and A. Rahman, "Sentiment Analysis on Twitter Data using KNN and SVM," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 6, pp. 19–25, 2017, doi: 10.14569/ijacsa.2017.080603.

[12] S. A. Aljuhani and N. S. Alghamdi, "A comparison of sentiment analysis methods on Amazon reviews

of Mobile Phones," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 6, pp. 608–617, 2019, doi: 10.14569/ijacsa.2019.0100678.
[13] B. Olabenjo, "Applying Naive Bayes Classification to Google Play Apps Categorization," 2016.
[14] A. Mueez, K. A. T. Islam, and W. Iqbal, "Exploratory Data Analysis and Success Prediction of Google Play Store Apps Authors," no. December, 2018.
[15] R. S. Saleh, "How Should You plan Your App's Features ? Selecting and Prioritizing A Mobile App's Initial Features Based on User Reviews," 2017.
[16] D.-F. Xia, S.-L. Xu, and F. Qi, "A proof of the arithmetic mean-geometric mean-harmonic mean inequalities," *RGMIA Res. Rep. Collect.*, vol. 2, no. 1, pp. 85–87, 1999.
[17] D. M. W. Powers, "l," pp. 151–160, 1998.
[18] A. More, "Survey of resampling techniques for improving classification performance in unbalanced datasets," vol. 10000, pp. 1–7, 2016.
[19] J. Heathcote, "An Experimental Survey of Simple k -Nearest Neighbour Condensing and Editing Algorithms," vol. 28, 2013.