

Pengaruh Jenis Stemmer Terhadap Algoritma Svm Pada Analisis Sentimen Berbasis Lexicon Dengan Afinn Lexicon Resource

Luthfi Nurul Huda¹, Andi Sunyoto², Kusnawi³

^{1,2,3} Magister Teknik Informatika Universitas Amikom,
Yogyakarta, Indonesia

Article Info

Article history:

Diterima 13 Februari 2024

Revisi 20 Februari 2024

Diterbitkan 5 April, 2024

Keywords:

Analisis sentimen

Lexicon based

Preprocessing

Stemming

SVM

ABSTRAK

Analisis sentimen merupakan bidang ilmu yang memiliki potensi besar dalam penelitian dan aplikasi praktis. Ini merupakan sebuah tugas dari NLP yang dieksploitasi untuk mengekstraksi dan mengklasifikasi konten berdasarkan sentimen emosi baik positive, negative dan netral. Analisis sentimen sendiri dibagi menjadi tiga teknik: teknik berbasis leksikon (lexicon-based), teknik berbasis machine learning (machine learning-based), dan teknik hybrid-based. Penelitian ini mengangkat teknik hybrid-based. Penelitian ini befokus untuk menemukan jenis stemmer yang dapat meningkatkan performa dari algoritma SVM pada analisis sentimen berbasis lexicon. Penelitian ini menerapkan tiga jenis stemmer yang berbeda yakni porter stemmer, snowball stemmer, dan Lancaster stemmer. Kemudian menggunakan AFINN lexicon dictionary. Terakhir algoritma SVM akan dievaluasi menggunakan confusion matrix. Penelitian ini melakukan tiga skenario, yakni gabungan antara jenis stemmer yang digunakan dengan algoritma SVM. Dari ketiga skenario yang dilakukan, gabungan SVM dan Snowball stemmer mendapatkan nilai Accuracy, Precision, Recall dan F1-Score paling tinggi dari dua skenario lainnya. Yakni dengan nilai Accuracy sebesar **95,67 %**, Precision sebesar **95,68 %**, Recall sebesar **95,67 %** dan F1-Score sebesar **95,67 %**.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Andi Sunyoto,

Magister Teknik Informatika Universitas Amikom, Jl. Ring Road Utara, Ngringin, Condongcatur, Kec.

Depok, Kabupaten Sleman, Daerah Istimewa Yogyakarta 55281, Indonesia

Email: andi@amikom.ac.id

1. PENDAHULUAN

Analisis sentimen merupakan sebuah bidang ilmu yang memiliki potensi besar dalam penelitian dan aplikasi praktis. Membantu menjelaskan bahwa penelitian analisis sentimen khususnya teks semakin menarik perhatian baik di dalam maupun di luar negeri [1]. Ini merupakan tugas penting untuk mendeteksi polaritas sentimen dalam bentuk teks, yang banyak diterapkan dalam sistem *e-commerce*, blog, dan sosial media [2]. Secara umum analisis sentimen merupakan proses atau cara untuk mengevaluasi sejauh mana bahasa tertulis atau secara lisan ditentukan sebagai ekspresi positif, negatif, maupun netral [3]. Ini merupakan teknik yang sangat berguna sebagai analisa data teks dengan jumlah besar. Secara tradisional, ini berbeda dengan informasi peringkat (*ranked information*), *natural text* (teks alami) yang tidak dapat digunakan dengan tepat pada proses analisis [4].

Analisis sentimen sendiri merupakan tugas dari NLP yang dieksploitasi untuk mengekstraksi dan mengklasifikasi konten tekstual UGC berdasarkan sentimen seperti emosi (kemarahan, kesedihan, kebahagiaan, frustrasi), presisi polaritas (positif, negatif, netral), berbasis aspek (ulasan produk), dan konten multibahasa (membutuhkan algoritma untuk mengekstrak konten teks) [5]. Analisis sentimen dapat dibagi menjadi tiga teknik: teknik berbasis leksikon (*lexicon-based*), teknik berbasis machine learning (*machine learning-based*), dan teknik *hybrid-based* [6].

Lexicon-based menggunakan kamus kata yang telah diberi label polaritas (positif, negatif, atau netral). Yang kemudian polaritas dari dokumen (teks) ditentukan dengan menghitung jumlah kata positif, negatif dan netral dalam teks tersebut. Di sisi lain, membutuhkan input manual dari leksikon sentimen dan bekerja dengan baik di domain manapun. Pendekatan berbasis leksikon dapat dilakukan menggunakan *SentiStrength*, *SentiWordNet*, *linguistic inquiry word count (LIWC)* dan *affective norms for english words (ANEW)*[7]. *Machine learning-based* menggunakan machine learning untuk mempelajari hubungan anatara kata dan polaritasnya. Model ini dilatih pada dataset teks berlabel, di mana setiap teks mempunyai label polaritas. Model ini juga bergantung pada ukuran jumlah dataset karena menghitung polaritas sentimen melalui teknik statistik. *Naïve bayes (NB)*, *Multi-layer Perceptron (MLP)*, *Multinomial Naïve Bayes (MNB)*, *Random Forest (RF)*, *Maximum Entropy*, *Support Vector Machine (SVM)* merupakan contoh dari pendekatan berbasis *machine learning* [7]. *Hybrid-based* merupakan gabungan dari *Lexicon-based* dan *Machine Learning-based*. Kedua pendekatan tersebut memiliki kekurangan dan pendekatan *hybrid* akan membantu keterbatasan keduanya, sehingga meningkatkan kinerja, efisiensi, dan skalabilitas [7].

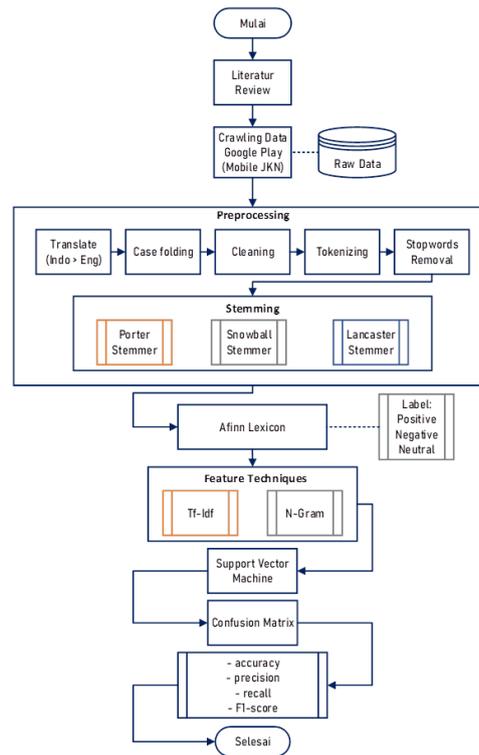
Beberapa tahapan menjadi tantangan dalam analisis sentimen. Baik pada *preprocessing* data, pemilihan leksikon, *feature extraction*, algoritma yang digunakan dan lain sebagainya. Beberapa penelitian telah dilakukan untuk menjawab tantangan-tantangan tersebut. Seperti yang dilakukan [8] berfokus pada *preprocessing* khususnya *stemming* menghasilkan akurasi tinggi pada analisis sentimen. Dengan menggunakan algoritma Naïve Bayes mereka dan *preprocessing* yang kompleks mereka mendapatkan akurasi sebesar 70.88%. Berikutnya [2] menentukan kombinasi antara *preprocessing* khususnya pada *stemming* dan algoritma pada analisis sentimen. Menunjukkan bahwa *preprocessing* yang optimal bisa menghasilkan akurasi sebesar 96% pada algoritma SVM. Terakhir oleh [9] melakukan eksperimen dengan berbagai algoritma *machine learning* untuk mencari dampak dari sejumlah fitur pada keseluruhan efisiensi model dianalisis untuk berbagai model klasifikasi. Menyimpulkan bahwa akurasi ditingkatkan dengan menggunakan *hybrid* dari *classifier* leksikon dengan model klasifikasi lainnya.

Pada penelitian ini, kami memilih untuk menggunakan pendekatan berbasis *hybrid*. Dengan berpandangan dari cara kerja pendekatan ini yakni menggunakan *lexicon* untuk mengidentifikasi kata-kata yang memiliki kecenderungan tertentu dan kemudian menggunakan pendekatan *machine learning* untuk mempelajari hubungan antara kata-kata dengan teks sentimen. Dan mempertimbangkan beberapa karya-karya sebelumnya, kami akan berfokus untuk mengeksplorasi beberapa jenis *stemmer*. Maka penelitian ini bertujuan untuk menemukan jenis *stemmer* yang dapat meningkatkan performa dari algoritma SVM pada analisis sentimen berbasis *lexicon*.

Penelitian ini disusun sebagai berikut, pada bagian II akan menjelaskan metode yang menyajikan formula atau rumusan yang digunakan untuk analisis sentimen. Bagian III memaparkan hasil dan diskusi dari metode yang digunakan. Dan bagian IV akan disimpulkan hasil dari penelitian yang sudah dilakukan.

2. METODE

Ada beberapa langkah yang dilakukan pada penelitian ini, yakni pertama pengumpulan data, *preprocessing* data dengan menggunakan tiga jenis *stemmer* yang berbeda yakni Porter Stemmer, Snowball Stemmer dan Lancaster Stemmer, ketiga klasifikasi dengan *lexicon* menggunakan *AFINN lexicon resource*, selanjutnya, implementasi algoritma *Support Vector Machine (SVM)* dan terakhir adalah *model evaluation* menggunakan *Confusin Matrix*. Langkah-langkah tersebut dipaparkan pada Gambar 1.



Gambar 1. Langkah-langkah penelitian

2.1. Pengumpulan Data

Penelitian ini menggunakan data yang ditambah dari internet, khususnya dari Google Play Store. Kami menggunakan library *google-play-scrapers* pada aplikasi “mobile JKN” dengan url `app.bpjs.mobile`. Data yang diperoleh sebanyak 20000 data. Yang kemudian diproses untuk diambil variabel penting dari data yakni variabel “reviewId”, “content”, dan “score”. Contoh dataset bisa dilihat pada Tabel 1.

Tabel 1. Contoh Dataset

reviewId	content	Score
c853ad87-fbca-4c61-99d2-0aa7120f5ccd	Mudah dimengerti sukses selalu mobile jkn	5
16e46f09-e8fa-4c91-b450-5b2cf8efa9ce	Sudah lama mendaftar ujung ujungnya tidak jadi, Klu buat apk sdh siap kapanpun	1
775b70be-fb59-4576-8793-6653e7a86605	Tolong permudah dalam penambahan anggota	4

2.2. Preprocessing

Setelah melewati beberapa tahapan, data yang sudah diperoleh kemudian akan melalui proses *preprocessing*. *Preprocessing* merupakan tahapan awal dalam beberapa tugas pemrosesan teks, termasuk analisis sentimen. *Preprocessing* adalah langkah yang penting dalam analisis sentimen dikarenakan dapat mempengaruhi akurasi hasil secara signifikan [8]. Proses ini dilakukan dengan menggali, mengelola dan menerjemahkan informasi serta menggali hubungan data terstruktur dan tidak menguji data dengan menghilangkan noise dan menyamakan kata serta menambah volume data. Tahapan *preprocessing* meliputi:

- *Case folding* berfungsi untuk mengganti semua *letters* (huruf) yang ada pada kalimat menjadi huruf kecil [8]
- *Cleaning* adalah tahapan untuk pembersihan kata dengan menghilangkan tanda baca seperti titik (.), koma (,) dan yang lain dengan tujuan untuk mengurangi noise [10]
- *Tokenizing*, tahapan ini memiliki fungsi utama untuk membagi teks menjadi token (kata) terpisah untuk menetapkan indeks unik pada setiap kata [11]
- *Stopwords* untuk menghilangkan konjungsi, kata ganti orang, dan kata lain yang tidak memiliki arti penting [8]

- *Stemming* merupakan proses yang digunakan untuk mengidentifikasi kata-kata dengan arti yang serupa. Ini akan membantu mengurangi redundansi untuk mendapatkan bentuk dasar kata dengan menghilangkan *suffix* (akhiran) [12]

Pada tahapan *stemming* kami akan menggunakan beberapa jenis *stemmer* yakni *porter stemmer*, *snowball stemmer*, dan *lancaster stemmer*. Dengan tujuan untuk menemukan jenis *stemmer* mana yang dapat meningkatkan performa dari algoritma SVM pada analisis sentimen berbasis lexicon.

2.3. Klasifikasi Sentimen Dengan *Lexicon Resource*

Tahap ini bertujuan menentukan apakah kalimat atau dokumen berlabel positif, negatif atau netral. Sebelum memberi label pada kalimat dihitung nilai polaritasnya terlebih dahulu. *Lexicon resource* merupakan kamus data yang berisi kata-kata sentimen dan skor sentimennya. Dapat dikatakan ekstraksi fitur, dengan menggunakan sumber daya (*resource*) dari leksikon berarti mendapatkan skor sentimen pada setiap kata dalam kalimat yang dianalisa dan kemudian dikalkulasi keseluruhannya [13]. Dalam penelitian ini kami menggunakan *lexicon source* dari AFINN.

AFINN merupakan sumber daya leksikal yang dibangun menggunakan *Affective Norms for English Words* (ANEW) yang merupakan kamus bahasa Inggris yang menyediakan sejumlah kata dengan skor emosional. Dari ANEW kemudian AFINN dibangun dengan menambahkan kamus tambahan seperti bahasa gaul, emotikon dan beberapa istilah yang biasanya digunakan. Terdiri dari 2477 kata dalam bahasa Inggris dengan skor sentimennya. Sentimen positif memiliki skor dari 1-5, sedangkan negatif memiliki skor dari -1 sampai -5 [14]. Maka pada penelitian ini kami asumsikan bahwa skor 0 adalah netral.

Dalam penelitian ini, tahap klasifikasi akan dilakukan dengan tiga skenario menggunakan hasil stemming dari tiga jenis *stemmer* yang digunakan. Dengan tujuan untuk mengetahui hasil klasifikasi dari setiap jenis *stemmer* dengan AFINN lexicon.

2.4. Feature Extraction

Feature extraction menjadi aspek signifikan dari teteksi kesalahan informasi karena efektivitas algoritma machine learning terutama bergantung pada feature extraction [15]. Menurut [16] pendekatan feature extraction digunakan untuk mengekstraksi fitur berharga dari kumpulan data berdimensi tinggi.

Di sisi lain [17] mengatakan ini adalah proses kerja yang penting. Proses ngurangan input untuk menganalisis, mengolah, atau mengelola data yang paling banyak (feature selection). Maka beberapa fitur diekstraksi dari dataset. Fitur yang diekstraksi harus dalam format tertentu yang dapat langsung menjadi inputan untuk algoritma klasifikasi.

Teknik feature extraction diterapkan pada data training dan data testing. Pada data training untuk melatih model yang dipilih dan pada data testing saat klasifikasi dilakukan [18]. Dengan cara yang sama, penggunaan berbagai teknik feature extraction terbukti meningkatkan akurasi dari klasifikasi. Banyak metode feature extraction pada text mining, beberapa teknik yang sering digunakan adalah term frequency (TF), inverst document frequency (IDF), TF-IDF, word2vec dan doc2vec [18].

Pada penelitian ini feature extraction yang digunakan adalah TF-IDF. TF-IDF merupakan metode feature extaction yang paling populer dan sering digunakan. TF-IDF dinilai penting, apabila sebuah kata lebih sering muncul dalam sebuah dokumen maka nilai kontribusinya akan semakin besar, namun apabila kata tersebut sering muncul dalam beberapa dokumen maka akan memiliki kontribusi yang lebih kecil [11]. TF-IDF terdiri atas Term Frequency (TF) dan Inverse Document Frequency (IDF) [19]. TF-IDF dapat dirumuskan seperti persamaan berikut :

$$TF\ IDF(t_k, d_j) = TF(t_k, d_j) * IDF(t_k) \quad (1)$$

Dalam tahap ini juga akan dilakukan dengan tiga skenario menggunakan semua hasil klasifikasi sentimen dengan AFINN dari setiap jenis *stemmer* yang digunakan.

2.5. Implementasi Algoritma *Support Vector Machine* (SVM)

Langkah ini bertujuan untuk memecah data yang sudah diproses sebelumnya. Data dipecah menjadi data test dan data train dengan perbandingan 80 / 20 %. Proses ini dilakukan secara berulang berdasarkan hasil *stemmer* dengan AFINN *lexicon source*. Karena hasil antara *stemmer* berbeda satu sama lain.

Sebelumnya Algoritma *Support Vector Machine* (SVM) merupakan sebuah supervised classifier, diterapkan secara luas untuk menyelesaikan masalah regresi dan klasifikasi. Algoritma ini dirancang sebagai peningkatan untuk *support vector classifier*, yang telah dikenalkan sebagai peningkatan untuk *maximal margin classifier*, berurusan dengan data sederhana dan dapat dipisahkan secara linier. SVM dapat menangani kasus-kasus yang sering dihadapi, karena algoritma ini memetakan ruang fitur ke dalam ruang berdimensi lebih tinggi di mana titik data non-linier diubah menjadi titik yang dapat dipisahkan secara linier [20].

Kemudian *feature extraction* (TF-IDF) dilatih pada algoritma SVM. Proses ini juga akan dilakukan dengan tiga skenario secara berulang, berdasarkan hasil klasifikasi dari setiap *stemmer* yang diimplementasi kemudian performa dari model akan dievaluasi menggunakan *Confusion Matrix*.

2.6. Evaluasi Algoritma

Model dievaluasi menggunakan *confusion matrix*. *Confusion matrix* merupakan sebuah alat untuk mengevaluasi kinerja algoritma machine learning yang berisi informasi tentang klasifikasi dan prediksi actual. Ada empat indikator yang diukur di dalamnya: *accuracy*, *precision*, *recall* dan *F1-Score* [21]. Pada Gambar 2 adalah skenario dari *Confusion Matrix*.

		Predicted class	
		Class = True	Class = False
Actual class	Class = True	True positive	False Negative
	Class = False	False Positive	True Negative

Gambar 2. Skenario *Confusion Matrix*

Adapun perhitungan dari 4 indikator adalah sebagai berikut:

$$Accuracy: Accuracy = \frac{TP+TN}{(TP+TN+FP+FN)} \quad (2)$$

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

$$Precision: Precision = \frac{TP}{(TP+FP)} \quad (4)$$

$$F1-Score : F1 - Measure = 2 \times \frac{precision \times recall}{precision + recall} \quad (5)$$

Dari proses ini akan ditemukan *stemmer* mana yang mampu meningkatkan akurasi dari SVM pada analisis sentimen berdasarkan leksikon AFINN.

3. HASIL DAN PEMBAHASAN

Pada bagian ini akan dijelaskan hasil dari keseluruhan langkah yang dipaparkan di bagian METHOD. Berikut adalah hasil dan pembahasan dari penelitian ini.

3.1. Pengumpulan Data

Dari proses pengumpulan data didapatkan data sebesar 20000 data. Yang kemudian diproses untuk diambil variable penting dari data. Yakni variabel "reviewId", "content", dan "score". Data yang diperoleh berbahasa Indonesia maka kemudian data diterjemah ke dalam bahasa Inggris dengan menggunakan *google translate* untuk memudahkan proses-proses selanjutnya khususnya pada bagian *stemming* dan klasifikasi sentimen dengan AFINN *Lexicon*, kedua proses tersebut menggunakan sumber daya berbahasa Inggris. Pada Tabel 2 ditunjukkan hasil dari proses terjemah.

Tabel 2. Hasil Terjemahan Dataset

content	content_googletrans
Mudah dimengerti sukses selalu mobile jkn	Easy to understand, success is always mobile, if not
Sudah lama mendaftar ujung ujungnya tidak jadi,Klu buat apk sdh siap kapanpun	It's been a long time registering, the ends don't work, if you make an apk, it's ready at any time
Tolong permudah dalam penambahan anggota	Please make it easy to add members

3.2. Preprocessing

Dari keseluruhan tahap pada *preprocessing* yakni, *case folding*, *cleaning*, *tokenizing*, *stopword* dan *stemming*, kami akan berfokus memaparkan hasil dari proses *stemming* dari tiga jenis *stemmer*. Berikut adalah contoh hasil *stemming* dari *stemmer* yang digunakan yakni *porter stemmer* pada Tabel 3, *snowball stemmer* pada Tabel 4, dan *lancaster stemmer* pada Tabel 5.

Tabel 3. Hasil Porter Stemmer

content_googletrans	content_porter
Easy to understand, success is always mobile, if not	easi understand success alway mobil
It's been a long time registering, the ends don't work, if you make an apk, it's ready at any time	long time regist end dont work make apk readi time
Please make it easy to add members	pleas make easi add member

Tabel 4. Hasil Snowball Stemmer

content_googletrans	content_snowball
Easy to understand, success is always mobile, if not	easi understand success alway mobil
It's been a long time registering, the ends don't work, if you make an apk, it's ready at any time	long time regist end dont work make apk readi time
Please make it easy to add members	pleas make easi add member

Tabel 5. Hasil Lancaster Stemmer

content_googletrans	content_lancaster
Easy to understand, success is always mobile, if not	easy understand success alway mobil
It's been a long time registering, the ends don't work, if you make an apk, it's ready at any time	long tim reg end dont work mak apk ready tim
Please make it easy to add members	pleas mak easy ad memb

3.3. Klasifikasi Sentimen Dengan Lexicon Resource

Setelah dilakukan proses *stemming* dengan semua jenis *stemmer* yang digunakan pada penelitian ini, proses berikutnya adalah klasifikasi sentimen menggunakan *lexicon resource*. Seperti yang telah dijelaskan pada bagian 2.3 *lexicon resource* yang digunakan pada penelitian ini adalah *AFINN resource*. Penelitian ini menggunakan tiga class sentimen yakni *positive*, *negative* dan *neutral*. Pada Tabel 7 ditampilkan jumlah setiap class dari masing-masing *stemmer* yang digunakan dengan *AFINN lexicon resource*.

Tabel 6. Hasil Klasifikasi Content Stemmer + AFINN Resource

Class	TABLE I. HASIL KLASIFIKASI CONTENT STEMMER + AFINN RESOURCE		
	Porter + Afinn	Snowball + Afinn	Lancaster + Afinn
Positive	10190	10256	9293
Negative	7282	7226	7825
Neutral	2528	2518	2882

3.4. Feature Extraction

Feature extraction menjadi aspek signifikan dari teteksi kesalahan informasi karena efektivitas algoritma machine learning terutama bergantung pada feature extraction [15]. Menurut [16] pendekatan feature extraction digunakan untuk mengekstraksi fitur berharga dari kumpulan data berdimensi tinggi. Seperti yang dijelaskan pada bagian metode, penelitian ini menggunakan TF-IDF khususnya menggunakan TfidfVectorizer pada library sklearn. Cara kerjanya adalah menghitung bobot pada setiap kata dan dapat menyajikan nilai atau skor frekuensi setiap kata dengan tingkat frekuensi kemunculan paling tinggi pada dokumen dalam kasus ini content pada setiap hasil stemming.

Berikut merupakan hasil kemunculan kata paling tinggi dari proses feature extraction menggunakan TfidfVectorizer pada setiap jenis stemmer yang digunakan.

Tabel 7. Kata paling penting pada *Porter Stemmer*

TABLE II. KATA PALING PENTING PADA PORTER STEMMER

kata	nilai TF-IDF
help	2145.9881
good	1917.6983
greate	1169.7887

Tabel 8. Kata paling penting pada *Snowball Stemmer*

TABLE III. KATA PALING PENTING PADA SNOWBALL STEMMER

kata	nilai TF-IDF
help	2149.2451
good	1917.8993
greate	1179.9658

Tabel 9. Kata paling penting pada *Lancaster Stemmer*

TABLE IV. KATA PALING PENTING PADA LANCASTER STEMMER

kata	nilai TF-IDF
help	2158.8106
good	1922.6153
gre	1184.7814

3.4. Implementasi Algoritma Support Vector Machine (SVM)

Setelah melakukan preprocessing, klasifikasi sentimen dengan AFINN lexicon dan feature extaction dengan TF-IDF, proses selanjutnya adalah implementasi algoritma SVM. Proses ini menggunakan menggunakan hasil dari setiap stemmer yang digunakan. Langkah sebelumnya adalah menentukan variabel X dan Y, variabel X adalah hasil stemming (dokumen/conten) dan Y adalah label dari hasil polaritas AFINN lexicon.

Selanjutnya dataset dibagi menjadi data train dan data test. Dengan rasio perbandingan yang digunakan pada split data yakni 80 : 20. Hasil klasifikasi dari setiap jenis stemming berbeda satu sama lain, maka jumlah data-split yang dihasilkan juga berbeda. Berikut adalah hasil dari data-split pada masing-masing jenis stemming.

Tabel 10. Rasio Jumlah Data-split *Porter Stemmer*

TABLE V. RASIO JUMLAH DATA-SPLIT PORTER STEMMER

Class	Testing Set	Training Set
	Positive	2038
Negative	506	2022
Neutral	1456	5826

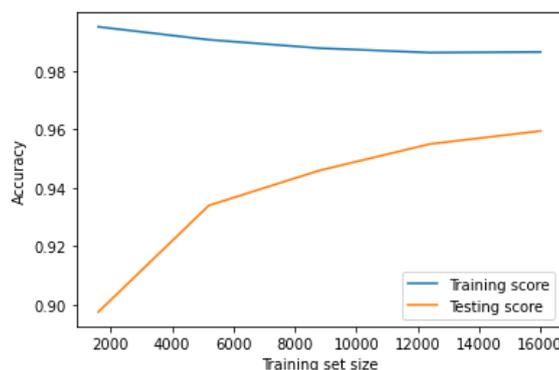
Tabel 10. Rasio Jumlah Data-split *Snowball Stemmer*

Class	TABLE I. RASIO JUMLAH DATA-SPLIT <i>SNOWBALL STEMMER</i>	
	Testing Set	Training Set
Positive	2051	8205
Negative	504	2014
Neutral	1445	5781

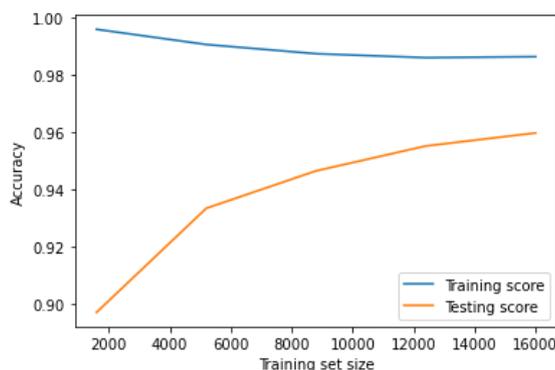
Tabel 10. Rasio Jumlah Data-split *Lancaster Stemmer*

Class	TABLE I. RASIO JUMLAH DATA-SPLIT <i>LANCASTER STEMMER</i>	
	Testing Set	Training Set
Positive	1859	7434
Negative	576	2306
Neutral	1565	6260

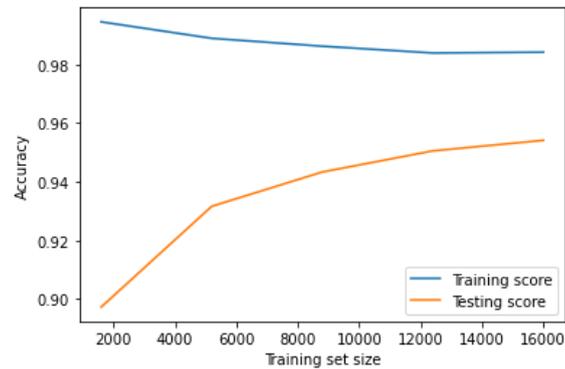
Kemudian data training dan data testing diimplementasikan pada algoritma SVM. Pada penelitian ini kami menggunakan Linear SVC pada library sklearn. Seperti proses yang lain implementasi SVM dilakukan berulang pada setiap hasil stemmer. Berikut adalah hasil dari implementasi algoritma SVM pada setiap jenis stemmer yang ditunjukkan dalam bentuk learning curve.



Gambar 3. Learning Curve SMV + Porter Stemmer



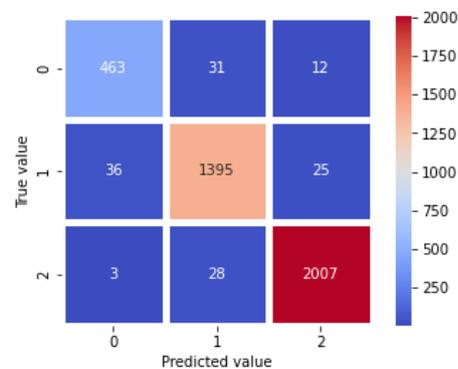
Gambar 4. Learning Curve SMV + Snowball Stemmer



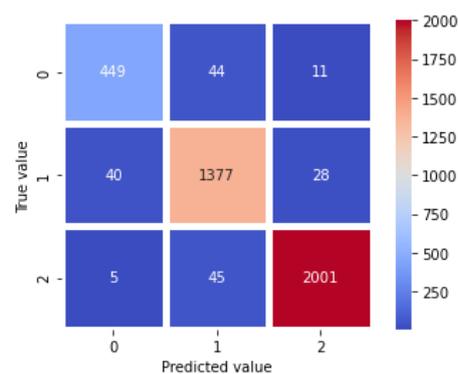
Gambar 5. Learning Curve SMV + Lancaster Stemmer

3.5. Evaluasi Algoritma

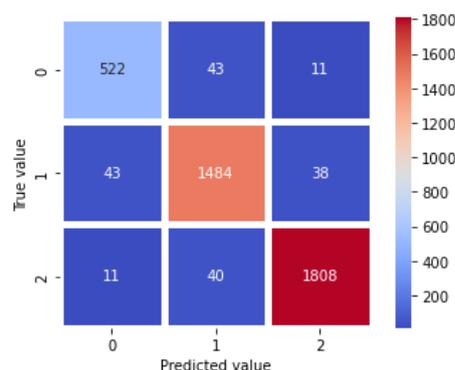
Evaluasi algoritma merupakan tahap terakhir setelah proses implementasi model algoritma SVM. Tahap ini dilakukan untuk mengukur performa terhadap model SVM dengan menggunakan confusion matrix seperti yang dipaparkan pada bagian metode. Tahap ini juga dilakukan pada tiga skenario berdasarkan jenis stemmer. Berikut adalah hasil confusion matrix dari setiap skenario yang dilakukan.



Gambar 6. Confusion Matrix SMV + Porter Stemmer



Gambar 7. Confusion Matrix SMV + Snowball Stemmer



Gambar 8. Confussion Matrix SMV + Lancaster Stemmer

Pada Tabel 11 akan dipaparkan nilai Accuracy, Precision, Recall dan F1-Score dari tiga skenario yang dilakukan pada penelitian ini.

Tabel 11. Nilai Accuracy, Precision, Recall dan F1-Score

Skenario	TABLE I. NILAI ACCURACY, PRECISION, RECALL DAN F1-SCORE			
	Accuracy	Precision	Recall	F1-Score
SVM + Porter Stemmer	95,63 %	95,62 %	95,63 %	95,61 %
SVM + Snowball Stemmer	95,67 %	95,68 %	95,67 %	95,67 %
SVM + Lancaster Stemmer	95,35 %	95,35 %	95,35 %	95,35 %

Dari semua skenario yang dilakukan, skenario antara SVM dan Snowball Stemmer mendapatkan nilai Accuracy, Precision, Recall dan F1-Score paling tinggi dari dua skenario lainnya.

4. KESIMPULAN

Pada penelitian tentang analisis sentimen kali ini yang menggunakan data yang didapatkan dari hasil crawling data google play dengan topik “Mobile JKN” dengan data sebanyak 20000 baris data. Dan berdasarkan tujuan dari penelitian ini yakni mengeksplorasi hasil klasifikasi berbasis lexicon berdasarkan stemmer. Serta berdasarkan hasil dari tiga skenario yang dilakukan dengan tiga jenis stemmer yang berbeda dapat disimpulkan bahwa jenis stemmer dapat mempengaruhi performa model algoritma yang digunakan dalam hal ini SVM. Dari ketiga skenario yang dilakukan, skenario dari gabungan SVM dan Snowball stemmer mendapatkan nilai Accuracy, Precision, Recall dan F1-Score paling tinggi dari dua skenario lainnya. Yakni dengan nilai Accuracy sebesar **95,67 %**, Precision sebesar **95,68 %**, Recall sebesar **95,67 %** dan F1-Score sebesar **95,67 %**. Penelitian ini hanya menggunakan satu jenis feature extraction yakni TF-IDF. Maka untuk penelitian selanjutnya akan dilakukan eksperimen menggunakan gabungan antara jenis stemmer dan jenis feature extraction yang berbeda pada klasifikasi sentimen berbasis lexicon. Serta lebih baik menggunakan dataset dan algoritma machine learning yang berbeda untuk menghasilkan eksperimen yang lebih kompleks dan teruji.

UCAPAN TERIMA KASIH

Kami atas nama penulis mengucapkan terima kasih kepada pembimbing kami, yakni Bapak Dr. Andi Sunyoto, M. Kom. dan Kusnawi, S.Kom., M.Eng. Karena sudah membimbing kami dalam menyelesaikan penelitian ini dengan memberikan saran, pendapat serta memotivasi kami selama penelitian ini dikerjakan.

REFERENSI

- [1] K. X. Han, W. Chien, C. C. Chiu, and Y. T. Cheng, “Application of support vector machine (SVM) in the sentiment analysis of twitter dataset,” *Appl. Sci.*, vol. 10, no. 3, 2020, doi: 10.3390/app10031125.
- [2] V. Nurcahyawati and Z. Mustafa, “Improving sentiment reviews classification performance using support vector machine-fuzzy matching algorithm,” *Bull. Electr. Eng. Informatics*, vol. 12, no. 3, pp. 1817–1824, 2023, doi: 10.11591/eei.v12i3.4830.
- [3] K. Tamara and N. Milićević, “Comparing Sentiment Analysis and Document Representation Methods of Amazon Reviews,” *SISY 2018 - IEEE 16th Int. Symp. Intell. Syst. Informatics, Proc.*, pp. 283–288, 2018, doi: 10.1109/SISY.2018.8524814.
- [4] H. Zou, X. Tang, B. Xie, and B. Liu, “Sentiment classification using machine learning techniques with syntax features,” *Proc. - 2015 Int. Conf. Comput. Sci. Comput. Intell. CSCI 2015*, pp. 175–176, 2016, doi: 10.1109/CSCI.2015.44.
- [5] D. Dangi, A. Bhagat, and D. K. Dixit, “Sentiment analysis of social media data based on chaotic coyote optimization algorithm based time weight-AdaBoost support vector machine approach,” *Concurrency and Computation: Practice and Experience*, vol. 34, no. 3. 2022. doi: 10.1002/cpe.6581.

- [6] A. Muhammad, S. Abdullah, and N. S. Sani, "Optimization of sentiment analysis using teaching-learning based algorithm," *Comput. Mater. Contin.*, vol. 69, no. 2, pp. 1783–1799, 2021, doi: 10.32604/cmc.2021.018593.
- [7] C. Shofiya and S. Abidi, "Sentiment analysis on covid-19-related social distancing in Canada using twitter data," *Int. J. Environ. Res. Public Health*, vol. 18, no. 11, 2021, doi: 10.3390/ijerph18115993.
- [8] F. Resyanto, Y. Sibaroni, and A. Romadhony, "Choosing The Most Optimum Text Preprocessing Method for Sentiment Analysis: Case:iPhone Tweets," *Proc. 2019 4th Int. Conf. Informatics Comput. ICIC 2019*, pp. 2–6, 2019, doi: 10.1109/ICIC47613.2019.8985943.
- [9] S. Rani, N. Singh Gill, and P. Gulia, "Analyzing impact of number of features on efficiency of hybrid model of lexicon and stack based ensemble classifier for twitter sentiment analysis using WEKA tool," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 22, no. 2, p. 1041, 2021, doi: 10.11591/ijeecs.v22.i2.pp1041-1051.
- [10] Y. Handayani, A. R. Hakim, and Muljono, "Sentiment analysis of Bank BNI user comments using the support vector machine method," *Proc. - 2020 Int. Semin. Appl. Technol. Inf. Commun. IT Challenges Sustain. Scalability, Secur. Age Digit. Disruption, iSemantic 2020*, pp. 202–207, 2020, doi: 10.1109/iSemantic50169.2020.9234230.
- [11] B. AlBadani, R. Shi, and J. Dong, "A Novel Machine Learning Approach for Sentiment Analysis on Twitter Incorporating the Universal Language Model Fine-Tuning and SVM," *Appl. Syst. Innov.*, vol. 5, no. 1, 2022, doi: 10.3390/asi5010013.
- [12] N. V. Babu and E. G. M. Kanaga, "Sentiment Analysis in Social Media Data for Depression Detection Using Artificial Intelligence: A Review," *SN Comput. Sci.*, vol. 3, no. 1, pp. 1–20, 2022, doi: 10.1007/s42979-021-00958-1.
- [13] B. T. Pratama, E. Utami, and A. Sunyoto, "An optimization of a lexicon based sentiment analysis method on Indonesian app review," *2019 4th Int. Conf. Inf. Technol. Inf. Syst. Electr. Eng. ICITISEE 2019*, pp. 341–346, 2019, doi: 10.1109/ICITISEE48480.2019.9003900.
- [14] B. T. Pratama, E. Utami, and A. Sunyoto, "A comparison of the use of several different resources on lexicon based Indonesian sentiment analysis on app review dataset," *Proceeding - 2019 Int. Conf. Artif. Intell. Inf. Technol. ICAIIT 2019*, pp. 282–287, 2019, doi: 10.1109/ICAIIIT.2019.8834531.
- [15] Y. Barve, J. R. Saini, K. Pal, and K. Kotecha, "A Novel Evolving Sentimental Bag-of-Words Approach for Feature Extraction to Detect Misinformation," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 4, pp. 266–275, 2022, doi: 10.14569/IJACSA.2022.0130431.
- [16] P. Verma, A. Dumka, A. Bhardwaj, and A. Ashok, "Product Review-Based Customer Sentiment Analysis Using an Ensemble of mRMR and Forest Optimization Algorithm (FOA)," *Int. J. Appl. Metaheuristic Comput.*, vol. 13, no. 1, pp. 1–21, 2022, doi: 10.4018/ijamc.2022010107.
- [17] M. B. Rissan and R. F. Hassan, "Naïve-Bayes family for sentiment analysis during COVID-19 pandemic and classification tweets," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 28, no. 1, pp. 375–383, 2022, doi: 10.11591/ijeecs.v28.i1.pp375-383.
- [18] F. Rustam, I. Ashraf, A. Mehmood, S. Ullah, and G. S. Choi, "Tweets classification on the base of sentiments for US airline companies," *Entropy*, vol. 21, no. 11, pp. 1–22, 2019, doi: 10.3390/e21111078.
- [19] C. H. Yutika, A. Adiwijaya, and S. Al Faraby, "Analisis Sentimen Berbasis Aspek pada Review Female Daily Menggunakan TF-IDF dan Naïve Bayes," *J. Media Inform. Budidarma*, vol. 5, no. 2, p. 422, 2021, doi: 10.30865/mib.v5i2.2845.
- [20] R. Obiedat *et al.*, "Sentiment Analysis of Customers' Reviews Using a Hybrid Evolutionary SVM-Based Approach in an Imbalanced Data Distribution," *IEEE Access*, vol. 10, pp. 22260–22273, 2022, doi: 10.1109/ACCESS.2022.3149482.
- [21] P. H. Prastyo, A. S. Sumi, A. W. Dian, and A. E. Permanasari, "Tweets Responding to the Indonesian Government's Handling of COVID-19: Sentiment Analysis Using SVM with Normalized Poly Kernel," *J. Inf. Syst. Eng. Bus. Intell.*, vol. 6, no. 2, p. 112, 2020, doi: 10.20473/jisebi.6.2.112-122.